

거리계를 이용한 이동로봇 'KMR-2'의  
경로주행제어에 관한 연구

조 형석\*, °이대엽\*\*, 이 종원\*\*

\*한국과학기술원 생산공학과 \*\*한국과학기술원 CAD/CAM연구실

Path control of a Mobile Robot 'KMR-2'  
using odometer system

HyungSuck Cho, DaeYoup Lee, ChongWon Lee

Dept. of Production Engineering KAIST, CAD/CAM Lab. KAIST

ABSTRACT

Free-path-type guidance system does not need a hardwired path in the environment so that it gives a mobile robot a flexible path. In this study to achieve the free-path-type guidance system for a mobile robot which is steered by the differential steering of both drive forewheels, position recognition systems are constructed using odometer system as an internal position sensor. Two odometer systems, a auxiliary wheel odometer and a 2-encoder odometer system are constructed and path following algorithms using these odometer systems are designed and experimented. PID control type is adopted in the path following algorithms.

1. 서론

궁극적인 산업용 로봇에는 외부의 환경을 인식하고 판단해서 독자적으로 행동할 수 있는 자율적인 이동로봇 (autonomous mobile robot)의 기능이 필요하게 된다. 또한 무인 자동화 공장에서의 작업물 운반과 원자력 또는 해저 작업과 같은 위험한 작업들에서는 필수적으로 이동기능 또는 주행기능이 필요하게 된다. 따라서 이러한 기능을 갖는 이동로봇 또는 무인운반차에 대한 기본적인 연구로서 주행과 위치인식에 대한 연구를 하였다. 본 연구에서는 주행 경로에 유연성을 갖도록 하고 주행면에 유도케이블이나 위치마크등의 고정된 경로를 설치할 필요가 없는 자유 경로방식의 무인운반차에 대한 연구를 하였다. 자유경로방식의 주행을 위해서는 신뢰성있는 위치인식 시스템이 필요하게되고, 이를 위해서 간단한 구성으로써 주행중의 위치 인식이 가능한 거리계 시스템을 채택하였다. 내부 센서로서 거리계를 이용한 방식은 간단한 구성으로 주행중의 위치 인식이 가능한 장점이 있으나 필연적으로 주행거리의 증가에 따른

누적오차가 발생하게 된다. 따라서 내부 센서를 이용한 시스템의 주행 한계점을 규명하고 일정거리의 주행후에는 누적오차를 보정해 주는 시스템을 구성하면 실제의 응용에도 가능할 것이다.

2. 이동로봇 시스템의 구성

본 연구의 목적인 내부센서를 이용한 경로 주행을 실현하기 위해서는 위치인식 알고리즘과 주행 알고리즘이 필요하게 된다. 다음에서는 위치 인식의 기본 원리및 구성과 주행 제어부의 구성, 그리고 전체 시스템의 구성을 살펴보도록 한다.

2.1. 위치 인식의 기본 원리

이동 로봇이 자신의 현재 위치및 방향을 인식하기 위해서는 여러가지 방법이 사용되고 있으나 본 연구에서는 자유경로 방식의 유도주행을 실현하기 위하여 거리계(odometer), 즉 지나온 궤적을 계산하여 위치를 인식하는 장치를 구성하였다. 거리계를 이용한 위치 인식의 기본 원리는 다음과 같다. 주어진 경로로의 유도 주행을 위해서는 이동로봇의 주행중의 위치와 방위각(azimuth angle)을 알아야 한다. 이러한 정보로부터 기준경로에 대한 거리편차(lateral deviation)와 방향 편차(directional deviation)를 계산할 수 있게 된다. 단위 시간 동안의 이동거리가 작을 경우에는 그림1에 나타낸 바와 같이 다음의 관계를 구할 수 있다.

$$\begin{aligned} X_n &= X_0 + \sum_{i=1}^n \Delta X_i \\ &= X_0 + \sum_{i=1}^n (2\Delta l_i / \Delta \theta_i) \cdot \sin(\Delta \theta_i / 2) \cdot \sin(\theta_{i-1} + \Delta \theta_i / 2) \\ Y_n &= Y_0 + \sum_{i=1}^n Y_i \\ &= Y_0 + \sum_{i=1}^n (2\Delta l_i / \Delta \theta_i) \cdot \sin(\Delta \theta_i / 2) \cdot \cos(\theta_{i-1} + \Delta \theta_i / 2) \\ \theta_n &= \theta_0 + \sum_{i=1}^n \Delta \theta_i \end{aligned}$$

여기서 초기위치는 P0(X0,Y0)로 주행중의 위치는 Pm(Xm,Ym)으로 나타내었다.

거리편차의 정의는, 기준 경로가 직선으로 구성된 경우는 그림2.(a)와 같다. 기준 경로가 곡선인 경우에는 그림2.(b)에 나타내었고, 이동 로봇의 주행중의 위치 Pm(Xm,Ym)에서 기준 경로에 대한 최단거리의 점 Pr(Xr,Yr)의 결정은 다음과 같이 하였다. Pm과 Pr을 잇는 직선은 점 Pr에서 경로에 대한 접선과 직교하게 되므로

$$\frac{f(Xr) - Ym}{Xr - Xm} \cdot f'(Xr) = -1$$

이 성립하고, 정리하면

$$(f(Xr)-Ym) \cdot f'(Xr) + (Xr-Xm) = 0$$

이 된다. 여기서 f(x)는 곡선 기준 경로를 나타내는 함수가 되고, bisection method에 의해서 잇속의 해 Xr을 구하도록 하였다.

## 2.2. 위치 인식부의 구성

이동 로봇의 주행중의 위치와 방위각을 계산하기 위해서는 단위 시간 동안의 이동거리 $\Delta l$ 과 회전각 $\Delta \theta$ 가 필요하게 된다. 이러한 값들을 측정하기 위해서 본 연구에서는 보조바퀴를 이용한 방식과 좌.우 동륜에 2개의 encoder를 각각 부착한 거리계 시스템을 각각 구성하여 비교하였다. 그림3에 나타난 보조바퀴에는 직진 성향을 주기 위해서 caster angle이 주어져 있기 때문에 이로 인해 생기는 측정오차를 보정해 주어야 한다. 관계식을 유도하기 위한 모델을 그림4에 나타내었다. 그림으로부터 보조바퀴의 회전각과 이동 로봇의 회전각의 관계를 구하면

$$\Delta \theta_m = \frac{\Delta l}{n} \sqrt{\frac{1 - \cos \theta_{pc}}{1 + b^2 - 2b \cos \theta_{pc}}}, \text{ 단 } b = \frac{C}{n}$$

가 된다.

여기서

$\Delta \theta_m$ 은 단위 시간동안의 이동 로봇의 회전각이고

$\theta_{pc}$ 는 보조바퀴의 회전각,

$\Delta l$ 은 단위시간 동안의 보조바퀴의 이동거리,

$h$ 는 좌우 동륜 중심으로부터 보조바퀴의 회전 중심까지의 거리를

그리고  $C$ 는 caster angle에 의한 offset을 나타낸다. 또한 보조바퀴의 이동거리와 이동로봇의 이동거리의 관계식은

$$\Delta l_m = \frac{b - \cos \theta_{pc}}{1 - b \cos \theta_{pc}} \cdot \Delta l_a, \text{ 단 } b = \frac{c}{h}$$

와 같이 된다.

이상의 결과로부터 보조바퀴에 부착된 potentiometer와 encoder로 측정된 신호를 이용하여 주행중의 위치 인식을 할 수 있음을 알 수 있다.

2개의 encoder를 이용하는 방식은 그림5 다음의 관계로부터 단위시간 동안의 이동 로봇의 회전각과 이동거리를 구할 수 있게 됨이 알려져 있다(1)

$$\Delta \theta_m = \frac{\Delta l_l - \Delta l_r}{E}$$

$$\Delta l_m = \frac{\Delta l_l + \Delta l_r}{2}$$

여기서  $\Delta l_l$ 과  $\Delta l_r$ 은 단위 시간동안의 좌우 동륜의 이동거리를,  $E$ 는 좌우 동륜간의 평균거리를 나타낸다.

## 2.3. 주행부의 구성

기준 경로로의 유도 주행을 위해서는 앞에서 살펴본 거리계 시스템에 의해서 인식된 주행 위치로부터 기준 경로에 대한 편차를 계산하게 되고 이를 이용해서 경로 주행을 실현하게 된다. 본 연구에서는 좌우 동륜의 속도차를 이용한 주행 방식을 사용하였으므로, 주행 제어는 기준 경로에 대한 거리 편차가 없어지도록 좌.우 동륜의 회전 속도를 제어하는 것이 된다. 주행 제어 기준점에서의 주행 속도가 일정하게 되기 위해서는 좌.우 동륜에 대한 속도 명령은 다음과 같이 하도록 하였다.

$$V_{left} = V_{command} - V_{error}$$

$$V_{right} = V_{command} + V_{error}$$

여기서  $V_{error}$ 의 control type은 다음과 같이 결정하였다.  $V_{error} = k \cdot x$ 일 경우에는, 즉 거리편차에 대한 비례제어를 할 경우에는 steady state의 거리편차를 제거할 수 없게 된다. 따라서 거리편차에 대한 적분 제어를 추가함으로써 steady state에서의 거리편차를 제거하도록 한다. 그러나 거리편차에 대한 PI control만을 적용할 경우에는  $90^\circ$ 보다 큰 초기 방향 편차를 제거할 수 없게 된다. 또한 주행궤적에 적절한 damping을 주기 위해서도 방향 편차에 대한 제어가 필요하게 된다. 그림6에서

$$dEx = dl \cdot \sin E\theta$$

이므로

$$\frac{dEx}{dl} \propto E\theta \text{ 단, } -90^\circ < E\theta < 90^\circ$$

위의 관계를 갖게된다. 즉 방향 편차에 대한 제어는 단위 시간 동안의 이동거리  $l$ 이 일정하다고 가정하면 거리 편차  $x$ 의 변화량에 대한 제어와 같은 효과를 갖게 된다. 따라서 이상의 결과로부터  $V_{error}$ 의 control law를 다음과 같이 결정하였다. 즉 거리 편차에 대한 PID control을 하였다.

$$V_{error} = K_p \cdot Ex_n + K_i \sum_{i=1}^n Ex_i + K_d \cdot \Delta Ex_n$$

$$\text{단, } \Delta Ex_n = Ex_n - Ex_{n-1}$$

주행부의 구성은 그림7에 나타내었다. (a)의 경우는 위치 인식을 위한 거리계로써, 보조바퀴를 이용한 경우이고, (b)의 경우는 2개의 encoder를 이용한 경우이다. 그림8에는 본 연구에 사용된 이동로봇의 외관을 나타내었다.

### 3. Simulation 및 주행 실험

주어진 경로와 주행속도에 따른 최적의 주행을 위해서는 주행 제어부의 적절한 PID gain을 결정하는 것이 필요하다. 본 연구에서는 주행 제어부의 simulation을 통해서 PID gain을 구하고 이를 실험을 통해서 확인해보도록 하였다.

#### 3.1. Simulation 및 PID gain의 결정

그림9에는 주행제어 algorithm을 나타내었다. 그림에서 이론적으로 결정하기 곤란한 parameter는 identification을 통해서 구하였다. 즉 motor에 step input을 가한후 이의 response와 주어진 전달함수를 비교함으로써 parameter를 근사적으로 구하였다.

또한 simulation에서 disturbance항은, modelling으로 부터 이동 로봇의 주행 속도가 저속일 경우에는 좌,우 동륜에 작용하는 외력의 대부분은 동륜의 회전 방향에 대한 coulomb 마찰력이라는 것을 알 수 있으므로

$$N(s) = (\mu \cdot Fdz \cdot R_w \cdot \frac{1}{50}) \cdot \frac{1}{s} = n(s)$$

이 된다. 여기서 Fdz는 동륜에 작용하는 normal force 이고  $\mu$ 는 coulomb 마찰계수로써 여기서는 고무와 인조 대리석의 마찰에 해당된다. 그리고 R<sub>w</sub>는 좌우동륜의 반지름을, 50은 harmonic drive 감속기의 기어비를 나타내고, n(s)는 random anaplitude signal로서 정량적으로 고려할 수 없는 미소한 외력에 의한 영향을 보기위한 것이다.

PID gain을 결정하기 위한 index로서는 ITAE(Integral-of-time-multiplied absolute-error criterion)을 고려하였다. 즉 다음과 같은 performance index를 최소화시키는 PID gain을 구하였다.

$$\int_0^{\infty} t \cdot |e(t)| dt$$

각 PID gain의 변화에 따른 simulation 결과와 주행 실험결과를 그림10에 나타내었다. 주행 실험방법은 2m 주행후 우측으로 90° 회전하고, 다시 2m 주행하는 것이다.

#### 3.2. 경로주행실험

여러가지 오차, 즉 바퀴지름의 부정확, 좌우동륜간의 평균거리의 변화, 여러가지 변수들의 실제값과 측정값의 차이, 그리고 바퀴와 주행면 사이의 미끄럼에 의해서 실제 주행 궤적과 주어진 기준 경로에는 차이가 발생하게 된다. 따라서 이러한 편차를 실험을

통해서 보상 해 주기 위해서 그림11에서와 같은 실험 계수를 도입하였다. 각 실험 계수의 역할을 간단히 살펴보면 EC1은 직선 주행시의 거리편차를 보정해 주기 위한 것이고, EC2는 주행거리의 오차를 보정해 주기위한 실험계수이고, EC3는 oversteer 또는 understeer 현상으로 인해 발생한 실제 주행 궤적에서의 편차를 보정해 주기 위한 계수이다. 그리고 EC4는 방향전환의 기준으로서 방향전환시 발생하는 기준 경로로 부터의 이탈을 줄여 주는 역할을 한다. 이 계수의 역할을 좀더 자세히 살펴보면, 한 구간에서의 최종 주행점이 다음 구간의 출발점이 되기 때문에 다음 구간으로의 방향 전환시에는 이전 구간에서의 최종 편차를 고려하여야 한다. 그림12에서 경로 S<sub>n</sub>에서 경로 S<sub>n+1</sub>로 주행을 할때, 방향전환의 기준점 P<sub>c</sub>에 도달했을 때의 위치가 다음 경로 S<sub>n+1</sub>의 초기점이 된다. 따라서

$$X0)_{n+1} = EC4 \cdot \sin \theta_p + \epsilon_x) \cdot n \cdot \cos \theta_p$$

$$Y0)_{n+1} = -EC4 \cdot \cos \theta_p - \epsilon_x) \cdot n \cdot \sin \theta_p$$

$$\theta0)_{n+1} = -\theta_p + \epsilon_\theta) \cdot n$$

여기서  $\theta_p$ 는 두 경로의 교차각을 나타낸다.

직선 경로에 대한 실험결과는 그림13에, 곡선경로에 대한 실험결과는 그림14에 나타내었다.

## 4. 결 론

4.1. 이동 로봇에서 주행중의 위치를 인식하기 위한 거리계 시스템으로서, 보조바퀴를 이용한 방식과 2개의 encoder를 이용하는 방식을 구성하였고, 각각의 시스템을 이용한 경로 주행 알고리즘을 구성하고, 실험을 하였다.

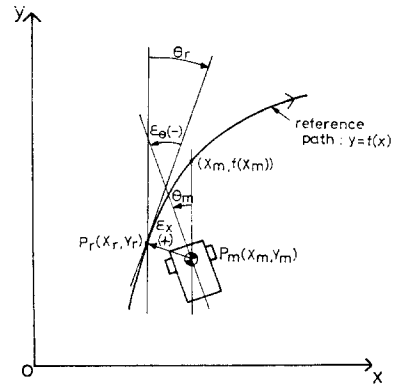
4.2. PID gain을 결정하기 위한 index로서, 주행 경로가 기준 경로에 대하여 critical 또는 over-damping의 형태를 갖도록 하였고, simulation과 실험을 통해서 최적의 주행 경로를 갖도록 Pid gain을 결정하였다.

4.3. 이동 로봇의 주행 특성을 파악하기 위한 실험을 하였고 여러가지 실험계수를 도입하여 최적의 주행 경로로 직행할 수 있도록 보정을 하였다. 보조 바퀴를 이용한 거리계 시스템에서는 oversteer특성이 있었고, 2개의 encoder를 이용한 system에서는 neutral steering이 가능하였다.

4.4. 연장 8m 정도의 주행 경로에 대하여 직선으로 구성된 경로 주행의 경우에는 25cm 정도의 최종거리 편차내에서 주행이 가능하였고, 곡선으로된 경로주행에서는 거리편차 20cm 이내에서 주행이 가능하였다. 따라서 일정 거리의 주행은 내부 센서만에 의해서 주행이 가능함을 알 수 있다.

참고문헌

1. M.Julliere, L.Morce, H.Parrichot, "A Guidance system for a vehicle which has to follow a memorized path", Proceeding of the 2nd Int'l conferences on Automated Guided Vehicle systems, 1983.
2. J.Iijima et al., "Elementary function of a self-contained robot 'YAMABICO 3.1.", 11th ISIR, 1981
3. J.Y.Wong, "Theory of ground vehicles", John wiley & sons, 1978
4. Samuel D. Conte, "Numerical Analysis", McGraw-Hill, 1980.



(b) case of curvilinear reference path

Fig 2. Definition of the lateral and the directional deviations

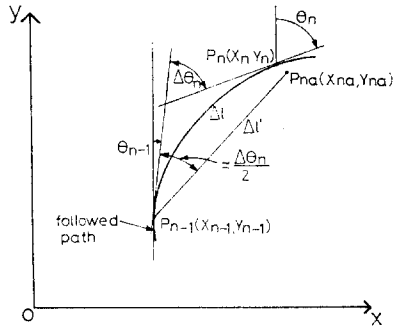


Fig 1. Definition of calculating the position and the azimuth angle

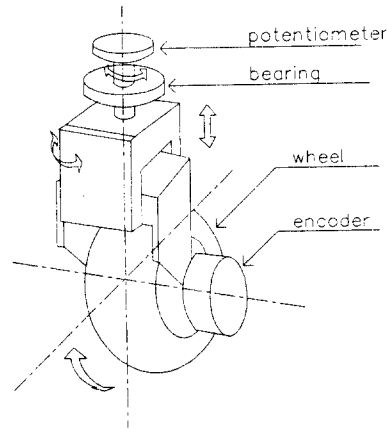
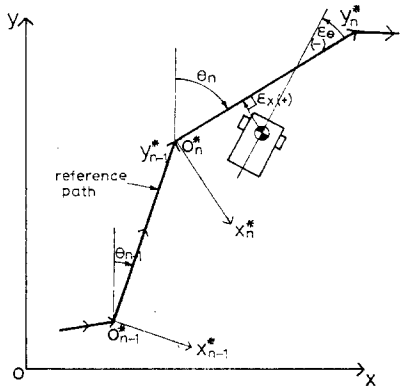


Fig 3. Structure of the auxiliary wheel



(a) case of straight reference path

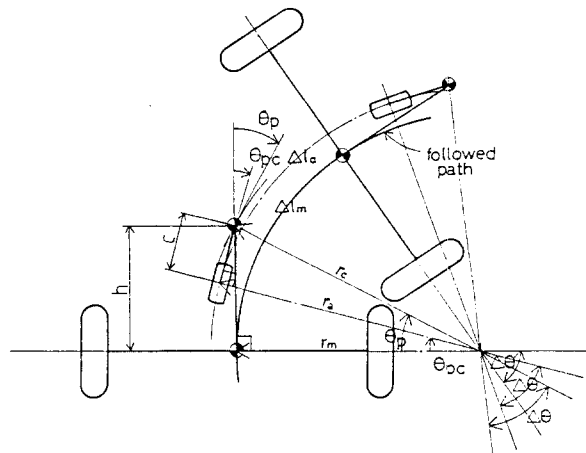


Fig 4. Model for calculating the correction of the travelled distance sensed on the aux. wheel

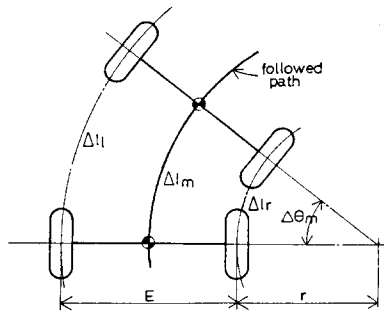


Fig 5. Model for calculating the travelled distance and the heading change

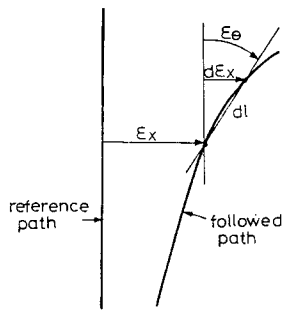
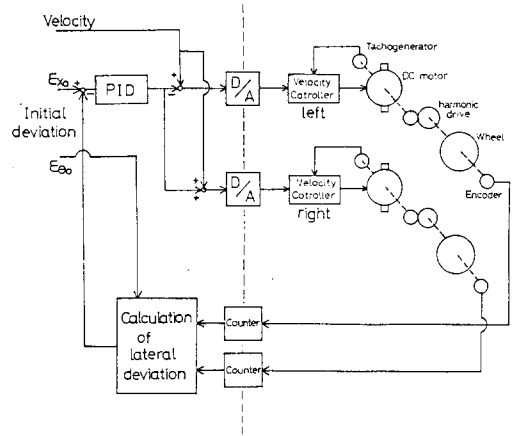


Fig 6. Relation between lateral deviation and directional deviation



(b) two-encoder odometer system

Fig 7. Structure of the control part

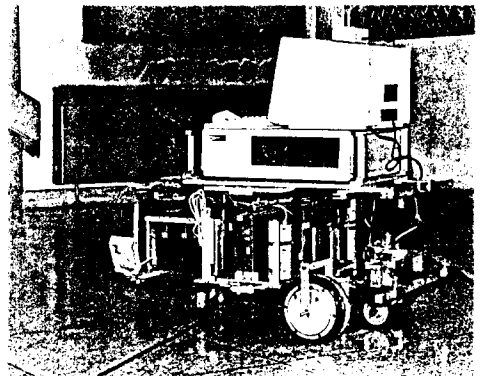
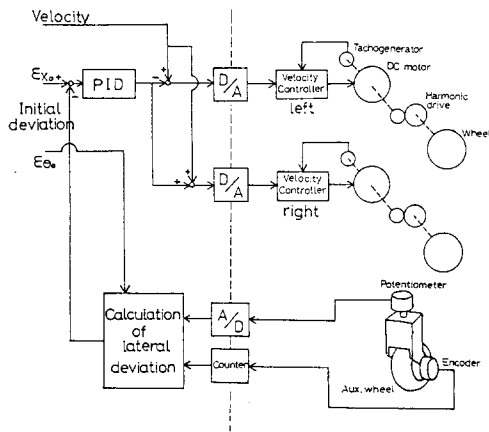
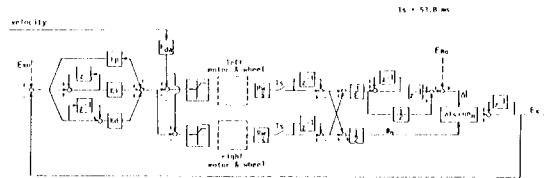


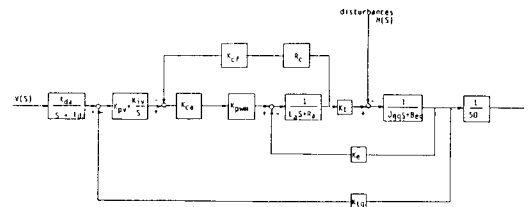
Fig 8. External view of the mobile robot 'KMR-2'



(a) aux. wheel odometer system



(a) block diagram of the overall algorithm



(b) block diagram of the motor and wheel system

Fig 9. Block diagram of the algorithm for path following

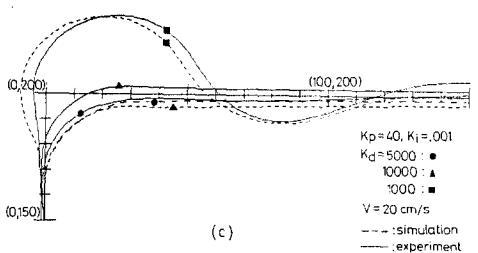
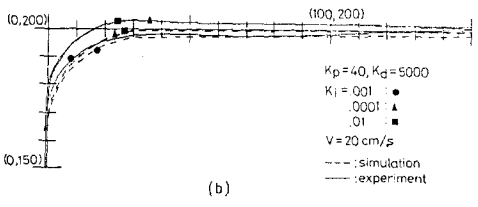
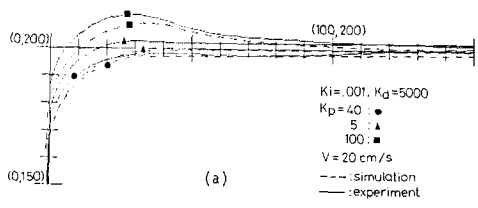


Fig 10. Comparison of the experimental results and the simulation results

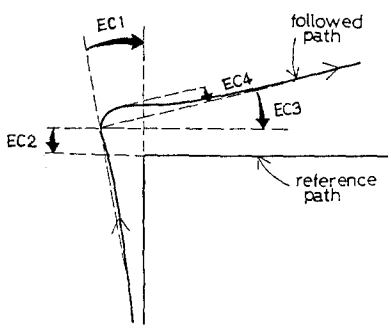


Fig 11. roles of the experimental coefficients

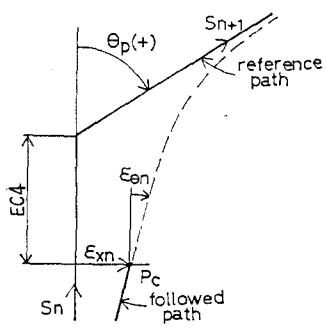
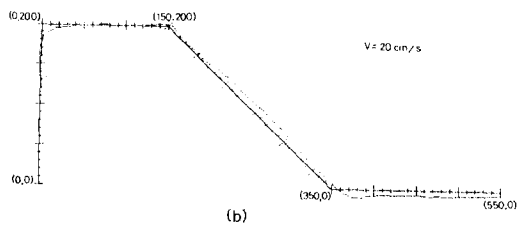
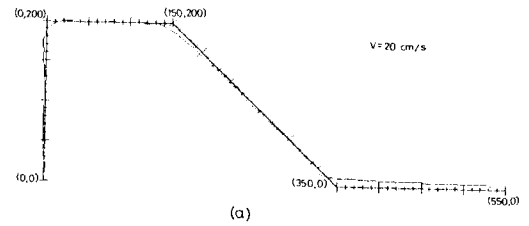
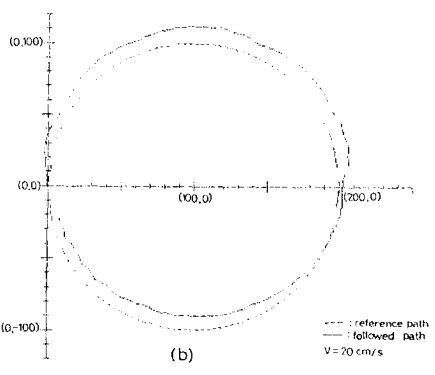
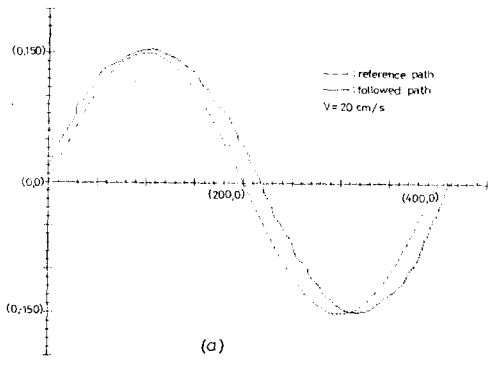


Fig 12. Definition for calculating starting point of a segment



(a) aux. wheel odometer system  
(b) two-encoder odometer system

Fig 13. Experimental results of the path following



(a) sine-reference path  
(b) circle-reference path

Fig 14. Experimental results of the path following