

Effects of the Sampling Time in Motion Controller Implementation for Mobile Robots

Tae-ho Jang · Youngshik Kim[†]

Department of Mechanical Engineering, Hanbat National University

모바일 로봇 모션 제어에 있어 샘플링 시간의 효과

장태호 · 김영식[†]

한밭대학교 기계공학부

In this research we investigate motion controller performance for mobile robots according to changes in the control loop sampling time. As a result, we suggest a proper range of the sample time, which can minimize final posture errors while improving tracking capability of the controller. For controller implementation into real mobile robots, we use a smooth and continuous motion controller, which can respect robot's path curvature limitation. We examine motion control performance in experimental tests while changing the control loop sampling time. Toward this goal, we compare and analyze experimental results using two different mobile robot platforms; one with real-time control and powerful hardware capability and the other with non-real-time control and limited hardware capability.

Keywords : Sampling Time, Motion Controller, Mobile Robot, Gain

1. 서론

본 연구에서는 모션제어기(motion controller)를 실제 모바일 로봇에 적용하는 데 있어 중요한 제어기 Sampling Time의 영향을 실험을 통하여 연구한다. 이를 위하여 Real-Time Operating System(OS) 및 Rapid Control Prototyping (RCP)을 위한 하드웨어와 소프트웨어 기반의 상대적으로 성능이 우수한 로봇 플랫폼과 상대적으로 하드웨어 성능에 제약이 큰 마이크로 컨트롤러 기반의 로봇 플랫폼에 동일한 제어기를 적용하여 Sampling Time 변화에 따른 위치 제어 및 안정성을 비교 분석한다. 모바일 로봇 제어에 대한 연구는 많은 연구자들이 관심을 갖고 지속적으로 진행되고 있다. 기존 모바일 로봇 제어 연구들에서 시물레

이션이나 실제 로봇에 적용하여 모션제어기를 검증하였지만[2, 4, 5, 7, 8, 9, 10], 제어 파라미터와 Sampling Time 등의 변화에 따른 제어기 성능에 대하여 분석한 연구는 찾아보기 어렵다. Akimasa[6]은 로봇간의 무선 통신과 Sampling Time과의 관계에 대해 연구하였다. Daniel[1]은 Emulab을 사용한 모바일 로봇 무선 제어 시스템에서 Sampling Time과 컨트롤러 반응에 대해 설명한다. 하지만 자율이동 형태의 로봇 시스템에서 Sampling Time에 따른 제어기 성능에 대해서는 고려되지 않았다.

본 논문에서는 물리적 제약 조건을 고려한 부드럽고 연속적인 모바일 로봇 모션 컨트롤러[3]를 실제 로봇에 적용하여 검증한다. 그리고 로봇의 물리적 제약을 고려하기 위해 제어기에 사용된 곡률 반경 변수의 범위를 선정한다. 또한 모바일 로봇의 모션 컨트롤러의 성능 향상을 위해 적절한 모터 제어기 Gain 값 및 제어기 Sampling Time 범위를 연구한다. 이를 통해 Sampling Time이 제어기 성능에 미치는 영향을 알 수 있다.

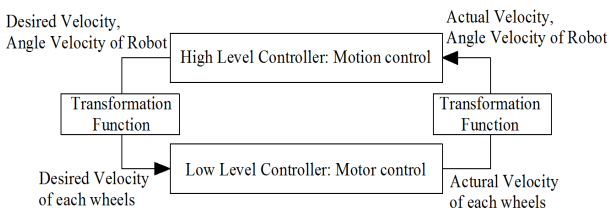
실험 진행은 다음과 같다. 설계된 제어기를 실제 모바일 로봇에 적용하고 적절한 모터 제어기 Gain 값을 실험적으로 찾아내고 근궤적(Root Locus)으로 안정성 등을 확인한다. 그리고 로봇의 물리적 제약과 제어성을 고려하여 적절한 곡률반경 범위를 선정한다. 또한 제어기의 Sampling Time을 변화시키며 로봇의 모션제어 반응을 분석하여 최종적으로 적절한 Sampling Time 범위를 제시한다. 이를 위하여 상대적으로 하드웨어 성능이 우수하고 Real-Time 제어가 가능한 모바일 로봇과 하드웨어 성능에 제약이 있고 고속 실시간 제어에 제약이 있는 모바일 로봇에 동일한 제어기를 적용하고 Sampling Time 변화에 따른 모션제어 결과를 비교 분석한다.

논문 구성은 다음과 같다. 제 2장에서는 모바일 로봇 컨트롤러의 구조에 대하여 논한다. 제 3장에서는 모션제어 실험 방법을 설명하고 실험 결과를 분석한다. 마지막으로 제 4장에서는 본 연구의 결론을 제시한다.

2. 컨트롤러 구조

모바일 로봇을 제어하기 위해 사용한 컨트롤러의 구성도는 <Figure 1>과 같다. 먼저 상위 레벨 제어기(High-Level Controller)에서 실제 모바일 로봇의 현재 위치와 상태로부터 로봇의 직선 속도명령과 각속도 명령을 생성한다. 속도 변환함수를 통해 그 값을 좌, 우 모터의 속도 값으로 변환하고 하위 레벨 제어기(Low-Level Controller)에서 이를 PWM으로 변환하여 모터를 구동시킨다. 이때 얻어지는 좌 우 모터의 속도 값은 다시 속도 변환함수를 통해 상위 레벨 제어기로 피드백 된다.

2.1 모션 컨트롤러 : High-Level Controller



<Figure 1> Control Scheme

<Figure 1>에서 볼 수 있듯이, 모션 컨트롤러는 원하는 위치와 현재 위치 값을 동시에 고려하여 로봇의 직선 속도 명령과 각속도 명령을 생성한다.

<Figure 2>는 로봇의 운동학을 보여준다. 여기서 프레임 R은 가상 로봇이며 참조 좌표(Reference Frame)를 나타낸다. 가상 로봇은 실제 로봇에게 경로를 제공한다.

Path manifold 기반 모션제어 방법[3]을 모바일 로봇에 적용하면 원하는(desired) 속도(v_d)와 각속도($\dot{\phi}_d$) 명령을 다음과 같이 구할 수 있다.

$$v_d = \frac{\left\{ \begin{array}{l} k_1 \eta (z_1 + \sqrt{2} r \eta) \tanh z_1 + v_r \eta (z_1 + \sqrt{2} r \eta) \cos \theta \\ + v_r r \sqrt{2} \sin \theta (\sin \theta + k_r (z_1 + \sqrt{2} r \eta)) \end{array} \right\}}{\eta (z_1 + \sqrt{2} r \eta) \cos (z_2 - \theta) + r \sqrt{2} \sin 2\theta \sin (z_2 - \theta)} \quad (1)$$

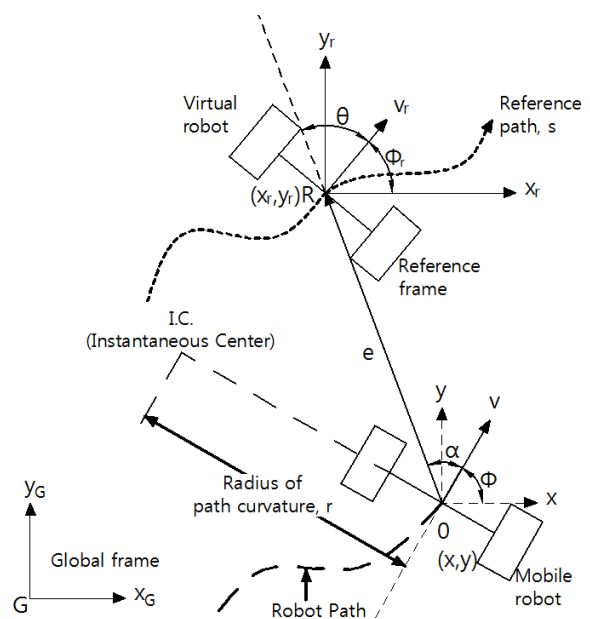
$$\dot{\phi}_d = kv = k_2 \tanh z_2 + 2\dot{\theta} + \dot{\phi}_r \quad (2)$$

위 식에서 k_1, k_2 는 Gain 값이며 아래 식 (3), 식 (4)로 표현된다. 변수 r은 실제 로봇의 수렴 경로 반경, v_r 과 k_r 은 각각 가상 참조 로봇의 속도와 곡률이다. η, z_1, z_2 는 다음의 식 (5)와 같다.

$$k_1 = (k_{1max} - k_{1min}) \left(1 - \tanh \left(\frac{g_1}{e} \right) \right) + k_{1min} \quad (3)$$

$$k_2 = 0.3 \tanh \left(\frac{1}{e(0)} \right) \tanh \left(\frac{1}{2|a|} \right) + 0.3 \tanh \left(\frac{1}{e} \right) \quad (4)$$

$$\begin{aligned} \eta &= \sqrt{1 - \cos 2\theta} \\ z_1 &= e - \sqrt{2} r \eta \\ z_2 &= \theta + a \end{aligned} \quad (5)$$



<Figure 2> Kinematics of a Mobile Robot

식 (3)에서 g_1, k_{1max}, k_{1min} 은 각각 1.3, 0.7, 0.2로 선정한다. 변수 k_{1min} 는 로봇의 최대 직선 속도와 관련된다. 식 (5)에

서 e , θ , α 는 극 좌표에서 로봇의 상태를 나타낸 것이며 식 (6)과 같다.

$$\begin{aligned}
 e &= \sqrt{(x-x_r)^2+(y-y_r)^2} \\
 \theta &= ATAN2(-(y-y_r), -(x-x_r))-\phi_r \\
 \alpha &= \theta-\phi+\phi_r
 \end{aligned} \tag{6}$$

식 (6)에서 x , y 는 실제 로봇의 위치이며 x_r , y_r 은 참조 (Reference) 위치 즉 최종 목표 위치이다. 변수 ϕ , ϕ_r 은 실제 로봇의 회전각과 가상 참조 로봇 즉 최종 위치에서의 회전각이다.

2.2 모터 컨트롤러 : Low-Level Controller

<Figure 1>에서 볼 수 있듯이, 모터 컨트롤러는 모션 컨트롤러로부터 계산된 제어용 직선 속도 명령(v_d)과 각 속도 명령($\dot{\phi}_d$)을 사용하여 실제 모터를 구동한다. 이 때 모터 컨트롤러의 Gain 값에 따라 로봇의 경로 추종 성능이 달라지므로[2] 모터 제어기의 적절한 Gain 값을 선정한다.

<Figure 3>는 적용한 모터 제어기의 시스템 블록선도이다. G_c 는 컨트롤러이며 전달함수 G_p 는 모터 시스템을 나타낸다. 여기서 G_c 는 PI제어기를 사용하므로 아래 식 (7)로 표현된다. G_p 는 실험을 통해 식 (8)처럼 구하여진다.



<Figure 3> Block diagram for Motor Control

$$G_c = K_p + \frac{1}{s}K_I \tag{7}$$

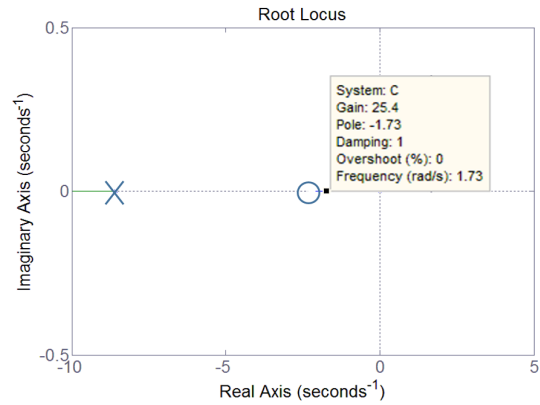
$$G_p = \frac{517.628}{S+8.57} \tag{8}$$

실험으로 선정한 모터 제어 Gain 값의 안정성을 근궤적으로 확인한다. 위 식을 특성화방정식에 대입하면 식 (9)와 같고 이를 근궤적(root-locus)으로 그리면 <Figure 4>와 같다.

$$1+K_p\left(\frac{S+\frac{K_I}{K_P}}{S\cdot(S+8.57)}\right)=0 \tag{9}$$

위 식에서 Gain 값 K_P , K_I 는 실험적으로 각각 25와 55

로 정하였고 근궤적을 통해 반응속도나 안정성 등에서 적절함을 확인했다. 로봇의 왼쪽 모터의 Gain도 위와 같이 선정한다.



<Figure 4> Root Locus

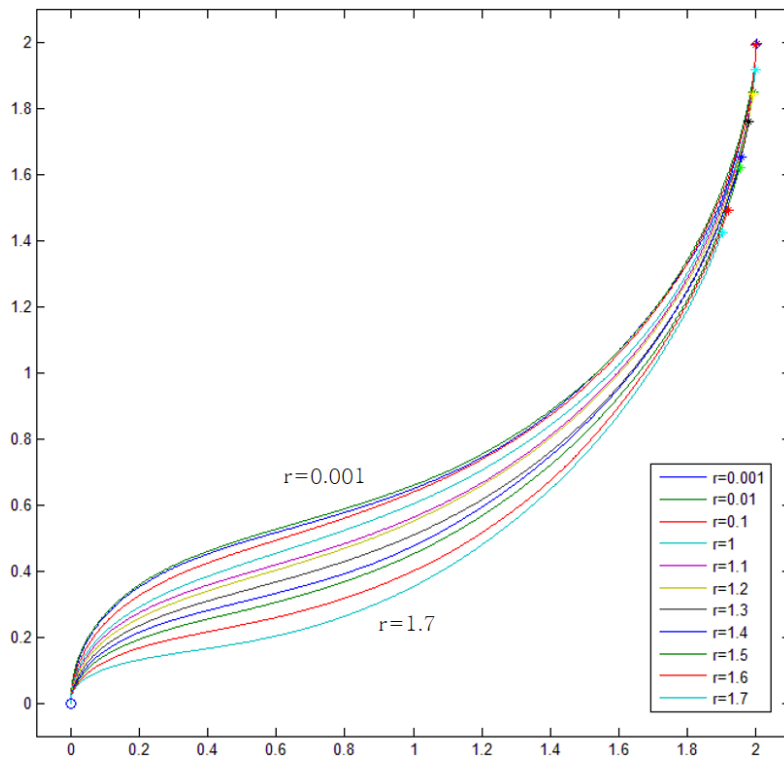
3. 실험

3.1 곡률 반경 선정

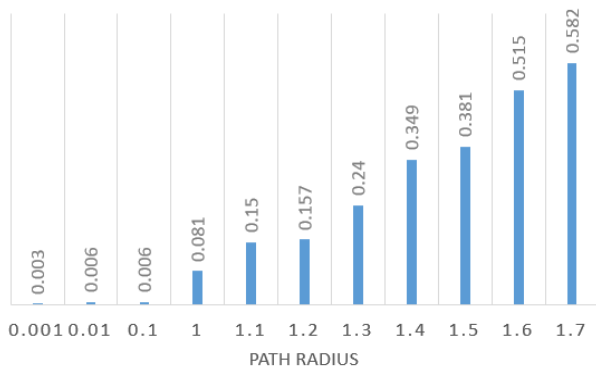
식 (1)에서 곡률반경(r)은 로봇의 속도 제어와 관련 있다. 여기서 곡률반경은 모바일 로봇의 물리적 회전반경을 만족시키기 위해 사용한다. 본 연구에서는 곡률 반경을 변화시키며 로봇의 위치 제어와 안정성을 분석하고 적절한 범위를 선정한다.

<Figure 5>는 원점(0m, 0m)에서 원하는 위치(2m, 2m)로 모바일 로봇을 자율주행 시켰을 때 곡률 반경에 따른 모바일 로봇의 경로 변화를 보여준다. 곡률반경이 1.7m에서 0.001m로 줄어들 때 로봇의 경로는 곡률반경이 0.001m일 때의 경로로 수렴하며 경로의 곡률도 작게 변화된다. 곡률반경 값이 0.001m 미만이나 1.7m 이상이 되면 모바일 로봇이 오작동하였다. 실험 결과, 물리적 회전반경을 만족시키기 위해 사용한 곡률반경이 최종 위치 에러에도 영향을 준다는 점을 알 수 있다.

<Figure 5>에서 곡률반경이 증가할수록 실험용모바일 로봇의 최종 위치가 목표 위치(2, 2)에서 멀어지는 것을 알 수 있다. <Figure 6>은 곡률 변화에 따른 모바일 로봇의 최종 위치 에러 값이다. 곡률 반경이 1m일 때 에러 값(0.081m)이 곡률반경 0.1m일 때 에러 값(0.006m)보다 21 배 급격히 증가하며 곡률반경이 1.7m까지 증가할수록 최종 위치 에러도 증가한다. 그러므로 곡률반경 범위는 에러 값이 작은 0.001m에서 0.1m까지로 선정한다. 이번 연구에서는 곡률 반경을 곡률반경 선정범위의 중간 값인 0.05m로 선정하여 실험한다.



<Figure 5> Actual Trajectory Paths of a Mobile Robot Changing the Radius of Path Curvature



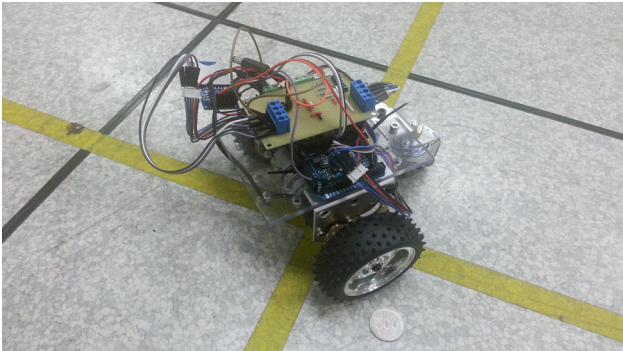
<Figure 6> Final Position Error as a Function of the Radius of Path Curvature(Unit : m)

3.2 샘플링 타임 선정

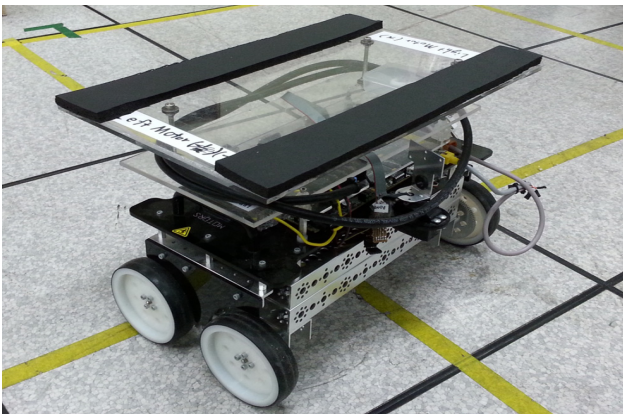
Akisma Otsuka[6]는 PC 서버와 모바일 로봇들 간의 무선 통신 Sampling 주기(Period) 변화에 따라 통신 성공률을 분석하고 적절한 통신 Sampling 주기를 제시하였다. 이와 같이 본 연구에서도 제어기 자체의 Sampling Time 변화에 따른 제어기 성능의 변화를 분석하고 적절한 Sampling Time 범위를 선정한다. 적절한 샘플링 타임 범위를 선정하기 위한 실험 과정은 다음과 같다. 먼저, 동일한 모션제어기를 각각 Real-Time 제어가 가능한 모바일 로봇과 Real-Time 제어가 보장되지 않는 상대적으로 성능이 제한된 모

바일 로봇에 적용한다. 각기 다른 사양의 OS와 제어 하드웨어 기반의 모바일 로봇으로 실험하여 모션제어결과를 비교 분석한다. 장애물이 없고 지면이 비교적 균등한 실내 환경에서 제어기 Sampling Time 변화에 따른 각각의 로봇 움직임을 비교 관찰한다. 여기서 Sampling Time 변화는, Real-Time 제어가 가능한 모바일 로봇에서는 Loop Time으로 제어하고 상대적으로 하드웨어 성능에 제약이 큰 실험용 모바일 로봇에서는 delay 함수를 사용하여 조정한다. Sampling Time의 단위는 milli sec(ms)이다. Sampling Time을 변화시키며 여러 번 반복 실험을 한다. 동일한 실험 조건 아래서 실험하기 위하여 좌표 시스템과 최대속도 제한 그리고 제어기를 시스템 구성과 사양이 서로 다른 두 로봇에 동일하게 적용하였다. 제어 시스템에 Sampling Time의 영향을 로봇의 최종 위치 에러와 추종 성능 등을 비교 분석하고 적절한 샘플링 타임 범위를 최종적으로 선정한다.

<Figure 7>과 <Figure 8>은 실험에 사용한 로봇들이다. 본 연구에서 사용된 실험용 모바일 로봇<Figure 7>은 고속 Real-Time 연속 제어에 제약이 있는 마이크로 컨트롤러를 사용한다. NI Starter Kit Robot <Figure 8>은 Real-Time 제어가 가능한 Single-Board RIO(sbRIO) 컨트롤러로 제어된다. 각 로봇의 바퀴 지름은 0.085m, 0.1m이며 바퀴간의 거리는 각각 0.155m, 0.357m이다. <Table 1>은 실험용 로봇과 NI Starter Kit Robot의 하드웨어 및 소프트웨어 사양을 비교하여 보여준다.



<Figure 7> Our Experimental Mobile Robot with Limited Hardware Capability

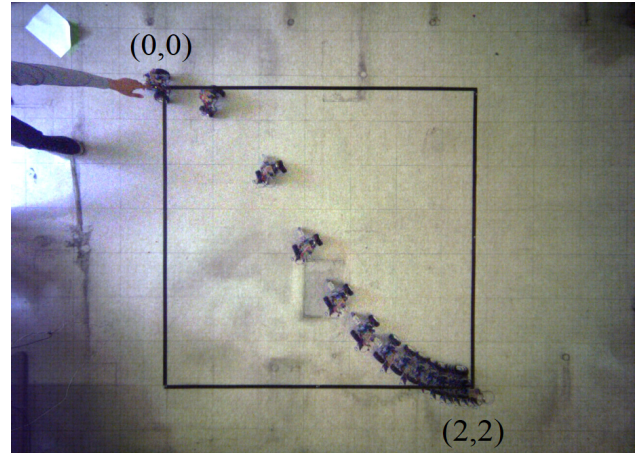


<Figure 8> NI Starter Kit Robot with Real-Time Control and Powerful Hardware Capability

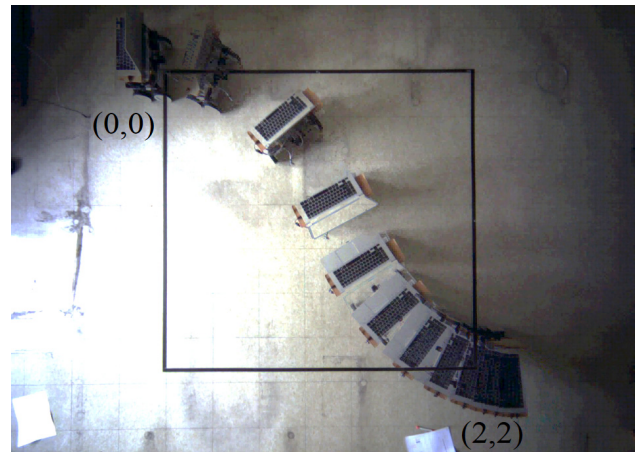
NI Starter Kit Robot이 실험용 로봇보다 고가이며 우수한 제어 시스템 성능을 가지고 있다. 참고로 NI Starter Kit Robot은 약 3배 더 빠른 CPU clock 속도를 갖는다. <Figure 9>는 실험용 로봇과 NI Starter Kit Robot의 실험 장면이다. 원점(0m, 0m)에서 목표점(2m, 2m)으로 이동시키며 2초 단위로 Capture하였다. 두 로봇 모두 목표점에 가까워질수록 간격이 좁아지는데 이는 속도가 느려지는 것이다. 실험에서 두 모바일 로봇의 이동 경로는 유사하며 적용한 제어기도 잘 작동한다.

<Table 1> Hardware Specifications of Robots

	Our Experimental Mobile Robot	NI Starter Kit Robot
Board	Arduino	Single-Board RIO(sbRIO)
Weight	1.5kg	3.6 kg
Size(mm)	131×178×98	405×368×150
Wheel Diameter(mm)	85	100
Motor	7W DC 12V	34W DC 12V
CPU clock	84Mhz	266Mhz



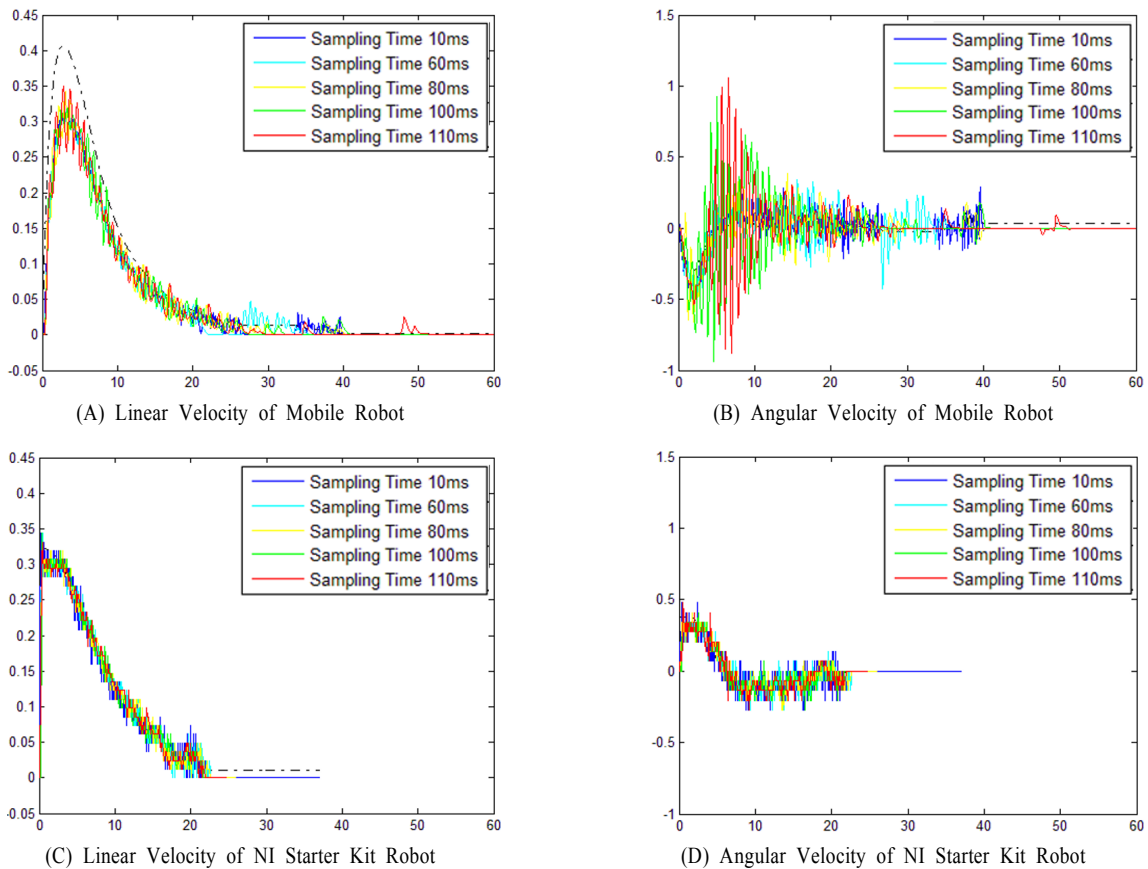
(A) Our Experimental Mobile Robot



(B) NI Starter Kit Robot

<Figure 9> Experimental Trajectory Paths

다수의 실험을 통해 얻어진 <Figure 10>는 원점에서 목표점으로 이동할 때 실험용 모바일 로봇 과 NI Starter Kit Robot 중심에서의 실제 직선 속도와 각속도를 보여준다. 직선 속도의 단위는 m/s, 각속도의 단위는 rad/s이다. 먼저 실험용 모바일 로봇과 NI Starter Kit Robot의 직선 속도를 비교하면, 최대 속도는 약 0.3m/s로 비슷하나 NI Starter Kit Robot의 속도가 실험용 모바일 로봇 보다 더 빨리 0에 수렴한다. 이는 NI Starter Kit Robot이 목표점에 더 빨리 도달하는 것이다. <Figure 10A> 실험용 모바일 로봇의 직선 속도에서는 Sampling Time이 10ms에서 60ms까지는 시그널 노이즈 크기가 0.01정도로 작으나 80ms부터 0.05 이상으로 증가한다. 시그널 노이즈는 Sampling Time에 비례하여 커진다. 0~10초까지는 상대적으로 명령 값과 실제 값에 큰 에러가 있지만 전반적으로 직선 속도 명령 값을 잘 따라간다. <Figure 10>(C)에서, NI Starter Kit Robot의 직선 속도는 Sampling Time에 상관없이 전반적으로 약 0.05크기의 시그널 노이즈가 존재한다.



<Figure 10> Desired and Actual Robot Velocities from Experimental Tests

실험용 모바일 로봇의 각속도에서는 Sampling Time이 100ms에서부터 노이즈가 최대 1.5 이상으로 크게 발생하며 실제 움직임도 매우 부자연스러워진다(<Figure 10>(B) 참조). <Figure 10>(D)에서, NI Starter Kit Robot은 Sampling Time에 상관없이 0.3크기의 일정한 노이즈를 갖는다. 각속도에서도 NI Starter Kit Robot이 실험용 모바일 로봇 보다 더 빨리 0에 수렴한다. 이는 NI Starter Kit Robot에 Dynamic Extension[3]을 사용하지 않았기 때문이다. 하지만 속도와 각속도 경향은 두 로봇 모두에서 유사하다.

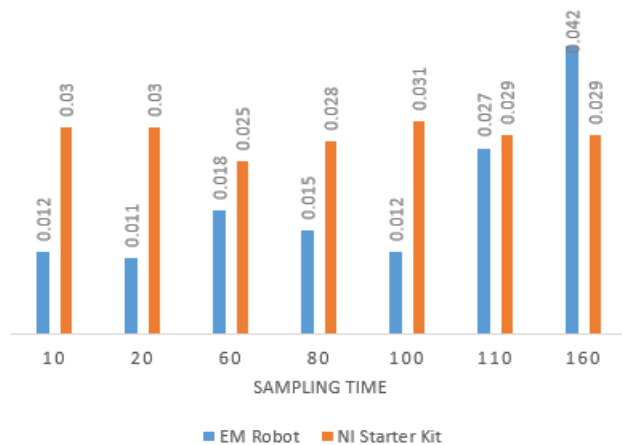
<Figure 10>에 의하면, 실험용 모바일 로봇은 NI Starter Kit Robot보다 Sampling Time에 영향을 많이 받지만 Sampling Time이 작으면 노이즈가 감소하여 에러가 작아진다. NI Starter Kit Robot은 전반적으로 노이즈가 발생하나 실험용 모바일 로봇보다 Sampling Time 변화에 영향을 적게 받아 안정적이며 목표점에 빨리 수렴한다.

<Figure 11>은 실험용 모바일 로봇과 NI Starter Kit Robot의 최종 위치 에러이다. NI Starter Kit Robot이 실험용 모바일 로봇 보다 바퀴간의 거리가 2.3배 더 크므로 이를 고려하여 NI Starter Kit Robot의 에러 값은 실제 에러 값을 2.3으로 나눈 값이다. 먼저 실험용 모바일 로봇의 최종 위치 에러는 Sampling Time이 20ms일 때 가장

작고 110ms일 때 크게 증가하기 시작하여 160일 때 가장 크다. 에러가 가장 클 때는 가장 작을 때보다 약 400%증가한다. NI Starter Kit Robot은 Sampling Time이 60ms일 때 에러가 가장 작고 100ms일 때 가장 크지만 전반적으로 Sampling Time 효과가 크지 않다.

본 연구를 통하여 제한된 하드웨어와 고속 Real-Time 제어에 제약이 있는 실험용 모바일 로봇은 에러 값이 주어진 Sampling Time 변화에 크게 영향을 받으나 NI Starter Kit Robot은 그렇지 않다는 사실을 확인하였다. 또한 실험 결과, Sampling Time이 가장 작을 때 에러가 가장 작은 것은 아니며 Sampling Time이 증가함에 따라 무조건 에러가 증가하는 것도 아님을 확인하였다. 따라서 고성능 Board 사용한다고 반드시 위치 에러가 더 줄어드는 것은 아님을 알 수 있다.

실험용 모바일 로봇이 NI Starter Kit Robot보다 Sampling Time에 영향을 크게 받지만 최종 위치 에러가 상대적으로 작음을 알 수 있다. 그리고 실험용 모바일 로봇의 실험결과를 바탕으로 적절한 Sampling Time 범위를 선정할 수 있다. Sampling time이 110ms부터 모바일 로봇의 최종 위치 에러가 눈의 띄게 증가한다. 그러므로 본 연구에서 얻은 적절한 Sampling time의 범위는 10~100ms이다.



<Figure 11> Final Position Error of Robots as a Function of the Sampling Time(Unit : ms)

4. 결론

본 연구에서는 Sampling Time의 변화에 따른 실제 모바일 로봇의 모션반응을 관찰하고 최소 위치 에러와 부드러운 경로를 갖는 적절한 Sampling Time 범위를 선정한다. 이를 위해 물리적 제약 조건을 고려한 부드럽고 연속적인 제어를 모바일 로봇에 적용한다. 먼저 모바일 로봇의 위치 오차를 줄이기 위해 적절한 모터제어 Gain 값과 곡률 반경을 선정하였다. 그리고 제어기 Sampling Time을 변화시키며 다른 두 컨트롤러 기반의 모바일 로봇들의 경로 및 위치 오차를 관찰하였다. 연구 결과, 먼저 상대적으로 하드웨어 성능에 제약이 큰 실험용 모바일 로봇의 모션제어 결과는 주어진 Sampling Time 변화에 큰 영향을 받는다. Sampling Time의 증가에 따라 위치 오차도 증가하며 경로도 부드럽지 않게 된다. 반면 Real-Time 제어가 가능한 상대적으로 성능이 우수한 NI Starter Kit Robot은 주어진 Sampling Time에 크게 영향을 받지 않았다. 즉, Sampling Time의 변화에 상관없이 전반적으로 비슷한 위치 에러를 가지며 주행 경로도 크게 변화하지 않았다. 참고로 적용된 모션 제어기는 두 모바일 로봇 모두에서 기대한 대로 잘 동작하였다. 따라서 두 로봇 모두 원하는 로봇 속도 명령을 전반적으로 잘 추종하였고 그 결과 유사한 경로로 이동하여 최종 위치에 작은 에러를 보이며 도달하였다.

적절한 Sampling Time 범위를 선정하기 위해 Sampling Time에 크게 영향을 받는 실험용 모바일 로봇 실험값을 사용한다. 모바일 로봇의 최종 위치 에러와 경로 Chattering 현상 등이 Sampling Time이 110ms부터 증가한다. 그러므로 적절한 Sampling Time 범위는 '10~100ms로 선정한다. 모바일 로봇 모두 제어기의 명령을 잘 따라가지만 소형

로봇의 경우 위의 주어진 Sampling Time 범위에서 구동 시 최상의 결과를 얻는다. 이 연구를 통해 Sampling Time이 모바일 로봇 제어 결과에 영향을 줄 수 있으며 적절한 Sampling Time 선정 시 위치 에러를 크게 줄일 수 있음을 알 수 있다. 향후 비정형 환경에서의 제어 연구와 최종 위치 오차를 줄이기 위한 센서 융합 등에 관한 연구를 수행할 예정이다.

Acknowledgements

This research was supported by the research fund of Hanbat National University in 2013. This research was also supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (2012R1A1A1011457).

References

- [1] Flickinger, D.M. and Minor, M.A., Remote Low Frequency State Feedback Kinematic Motion Control for Mobile Robot Trajectory Tracking. in *Robotics and Automation, 2007 IEEE International Conference on*, 2007, p 3502-3507.
- [2] Kim, J.-H., Trajectory Tracking Control for A Wheeled Mobile Robot. *The Institute of Electronics Engineers of Korea-IE*, 2009, Vol. 46, p 73-77.
- [3] Kim, Y. and Minor, M. A., Path manifold-based kinematic control of wheeled mobile robots considering physical constraints. *International Journal of Robotics Research*, 2007, Vol. 26, No. 9, p 955-975.
- [4] Ko, N.Y., Noh, S.W., Seo, Y.N., and Seo, D.J., A Probabilistic Approach to Grid-Map Generation for Autonomous Navigation of a Mobile Robot. 2009, p 280-282.
- [5] Lenain, R., Thuilot, B., Cariou, C., Bouton, N., and Berducat, M., Off-road mobile robots control : An accurate and stable adaptive approach. in *Communications, Computing and Control Applications (CCCA), 2012 2nd International Conference on*, 2012, p 1-6.
- [6] Otsuka, A., Nagata, F., and Okino, T., Measurement of dynamic sampling period of multi mobile robots system with wireless communication. in *Mechatronics and Automation (ICMA), 2013 IEEE International Conference on*, 2013, p 1141-1146.
- [7] Seul, N.O., Lee, K.J., and Yoo, S.G., Mobile Robot Control using SLAM and Passive Theory. *Journal of the*

- Korea Entertainment Industry Association*, 2013, Vol. 7, No. 1, p 134-140.
- [8] Son, J.B., Choi, S.H., and Lee, J.M., A Study on path planning of an intelligent autonomous mobile robot control using fuzzy control. 2010, p 115-118.
- [9] Wu, W., Simulation of Mobile Welding Robot with Variable Structure Control. in *Innovative Computing and Communication, 2010 Intl Conf on and Information Technology and Ocean Engineering, 2010 Asia-Pacific Conf on (CICC-ITOE)*, 2010, p 106-109.
- [10] Yang, G.-J. and Choi, B.-W., Joint Space Trajectory Planning Considering Physical Limits for Two-wheeled Mobile Robots. *Journal of Institute of Control, Robotics and Systems*, 2013, Vol. 19, No. 6, p 540-546.