

Dynamic Framed Slotted ALOHA 기반 RFID 태그 충돌방지 알고리즘 개선

Improvement of RFID Tag Anti-Collision Algorithm Based on the Dynamic Framed Slotted ALOHA

박노경, 윤형기[†], 문대철

(Nho-Kyung Park, Hyoung-Kie Yun[†], and Dai-Tchul Moon)

호서대학교 정보통신공학과

(접수일자: 2013년 10월 11일; 수정일자: 2013년 12월 19일; 채택일자: 2013년 12월 24일)

초 록: 본 논문에서는 DFSA (Dynamic Framed Slotted ALOHA) 기반 충돌방지 알고리즘에 대해 논하고, 개선된 태그 충돌 방지 알고리즘을 제안한다. 제안된 방법은 기대 값을 이용하는 방법으로서 한 번의 연산을 통해 측정값에 근접한 기대 값을 갖는 태그 수를 추정하므로 속도가 빠르다는 장점이 있으며 태그 수에 따른 빈 슬롯 개수 및 충돌 슬롯의 개수를 이용하여 기대 값을 구한 후 실제 태그 개수를 추정할 수 있다. 제안된 알고리즘은 시뮬레이션을 통해 기존 방법과 비교 및 분석하였다. 그 결과, 전체 태그에 대하여 평균 18.8 라운드에 모든 태그를 식별하였다. 태그의 수가 1000개 이하일 경우 평균 18.2 라운드에 모든 태그를 식별하였으나 태그 수가 1000개 이상인 경우 19.2 라운드에 태그를 식별하였다. 제안된 방식 및 기존 방식을 비교하였을 때 평균 태그 수에 따른 라운드 수가 상쇄 기법이 적용된 DFSA 방식보다 3.1 %, DFSA 방식보다 10.1 %, FSA (Framed Slotted ALOHA) 방식보다 37.5 % 가량 감소하여 처리 속도가 향상됨을 확인할 수 있었다.

핵심용어: Slotted ALOHA, DFSA, RFID, 충돌 방지 알고리즘, 태그 추정

ABSTRACT: In this paper, a DFSA (Dynamic Framed Slotted ALOHA) based anti-collision algorithm is described and a performance improved algorithm of DFSA and FSA is proposed. The proposed method makes use of expected values and has merits in operation speed by estimating closest value of the number of tags on a single operation. The algorithm compares and analyses number of empty slots and collide slots in accordance with the number of tags, which enables estimation of actual number of tags. we simulated the proposed algorithm and compared it with conventional methods. Results show that our method needs average 18.8 rounds to identify all tags. In case of less than 1000 tags, an average of 18.2 rounds is necessary and an average of 19.2 rounds is needed to identify all tags for more than 1000 tags. The proposed algorithm improves processing speed by 3.1 % comparing to DFSA with cancellation strategy, 10.1 % comparing to DFSA, 37.5 % comparing to FSA(Framed Slotted ALOHA).

Keywords: Slotted ALOHA, DFSA, RFID, Anti-collision, Estimation of tags

PACS numbers: 43.60. Dh

1. 서 론

RFID(Radio Frequency Identification)는 IC 칩에 내장된 정보를 무선주파수를 이용하여 비 접촉방식으로

읽어내는 기술로서 대상을 자동으로 식별할 수 있는 기술이다. 일반적으로 안테나와 칩으로 구성된 RFID 태그에 정보를 저장하여 적용 대상에 부착한 후 판독기에 해당하는 RFID 리더를 통해 정보를 인식한다.^[1]

RFID 시스템의 구성은 태그와 통신하기 위한 리

[†]Corresponding author: Hyoung-Kie Yun (mist098@naver.com)
Department of Information & Communication Engineering, Hoseo University, 20 Hoseo-Ro 79Beon-Gil, Baebang-Eup, 336-795, Republic of Korea
(Tel: 82-41-540-5378, Fax: 82-41-540-5693)

더, 물체에 부착되어 물체의 여러 정보를 저장하고 있는 태그, 시스템을 전체적으로 제어하고 수신한 태그의 정보를 처리하여 리더에게 전송하는 백-엔드 서버(back-end server), 리더와 연결되어 리더가 전송받은 정보를 처리해 백-엔드 서버에게 전송하는 리더 서브시스템 및 안테나로 구성된다. 일반적으로 UHF 대역(Ultra-High Frequency, 900 MHz 대역) RFID 시스템에서는 저가인 수동형 태그가 사용되나 그 능력이 매우 제한적이어서 다른 태그들과 통신을 할 수 없고 리더로만 통신이 가능하다. 리더는 무선 채널을 통하여 각각의 태그들과 통신을 하는데, 모든 태그들이 리더가 보낸 신호를 동시에 받게 되며 리더에서 요청하는 전송 요구에 응답을 한다. 이 때 하나의 리더가 동시에 응답한 여러 개의 태그를 인식해야 하는 문제가 발생하는데 이를 태그 충돌이라고 하며, 이로 인한 태그 인식 속도 저하로 데이터 처리 속도가 감소하게 된다. 이러한 문제로 여러 개의 태그를 인식하기 위해서는 태그 충돌 문제를 해결하는 충돌방지 알고리즘이 필요하다.^[2]

기존 FSA 및 DFSA 충돌방지 알고리즘의 경우, 비메모리 시스템에 적합한 방식으로서 빠른 태그 개수 추정이 가능하나 태그 수가 늘어날수록 측정 값과 기대 값 사이의 오차가 크다는 단점이 있으며, 테이블을 저장할 메모리가 필요하므로 메모리에 제약이 있는 시스템에서는 사용이 어렵다는 단점이 있다. 이러한 단점을 보완하기 위해 본 논문에서는 최적의 프레임 크기만 적용한 테이블만 적용하여 적은 용량의 메모리만을 사용한 태그 수 추정을 통해 기존 DFSA 방식을 기반으로 개선된 충돌 방지 알고리즘에 대해 제안하였다.

II. Slotted ALOHA 기반 충돌방지 알고리즘

Slotted ALOHA 방식은 태그가 응답하는 시간을 고정된 몇 개의 슬롯으로 나누고 태그들이 각자 선택한 슬롯에 데이터를 전송하는 방식이다. 태그가 자신의 데이터를 리더에 전송할 때 전송하는 시간 간격으로 슬롯 내에 데이터를 전송 시 데이터끼리 충돌이 발생되지 않을 때에만 태그를 인식할 수 있다.

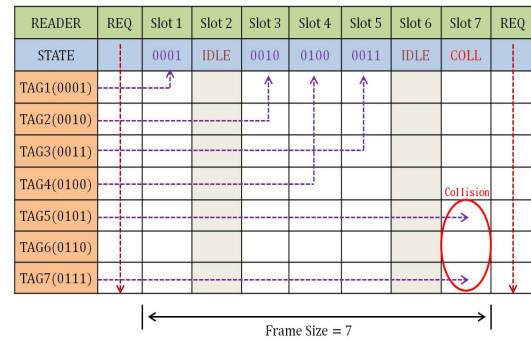


Fig. 1. Recognition operations of FSA anti-collision algorithm.

일반적으로 slotted ALOHA 알고리즘은 슬롯의 사용 방식과 사용되는 프레임의 크기에 따라 FSA 방식, DFSA 방식 등으로 구분할 수 있다.

먼저 FSA 방식이란 리더에서 명령을 내보내고 다음 명령 전송까지 데이터가 전달되는 프레임의 크기가 고정되어 있는 slotted ALOHA 방식이다. FSA 방식에서 리더는 태그들에게 고정된 슬롯의 수와 슬롯 선택에 대한 정보를 제공하는 역할을 한다. 리더가 ID 전송 요구를 하면 태그는 자신이 원하는 슬롯을 선택하여 데이터를 전송하며, 이 때 전체 슬롯에 대해서 태그가 응답하는 과정을 한 라운드로 정의할 수 있다. Fig. 1은 리더가 7개의 슬롯을 가질 때 리더 명령에 대한 각 태그들의 응답 상태를 보여준다. 리더는 하나의 데이터만 인식된 슬롯만을 성공적으로 인식한다. 그러나 Slot 2와 Slot 6은 자신의 ID를 전송한 태그가 없으므로 IDLE 상태가 된다. Slot 7에서는 TAG 5와 TAG 7이 ID를 동시에 전송하였기 때문에 전송된 데이터 간 충돌이 발생하여 태그들이 전송한 데이터를 잃게 된다. 리더는 충돌이 발생한 TAG 5와 TAG 7에 대해 다시 ID 전송요구를 발생시키며, 첫 번째 ID 전송 요구를 했을 때와 같은 방식으로 태그 인식을 수행한다.^[3,4]

두 번째로 DFSA 방식은 프레임 내에서 태그의 ID 전송이 균일하게 분포한다고 가정하고 태그 수를 추정하여 프레임의 크기를 늘려주는 방식이다. 그렇기 때문에 프레임의 크기가 지속적으로 늘어나 많은 양의 태그의 인식이 가능하다. 처리율이 최대일 때 충돌이 발생한 슬롯 수의 비율 C_{rate} 를 구하고 이를 이용하여 하나의 슬롯에서 충돌이 발생한 태그의 수

C_{tags} 를 구하면 식(1)과 같다. 이 식에서 P_{Scoll} 은 태그가 충돌이 발생할 확률이며 P_{Ssucc} 는 태그가 성공적으로 ID를 전송할 확률을 나타낸다.^[3]

$$C_{rate} = \frac{P_{Scoll}}{1 - P_{Ssucc}} = \lim_{n \rightarrow \infty} \frac{1 - (1 - \frac{1}{n})^n - (1 - \frac{1}{n})^{n-1}}{1 - (1 - \frac{1}{n})^{n-1}} = 0.4180, \quad (1)$$

$$C_{tags} = \frac{1}{C_{rate}} = 2.3922.$$

이 때 한 라운드 이후 한 프레임 내에서 충돌이 발생한 슬롯의 수를 a_c 이라고 하면 추정된 태그의 수 n_{est} 는 식(2)가 된다.

$$n_{est} = \lceil 2.3922 \times a_c \rceil. \quad (2)$$

이 식에서 $\lceil k \rceil$ 는 k 보다 작거나 작은 최대 정수를 의미한다. 여기에 n 을 태그의 수라 하면 최적화 된 프레임 크기는 식(3)과 같다.

$$L_{optimal} = n. \quad (3)$$

위의 수식을 이용하면 최적화 된 프레임 크기는 추정된 태그의 수와 같다는 것을 알 수 있다. 즉, 연산된 프레임의 크기를 이용하여 매 라운드 마다 프레임의 크기를 변경하여 태그의 수를 추정하는 방식이다.

III. DFSA 기반 태그 개수 추정 방법

태그의 개수를 추정하는 방법은 크게 두 가지 방법이 있으며, 기대 값을 이용하는 방법과 측정된 파라미터를 이용하는 방법으로 구분할 수 있다.

3.1 기대 값을 이용한 태그 개수 추정 방법

일정한 슬롯 수에서 태그 개수를 알고 있다고 가정할 때, 한 라운드 동안 발생하는 빈 슬롯 수, 인식 슬롯 수, 충돌 슬롯 수에 대한 기대 값을 구할 수 있다. 한 슬롯에 r 개의 태그가 응답할 확률은 이항분포

를 따르기 때문에 식(4)와 같이 표현된다.^[2]

$$B(r) = \binom{n}{r} \left(\frac{1}{N}\right)^r \left(1 - \frac{1}{N}\right)^{n-r}, \quad (4)$$

여기서 N 은 슬롯의 수, n 은 태그의 수이다. a_i 는 빈 슬롯 수의 기대 값, a_s 은 식별 슬롯 수의 기대 값, a_c 는 충돌이 발생한 슬롯 수의 기대 값을 나타내며 식(1)을 이용하여 다음과 같은 식을 얻을 수 있다.

$$a_i = N \times B(0) = N \left(1 - \frac{1}{N}\right)^n, \quad (5)$$

$$a_s = N \times B(1) = N \left(1 - \frac{1}{N}\right)^{n-1}, \quad (6)$$

$$a_c = N - N \left(1 - \frac{1}{N}\right)^n - N \left(1 - \frac{1}{N}\right)^{n-1}. \quad (7)$$

측정된 파라미터 값에 가장 가까운 기대 값을 가지는 태그의 수를 찾는 방법은 두 가지로 나뉜다. 첫 번째는 태그 수에 따른 파라미터의 기대 값을 테이블로 만들어 태그 수를 추정하는 방법으로서 측정값에 가까운 기대 값을 갖는 태그 수를 직접 테이블을 이용해 찾아내기 때문에 추정 태그 수를 빠르게 찾아낼 수 있지만, 테이블을 저장할 메모리가 필요하므로 비메모리 시스템에서는 사용할 수 없다.

두 번째는 다양한 탐색 알고리즘을 통해 태그 수를 추정하는 방법이다. 이 방법은 태그 개수를 추정하기 위해서 여러 번의 연산을 통해 기대 값을 구해야 하기에 메모리에 데이터가 누적 될수록 연산 속도가 느려지는 단점이 있다. 기대 값을 이용한 태그 수 추정 방법으로는 Vogt^[5]가 제안한 태그 수 추정 방식이 있다.

$$e_{vd} = \min_n \left| \begin{pmatrix} a_i \\ a_s \\ a_c \end{pmatrix} - \begin{pmatrix} C_i \\ C_s \\ C_c \end{pmatrix} \right|, \quad (8)$$

여기서 e_{vd} 는 추정된 태그 수이며 C_i, C_s, C_c 는 한 라운드 동안 측정되는 빈 슬롯 수, 인식 슬롯 수, 충돌

슬롯 수에 대한 기대 값을 나타낸다. 식(8)을 이용해 세 파라미터에 대한 기대 값과 측정 값의 오차를 최소로 만드는 태그 개수를 찾아낸다. 그러나 세 가지 파라미터에 대한 기대 값을 모두 구해야 하므로 필요한 연산량이 많은 단점이 있다.

3.2 단일 연산을 통한 태그 개수 추정 방법

단일 연산을 통해 태그 개수를 추정하는 방법은 한번의 연산으로 태그 개수를 추정하므로 시스템의 메모리에 대한 제약이 없고 추정속도도 빠르다. 첫 번째 방법으로 태그 개수의 최소치를 추정하는 방법이 있다. 태그 개수의 e_{min} 은 충돌 슬롯에는 태그가 항상 두 개 있다는 가정 하에 계산된 것이므로 추정된 태그 개수는 최소치를 의미한다. 이에 $C_c = N - C_i + C_s$ 를 대입하여 정리하면 다음 식(9)를 얻을 수 있다.^[3]

$$e_{min} = C_s + 2C_c = 2N - 2C_i - C_s \leq 2N. \quad (9)$$

두 번째 방법은 충돌이 발생한 슬롯에 응답하는 평균 태그 개수가 식(1)에서 계산된 값인 2.3922개임을 이용해 실제 태그 개수를 추정하는 것으로, 추정되는 태그 수는 다음 식(10)과 같이 된다.

$$n = C_s + 2.3922 \times C_c. \quad (10)$$

세 번째 방법은 태그 개수에 따른 빈 슬롯 개수, 식별 슬롯 개수의 기대 값을 이용한 방법이다. 식(5)를 식(6)으로 나누어 정리하고, 체비셰프 부등식에 근거하여 기대 값인 a_i, a_s 대신 측정값 C_i, C_s 를 대입하여 식(11)을 얻을 수 있다.

$$n = (N-1) \frac{a_i}{a_s} = (N-1) \frac{C_i}{C_s}. \quad (11)$$

대부분의 태그 개수 추정 방법은 식별 슬롯 개수와 빈 슬롯 개수의 비를 이용하여 태그 개수를 추정하므로 추정 값이 측정 값 오차에 민감하게 되는 문제가 있다. 이는 식별 슬롯 개수가 늘어나면 빈 슬롯의 개수가 감소하므로 둘 사이의 비를 이용하게 되

면 추정 값의 정확도가 측정 값 오차에 크게 영향을 받게 되기 때문이다.

IV. 제안된 DFSA 기반 태그 충돌방지 알고리즘

기존의 태그 개수 추정 방법들은 테이블 없이 추정 값을 구하기에 속도가 저하된다는 단점이 있었다. 이를 극복하기 위한 방법으로 제시된 단일 연산을 통한 태그 개수 추정 방법들은 추정의 정확도가 떨어지는 문제가 있었다. 따라서 본 논문에서는 이러한 문제를 개선하기 위해 단일 연산을 통해 태그 개수의 고속 추정이 가능하면서도 테이블을 이용하여 보다 정확도가 높은 태그 개수 추정 방법을 제안한다.

제안된 태그 개수 추정의 첫 번째 방법은 한 라운드의 태그 인식 과정을 거쳐 얻을 수 있는 파라미터 중 빈 슬롯 개수를 이용한다. 한 슬롯에 태그가 응답할 확률은 이항분포 형태를 가짐을 이용해 빈 슬롯 개수의 기대 값을 얻는다. 빈 슬롯 개수의 기대 값에 대한 식(5)를 보면 간단한 지수 함수의 형태로 나타나므로 이를 정리하여 태그 수 n 에 대한 식으로 나타내고, 여기서 체비셰프 부등식에 근거하여 기대 값인 a_0 대신 측정값 C_0 를 넣음으로써 식(12)와 같이 정리된다.^[3,6]

$$n = \frac{\ln\left(\frac{a_i}{N}\right)}{\ln\left(1 - \frac{1}{N}\right)} = \frac{\ln\left(\frac{C_i}{N}\right)}{\ln\left(1 - \frac{1}{N}\right)}. \quad (12)$$

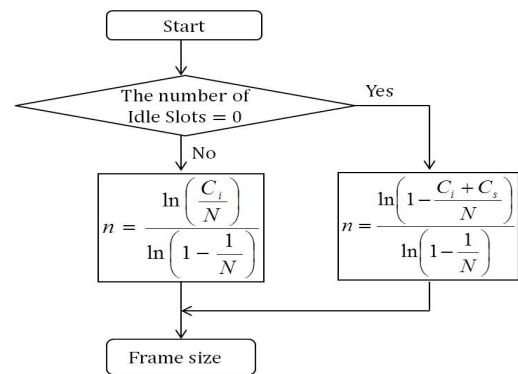


Fig. 2. Flow chart for proposed algorithm based on DFSA using estimation the number of tags.

두 번째 방법은 빈 슬롯 개수가 적어도 1개 이상 존재한다면 사용할 수 있는 방법이다. 이항 분포를 이용하여 식(5)의 식을 이용하여 유도한다. 여기에서 충돌 슬롯의 개수와 빈 슬롯의 개수가 같다면 식(13)과 같이 정리할 수 있다.^[7]

$$a_c = N \left(1 - \frac{1}{N} \right)^n \tag{13}$$

식(13)에서 $a_c = N - a_i - a_s$ 를 대입하면 식(14)를 얻을 수 있으며, 이를 자연로그로 정리하여 기대 값 a_i, a_s 대신 측정 값 C_i, C_s 을 대입하면 식(15)와 같이 정리되며 빈 슬롯이 존재할 때 태그 개수의 추정값을 얻을 수 있다. 단, 식(15)는 N 이 1이 아니거나 $C_i + C_s \neq N$ 일 경우에 한해서만 적용 가능하다.

$$N - a_i - a_s = N \left(1 - \frac{1}{N} \right)^n, \tag{14}$$

$$n = \frac{\ln \left(1 - \frac{a_i + a_s}{N} \right)}{\ln \left(1 - \frac{1}{N} \right)} = \frac{\ln \left(1 - \frac{C_i + C_s}{N} \right)}{\ln \left(1 - \frac{1}{N} \right)}. \tag{15}$$

제안된 방법은 한 번의 연산을 통해 측정값과 가장 가까운 기대 값을 가지는 태그 개수를 추정하므로 추정 속도가 빠르며 적용 범위에 제한이 없고 하나의 파라미터만을 사용하므로 추정의 정확도가 높다는 장점이 있다.

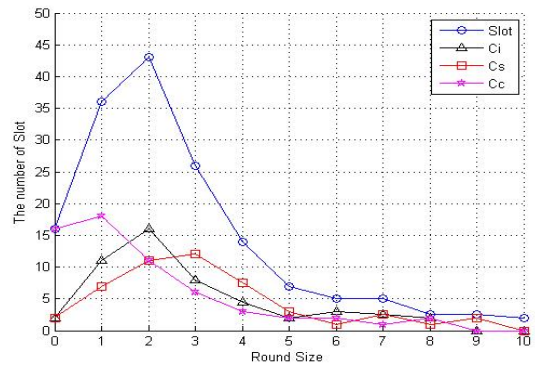
IV. 시뮬레이션 결과 및 성능 평가

제안된 방법에서 태그 수의 추정치는 태그 수에 따른 최적의 프레임을 Table 1에서 선택하여 그 크기를 변화시켜 준다.

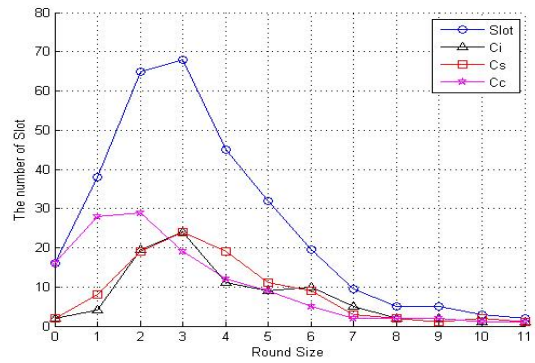
Fig. 3은 태그의 숫자가 50개, 100개, 500개, 1000개

Table 1. Framed size table.

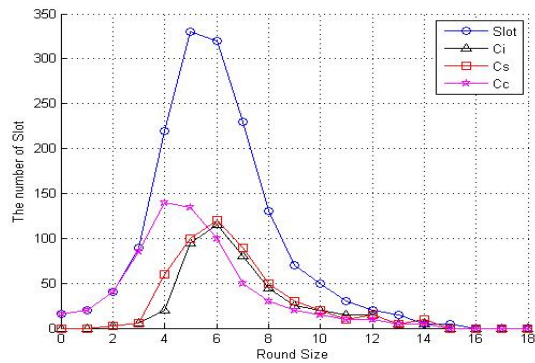
| N Slot | 1 | 4 | 8 | 16 | 32 | 64 | 128 | 256 |
|--------|---|---|---|----|----|----|-----|-----|
| Low | - | - | - | 1 | 10 | 17 | 51 | 112 |
| High | - | - | - | 9 | 27 | 56 | 129 | ∞ |



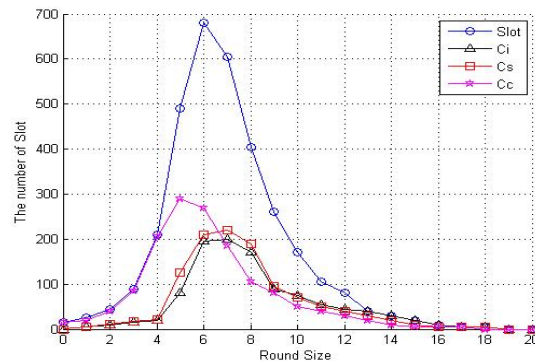
(a) the number of tags 50



(b) the number of tags 100



(c) the number of tags 500



(d) the number of tags 1000

Fig. 3. Compare with changing the number of slots by rounds size on proposed method.

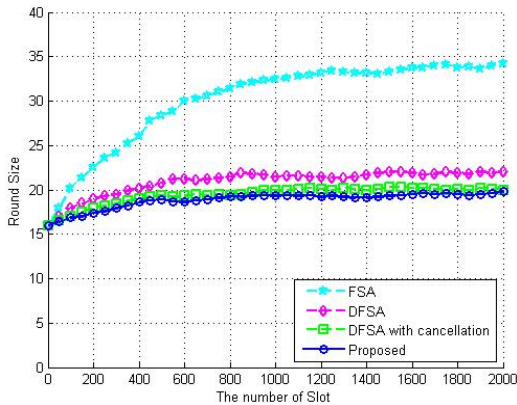


Fig. 4. Compare change of round size by the number of tags.

일 경우의 시뮬레이션 결과를 나타내었다. 제안된 방식의 경우 태그 수에 따른 C_i , C_s , C_c 의 변화가 비슷한 형태를 보인다. 이는 태그의 수가 변하더라도 모든 태그를 검색하는데 걸리는 라운드 수의 변화가 적다는 것을 보여준다. 이전 단계에서 발생된 값들에 의해 태그의 개수를 추정하고 추정된 태그의 개수에 의해서 슬롯의 수가 증가함에 따라 처음에는 검색된 값이 적지만 2-3 라운드 뒤에는 추정된 태그 개수에 따라 슬롯의 크기가 증가하여 한 번에 많은 양의 태그를 읽어 들여 라운드 수의 증가폭이 줄어든다.

Fig. 4에서는 제안된 방법 및 FSA 방식, DFSA 방식 및 상쇄 기법이 적용된 DFSA 방식의 성능을 비교하여 검증하였다. 사용된 태그 ID는 128비트 수동태그이며, 태그 개수는 50~2000개까지 50개 단위마다 실험하였다. 그리고 실험 결과는 10회씩 실험한 평균치를 계산하였다. 제안된 알고리즘에 사용되는 프레임의 크기는 최초 프레임의 크기를 16개, 최대 프레임의 크기를 256개로 가정하여 시뮬레이션을 수행하였다. 성능 평가는 제안된 알고리즘과 기존 FSA 방식, DFSA 방식, 상쇄 기법이 적용된 DFSA 방식의 라운드 수에 따른 태그 수의 추정 데이터에 대한 시뮬레이션을 수행하고, 결과 값들의 평균 값을 구해 그 결과를 비교, 분석하였다. 시뮬레이션 결과 제안된 알고리즘은 평균 18.8 라운드에 태그를 식별하였고, 상쇄 기법이 적용된 DFSA 방식은 19.4 라운드, DFSA 방식은 20.9 라운드에 태그를 식별하였으며

FSA 방식은 30.1 라운드에 태그를 식별한 것을 확인할 수 있었다.^[3,4,8]

V. 결론

본 논문에서 제안된 방법은 태그 개수에 따른 빈 슬롯 개수 및 충돌 슬롯의 개수를 통해 구해진 기대 값을 이용해 실제 태그 개수를 추정하는 충돌 방지 방법으로서, 시뮬레이션을 통해 제안된 방법을 기존 방법과 비교, 분석한 결과 태그의 수가 1000개 이하일 경우 제안된 알고리즘은 평균 18.2 라운드에 모든 태그를 식별하였으나 태그 수가 1000개 이상인 경우 19.2 라운드에 태그를 식별하였다. 또한 평균 태그 수에 따른 라운드 수가 상쇄 기법이 적용된 DFSA 방식보다 3.1%, 기존 DFSA 방식보다 10.1%, FSA 방식보다 37.5% 가량 감소하여 기존 방식에 비해 라운드 수 당 태그 수 및 처리 속도가 향상됨을 확인할 수 있었다. 본 논문에서 제안된 방식은 단순히 최적의 프레임 크기를 이용한 테이블만을 사용하기에 구현이 쉬우며, 단일 연산 처리를 통해 보다 정확하면서도 빠른 태그 수 추정이 가능하다.

최근 RFID를 이용한 디지털 스토리텔링 기법과 같은 다양한 솔루션이 개발 및 제공되고 있다. 예를 들어 국 내외 박물관에서는 입장권에 내장된 RFID를 통해 방문객 동선 추적을 이용하여 전시물의 개인화된 RFID 기반 정보 서비스를 제공하고 관람 경험을 풍부하게 함과 동시에 재관람을 유도하고 있다. 입장권은 가진 관람객들은 각 전시실의 리더기를 통해 실시간으로 원하는 정보를 제공받는다. 그러나 이처럼 적지 않은 양의 태그를 처리하게 되면 태그 간 충돌 문제가 발생하여 실시간 데이터 전송 시 충돌로 인해 신호의 전송이 되지 않는 문제가 발생한다. 본 논문에서 제안된 방법은 이와 같은 다중 태그 시 빠른 태그 수 추정을 통해 각 태그의 신호 충돌을 방지하는데 사용 가능하다.^[9]

References

1. S. H. Lee, J. H. Park, and T. J. Lee, "An efficient tag estimation method for dynamic framed slotted ALOHA

- based anti-collision algorithm in RFID System” (in Korean), in Proc. Conf. IEEK society proceeding, Suppl.2(s) **30**, 110-111 (2007).
2. S. R. Lee and C. W. Lee, “Status of multi-recognition technology of RFID systems” (in Korean), J. Soc. KIEES, Suppl.2(s) **15**, 44-45 (2004).
 3. J. R. Cha and J. H. Kim, “ALOHA-type anti-collision algorithms using tag estimation method in RFID system,” (in Korean), J. Soc. KICS, Suppl.9A(s) **30**, No.9A, 814-821 (2005).
 4. K. Finkenzeller, *RFID Handbook ; Fundamentals and Applications in Contactless Smart Cards and Identification*, 2nd Ed. (John Wiley & Sons Inc., New Jersey, 2004), pp. 195-219.
 5. H. Vogt, “Efficient object identification with passive RFID tags,” in Proc. Int. Conf. Pervasive Computing, 98-113 (2002).
 6. C. H. Quan, H. S. Mo, and G. Y. Choi, “Tag number estimation scheme in slotted aloha based RFID systems” (in Korean), in Proc. Conf. KICS, **32**, 955-958 (2005).
 7. W. T. Chen and G. H. Lin, “An efficient anti-collision method for tag identification in a RFID system,” in Proc. IEICE Trans, Suppl.12(s) **E89-B**, 3386-3392 (2006).
 8. C. Jiang, Y. Xu, and W. Qiao, “Cancellation strategy in dynamic framed slotted ALOHA for RFID system,” in Proc. Int. Conf. WCNC, 854-859 (2013).
 9. S. I. Joung and D. S. Kim, “Case study for RFID applications from business model perspective” (in Korean), J. Soc. Information Technology Applications & Management, Suppl.1(s) **20**, 197-216 (2013).

저자 약력

▶ 박 노 경(Nho-Kyung Park)



1990년 2월: 고려대학교 전자공학과 공학박사
 1984년 3월 ~ 1987년 2월: 삼성반도체통신(주) 반도체 연구소 연구원
 1999년 3월 ~ 2000년 2월: 미국 오리곤 주립 대학교 ECE 연구교수
 2011년 9월 ~ 2012년 8월: 미국 노스 캐롤라이나 A&T 주립 대학교 교환교수
 1988년 4월 ~ 현재: 호서대학교 정보통신공학전공 정교수

▶ 윤 형 기(Hyoung-Kie Yun)



2008년 2월: 호서대학교 정보통신공학과 졸업
 2010년 2월: 호서대학교 정보통신공학과 공학석사
 2010년 3월 ~ 현재: 호서대학교 정보통신공학과 박사과정 재학

▶ 문 대 철(Dai-Tchul Moon)



1981년 2월: 고려대학교 전자공학과 공학박사
 1994년: North Carolina State Univ. 연구교수
 2005년: Minnesota State Univ. Duluth 객원교수
 1984년 ~ 현재: 호서대학교 정보통신공학전공 정교수