Numerical Integration

Erin Catto
Blizzard Entertainment

Basic Idea

- Games use differential equations for physics.
- These equations are *hard* to solve exactly.
- We can use numerical integration to solve them approximately.

Overview

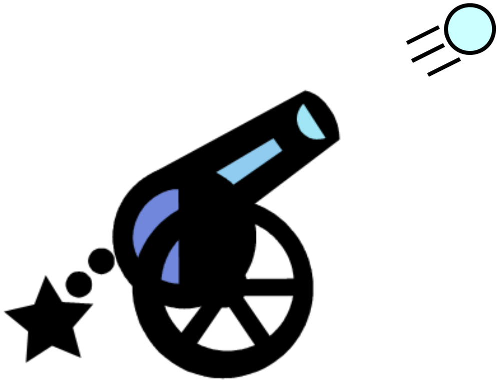
- Differential Equations
- Numerical Integrators
- Examples

Differential Equations

- Consider the movement of a projectile.
- The independent variable is time.
- The dependent variable is position and velocity.
- We have initial conditions for all variables.

Projectile Motion

- Consider the vertical motion of a projectile.



$$ma = F$$

$$ma = -mg$$

$$a = -g$$

Black Box

- Differential equations must be put into a special format.

$$\frac{dx}{dt} = f\left(t, x\right)$$

$$x\left(0\right) = x_0$$

Black Box

- Arrays of equations too.

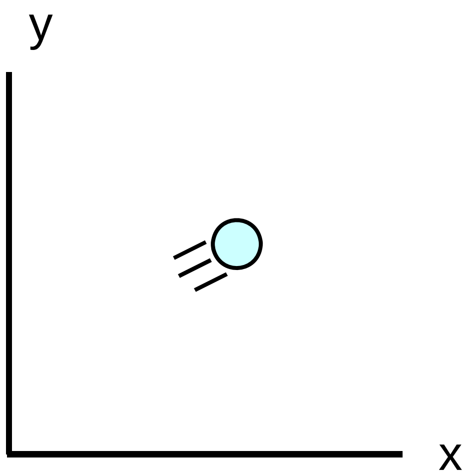
$$\frac{dx_1}{dt} = f_1\left(t, x_1, \cdots, x_n\right)$$

$$\vdots$$

$$\frac{dx_n}{dt} = f_n\left(t, x_1, \cdots, x_n\right)$$

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}\left(t, \mathbf{x}\right)$$

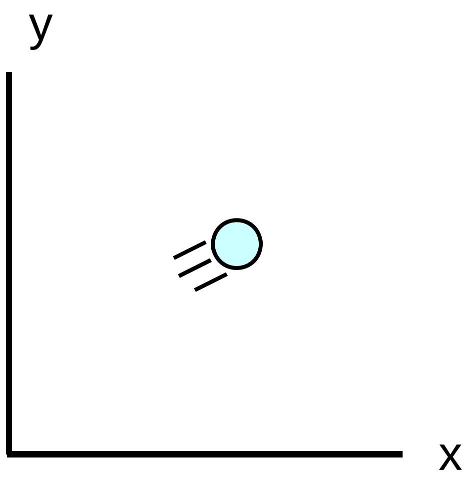
Projectile Motion



$$\frac{dy}{dt} = v$$

$$\frac{dv}{dt} = -g$$

Initial Conditions

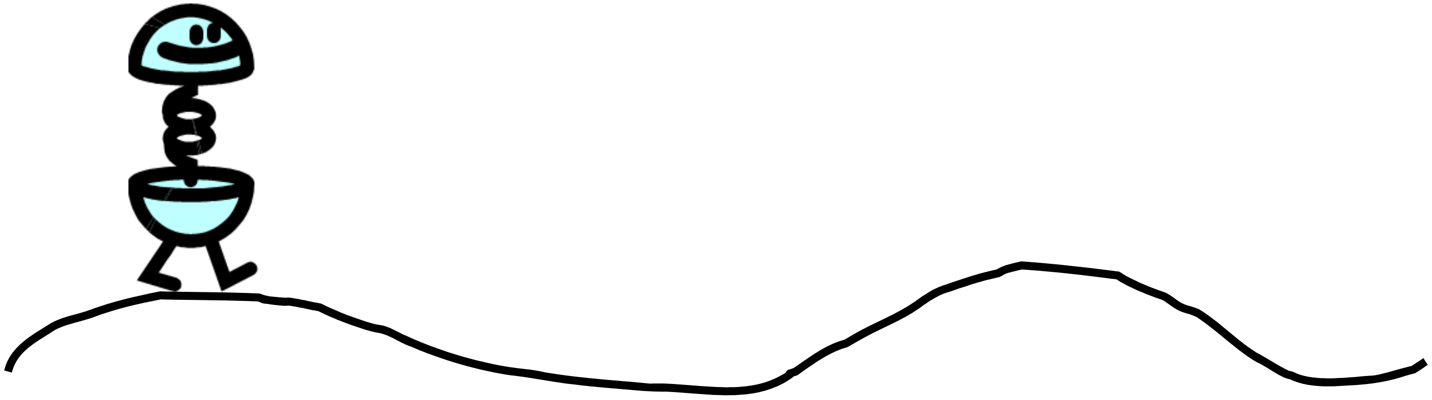


$$y(0) = y_0$$

$$v(0) = v_0$$

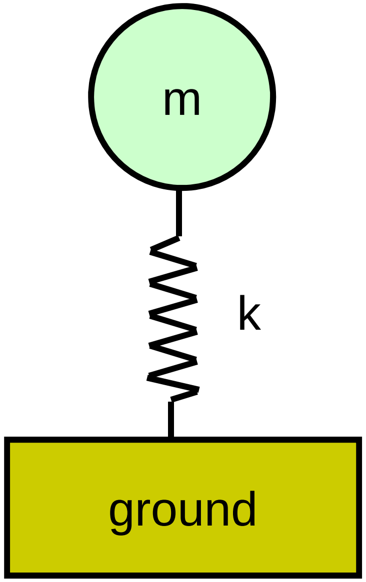
Mass-Spring Motion

- Consider the vertical motion of a character.



Mass-Spring Motion

- Idealized model.



$$ma = F$$

$$ma = -kx$$

$$a = -\frac{k}{m}x$$

Mass-Spring Motion

- Black box equations.

$$\frac{dx}{dt} = v \qquad x(0) = x_0$$

$$\frac{dv}{dt} = -\frac{k}{m}x \qquad v(0) = v_0$$

Numerical Integration

- Start with our black box equation. This is called an ordinary differential equation (ODE).

$$\frac{dx}{dt} = f(t, x)$$

$$x(0) = x_0$$

Solving an ODE

- Sometimes we can solve our ODE exactly.
- Many times our ODE is too complicated to be solved exactly.
- How can we approximate the solution?

A Simple Idea

- Forward difference:

$$\frac{dx}{dt} \approx \frac{x(t+h) - x(t)}{h}$$

A Simple Idea

- Shuffle terms:

$$\frac{x(t+h)-x(t)}{h}=f\left(t,x(t)\right)$$

$$x(t+h)=x(t)+h\,f\left(t,x(t)\right)$$

A Simple Idea

- Using this formula, we can make a time step *h* to find the new state.
- We can continue making time steps as long as we want.
- The time step is usually small

$$x(t+h) = x(t) + h\,f\left(t, x(t)\right)$$

Explicit Euler

$$x(t+h) = x(t) + h\, f\left(t, x(t)\right)$$

- This is called the *Explicit Euler* method.
- All terms on the right-hand side are known.
- Substitute in the known values and compute the new state.

What If …

$$x(t+h) = x(t) + h\,f\left(t+h, x(t+h)\right)$$

- This is called the *Implicit Euler* method.
- The function depends on the new state.
- But we don’t know the new state!

Implicit Euler

$$x(t+h) = x(t) + h\, f\left(t+h, x(t+h)\right)$$

- We have to solve for the new state.
- We may have to solve a nonlinear equation.
- Can be solved using Newton-Raphson.
- Usually impractical for games.

Implicit vs Explicit

- Explicit is fast.
- Implicit is slow.
- Implicit is more stable than explicit.
- More on this later.

Opening the Black Box

- Explict and Implicit Euler don’t know about position or velocity.
- Some numerical integrators work with position and velocity to get gain some advantages.

The Position ODE

$$\frac{dx}{dt} = v$$

- This equation is trivially linear in velocity.
- We can exploit this to our advantage.

Symplectic Euler

$$\frac{v(t+h)-v(t)}{h} = f\left(t, x(t), v(t)\right)$$

$$\frac{x(t+h)-x(t)}{h} = v(t+h)$$

- First compute the new velocity.
- Then compute the new position using the new velocity.

Symplectic Euler

- We get improved stability over Explicit Euler, without added cost.
- But not as stable as Implicit Euler

Verlet

- Assume forces only depend on position.
- We can eliminate velocity from Symplectic Euler.

$$\frac{v(t+h)-v(t)}{h} = f\left(t, x(t)\right)$$

$$\frac{x(t+h)-x(t)}{h} = v(t+h)$$

Verlet

- Write two position formulas and one velocity formula.

$$x_1 = x_0 + h\,v_1$$

$$x_2 = x_1 + h\,v_2$$

$$v_2 = v_1 + h\,f_1$$

Verlet

- Eliminate velocity to get:

$$x_2 = 2x_1 - x_0 + h^2 \; f_1$$

Integrator Quality

1. Stability
2. Performance
3. Accuracy

Demos

- Projectile Motion
- Mass-Spring Motion

Stability

- Extrapolation
- Interpolation
- Mixed
- Energy

Performance

- Derivative evaluations
- Matrix Inversion
- Nonlinear equations
- Step-size limitation

Accuracy

- Accuracy is measured using the Taylor Series.

$$f(x+h) = f(x) + f'(x)h + \frac{1}{2}f''(x)h^2 + \cdots$$

Further Reading & Sample Code

- http://www.gphysics.com/downloads/
- Hairer, Geometric Numerical Integration