

Navigation of a Mobile Robot Using Nonlinear Least Squares Optimization

김 곤 우[†] · 차 영 열^{*}
(Gon-Woo Kim · Young-Youp Cha)

Abstract - The fundamental research for the mobile robot navigation using the numerical optimization method is presented. We define the mobile robot navigation problem as an unconstrained optimization problem to minimize the cost function with the pose error between the goal position and the position of a mobile robot. Using the nonlinear least squares optimization method, the optimal speeds of the left and right wheels can be found as the solution of the optimization problem. Especially, the rotational speed of wheels of a mobile robot can be directly related to the overall speed of a mobile robot using the Jacobian derived from the kinematic model. It will be very useful for applying to the mobile robot navigation. The performance was evaluated using the simulation.

Key Words : Mobile robot, Navigation, Nonlinear least squares, Differential-drive

1. 서 론

이동로봇을 목적지까지 효율적으로 주행시키기 위한 자율 주행 기법에 대한 많은 연구가 진행되고 있으며, 최근 10년간 이동로봇이 다양한 분야와 환경에 적용됨에 따라 자율 주행에 관한 연구가 더욱 활발해지고 있다[1-7].

이동로봇의 자율 주행을 위한 경로계획 방법은 접근 방법에 따라 분류할 수 있다[1-6]. 대표적인 경로 생성 방법으로는 로드맵을 이용한 경로 계획[1] 방법으로 visibility graph, Voronoi diagram, cell decomposition 방식 등을 들 수 있다. 이러한 방법은 시작점부터 도착점까지 로봇을 주행하기 위해 환경을 그래프 형태로 구성하여 그래프 탐색 방법을 이용하여 경로를 생성하는 방식이다. 이렇게 생성된 경로를 따라 주행할 수 있도록 별도의 제어 입력을 계산하여 목적지까지 주행하는 방법이다.

포텐셜 필드 방식[1]은 가상의 필드를 생성하여 장애물과의 충돌 없이 목적지에 도달하도록 하는 주행기술로 인력(attractive potential)과 척력(repulsive potential)을 생성하여 이동로봇을 목적지까지 주행시키는 방식이다. 포텐셜 필드 방식의 경우 경로에 oscillation이 발생하는 단점이 있는데 이를 보완하기 위해 수정된 Newton 방식을 이용한 주행 방법이 제안되었다[6]. Han과 Lee[2]는 원형 경로 계획(circular path planning) 알고리즘을 이용한 이동로봇의 주행 알고리즘을 제안하였다. 또한 로봇의 기구학적 제한요소를

고려하여 장애물을 회피하기 위해 DWA(Dynamic Window Approach)를 이용한 실시간 장애물 회피 알고리즘이 제안되었고[3], Brook[4]은 DWA의 단점인 국부최소(local minima)문제를 해결한 전역 DWA(Global DWA) 알고리즘을 제안하였다. 하지만 전역 DWA의 경우 경유점 근처에서 경로를 벗어나거나 협소한 지역에서의 진입 실패 문제 등이 발생하는 문제가 있다. Konolige[5]는 Gradient method를 이용하여 전역 DWA 문제를 보완하였다. 현재까지 제안된 주행 기술의 경우 대부분 상위 경로를 생성하고, 경로에 따라 주행하기 위한 별도의 처리과정이 필요하다. 이 때 제어과정에서 실험을 바탕으로 파라미터를 설정하는 과정이 필요하다.

본 논문에서는 통합화된 단순한 주행 알고리즘을 제안하고 그 효율성을 시뮬레이션을 통해 검증하였다. 비선형 최적화 방법을 이용하여 이동로봇을 목적지까지 주행시킴에 있어 기구학 모델의 자코비안 행렬을 이용하여 주행 알고리즘의 해를 좌우측 바퀴의 속도로 직접 얻을 수 있는 효율적인 방법을 제안한다. 이렇게 얻어진 좌우측 바퀴의 휠 속도를 이용하여 별도의 변환 없이 직접 로봇을 제어함으로써 효율적이고 간단하게 이동로봇을 목적지까지 안정적으로 주행시킬 수 있다. 본 논문에서 제안하는 방법은 별도의 경로 계획 과정이나 제어입력 계산을 위한 처리과정 없이 실시간으로 직접 로봇의 경로 주행을 가능하며, 시뮬레이션을 통해 제안한 알고리즘의 적합성을 검증하였다.

[†] 교신저자, 정회원 : 원광대학교 전자및제어공학부 조교수
원광대학교 공업기술개발연구소 연구위원

E-mail : kgw0510@wku.ac.kr

^{*} 비회원 : 원광대학교 기계자동차공학부 교수
접수일자 : 2011년 2월 8일
최종완료 : 2011년 5월 24일

2. 차분 구동형 이동로봇 기구학 모델

차분 구동형(differential-drive) 이동 로봇의 기구학 모델을 구하기 앞서 로봇의 위치를 정의하기 위한 2차원 평면에 대한 기준 좌표계를 그림 1과 같이 설정한다. 기준 좌표계

는 고정된 전역 기준 좌표계(global reference frame)과 로봇 기준 좌표계(robot reference frame)로 구성된다.

이동 로봇의 위치는 전역 기준 좌표계과 로봇 기준 좌표계의 관계식을 이용하여 정의할 수 있다. 이동로봇의 위치와 방위각을 표현하는 자세정보는 전역 기준 좌표계 상에서 다음과 같이 정의할 수 있다.

$$\mathbf{q}_R = [x_R \quad y_R \quad \theta_R]^T \quad (1)$$

여기서 x_R, y_R, θ_R 은 각각 이동로봇의 위치와 방위각을 나타낸다.

휠 반경 r 과 휠 간 거리 W 를 갖는 차분 구동형 이동로봇의 기구학 모델은 자코비안 행렬(Jacobian matrix)을 이용하여 다음과 같이 구할 수 있다.

$$\dot{\mathbf{q}}_R = \mathbf{J}_{kin} \cdot \dot{\boldsymbol{\phi}}$$

$$\mathbf{J}_{kin} = \begin{bmatrix} r \cos \theta_R / 2 & r \cos \theta_R / 2 \\ r \sin \theta_R / 2 & r \sin \theta_R / 2 \\ r / W & -r / W \end{bmatrix} \quad (2)$$

여기서 $\dot{\boldsymbol{\phi}} = [\dot{\varphi}_r \quad \dot{\varphi}_l]$ 는 좌우측 휠의 회전 속도, $\dot{\varphi}_r$ 과 $\dot{\varphi}_l$ 로 구성된 벡터를 나타낸다. 이러한 기구학 모델을 이용하면 전역 기준 좌표계에서의 로봇의 속도를 예측할 수 있다.

3. 비선형 최소자승법을 이용한 이동로봇의 주행

3.1 이동로봇 주행을 위한 비용함수

이동로봇을 원하는 위치로 주행시키기 위해서는 이동로봇의 현재 자세 정보와 목적지의 위치정보가 필요하다. 본 논문에서는 목적지의 위치뿐만 아니라 원하는 자세로 로봇을 구동하기 위하여 목적지의 자세정보를 이용하여 이동로봇을 주행시키고자 한다.

비선형 최소자승법을 이용하여 이동로봇을 주행시키기 위해서는 최적화를 위한 비용함수의 정의가 필요하다. 여기서 최적화의 목적은 이동로봇의 현재 자세와 목적지의 자세의 오차를 최소화하는데 있다. 따라서 다음과 같이 오차 함수를 정의한다.

$$\mathbf{e}(\boldsymbol{\varphi}) = \mathbf{q}_R(\boldsymbol{\varphi}) - \mathbf{q}_{Goal} \quad (3)$$

여기서 이동로봇이 목적지를 향해 주행을 할 때 목적지 자세 정보와의 오차를 사용하면 적절한 주행을 할 수 없으므로 목적위치에 일정 오차 이내로 접근할 때까지는 로봇의 방위각과 목적지 방향의 오차를 이용하여 다음과 같이 정의한다.

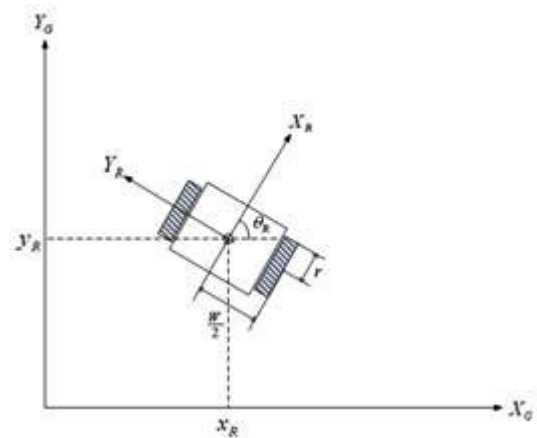


그림 1 기준 좌표계 정의: 전역 기준 좌표계 {G}와 로봇 기준 좌표계 {R}

Fig. 1 Definition of the reference frames: global reference frame {G} and robot reference frame {R}

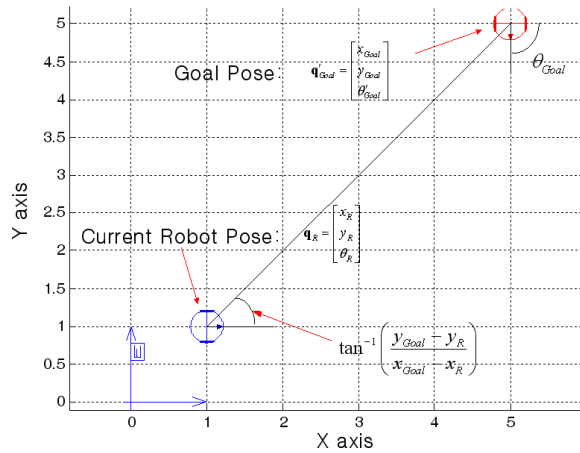


그림 2 전역 기준 좌표계 상의 로봇과 목적지 자세

Fig. 2 The pose of a robot and a goal in the global reference frame

$$\theta'_{Goal} = \begin{cases} \tan^{-1} \left(\frac{y_{Goal} - y_R}{x_{Goal} - x_R} \right) & \text{if } pos. \text{ error} > \varepsilon \\ \theta_{Goal} & \text{else} \end{cases} \quad (4)$$

위와 같이 수정된 목적지 자세정보 \mathbf{q}'_{Goal} 과 위치 오차와 방위 오차의 효율적으로 최소화하기 위한 가중치 행렬을 이용하여 오차 함수를 다음과 같이 다시 정의한다.

$$\mathbf{e}(\boldsymbol{\varphi}) = \mathbf{A} \cdot (\mathbf{q}_R(\boldsymbol{\varphi}) - \mathbf{q}'_{Goal})$$

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \gamma \end{bmatrix} \quad (5)$$

이 때 가중치, γ 는 방위각에 대한 가중치를 결정하는 scale factor로 0보다 큰 값을 가지며 값에 따라 다양한 경로를 생성하며 큰 값을 가질수록 보다 빨리 목적지 방향으로 수렴하여 이동로봇을 효율적으로 주행시킬 수 있다.

가중치 변화에 따른 이동로봇의 주행경로는 그림 3과 같다. 그림 3(a)와 같이 가중치가 1보다 적을 경우 방위각에 대한 수렴이 느려 효율적인 경로를 갖지 않는 것을 알 수 있다. 가중치가 1인 경우 그림 3(b)와 같이 곡선형태의 주행을 하는 것을 알 수 있으며 가중치가 1보다 클수록 보다 빨리 방위각을 목적지로 향하도록 주행경로가 수렴하는 것을 알 수 있다. 따라서 본 논문에서는 γ 를 이동로봇과 목적지와의 거리에 비례하여 설정하여 그림 3(c)와 같이 주행의 효율성을 높였다.

이동로봇을 목적지로 이동시키기 위해 식 (5)에서 정의한 오차 함수를 최소화하기 위한 비용 함수(cost function)를 다음과 같이 정의한다.

$$\mathbf{E}(\boldsymbol{\varphi}) = \frac{1}{2} \mathbf{e}(\boldsymbol{\varphi})^T \mathbf{e}(\boldsymbol{\varphi}) \quad (6)$$

비용함수는 비선형 함수로 오차 함수의 제곱 형태로 정의되며 이를 최소화하기 위해 비선형 최소자승법을 이용한 최적화 방법을 적용한다.

3.2 Gauss-Newton 방법을 이용한 주행 제어

이동 로봇을 원하는 목적 위치와 자세로 주행시키기 위한 비선형 최소자승 최적화 문제를 다음과 같이 정의한다.

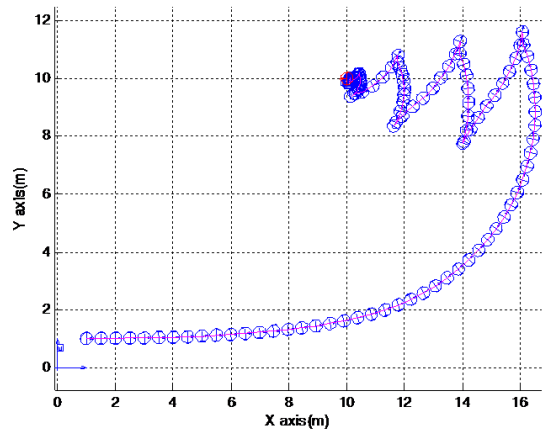
$$\min_{\boldsymbol{\varphi} \in R^2} \mathbf{E}(\boldsymbol{\varphi}) = \min_{\boldsymbol{\varphi} \in R^2} \left\{ \frac{1}{2} \mathbf{e}(\boldsymbol{\varphi})^T \mathbf{e}(\boldsymbol{\varphi}) \right\} \quad (7)$$

여기서 $\partial \mathbf{E}(\boldsymbol{\varphi}^*) / \partial \boldsymbol{\varphi} = 0$ 를 만족하는 쥘 위치, $\boldsymbol{\varphi}^*$ 에서 비용 함수가 최소값을 갖는다. 쥘 위치, $\bar{\boldsymbol{\varphi}}$ 근방에서의 선형적이라 가정하면, 비용 함수는 테일러 급수 확장(Taylor series expansion)을 통해 다음과 같은 근사화된 선형 모델을 얻을 수 있다.

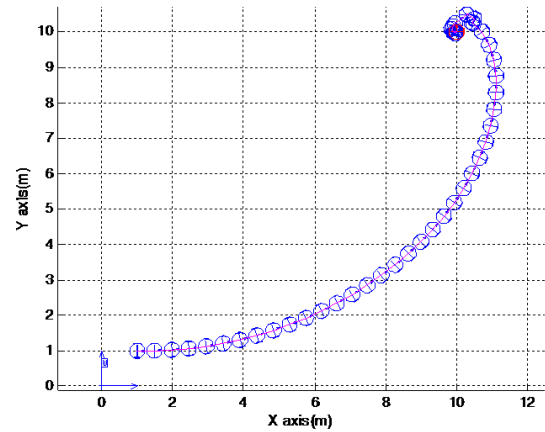
$$\hat{\mathbf{E}}(\boldsymbol{\varphi}) = \mathbf{E}(\bar{\boldsymbol{\varphi}}) + \nabla \mathbf{E}(\bar{\boldsymbol{\varphi}}) \cdot (\boldsymbol{\varphi} - \bar{\boldsymbol{\varphi}}) \quad (8)$$

비용 함수가 $\boldsymbol{\varphi}$ 에 대해 2차 미분 가능하다면 근사 모델 (approximation model)의 변화율(gradient)은 다음과 같다.

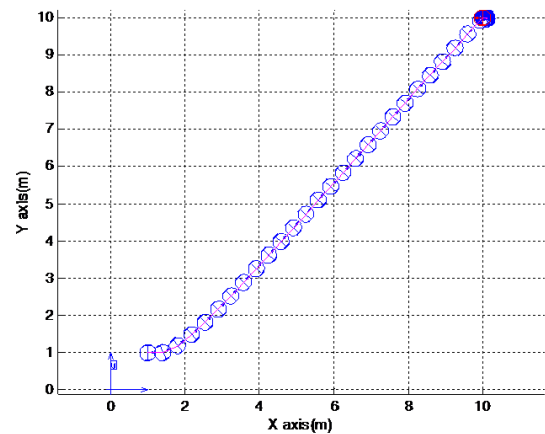
$$\frac{\partial \hat{\mathbf{E}}(\boldsymbol{\varphi})}{\partial \boldsymbol{\varphi}} = \nabla \mathbf{E}(\bar{\boldsymbol{\varphi}}) + \nabla^2 \mathbf{E}(\bar{\boldsymbol{\varphi}}) \cdot (\boldsymbol{\varphi} - \bar{\boldsymbol{\varphi}}) \quad (9)$$



(a)



(b)



(c)

그림 3 오차함수의 가중치 변화에 따른 이동로봇의 주행경로: (a) $\gamma=0.1$ 일 때, (b) $\gamma=1$ 일 때, (c) γ 가 이동로봇과 목적지와의 거리오차에 비례

Fig. 3 Path of a mobile robot according to the change of the weighting factor: (a) $\gamma=0.1$, (b) $\gamma=1$, (c) proportion of the distance between the robot and the goal

이 때 φ_k 에서 식 (9)의 근사 모델의 변화율이 0이 된다면 비용 함수를 최소화 하는 해가 될 것이다. 따라서 식 (2)에서 정의한 기구학 모델의 자코비안 행렬, $J_{kin}(\varphi) = \partial q_R / \partial \varphi$ 을 이용하여 비용 함수의 1차, 2차 미분을 구하면 다음과 같다.

$$\begin{aligned} \nabla \mathbf{E}(\varphi) &= \sum_{i=1}^n e_i(\varphi) \cdot \nabla e_i(\varphi) \\ &= \mathbf{J}_{kin}(\varphi)^T \mathbf{e}(\varphi) \end{aligned} \quad (10)$$

$$\begin{aligned} \nabla^2 \mathbf{E}(\varphi) &= \sum_{i=1}^n \left(\nabla e_i(\varphi) \cdot \nabla e_i(\varphi)^T + e_i(\varphi) \cdot \nabla^2 e_i(\varphi) \right) \\ &= \mathbf{J}_{kin}(\varphi)^T \mathbf{J}_{kin}(\varphi) + \mathbf{R}(\varphi) \end{aligned} \quad (11)$$

여기서 e_i 는 식 (5)에서 정의한 오차함수의 component를 의미하며, $\mathbf{R}(\varphi) = \nabla (\mathbf{J}_{kin}(\varphi))^T \mathbf{e}(\varphi)$ 는 헤시안(Hessian) 항목을 포함하는 레지듀얼(residual) 항목으로 무시할 수 있다. 식 (10), (11)에서 구한 1차, 2차 미분을 식 (9)에 적용하여 비용 함수를 최소화 하는 휠 위치 해를 반복적 형태로 구하면 다음과 같다.

$$\varphi_{k+1} = \varphi_k - \left(\mathbf{J}_{kin}(\varphi_k)^T \mathbf{J}_{kin}(\varphi_k) \right)^{-1} \mathbf{J}_{kin}(\varphi_k)^T \mathbf{e}(\varphi_k) \quad (12)$$

위 식을 이용하면 목적지까지 이동로봇을 주행시킴에 있어서 직접 좌우측 휠의 속도 제어가 가능하여 단순하고 효과적인 주행이 가능하다. 속도 제어를 위해 일정한 제어 주기 Δt 를 이용하여 휠 속도를 얻을 수 있다.

3.3 속도 구속 조건(Velocity Constraint)을 고려한 휠 속도 제어

3.2절에서 얻은 휠의 속도의 경우 이동 로봇의 속도 제한 범위를 넘는 해를 제공할 가능성이 매우 크다. 따라서 본 절에서는 속도 제한 조건을 이용하여 이동로봇 좌우측 휠을 속도 제한을 위한 효율적인 방법을 제시한다.

만약 이동로봇의 최고 속도를 v_{max} 라 하였을 때, 휠의 속도는 다음과 같은 구속 조건을 갖는다.

$$\dot{\varphi}_l = \omega_l \text{ or } \dot{\varphi}_r = \omega_r \leq \omega_{max} \quad (13)$$

여기서 r 은 휠의 반경을 말한다. 따라서 휠의 최고 속도 ω_{max} 는 이동로봇의 최고 속도에 따라 $\omega_{max} = v_{max}/r$ 과 같이 정의할 수 있다. 여기서 휠의 속도를 최고 속도를 기준으로 제한할 때 식 (12)에서 구한 좌우측 휠 속도를 비율적으로 제한하여야 이동 로봇의 회전 속도 제어에 문제가 발생하지 않고 효율적으로 주행할 수 있다. 이를 위해 다음과 같이 계산된 최대 휠의 속도를 정의한다.

$$\dot{\varphi}_{max} = \max(\dot{\varphi}) \quad (14)$$

위 식에서 $\max()$ 함수는 최대값을 갖는 element를 찾는 함수로 정의한다. 여기서 휠의 최대 속도 $\dot{\varphi}_{max}$ 는 스칼라 값으로 만약 휠의 최대 속도가 식 (13)에서 정의한 최대 속도 ω_{max} 를 넘어서는 경우 다음과 같이 속도 구속 조건을 고려하여 휠의 속도를 제어한다.

$$\dot{\varphi}_{scaled} = \begin{cases} \frac{\omega_{max}}{\dot{\varphi}_{max}} \dot{\varphi} & \text{if } \dot{\varphi}_{max} > \omega_{max} \\ \dot{\varphi} & \text{else} \end{cases} \quad (15)$$

속도 구속 조건을 고려하여 휠의 속도를 비례적으로 제한함으로써 보다 효율적으로 이동 로봇을 제어할 수 있다. 또한 (15)와 같이 비례적으로 속도를 제한하는 경우 회전 속도 제어의 방해요소로 작용하지 않아 효율적인 주행을 가능하게 하며, 이를 시뮬레이션을 통해 검증하였다.

4. 시뮬레이션 결과

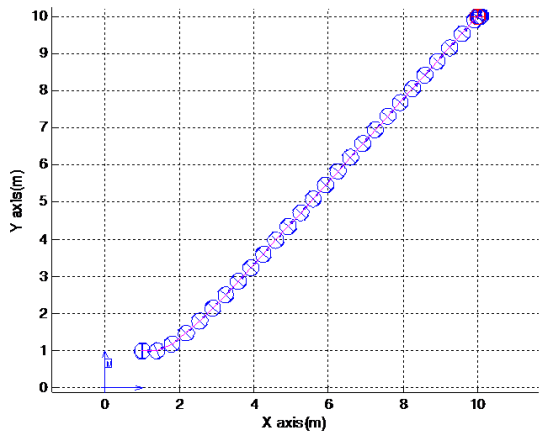
본 장에서는 제안한 주행 알고리즘을 적용하여 시뮬레이션을 수행하였다. 식 (5)에서 가중치는 로봇과 목적지와의 거리를 이용하여 다음과 같이 적용하였다.

$$\gamma = \begin{cases} d & \text{if } d > 1 \\ 1 & \text{else} \end{cases} \quad (16)$$

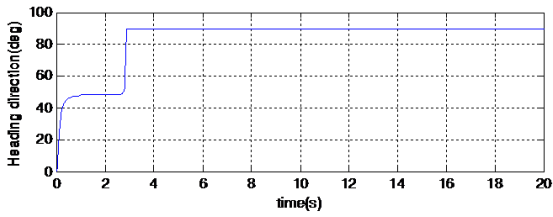
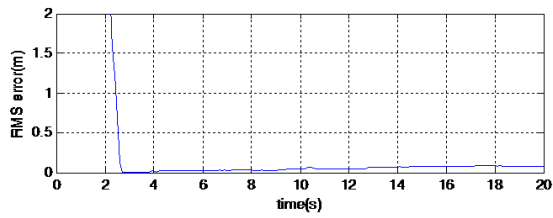
위 식에서 d 는 로봇과 목적지까지의 유클리디언 거리로 $d = \sqrt{(x_{Goal} - x_R)^2 + (y_{Goal} - y_R)^2}$ 와 같이 정의하며 단위는 m로 정의한다. 이동로봇의 초기 자세는 $[1 \ 1 \ 0]^T$ 이며 이동로봇의 제어 주기는 100ms로 가정하였으며 정상상태 오차를 고려하여 $\|e(\varphi_k)\|_2 \leq 0.1m$ 이면 정상상태로 판정하였다. 또한 휠의 속도 제어를 위한 이동로봇의 최대 속도를 5m/s로 설정하였으며 휠의 반경을 20cm로 설정하여 휠의 최대 속도가 25rad/s를 넘지 않도록 제한하였다. 이 때의 로봇의 회전 속도와 방위각은 그림 4(b), (c)와 그림 5(b), (c)에서 보는 바와 같다. 이동 로봇의 최대 속도로 제한하였을 때 이동 로봇의 회전 속도 제어가 효율적이고 안정적으로 수행되었음을 알 수 있다.

제안된 알고리즘의 효율성을 검증하기 위해 목적지의 위치 및 방위각을 달리하여 이동로봇의 주행 궤적, 이동로봇의 직선 및 회전 속도, 그리고 위치오차 및 방위각에 대한 결과를 분석하였다.

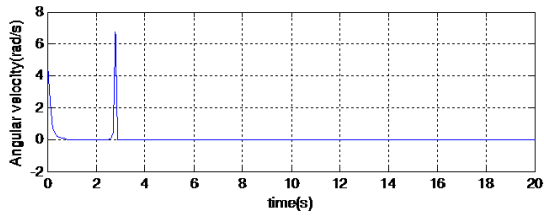
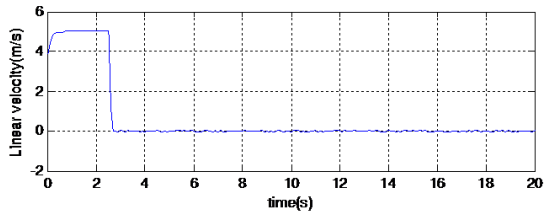
먼저 목적지의 자세정보가 $[10 \ 10 \ 90]^T$ 일 때의 시뮬레이션 결과는 그림 4와 같다. 이동로봇은 효율적인 경로를 따라 목적지에 접근하는 것을 알 수 있으며, 위치오차 범위 이내로 접근하여 빠르게 목적지 자세정보에 따라 이동로봇의 방위각을 보정함을 알 수 있다. 목적지까지 도달하는데 2.7sec의 시간이 소요되었으며 위치 오차는 0.0235m로 매우 효율적으로 주행하는 것을 확인하였다.



(a)



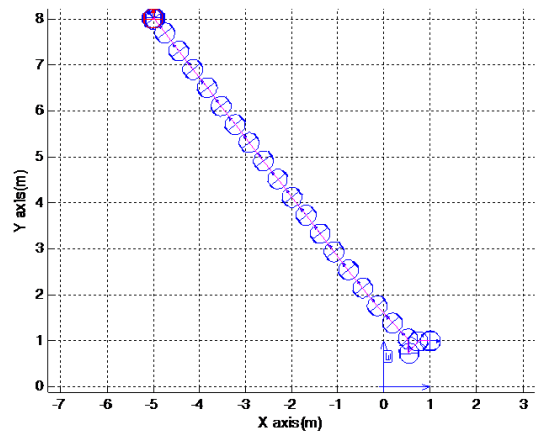
(b)



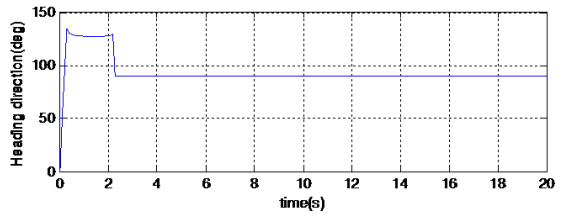
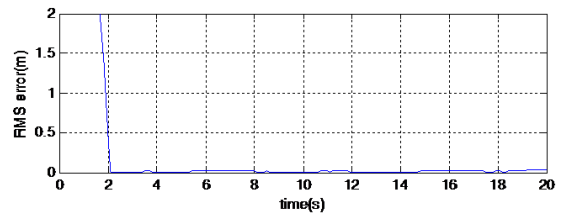
(c)

그림 4 시뮬레이션 결과: $\dot{q}_{Goal} = [10 \ 10 \ 90^\circ]^T$ 일 때 (a) 이동 로봇의 궤적, (b) 위치 오차와 이동로봇의 방위각 (c) 이동로봇의 직선 및 회전속도

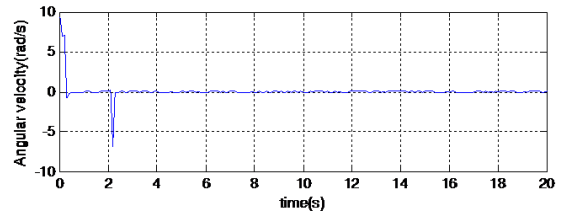
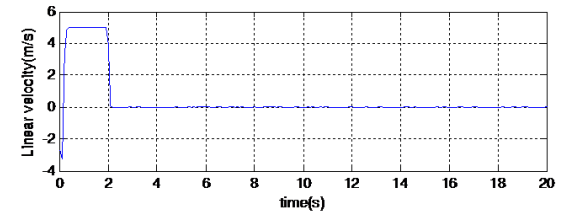
Fig. 4 Simulation results: (a) trajectory, (b) pose of the robot and (c) linear and rotational velocities



(a)



(b)



(c)

그림 5 시뮬레이션 결과: $\dot{q}_{Goal} = [-5 \ 8 \ 90^\circ]^T$ 일 때 (a) 이동 로봇의 궤적, (b) 위치 오차와 이동로봇의 방위각 (c) 이동로봇의 직선 및 회전속도

Fig. 5 Simulation results: (a) trajectory, (b) pose of the robot and (c) linear and rotational velocities

그림 5는 목적지의 자세정보가 $[-5 \ 8 \ 90]^T$ 일 때의 시물레이션 결과이다. 흥미로운 것은 방위각을 빠르게 보정하기 위해 최초 로봇의 자세인 $[1 \ 1 \ 0]^T$ 에서 후방으로 V 형태의 궤적을 그리며 주행을 하는 점이다. 이는 식 (16)에서의 가중치 값을 로봇과 목적지와의 거리에 비례하여 큰 값을 갖게 함으로써 주행 궤적의 효율성을 높일 수 있었음을 확인할 수 있었다. 목적지까지 도달하는데 2.2sec의 시간이 소요되었으며 위치오차는 0.0371m로 목적지의 자세정보가 바뀌어도 효율적으로 주행하는 것을 확인하였다.

5. 결 론

본 논문에서 제안한 방법을 이용하면 매우 효과적으로 이동로봇을 목적지까지 주행할 수 있음을 시물레이션을 통해 보였다. 제안한 방법은 실내/외 이동로봇의 주행에 광범위하게 적용될 수 있을 것이며, 매우 실용적으로 이동로봇을 제어할 수 있을 것으로 판단된다.

향후 다수의 장애물이 존재하는 경우 수정된 비용함수를 적용하여 효율적으로 장애물을 회피하여 주행을 할 수 있는 방법에 대한 연구를 진행하고 실험을 통해 이를 검증할 예정이다.

감사의 글

본 연구는 2010년도 원광대학교의 교비 지원에 의해서 수행되었습니다.

참 고 문 헌

[1] R. Siegwart, and I. R. Nourbakhsh, *Introduction to Autonomous Mobile Robots*, The MIT Press, 2004.

[2] S. M. Han and K. W. Lee, "Mobile Robot Navigation using Circular Path Planning Algorithm," Int'l Conf. Control, Automation and Systems, pp. 2082-2086, 2008.

[3] D. Fox, W. Burgard, S. Thrun, "The Dynamic Window Approach to Collision Avoidance," IEEE Robotics and Automation Magazine, 4:23-33, 1997.

[4] O. Brock, and O. Khatib, "High-Speed Navigation Using the Global Dynamic Window Approach," IEEE Int'l Conf. Robotics and Automation, 1999.

[5] K. Konolige, "A Gradient Method for Real-Time Robot Control," Int'l Conf. Intelligent Robots and Systems, pp. 639-646, 2000.

[6] J. Ren, K. A. McIsaac, and R. V. Patel, "Modified Newton's Method Applied to Potential Field-Based Navigation for Mobile Robots," IEEE Trans. Robotics, pp.384-391, APR. 2006

[7] G. W. Kim, K. T. Nam, S. M. Lee, and W. H. Shon, "Visual Servo Navigation of a Mobile Robot using Nonlinear Least Squares Optimization for Large Residual," J. Korea Robotics Society, vol. 2, no. 4, pp. 326-332, 2007.

저 자 소 개



김 곤 우 (金 坤 佑)

1974년 5월 10일생. 2006년 서울대학교 전기컴퓨터공학부 대학원(공학박사). 2006년~2008년 한국생산기술연구원 로봇기술본부 위촉연구원, 2008년~현재 원광대학교 전자및제어공학부 조교수. 관심분야는 Visual servoing, Robot Navigation, Sensor Fusion, Map Building and Localization.



차 영 엽

1984년 부산대 기계공학과 졸업. 1987년 한국과학기술원 생산공학과 공학석사. 1995년 한국과학기술원 정밀공학과 공학박사. 1995년~현재 원광대학교 기계자동차공학부 교수. 관심분야는 이동로봇, 영상처리.