

RFID 시스템에서 이동체의 위치 추적을 위한 알고리즘

주운기^{*}

선문대학교 산업경영공학과

Algorithms for Localization of a Moving Target in RFID Systems

Un Gi Joo

Department of Industrial and management Engineering, Sun Moon University

This paper considers a localization problem of a moving tag on RFID(Radio Frequency Identification) systems, where a positioning engine collects TDOA(Time-difference of Arrival) signal from a target tag to estimate the position of the tag. To localize the tag in the RFID system, we develop two heuristic algorithms and evaluate their performance in the estimation error and computational time by using randomly generated numerical examples. Based upon the performance evaluation, we can conclude our algorithms are valuable for localization the moving target.

Keyword: RFID, TDOA, localization, moving target

1. 서론

본 논문은 실내외에서 이동하는 관심 대상체의 위치를 실시간으로 측정할 수 있는 알고리즘을 개발하는 것을 목적으로 한다. 이를 위해 본 논문에서는 실내에서도 적용이 가능하도록 RFID(Radio Frequency Identification)를 이용하는 RTLS(Real-Time Location Service) 방식을 고려하였고, RFID의 알려진 고정 위치의 리더 정보를 활용한 대상체(태그)의 위치 추정을 위해 삼각법 방식(triangulation)을 이용하였다. 또한 구현의 용이성과 비용을 고려하여 삼각법 중 TDOA(Time Difference Of Arrival) 방식을 이용한다. TDOA는 다른 기법에 비해 구현이 용이하고 저렴하게 측위할 수 있는 기법으로, ISO(International Standard Organization)에서는 ISO/IEC 24730-2에서 RTLS를 위한 리더와 태그 간 데이터 구조와 프로토콜(protocol)을 TDOA를 포함한 여러 측위 신호 방식을 이용할 수 있도록 표준화하였다(ISO/IEC, 2006).

TDOA 방식을 이용한 측위 문제는 일반적으로 다수의 관측치를 이용하여 소수의 변수(태그 위치) 값을 추정하는 문제가 되는데, Foy(1976)는 테일러 확장(Taylor series expansion) 기법을

이용하여 측위하였고, Chan and Ho(1994)는 태그 위치에 대한 MLE(Maximum Likelihood Estimator)를 이용하여 측위하는 방안을 제시하였다. 그리고 Cheung *et al.*(2006)은 LSE(Least Squared Estimator)를 이용한 측위 방안을 제시하였다. 그러나 테일러 확장 방식을 이용하는 측위 기법은 반복 연산을 위한 초기해를 어떤 것으로 선택하느냐에 따라서 최종적인 해의 성능 및 수렴 여부가 영향을 받는데, 좋은 초기해를 얻기가 어렵다는 문제가 있고, MLE는 관련 확률 분포를 알고 있는 경우에만 적용 가능한 기법이다. 그리고 LSE를 위해서는 역행렬(inverse matrix) 계산이 필요한데, 리더와 태그 배치 형태에 따라서는 역행렬이 존재하지 않는 경우도 있으므로 적용에 한계가 있다. 측위를 위한 메타 휴리스틱 알고리즘으로, Li and Wei(2005)는 유전(genetic) 알고리즘을 이용하여 측위하는 방안을 제안하였으나, 여기에서도 적합도(fitness) 함수를 위해 MLE를 이용하였다.

본 논문은 3차원 공간에서의 RFID 시스템에 대한 이동 태그 위치 추정에 대한 것으로, RFID 태그를 부착한 이동체가 미리 설치된 RFID 리더의 신호 측정 범위 내에 존재한다면 실내 및 실외 환경에 관계없이 태그 위치를 측정할 수 있는 알고리즘

이 연구는 2009년도 선문대학교 교내학술연구비 지원에 의하여 이루어졌음.

^{*}연락처 : 주운기 교수, 336-708 충남 아산시 당정면 갈산리 100 선문대학교 산업경영공학과, Fax : 041-541-7426,

E-mail : ugjoo@sunmoon.ac.kr

투고일(2009년 09월 29일), 심사일(1차 : 2010년 03월 31일), 게재확정일(2010년 04월 13일).

개발을 목적으로 한다. 이를 위해 본 논문은 TDOA 방식이 이용하여 측정된 정보에 대해, 가중 오차 제곱합(sum of squared error)과 벌과(penalty) 함수를 고려한 적합도 함수를 이용하는 두 가지 형태의 PSO(Particle Swarm Optimization) 알고리즘을 제안한다. 본 논문은 정적 PSO(sPSO : static PSO)와 동적 PSO(dPSO : dynamic PSO)의 두 가지 PSO 알고리즘을 위한 적합도 함수로 오차 제곱합을 이용함으로써 역행렬 계산의 필요성을 제거하였고, 관측치에 대한 확률 분포를 모르더라도 적용 가능하도록 하였다. 그리고 벌과 함수를 적합도 함수에 추가하여 관측치 간 독립성(independency)의 가정을 만족시키지 못하는 상황을 고려하였다. PSO 알고리즘은 1995년에 Kennedy and Eberhart가 개발한 것으로, 난수(random number)를 이용한 수치 탐색법의 일종이므로 역행렬 계산이 필요 없고, 다수의 개체(particle) 중 최선의 해(best solution)를 찾는 메타 휴리스틱 기법의 일종이다(Dong *et al.*, 2005). 따라서 본 논문에서 제안하는 측위 알고리즘은 관측치의 분포 특성 및 리더의 배치 특성에 관계없이 태그 위치를 추정할 수 있는 장점이 기대된다. 기존의 대부분의 연구는 정지 상태의 측위 문제에 대한 정확도를 높이는 연구가 주로 이루어 졌는데, 이를 위해서는 태그로부터 수신한 여러 번의 신호를 통해 추정된 여러 번의 위치 추정치들에 대해 평균값을 이용하는 방식을 취하고 있다. 그러나 이동 중인 객체의 측위를 위해서는 여러 번의 측위치에 대한 평균값을 취하는 방식은 정확도에 문제가 있다(Kim, 2006). 더구나, 이동하는 대상체(태그)의 현재 위치는 이전 시점에서의 위치에 영향을 받는다. 이러한 시간 별 태그 위치 간 연관성을 모사하기 위해, 본 논문에서는 sPSO 외에도 동적인 PSO(dPSO)을 제안하고, 성능 분석을 통해 유용성을 확인한다.

본 논문의 구성은 다음과 같다. 제 2장에서는 TDOA에 대한 수리적 모형을 수립한다. 제 3장에서는 측위를 위한 정적 PSO(sPSO)와 동적 PSO(dPSO)의 두 가지 PSO 알고리즘을 제시하며, 이에 대한 성능 분석은 제 4장에서 수행한다. 마지막으로, 제 5장에서는 본 연구의 의의와 추후 연구에 대해 살펴본다.

2. 측위 모형

RFID를 이용한 위치 추적 시스템은 태그, 리더(reader), 안테나, 그리고 측위 엔진(positioning engine) 또는 중앙 서버로 <그림 1>과 같이 구성된다. 위치 추정을 위해서는 리더를 통해 태그의 위치 정보를 수집한 후, 수집한 정보를 위치 계산에 필요한 정보로 변환하여 위치 값을 계산하는 과정을 거친다. 위치 정보의 수집은 TDOA의 경우 태그에서 주기적으로 보내오는 신호(120회/초)의 비행시간 정보를 리더가 수신하고, 수신된 정보를 이용하여 측위 엔진에서 측위를 한다.

M개의 리더(reader)를 이용하여 태그의 위치를 측정하는 RFID 시스템을 고려하자. 각 리더 i 는 3차원 공간상의 고정 위치 (x_i, y_i, z_i) 에 존재하고, 이들 리더의 위치는 모두 알려져

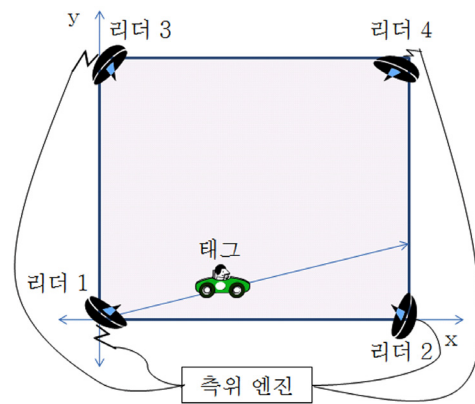


그림 1. RFID 측위 시스템

있다고 가정하자. (x, y, z) 에 위치하고 있는 태그가 위치 신호를 방송하면, 이 신호를 수신한 리더는 신호의 거리차를 계산하여 태그 위치 추정을 하는데 이용한다.

태그와 리더 i 간 직선 거리를 r_i 로 표시하면 r_i 는 다음과 같다.

$$r_i = \sqrt{(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2}, \quad i = 1, 2, \dots, M \quad (1)$$

태그에서 신호를 방송한 시점을 s 라 할 때, 리더 i 가 이를 수신하는 시점은 리더 i 와 태그 간 거리 r_i 에 비례하게 다음과 같이 t_i 로 정해진다.

$$t_i = r_i/c + s, \quad i = 1, 2, \dots, M \quad (2)$$

여기서 c 는 전파의 진행 속도를 나타내는 상수이다. 임의로 리더 1을 기준리더(reference reader)로 삼는 경우, 리더 i 와 기준 리더 1 간 신호 도달 시간차를 t_{i1} 라 하면, t_{i1} 는 다음과 같다.

$$t_{i1} = t_i - t_1 = (r_i/c + s) - (r_1/c + s) = (r_i - r_1)/c, \quad i = 2, \dots, M \quad (3)$$

즉, 리더 간 신호 도달 시간 차(TDOA) 정보를 이용하면, 변수 s 가 소거되므로 동기화의 필요성이 없어진다. 여기서 c 는 상수이므로, 시간차를 나타내는 t_{i1} 와 유사하게 거리차를 다음과 같은 r_{i1} 로 나타내기로 하자.

$$r_{i1} = r_i - r_1, \quad i = 2, 3, \dots, M \quad (4)$$

여기서 r_{i1} 는 이론적인 거리 차를 나타내는 것인데, 실제로는 측정 오류와 같은 오차가 개입되게 되므로 리더가 측정된 값이 이론적 수치와 동일한 값일 가능성은 거의 없다. 이러한 오차를 ϵ_{i1} 로 표현하고, 측정 거리 차를 d_{i1} 로 표시하면, 다음과 같이 표현된다.

$$d_{i1} = r_{i1} + \epsilon_{i1}, \quad i = 2, 3, \dots, M \quad (5)$$

식 (4)를 식 (5)에 대입하고, 오차가 없는 경우를 나타내는 관계식인 $(d_{i1} + r_1)^2 = r_i^2$ 을 전개하면 다음과 같은 관계식이 도출된다.

$$(1/2)(d_{i1}^2 - K_i + K_1) = -x_i x - y_i y - z_i z - d_{i1} r_1, i = 2, \dots, M \quad (6)$$

여기서 $K_i = x_i^2 + y_i^2 + z_i^2$, $x_{i1} = x_i - x_1$, $y_{i1} = y_i - y_1$, $z_{i1} = z_i - z_1$, $i = 1, 2, \dots, M$ 를 나타낸다.

식 (6)은 미지의 변수인 x, y, z 외에 r_1 의 함수인데, 본 논문의 목적은 $M-1$ 개의 주어진 식에 대해, 4개의 미지의 변수 값을 구하는 것 이므로, 식 (1)에서 보는 바와 같이 r_1 은 변수 x, y, z 와 비 선형의 관계를 갖는 문제이다. 따라서 $M \leq 4$ 이면 무수히 많은 해가 존재하는 문제가 되지만, 반대로 $M > 4$ 이면 모든 관계식을 만족시키는 해가 존재하지 않을 수 있다. 따라서 일반적으로 LSE(Least Sum of squared Estimator)나 MLE(Maximum Likelihood Estimator)를 계산하여 위치 추정치로 이용한다.

측위를 위한 측정 과정에서의 오차를 포함시켜서 식 (6)을 표현하면 식 (7)과 같다.

$$Y = AX + E, \quad (7)$$

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

$$A = (-1) \begin{pmatrix} x_{21} & y_{21} & z_{21} & d_{21} \\ x_{31} & y_{31} & z_{31} & d_{31} \\ \vdots & \vdots & \vdots & \vdots \\ x_{M1} & y_{M1} & z_{M1} & d_{M1} \end{pmatrix}, X = \begin{pmatrix} x \\ y \\ z \\ r_1 \end{pmatrix}, E = \begin{pmatrix} \xi_{21} \\ \xi_{31} \\ \vdots \\ \xi_{M1} \end{pmatrix} \text{ 이고, } \xi_{i1} \text{ 는}$$

식 (7)에서의 오차를 나타낸다.

식 (7)에서의 X 의 추정량으로 식 (8)을 이용할 수 있다.

$$(A^T Q^{-1} A)^{-1} A^T Q^{-1} Y \quad (8)$$

여기서 Q 는 오차(ϵ_{i1})들의 공분산(covariance) 행렬로, RFID 운영환경에 따라 결정되는 상수 값이다. 식 (8)은 오차 E 가 평균이 0이고 공분산이 Q 인 정규분포를 따르는 서로 독립인 확률 벡터인 경우는 MLE가 되고, 가중행렬이 Q^{-1} 인 LSE이기도 하므로, 측위를 위해 대부분의 기존 문헌에서는 식 (8)을 이용하였다(Foy, 1976; Chan and Ho, 1994; Cheung et al., 2006). 그러나 식 (8)을 이용하여 측위를 하기 위해서는 행렬 Q 와 $A^T Q^{-1} A$ 의 역행렬이 존재하는 상황에서만 이용 가능하다는 문제가 있다. 태그와 리더의 위치에 따라 역행렬의 존재 여부가 정해지지만, 주어진 RFID 시스템 환경이 역행렬 계산이 가능한 환경인 지 여부를 사전에 확인하기 어렵기 때문에 식 (8)을 이용한 측위 알고리즘은 한계가 있다. 따라서 본 논문에서는 역행렬을 이용하지 않는 측위 알고리즘을 개발한다.

본 논문의 목적은 주어진 측정 거리 차 $\{d_{i1}\}$ 를 이용하여 태그 위치 (x, y, z) 에 대해 최소의 오류를 갖는 추정치를 구하는 것이다. 기존의 대부분의 문헌에서는 고정 위치의 태그 측위 문제를 다루었고, 이동하는 태그의 측위를 위해서는 고정 위치 태그 측위 알고리즘을 반복해서 적용하는 방식을 이용하였다. 그러나 이동하는 태그의 경우, 태그 위치가 이동 가능 영역의

어느 곳에나 존재할 수 있으므로 모든 존재 가능한 위치의 태그 측위를 효과적으로 할 수 있는 방안이 필요하다. 본 논문에서는 이동하는 동적인 태그의 특성을 반영할 수 있도록 동적인 알고리즘(동적 PSO)을 포함한 PSO(Particle Swarm Optimization) 알고리즘을 개발한다. PSO 알고리즘은 여러 개체(particle)를 이용한 군집(swarm)을 이용한 수치적인 최적해 탐색 방법이므로, 앞에서 언급한 역행렬 계산이 필요 없으므로 태그와 리더 간 상대적 위치에 관계없이 해를 구할 수 있는 장점이 있다. 그리고 여러 개체 중 최선의 목적함수 값을 갖는 개체를 찾는 방식이므로, 군집 중 최선의 해를 보장한다.

3. 측위 알고리즘

생산 현장에 있는 주요 설비나 자재, 또는 중요한 사람의 이동 경로 이동체의 이동 특성을 사전에 알 수 있다면 이를 측위에 활용할 수 있을 것이지만, 이동 경로가 무작위로 이루어지는 경우도 있으므로 이를 미리 알아내기가 쉬운 것은 아니다. 이와 같이 이동이 가능한 객체의 위치 추적을 위해서 본 논문에서는 두 가지 형태의 PSO 알고리즘을 개발한다. 첫 번째는 기본적인 PSO로, 측정 거리 차 $\{d_{i1}\}$ 에 대해 독립적으로 군집(swarm)을 구성하여 측위하는 방식이고, 다른 하나는 이전의 군집에 대한 탐색 결과를 이용하여 연속적인 측위를 하는 형태이다. 전자는 각 측위를 위해 주어진 정보만 이용하는 형태로 정적 PSO(static PSO)라 부르고, 후자를 동적 PSO(dynamic PSO)라 부르기로 하자.

PSO의 장점 중 하나는 문제 구조(특성)에 관계없이 다음과 같은 절차를 이용하여 수치적인 방법으로 해를 구할 수 있다는 점이다.

단계 1(s): 무작위로 N 개의 개체 위치를 발생시킨다. 여기서 각 위치는 문제에 대한 하나의 해가 된다.

단계 2: 발생시킨 각 개체를 적합도 함수(fitness function)를 이용하여 평가하여, 최선의 적합도를 갖는 개체를 찾고 이를 $gBest$ 값으로 둔다. 여기서 $gBest$ 는 현재까지 발생된 모든(global) 개체 위치 중에서 최선(최소화 문제의 경우는 최소의 적합도 함수 값을 가지는)의 해를 나타낸다. 또한 각 개체 i 별 현재까지의 최선의 위치를 $lBest_i$ 로 두는데, 초기 단계에서는 현재의 각 개체의 위치 값이 $lBest_i$ 가 된다. $lBest_i$ 는 현재까지 개체 i 를 위해 발생한 위치 중에서 최선의 위치를 나타낸다. 만약, 원하는 횟수(ITER)를 반복 수행하였거나 해의 개선이 없는 채로 일정 개수의 반복이 이루어졌다면, 단계 5를 수행하고, 그렇지 않다면 단계 3을 수행한다.

단계 3: 현재의 각 개체의 위치를 난수(random number)를 이용하여 다음과 같이 변경시킨다.

개체 i 의 다음 위치 = 개체 i 의 현재 위치 + 개체 i 를 위한 변경량

단계 4: 변경된 각 개체 별 위치에 대해, 단계 2를 수행한다.

단계 5: 현 단계의 $gBest$ 를 측위 추정치로 선택한다.

태그의 신호를 수신한 리더는 Y 와 A 를 측위 엔진에 전송하면, 측위 계산 모듈은 앞의 PSO 알고리즘을 이용하여 측위를 한다. 측위를 위해 PSO 알고리즘에서 결정해야 하는 사항은 단계 1에서의 개체의 표현방법과 발생하는 개체의 수(N), 단계 2에서의 적합도 함수의 형태, 단계 3에서의 변경 분량 결정을 위한 방식 등이다.

먼저, 본 논문의 목적이 x, y, z, r_1 값을 추정하는 것이므로, 각 개체 i 를 ($p_i^x, p_i^y, p_i^z, p_i^{r1}$)로 표현하기로 한다, $i = 1, 2, \dots, N$, 여기서 N 값의 크기는 성능에 영향을 줄 수 있는데, 이에 대한 분석을 제 4장에서 실시한다. 다음으로, 단계 2에서 필요한 적합도 함수는 본 논문의 목적이 오류가 최소가 되는 위치의 추정치를 구하는데 있으므로, 다음의 함수 F_i 를 개체 i 를 위한 적합도 함수로 선택하여 최소화를 추구하였다.

$$F_i = (E^T QE)_i + (\hat{r}_1 - r_1)^2 \quad (9)$$

여기서 Q 는 오류 벡터 E 의 공분산 행렬로, E 의 행의 수가 $M-1$ 이므로 Q 는 $(M-1) \times (M-1)$ 이다. 따라서 $(E^T QE)_i$ 는 태그가 개체 i 의 위치에 있다고 할 때의 가중 오차 제곱합을 나타낸다. 본 논문의 목적은 태그의 실제 위치와 차이가 적은 추정치를 구하는 것이지만, 태그의 정확한 실제 위치는 알 수가 없으므로, 적합도 함수를 위해서 오차에 대한 제곱합을 취하였다. 여기서, 추정된 위치 값 $(\hat{x}, \hat{y}, \hat{z})$ 은 관계식 $r_i^2 = K_1 - 2x_1x - 2y_1y + x^2 + y^2$ 을 만족하지 않을 수 있으므로, 추정 결과 얻어진 거리 값인 \hat{r}_1 과 계산을 통해 얻어지는 거리 $\hat{r}_1, \hat{r}_1 = \sqrt{(\hat{x} - x_1)^2 + (\hat{y} - y_1)^2 + (\hat{z} - z_1)^2}$ 간 차이가 있을 수 있다. 이러한 차이는 0인 것이 최선이므로 이를 벌과 함수(penalty function)로 하여 적합도 함수의 두 번째 항으로 $(\hat{r}_1 - r_1)^2$ 을 고려하였다. 마지막으로, 단계 3을 위한 개체의 위치 변경을 위해서는 Dong et al.(2005)이 분석한 다음과 같은 식을 이용하였다.

$$p_i^j = p_i^j + V_i^j, \quad i = 1, 2, \dots, N; j=x, y, r1 \quad (10)$$

여기서, $V_i^j = w * V_i^j + 2(R1)(LBest_i^j - p_i^j) + 2(R2)(gBest_i - p_i^j)$ 이고, $w = 1.2 - 0.8(\text{반복 횟수})/ITER$ 이다. 그리고 $ITER$ 은 최대 반복 횟수를 나타내고, p_i^j 는 개체 i 의 위치를 나타내며, V_i^j 는 개체 i 를 위한 변경량(velocity)을 나타낸다. 변경량 V_i^j 는 모수 w 와 $R1, R2$ 의 함수로, w 는 (0.4, 1.2) 구간에서 선형으로 감소하는 함수 형태를 이용하였다. $R1$ 과 $R2$ 는 (0, 1) 구간의 일양분포(uniform distribution)를 따르는 난수를 나타내는 것으로, 다른 메타휴리스틱(meta-heuristic) 알고리즘과 유사하게 해(solution) 탐색

과정에서 임의성을 주는 역할을 한다.

본 논문은 앞에서 기술한 바와 같은 정적 PSO(sPSO)와 동적 PSO(dPSO)의 두 가지 형태의 PSO 알고리즘을 고려하였는데, 전술한 PSO 알고리즘을 sPSO(정적 PSO) 알고리즘이라고 하기로 한다. 추가로, 이동체의 동적인 상황을 고려하기 위해 동적인 PSO(dPSO)를 개발하였는데, dPSO는 sPSO의 단계 1(s)대신 다음의 단계 1(d)를 이용하는 것만 차이가 있고 나머지 단계는 동일한 것으로 구성하였다.

단계 1(d): 측위를 처음으로 시작하는 경우는, 무작위로 N 개의 개체 위치를 발생시킨다. 그러나 처음으로 측위하는 경우가 아니라면, 각 개체의 위치는 이전 측위 과정에서의 결과인 단계 5에서의 지역 최선해(local best solution) $lBest_i^j, i = 1, 2, \dots, N; j = x, y, r1$ 로 선택한다.

추적 대상이 계속해서 이동하는 상황을 고려하므로, 리더가 정보를 수신한 시점에서의 최선의 측위 결과인 $lBest_i^j$ 를 이동체가 위치 이동을 한 후의 측위를 위해서 사용하는 방안을 고려할 수 있다. 이러한 개념 하에서 단계 1(d)를 이용하는 dPSO를 개발하였다. 즉, 동적 PSO는 정적 PSO의 단계 1(s)에서 N 개의 해를 무작위로 발생하는 대신, 이전 단계의 군집에 대한 최선의 결과 값을 사용하는 것만 차이가 있다. dPSO에서는 이동체의 초기 위치 추정을 위해서는 N 개의 무작위 군집을 이용하지만, 그 다음부터는 현 단계의 군집을 이용하여 측위하는 방식을 통해 측위 정확도 개선을 하고자 한다.

4. 성능 평가

제 3장에서 제안한 알고리즘은 3차원 공간상에서의 측위를 위해 이용 가능하지만, 성능 분석을 위해서는 편의상 $z = 0$ 로 갖는 2차원 공간에서의 측위 문제를 고려하였다. 태그는 측정 영역 내에서 임의의 속도로 위치 변경을 하면서, 자신의 측위 신호(신호 형태는 ISO/IEC에 의해 규정되었음)를 방송한다. 방송된 신호를 수신한 리더들은 태그와의 거리 차를 계산하여 태그의 위치를 추정하게 되는데, 여기에서는 태그가 다음의 선형식을 따라서 이동하는 경우를 고려하였다.

$$y = ax + b \quad (11)$$

여기서 (x, y) 는 신호 수신 시점(또는 측위시점)에서의 태그의 위치를 나타내고, a 와 b 는 태그의 이동경로에 따라 결정되는 상수이다. 성능 분석을 위해서 선형의 이동경로를 가정하였지만, 태그의 이동 경로가 선형이 아니다 하더라도 태그가 보내는 신호의 방송 간격이 매우 작으므로(2.4GHz 시스템에서는 638ms이하), 그 기간 동안의 이동 경로를 선형으로 간주한다고 하더라도 그 오차는 작지 않을 것이기 때문에 선형식에 따른

태그 이동의 가정이 현실과 큰 차이가 있지는 않을 것이다.

태그는 ISO/IEC(2006)에서 권고한 300m 신호 도달 거리를 고려하여 300m×300m의 정사각형 구역 내에서 이동한다고 가정하자. 이러한 구역 내의 태그 위치 감지를 위해서 4개의 리더가 각각 $(x_1, y_1) = (0, 0)$, $(x_2, y_2) = (300, 0)$, $(x_3, y_3) = (0, 300)$, $(x_4, y_4) = (300, 300)$ 에 고정적으로 존재하는 경우에 대한 실험을 하였다. 태그에서의 신호에 대한 측정 오차의 공분산 Q 는 오차의 분산을 σ^2 라 할 때, Chan and Ho(1994)와 같이 대각원소는 모두 σ^2 이고, 그 외의 원소는 모두 $0.5\sigma^2$ 인 것을 고려하였다. 이동하는 태그가 보내온 신호를 리더가 수신할 때마다 태그의 위치를 추정하는데, 추정 결과 값인 (\hat{x}, \hat{y}) 의 성능분석을 위한 척도로는 태그의 참값 (x, y) 에 대한 평균제곱 오차 MSE (Mean Squared Error)를 이용하였다.

$$MSE = \sum_{j=1}^q [(\hat{x} - x)^2 + (\hat{y} - y)^2] / q \quad (12)$$

여기서 q 는 총 측위 횟수를 나타낸다. MSE 는 2차원 평면상에서의 태그의 실제 위치와 추정치 간 거리차이의 제곱평균을 나타낸다. 따라서 \sqrt{MSE} 는 평균적인 오차 거리를 나타내므로 측위 정확도의 평가 기준으로 이용하였다.

측위를 위한 PSO 알고리즘은 이용하는 개체(particle) 수(N)와 최대 반복수($ITER$) 값 등에 영향을 받을 수 있다. 따라서 이들 파라메타에 대한 성능의 영향을 살펴보기 위해 Visual C++을 이용하여 2.8GHz Intel Core2 Duo PC에서 성능 분석을 하였다.

먼저, 파라메타 값 설정을 위한 사전 실험을 실시하였다. 본 사전 연구는 각 파라메타의 값을 얼마로 할지에 대한 것 뿐 아니라 각 파라메타 값 변화에 따른 성능 변화 추세를 파악하는 것에도 목적이 있기 때문에, 결정하고자 하는 하나의 파라메타 값만 변경시키면서 성능 비교를 하는 방식을 취하였다. 첫째로, 개체 수에 따른 성능 변화를 위해, $N = 10, 20, 30, 40, 50$ 인 경우에 대한 성능 변화를 살펴보았는데, 300m×300m의 2D 공간에서 태그는 원점에서 출발하여 선형식 $y = 0.3x$ 에 따라 단위 시간당 x 축 방향으로 10m씩 태그가 이동하는 경우를 고려하자. 10m 이동 후 태그는 자신의 위치 신호를 방송하면, 300m×300m 각 구석 꼭지점에 위치하고 있는 4개의 리더가 이를 감지하여 측위 엔진 내에 있는 PSO를 이용하여 태그 위치를 추정한다. 여기서, 측정 오류의 분산은 $\sigma^2 = 0.001$ 이라고 가정하자(분산을 0.001로 고정하였지만, 분산 값의 변화에 따른 성능 분석은 제4장의 마지막 부분에서 이루어진다). PSO를 위한 개체 수에 따른 성능분석을 위해 최대 반복횟수를 $ITER = 5,000$ 로 하여, N 에 따른 성능 분석을 한 결과가 <그림 2>이다.

첫 번째 평가의 목적이 N 에 따른 성능 변화의 특성 파악에 있으므로, <그림 2>에서는 MSE 나 time(초) 대신 $10\log(MSE)$ 와 $10\log(\text{time})$ 를 각각 오차와 처리시간 척도로 하여 표시하였다. 성능 평가 결과, 일반적으로 N 이 증가함에 따라 계산시간은 N 에 선형으로 증가하지만 $N = 30$ 이어서도 0.1초($10\log 0.1 = -10$) 정도의 시간 내에 측위를 하였고, 측정 오류는 N 이 20이상에서는

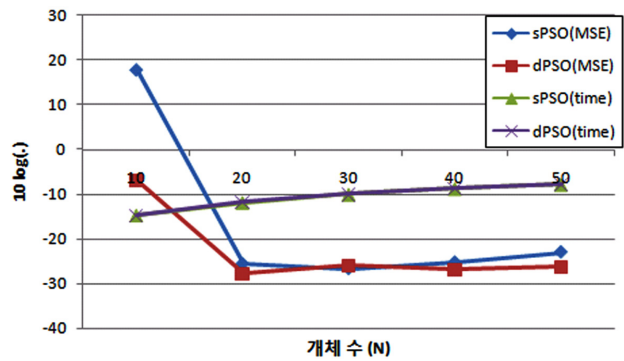


그림 2. 개체 수에 따른 오차 및 계산시간

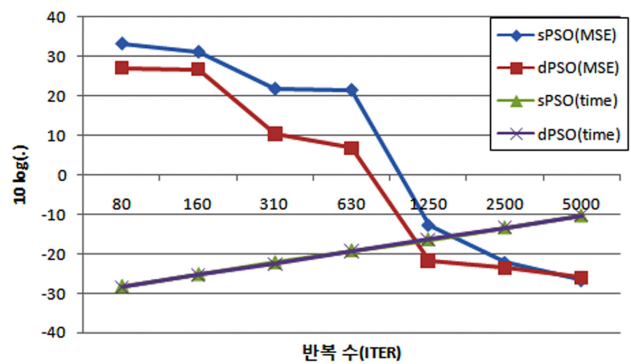


그림 3. 반복수에 따른 성능

큰 차이가 없었다. 그리고 큰 차이는 아니지만, 동일한 N 에 대해서는 dPSO가 sPSO보다 약간 MSE 가 작으므로 이동체의 측위를 위해 dPSO가 유용함을 알 수 있다. 이 분석 결과를 이용하여 나머지 성능 분석을 위해서는 $N=30$ 인 값으로 고정하여 분석을 실시하였다.

둘째로, 반복시행 횟수 $ITER$ 에 따른 성능을 분석한 결과는 <그림 3>과 같다. 실험 환경은 최대 반복수를 5000으로 고정하지 않고 변동시키면서 성능 평가를 수행한 것 외에는 $N=30$ 인 첫 번째의 실험 환경과 동일하다. <그림 3>에서 보는 바와 같이, dPSO는 전 단계의 최선의 해를 이용하여 해 탐색을 시작하므로, 반복횟수가 작은 경우에는 sPSO보다 MSE 가 작은 값을 갖는다. 그러나 반복횟수가 5,000일 때는 sPSO와 dPSO간 성능에 큰 차이가 없음을 알 수 있다. 즉, 예상되는 바와 같이, 적은 반복을 통해 빠른 시간 내에 측위를 하고자 하는 경우는 dPSO가 sPSO보다 오류가 적은 측위를 할 수 있음을 알 수 있다. 이 분석결과를 이용하여 이후의 실험에서는 반복수에 따른 sPSO와 dPSO간 차이를 줄이기 위해서 최대 반복수 $ITER$ 을 5,000으로 고정하여 실시하였다. 셋째로, 태그 이동속도에 따른 평가를 위해 단위 시간 당 x 축 방향의 이동 거리를 2, 4, 6, 8, 10, 12, 14, 16, 18, 20으로 변화하면서 측위 정확도를 MSE 값으로 평가한 결과가 <그림 4>와 같다. 이를 위한 실험 환경은 x 축 방향의 이동 속도를 10m대신 여러 가지로 변경시키면서 실험한 것 외에는 $ITER=5,000$ 인 두 번째 실험환경과 동일하다. 분석 결

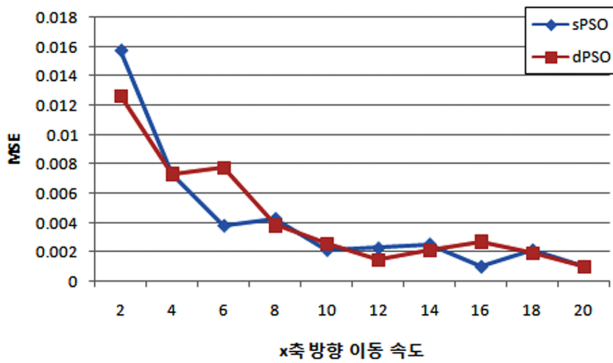


그림 4. 태그 이동속도에 따른 측정 오차

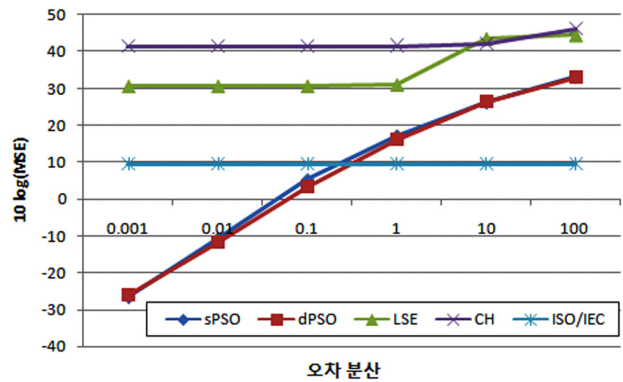


그림 5. 측정 오차 분산에 따른 성능

과, 이동 속도가 빠를수록 정확도가 높은 측위를 할 수 있음을 보인다. dPSO는 이전 단계의 최선의 해를 활용하는 것이므로, 이전 단계의 태그와 이동 후의 태그 위치에 큰 차이가 없다면(즉, 이동 속도가 느리다면) sPSO보다 정확도가 높은 측위를 할 것으로 기대하였다. 그러나 이동속도에 따라 변동은 있지만, 전체 평균 오차는 sPSO의 경우 0.0042이고, dPSO는 0.0043이었다. 따라서 이동속도에 따른 dPSO와 sPSO간 오차에는 큰 차이가 없는 것으로 나타났다. 따라서 향후의 실험에서는 x방향의 이동 속도가 10m인 경우로 고정하여 실험을 실시하였다. 이와 같은 사전 실험 결과를 이용하여, 이동경로에 따른 성능과 측정 오류의 분산 값의 변동에 따른 성능 분석을 실시하였다. 본 논문에서 제안한 PSO 알고리즘의 상대적 성능 비교를 위한 대상으로, 측위를 위해 많이 이용하는 식 (8)의 LSE(LSE로 표시)와 Chan and Ho(1994)의 해(CH로 표시)를 이용하였다. 네 번째 실험은 태그의 이동 경로에 따른 성능 분석을 위한 실험으로, 다음의 6가지 태그 이동 경로를 고려하였다 : $y = 50; y = 100; y = 150; y = 0.1x, y = 0.3x, y = 0.5x$. 이를 위한 다른 파라메타는 첫 번째 실험분석 환경과 동일하게 하였다. 즉, 이동궤적이 $y = 0.3x$ 외에 다른 형태로 변하는 것 외에는 이동속도가 x축 방향으로 10m인 세 번째 실험 조건과 동일한 환경을 고려하였다. 이동궤적을 나타내는 6가지 a 와 b 값에 대해 MSE를 비교한 결과는 <표 1>과 같다.

성능 분석 결과, PSO(sPSO, dPSO)의 정확도가 LSE나 CH보다 좋았다. LSE와 CH는 태그가 리더의 중앙 부근에 위치하는 경우에는 오차가 크지 않았으나, 태그가 탐색 지역의 경계점 부근에 위치하는 경우에는 매우 큰 오차가 발생하였다. 따라서 이러한 성능 악화가 LSE나 CH의 매우 큰 MSE를 유도한 것으로

보인다. 반면에, sPSO와 dPSO는 오차가 매우 작은 추정치를 찾아주므로, 이동 궤적에 관계없이 유용하게 사용할 수 있을 것으로 판단된다.

마지막으로, 측위를 위한 측정 데이터의 오차(분산)에 따른 측위 오차를 평가하기 위해, 분산 값을 0.001, 0.01, 0.1, 10, 100로 변화시키면서 평균 $10\log(MSE)$ 값의 변화를 분석 하였다. 이를 위한 실험 환경은 이동궤적이 $y = 0.3x$ 인 네 번째 실험환경과 동일한 것을 고려하였으나, 측정 오류 분산을 여러 가지로 변경해가면서 실험하였다. 분석 결과 PSO가 LSE나 CH에 비해 정확도가 높은 측위를 하였고, dPSO가 sPSO보다 정확도가 약간 높음을 확인할 수 있다. <그림 5>에는 ISO/IEC에서 권고한 오차 한계인 $3m(10\log 9 = 9.54)$ 를 ISO/IEC로 추가로 표시하였다. 분석 결과, 분산이 0.1인 경우에는 sPSO의 \sqrt{MSE} 가 $1.82m(10\log MSE = 5.21)$, dPSO의 \sqrt{MSE} 가 $1.45m(10\log MSE = 3.23)$ 가 되어서 ISO/IEC 규격에도 만족되는 측위 정확도를 가지는 것을 볼 수 있으므로, PSO는 이동객체의 측위에 유용한 알고리즘임을 확인할 수 있다. 예상했던 바와 같이, 오차의 분산이 클수록 측위 정확도가 낮아지는 것을 확인 할 수 있다. 이는 데이터의 정확도가 분산이 클수록 낮아지기 때문이다. 분산에 따른 민감도 측면에서 보면, LSE와 CH가 오차 분산에 대한 둔감성이 PSO보다 좋았지만, 작은 MSE를 갖는 것이 우선적으로 요구되는 사항이므로, 이동 객체의 측위를 위해서는 PSO에 비해 유용하지 않음을 알 수 있다.

본 논문에서 개발한 sPSO와 dPSO의 다양한 파라메타 값 변화에 따른 성능 평가 결과, PSO가 기존의 LSE나 CH 방안보다 정확도가 높은 측위를 할 수 있다는 것을 확인하였고, 개체의

표 1. 태그 이동 궤적 $y = ax+b$ 에 따른 오차

| | 0 | 0 | 0 | 0.1 | 0.3 | 0.5 |
|------|------------|------------|-------------|------------|------------|------------|
| a | | | | | | |
| b | 50 | 100 | 150 | 0 | 0 | 0 |
| LSE | 1025.6651 | 164.4051 | 767986.1513 | 2249.8318 | 1167.9391 | 480.7984 |
| CH | 13331.7017 | 13646.1278 | 185415.1934 | 14083.8396 | 14100.7053 | 15312.3444 |
| sPSO | 0.0027 | 0.0016 | 0.0008 | 0.0033 | 0.0022 | 0.0009 |
| dPSO | 0.0033 | 0.0018 | 0.0008 | 0.0042 | 0.0026 | 0.0011 |

수나 반복수가 작을 경우에는 dPSO가 sPSO보다 정확도가 높은 측위 값을 제공할 수 있음을 보였다. 이러한 분석 결과를 통해서, 본 논문에서 제안하는 PSO 알고리즘이 제한된 영역에서의 이동하는 객체(태그)의 위치 추적을 위해 유용하게 이용될 수 있다고 판단한다.

5. 결 론

본 논문은 이동하는 객체 위치의 실시간 확인을 위한 알고리즘을 제안하는 것으로, RFID 시스템에서 이동하는 태그 위치를 추정하기 위해 TDOA 방식을 이용하는 경우를 다루었다. 이동하는 객체의 측위 문제는 고정위치의 태그 측위를 위한 알고리즘을 이용하여 할 수 있으나, 고정 위치를 위한 측위 알고리즘이 태그의 위치나 측위 환경 파라메타 값에 따라 정확도에 변동이 크므로, 태그 위치 변화에 둔감한 측위 알고리즘이 필요하다. 본 논문은 측위를 위해 sPSO와 dPSO의 두 가지 PSO 알고리즘을 제안하였고, 제안한 알고리즘의 우수성을 보이기 위해 다양한 수치 예를 이용한 성능 평가를 하였다. 성능 분석 결과 PSO는 이동하는 태그에 대해 기존의 알고리즘(LSE와 CH)에 비해 정확도가 높은 측위 값을 제공하고, 특히 빠른 계산을 위해 개체 수나 반복수를 작게 할 필요가 있는 상황에서도 dPSO는 정확도가 높은 측위를 위해 활용가능하다는 것을 확인하였다. 본 논문에서 제안한 dPSO는 기존의 문헌에서는 발표된 바가 없는 PSO 알고리즘의 형태로, 연속적인 해 탐색이 필요한 다른 문제 상황에서도 적합도 함수의 수정을 거쳐서 사용할 수 있을 것이다.

본 연구에서는 TDOA 방식의 측위에 한정하였지만, 정확도 향상을 위해서는 AOA와 같은 방법이나 GPS 정보를 혼용하는 측위 기법을 개발할 필요가 있다. 이러한 혼합 형태(hybrid) 측

위 기법은 RFID 시스템의 적용 영역을 넓히는 역할을 할 것으로 기대되고, 혼합 측위 기법에 대한 PSO 알고리즘의 개발도 추후 연구 과제 중 하나이다. 본 연구에서 제안하는 PSO 알고리즘은 태그의 위치에 관계없이 기존 알고리즘보다 측위의 평균 오차(MSE)가 작은 측위값을 제공함을 보였으나, 오차의 분산 값에는 비교 대상 알고리즘에 비해 민감한 것으로 분석되었다. 따라서 분산에 둔감한 측위 알고리즘 개발도 필요한 상태이다.

참고문헌

- Chan, Y. T. and Ho, K. C. (1994), A Simple and Efficient Estimator for Hyperbolic Location, *IEEE Transaction on Signal Processing*, 42(8), 1905-1915.
- Cheung, K. W., So, H. C., Ma, W.-K., and Chan, Y. T. (2006), A Constrained Least Squares Approach to Mobile Positioning : Algorithms and Optimality, *EURASIP Journal on Application Signal Processing*, 1-23, www.ee.cityu.edu.hk/~hcs/asp_06.pdf.
- Dong, Y., Tang, J., Xu, B., and Wang, D. (2005), An Application of Swarm Optimization to Nonlinear Programming, *Computers and Mathematics with Applications*, 49, 1655-1668.
- Foy, W. H. (1976), Position-location Solutions by Taylor-series Estimation, *IEEE Transaction on Aerospace and Electronic Systems*, AES-12(2), 187-194.
- ISO/IEC 24730-2 (2006), *Information Technology-Real-Time Locating Systems (RTLS)-Part 2 : 2.4 GHz Air Interface Protocol*, ISO/ICE.
- Kim, H. (2006), Speed-adaptive Location Estimation Approach to a Wireless LAN-based RTLS System, *Telecommunications Review*, 16(4), 580-590.
- Li, L. and Wei, F. (2005), Position Estimation by Improved Genetic Algorithm for Hyperbolic Location, *14-th Mobile & Wireless Communications Submit*, www.eurasip.org/Proceedings/Ext/IST05/papers/122.pdf.
- Nanotron technologies GmbH (2006), *Real Time Location Systems (RTLS)*, nanotron.com/EN/pdf/WP_RTLS.pdf.
- Torrieri, D. J. (1984), Statistical Theory of Passive Location Systems, *IEEE Transaction on Aerospace and Electronic Systems*, AES-20(2), 183-198.



주 운 기

한국과학기술원 산업공학과 박사
 한국전자통신연구원 선임/초빙연구원
 현재 : 선문대학교 산업경영공학과 교수
 관심분야 : 통신 시스템, 생산시스템, 산업정보화