

# 벤치마크 태그를 이용한 도착시간 차 기반의 RFID 측위 알고리즘

주운기\*

## TDOA-Based Localization Algorithms for RFID Systems Using Benchmark Tags

Un Gi Joo\*

### ■ Abstract ■

This paper considers a localization problem in time difference of arrival (TDOA)-based radio frequency identification (RFID) systems. To estimate the position of a target tag, this paper suggests three localization algorithms that use benchmark tags. The benchmark tags are the same type as the target tag, but either the locations or distance of the benchmark tags are known. Two algorithms use the benchmarks for auxiliary information to improve the estimation accuracy of the other localization algorithms such as least squared estimator (LSE). The other one utilizes the benchmarks as essential tags to estimate the location. Numerical tests show that the localization accuracy can be improved by using benchmark tags especially when an algorithm using the LSE is applied to the localization problem. Furthermore, this paper shows that our benchmark algorithm is valuable when the measurement noise is large.

Keywords : RFID, TDOA, Benchmark Tags, Localization Algorithms

## 1. 서 론

주요 자원이나 인물의 위치 정보는 중요하게 다루어지고 있다. 예를 들어, 제조업에서 생산을 위한 주요 장비의 현 위치 정보를 이용하면 생산 일정 계획 수립을 효율적으로 할 수 있고, 장비의 활용도를 높일 수 있다. 생산된 제품의 배송 과정에서도 배송 제품의 현 위치 정보를 이용하여 배송 상태 및 경로 파악을 할 수 있고, 배송 완료 시점 예측을 용이하게 할 수 있으므로 배송 서비스의 경쟁력을 높일 수 있다. 또한, 장애자나 환자의 현재 위치 파악을 통해 시급한 응급 의료 서비스를 제공할 수 있고, 중요 인물의 위치 파악을 실시간으로 하여 신변 보호 조치를 용이하게 할 수 있다. 본 논문은 위치 추정에 대한 것으로, RFID(Radio Frequency Identification) 태그(tag)를 측위하고자 하는 대상체에 부착한 후, 부착된 태그의 위치를 추정함으로써 대상체의 위치를 파악하는 방법을 다룬다.

측위 방식은 사용하는 통신망에 따라서 위성을 이용하는 GPS(Global Positioning System), 이동 통신망을 이용하는 LBS(Location Based Service), 근거리 망(LAN)을 이용하는 RTLS(Real-Time Location Service) 방식으로 구분할 수 있다. 또한 측위하는 방법에 따라 삼각법(triangulation), 근접방식(cell-ID) 및 확률적 방법(fingerprinting) 등으로 구분할 수 있다[1].

본 논문은 실내 외에서 관심 대상체의 위치를 실시간으로 측정할 수 있는 알고리즘을 개발하는 것을 목적으로 한다. 이를 위해 본 논문에서는 실내에서도 적용이 가능하도록 RFID를 이용하는 RTLS 방식을 고려하였고, RFID 시스템의 알려진 고정 위치의 리더(reader)와 벤치마크 태그 정보를 활용하는 삼각법 방식의 측위 알고리즘을 개발한다. 삼각법은 이미 위치를 알고 있는 기준점과 위치 추적 대상과의 각도, 신호 세기, 거리 등의 계산을 통해 위치 측위를 하는 방식으로, RSS(Received Signal Strength), AOA(Angle Of Arrival), TOA(Time Of Arri-

val), TDOA(Time Difference Of Arrival) 등의 방식이 있다. TDOA(도착 시간차)는 다른 기법에 비해 구현이 용이하고 저렴하게 측위할 수 있는 기법으로, ISO(International Standard Organization)에서는 ISO/IEC 24730-2에서 TDOA 기반의 RTLS를 위한 리더와 태그 간 데이터 구조와 프로토콜을 표준화하였다[9]. 본 논문은 TDOA(도착 시간차) 방식을 이용하는 측위 알고리즘에 대한 것으로, TDOA 방식을 이용한 측위 문제는 일반적으로 다수의 관측치를 이용하여 소수의 변수(태그 위치) 값을 추정하는 문제가 된다.

TDOA 방식의 측위를 위한 알고리즘으로, Foy [7]은 테일러 확장(Taylor series expansion) 기법을 제안하였고, Chan and Ho[4]는 태그 위치에 대한 MLE(Maximum Likelihood Estimator)를 이용하여 측위하는 방안을 제시하였다. 그리고 Cheung et al.[6]은 LSE(Least Squared Estimator)를 이용한 측위 방안을 제시하였다. 다수개의 측정 데이터를 이용한 위치 추정을 위해 대부분의 방법은 총 추정 오류의 제곱 합이 최소가 되는 추정량인 LSE를 주로 이용하였는데, LSE는 Torrieri[12]가 밝힌 바와 같이, 측정 오류가 정규분포를 하는 경우는 MLE와 동일한 위치를 추정하게 된다. 그러나 LSE나 MLE는 TDOA 방식으로 측정한 거리 차이 측정 데이터간 종속성을 고려하지 않은 추정량이므로, 측정 데이터 간 종속성을 반영하기 위해 Chan and Ho[4]는 추가적인 보정 연산을 수행한 측위 방안을 제시하였고, 측정 오류의 신호-잡음비가 큰 경우에 효과적인 측위를 할 수 있음을 보였다. LSE나 MLE를 이용하지 않은 방안으로, 이현재 외[2]은 TDOA 방식의 측위를 위한 최소한의 리더 수인 3개의 리더 집합 별로 각각 측위를 한 후, 이들 측위 값의 평균치를 최종적인 측위 값으로 추정하는 방안을 제시하였다. 또한 주운기[3]는 메타휴리스틱 기법의 일종인 PSO(Particle Swarm Optimization)을 이용하여 이동 물체의 측위 방안을 제시하였고, 측정 오차의 분산에 둔감한 측위 방안의 개발 필요성을 언급하였다. 그러나 이들 방법은 모

두 리더의 위치 정보만을 이용한 측위 방법으로, 리더를 추가로 설치하는 것이 측위 정확도 개선에 도움이 됨을 보였지만, 리더의 비용이 상대적으로 고가라는 문제가 있다. 본 논문은 태그의 위치 또는 거리 정보를 이용하여 비용 경제적이고 정확도 높은 측위 알고리즘을 개발하는 것이 목적이다.

태그를 이용한 측위 방법으로, Ni et al.[11]이 미리 위치가 알려져 있는 다수의 벤치마크(benchmark tags; reference tags)를 이용하는 RSS 기반의 측위 방안을 제시하였다. 여기서, 벤치마크 태그는 추적 대상체에 부착하는 태그(target tag)와 동일한 것이지만, 그 위치나 거리 간격이 정확히 알려져 있는 태그를 말한다. Ni et al.[11]도 본 논문의 세가지 측위 알고리즘 중 하나인 BM(Bench Marks) 알고리즘과 마찬가지로 벤치마크 태그의 위치 정보만 이용하여 측위하였지만, RSS 기반 방식을 다룬 것이고, BM 알고리즘은 TDOA 기반의 측위 환경을 고려했다는 차이가 있다. 또한, 본 논문은 벤치마크 태그의 위치를 모르는 경우에도 적용할 수 있는 측위 알고리즘(이중태그 알고리즘)과 정확도 개선을 위한 벤치마크 태그 활용 방안도 추가로 제시한다. 무선 센서망(Wireless Sensor Network WSN)에서의 특정 센서 노드의 측위를 위해서도 위치가 알려져 있는 인접 노드(anchor)의 위치 정보를 이용하는 방안을 사용한다[5, 10]. 그러나 이를 위해서는 각 노드가 데이터의 중계 기능도 가지고 있어야 하므로, WSN에서의 측위 방식을 본 논문의 RFID 시스템을 위해 이용하기는 어렵다.

본 논문에서는 벤치마크를 이용하는 BT(Benchmark tags), DT(Double Tags) 및 BM(Bench marks)의 세 가지 측위 알고리즘을 개발하고, 시뮬레이션을 통해 성능 분석을 한다. BT 및 DT 알고리즘은 주어진 1차적인 측위 값의 정확도 개선을 위해 벤치마크 정보를 이용하는 것으로, 1차적인 측위 값을 위해서는 발표되어 있는 임의의 측위 알고리즘을 사용하면 되지만, 본 논문에서는 LSE와 Chan and Ho[4]의 측위 방법을 이용하여 1차적인 측위 값을 구하는 방안을 이용하였다. BM 알고리즘은 TDOA

방식의 측위 데이터를 이용한 추적 태그와 근접한 벤치마크 태그의 위치 정보를 이용하는 측위 방법으로, 측정 데이터의 잡음 오류에 둔감한 측위를 할 수 있는 방안을 보인다. 본 논문에서 제시하는 측위 알고리즘은 고가의 리더 대신 태그를 이용하므로, 측위 정확도를 경제적으로 개선할 수 있을 것으로 기대한다[8].

본 논문의 구성은 제 2장에서는 측위 문제를 설명하였고, 제 3장에서는 3가지의 측위 알고리즘을 제시하였다. 제 4장에서는 제시한 3가지 알고리즘의 성능 분석을 위해 다양한 시뮬레이션을 수행하였고, 제 5장에서 본 알고리즘의 기대 효과 및 추후 연구 과제를 기술하였다

## 2. 측위 모형

RFID를 이용한 위치 추적 시스템은 태그, 리더, 안테나, 그리고 측위 엔진(positioning engine) 또는 중앙 서버로 구성된다. 태그에서 주기적으로 보내오는 신호를 리더가 수신하여 측위 엔진에 송신하면, 측위 엔진에서는 측위 알고리즘을 이용하여 측위를 한다. 본 논문은 측위 엔진에서 사용될 수 있는 측위 알고리즘 개발을 목적으로 한다.

$M$ 개의 리더를 이용하여 태그의 위치를 측정하는 RFID 시스템을 고려하자. 각 리더  $i$ 는 2차원 공간상의 고정 위치  $(x_i, y_i)$ 에 존재하고, 이들 리더의 위치는 모두 알려져 있다고 가정하자, 여기서  $i = 1, 2, \dots, M$ . 다수개의 추적 대상 태그가 존재하는 경우, 각 추적 대상 태그 별로 독립적으로 측위할 수 있으므로, 본 연구에서는 편의 상 하나의 추적 태그가 미지의 위치  $(x, y)$ 에 있는 경우를 고려한다. TDOA 기반 방식은 미지의 위치인  $(x, y)$ 를 추정(estimation)하기 위해 거리 차이를 이용하므로, 거리 차이 계산을 위한 기준점이 필요한데, 여기에서는 임의로 위치  $(x_1, y_1)$ 의 리더 1을 기준 리더(reference reader)로 정하기로 한다. 태그와 리더  $i$ 간 직선 거리를  $r_i$ 로 표시하면, 기준 리더에 대한 거리 차이  $r_{i1}$ 를 식 (1)과 같이 표시할 수 있다.

$$\begin{aligned} r_{i1} &= r_i - r_1, \quad i = 2, \dots, M \\ &= \sqrt{(x-x_1)^2 + (y-y_1)^2} - \sqrt{(x-x_1)^2 + (y-y_1)^2} \end{aligned} \quad (1)$$

식 (1)을  $i = 2, \dots, M$ 에 대해 동시에 만족하는  $(x, y)$  값이 태그의 위치 값이 되지만,  $r_i$ 를 위해서는  $(x, y)$  값을 알아야 하므로 거리 차이 값인  $r_{i1}$ 의 정확한 값을 알 수가 없다. 따라서 이론적인 거리 차  $r_{i1}$  대신, 실제 측정을 통해 얻은 거리차이( $d_{i1}$ )를 이용하여 측위를 한다. 위치  $(x, y)$ 의 태그가 주기적으로 방송하는 신호를 각 리더가 수신한 후, 측위 엔진에서 거리 차이 값  $d_{i1}$ 을 계산하는데, 이 값에는 측정 오류를 포함한 오류  $\epsilon_{i1}$ 가 존재한다:  $d_{i1} = r_{i1} + \epsilon_{i1}$ ,  $i = 2, \dots, M$ . 따라서 측정 오류 값을 포함하고 있는  $M-1$ 개의  $d_{i1}$  값을 모두 만족하는  $(x, y)$ 는 존재하지 않을 수 있는데, 실제 위치인  $(x, y)$ 와의 차이가 작은 추정 값을 찾는 알고리즘을 제시하는 것이 본 논문의 목적이다. 그러나 측위를 위한 실제 환경에서는 추적 대상 태그의 위치  $(x, y)$ 를 알 수 없기 때문에, 측위한 결과와 실제 위치 간 거리 차이 정도를 정확히 평가할 수 있는 방법이 없다. 따라서 본 논문에서는 정확한 위치나 거리 차이를 알고 있는 벤치마크 정보를 이용하는 방안을 제시한다.

### 3. 측위 알고리즘

본 논문에서는 측위를 위해 두 가지 유형의 태그를 이용한다. 두 가지 중 한 태그 유형은 위치 추적 대상이 되는 태그(target tag)로, 이동하는 대상체에 부착된 태그이다. 다른 하나의 태그 유형은 벤치마크 태그(benchmark tag)로, 사전에 알려진 위치나 알려진 거리 간격으로 배치된 태그를 말한다. 본 논문에서는  $T$ 개의 벤치마크 태그 정보를 이용하여 추적 대상 태그의 위치를 추정한다. 이를 위해  $(x_t^b, y_t^b)$ ,  $t = 1, 2, \dots, T$ 를 벤치마크 태그  $t$ 의 위치라고 하자. 여기서, 윗 첨자  $b$ 는 벤치마크 태그임을 표시한다.

### 3.1 벤치마크 태그(Benchmark Tags : BT)

#### 알고리즘

벤치마크 태그(BT : Benchmark Tags) 알고리즘은 추적 태그의 추정 위치 정확도를 높이기 위한 보조 수단으로 벤치마크 태그의 위치 정보를 이용하는 방안으로, 이미 알고 있는 위치에 배치된  $T$ 개의 벤치마크 태그의 위치 정보를 이용한다. BT 알고리즘을 위한 벤치마크 태그는 위치 추적 대상 태그와 동일한 태그이지만, 벤치마크  $t$ 의 위치가  $(x_t^b, y_t^b)$ 라는 것을 미리 알고 있는 태그로, BT 알고리즘은 리더의 위치 정보를 이용하여 추적 태그의 위치를 먼저 추정하고, 추정된 위치의 정확도를 개선하기 위해 벤치마크 위치 정보를 다음과 같이 활용한다.

**단계 1 :** 추적 대상 태그로부터의 직선 거리가 가장 짧은 것으로 예측되는 벤치마크 태그를 하나 찾는다. 추적 대상 태그와 찾은 벤치마크 태그와의 거리가 미리 정한 거리 이내이면, 단계 2를 수행하고, 그렇지 않으면, 추적 대상 태그의 위치를 추정하고 마친다.

**단계 2 :** 추적 대상 태그와 최단거리 벤치마크 태그의 위치를 각각 추정한다.

**단계 3 :** 벤치마크 태그의 추정 값과 실제 위치 간 차이를 계산하고, 그 차이 값 만큼을 추적 대상 태그의 추정 위치 값에 보정하여 추적 대상 태그의 위치를 추정한다.

BT 알고리즘은 벤치마크 태그와 추적 대상 태그 간 거리 차이 값의 크기에 따라서, 벤치마크 태그의 위치 정보 값 이용 여부를 단계 1에서 결정한다. 단계 1에서는  $T$ 개의 벤치마크 태그 중에서 추적 대상 태그와 가장 근접한 위치에 있는 벤치마크 태그를 찾기 위해 식 (2)를 이용한다.

$$t0 = \arg_{1 \leq t \leq T} \text{Min} \left\{ \sum_{i=1}^M (d_{i1} - d_{i1}^t)^2 \right\} \quad (2)$$

$$\text{Min}\left\{\sum_{i=1}^M (d_{i1} - d_{i1}^t)^2\right\} \leq 36(M-1) \text{의 경우,}$$

여기서  $d_{i1}^t$  벤치마크 태그  $t$ 에 대해 TDOA 기반으로 측정된 기준 리더에 대한 거리 차이 값을 나타낸다. TDOA 기반의 거리 차이 식 (1)은 쌍곡선을 형성하므로, 식 (2)을 이용하여 찾은 벤치마크 태그가 추적 대상 태그와 가장 근접하다는 보장을 할 수는 없으나, 만약 벤치마크 태그  $t$ 와 추적 대상 태그가 근접한 위치에 있다면,  $d_{i1}$ 와  $d_{i1}^t$ 간 차이도 적을 가능성이 크므로, 이 차이 값을 식 (2)와 같이 계산하여 근접성을 판단하였다. 그러나 식 (2)에 의한 차이 값이 매우 크다면, 두 태그간 거리가 상당히 떨어져 있는 것일 가능성이 큰데, 이 경우는 벤치마크 태그 위치의 추정 오차 값이 추정 태그 위치 보정에 큰 도움이 되지 못할 수 있다. 이는 두 태그의 신호 전송 경로가 큰 차이가 있다는 의미이므로 본 논문에서는  $36(M-1)$ 를 임계 값으로 설정하여, 만약  $\text{Min}\left\{\sum_{i=1}^M (d_{i1} - d_{i1}^t)^2\right\} > 36(M-1)$ 라면, 벤치마크 태그의 위치 정보를 이용하지 않는다. 여기서 임계 값  $36(M-1)$ 은 ISO/IEC 24730-2[9]에서 최대 오차를 3m 이내로 해야 한다는 권고를 따르기 위한 값이다.

단계 1에서 의미가 있는 근접 벤치마크 태그  $b_0$ 를 찾은 경우에는 단계 2를 수행하고, 그렇지 않은 경우에는 추적 태그의 측위를 수행한다. 단계 1과 2에서 태그 위치 추정을 위해 사용하는 측위 알고리즘은 기존에 발표된 측위 알고리즘을 포함한 임의의 알고리즘을 이용하면 된다. 본 논문에서는 이를 위해 LSE(Least Squared Estimator)와 Chan and Ho[4]가 제안한 측위 알고리즘을 이용하였다. LSE는 식 (3)을 이용하여 구할 수 있다[4, 6, 12].

$$(G^T Q^{-1} G)^{-1} G^T Q^{-1} H, \quad (3)$$

$$\text{여기서, } G = (-1) \begin{pmatrix} x_{21} & y_{21} & d_{21} \\ x_{31} & y_{31} & d_{31} \\ \vdots & \vdots & \vdots \\ x_{M1} & y_{M1} & d_{M1} \end{pmatrix},$$

$$H = (1/2) \begin{pmatrix} d_{21}^2 - K_2 + K_1 \\ d_{31}^2 - K_3 + K_1 \\ \vdots \\ d_{M1}^2 - K_M + K_1 \end{pmatrix} \text{이고,}$$

$Q$ 는 측정 오차의 공분산 행렬(covariance matrix)을 나타낸다. 이들 행렬에 사용된 기호는 각각  $i = 1, 2, \dots, M$ 에 대해  $K_i = x_i^2 + y_i^2$ ,  $x_{i1} = x_i - x_1$ ,  $y_{i1} = y_i - y_1$ 을 나타낸다.

마지막으로, 단계 3에서는 측위 알고리즘의 측위 오차를 수정하기 위한 과정으로, 정확한 위치 값을 알고 있는 벤치마크 태그 정보를 이용하여 측위 오차를 계산하고, 이 오차가 추적 대상 태그 측위에도 동일하게 발생한 것으로 간주하여, 이 추정 오차 값을 보정한다. 즉, 단계 2에서의 추적 대상 태그 위치의 추정 값을  $(\hat{x}, \hat{y})$ 라 하고, 근접 벤치마크 태그  $b_0$ 의 추정 위치를  $(\hat{x}_{b_0}^b, \hat{y}_{b_0}^b)$ 라 하자. 벤치마크 태그  $b_0$ 의 실제 위치인  $(x_{b_0}^b, y_{b_0}^b)$ 에 대해, 위치 추정치의 각 좌표 방향으로  $\Delta_x = x_{b_0}^b - \hat{x}_{b_0}^b$ 와  $\Delta_y = y_{b_0}^b - \hat{y}_{b_0}^b$ 이므로, 이 값을 추적 태그 위치에 더해줘서 추적 태그 추정 값을 보정할 수 있다. 여기서, 단계 1에서와 같이 위치 보정은 오차가 3m를 초과하는 경우에만 하였다. 즉,  $\sqrt{\Delta_x^2 + \Delta_y^2} > 3$ 이면,  $(\hat{x} + \Delta_x, \hat{y} + \Delta_y)$ 을 추적 태그의 위치로 추정하고, 그렇지 않으면,  $(\hat{x}, \hat{y})$ 를 추적 태그 위치로 추정한다.

### 3.2 이중 태그(Double Tags : DT) 알고리즘

앞의 BT 알고리즘이 추적 대상 태그가 임의의 위치에 있더라도 벤치마크 태그 정보를 이용할 수 있고, 측위 정확도 개선을 위해 추적 대상체에는 추가적인 비용이 들지 않는다는 장점은 있으나, 측위 정확도 개선에 벤치마크 태그가 도움이 된다는 보장을 하기가 어렵다는 문제가 있다. 즉, BT 알고리즘은 벤치마크 태그와 추적 대상 태그의 상대적인 위치가 식 (2)를 만족하는 태그가 없는 경우에는 벤치마크 태그 위치 정보가 사용되지 않는다. 따라서 본 장에서는 이러한 문제 해결을 위한 방안으로 미

리 설정한 간격으로 총 두 개의 태그를 측위 대상체에 부착하는 방안을 제시한다. 즉, DT(Double Tags) 알고리즘은 거리 차이 값을 알고 있는 두 추적 태그의 위치 값을 추정하여, 이를 통한 추정 오차를 개선하는 방안으로, 추적 대상 위치  $(x, y)$ 에 추적 대상 태그를 부착하고, 추가로 하나의 벤치마크 태그를 추적 대상 태그와의 거리가  $D$ 인 위치에 부착하여 다음과 같은 절차로 위치 추정을 한다.

**단계 1 :** 두 개의 태그 위치를 각각 추정한다.

**단계 2 :** 두 태그의 추정 위치간 거리 차가 미리 정한 오차 범위 내이면, 첫째 태그의 위치를 추적 대상체의 위치 값으로 선택한다. 그러나 두 태그 간 위치 추정치의 거리 차가 일정 값 이상이면, 거리 차의 1/2만큼 위치 값을 보정하여 측위 값으로 추정한다.

DT 알고리즘을 위한 두 태그의 위치는 각각  $(x, y)$ 와  $(x_1^b, y_1^b)$ 로, 두 태그의 위치는 거리  $D$ 와 같은데  $D = \sqrt{\{(x-x_1^b)^2 + (y-y_1^b)^2\}}$ 의 관계가 있다. 즉, DT 알고리즘은 태그 위치  $(x, y)$ 와  $(x_1^b, y_1^b)$ 는 알지 못하는 값이지만 이들 두 태그간 거리가  $D$ 라는 것을 알고 있는 경우에 이용할 수 있는 것으로, 거리  $D$  간격으로 두 개의 태그 쌍으로 구성된 태그를 추적 대상체에 부착하여 측위 할 수 있다.

DT의 단계 1에서는 두 개의 태그에 대한 측위

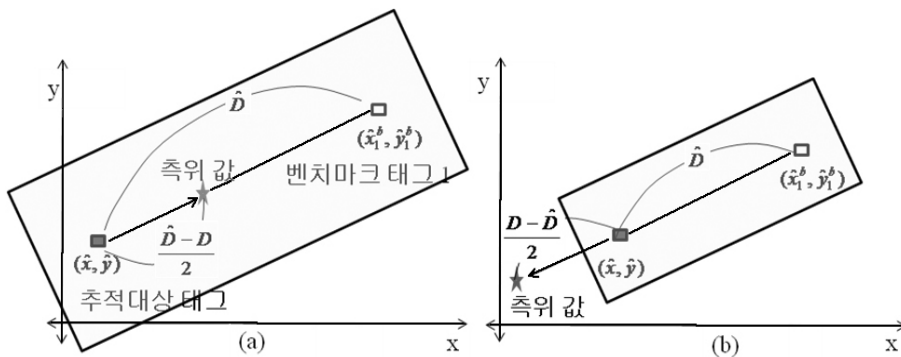
를 각각 독립적으로 수행한다. 단계 1에서의 측위를 위해서는 BT 알고리즘과 마찬가지로 식 (3)의 LSE나 Chan and Ho[4]를 포함한 임의의 측위 방법을 이용할 수 있다. 단계 2에서는 이 측위 추정 값이 각각  $(\hat{x}, \hat{y})$ 와  $(\hat{x}_1^b, \hat{y}_1^b)$ 라 할 때, 두 추정 위치간 거리 차이가  $D$ 에 대해  $3m$  이내이면  $(\hat{x}, \hat{y})$ 를 추정 위치로 선택하고, 그렇지 않으면 추정치간 거리가  $D$ 가 되도록 보정한다. 즉, <그림 1>에서 두 태그 간 추정 거리가 <그림 1>(a)와 같이 실제 거리  $D$ 보다 긴 경우는 길이 차이  $\hat{D}-D$ 의 1/2만큼의 거리가 단축되도록 원래의 측위 위치  $(\hat{x}, \hat{y})$ 에서 <그림 1>(a)의 측위 값 위치로 보정한다. 반대로,  $\hat{D}$ 가  $D$ 보다 작은 경우에는  $(D-\hat{D})/2$ 길이만큼 길이를 신장하는 위치 값으로 추적 대상체의 측위 값으로 <그림 1>(b)와 같이 보정한다. 이와 같은 보정은 단계 1의 결과인  $(\hat{x}, \hat{y})$ 와  $(\hat{x}_1^b, \hat{y}_1^b)$ 에 대해 식 (4)를 이용하여 수행한다.

$$\hat{x} = \frac{(\hat{D}-D)\hat{x}_1^b + (\hat{D}+D)\hat{x}}{2\hat{D}} \text{ 이고}$$

$$\hat{y} = \frac{(\hat{D}-D)\hat{y}_1^b + (\hat{D}+D)\hat{y}}{2\hat{D}}, \quad |D-\hat{D}| \geq 3\text{일 때 (4)}$$

여기서  $\hat{D} = \sqrt{(\hat{x}-\hat{x}_1^b)^2 + (\hat{y}-\hat{y}_1^b)^2}$ 로, 두 추정 좌표 값  $(\hat{x}, \hat{y})$ 와  $(\hat{x}_1^b, \hat{y}_1^b)$ 간 직선 거리를 나타낸다.

DT 알고리즘은 추적 대상 태그와의 거리가  $D$ 인



<그림 1> 이중태그 측위 보정 방식

하나의 벤치마크 태그만 이용한다 :  $T = 1$ . 따라서 BT에서와 같이 2개 이상의 많은 벤치마크 태그가 필요하지 않고, 또한 벤치마크 태그의 정확한 위치 값을 알지 않아도 되므로, 태그 위치 값을 관리할 필요가 없다는 장점이 있다.

### 3.3 벤치마크(Bench Mark : BM) 알고리즘

벤치마크의 정보만을 이용하여 추적 태그의 위치를 추정하는 방안을 생각할 수 있다. BM(Bench Mark) 알고리즘은 앞에서 다룬 BT 알고리즘과 마찬가지로, 위치 값을 모두 알고 있는  $T$ 개의 벤치마크 태그를 이용한다. 이를 위해서는  $T$ 개의 벤치마크 태그 중, 추적 대상 태그와 근접한 벤치마크 태그의 위치 정보만을 이용하는 방안을 생각할 수 있고, 본 BM 알고리즘은 근접한  $B$ 개의 벤치마크 태그에 대해 식 (5)를 이용하여 측위한다.

$$(\hat{x}, \hat{y}) = \left( \sum_{t=1}^B w_{[t]} x_{[t]}^b, \sum_{t=1}^B w_{[t]} y_{[t]}^b \right), \quad (5)$$

여기서  $B$ 는 1과  $T$ 사이의 한 정수 값이고,  $[t]$ 는  $\sum_{i=1}^M (d_{i1} - d_{i1}^t)^2$  값이 작은 것부터 커지는 순으로 나열하였을 때,  $t$ 번째의 벤치마크 태그를 표시한다.  $w_{[t]}$ 는  $t$ 번째 벤치마크 태그의 가중치로, 식 (6)과 같이 계산할 수 있다. 측위를 위해 선정하는  $B$  값은 BM 알고리즘의 성능에 영향을 주는 인자 중 하나로, BM 알고리즘을 이용하기 위해서는  $B$  값을 미리 결정해야 한다.

$$w_{[t]} = \frac{1}{\sum_{i=1}^M (d_{i1} - d_{i1}^t)^2 \sum_{i=1}^T [1 / \sum_{i=1}^M (d_{i1} - d_{i1}^t)^2]} \quad (6)$$

BM 알고리즘은 벤치마크 태그의 위치 정보만을 이용하여 측위하므로, 측위 정확도는 추적 태그에 근접한 벤치마크 태그가 많을수록 증가할 것으로 예상된다. 또한 측정 거리 값인  $d_{i1}$ 의 측정 오차의

분산에 영향이 작은 측위를 할 수 있는 장점이 기대된다.

## 4. 성능 분석 결과

본 논문에서 제안한 측위 알고리즘 BT와 DT를 위해서는 1차적인 측위 값을 찾는 방안이 필요하다. 이를 위해 식 (3)의 LSE와 Chan and Ho[4]의 추정량을 이용하였고, 본 장에서는 이를 각각 LSE와 CH로 표시하였다. 본 논문에서 고려한 벤치마크 태그의 효과를 분석하기 위해 시뮬레이션을 통해 LSE, BT\_LSE, DT\_LSE, CH, BT\_CH, DT\_CH, 그리고 BM 알고리즘의 측위 정확도를 비교하였다. 여기서, LSE와 CH는 벤치마크 태그 정보를 이용하지 않는 경우의 측위 방안이다. BT 알고리즘의 단계 1 및 2에서 LSE와 CH를 이용하는 BT 알고리즘을 각각 BT\_LSE와 BT\_CH로 표시하였다. 그리고, DT\_LSE와 DT\_CH는 각각 DT 알고리즘의 단계 1에서 LSE 또는 CH를 이용하는 DT 알고리즘을 나타낸다. 시뮬레이션은 각 실험 환경 별로 300개의 문제를 임의로 발생시켜서 이 문제들에 대해 식 (7)의 MSE (Mean Squared Error)을 비교하였다.

$$MSE = \sum_{j=1}^{300} \frac{(x_j - \hat{x}_j)^2 + (y_j - \hat{y}_j)^2}{300}, \quad (7)$$

여기서  $(x_j, y_j)$ 와  $(\hat{x}_j, \hat{y}_j)$ 는 각각  $j$ 번째 문제의 추적 대상 태그 위치와 이에 대한 추정 위치를 나타내므로,  $\sqrt{MSE}$ 는 이들 위치간 평균 거리 오차가 된다.

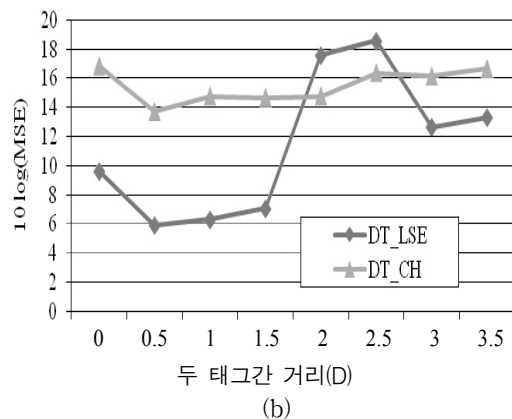
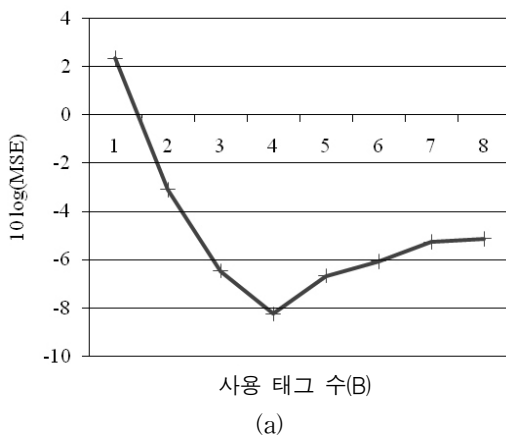
시뮬레이션을 위해서 Microsoft Visual C++ 6.0로 2.80G Hz Intel Core 2 Duo CPU PC에서 모든 비교 대상 알고리즘을 구현하였다. 실시간 측위를 이해서는 알고리즘의 실행 시간이 짧아야 하는데, 시뮬레이션 결과, 모든 알고리즘의 계산시간(CPU time)은 0.5ms 이하로 작아서 본 논문에서는 별도로 다루지 않았다.

측위 알고리즘을 이용하기 위해서, BM은 사용할

벤치마크 태그의 갯수  $B$ 를 결정해야 하고, DT는 태그간 거리  $D$ 를 정해야 한다. 이를 위해서 4개의 리더가  $30\text{m} \times 30\text{m}$ 의 직사각형 공간의 좌표  $(0, 0)$ ,  $(0, 30)$ ,  $(30, 0)$ ,  $(30, 30)$ 에 배치되어 있는 상황을 고려하였다. 벤치마크 태그는  $30\text{m} \times 30\text{m}$  공간 내에  $3\text{m}$  간격의 격자 구조로 배치되어 있지만, 추적 대상 태그는 임의의 위치에 존재하는 것으로 하였다. ISO/IEC 24730-2[9]에서는  $300\text{m} \times 300\text{m}$  영역 내에서의 측위 오차가  $3\text{m}$  이내가 되기를 권고하고 있으나, 이 크기의 공간에서 오차 한계를 보장하기 위해  $3\text{m}$  간격의 벤치마크 태그를 배치하는 경우 필요한 벤치마크 태그 수(약 10,000개)가 C++ 언어의 배열 처리 용량을 초과하므로, 사전 파라미터 설정을 위해서는  $30\text{m} \times 30\text{m}$  공간을 고려하였다. 각 태그에서 보내온 신호의 측정 오차는 평균이 0이고 대각 원소 값은  $\sigma^2$ 이지만 기타 원소 값은  $0.5\sigma^2$ 인 공분산 행렬  $Q$ 를 갖는 정규분포로 가정하였다. 여기서  $\sigma^2$ 은 Chan and Ho[4]와 같이 0.001로 설정하였다. 이 상황에서  $B = 1, 2, 3, 4, 5, 6, 7, 8$ 에 대해 BM에 따른 MSE를 비교한 결과, <그림 2>(a)에서 보는 바와 같이  $B = 4$ 일 때의 MSE가 가장 작았다. 따라서 지금부터는  $B = 4$ 로 고정하여 BM의 성능을 비교 분석하기로 한다. 이는 추적 대상 태그에 대해 4개의 방향으로 하나씩 등 간격으로 벤치

마크 태그가 배치되어 있다면, 이들 벤치마크 태그의 중앙값인 추정 위치에 추적 대상 태그가 존재할 가능성이 증대하기 때문인 것으로 추측된다. 또한,  $D = 0 \sim 3.5\text{m}$ 에서의  $0.5\text{m}$  간격으로 DT의  $D$ 에 대한 MSE를 분석한 결과는 <그림 2>(b)와 같이  $D = 0.5\text{m}$ 일 때 오류가 작으므로, 본 논문에서는 지금부터  $D = 0.5$ 로 고정하여 시뮬레이션을 하기로 한다. 거리가 너무 멀다면 실용성에도 문제가 있고, 더구나, 너무 긴 거리  $D$ 는 두 벤치마크 간 오류 특성치에 유사성이 감소하여 측위 정확성이 감소하게 된다. 반면에 너무 짧은 경우에는 두 개의 태그를 이용하는 효과가 감소하기 때문에  $D = 0.5$ 에서 작은 오류가 나온 것으로 보인다. <그림 2>에서 식 (7)의 MSE 대신  $10\log(\text{MSE})$ 를 표시하였는데, 이는 로그 척도로 변환한 것이 알고리즘간 성능의 상대적 비교가 용이하기 때문에 이용하였고,  $10\log(\text{MSE})$ 를 다른 그림에서도 척도로 이용하였다.

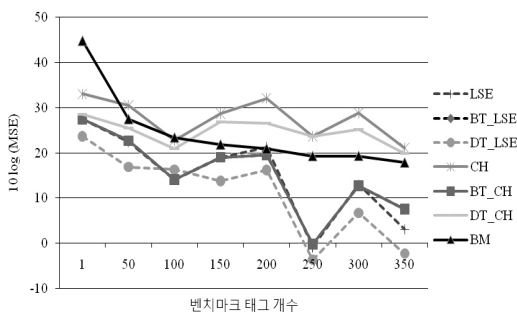
앞에서 결정한 파라미터 값을 이용하는 BM 및 DT 알고리즘을 포함한 BT 알고리즘의 성능 분석을 위해, 실험 환경을 ISO/IEC 24730-2에서와 같은  $300\text{m} \times 300\text{m}$ 의 영역으로 확장하였다. 그리고 DT 알고리즘을 위해서는 이중태그 중 하나가 추적 태그 역할을 하고, 다른 하나는 추적 태그와  $0.5\text{m}$  간격의 위치에 있는 것으로 설정하였다.



<그림 2>  $B$  및  $D$ 에 따른 BM 및 DT 알고리즘의 측위 오차



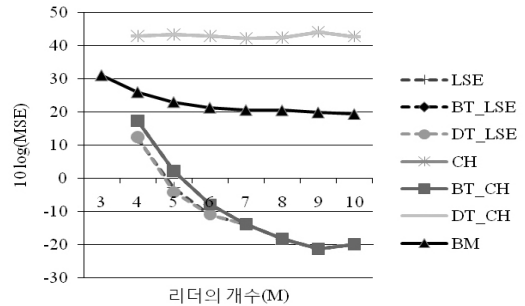
먼저, 벤치마크 태그의 수에 따른 성능 분석을 실시하였다. 배치된 벤치마크 태그의 수가 많을수록 설치 비용은 증가되지만, 식 (2)를 만족하는 태그의 존재 가능성이 커지고, 벤치마크 태그의 위치 정보 활용도가 커지므로 측위 정확도는 높아질 것으로 예상된다. 이에 따른 성능 비교를 위해 300m×300m 공간에서 각각 1, 50, 100, 150, 200, 300, 350 개의 벤치마크 태그가 이용되는 경우의 측위 오차를 측정하였다. 여기서, 각 벤치마크 태그는 300m×300m의 공간내의 임의의 위치에 배치하고, 리더는 4개의 리더가 좌표 (0, 0), (0, 300), (300, 0), (300, 300)에 고정 배치되어 있는 상황을 고려하였고, 측정 오차는  $\sigma^2 = 0.001$ 로 설정하였다. <그림 3>의 성능 분석 결과에서 볼 수 있듯이, 예상대로 벤치마크 태그 수가 증가함에 따라 측위 오차는 감소하는 경향을 보인다. <그림 3>의 DT\_LSE와 BT\_LSE, BT\_CH가 BM이나 CH보다 측위 오차가 작으므로, BT나 DT를 사용하는 것이 이를 사용하지 않는 경우보다 측위 오차 개선에 도움이 됨을 알 수 있는데, 특히 DT\_LSE의 측위 오차가 가장 작았다. 즉, LSE를 이용해서 구한 1차적인 측위 값에 대해 DT 알고리즘으로 개선하는 방식이 측위 정확도 개선에 가장 효과적이라는 것을 보인다.



<그림 3> 벤치마크 태그 수에 따른 측위 오차

다음으로, 리더의 개수에 따른 성능 분석을 하였다. 리더의 개수가 각각 3, 4, 5, 6, 7, 8, 9, 10이 됨에 따른 측위 오차 비교를 위해, 각 리더는 300m×300m 내에서 임의의 위치에 배치되는 상황을 고려

한다. 이 경우 앞의 실험 환경과는 달리 추적 대상 태그가 리더들로 형성되는 내부 탐색 공간(convex hull) 밖에 존재할 수 있다. 여기서, 벤치마크 태그는 350개가 임의의 위치에 배치되며, 추적 대상 태그 및 이중태그도 임의의 위치에 배치되는 상황을 고려하였다.

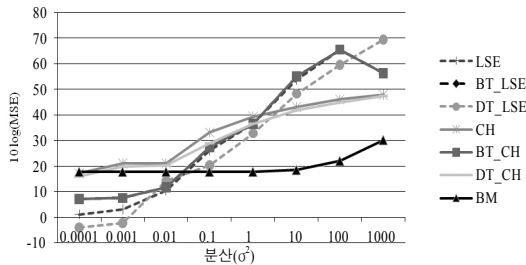


<그림 4> 리더 수에 따른 측위 오차

리더의 개수에 따른 측위 오차는 <그림 4>에 표시된 바와 같이 리더의 개수가 증가함에 따라 측위 오차는 감소하는 경향을 갖는다, 즉, 측위 오차의 개선을 위해서 리더의 수를 증가하는 것이 도움이 된다는 것을 알 수 있다. <그림 4>에서 리더의 수가 3개인 경우에는 총 300개의 문제 중 일부의 문제에서 식 (3)의 LSE를 계산할 수 없었기 때문에, BM 알고리즘 외의 알고리즘은 측위 오차를 표시하지 못하였다. 이러한 현상은 식 (3)의 계산에 필요한 역 행렬이 존재하지 않는 리더의 배치에서 발생하였다. 분석 결과는 CH의 해에 대해 BT 알고리즘(BT\_CH)이 측위 정확도를 크게 개선한다는 것을 보인다. 이는 리더 수가 증가함에 따라 식 (2)를 만족하는 벤치마크 태그가 존재할 가능성이 증대하기 때문인 것으로 보인다. LSE의 경우에는 BT 및 DT 방식의 이용 효과가 크지 않은데, 이는 LSE의 평균 측위 오차 MSE가  $M = 4$ 일 때는  $4.005m^2$ 이고,  $M = 9$ 일 때는  $0.08m^2$ 로 작아서 개선의 가능성이 적기 때문인 것으로 보인다.

마지막으로, 측위 정확도 분산 값에 따른 분석을 수행하였다. 이를 위해서 좌표 (0, 0), (0, 300), (300, 0),

(300, 300)에 고정 배치된 4개의 리더와 350개의 임의 위치 벤치마크 태그를 고려하였다. 분산( $\sigma^2$ ) 값의 0.0001~1,000범위에서의 변동에 따른 측위 오차는 <그림 5>와 같이 분석되었다.



<그림 5> 측위 오차 분산에 따른 측위 정확도

측정 오차의 분산이 커지면 측정 거리의 정확도가 낮아지는 효과를 가지므로, 측위 알고리즘의 오차는 분산이 증가함에 따라 증가할 것으로 예상할 수 있고, 이는 <그림 5>에서도 확인할 수 있다. 반면에, BM 알고리즘의 경우는 이미 위치를 알고 있는 벤치마크 태그 정보를 이용하므로, 측정 오차 분산에 영향을 가장 작게 받아서, 분산의 증가에 따른 오차 정확도가 둔감함을 알 수 있다. 또한, 분산이 0.01 이하인 상황에서는 LSE나 BT 및 DT를 이용하는 방안이 BM 알고리즘에 비해 측위 정확도가 좋을 수 있다.

시뮬레이션 결과, 벤치마크 태그 정보를 이용하는 BT와 DT 및 BM 알고리즘의 측위 오차 개선 효과가 있음을 알 수 있다. 특히 BT\_LSE 및 BT\_CH 알고리즘과 DT\_LSE 알고리즘의 개선 효과가 큰데, 이는 리더 가격의 1% 이하로 저가인 태그를 이용하여 효과적으로 측위 정확도를 개선할 수 있다는 것을 보인다[8]. <그림 4>에서 보는 바와 같이 태그가 리더의 내부 탐색 공간 외부에 존재하는 경우에 있어서의 CH의 개선에도 BT 및 DT 알고리즘은 큰 효과가 있음을 알 수 있다. 또한, NLOS(non-light-of-sight)와 같은 잡음이 심한 환경에서는 BM 알고리즘을 이용하여 잡음에 둔감한 측위를 할 수 있음을 알 수 있다. BT 및 BM 알고리즘의

측위 오차는 벤치마크 태그를 많이 사용할수록 더욱 개선된다는 것도 확인할 수 있다.

## 5. 결 론

본 논문은 TDOA 방식의 거리 측정을 통한 태그 위치 추적 문제를 다룬다. 측위 정확도 개선을 위해서 본 논문에서는 위치나 거리 간격이 알려진 다수의 벤치마크 태그를 이용하는 방안으로 BT, BM, DT 알고리즘의 3가지를 제시하였다. BT 알고리즘은 추적 대상체에 TDOA 기반으로 가장 근접한 하나의 벤치마크 측위 오차를 이용하는 방식이고, BM은 추적 대상체에 TDOA 기반으로 근접한  $B$ 개의 벤치마크 태그의 위치를 이용하는 방안이다. 마지막으로, DT 알고리즘은 미리 정한 길이의 간격이 있는 두 개의 태그에 대한 거리간격 오차 값을 이용하는 방안이다. 따라서, BT와 BM 알고리즘은 태그의 가격이 저렴하고 태그의 위치 관리가 용이한 상황에서 유용한 방안이고, 반대로 DT 알고리즘은 태그 비용이 상대적으로 고가이거나 태그 위치 관리가 어려운 경우에 적합한 방안이다. 그리고, BT 알고리즘은 추적 대상 태그가 위치한 장소에 장애물이 적어서 측정 오차 오류가 작은 상황에서 정확도가 높은 방안이지만, BM 알고리즘은 측정 오차 오류가 큰 경우에도 적용할 수 있는 측위 방안이다. 다양한 시뮬레이션 결과는 본 논문의 벤치마크 태그를 이용하는 측위 알고리즘이 측위 정확도를 효과적이고도 경제적으로 개선하는데 이용될 수 있음을 보인다. 시뮬레이션 결과, BT, DT 알고리즘이 LSE나 CH의 측위 정확도 개선에 도움이 큰 것을 알 수 있고, BM 알고리즘은 측정 오차의 분산이 큰 경우에도 분산에 둔감하게 측위할 수 있다는 장점이 기대된다.

본 논문은 TDOA 기반의 측정 환경에서의 측위 문제를 다루었지만, 태그간 시간 동기 기능이 있거나 신호 세기 및 신호 각도 측정을 용이하게 할 수 있다면, TOA, RSS, AOA 등의 타 측정 기반 방식과의 연계 측위 방안 개발도 유용할 것으로 판단된다.

## 참고 문헌

- [1] 김학용, “무선 랜 기반 위치 정보 서비스”, 『Telecommunications Review』, 제16권, 제2호(2006), pp.188-202.
- [2] 이현재, 정승희, 오창현, “RTLS 서비스를 위한 고 정밀 위치 추정 기법 연구”, 『Telecommunications Review』, 제18권, 제6호(2008), pp. 1072-1088.
- [3] 주운기, “RFID 시스템에서 이동체의 위치 추적을 위한 알고리즘”, 『IE Interfaces』, 제23권, 제3호(2010), pp.239-245.
- [4] Chan, Y.T. and K.C. Ho, “A Simple and Efficient Estimator for Hyperbolic Location,” *IEEE Transactions on Signal Processing*, Vol.42, No.8(1994), pp.1905-1915.
- [5] Chen, W., T. Mei, M.Q.-H. Meng, H. Liang, Y. Liu, Y. Li, and S. Li, “Localization Algorithm based on a Spring Model(LASM) for Large Scale Wireless Sensor Networks,” *Sensors*, Vol.8, No.3(2008), pp.1797-1818.
- [6] Cheung, K.W., H.C. So, W.-K. Ma, and Y.T. Chan, “A Constrained Least Squares Approach to Mobile Positioning : Algorithms and Optimality,” *EURASIP Journal on Applied Signal Processing*, Vol.2006(2006), pp.1-23.
- [7] Foy, W.H., “Position-Location Solutions by Taylor-series Estimation,” *IEEE Transactions on Aerospace and Electronic Systems*, Vol.AES-12, No.2(1976), pp.187-194.
- [8] IDTechEx, *RFID Forecasts, Players and Opportunities 2008~2018*, 2008.
- [9] ISO/IEC 24730-2, *Information Technology-Real-time Location Systems (RTLS)-part 2 :2.4GHz Air Interface Protocol*, ISO/IEC, 2006.
- [10] Mao, G., B. Fidan, and B.D.O. Anderson, “Wireless Sensor Network Localization Techniques,” *Computer Networks*, Vol.51, No. 10(2007), pp.2529-2553.
- [11] Ni, L.M., Y. Liu, and Y.C. Lau, “LANDMARC : Indoor Location Sensing Using Active RFID,” *Wireless Network*, Vol.10, No.6(2004), pp.701-710.
- [12] Torrieri, D.J., “Statistical Theory of Passive Location Systems,” *IEEE Transactions on Aerospace and Electronic Systems*, Vol.AES-20, No.2(1984), pp.183-198.