

이동로봇을 위한 Limit-cycle 항법

Limit-cycle Navigation Method for Fast Mobile Robots

류근호, 김동한*

(Keun-Ho Rew and Donghan Kim)

Abstract : A mobile robot should be designed to navigate with collision avoidance capability in the real world, coping with the changing environment flexibly. In this paper, a novel navigation method is proposed for fast moving mobile robots using limit-cycle characteristics of the 2nd-order nonlinear function. It can be applied to robots in dynamically changing environment such as the robot soccer. By adjusting the radius of the motion circle and the direction of the obstacle avoidance, the mobile robot can avoid the collision with obstacles and move to the destination point. To demonstrate the effectiveness and applicability, it is applied to the robot soccer. Simulations and real experiments ascertain the merits of the proposed method.

Keywords : 2nd-order nonlinear function, limit-cycle, navigation, robot soccer

I. 서론

항법은 로봇의 내부와 외부에 있는 제한 조건 안에서 주어진 일을 수행하도록 특정한 위치로 로봇을 움직이도록 하는 것이다. 그러기 위해서는 로봇은 자신 주위에 대한 환경을 인식하고 변화하는 환경을 인식할 수 있는 센서를 이용하여 항법 계획을 세워야 한다. 실제 많은 환경에서 쓰여왔던 이동 로봇의 항법은 크게 *deliberative* 방법과 *reactive* 방법, 두 가지로 나눌 수 있다. *deliberative* 방법은 주위 환경에 대한 정확한 모델을 가지고 임무 수행에 필요한 계획을 만드는 것으로, 이론적으로 센싱, *planning* 그리고 *acting*의 세 단계로 나눌 수 있으며, 이를 통해서 원하는 목표까지의 궤적을 만들어 나가게 된다. 하지만 실제 실험하는 경우엔 매순간 환경이 변하는데 따른 새로운 경로 계획 수립과 환경에 대한 불확실성으로 야기되는 실패 때문에 많은 문제가 발생한다. 궤적을 만드는 앞서의 세 과정을 빠르게 반복한다면 환경의 불확실성을 해결하는 강인한 해결 방법이 될 수 있다. 하지만, 많은 계산량으로 인해 실제적인 적용시 빠른 대처가 힘들다. 이에 반해, *reactive* 방법[3-5]은 로봇 주위의 환경에 대해서 모델이 조금 필요하거나 거의 필요없으며, 로봇의 센서와 구동기 사이에 적절한 전달 함수를 이용해서 연결시킨다. 일반적으로 비교적 적은 계산량을 필요로 하므로, 로봇으로 하여금 아주 적은 시간 동안에 행동을 취하도록 하고, 지속적인 행동의 계산을 통해서 환경의 불확실성을 이겨낼 수 있다. 그렇지만, 표현과 추론 방법의 한계로 인해 순수한 *reactive* 시스템은 목표지향적이지 못하다.

로봇 축구는 환경이 능동적으로 계속 변화하는 전형적인 다개체 시스템의 예이며, 같은 팀과는 협동을 해야하고 상대 팀과는 경쟁을 해야한다. 뿐만 아니라, 제어 방법의 효율을

골득실로 평가할 수 있다.

로봇 축구에서 당연한 문제는 비전 시스템을 통한 정확한 위치 추적, 위치를 이용한 항법 계획, 계획된 경로 추종 등의 세 가지로 크게 나눌 수 있다. 이 중에 항법 계획은 로봇 축구팀의 전력을 판가름하는 중요한 잣대가 된다. 항법의 관점에서 볼 때, 로봇 축구는 이동 장애물과 이동 목표물이 있는 능동적인 환경에서의 항법이다. 심현식[6]은 공 주위를 네 개의 구역으로 나누어 각기 다른 제어기를 적용하는 방법인 *four-mode* 제어 구조를 제안했다. 김용재[5]는 공의 앞뒤에 두 점을 정하고, 이 점들에서 현재 위치까지의 *vector*들 사이의 각도를 구한 후, 임의의 정수를 곱한 각도를 로봇이 따라가게 하는 방법인 *uni-vector field* 법을 제안했다. 이문수[7]는 기하학적인 방법과 퍼지 논리를 합쳐서 항법을 만든 후, 이를 진화 연산을 통해서 최적화시켰다.

본 논문에서는 장애물이나 로봇이 추가/감소되거나 환경이 계속 변화하는 상황에서도 적용될 수 있는 새로운 *limit-cycle* 항법을 제안한다. 이 방법은 *reactive* 방법의 유연성과 *deliberative* 방법의 목표지향적이라는 장점을 합친 것으로, 경로를 얻기 위해 2차 비선형 함수의 *limit-cycle* 특성을 이용한다. 이를 통해 *local minima*와 원치 않는 장애물의 영향을 극복하며, 로봇과 목표 사이에 여러 개의 장애물이 있는 경우에도 쓰일 수 있도록 확장한다.

II 절에서는 *limit-cycle* 항법을 제안하고, II 절에서 로봇 축구에 맞추어 확장한다. 모의실험과 실제 실험 결과는 IV 절에서 보여지며, V 절에서 결론을 맺는다.

II. 항법을 위한 Limit-cycle 방법

1. 이전의 연구

지금까지 많은 *reactive* 방법이 제안되어 왔으며, 이들 방법은 로봇의 센서와 구동기를 적절한 전달 함수로 연결함으로써 로봇 주위 환경에 대한 모델이 거의 필요하지 않으며, 비교적 적은 계산량을 필요로 하는 등의 장점을 가지고 있다. *Potential field* 법은 *reactive* 방법의 대표적인 방법으로 인력과 척력을 이용해서 센서와 구동기를 연결하는 것으로 로봇의 현재 위치에서 각각의 *gradient*들의 효과를 합쳐서 로봇의 행동을 결정한다. 예를 들어, 그림 1에서와 같이 로봇은 장애물

* 책임저자 (Corresponding Author)

논문접수 : 2008. 6. 26., 채택확정 : 2008. 8. 11.

김동한 : 경희대학교 전자정보대학(donghani@khu.ac.kr)

류근호 : 호서대학교 로봇공학과(khrew@hoseo.edu)

※ 본 연구는 지식경제부의 지방기술혁신사업(RTI04-01-02) 지원으로 수행 되었음.

※ 본 연구는 2007년도 경희대학교 연구비지원에 의한 결과 임. (KHU-20070778)

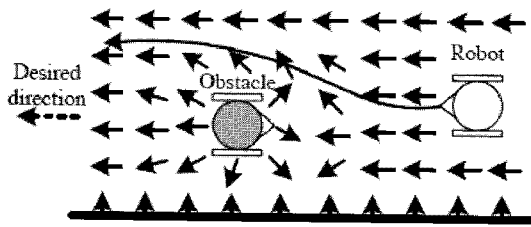
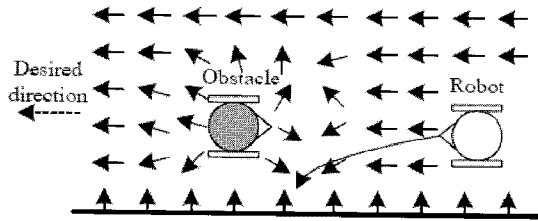
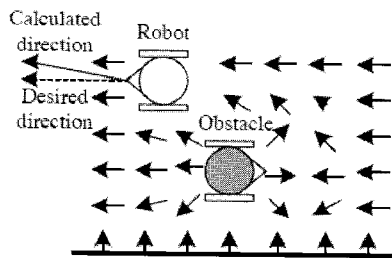


그림 1. Potential field 법.
Fig. 1. Potential filed method.



(a) moeel with local minima.



(b) undesirable obstacle influence.

그림 2. Potential field 법의 단점들.
Fig. 2. Disadvantages of potential field method.

을 피해 경기장을 가로 지르려고 한다고 할 때, 경기장의 왼쪽에 목표점은 인력을 주고, 장애물과 경기장 벽은 척력을 준다. 이들 힘의 합을 구하면 각 지점에서의 로봇의 경로가 결정된다.

하지만, 몇 가지 단점들이 potential field 법의 사용을 제한한다. 장애물을 피하는데 있어서 어느 방향으로 피할지의 정보를 쓰지 않기 때문에 local minima에 빠지기 쉽다. 그래서 그림 2(a)와 같이 로봇이 장애물인 벽 근처에 있을 때, potential field 법은 로봇이 벽 근처에 갇히도록 한다. 또 다른 단점으로는, 로봇의 진행 방향이 아닌 장애물에 대해서도 척력과 같은 불필요한 힘을 받기 때문에 그림 2(b)과 같이 원하지 않는 방향으로 로봇이 움직이게 된다. 이러한 단점들을 극복하려는 여러 가지 시도가 있어 왔는데, Gaussian 잡음을 더해서 local minima를 극복하기도 하고[8-9], Slack은 회전하는 field를 장애물 주위에 놓는 navigation template 법을 제안했다. 또한 Rimon과 Koditschek 은 potential function을 정의하여 이러한 단점을 극복했다[12].

본 절에서는 potential field 법의 단점들을 극복할 뿐만 아니라 그림 3과 같은 로봇 축구 상황에서 로봇이 상대 로봇을 피하면서 공을 상대 골대 쪽으로 차게 할 수 있는 새로운 방법을 제안하고자 한다.

2. 이동 로봇의 모델링

본 논문에서는 미끄러짐이 없고 순수하게 구름만이 있는

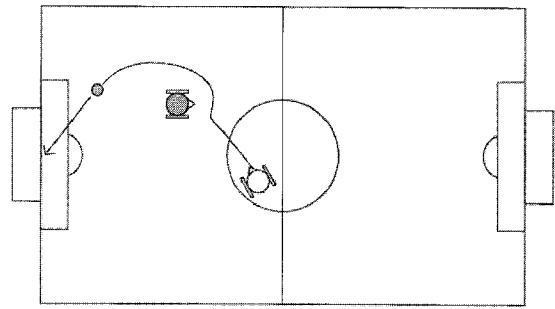


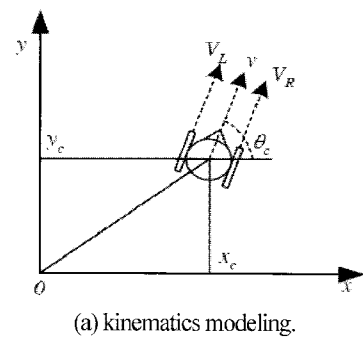
그림 3. 로봇축구.
Fig. 3. Robot soccer

차동 구동 방식의 이동 로봇을 고려한다. 속도 벡터, $Q = [v \ w]^T$ 는 로봇의 중심 속도, v 와 로봇의 중심을 기준으로 하는 회전 각속도, w 로 구성된다. 이 속도 벡터 Q 와 자세 벡터, $P = [x \ y \ \theta]^T$ 와는 다음의 로봇 기구학식으로 관련지어질 수 있다.

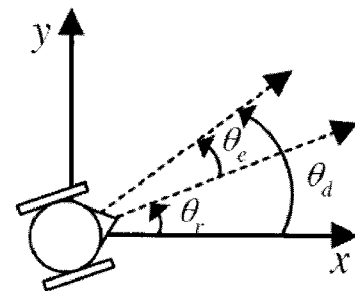
$$\dot{P} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} = J(\theta)Q \quad (1)$$

$$Q = [v \ \omega]^T = \begin{bmatrix} \frac{V_r + V_l}{2} & \frac{V_r - V_l}{L} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{L} & \frac{1}{L} \end{bmatrix} \begin{bmatrix} V_l \\ V_r \end{bmatrix} \quad (2)$$

여기서 V_l 과 V_r 는 각각 로봇의 왼쪽, 오른쪽 바퀴의 속도이다. 그러므로, 로봇은 이 V_l 과 V_r 로 자세를 제어할 수 있다. 그림 4(b)과 같이, 현재 위치 (x, y) 에서 로봇의 원하는 방향인 θ_d 을 계산하고, 로봇의 각도 θ_r 를 이용하면, 다음과 같이 각도 오차 θ_e 가 구해진다:



(a) kinematics modeling.



(b) error of angle.

그림 4. 로봇 모델링.
Fig. 4. Robot modelling.

$$\theta_e = \theta_d - \theta_r \tag{3}$$

이를 이용해서 로봇이 원하는 방향 θ_d 로 움직이게 하는 제어 방법은 다음의 식과 같이 구할 수 있다.

$$\begin{aligned} V_l &= v - K_p \cdot \theta_e - K_d \cdot \dot{\theta}_e \\ V_r &= v + K_p \cdot \theta_e + K_d \cdot \dot{\theta}_e \end{aligned} \tag{4}$$

여기서 K_p 와 K_d 는 각각 비례 이득과 미분 이득이고, v 는 원하는 로봇의 중심 속도이다. 이동 거리가 멀면 빠른 속도로 이동하다가 거리가 가까워 짐에 따라 중심 속도를 줄여야 하기 때문에 이를 거리 오차와 관련시켜 변화시켜야 한다. 가장 간단하게 생각할 수 있는 방법은 거리 오차에 비례하게 하는 것이다.

3. Limit-cycle

다음의 2차 비선형 시스템을 고려하자[11].

$$\begin{aligned} \dot{x}_1 &= x_2 + x_1(1 - x_1^2 - x_2^2) \\ \dot{x}_2 &= -x_1 + x_2(1 - x_1^2 - x_2^2) \end{aligned} \tag{5}$$

그리고, 이 함수의 Lyapunov 함수를 다음과 같이 가정한다.

$$V(x) = x_1^2 + x_2^2$$

시스템의 궤적을 따라 $V(x)$ 를 미분한 값은 다음과 같다.

$$\begin{aligned} \dot{V}(x) &= 2x_1\dot{x}_1 + 2x_2\dot{x}_2 \\ &= 2x_1x_2 + 2x_1^2(1 - x_1^2 - x_2^2) \\ &\quad - 2x_1x_2 + 2x_2^2(1 - x_1^2 - x_2^2) \\ &= 2V(x)(1 - V(x)). \end{aligned} \tag{6}$$

위의 식에서 $V(x)$ 의 미분값은 $V(x) < 1$ 인 경우엔 양수이고, $V(x) > 1$ 인 경우엔 음수이다. 그러므로, $0 < c_1 < 1$ 의 경우 $V(x) = c_1$ 인 level 평면에서는 모든 궤적이 밖으로 향하고, $c_2 > 1$ 인 경우엔 $V(x) = c_2$ 의 level 평면에서는 모든 궤적이 안으로 향한다. 이것은 다음과 같은 주기적인 영역,

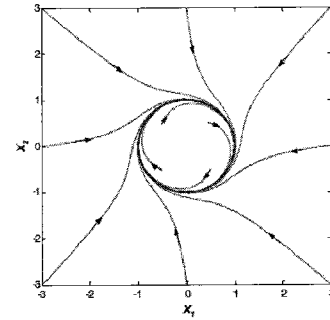
$$M = \{x \in R^2 \mid c_1 \leq V(x) \leq c_2\}$$

이 positively invariant하다는 뜻이다. 또한 닫혀있고, bounded 되어 있으며, 원점 $x=0$ 이 유일한 equilibrium point이기 때문에 Poincare-Bendixson 정의에 의해 M 안에는 한 개의 주기적인 궤도가 존재한다. 이는 어떠한 $c_1 < 1$ 과 어떠한 $c_2 > 1$ 에도 성립하므로, c_1 과 c_2 는 1에 접근한다. 결국엔 집합 M 은 그림 5와 같이 단위원으로 수렴한다. 이런 주기적인 궤도를 limit-cycle이라고 한다.

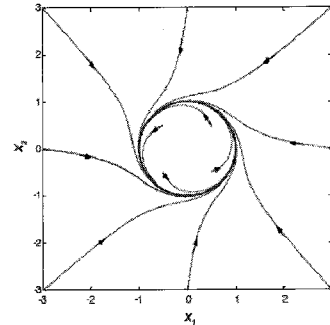
그림 5(a)은 (5)의 x_1 과 x_2 을 x -축과 y -축에 각각 대응시켜 phase portrait을 나타낸 것이다. 원점을 포함한 모든 점 (x, y) 에서의 궤적은 앞서 설명한 시계방향의 단위원으로 수렴한다. 반시계방향의 단위원으로 수렴시키기 위해서는 앞의 2차 비선형식 (5)를 다음과 같이 바꿔야 한다.

$$\begin{aligned} \dot{x}_1 &= -x_2 + x_1(1 - x_1^2 - x_2^2) \\ \dot{x}_2 &= x_1 + x_2(1 - x_1^2 - x_2^2). \end{aligned} \tag{7}$$

그러면, 모든 궤적은 그림 5(b)와 같이 반시계 방향의 단위



(a) phase portrait of limit cycle (clockwise).



(b) phase portrait of limit cycle (counterclockwise).

그림 5. Limit-cycle의 모양.

Fig. 5. Limit-cycle.

원으로 수렴한다. 단위원이 아닌 일반적인 반지름 r 에 대한 (5)의 일반적인 형태는 다음과 같다.

$$\begin{aligned} \dot{x}_1 &= x_2 + x_1(r^2 - x_1^2 - x_2^2) \\ \dot{x}_2 &= -x_1 + x_2(r^2 - x_1^2 - x_2^2). \end{aligned} \tag{8}$$

그리고, Lyapunov 함수는 동일하며,

$$V(x) = x_1^2 + x_2^2$$

시스템의 궤적을 따라 $V(x)$ 를 미분한 값은 다음과 같다.

$$\begin{aligned} \dot{V}(x) &= 2x_1\dot{x}_1 + 2x_2\dot{x}_2 \\ &= 2x_1x_2 + 2x_1^2(r^2 - x_1^2 - x_2^2) \\ &\quad - 2x_1x_2 + 2x_2^2(r^2 - x_1^2 - x_2^2) \\ &= 2V(x)(r^2 - V(x)). \end{aligned} \tag{9}$$

그러므로, (8)을 이용해서 단위원이 아닌 임의의 반지름 r 을 갖는 원으로 좌표상의 모든 점에서 시작한 궤적을 수렴시킬 수 있다.

지금까지 limit-cycle의 방향과 크기를 조절하는 방법을 설명했다. 앞으로 이를 이용해서 limit-cycle 항법을 만들도록 한다. 이 방법은 로봇으로 하여금 장애물을 회피하는데 어느 방향으로, 어느 정도로 피할지를 정할 수 있게 한다.

4. Limit-cycle 항법

본 절에서는 앞서 설명한 limit-cycle을 이용해서 장애물을 피해 목표점까지 가게 하는 local 항법을 구현하고자 한다. 그림 6은 limit-cycle 항법의 개념을 그린 것이다. 이 때, 로봇은 장애물을 피하는 방향 즉, 시계방향인지 반시계 방향인지와 장애물을 회피하는 거리를 정해야 한다. 그림 7은 가장

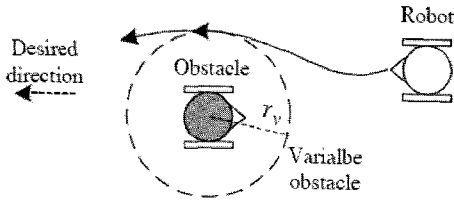


그림 6. Limit-cycle 항법.
Fig. 6. Limit-cycle navigation.

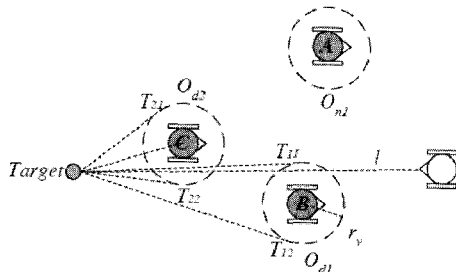


그림 7. 로봇 축구 상황.
Fig. 7. Robot soccer situation.

오른쪽에 있는 로봇이 세 개의 장애물을 피해서 목표인 공으로 움직이는 로봇 축구 상황을 보여준다.

Limit-cycle을 항법에 이용하기에 앞서 몇 가지 용어를 다음과 같이 정의한다.

- 회전 방향 (R_d): 장애물을 시계방향 (CW)으로 피할지, 반시계방향 (CCW)으로 피할지의 방향을 정한다.
- Variable 장애물 (O_v): 일반적으로, 모의 실험 상황에서 로봇은 점질량으로 가정한다. 하지만 이것은 자칫 실제 실험에서 장애물과의 충돌을 유발할 수 있다. 본 논문에서는 로봇과 장애물의 크기, 상대 위치에 따라 반지름이 변하는 variable 장애물을 정의한다. 실험을 간단하게 하기 위해, 모든 variable 장애물은 원으로 가정한다. 이렇게 함으로써 variable 장애물 O_v 원을 limit-cycle의 회전원이라고 하면 로봇이 장애물과 충돌없이 피할 수 있다. 자세한 설명은 다음 절에서 설명하겠다.
- Variable 반지름 (r_v): variable 장애물의 반지름이다. 로봇과 장애물의 크기와 상대 위치에 따라 변한다. Limit-cycle의 회전 반지름을 r_v 라고 하면, 로봇은 장애물을 충돌없이 피할 것이다.
- Disturbing 장애물 (O_d): 로봇과 목표 사이에 있는 variable 장애물을 말한다. 이 장애물은 로봇으로부터의 거리에 따라 번호를 부여 받는다. 예를 들어 가장 가까운 disturbing 장애물은 O_{d1} 라고 한다.
- Non-disturbing 장애물 (O_n): 로봇과 목표 사이에 없는 장애물을 말한다.
- 접점 (T_n, T_d): variable 장애물의 접선 중에서 목표점을 지나는 접선의 접점이다. 각 장애물마다 두 개의 접점이 존재한다.

이제부터, limit-cycle을 이용한 항법의 과정을 설명한다.

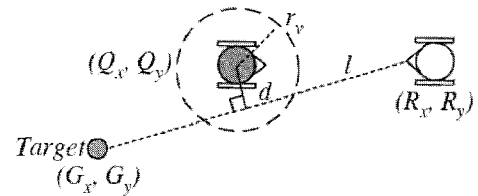


그림 8. 회진방향의 계산.
Fig. 8. Direction of obstacle avoidance.

1. Global 좌표계, Σ_{XYO} 에서 로봇과 목표 사이를 잇는 직선 l 을 다음과 같은 식을 이용하여 그린다.

$$ax + by + c = 0 \quad (10)$$

2. 직선 l 이 지나고 있는 variable 장애물은 disturbing 장애물, O_d 로 간주하고, 지나지 않는 장애물은 non-disturbing 장애물, O_n 으로 간주한다.
3. O_d 가 없다면 목표로 향해서 간다.
4. 그림 8에서, 가장 가까운 disturbing 장애물의 중심에서 직선 l 까지의 거리 d 를 다음의 식으로 구한다.

$$d = \frac{aQ_x + bQ_y + c}{\sqrt{a^2 + b^2}} \quad (11)$$

여기서 (Q_x, Q_y) , (G_x, G_y) 와 (R_x, R_y) 는 각각 장애물, 목표 그리고, 로봇의 xy -값이 된다. (5)는 d 와 r_v 를 추가하고, x_1 과 x_2 를 global 좌표계 Σ_{XYO} 의 x 와 y 으로 바꾸면, 다음과 같이 확장된다:

$$\begin{aligned} \dot{x} &= \frac{d}{|d|}y + x(r_v^2 - x^2 - y^2) \\ \dot{y} &= -\frac{d}{|d|}x + y(r_v^2 - x^2 - y^2) \end{aligned} \quad (12)$$

여기서 x 와 y 는 장애물로부터의 상대 위치이다. 위의 식에서 d 가 양수이며, 로봇은 장애물 O_d 를 시계방향으로 피한다. 마찬가지로 음수이면, 반시계 방향으로 피한다. r_v 는 다음의 식과 같이 로봇과 장애물의 크기와 상대 위치에 따라 정해진다.

$$r_v = r_r + r_o + \delta, \quad (13)$$

여기서 r_r 과 r_o 는 각각 로봇과 장애물의 크기이다. r_o 는 장애물이 disturbing 장애물인 경우엔 실제 장애물의 반지름인 r 이고, non-disturbing 장애물인 경우엔 0이다. δ 는 충돌 회피를 위한 안전 여유값이다.

로봇이 움직이면서 l 은 계속 변하고, 이에 따라 step 2 ~ 4를 임무를 완수할 때까지 반복한다.

예를 들어, 그림 9를 보자. 로봇과 목표 사이에 세 개의 장애물이 있다. 로봇은 A, B와 C로 표시된 장애물을 피해서 목표로 가야 한다. 우선 로봇에서 목표까지 직선 l 을 그린다 (step 1). 이 직선은 그림에서처럼 장애물 B와 C를 지나므로, 이들을 각각 O_{d1} 과 O_{d2} 라 하고, 장애물 A를 O_n 이라고 한다 (step 2). 직선 l 이 O_{d1} 을 지날 때 중심에서부터의 거리를 구하고, 이를 이용해서 장애물을 회피하는 방향을 정한다. 로봇은 그림 9(a)에서처럼 반시계 방향으로 장애물 B를 피한다 (step

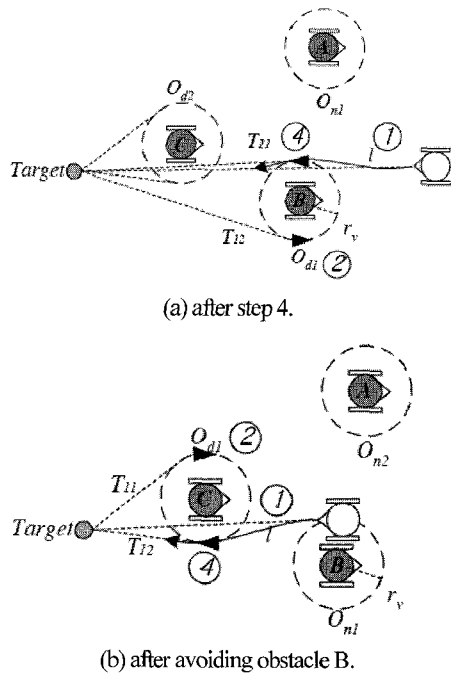


그림 9. 항법의 예.
Fig. 9. Example of navigation.

4). 장애물 B를 완전히 피할 때 까지는 앞과 같은 과정을 반복하다가 로봇이 장애물 B의 접점을 지나는 순간 직선 l 은 더 이상 장애물 B를 지나지 않게 된다. 그러므로, 장애물 B는 O_{b1} 이 되고, 장애물 C와 A가 각각 O_{a1} 과 O_{a2} 이 된다. 다시 앞서의 limit-cycle 항법 과정을 반복하면 그림 9(b)과 같이 장애물을 피해서 목표까지 가게 된다.

III. Limit-cycle 항법의 확장

1. 확장된 항법

그림 10에서 로봇은 limit-cycle 항법을 이용해서 장애물 A를 반시계 방향으로 피한다. 하지만, 다음 순간에 장애물 B를 시계 방향으로 피해야 하므로 로봇은 장애물 A와 B 사이에 끼게 된다. 이와 같은 local minima 문제를 해결하기 위해서 앞에서 설명한 항법 중에 step 4를 다음과 같이 수정해야 한다.

장애물 O_a 의 중심에서 직선 l 까지의 거리 d 를 다음의 식을 이용해서 구한다.

$$d = \frac{aQ_x + bQ_y + c}{\sqrt{a^2 + b^2}} \tag{14}$$

여기서 (Q_x, Q_y) , (G_x, G_y) 과 (R_x, R_y) 는 각각 장애물, 목표, 로

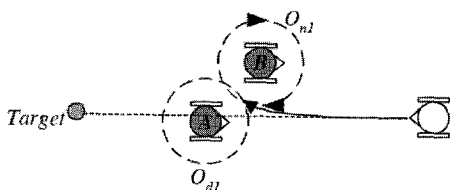


그림 10. Variable 장애물이 겹쳐져서 생긴 local minima.
Fig. 10. Local minima situation.

봇의 위치이다. 만약 2개 이상의 variable 장애물이 겹쳐있다면, 이들을 하나의 장애물로 간주하고, 장애물의 새로운 중심을 다음과 같이 구한다.

$$Q_x = \frac{1}{n} \sum_{k=0}^n Q_{xk}, \quad Q_y = \frac{1}{n} \sum_{k=0}^n Q_{yk} \tag{15}$$

여기서 Q_{xk} 과 Q_{yk} 은 겹쳐진 variable 장애물의 xy -값이다. 계산된 (Q_x, Q_y) 를 이용해서 겹쳐진 장애물 O_d 들의 새로운 d 를 구한다. 그러므로, global 좌표계, Σ_{XYO} 에서 로봇의 위치에서의 움직임여야 하는 방향은 (12)을 이용하여 구할 수 있다. 여기서 x 와 y 는 각각 장애물과 로봇 사이의 상대 위치이다. r 은 이러한 variable 장애물을 하나의 장애물로 가정하여 $r_v + d$ 로 구하고 r_v 는 (13)을 통해서 구할 수 있다. 예를 들어, 그림 11과 같이 세 개의 variable 장애물이 겹쳐있다고 가정하자. 우선, 새로운 (Q_x, Q_y) 를 앞의 step 4와 같이 구하고, 이를 (14)에 대입하면, d 를 구할 수 있다. 장애물 B가 로봇에서부터 가장 가까우므로, 이것에 대해서 r , r_v 와 d , 그리고, (12)를 사용하면 그림 11과 같이 장애물을 피해 움직일 수 있다.

2. 로봇 축구에의 적용

로봇 축구는 환경이 능동적으로 계속 변화하는 전형적인 다개체 시스템의 예이며, 같은 팀과는 협동을 해야 하고 상대팀과는 경쟁을 해야 한다. 뿐만 아니라, 제어 방법의 효율을 골득실로 평가할 수 있다. 로봇 축구에서 당면한 문제는 비전 시스템을 통한 정확한 위치 추적, 위치를 이용한 항법 계획, 계획된 경로 추종 등의 세 가지로 크게 나눌 수 있다. 이 중에 항법 계획은 로봇 축구팀의 전력을 평가하는 중요한 잣대가 된다.

항법의 관점에서 볼 때, 로봇 축구는 이동 장애물과 이동 목표물이 있는 능동적인 환경에서의 항법이다.

지금까지 limit-cycle 항법을 장애물을 피해서 목표로 가는 것에 주안점을 두고 제안했다. 하지만, 로봇 축구에서는 상대 로봇을 피해서 공을 상대 진영으로 차야 하므로, 그림 12(a)과 같이 로봇이 공에 도달할 때 상대 진영 쪽으로 향해서 공의 뒤쪽에 있어야 한다.

로봇 축구에 limit-cycle 항법을 적용하기 위해서는 앞 절의 step 4에 다음의 규칙을 더해야 한다.

공의 양 옆에, 상대 골대에서 공으로의 연장선상에 접하도록 가상의 variable 장애물을 가정한다.

$$\begin{cases} d = -1 & \text{if a obstacle is on the left of the ball,} \\ d = 1 & \text{if a obstacle is on the right of the ball.} \end{cases} \tag{16}$$

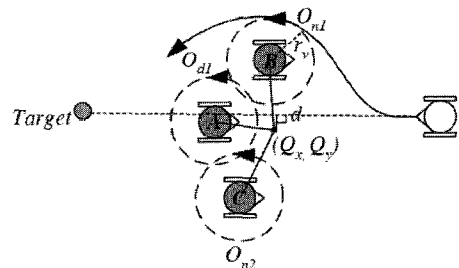


그림 11. 확장된 항법.
Fig. 11. Extended navigation

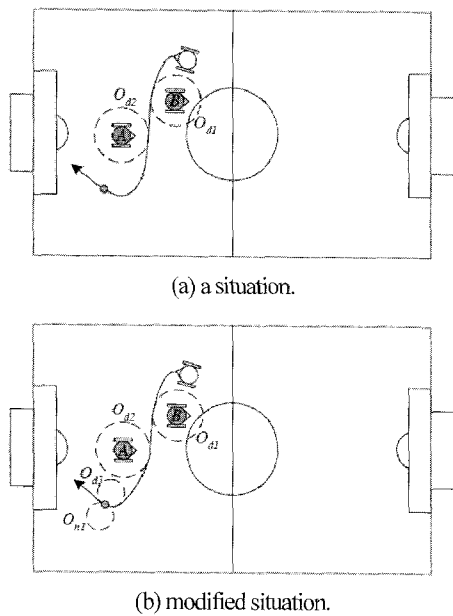


그림 12. 로봇축구 예제.
Fig. 12. Robot soccer example

예를 들어, 그림 12(a)와 같은 로봇축구 상황은 공의 양 옆에 두 개의 가상의 variable 장애물을 가정해서 그림 12(b)과 같이 수정해야 한다. 이 때, 장애물의 반지름은 최소의 variable 반지름인 r_{vmin} 로 한다. 로봇 축구에서 로봇은 150cm/s의 빠른 속도로 움직이므로, r_{vmin} 를 정하는데 있어서 로봇의 중심 속도와 미끄러짐을 고려하지 않으면 로봇이 제대로 원을 그리며 움직이지 못한다. 다음의 식은 로봇이 미끄러지지 않고 움직일 수 있는 최소의 반지름을 나타낸다.

$$r_{vmin} \leq \frac{mv^2c}{F_c} \tag{17}$$

여기서 m 은 로봇의 질량이고, v_c 는 로봇의 중심 속도이며, F_c 는 로봇의 마찰력이다. 위의 식에서 r_{vmin} 는 로봇의 중심 속도와 마찰력에 의해 제한되는 것을 알 수 있다. 마찰력은 고정되어 있고 측정이 가능하므로 최소의 회전 반경, r_{vmin} 는 로봇의 중심 속도에 의해 정해질 수 있다.

그림 12(b)에서 O_m 는 공의 왼쪽에 있는 가상의 장애물이므로, 추가된 규칙에 의해 $d = -1$ 이다. 마찬가지로 O_{ob} 의 $d = 1$ 이다. 그러므로, 로봇은 그림 12(b)와 같이 limit-cycle 항법을 사용해서 움직인다. 만약에 앞서 설명한 규칙의 수정이 없다면, 그림 12(b)의 장애물 A가 없는 경우 d 가 음수가 되어 로봇이 공을 우리편 쪽으로 차게 된다.

지금까지 설명한 limit-cycle 항법은 로봇의 현재 위치에서 로봇의 모든 궤적을 계산하는 것이 아니고, 로봇과 목표의 상대 위치, 장애물과의 상대 위치를 이용해서 로봇의 다음 위치를 결정한다. 그러므로, 이 방법은 변화하는 환경에 대해서 유연한 특성을 가진다. 기존의 potential field 법은 로봇이 장애물을 피할 때 어느 방향으로 어느 정도로 피할지를 결정할 수 없으므로, 로봇축구에 바로 적용할 수 없다. Slack[8]은 장애물의 주위에 회전하는 potential field를 더해서 장애물을 피했지만, 로봇이 장애물을 어느 정도로 떨어져서 피할지를

정해줄 수 없다. 지금까지 설명한 limit-cycle 항법은 장애물을 어느 방향으로 피할지와 어느 정도로 피할지를 정할 수 있으므로 환경의 변화가 심한 경우에 적합하며, 로봇축구에도 사용 가능하다.

IV. 모의 실험과 실험 결과

1. 모의 실험

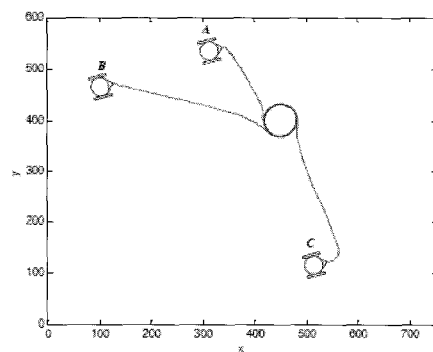
제안된 limit-cycle 항법은 펜티엄 III 800MHz의 PC에서 모의 실험을 했다. 우선 그림 13(a)와 같이, 로봇은 세 개의 각각 다른 위치에서 출발하여 (450, 400)에 있는 반지름 30cm의 원으로 수렴하도록 한다. (530, 120)에서 출발한 로봇은 초기 각도 때문에 잠시 반대 방향으로 움직이다가 원을 향하여 움직이는 것을 알 수 있다.

앞서 설명한 것과 같이, limit-cycle 항법은 회전하는 반경을 쉽게 조절할 수 있다. 그림 13(b)에서, 로봇은 반지름이 각각 30cm, 60cm, 90cm인 원으로 움직인다. 여기서 특이한 현상이 발견되는데, 반지름 90cm의 원으로 수렴하는 로봇이 잠시동안 overshoot과 같이 출렁거리는 것을 볼 수 있다. 이 현상을 설명하기 위해 점근 계수, α 를 정의하고, limit-cycle 항법에서 사용한 식을 수정한다.

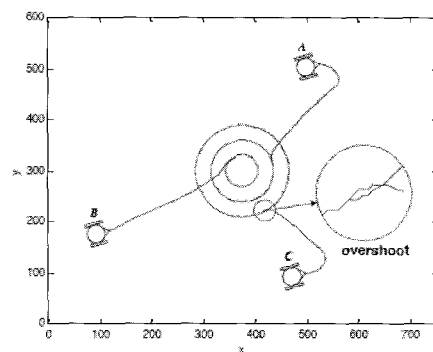
$$\begin{aligned} \dot{x} &= y + \alpha x (1 - x^2 - y^2) \\ \dot{y} &= -x + \alpha y (1 - x^2 - y^2) \end{aligned} \tag{18}$$

여기서 $\alpha > 0$ 이다. 점근 계수, α 를 이해하기 위해서 원래의 2차 비선형 함수를 다음의 극좌표계로 나타낸다.

$$x = r \cos \theta, \quad y = r \sin \theta \tag{19}$$



(a) various starting points.



(b) various radius.

그림 13. 모의실험(1).
Fig. 13. Simulation(1)

식 (19)과 x, y 의 미분값을 이용해서 다음의 식을 유도할 수 있다.

$$\begin{aligned} \dot{x} &= \dot{r} \cos \theta - r \sin \theta \cdot \dot{\theta} \\ &= r \cos \theta + \alpha r \sin \theta (1 - r^2 \sin^2 \theta - r^2 \cos^2 \theta) \end{aligned} \quad (20)$$

$$\begin{aligned} \dot{y} &= \dot{r} \sin \theta + r \cos \theta \cdot \dot{\theta} \\ &= r \sin \theta + \alpha r \cos \theta (1 - r^2 \sin^2 \theta - r^2 \cos^2 \theta) \end{aligned} \quad (21)$$

그리고, 다음의 식을 만족해야 한다.

$$\dot{r} = \alpha r (1 - r^2), \quad \dot{\theta} = -1, \quad (22)$$

그러므로, (23)으로 점근 계수, α 가 거리의 변화를 조절할 수 있다는 것을 알 수 있다. 그림 14는 반지름이 90cm인 원에 대해 네 개의 다른 점근 계수를 대입해서 실험한 결과이다. 그림에서 $\alpha=2$ 일때, 궤적이 부드럽고, 적절하다.

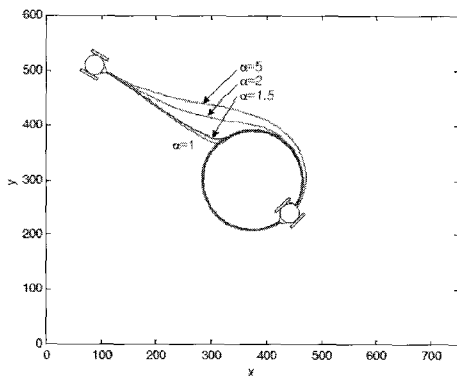
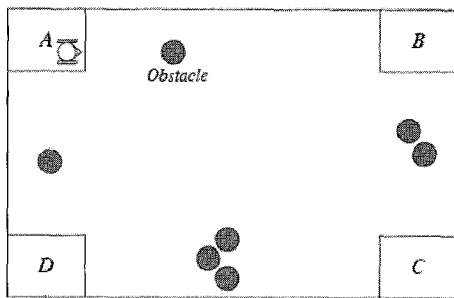
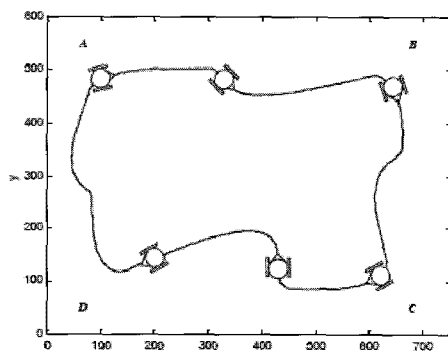


그림 14. 모의실험 (2)
Fig. 14. Simulation (2).



(a) navigation though obstacles.



(b) simulation result.

그림 15. 모의실험 (3)
Fig. 15. Simulation (3).

다음은 limit-cycle 항법을 이용해서 로봇이 여러 장애물이 있는 지역을 어떻게 이동하는지를 실험한다. 그림 15(a)에서, 로봇은 A 지역에서 출발해서 B → C → D 지역으로 여러 장애물을 피하면서 이동한다. 이 때 그림에서와 같이 가까이 있는 두 개내지 세 개의 장애물은 하나의 큰 장애물로 간주한다.

그림 15(b)는 또 다른 경우의 모의 실험 결과를 보여준다. 로봇은 (150, 500)에서 출발하여 (350, 500)에서 처음 장애물을 만나고, 이를 반시계 방향으로 피한다. B 지역을 지난 후, 로봇은 두 개의 variable 장애물이 겹쳐져서 하나의 큰 장애물로 간주되는 장애물들을 만난다. 이것을 확장된 limit-cycle 항법으로 피한 후, C 지역을 지난다. 로봇은 겹쳐져 있는 세 개의 variable 장애물을 만나게 되는데, 이 경우 로봇은 가장 아래에 있는 장애물을 반시계 방향으로 피하려고 하다가 위쪽의 다른 장애물을 만나게 된다.

그림 16은 이 세 장애물을 피하는 행동을 확대한 것이다. Limit-cycle 항법의 step 4에서 겹쳐진 variable 장애물들의 d 를

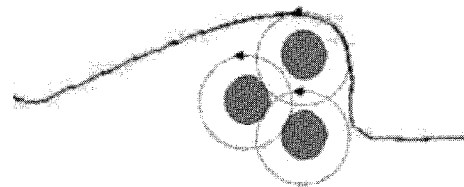
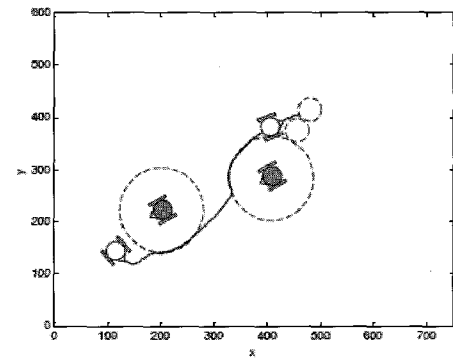
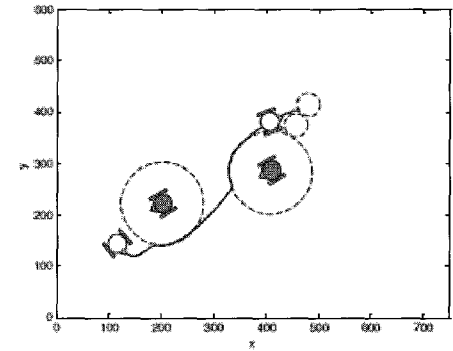


그림 16. 가까이 있는 세 개의 장애물 (확대된 그림).
Fig. 16. Zoomed image.



(a) navigation for a soccer robot.



(b) simulation result.

그림 17. 모의실험 (4)
Fig. 17. Simulation (4).

구한 후, 로봇은 세 장애물들을 반시계 방향으로 피한다.

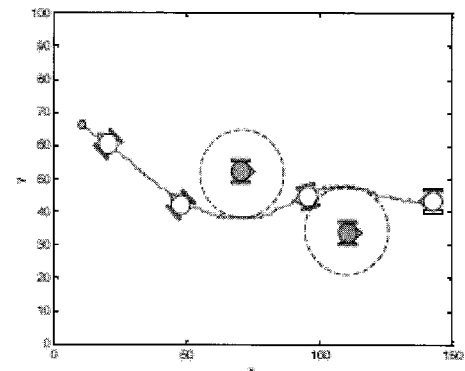
지금까지는 장애물만 있는 상황에서 limit-cycle 항법을 적용했다. 이번에는 그림 17(a)과 같은 로봇 축구 상황을 보자. 왼쪽 구석의 로봇은 두 대의 상대 로봇을 피해서 공을 상대 골대 쪽으로 차야 한다. 그림 17(b)에서 로봇은 (150, 150)에서 출발한다. 로봇은 처음에 아래를 향해 있었으므로 공 쪽으로 회전한 후에 공으로 향하다가 첫번째 장애물을 만난다. 반시계 방향으로 첫번째 장애물을 피한 후, 두번째 장애물을 시계 방향으로 피한다. 공의 오른쪽에 가상의 variable 장애물을 가정 한 후에 시계방향의 궤적을 따라 가면, 로봇은 공을 상대 진영으로 차게 된다.

이번 절에서는 limit-cycle 항법을 여러 상황에 대해서 모의 실험했다. 로봇이 여러 위치에서 출발해서 일정한 원으로 움직이게 하거나, 여러 반지름의 원으로 움직이게 했다. 모의 실험을 하는 동안, 궤적의 변화율을 조절할 수 있는 접근 계수를 정의하고 분석했다. 여러 개의 variable 장애물이 겹쳐있을 때에, 로봇이 확장된 limit-cycle 항법을 사용해서 장애물들을 회피했으며, 두 개의 가상의 variable 장애물을 가정하여 로봇 축구에 성공적으로 적용했다.

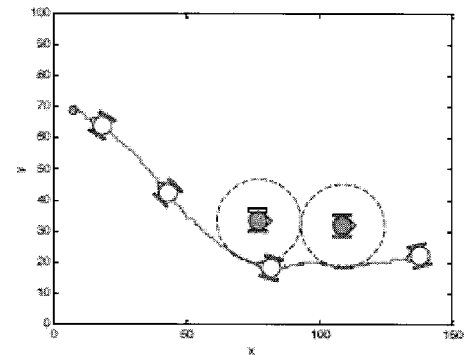
2. 실험 결과

그림 18에서 제안한 limit-cycle 항법을 로봇 축구 시스템인 SOTY IV를 가지고 실험한 결과를 보인다. 각각의 경우에 로봇은 회색의 장애물을 피해서 공으로 움직여야 한다. 그림에서 직선은 로봇의 궤적이다. 그림 18(a)의 경우, 로봇은 처음에 (140, 45)에서 출발한다. 첫번째 장애물을 만날 때, 식 (8)을 통해 구한 d 는 음수였다. 로봇과 장애물의 반지름을 4cm로 하고, 안전 여유값, δ 를 2cm로 하면, 식 (12)는 로봇으로 하여금 반시계 방향으로 돌아서 피하도록 한다. 첫번째 장애물을 피하고 난 후, 이 장애물을 non-disturbing 장애물로 간주하게 되고, 로봇은 공으로 움직이게 된다. 두번째 장애물도 limit-cycle 항법을 이용해서 피하게 된다. 그림 18(b)는 로봇이 두 개의 장애물을 식 (8)과 식 (12)를 이용해서 피하는 결과를 보여주고, 그림 18(c)의 경우엔, 두 개의 variable 장애물이 겹쳐져 있기 때문에 식 (14)와 식 (15)를 이용해서 d 를 구하고 limit-cycle 항법을 이용해서 장애물을 피한다. 만약 로봇이 첫번째 장애물을 피한 다음에 이를 무시한 채 다음 장애물을 limit-cycle 항법을 이용해서 반시계 방향으로 피한다면 앞 절의 그림 10과 같은 local minima에 빠져서 로봇이 움직이지 못하는 상황이 된다. 그러나, 식 (14)와 식 (15)를 이용하면, 로봇은 이들 겹쳐진 variable 장애물을 하나로 간주해서 d 를 구하고 피한다면 두번째 장애물을 시계방향으로 피하게 된다. 그림 18(d)는 (140, 65)에서 출발한 로봇이 두 개의 장애물을 피해서 목표로 가는 것이다.

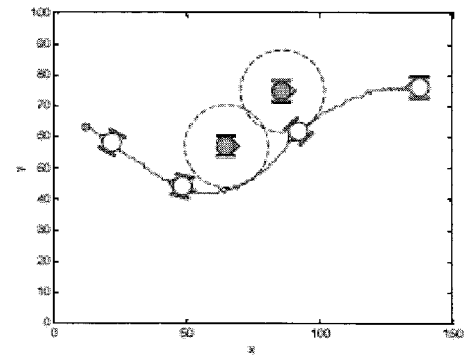
실제 실험을 통해서, limit-cycle의 장점을 보였다. Potential field법의 local minima와 원하지 않는 장애물의 영향을 회피하는 원의 방향과 크기를 조절함으로써 해서 해결했다. 가상의 variable 장애물을 가정하여, 목표점에 도달했을 때 원하는 방향으로 향하도록 했다. 이것은 로봇 축구에서 공을 상대 골대로 차는 경우에 사용되었다. 이 절에서의 실험 결과는 limit-cycle 항법이 reactive 방법처럼 유연하고, deliberative 방법처럼 목표지향적임을 나타내며, 로봇 축구와 같이 변화하는



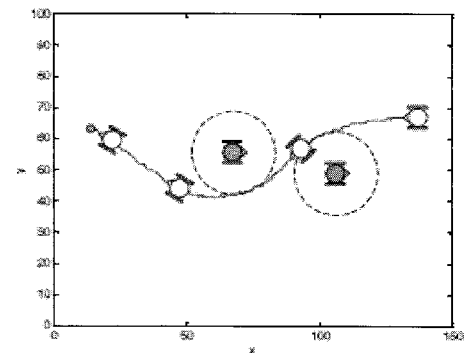
(a) situation (1).



(b) situation (2).



(c) situation (3).



(d) situation (4).

그림 18. 실험 결과.

Fig. 18. Experimental results.

환경의 경우에도 잘 적용됨을 보였다.

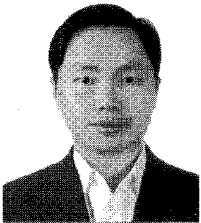
V. 결론

본 논문에서는 2차 비선형 함수의 limit-cycle 특성을 이용해서 로봇 축구와 같은 환경이 빠르게 변화하는 경우에 대해

서도 적용 가능한 새로운 방법을 제안했다. 로봇이 장애물을 회피하는 방향뿐만 아니라 거리를 조절함으로써 해서 어떠한 경우에도 원하는 목표로 갈 수 있었다. 제안한 방법의 우수성을 보이기 위해 컴퓨터를 통한 모의 실험과 실제 로봇을 이용한 실험 결과를 보였다.

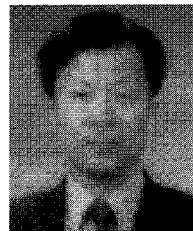
참고문헌

- [1] R. C. Arkin, "Motor Schema-Based Mobile Robot Navigation," *The International Journal of Robotics Research*, vol. 8, no. 4, pp. 92-112, 1989.
- [2] R. C. Arkin, "Integrating Behavioral, Perceptual and World Knowledge in Reactive Navigation," *Robotics and Autonomous Systems*, vol. 6, pp. 105-122, 1990.
- [3] R. A. Brooks, "A Robust Layered Control System for a Mobile Robot," *IEEE Journal on Robotics and Automation*, vol. RA-2, no. 1, 1986.
- [4] D. H. Kim, Y. J. Kim, K. C. Kim, J. H. Kim and P. Vadakkepat, "Vector Field Based Path Planning and Petri-net Based Role Selection Mechanism with Q-learning for the Soccer Robot," *Intelligent Automation and Soft Computing*, vol. 6, no. 1, pp. 75-88, 2000.
- [5] Y. J. Kim, J. H. Kim and D. S. Kwon, "Evolutionary Programming-based Univector Field Navigation Method for Fast Mobile Robots," *IEEE Trans. on Systems Man, and Cybernetics*, vol. 31, no. 3, pp. 450-458, 2001.
- [6] H. S. Sim, M. J. Jung, H. S. Kim, J. H. Kim and P. Vadakkepat, "A Hybrid Control Structure for Vision Based Soccer Robot System," *Intelligent Automation and Soft Computing*, *Intelligent Automation and Soft Computing*, vol. 6, no. 1, pp. 89-101, 2000.
- [7] M. S. Lee, M. J. Jung and J. H. Kim, "Evolutionary Programming-based Fuzzy Logic Path Planner and Follower for Mobile Robots," *Congress on Evolutionary Computation*, vol. 1, pp. 139-144, San Diego, USA, July 2000.
- [8] M. G. Slack, "Navigation Template: Mediating Qualitative Guidance and Quantitative Control in Mobile Robots," *IEEE Trans. on Systems, Man and Cybernetics*, vol. 23, no. 2, pp. 452-466, 1993.
- [9] Y. Koren and J. Borenstein, "Potential Field Methods and their Inherent Limitations for Mobile Robot Navigation," *Proceedings of the IEEE Conference on Robotics and Automation*, vol. 2, pp. 1398-1404, 1991.
- [10] E. Gat, "Navigation Templates: Enhancements, Extensions and Experiments," *IEEE International Conference on Robotics and Automation*, vol.1, pp. 541-547, 1993.
- [11] H. K. Khalil, *Nonlinear Systems*, 2nd ed., Prentice Hall, chap. 7, pp. 289-312, 1996.
- [12] E. Rimon and D. Koditschek, "Exact navigation using artificial potential functions," *IEEE Trans. on Robotics and Automation*, vol. 8, no. 5, pp. 501-518, 1992



김 동 한

1973년 3월 27일생. 1995년 KAIST 전기 및 전자공학과(공학사). 1998년 동대학원(공학석사). 2003년 동대학원(공학박사). 2003년~2004년 일리노이 주립 대학 Post Doc. 2004년~2007년 삼성전자 무선사업부 책임연구원. 2007년~현재 경희 대학교 전자정보대학 교수. 관심분야는 로봇제어, 다개 체 지능제어.



류 근 호

1971년 1월 30일생. 1994년 KAIST 기계공학과(공학사). 1996년 동 대학원 (공학석사). 2001년 동 대학원 (공학박사). 1998~2003년 미래산업, 2003~2005년 파인디앤씨, 2005년~ 현재 호서대학교 로봇공학 과 교수. 관심분야는 모터의 잔류 진동제어, 온도제어, 생체신호처리, 반도체/디스플레이 장비 개발.