

자이로 센서를 이용한 이동로봇 Odometry 오차 보정에 관한 연구

論 文
55D-2-4

Odometry Error Correction with a Gyro Sensor for the Mobile Robot Localization

朴 恃 擎[†] · 洪 顯 珠^{*} · 崔 愿 太^{**}
(Shi-Na Park · Won-Tae Choi · Hyun-Ju Hong)

Abstract - To make the autonomous mobile robot move in the unknown space, we have to know the information of current location of the robot. So far, the location information that was obtained using Encoder always includes Dead Reckoning Error, which is accumulated continuously and gets bigger as the distance of movement increases. In this paper, we analyse the effect of the size of the two wheels of the mobile robot and the wheel track of them among the factors of Dead Reckoning Error. And after this, we compensate this Dead Reckoning Error by Kalman filter using Gyro Sensors. To accomplish this, we develop the controller to analyse the error components of Gyro Sensor and to minimize the error values. We employ the numerical approach to analyse the error components by linearizing them because each error component is nonlinear. And we compare the improved result through simulation.

Key Words : Indirect Kalman Filter, Gyro Sensor, Odometry, Mobile Robot

1. 서 론

그동안 이동로봇의 위치 인식을 위해 가장 널리 사용된 방법은 오도메트리(Odometry)이다. 하지만, 오도메트리는 오차를 축적한다는 단점을 가지고 있다. 따라서, 주행거리가 멀어 진다면 축적된 오도메트리에 의한 위치 정보는 신뢰할 수 없게 된다. 이에 오도메트리 오차를 감소하기 위해 Borenstein[1]은 오도메트리 오차를 시스템 오차와 비시스템 오차로 분석하여 시스템 오차를 보정하는 방법을 제안하였으며, Barashan과 Durrant[2]는 자이로 센서를 이용하여 오차를 보정하였다. 그 동안 자이로 센서를 이용한 오차 보정 방법은 주로 Linearized Kalman Filter(LKF)와 Extended Kalman Filter(EKF)에 의해 이루어졌다[3][4][5]. 이 외에도 Landmark를 이용하여 로봇의 위치를 보정하거나, active beacon을 이용한 방법이 이용되어 왔었다. 그러나, Landmark 나 active beacon을 이용한 방법은 외부 환경에 의해 절대적인 영향을 받으므로 로봇이 스스로 위치 인식을 하지 못하는 경우가 발생할 수도 있다.

본 논문에서는 오도메트리에 의한 오차를 보정하기 위해 오도메트리 오차를 분석하고, 자이로를 이용하여 IKF에 의해 오차 성분을 최소화 하였다[6]. 또한, 제안된 알고리즘의 유효성을 확인하기 위해 실험을 통하여 증명하였다.

2. 오차 모델

2.1 Wheel 엔코더 오차 모델

엔코더에 의한 이동로봇의 Kinematic은 그림 1과 같이 각 바퀴의 이동량(dl , dr)과 바퀴사이의 거리(D)에 의해 결정된다. 그리고, 실제 이동 로봇의 위치는 하드웨어적인 오차 u_r , u_l 와 모델링 오차 ϵ_r , ϵ_l 로 구분할 수 있다. 이를 고려한 것이 식(1)~(4)이다.

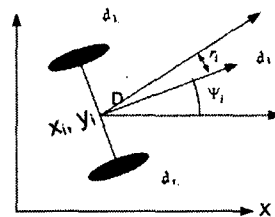


그림 1. 이동로봇의 기구학

Fig 1 Mobile Robot Kinematics

$$\begin{aligned} d_{r_i}^o &= (1 + \mu_{r_i})d_{r_i} + \epsilon_{r_i}, \\ d_{l_i}^o &= (1 + \mu_{l_i})d_{l_i} + \epsilon_{l_i}, \end{aligned} \quad (1)$$

$$\begin{aligned} d_i^o(\mu_r, \mu_l, \epsilon_r, \epsilon_l) &= \frac{d_{r_i}^o + d_{l_i}^o}{2} \\ &= \frac{(1 + \mu_r)}{2}d_r + \frac{(1 + \mu_l)}{2}d_l + \frac{\epsilon_r}{2} + \frac{\epsilon_l}{2} \end{aligned} \quad (2)$$

$$\begin{aligned} r_i^o(\mu_r, \mu_l, \epsilon_r, \epsilon_l, \Delta D_i) &= \frac{d_{r_i}^o - d_{l_i}^o}{D^o} \\ &= \frac{(1 + \mu_r)d_r - (1 + \mu_l)d_l + \epsilon_r + \epsilon_l}{D + \Delta D_i} \end{aligned} \quad (3)$$

† 교신저자, 正會員 : 울산대학교 전기전자정보시스템 공학부
E-mail : licjnp@nate.com

* 正會員 : 울산대학교 전기전자정보시스템공학부

** 正會員 : 케피코(주)

接受日字 : 2005年 12月 12日

最終完了 : 2005年 12月 28日

$$\vec{p}_{i+1}^o = \vec{p}_i^o + \begin{pmatrix} d_i^o(\mu_{r_i}, \mu_{l_i}, \varepsilon_{r_i}, \varepsilon_{l_i}) \cos(\psi_i^o) \\ d_i^o(\mu_{r_i}, \mu_{l_i}, \varepsilon_{r_i}, \varepsilon_{l_i}) \sin(\psi_i^o) \\ r_i^o(\mu_{r_i}, \mu_{l_i}, \varepsilon_{r_i}, \varepsilon_{l_i}, \Delta D_i) \end{pmatrix} \quad (4)$$

여기서, r_i 이동로봇의 회전 변화량이다. 이동로봇의 진행 방향에 대한 표현은 Ψ 로 표현한다. 따라서, 엔코더에 의한 이동로봇의 좌표는 식(4)에 의해 표현된다.

2.2 자이로 모델

자이로의 출력 신호는 식(5)로 표현된다[7].

$$r_i^g = (1 + \beta_i)r_i + \eta_i + \eta_{static} + \varepsilon_{g_i} \\ = (1 + \beta_i) \frac{d_{r_i} - d_{l_i}}{D} + \eta_i + \eta_{static} + \varepsilon_{g_i} \quad (5)$$

여기서, β_i 는 scale factor error이며, η_i 는 bias, η_{static} 는 static drift, ε_{g_i} 는 noise이다.

η_{static} 와 η_i 는 자이로 특성 실험을 통해서 제거 될 수 있다. 따라서, 자이로에 의한 각도는 식 (6)과 같이 표현된다.

$$\psi_{i+1}^g = \psi_i^g + r_i^g(\beta_i, \eta_i, \varepsilon_{g_i}) \quad (6)$$

3. 위치 측정 모델

3.1 프로세스 모델

위에서 언급한 오차 모델을 이용한 state vector는 식 (7) 과 같이 표현 할 수 있으며, 센서 noise vector는 (8), Error noise vector는 (9)로 표현할 수 있다.

$$\vec{x}_i = (\vec{p}_i^o, \mu_{r_i}, \mu_{l_i}, \Delta D_i, \psi_i^g, \beta_i, \eta_i)^T \quad (7)$$

$$\vec{\varepsilon}_i = (\varepsilon_{r_i}, \varepsilon_{l_i}, \varepsilon_{g_i})^T \quad (8)$$

$$\vec{\xi}_i = (0, 0, 0, \varepsilon_{\mu_r}, \varepsilon_{\mu_l}, \varepsilon_{\Delta D_i}, 0, \varepsilon_{\beta_i}, \varepsilon_{\eta_i})^T \quad (9)$$

따라서, 위의 식을 이용한 시스템 방정식은 식 (10)과 표현 할 수 있다. 식 (10)은 비선형 방정식이다. 이를 선형화하기 위해서는 자코비안 행렬을 이용한다.

$$\begin{pmatrix} \vec{p}_{i+1}^o \\ \mu_{r_{i+1}} \\ \mu_{l_{i+1}} \\ \Delta D_{i+1} \\ \psi_{i+1}^g \\ \beta_{i+1} \\ \eta_{i+1} \end{pmatrix} = \begin{pmatrix} \vec{p}_i^o \\ \mu_{r_i} \\ \mu_{l_i} \\ \Delta D_i \\ \psi_i^g \\ \beta_i \\ \eta_i \end{pmatrix} + \begin{pmatrix} d_i^o(\mu_{r_i}, \mu_{l_i}, \varepsilon_{r_i}, \varepsilon_{l_i}) \cos(\psi_i^o) \\ d_i^o(\mu_{r_i}, \mu_{l_i}, \varepsilon_{r_i}, \varepsilon_{l_i}) \sin(\psi_i^o) \\ r_i^o(\mu_{r_i}, \mu_{l_i}, \varepsilon_{r_i}, \varepsilon_{l_i}, \Delta D_i) \\ 0 \\ 0 \\ 0 \\ r_i^g(\beta_i, \eta_i, \varepsilon_{g_i}) \\ 0 \\ 0 \end{pmatrix} + \vec{\xi}_i \\ = \vec{f}_i(\vec{x}_i, \vec{\varepsilon}_i, \vec{\xi}_i) \quad (10)$$

Perturbative vector는 식 (11)로 표현되며, Error state vector는 식 (12)로 표현된다.

$$\vec{\delta x}_{i+1} = A_i \vec{\delta x}_i + \Gamma_i \vec{\varepsilon}_i + \vec{\xi}_i \quad (11)$$

$$\vec{\delta x}_i = \vec{x}_i - \vec{x}_i^{ideal} \\ = (\delta p_i^o, \mu_{r_i}, \mu_{l_i}, \Delta D_i, \delta \psi_i^g, \beta_i, \eta_i)^T \quad (12)$$

$$A_i = \left. \frac{\partial \vec{f}_i}{\partial \vec{x}} \right|_{\vec{x}_i = \vec{x}_i^{ideal}} \quad \text{이며,} \quad \Gamma_i = \left. \frac{\partial \vec{f}_i}{\partial \vec{\varepsilon}_i} \right|_{\vec{x}_i = \vec{x}_i^{ideal}} \quad \text{이다.}$$

3.2 관측 모델

관측 모델은 엔코더에 의해 구해진 각도와 자이로에 의해 구해진 각도 차이를 나타낸다. 따라서, 식 (13)과 같이 표현 할 수 있다.

$$z_i^{og} = r_i^o(\vec{x}_i) - r_i^g(\vec{x}_i) = g_i(\vec{x}_i) \quad (13)$$

위 식을 선형화하면 식 (14)와 같이 표현된다.

$$\vec{\alpha}_i^{og} = \left. \frac{\partial g_i}{\partial x_i} \right|_{\vec{x}_i = \vec{x}_i^{ideal}} \vec{\delta x}_i + \left. \frac{\partial g_i}{\partial \varepsilon_i} \right|_{\vec{x}_i = \vec{x}_i^{ideal}} \vec{\varepsilon}_i + \kappa_i \\ = \vec{\alpha}_i^o - \vec{\alpha}_i^g \\ = (0 \ 0 \ 0 \ \frac{d_{r_i}}{D} \ -\frac{d_{l_i}}{D} \ -\frac{d_{r_i} - d_{l_i}}{D^2} \ 0 \ -\frac{d_{r_i} - d_{l_i}}{D} \ -1) \vec{\delta x}_i \\ + (\frac{1}{D} \ -\frac{1}{D} \ -1) \vec{\varepsilon}_i + \kappa_i \\ = H_i \vec{\delta x}_i + \gamma_i \vec{\varepsilon}_i + \kappa_i \quad (14)$$

4. Indirect Kalman Filter

IKF에 의한 시스템 모델은 식(15)와 같다.

$$\vec{\delta x}_{i+1} = A_i \vec{\delta x}_i + \Gamma_i \vec{\varepsilon}_i + \vec{\xi}_i \\ \vec{\delta z}_i = H_i \vec{\delta x}_i + Y_i \vec{\varepsilon}_i + k_i \quad (15)$$

여기서, $\vec{\xi}_i$ 와 k_i 는 noise 성분으로 아래와 같이 분석할 수 있다.

$\vec{\varepsilon}_i \sim N(0, Q)$: 센서 noise로 white noise이다.

$\vec{\xi}_i \sim N(0, G)$: process 모델의 noise이다.

$k_i \sim N(0, R)$: 측정 모델의 noise이다.

식(15)의 연산결과와 참값의 오차를 최소화하기 위해 IKF 식을 사용하였다.

5. 실험

5.1 실험환경

이동로봇은 TRC의 LABMATE이며, 자이로 센서는 silicon sensing systems사의 CRS-03을 이용하였다. H/W 구성은 그림 2와 같이 센서 정보를 Mobile robot 제어기에서 취득하여 직렬통신을 이용하여 이동 로봇의 위치계산용 PC에 보낸다. 각 센서의 Sampling time은 표1과 같이 설정하였다.

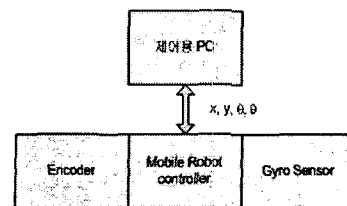


그림 2. Mobile Robot H/W 구조
Fig. 2 Architecture of Mobile Robot HW

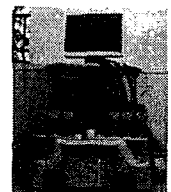


그림 3. Labmate
Fig. 3 Labmate

표 1. Sampling time

Table 1 Sampling time

	Sampling time
엔코더	100ms
자이로	10ms

5.2 실험

실험은 2m*2m의 사각형 형태의 경로를 각 10회 주행 후 오도메트리에 의해 실험한 결과와 자이로를 이용한 IKF에 의한 값을 비교하였다. 실험 결과 위치 오차와 각도 오차가 줄어 든 것을 표4를 통해 알 수 있다. 실험을 위한 초기 값은 센서의 특성실험과 시행착오에 의해 다음과 같이 결정하였다.

$$\text{diag} \{Q_i\} = \left\{ 1, 1, 0.02 \times \frac{\pi}{180^2} \right\}$$

$$\text{diag} \{G_i\} = \left\{ 0, 0, (0.1 \times \frac{\pi}{180})^2, 0.00001^2, 0.00001^2, 0.001^2, (0.1 \times \frac{\pi}{180})^2, 0.0001^2, (0.001 \times \frac{\pi}{180})^2 \right\}$$

$$R_i = (0.001 \times \frac{\pi}{180})^2$$

표 2. 위치 및 각도의 오차 평균

Table 2 Average of position and angle errors

	Odometry	IKF
x [mm]	108.62	52.39
y [mm]	88.08	23.06
θ [deg]	6.70	3.63

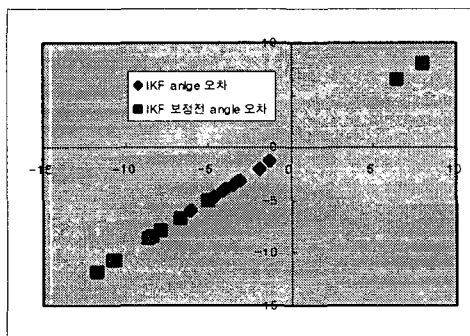
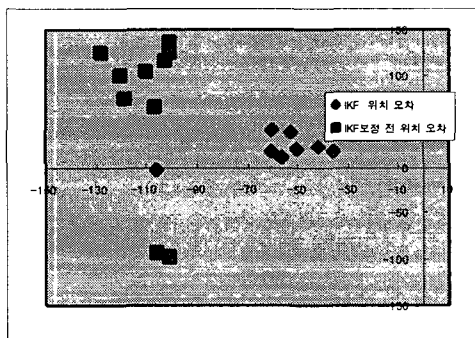


그림 4. 주행 실험 후의 로봇의 위치

Fig. 4 Robot position after navigation experiment

6. 결 론

IKF에 의한 이동로봇의 위치인식 오차 보정을 위한 방법은 엔코더와 자이로의 측정에 의한다. 따라서, 두 센서의 오차를 선형화 한 오차 모델을 이용하여, IKF에 적용하였다. 실험 결과 엔코더에 의한 Odometry 오차를 보정함을 알 수 있다. 제안된 알고리즘을 이용하여 이동로봇의 자율 주행을 수행한다면 좀 더 정확한 위치 인식을 할 수 있다는 것을 알 수 있다.

자이로 센서도 오차가 발생하게 된다면 누적되므로 향후에는 GPS와 같이 절대 위치 정보를 얻을 수 있는 방법을 추가해 준다면 주행거리가 길어지더라도 계속 정확한 위치 정보를 획득 할 수 있다.

제안된 알고리즘을 활용하여 Map building을 한다면 SLAM(Simultaneous Localization And Map building)에 적용가능하다. 따라서, 향후 과제로 SLAM에 활용할 수 있는 방안을 연구하여 좀 더 정확한 위치 인식 및 지도 작성에 이용할 것이다.

참 고 문 헌

- [1] Borenstein, J., and Feng, L. 1996a. Measurement and correction of systematic odometry errors in mobile robots. IEEE Transactions on Robotics and Automation 12:869 - 880.
- [2] Barshan, B., and Durrant-Whyte, H. F. 1995. Evaluation of a solid-state gyroscope for robotics applications. IEEE Transactions on Instrumentation and Measurement 44:61 - 67.
- [3] Borenstein, J. 1998. Experimental evaluation of a fiber optics gyroscope for improving dead-reckoning accuracy in mobile robots. In Proc. IEEE Int. Conf. on Robotics and Automation, Vol. 4, pp. 3456 - 3461.
- [4] Borenstein, J., and Feng, L. 1996b. Gyrodometry: A new method for combining data from gyros and odometry in mobile robots. In Proc. IEEE Int. Conf. on Robotics and Automation, Vol. 1, pp. 423 - 428.
- [5] Park, K., Chung, D., Chung, H., and Lee, J. 1996. Dead reckoning navigation of a mobile robot using an indirect Kalman filter. In Proc. IEEE/SICE/RSJ Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems, pp. 132 - 138.
- [6] Alexander Rudolph, 2003. Differential Odometry Errors in Mobile Robotics with Redundant Sensor Information. The International Journal of Robotics Research, pp.117-128.
- [7] Beyer, J. 1993. Nichtlineare Schätzung inertialer Navigationsgrößen durch die fehlertolerante Verarbeitung zusätzlicher Stützinformation. PhD Thesis, Darmstadt University of Technology, Germany.