

## The Motion Estimation of Caterpillar-type Mobile Robot Using Robust SLAM

변성재\* · 이석규\*\* · 박주현†  
(Sung-Jae Byun · Suk-Gyu Lee · Ju-Hyun Park)

**Abstract** - This paper proposes a robust method for mapping of a caterpillar-type mobile robot which inherently has uncertainty in its modeling by compensating for the estimated pose error of the robot. In general, a caterpillar type robot is difficult to model, which results in inaccuracy in Simultaneous Localization And Mapping(SLAM). To enhance the robustness of the SLAM for a caterpillar-type mobile robot, we factorize the SLAM posterior, where we used particle filter to estimate the position of the robot and Extended Kalman Filter(EKF) to map the environment. The simulation results show the effectiveness and robustness of the proposed method for mapping.

**Key Words** : SLAM(Simultaneous Localization and Mapping), Particle Filter, EKF(Extended Kalman Filter), Kinematics uncertainty, Autonomous mobile robot

### 1. 서론

이동 로봇이 주어진 작업을 수행하기 위해서는 기본적으로 주변 환경에 대한 정확한 인식(mapping)과 자기 위치 추정(localization)이 필요하다. 이러한 주변 환경에 대한 지도 작성과 로봇의 자기 위치 추정은 상호 의존적인 관계에 있기 때문에 이들은 동시에 고려되어야 하며, 이를 SLAM(Simultaneous Localization and Mapping)이라한다. 레이저, 카메라, 초음파 센서 등과 같은 다양한 센서를 단독 혹은 융합적으로 이용한 SLAM에 관한 연구는 매우 활발하게 진행되고 있으나, 아직 괄목할만한 성과는 나타내지 못하고 있으며, 주어진 상황에 따라 상대적으로 적절한 알고리즘이 개발되고 있다 [1,6].

본 논문에서는 SLAM에 필요한 로봇의 시스템 모델링이 비교적 어려운 무한궤도형 로봇의 기구학적 문제를 해결하기 위한 방법을 제시하고 시뮬레이션을 통하여 제안한 방법의 타당성을 기술한다. 본 논문에서 위치 추정기로 사용될 파티클 필터에서는 주행환경에서 로봇이 위치할 확률을 샘플의 분포로 나타낸다 [4,5]. 다수의 샘플을 사용하고 적절한 센서 모델을 사용하여 로봇이 위치할 가능성이 높은 지역에 샘플이 수렴하도록 한다. 파티클 필터는 비교적 정확한 로봇의 위치 추정 결과를 보여주고 있기 때문에, 전역 위치 추정, kidnapped robot 문제 등을 현재까지 가장 잘 해결할 수 있는 방법이다. 본 논문에서는 SLAM 문제를 해결하는데

파티클 필터를 사용하였다.

본 논문의 구성은 다음과 같다. 2장에서는 SLAM에서의 로봇의 위치추정과 환경 지도 작성에 대한 알고리즘과 그에 대한 수학적 표현을 다루며, 3장은 실제 로봇에 대한 기구학적인 문제와 이를 보정할 수 있는 방법에 대해 설명하였다. 4장에서는 실험결과를 기술하였고, 5장에서는 결론 및 향후 연구 과제에 대하여 언급하였다.

### 2. SLAM

#### 2.1 위치추정

로봇의 위치는 카메라, 초음파 센서, RFID 등과 같은 외부 센서와 구동장치에 부착된 엔코더와 같은 로봇 내부 센서를 이용하여 추정할 수 있다. 그러나 내부 혹은 외부 센서 단독으로는 정확한 위치를 추정하는 것은 거의 불가능하다. 실제로 로봇 자체의 기계적인 오차 뿐 아니라 외부 환경에 대한 센서 오차가 존재하며, 로봇은 이러한 내외부적인 오차에 의해 주행에 심각한 영향을 받게 된다. 또한 부정확한 로봇의 자세와 위치에 의해 작성된 환경 정보 또한 부정확하게 되어 로봇의 이동거리가 길어지면 그 오차는 누적되어 그 불확실성이 더욱 확대되는 상황에 이르게 된다.

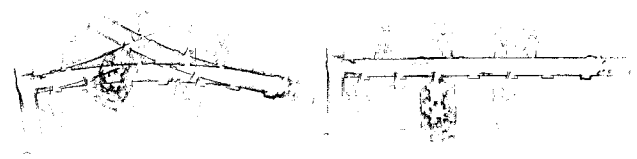


그림 1 위치추정 오차와 지도 작성 오차간의 관계  
Fig. 1 The relationship between the estimated localization error and the map building error

\* 준회원 : 영남대학 전기공학과 석사졸업  
\*\* 정회원 : 영남대학 전기공학과 교수  
† 교신저자, 정회원 : 영남대학 전기공학과 교수 · 공박  
E-mail : jessie@ynu.ac.kr  
접수일자 : 2008년 8월 28일  
최종완료 : 2009년 2월 23일

Fig. 1은 레이저 스캐너를 사용하여 로봇의 환경지도를 작성한 결과를 나타내고 있다 [1]. 왼쪽 그림은 로봇의 위치 추정에 대한 보정 없이 환경 지도를 작성한 결과이며, 오른쪽의 그림은 로봇의 위치 추정에 대한 보정이 이루어졌을 때의 환경 지도 작성의 결과이다. Fig. 1에서 보는 바와 같이 위치 보정을 한 오른쪽 결과가 왼쪽 그림과 비교하여 그 정확도 높다.

2.1.1 위치 추정의 초기화 단계

로봇 위치의 확률적 표현은 베이지안 필터에 의해 표현될 수 있다. 이 방법은 확률 분포를 사용하여 로봇의 위치를 추정하는 방법으로, 확률 밀도를 이용하여 샘플을 추출하여 그 밀도가 높은 부분으로 로봇의 위치를 추정해간다 [2].

위치 추정의 초기화 단계를 실행하게 되면 각 샘플( $X$ )은 그 자세 정보( $x, \omega$ )와 확률 정보( $1/M$ )를 가지며, 이를 수식으로 표현하면 다음과 같다.

$$X = \langle x_0^{[1]}, \omega_0^{[1]}, x_0^{[2]}, \omega_0^{[2]}, \dots, x_0^{[m]}, \omega_0^{[m]} \rangle$$

$$= \langle x_0^{[1]}, 1/M, x_0^{[2]}, 1/M, \dots, x_0^{[m]}, 1/M \rangle$$

$M = 1, 2, \dots, m$ : 파티클의 갯수

같은 확률을 가지고 있는 각각의 샘플들은 로봇 주위 또는 전체에 산재한다.

2.1.2 위치 추정의 예측 단계

예측 단계에서는 각각의 파티클들은 입력변수에 의해 같은 확률을 가지고 움직인다고 가정하여 로봇의 모션 모델과 제어 입력으로 각 샘플들의 위치를 예측한다. 예측 단계는 Markov 가정에 의해서 새로운 위치  $X_t$ 는 이전의 위치  $X_{t-1}$ 과 제어 입력에만 의존한다.

아래 식 (2-3), (2-4)에서와 같이 이동 로봇의 왼쪽과 오른쪽 제어 입력( $\omega_{r,t}, \omega_{l,t}$ )에 노이즈( $\nu_r, \nu_l$ )를 추가하여 입력을 수정해 준다.

$$\dot{\omega}_{r,t} = \omega_{r,t} + \nu_r \quad (2-3)$$

$$\dot{\omega}_{l,t} = \omega_{l,t} + \nu_l \quad (2-4)$$

( $\nu_r, \nu_l$ ): 각 Odometry data에 대한 노이즈

아래 식 (2-5), (2-6), (2-7)은 실제 로봇의 궤적을 따르는 각 파티클의 모션 모델이며, 각각의 파티클마다 하나의 모션 모델을 가지고 있다.

$$x_t^{[m]} = x_{t-1}^{[m]} + T \frac{R_r \dot{\omega}_{r,t} + R_l \dot{\omega}_{l,t}}{2} \cos(\theta_{t-1}^{[m]}) \quad (2-5)$$

$$y_t^{[m]} = y_{t-1}^{[m]} + T \frac{R_r \dot{\omega}_{r,t} + R_l \dot{\omega}_{l,t}}{2} \sin(\theta_{t-1}^{[m]}) \quad (2-6)$$

$$\theta_t^{[m]} = \theta_{t-1}^{[m]} + T \frac{R_r \dot{\omega}_{r,t} - R_l \dot{\omega}_{l,t}}{2} \quad (2-7)$$

탐지 과정이 발생하지 않아 예측 과정을 반복하게 되면 입력에 의한 오차가 누적되어 파티클 군집의 분포가 확산되게 되어 로봇의 위치의 불확실성이 커지게 된다.

2.1.3 위치 추정의 갱신 단계

파티클 필터를 이용하여 로봇의 위치를 추정하기 위해서는 실제 로봇의 위치와 근접한 파티클의 군집을 형성할 필요가 있고, 그 군집을 통해 로봇의 위치를 추정하게 된다. 군집을 형성하기 위해서는 각 파티클이 실제 로봇과 얼마나 근접한 것인가를 판단해야 한다. 이를 위해 탐지를 통해 각 파티클의 탐지 정보와 실제 로봇의 탐지정보를 비교하여 가중치를 주어 그 가치를 판단하게 된다.

$$Z_t = (r, \phi) \quad (2-7)$$

여기서  $r$ 은 거리,  $\phi$ 는 로봇의 각도를 각각 나타내고 있다.

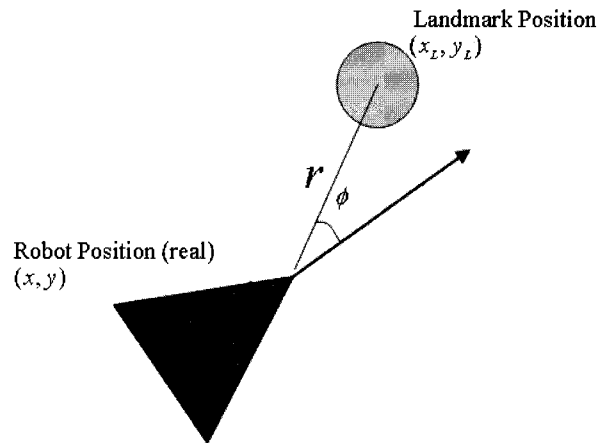


그림 2 탐지모델  
Fig. 2 The detection model

식 (2-7)은 실제 로봇의 탐지 정보로서 센서로 탐지한 거리 정보와 로봇 기준으로 탐지된 객체의 각도 정보이다. Fig. 2를 참조하여 보면 실제 로봇이 랜드마크를 탐지하였을 시, 로봇이 얻을 수 있는 정보는 거리정보 r과 로봇의 방향과 랜드마크 사이의 각도 정보 phi이다.

$$\hat{Z}_t = (\hat{r}, \hat{\phi}) \quad (2-8)$$

$$\omega_t^{[m]} = \eta p(\hat{Z}_t^{[m]} | X_t^{[m]})$$

$$= \frac{1}{\sqrt{2\pi(\sigma_r^2 + \sigma_\phi^2)}} e^{-\frac{(r - \hat{r})^2}{\sigma_r^2} - \frac{(\phi - \hat{\phi})^2}{\sigma_\phi^2}} \quad (2-9)$$

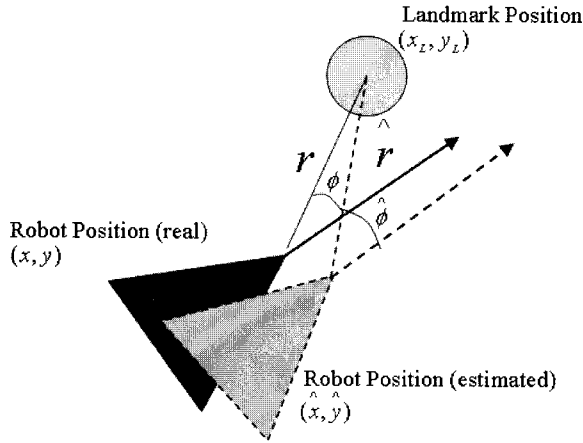


그림 3 실제 로봇과 파티클 사이의 탐지 오차

Fig. 3 The relationship between the real robot and the particle

식 (2-8)과 (2-9)는 각각의 파티클의 탐지 정보를 나타내고 있다. Fig. 3에서 보는 바와 같이 각각의 파티클들은 실제 로봇과 같이 탐지 모델을 가지고 있다. 실제 로봇이 랜드마크를 탐지하였을 경우, 얻어온 거리와 각도 정보를 가지고 각각의 파티클들이 이 정보를 공유하게 된다. 그러나 각각의 파티클들은 똑같은 모션 모델을 가지고 있지만 적용되는 오차가 조금씩 달라 누적된 오차로 인해 그 분산이 증가하게 되고 그 위치도 조금씩 달라진다. 로봇에서의 거리정보와 파티클 기준으로 탐지된 객체의 각도 정보가 있다. 여기서 거리오차  $\sigma_r$ 와 각도 오차  $\sigma_\phi$ 를 이용하여 각각의 파티클들에 수식 (2-9)에 따라 그 중요도를 부여할 수 있게 된다.

#### 2.1.4 위치 추정의 Resampling 단계

갱신 단계에서 가중치가 주어진 후, 각각의 파티클은 초기 확률이 아닌 가중치에 의해 다른 확률을 가지게 되었다. Resampling은 다른 확률을 가진 파티클들에 대해 로봇의 위치를 추정하기 위한 군집을 만드는 단계이다. 다른 확률을 가지고 산개해 있는 파티클들의 군집을 형성하기 위해서는 일정 임계치를 정하여 그 이하의 파티클과 이상의 파티클을 구분하여 임계치보다 낮은 확률을 가진 파티클은 실제 로봇의 위치와는 관련성이 적은 파티클이므로 로봇의 위치를 추정하는데 있어서 불필요하다. 따라서 그것들을 삭제하고, 삭제되는 파티클의 수만큼 임계치 이상의 파티클들이 존재하는 구역 내에 복제한다. 이 과정을 거치면 로봇의 위치를 결정하기 위한 군집이 형성되게 된다.

#### 2.1.5 위치 추정의 로봇 위치 및 자세 추정

$$K_t = P_{t|K-t} H^T S_t^{-1} \quad (2-10)$$

$$X_t = \frac{\sum_{m=1}^M \omega_t^{[m]} X_t^{[m]}}{\sum_{m=1}^M \omega_t^{[m]}} = \sum_{m=1}^M \omega_t^{[m]} X_t^{[m]} \quad (2-11)$$

Resampling 단계에서 만들어진 파티클의 군집을 이용하여 군집내의 각각의 파티클 위치 정보를 각의 축에 투영하여 그 히스토그램을 만들고 투영된 정보들의 평균을 통해 로봇의 위치를 결정한다. 식 (2-10)과 (2-11)은 그 과정을 수식으로 표현한 것이다. 이 단계가 끝나면 정규화에 의해 각 파티클들의 확률은 다시  $1/M$ 으로 변하게 된다. 그리고 예측의 단계로 되돌아가게 되고, 이 일련의 과정들을 계속 반복한다 [1,3,5,6].

## 2.2 확장 칼만 필터를 이용한 환경 내에 특징점 탐지

앞에서도 언급한 바와 같이 로봇의 위치 추정과 지도 작성의 관계는 서로 연관이 깊다. 로봇의 입장이 아닌 환경의 입장에서 고려해 보면, 로봇 자체로는 위치 추정의 정확성을 높일 수가 없다. 이에 계속해서 움직이는 로봇은 환경 내에 움직이지 않는 고정점 혹은 특징점을 통하여 그 위치 추정의 신뢰도를 높일 수 있다. 따라서 환경 내에 특징점을 추출하기 위해서는 환경에 대한 정확한 파악이 이루어져야 한다.

본 논문에서 로봇의 환경의 특징점인 랜드마크의 탐지와 그 위치의 갱신을 위해 확장 칼만 필터(EKF)를 사용하였다. EKF는 크게 예측과 보정으로 나눌 수 있으며, 상태 벡터와 오차 공분산은 예측 단계에서 예상한다. 그 후, 보정 부분에서 그것들이 교정되어 지며, 그 과정은 아래의 식과 같다.

#### 2.2.1 특징점 위치 탐지의 예측 단계

$$\hat{X}_{t|t-1} = F X_{t-1|t-1} + G U_t \quad (2-12)$$

$$P_{t|t-1} = F P_{t-1|t-1} F^T + G Q_{t-1} G^T \quad (2-13)$$

식 (2-12)와 (2-13)에 의해서 상태 벡터와 오차 공분산이 예측 된다. 상태 벡터  $X$ 는 그 요소로 로봇의 위치  $(x, y)$  및 방향 정보  $(\phi)$ 를 가지고 있는  $3 \times 1$ 의 형태이고, 랜드마크를 측정하였을 경우 그 위치가 상태벡터에 삽입되고  $5 \times 1$ 로 그 차원이 확장된다. 즉 랜드마크를 발견하기 이전에 상태 벡터는  $X_t = [x, y, \theta]^T$ 이었으나, 실제 로봇이 랜드마크를 발견하고 그 정보  $(r, \phi)$ 를 보내오면 그것을 이용하여 로봇은 가상의 랜드마크를 추정하면 그 위치  $(x_L, y_L)$ : 랜드마크의  $x, y$  좌표를 상태 벡터에 추가하여  $X_t = [x, y, \theta, x_L, y_L]^T$ 로 된다. 공분산  $P$ 는 랜드마크가 탐지되기 전까지는 로봇의 위치와 방향 오차만을 가진  $3 \times 3$ 의 형태이고, 랜드마크가 측정되면 상태 벡터와 마찬가지로 랜드마크의 위치 오차부분도 삽입되어 그 차원이  $5 \times 5$ 가 된다. 여기서  $F$ 는 상태전이 모델,  $G$ 는 입력 모델,  $U$ 는 제어입력 벡터,  $Q$ 는 시스템오차 모델이다.

2.2.2 특징점 위치 탐지의 보정 단계

$$\tilde{Y}_t = Z_t - H\hat{X}_{t|t-1} \quad (2-14)$$

$$S_t = HP_{t|t-1}H^T + R_t \quad (2-15)$$

$$K_t = P_{t|t-1}H^T S_t^{-1} \quad (2-16)$$

$$\hat{X}_{t|t} = \hat{X}_{t|t-1} + K_t \tilde{Y}_t \quad (2-17)$$

$$P_{t|t} = (I - K_t H)P_{t|t-1} \quad (2-18)$$

여기서  $Y$ 는 이노베이션 오차, 즉 로봇이 랜드마크를 재 발견하였을 때 이전에 탐지한 정보와 비교하여 오차를 구하는 식이며  $Z$ 는 측정벡터,  $H$ 는 측정 모델,  $S$ 는 로봇의 위치와 방향의 오차 공분산  $P$ 와 같은 형태인 이노베이션 공분산,  $K$ 는 칼만이득이다. 탐지 정보에 의해서 식 (2-14)와 (2-15)를 결정할 수 있다. 식 (2-16)에서 최적의 칼만 이득 ( $K$ )을 구하여 식 (2-17)과 (2-18)에서 예측된 상태 벡터 ( $\hat{X}_{t|t-1}$ )와 오차 공분산( $P$ )을 보정한다. 즉, 이전 탐지의 정보와 재발견한 특징점의 정보의 오차( $S$ )를 이용하여 그 이득( $K$ )을 구한 후 재발견된 정보에 이득을 합산하여 그 오차를 보정한다. 보정된 상태 벡터( $\hat{X}_{t|t}$ )와 오차 공분산( $P$ )을 다시 예측하는 파라미터로 들어가게 되어 순환적으로 그 오차를 재보정하게 된다.

2.3 위치 추정과 특징점 탐지의 조합

2.1절과 2.2절에서 설명한 두 가지 알고리즘의 조합으로 자율 이동 로봇에 대한 SLAM이 수행될 수 있다. 위치 추정 단계에서 각 파티클은 실제 로봇과 같은 모션모델을 가지고 랜드마크에 대한 탐지가 발생하기 전까지 예측 단계만을 수행하고 각각의 파티클에 주어진 오차에 의해 초기 위치에서 진행이 거듭될수록 오차가 누적되어 그 분산이 증가하게 된다. 다시 말해 랜드마크에 대한 탐지가 일어나기 전까지는 확장 칼만 필터는 사용되지 않는다. 그러나 랜드마크에 대한 탐지가 발생하였을 경우, 실제 로봇에서 전송되는 데이터( $Z=(r,\phi)$ )로 최초 발견된 랜드마크인지 재발견된 랜드마크인지에 대한 여부를 판단하고 최초 발견된 랜드마크일 경우 로봇의 상태 벡터에 추가만 하고 다시 예측 단계만을 진행한다. 만약 재발견된 랜드마크이면 로봇은 확장 칼만 필터 알고리즘에 의해 랜드마크의 위치에 대한 보정이 발생하게 되고 로봇의 위치추정에 사용된 파티클 필터 알고리즘에 의해 각 파티클에 대해 가중치를 부여할 수 있다. 이 가중치를 근거로 하여 일정 임계치 이하의 파티클은 삭제되고 삭제되는 수만큼 임계치 이상의 파티클을 복제하게 된다. 위와 같은 과정을 거듭하게 되면 랜드마크가 발견되기 전, 오차에 의해 분산되었던 파티클들은 점점 한 점으로 수렴하게 된다. 이 두 알고리즘의 조합으로 로봇의 위치와 랜드마크의 위치를 동시적으로 추정, 보정할 수 있다. 두 알고리즘의 조합된 흐름도를 Fig. 4에 나타내었다.

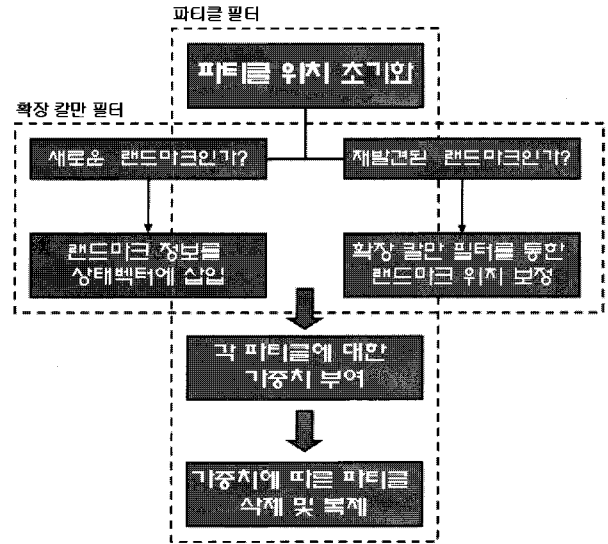


그림 4 두 알고리즘의 조합 흐름도  
Fig. 4 The flow chart of the algorithm

3. 강인한 SLAM의 구현

3.1 기구학적 문제

내부적 오차와 로봇의 위치 추정 및 자세 추정은 상호 의존적이며, 로봇 내부적인 오차는 로봇의 형태, 구동 방법에 따라 차이가 있다. 두 개의 바퀴를 이용한 차동 형태의 로봇의 경우, 가장 큰 내부적인 오차는 두 바퀴의 지름의 차이 때문에 두 바퀴에 대한 제어 입력이 같아도 로봇은 두 바퀴의 각속도의 차이와 바닥의 미끄러움까지 더해져 입력과는 다르게 이동하게 된다 [7-9].

본 논문에서 다루는 무한궤도 로봇의 무한궤도는 지면 상태의 불안정에도 강인하게 동작할 수 있으나 그 기구학적 모델링이 어려우며, 실제 구동 중 지면과의 슬립이 일반적인 차동형 바퀴 타입의 로봇보다 많이 발생한다. 또한 회전 시에 무게 중심이 주변 환경의 상황에 따라 비선형적으로 바뀌기 때문에 아직까지도 무한궤도형 이동 물체의 운동방정식은 많이 연구되고 있다.

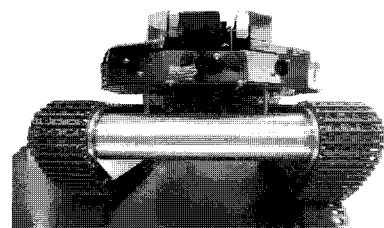


그림 5 무한궤도형 로봇  
Fig. 5 The caterpillar type mobile robot

Fig. 5는 본 논문에서 제안하는 알고리즘을 검증하기 위하여 제작된 무한궤도형 로봇이다. 본 논문에서는 이동로봇의 복잡한 운동방정식을 일반 바퀴형 차동 방정식의 모델로 쓰되, 미끄러움이나 회전오차를 보정하였다.

### 3.2 모션 모델의 비교

일반적인 두 개의 바퀴를 가진 차동형 이동 로봇의 경우 그 모션 모델은 바퀴의 각속도를 바퀴의 반지름과 곱하여 그것을 선속도화하여 현재 위치로부터 제어 입력(각속도)에 의해 다음 위치를 예측한다. 차동형 이동 로봇은 제자리 회전이 가능하며 그 오차도 무한궤도 형태의 로봇보다 작다. 일반적인 차동형 이동 로봇의 모션 모델은 아래와 같다.

$$x_{t+1} = x_t + T \frac{R_r \omega_{r,t} + R_l \omega_{l,t}}{2} \cos \theta_t \quad (3-1)$$

$$y_{t+1} = y_t + T \frac{R_r \omega_{r,t} + R_l \omega_{l,t}}{2} \sin \theta_t \quad (3-2)$$

$$\theta_{t+1} = \theta_t + T \frac{R_r \omega_{r,t} - R_l \omega_{l,t}}{l} \quad (3-3)$$

여기서  $R_r, R_l$ 는 바퀴의 좌우 반지름이고  $w_{r,t}, w_{l,t}$ 는 입력 변수인 바퀴의 좌우 각속도이다. T는 제어 시간 주기이고, l은 두 바퀴 사이의 반지름이다. 식 (3-1), (3-2), (3-3)에서 보는 바와 같이 로봇의 방향이 양쪽 바퀴의 속도 차에 의해서 결정이 되며 바퀴 간의 간격도 이에 영향을 미친다. 본 논문의 실험에서 사용된 무한궤도형 로봇의 모션 모델도 식 (3-1), (3-2), (3-3)과 크게 다르지 않다. 그러나 두 개의 바퀴를 이용한 차동형태의 로봇은 요철이 심한 지면을 주행할 때나 장애물을 넘을 때 무한궤도형 로봇에 비해 강인하게 동작하지 못한다. 따라서 본 논문에서는 외부 주행 환경에서도 강인하게 주행할 수 있는 무한궤도형의 로봇을 제안한다. 그러나 무한궤도형의 로봇은 무한궤도와 지면의 마찰면적이 적어지게 되는 때는 제어 입력을 가한 만큼 로봇이 이동, 회전 하는 것이 아니기 때문에 바퀴를 이용한 차동형의 로봇과 같이 각속도를 제어입력으로 가할 수 없다. 이를 보완하기 위하여 본 논문에서는 식(3-1), (3-2), (3-3)에서 보는 바와 같이 각속도 대신 선속도를 제어 입력으로 사용하였다.

$$x_{t+1} = x_t + T \frac{V_{r,t} + V_{l,t}}{2} \cos \theta_t \quad (3-4)$$

$$y_{t+1} = y_t + T \frac{V_{r,t} + V_{l,t}}{2} \sin \theta_t \quad (3-5)$$

$$\theta_{t+1} = \theta_t + T \frac{V_{r,t} - V_{l,t}}{l} \quad (3-6)$$

위 식의 모션 모델에서 로봇은 양쪽 무한궤도의 선속도 차이에 의해서 방향이 결정되고 위치 추정 알고리즘에 의해 그 위치와 방향, 슬립 등을 보정한다.

### 3.3 보정

무한궤도형 이동 로봇의 SLAM에서 기구학적 불확실성에 의해 로봇의 odometry 정보만으로는 로봇의 정확한 방향을 결정지을 수 없기 때문에 본 논문에서는 이를 보정하여 무한궤도형 로봇 운동성의 정확도를 높였다. 그 방법으로 다음

과 같이 수행하였다.

① 위치 추정 알고리즘인 파티클 필터의 수행 단계 중 그 첫 번째인 Initialize 단계에서 가상의 파티클들을 산개시킬 때, 상태 벡터의 파티클들의 방향 성분을 실제 로봇의 자이로로 측정한 방향 값의 일정 오차 내에 주어 각 샘플의 방향을 보정한다.

② 3.2절에서 수정한 모션 모델에 의해 파티클들이 자이로 센서의 측정된 방향에 한정되어 움직인다.

③ 랜드 마크에 대한 탐지가 발생하였다면, 가중치 부여에 의해 각 파티클의 확률이 달라지게 된다. 이는 resampling 단계에서 로봇의 위치 추정을 위한 군집을 만드는 과정의 기반이 된다.

④ 3단계에서 가중치가 주어진 파티클들에 대해 일정 임계치 이하의 것들은 삭제된다. 그리고, 그 외의 임계치 이상의 확률을 가진 파티클들에 일정 구역 내에 삭제된 것들 만큼의 개수가 복제된다. 이때, 다시 생성되는 파티클들의 상태 벡터의 각도 성분이 자이로 센서의 측정된 방향에 한정되어 생성된다. 이 과정에 의해 파티클들의 군집이 형성되고 상태 벡터의 각도 성분의 축으로 투영하여 그 값의 평균을 구하여 로봇의 위치를 추정하게 된다.

## 4. 실험 결과

아래 Fig. 6은 시뮬레이션 개발 환경을 나타내고 있다.

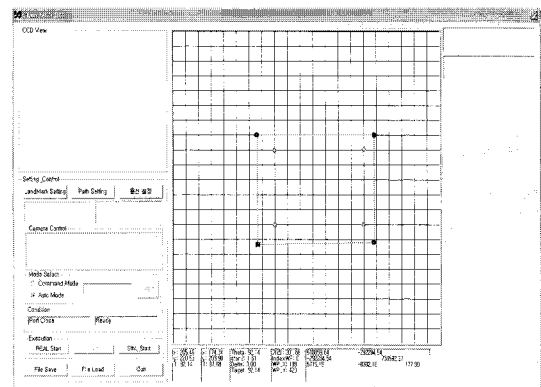


그림 6 시뮬레이션 환경  
Fig. 6 The simulation environment

로봇의 타당성을 검증하기 위해 실제와 동일한 환경을 가정하고 시뮬레이션을 실행하였다. 실험 환경에서는 기타 장애물은 없으며 랜드마크만 존재한다고 가정하였다. 로봇의 크기를 32cm, 폭을 22cm, 가상의 실험 영역을 가로 400cm, 세로 350cm 인 직사각형 공간으로 설정하였다. 이동로봇의 속도는 0.1m/sec, 제어주기를 0.4m/sec로 하였다. 랜드마크에 대한 탐지는 센서에 대한 제한 없이 전방향 180도 내, 감지 거리는 50cm, 오차 거리는 ±5cm, 각도 오차는 ±10°로 설정하였다. 무한 궤도형 이동로봇의 부정확한 이동 특성을 감안하여 좌우측의 입력 오차를 입력 선속도의 절반 이상인 0.06m/sec로 설정하였다. 시뮬레이션 실행 환경은 Visual C++로 구성하였다.

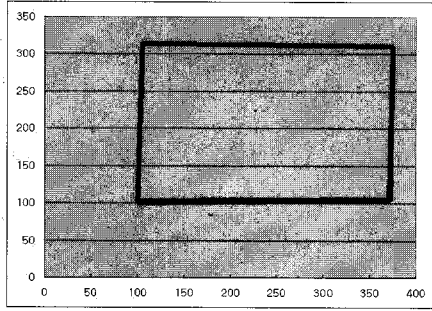


그림 7 로봇의 기준 경로

Fig. 7 The reference path of the mobile robot

Fig. 7은 로봇의 기준 경로를 나타내고 있다. 위의 그림은 x축, y축 각각 최대 400cm인 2차원 환경을 나타내고 있다. 로봇은 주어진 모션 모델을 이용하여 Fig. 7의 주행 경로를 따르게 된다.

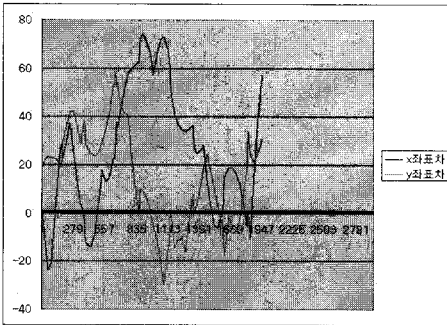
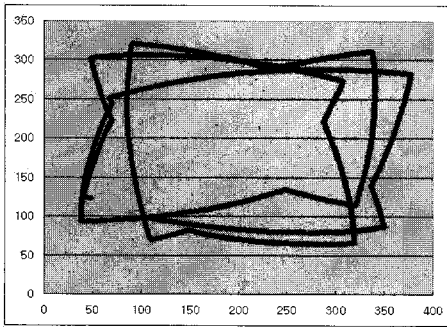


그림 8 Odometry 정보만을 이용하여 주행하였을 때의 x-y 오차

Fig. 8 x-y Errors with only odometry information

Fig. 8의 위의 그림은 경로의 보정이 없이 odometry 정보만을 이용하여 dead reckoning 방법만으로 주행하였을 경우 그 궤적을 나타낸 것이다. 보정 단계가 존재하지 않아 그 궤적이 참조궤적과 차이가 많이 난다는 것을 볼 수 있다. 아래 그림은 그 오차를 그래프로 나타낸 것이다.

Fig. 9은 방향 보정 없이 일반적인 랜드마크 3개를 사용하여 SLAM을 실행한 결과이다. 파티클 산개 시 실제 로봇에 장착된 자이로를 이용하여 절대 방향을 측정 후, 통일된 방향 정보를 각각의 파티클에 주었다. 보정 시점에서 좌표가 보정되는 것을 확인할 수 있지만, 그 외의 구간에서는 많은 오차 값으로 인해 다시 불안정한 주행을 하게 된다. 하지만

Fig. 8. 의 오차 그래프와 비교하여 시간당 오차면적이 상당히 줄어든 것을 확인할 수 있다. 여기서 로봇이 각각의 랜드마크를 측정하였을 경우 각각의 거리와 각도 정보뿐만 아니라 추가적으로 랜드마크의 인식 번호까지 파악할 수 있게 하였다. 또한 고정점인 랜드마크를 측정하고 그 위치를 확장 칼만 필터를 통해 보정함으로써 로봇의 위치를 보정할 수 있게 되었다.

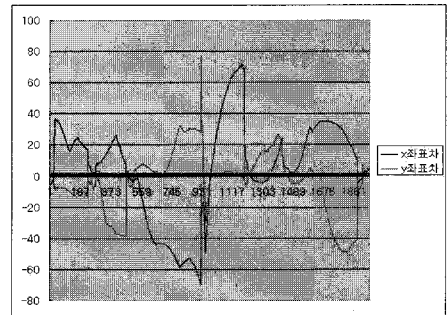
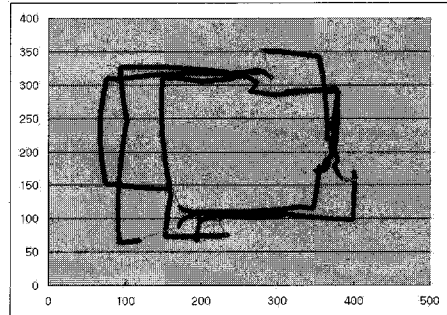


그림 9 파티클 산개 시 방향 보정 없이 주행하였을 경우 x-y 오차

Fig. 9 x-y Errors without directional compensation

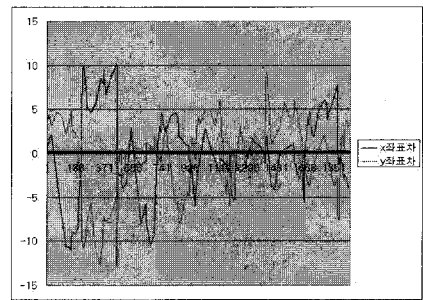
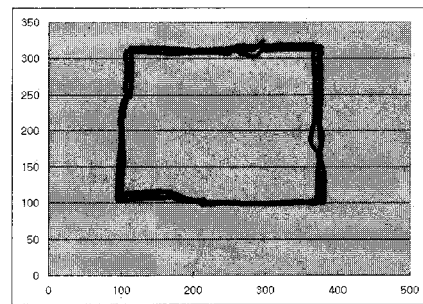


그림 10 파티클 산개 시 방향 보정 후 주행하였을 경우 x-y 오차

Fig. 10 x-y Errors after directional compensation

Fig. 10는 각도 보정을 resampling 시간마다 수행된 랜드마크 3개를 사용한 SLAM의 결과를 보여 주고 있다. 위의 그림에서 랜드마크 구간 후에도 기준 경로를 많이 벗어나지 않는 것을 알 수 있다. 기준 경로의 x-y좌표 차이도 거의  $\pm 10$ 을 유지하고 있으며, 결과적으로 Fig. 8 및 Fig. 9와 비교하여 오차 면적이 8배 정도 줄어든 것을 시뮬레이션 결과의 오차 그래프를 통해서 확인할 수 있다.

### 5. 결 론

본 논문은 산업화 및 가정용 서비스 로봇에 이르기까지 그 활용 분야가 확대되고 있는 이동 로봇의 가장 필수 기술인 SLAM에 관한 것으로, 무한 궤도형 로봇의 정확한 위치 추정을 위한 연구이다. 무한 궤도형 이동 로봇은 운동성이 탁월하여 이동할 수 있는 범위가 넓으나, 주변 환경에 따른 운동학적 계산 수식이 복잡하여 그 위치를 추정하기가 어렵다. 본 논문에서는 무한궤도형 로봇의 복잡하고 부정확한 운동 방정식 대신에 간단한 차동형 로봇의 운동방정식을 사용하여 로봇 위치와 환경 정보 값을 보다 정확하게 추정할 수 있는 방법을 제안하였다. 위치 추정과 환경 지도 작성을 각각 파티클 필터와 확장 칼만 필터를 사용하여 구현을 하되, 위치 추정 과정에서의 파티클 생성 과정에서 자이로(각도) 센서 값을 방향 성분 에 대한 보정 값으로 추가 하여 빠른 시간 내에 위치 정보 개선하였다. 이러한 연구를 보다 실제에 가깝게 구성하고 실제 실험에서 검증 을 거쳐 적용한다면 향후 무한궤도형 이동로봇에 대한 신뢰성 높은 제어를 할 수 있을 것이다.

### 참 고 문 헌

[1] Michael Montemerlo, "FastSLAM : A Factored Solution to the Simultaneous Localization and Mapping Problem with Unknown Data Association", School of Computer Science Carnegie Mellon University, 2003.

[2] R. Chellappa, A. Jain, "Markov Random Fields : Theory and Applications", Academic Press, 1993.

[3] A. Doucet, N. de Freitas, K. Murphy, S. Russell, "Rao-blackwellised particle filtering for dynamic bayes net", Proceedings of conference on Uncertainty in Artificial Intelligence, 2000.

[4] A. Doucet, N. de Freitas, and N. Gordon, Sequential Monte Carlo Methods in Practice. Springer-Verlag, 2001.

[5] F. Dellaert, D. Fox, W. Burgard, S. Thrun, "Monte carlo localization for mobile robots", Proceedings of the IEEE International Conference on Robotics and Automation, Vol. 2, pp. 1322-1328, 1999.

[6] S. Thrun, W. Burgard, D. Fox, "Probabilistic Robotics", MIT press, 2005.

[7] M. Deans, M. Herbert, "Experimental comparison of techniques for localization and mapping using a bearing-only sensor", Proceedings of the International

Symposium on Experimental Robotics (ISER), 2002.

[8] E. Nebot, F. Masson, J. Guivant, and H. Durrant-Whyte, "Robust simultaneous localization and mapping for very large outdoor environments", Springer Tracts in Advanced Robotics, Vol. 5, pp. 209-209, 2003.

[9] J. Tardos, J. Niera, P. Newman, and J. Leonard, "Robust mapping and localization in indoor environments using sonar data", International Journal of Robotics Research, Vol. 21, No. 4, pp. 311-330, 2002.

## 저 자 소 개



### 변 성 재 (卞 盛 載)

2006년 영남대학교 전기공학과(공학사).  
2008년 영남대학교 대학원 전기공학과(공학석사). 관심 분야 : 자율 이동 로봇의 위치 인식과 지도 작성(SLAM)  
Tel : 053-810-3923  
Fax : 053-810-4767  
E-mail : plugin2@gmail.com



### 이 석 규 (李 錫 圭)

1979년 서울대학교 전기공학과 (공학사).  
1981년 서울대학교 전기공학과 (공학석사). 1989년 Univ. of California, Los Angeles (공학박사). 1982년~현재 영남대학교 전기공학과, 관심 분야 : 홈네트워킹, 모바일 로봇, 임베디드 시스템.  
Tel : 053-810-2487  
Fax : 053-810-4767  
E-mail : sglee@ynu.ac.kr



### 박 주 현 (朴 柱 炫)

1990년 경북대학교 전자공학과 (공학사).  
1997년 POSTECH 전자전기공학과 (공학박사). 2000년~현재 영남대학교 전기공학과, 관심 분야 : 비선형제어이론 및 응용.  
Tel : 053-810-2491  
Fax : 053-810-4767  
E-mail : jessie@ynu.ac.kr