물리기반 모델링 2장

시뮬레이션 기초

Models and Simulation

물리 기반 애니메이션의 구성

- Model

  \- 모델은 일련의 법칙 또는 규칙을 따라 어떤 행동 또는 작동된다.

- Simulation

  \- 시뮬레이션은 모형을 모방하여 시간 경과에 따른 변화하는 행동을 예측 가능하게 하는 것이다.

A physically based model

- 물리 기반 모델은 시스템이 작동하는 방식의 물리학을 정의한 모델이다.

- 물체의 움직임을 지배하는 규칙
- 환경의 행동
- 실제 행동
- 기타

A physically based model

- 현실세계에서 작용 되는 법칙을 통해서 행동 되는 것을 기반으로 한다.
- 항상 “진짜 " 모델 조차도 단순화 되어 표현된다.
- 게다가 애니메이션을 목적으로 한 것은, 현실세계의 물리학을 준하지 않고 따로 정의한 물리학을 따르는 것이 바람직할때도 있다.

예로, Wile E. Coyote라는 절벽을 달리는 것으로 유명한 애니메이션이 있다.

NEWTON'S LAWS OF MOTION

- 1. 관성의 법칙
- 2. 가속도의 법칙
- 3. 작용반작용의 법칙

관성의 법칙

- 외력이 가해지지 않는 한 물체는 고정 된 속도로 계속 움직이려 한다.
- 물체의 관성을 극복하는 것이 얼마나 어려운지는 물체의 질량에 의해 결정된다.
- 관성과 관련된 운동량은 물체의 질량과 속도의 곱으로, P=mv 로 표기된다.

가속도의 법칙

- 힘과 운동량의 변화율 사이의 정확한 관계를 가속이라고 부른다.
- 물체에 작용한 힘과 물체의 질량 및 가속도 사이에는 '힘 = 질량×가속도'라는 관계가 성립한다. $F = ma$
- 물리 기반 애니메이션에서 우리는 시간이 지남에 따라 변화하는 속도에 중점을 두고있다.
- 속도는 위치의 시간 변화이며 $v = \frac{dx}{dt}$ 가속도는 속도의 시간 변화이다. $a = \frac{dv}{dt}$

가속도의 법칙

- v = $\int a\,dt = at + v_0$
- 이것을 통합하면 x = $\int (at + v_0)\,dt$
- 여기서 x = $\frac{1}{2}at^2 + v_0 t + x_0$

Foundations of Physically Based Modeling and Animation

- (Modeling)행동을 좌우하는 힘의 모델을 정의한다. 뉴턴의 두번째 법칙에 의하면, 이러한 힘은 가속도를 결정한다.
- (Provide Initial Conditions) 초기조건으로는 위치와 속도가 있다.
- (Simulation) 시간의 변화에 따른 위치와 속도의 적분을 통하여 이루어진다.

DROPPING A BALL IN 1D

- 예제로 1차원에서 공을 떨어뜨리는 실험을 할 것이다.
- 중력 이외에 외부 힘은 신경 쓰지 않고 공기의 저항은 무시 한다.
- 객체는 하나의 점만 가진다.

DROPPING A BALL IN 1D

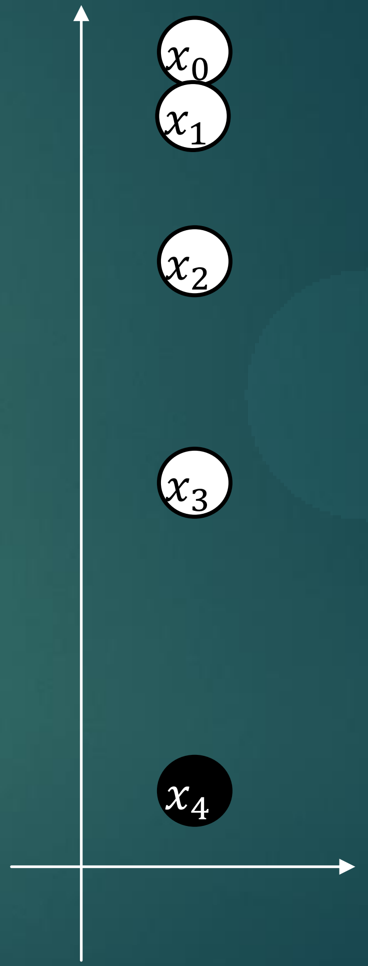
- 첫번째로 모델을 정의한다.

- 지상에 있는 물건을 다루고 있다고 가정.
- 중력은 거의 일정한 하향 힘 F=-mg를 발생 시키며, g는 중력상수이다.
- F=ma 이므로 a = -g 를 가진다. 공의 질량과는 독립적이다.
- 중력 가속도 상수 g 는 지구상에서 약 9.8m/s^2 으로 알려진 값을 사용.
- 실험은 정확한 값을 사용할 필요가 없으므로 g = 10 , $v_0 = 0$, $x_0 = 100$사용.

DROPPING A BALL IN 1D

- V = $at + v_0$ = -10t + 0 = -10t
- X = $\frac{1}{2}at^2 + v_0 t + x_0 = \frac{1}{2}(-10)t^2 + 0t + 100$ = 100 - 5t^2

Exact	t	0	1	2	3	4
	v	0	-10	-20	-30	-40
	x	100	95	80	55	20



Euler Integration

- 여러가지 수치적 기법 중 기본적인 오일러 통합
- 위의 예제에서 공의 속도는 v 위치는 x 이다. 초기 상태의 시간은 t = 0
- 변수에 대괄호로 묶은 위 첨자를 사용하여 루프 반복을 나타낸다.
- 오일러 통합의 기본 가정은 시뮬레이션에 통합 된 모든 값이 시간 변화 동안 일정하게 유지된다는 것이다.

Euler Integration

- 일반적인 시뮬레이션 루프는 다음과 같은 형식을 취한다.

현재 상태 = 초기 상태;
t = 0;
while $t < t_{max}$ do // note : 여기서 현재 상태는 시간 t에서의 상태
    현재 상태에서 힘을 결정합니다;
    뉴턴의 두 번째 법칙에 의한 가속도 결정; $a = \frac{1}{m} F$;
    새로운 상태 = 시간 경과에 따른 가속도의 통합 h;
    현재 상태 = 새로운 상태;
    $t = t + h$;
End

$$v^{[n+1]} = v^{[n]} + a^{[n]}h,$$
$$x^{[n+1]} = x^{[n]} + v^{[n]}h,$$

Euler Integration

- 위에서 하였던 볼 예제에 대한 오일러 통합 시뮬레이션 루프는 다음과 같다.

$v^{[0]} = 0;\ x^{[0]} = 100;$

$t = 0; n = 0;$

While $t < t_{max}$ do

  $a^{[n]} = -10;$

  if t 출력 프레임 시간 then

    Output $v^{[n]}, x^{[n]}$;

  end

  $v^{[n+1]} = v^{[n]} + a^{[n]}h;$

  $x^{[n+1]} = x^{[n]} + v^{[n]}h;$

  $n = n + 1; t = nh;$

end

Exact	t	0	1	2	3	4
	v	0	-10	-20	-30	-40
	x	100	95	80	55	20
Euler $h = 1$	v	0	-10	-20	-30	-40
	x	100	100	90	70	40