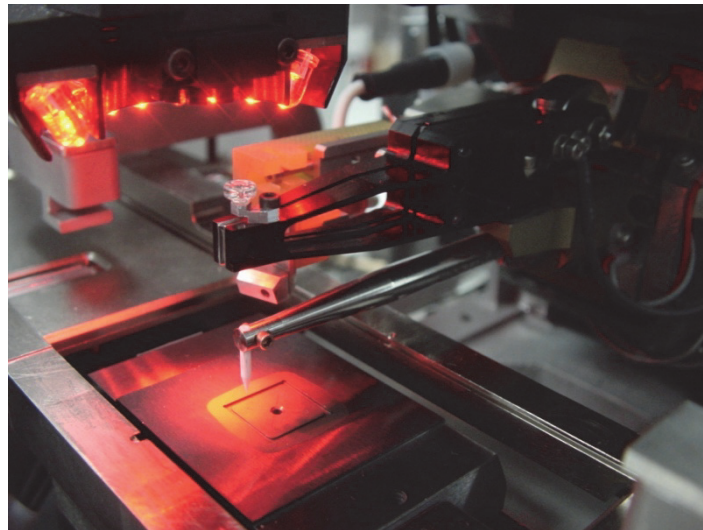


Design of a Motion Profiles – Part A

- 2.1 Introduction to Motion Profiles
- 2.2 Relation between Motion Profile and Current Consumption
- 2.3 Design of S-profiles using 3rd order equation



2.1 Introduction to Motion Profiles

모션 프로파일(motion profile)이란 구동대상에 대한 시간에 따른 목표궤적을 말한다. 따라서 모션 프로파일은 목표 궤적의 물리량에 따라 위치 프로파일, 속도 프로파일, 가속도 프로파일 저크 프로파일 등으로 구분된다.

서보 테이블의 구동을 위해서는, 우선적으로 제어의 목표값으로 이용되는 레퍼런스(reference) 궤적(profile)이 생성되어야 하는데, 이는 DSP의 매 인터럽트(interrupt)마다 목표치로 참조되는 점들의 집합으로, 고성능의 서보시스템으로 갈수록 모션 프로파일의 중요도는 높아지며, 이는 피드-포워드(feed-forward) 제어에도 중요한 영향을 미친다.

구동대상이 워크 피스(work piece)이든 정밀 측정 프루브(probe)이든 간에 항상 물리적으로 관성과 진동특성을 가지고 있기 때문에, 시스템에 맞는 모션 프로파일을 설계하는 것은 매우 중요한 일이며, 또한 시스템의 에러(error)량과 진동에 큰 영향을 준다. 다음의 Fig. 2.1은 실제의 서보시스템 구동시 기록한 종모양의 속도 프로파일과 구동시의 위치오차를 도시한 것이다.

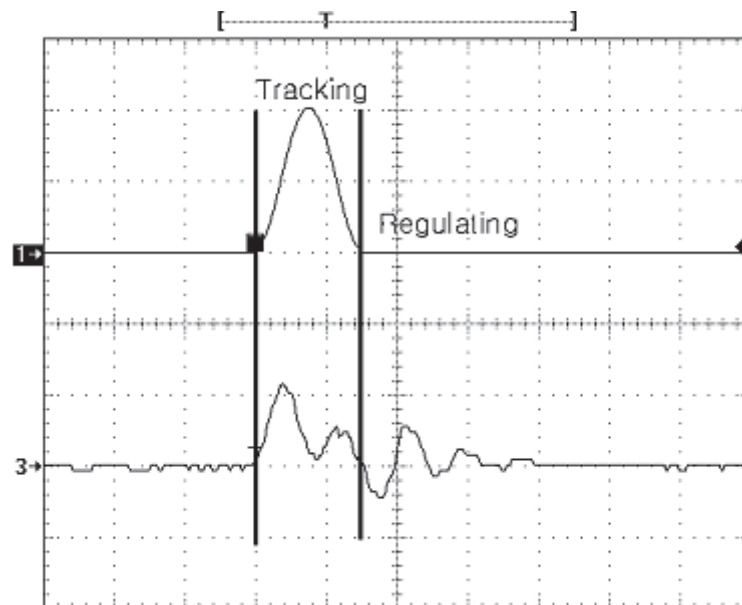


Fig. 2.1 Bell-shaped motion profile

짧은 스트로크를 가지는 고속서보시스템은 Fig. 2.1과 같은 등속구간이 없는 종 모양의 속도 프로파일을 가지는 경우가 많은데, 속도 프로파일은 시간에 따른 시스템의 대략적인 움직임의 표시에 적합하기 때문에 서보시스템의 에러 체크나 내부 신호의 트리거신호

로도 많이 사용된다. Fig. 2.1과 같이 에러와 속도 프로파일을 같이 도시하면, 움직임의 시작과 끝을 파악하기 쉬우므로 에러의 분석에 용이하며, 피드포워드(feedforward) 제어기의 튜닝에 매우 편리하다.

일반적으로 가장 흔하게 볼 수 있는 모션 프로파일은 다음 Fig. 2.2와 같은 사다리꼴 형상의 속도 궤적이다.

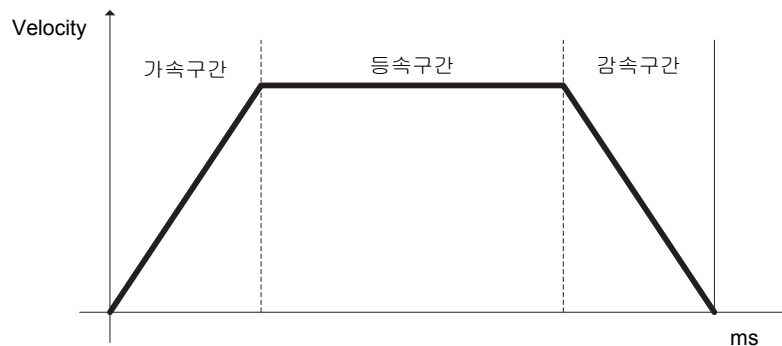


Fig. 2.2 Trapezoidal motion profile

Fig. 2.2의 사다리꼴 속도 프로파일은 가속구간, 등속구간, 그리고 감속구간으로 3등분되며, 감가속이 일정한 양으로 행해지는 모션 프로파일이라 계산이 쉽고, 프로그램이 간편한 장점이 있어서 일반적으로 많이 사용된다. 반면에 모션의 시작과 등속구간의 진입, 감속구간의 시작, 구동의 마침 부분에서 큰 가속도의 변화가 일어나므로, 이 부분에서 구조물의 진동을 많이 발생시키므로 아주 정밀한 시스템에는 잘 쓰이지 않는다. Fig. 2.2과 같은 속도 프로파일로 스테이지를 구동했을 때 위치 및, 가속도 프로파일도 같이 도시해보면 Fig. 2.3과 같다.

같은 모션 프로파일이라고 해도 위치 프로파일과 속도 프로파일은 그 형상이 다르므로 모션 프로파일은 반드시 속도 프로파일 인지 위치 프로파일 인지를 확인해야 하며, 위치,

속도, 가속도, 저크등의 물리량이 도함수 관계에 있으므로, 위치, 속도, 가속도, 저크 프로파일이 하나의 모션 프로파일 세트set라고 이해하면 편리하다.

일반적으로 모션 프로파일을 언급할 때 S-프로파일profile이라고 하는 경우에는 위치 프로파일을 대상으로 하는 경우가 많은데, 속도에 대한 S-프로파일 또한 많이 사용하므로, 프로파일을 지칭할 때는 위치인지 속도인지 가속도인지를 꼭 명기해야 한다.

Fig. 2.2과 같이 사다리꼴의 속도 프로파일을 가지는 경우, 가속도 프로파일은 스텝 형태의 가속도 값을 가지는데, 시스템의 마찰을 무시하면 이론적으로는 감가속 구간에서만 어떠한 일정량의 값을 가지게 된다.

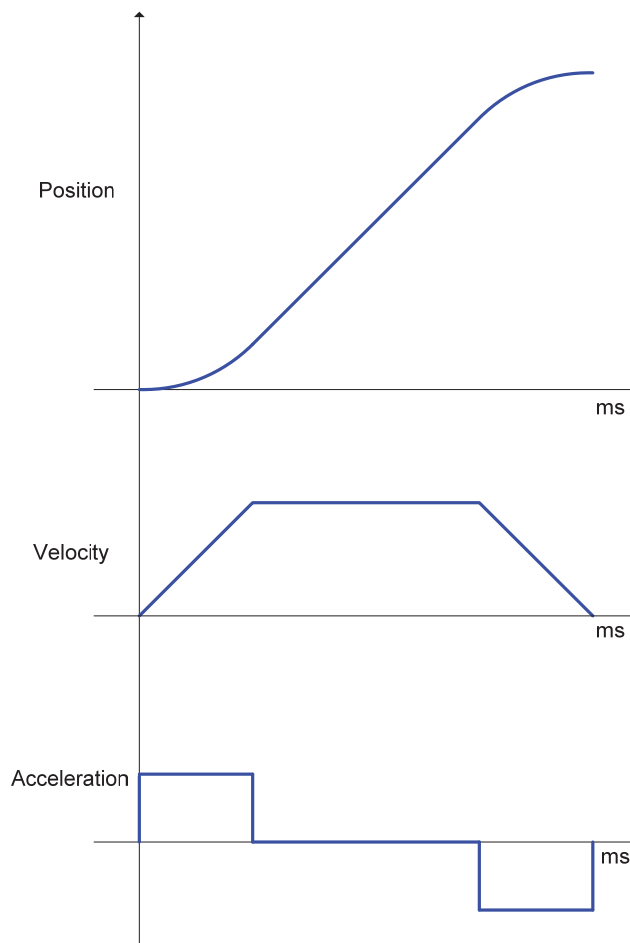


Fig. 2.3 Position, Velocity and Acceleration profiles

2.2 Relation between Motion Profile and Current

시스템에 따른 모션 프로파일의 최적 설계가 매우 중요한 이유 중 하나는 모션 프로파일 중 가속도 프로파일이 액추에이터(actuator)의 소모 전류와 가지는 연관성 때문이다. 뉴턴의 제2법칙에서,

$$F = ma \quad (2.1)$$

이므로 고정된 질량을 가지는 물체의 가속도는 그 물체에 가해지는 힘과 비례하며, 식 (2.2)와 같이 코일을 사용한 액추에이터에서 발생하는 힘(혹은 토크)은 일반적으로 공급해주는 전류의 양과 비례한다.

$$F = K_m i \quad (2.2)$$

여기서 물체의 질량과 모터상수는 변하지 않으므로 F 에 대한 시간에 대한 변화를 고려하여 식 (2.1)과(2.2)를 조합하면 식 (2.3)이 도출된다.

$$i(t) = \frac{m}{K_m} a(t) \quad (2.3)$$

식 (2.3)이 말해주는 바는, 모터에 공급해 주어야 하는 전류의 파형은 제어 대상체의 목표 가속도 프로파일인 $a(t)$ 에 비례한다는 것이다. 정리해서 간단하게 말하면, Fig. 2.2과 같은 사다리꼴 속도 프로파일로 X축 스테이지를 구동하면 해당 X축 모터에 가속도 프로파일과 유사한 형태의 전류가 소모된다는 것이다.

만약 마찰이 없는 이상적인 형태의 스테이지 기구물이 있다면, Fig. 2.3의 가속도 프로파일과 같은 형상으로 적절한 스케일의 전류를 오픈 루프open loop로 흘려주면, 그림 상단의 위치 프로파일과 같은 부드러운 형태의 시간에 따른 속도와 위치의 움직임을 보일 것이다. 따라서 어떠한 프로파일로 모터를 구동할 것인가가 정해지면 가속도 프로파일을 미리 계산하여 구동 전에 시간에 따른 전류 파형이 어떻게 될 것인지를 준비할 수 있다.

이러한 구동 계획에 따라 구동 시 인터럽트interrupt 마다 필요한 전류량을 미리 계산하여 제어기 출력에 더해 주면 피드백 제어기의 부담을 크게 덜 수 있으며, 잘 만들어진 시스템 일수록 질량(부하)의 변화가 미미할 때 상기한 가속도 프로파일과 제어기 출력되는 실제 전류 프로파일의 형태가 비슷하게 일치한다. 이렇게 시스템의 모델링된 부분에 대한 예측 전류치를 미리 계산해서 피드백 제어기의 출력에 더해주는 제어기를 피드-포워드feed-forward 제어기라 한다. 따라서 구동부의 질량의 변화가 없는 경우의 일반적인 피드-포워드 제어기는 에러의 함수가 아니라 시간에 따른 가속도 프로파일의 함수이다.

상기한 가속도 프로파일을 이용한 피드-포워드 제어기는 가장 간단한 형태의 피드-포워드 제어기이며, 여기에도 어떠한 주파수 영역에서의 특성 변환을 가져오는 기타의 알고리즘이 추가될 수 있다. 하지만 다양한 용도로 사용되는 범용 모션제어기에서는 많은 경우, 가속도 프로파일에다 튜닝된 계수 값을 곱한 형태의 간단한 피드-포워드 제어기가 좋은 성능을 보이는 경우가 많아 일반적으로는 가장 많이 사용된다.

그림 3.1.4는 일반적인 사다리꼴 형태의 속도 프로파일로 구동했을 때 실제 출력되는(피드백제어를 포함한) 모션 제어기의 전류제어 명령을 비교 도시한 것이다. 실제의 스테이지에서는 리니어가이드linear guide나 크로스롤러crossroller 베어링 등의 마찰력, 회전-직선 변환장치, 모터의 비선형성 등 다양한 원인으로 인하여 그림에 표기된 바와 같이 실제

출력 값은 약간 왜곡된 형상을 보이게 된다. 등속구간을 살펴보면 이상적인 상태(마찰력이 없는)에서는 전류가 입력되지 않는 것이 맞지만 실제로는 마찰력이 항상 존재하므로 일정 오프셋을 가지는 경향을 보인다.

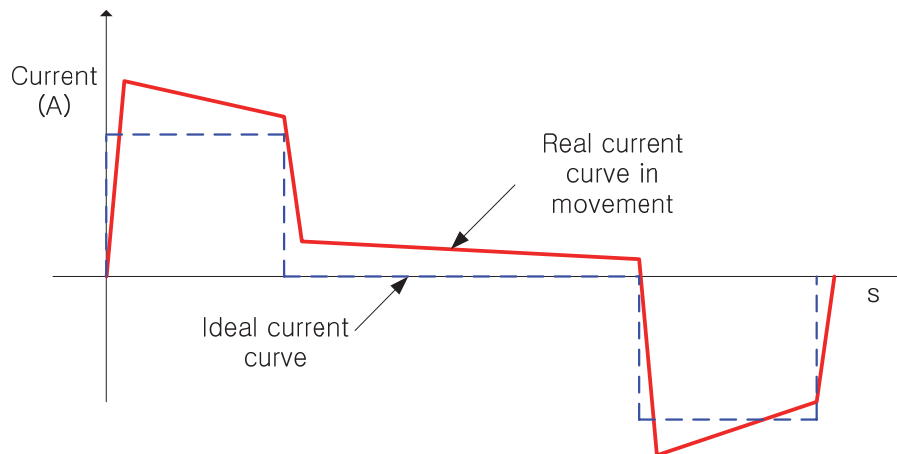


Fig. 2.4 Ideal Current vs. Real Current in movement

이러한 실제 시스템의 경향은 조립된 상태에 따라서도 제각기 다르므로 동일한 설계의 제품도 개별적으로 약간씩 다른 양상을 보인다. 이러한 이유 때문에 정밀한 스테이지의 경우는 제품 출고시 피드-포워드 feed-forward 제어기의 파라미터 parameter를 제품 개별적으로 튜닝 tuning 조정해 준다. 반면 피드백 feedback 제어기의 튜닝은 연구소나, 제품 개발실에서 일률적으로 최적 게인을 튜닝해서 정하는 경우가 일반적이다.

2.3 Design of S-profiles using 3rd order equation

본 절에서는 부드러운 곡선을 사용하여 모션 프로파일을 설계하는 방법에 대하여 설명할 것이다. 기본적으로 문제의 단순화를 위하여 일단 Fig. 2.1과 같은 등속구간이 없는 모션 프로파일을 먼저 생각해 보자. 다음의 식(2.4)과 그 도함수들인 식(2.5)과 (2.6)을 고려하자.

$$y = a + bt + ct^2 + dt^3 \quad (2.4)$$

$$\dot{y} = b + 2ct + 3dt^2 \quad (2.5)$$

$$\ddot{y} = 2c + 6dt \quad (2.6)$$

식(2.4), (2.5), (2.6)은 각각 위치, 속도 그리고 가속도의 프로파일을 나타낸다. 구동 프로파일을 유도할 때는 반드시 표준화된 방정식normalized equation으로 구하는 것이 나중에 실제 DSP에 프로그램 해 넣기 편리하다. 따라서 1이라는 시간에 1이라는 거리를 가는, 거리에 대하여 표준화된 프로파일normalized position profile을 구하는 것이 가장 편리하다.

위의 식(2.4)는 3차식이므로 여기서 결정해야 할 미지수는 상수까지 포함하여 4개이다. 따라서 4개의 경계조건boundary condition을 반영하여 구체화 시킬 수 있다. 우선 시간이 0인 시점에서의 위치와 속도는 모두 0 이고, 시간이 1일 때 위치는 1 그리고 속도는 역시 0이 되어야 한다는 가장 기본적인 4개의 경계조건을 고려하자.

$$\text{위치에 대한 구속조건: } y(0) = 0, y(1) = 1 \quad (2.7)$$

$$\text{속도에 대한 구속조건: } \dot{y}(0) = 0, \dot{y}(1) = 0 \quad (2.8)$$

경계조건들을 식 (2.4), (2.5)에 대입하면,

$$a = 0, b = 0 \quad (2.9)$$

$$c + d = 1, 2c + 3d = 0 \quad (2.10)$$

최종적으로 해를 구하면 최종 표준화된 위치 프로파일은 다음과 같이 구하여 진다.

$$y = 3t^2 - 2t^3 \quad (2.11)$$

이렇게 표준화된 위치 방정식 normalized position equation을 실제 DSP에 장착하기 쉬운 형태로 바꾸어 주면 다음과 같이 식(2.12)에서 식(2.15)로 표기할 수 있다.

$$y\left(\frac{t}{T}\right) = \left[3\left(\frac{t}{T}\right)^2 - 2\left(\frac{t}{T}\right)^3\right]S \quad (2.12)$$

$$\dot{y}\left(\frac{t}{T}\right) = \left[\frac{6}{T}\left(\frac{t}{T}\right) - \frac{6}{T}\left(\frac{t}{T}\right)^2\right]S \quad (2.13)$$

$$\ddot{y}\left(\frac{t}{T}\right) = \left[\frac{6}{T^2} - \frac{12}{T^2}\left(\frac{t}{T}\right)\right]S \quad (2.14)$$

$$\ddot{\ddot{y}}\left(\frac{t}{T}\right) = -\frac{12}{T^3}S \quad (2.15)$$

여기서 S 는 이동거리, T 는 구동 시간이라 정의한다. 위와 같은 방법으로 구한 구동 프로파일은 테이블 형태로 롬ROM에 저장 하든지 아니면, 수식의 형태로 인터럽트 루틴 내에 위치시킨다. 위에서 구한 식 (2.11)의 3차식 프로파일 테이블은 다음과 같이 간단한 루틴에 의해 Matlab을 이용하여 생성해 낼 수 있다.

Matlab에서 표준화된 테이블 normalized table 생성 이후 fopen과 fprintf 명령을 쓰면 간단하게 텍스트 text 파일로 저장할 수 있다. 일단 텍스트 파일로 바꾸면 이를 간단히 변형하여 C 언어나 어셈블리어에서 불러 쓸 수 있을 것이다. 이 과정에서 중요한 것은 바로 자리수인데, DSP가 허용하는 최대한의 자리 수를 확보해야 한다. 예제 프로그램은 소수점 후 12 자리까지 파일로 프린트하라고 하였는데, 독자 여러분은 각각 사용 하는 DSP의 상황

에 맞게 자릿수를 맞추어 주면 된다. 그리고 TMS320C3X등 Floating Point DSP를 사용하는 분은 테이블의 참조 알고리즘 작성시 각별히 주의해야 한다. 이에 대해서는 뒤에서 보다 자세히 다룰 것이다.

다음의 Matlab Source 2.1 은 3차 구동 방정식의 테이블 생성 프로그램 이며, Fig. 2.5~Fig. 2.7은 각각 구하여진 3차방정식의 위치, 속도, 가속도 프로파일이다.

Matlab Source 2.1

```

% 3th order motion profile generator
% Number of Table cell is 2048
% HanKim

clear;
i = 0;

for i=1:2048,
    nt = i/2048;
    PTABLE(i) = 3*nt^2 - 2*nt^3;
    VTABLE(i) = 6*nt - 6*nt^2;
    ATABLE(i) = 6 - 12*nt;
end

x = ((1/2048):(1/2048):1)';
figure(1);plot(x,PTABLE,'r-');grid on;
xlabel('Normalized Time');ylabel('Normalized Distance');
figure(2);plot(x,VTABLE,'r-');grid on;
xlabel('Normalized Time');ylabel('Normalized Velocity');
figure(3);plot(x,ATABLE,'r-');grid on;
xlabel('Normalized Time');ylabel('Normalized Acceleration');

fid = fopen('ptable.txt','w');
```

```

for j=1:512,
    i = j*4-3;
    fprintf(fid,'%13.12f, %13.12f, %13.12f, %13.12f, \n',
        PTABLE(i),PTABLE(i+1),PTABLE(i+2),PTABLE(i+3));
end
fclose(fid);

```

인터럽트 주파수가 빠른 경우(4kHz 정도 이상), 구동 프로파일은 주로 위치에 대한 테이블 position table로 저장되며, 보통 중력가속도를 기준으로 3G 이상의 고속 서보 시스템에서는 2048개 이상의 테이블 분해능을 사용한다. 속도나 가속도 프로파일의 계산은 저장된 위치 테이블을 인터럽트마다 수치 미분하여 사용하지만, 수치 미분이 매끄럽지 못할 경우 가속도 테이블을 하나 더 사용하는 방법도 있다.

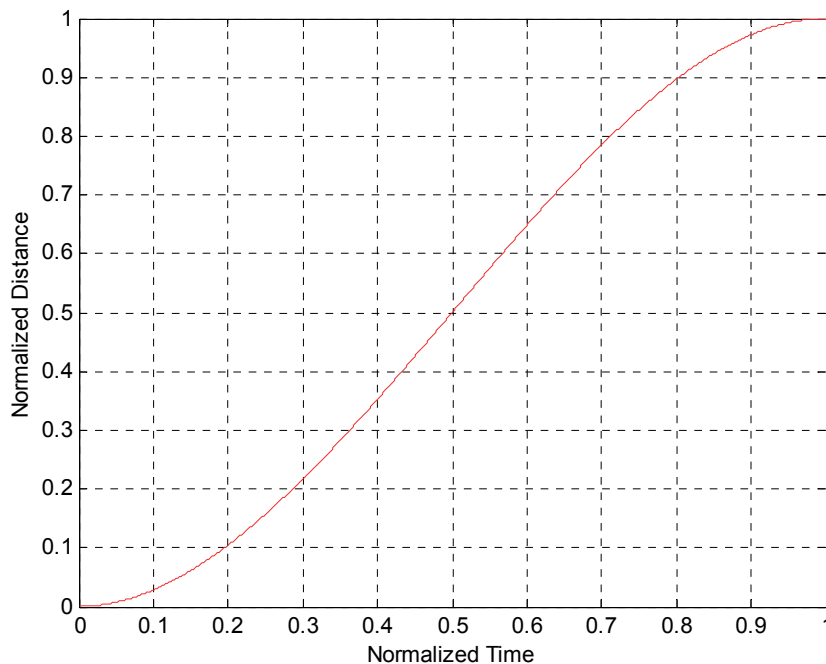


Fig. 2.5 Position Profile using 3rd order equation

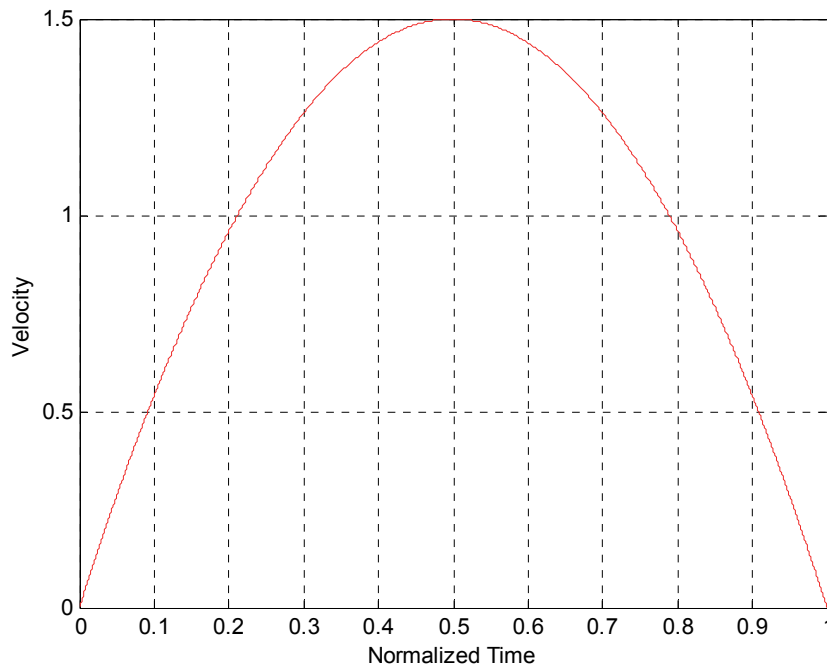


Fig. 2.6 Velocity Profile using 3rd order equation

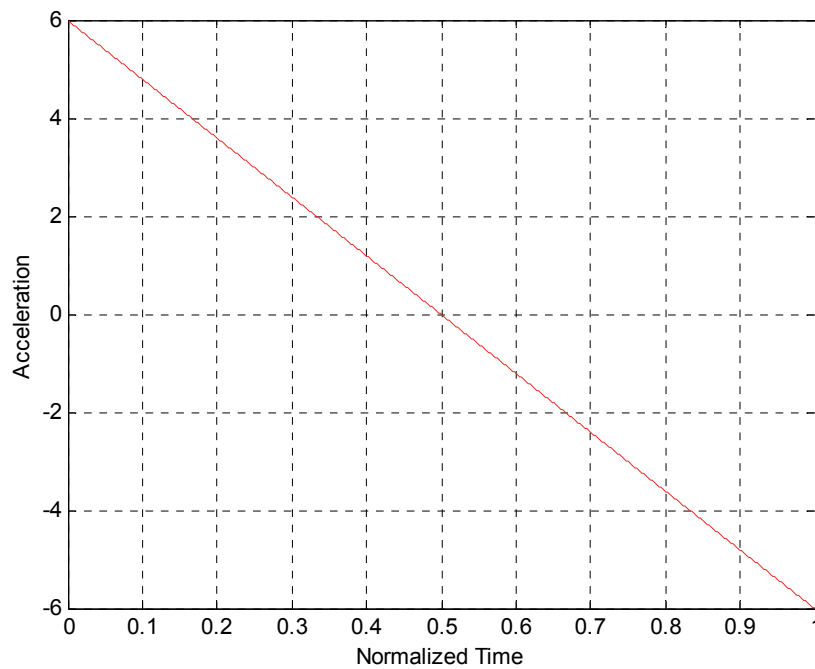


Fig. 2.7 Acceleration Profile using 3rd order equation

