Modeling and Simulation of Physical Systems for Hobbyists

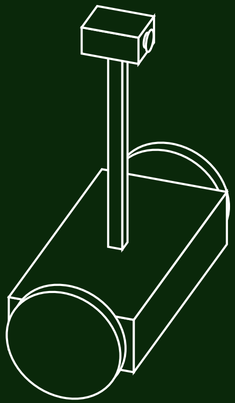
Manuel Aiple

35C3 — 29 December 2018

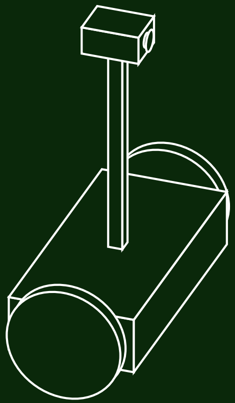
version f59c89b

Modeling	Simulation	Physical Systems	Hobbyists
Mathematical description of a system	Running a model	Hardware	With commonly available tools

Why use Simulation?



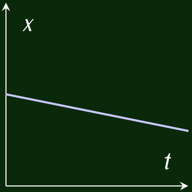
Why use Simulation?

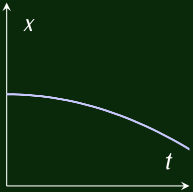


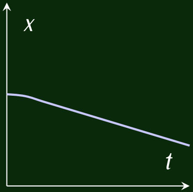
- Placeholder for hardware components
- Virtual test bench

Simple → Detailed → Very Detailed

Apple Moves Down	Apple Accelerates Down	Apple Accelerates Down Until Saturation







As simple as possible, as detailed as necessary

Differentiation & Integration

Position

$x(t)$

Velocity

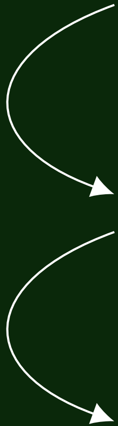
$v(t)$

Acceleration

$a(t)$

Differentiation & Integration

Position
$x(t)$

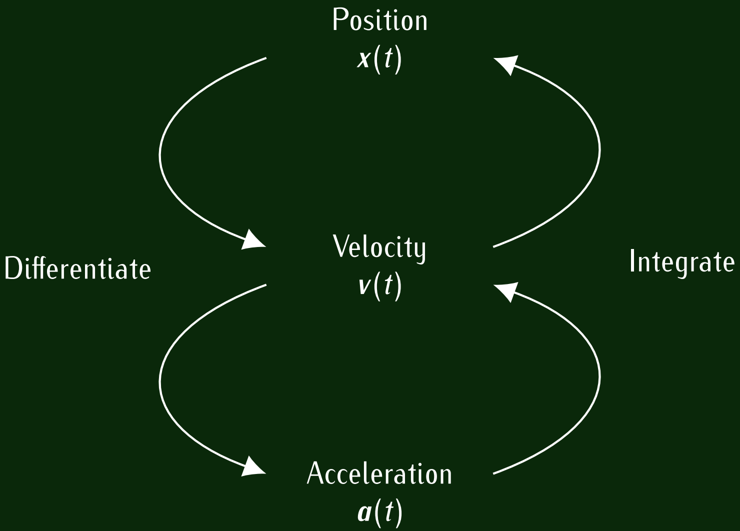


Differentiate

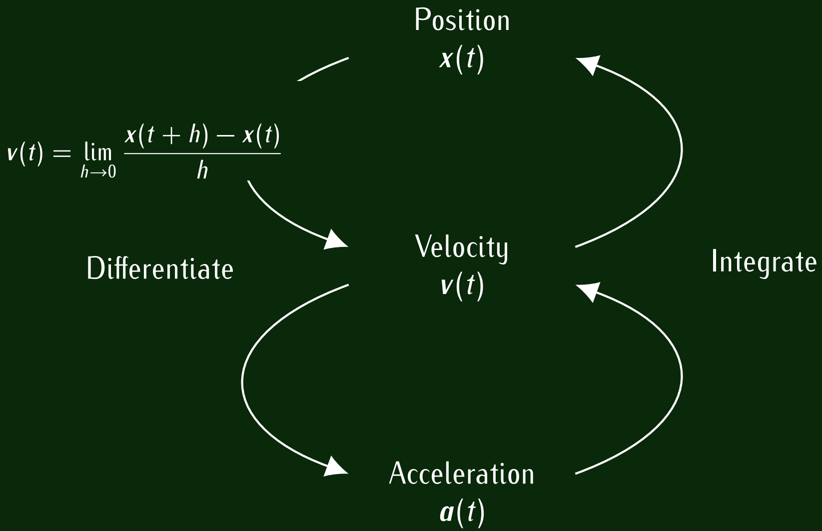
Velocity
$v(t)$

Acceleration
$a(t)$

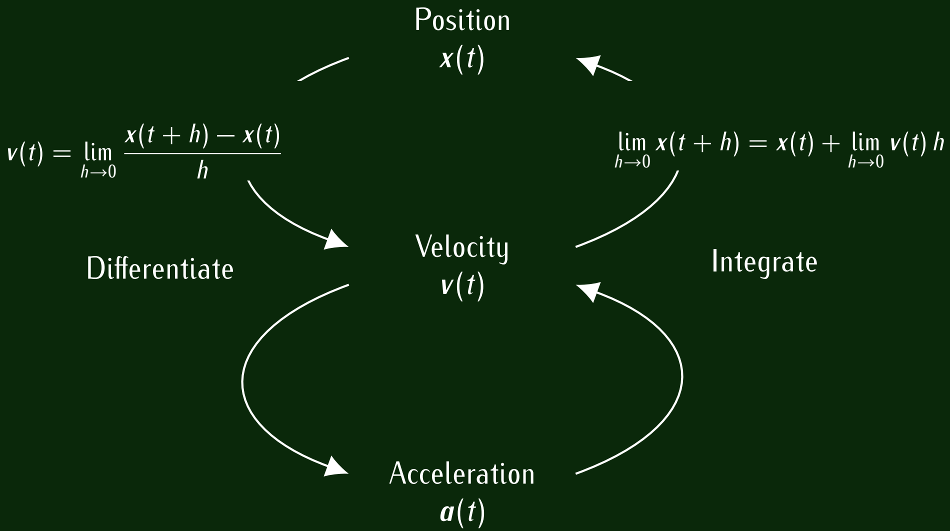
Differentiation & Integration



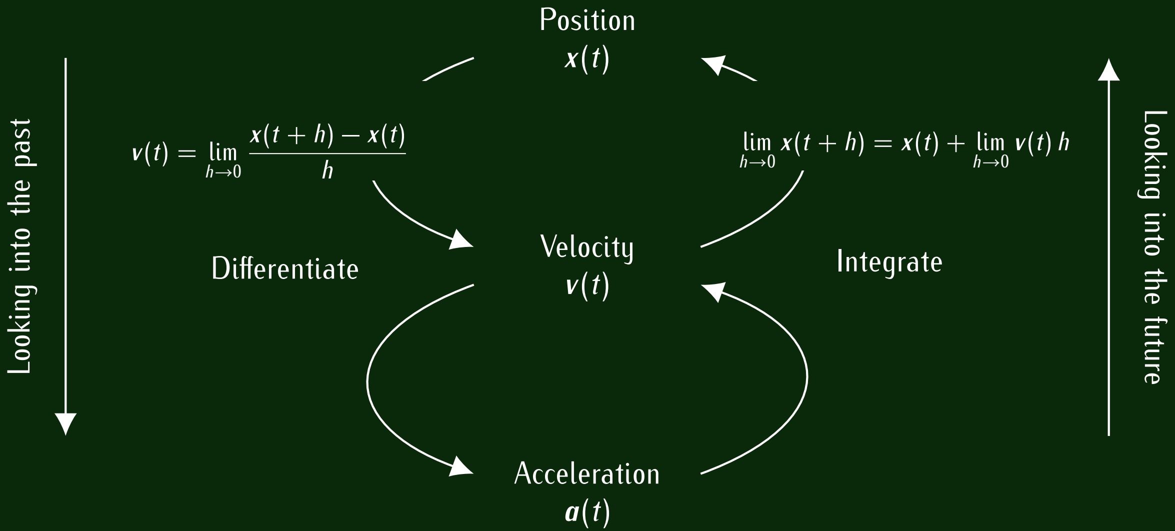
Differentiation & Integration



Differentiation & Integration



Differentiation & Integration



Always integrate for simulation

Euler Method

$$\lim_{h \to 0} x(t+h) = x(t) + \lim_{h \to 0} v(t)\, h \quad \to \text{Not usable for computation}$$

Euler Method

$$\lim_{h\to 0} \boldsymbol{x}(t+h) = \boldsymbol{x}(t) + \lim_{h\to 0} \boldsymbol{v}(t)\,h \quad \to \text{Not usable for computation}$$

Replace infinitesimal $\lim_{h\to 0} h$ with finite T_s and only calculate for integer multiples k of T_s:

$$t = k\,T_s$$

Euler Method

$$\lim_{h\to 0} \boldsymbol{x}(t+h) = \boldsymbol{x}(t) + \lim_{h\to 0} \boldsymbol{v}(t)\, h \quad \to \text{Not usable for computation}$$

Replace infinitesimal $\lim_{h\to 0} h$ with finite T_s and only calculate for integer multiples k of T_s:

$$t = k\, T_s$$

$$\boldsymbol{x}\left(\frac{t+T_s}{T_s}\right) = \boldsymbol{x}\left(\frac{t}{T_s}\right) + \boldsymbol{v}\left(\frac{t}{T_s}\right)\, T_s$$

Euler Method

$$\lim_{h\to 0} \boldsymbol{x}(t+h) = \boldsymbol{x}(t) + \lim_{h\to 0} \boldsymbol{v}(t)\, h \quad \to \text{Not usable for computation}$$

Replace infinitesimal $\lim_{h\to 0} h$ with finite T_s and only calculate for integer multiples k of T_s:

$$t = k\, T_s$$

$$\boldsymbol{x}\left(\frac{t+T_s}{T_s}\right) = \boldsymbol{x}\left(\frac{t}{T_s}\right) + \boldsymbol{v}\left(\frac{t}{T_s}\right)\, T_s$$

$$\boldsymbol{x}(k+1) = \boldsymbol{x}(k) + \boldsymbol{v}(k)\, T_s$$

Euler Method

$$\lim_{h\to 0} \boldsymbol{x}(t+h) = \boldsymbol{x}(t) + \lim_{h\to 0} \boldsymbol{v}(t)\, h \quad \rightarrow \text{Not usable for computation}$$

Replace infinitesimal $\lim_{h\to 0} h$ with finite T_s and only calculate for integer multiples k of T_s:

$$t = k\, T_s$$

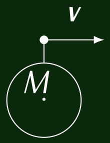
$$\boldsymbol{x}\left(\frac{t+T_s}{T_s}\right) = \boldsymbol{x}\left(\frac{t}{T_s}\right) + \boldsymbol{v}\left(\frac{t}{T_s}\right)\, T_s$$

$$\boldsymbol{x}(k+1) = \boldsymbol{x}(k) + \boldsymbol{v}(k)\, T_s$$

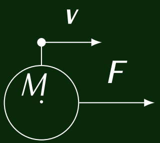
Keep T_s small

Building Blocks Mechanics

Building Blocks Mechanics

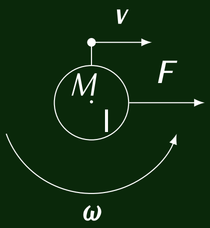


Building Blocks Mechanics



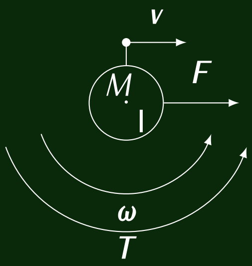
$$F = M\frac{\mathrm{d}v}{\mathrm{d}t}$$

Second law of motion



$$F = M \frac{\mathrm{d}v}{\mathrm{d}t}$$

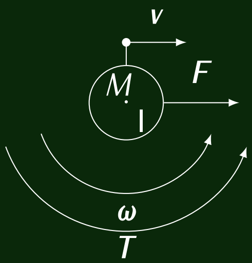
Second law of motion



Building Blocks Mechanics

$$F = M\,\frac{\mathrm{d}v}{\mathrm{d}t} \qquad T = \mathrm{I}\,\frac{\mathrm{d}\omega}{\mathrm{d}t}$$

Second law of motion



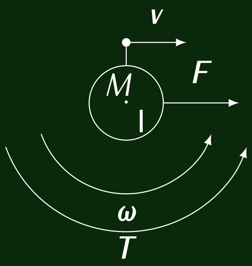
$$F = M\,\frac{\mathrm{d}v}{\mathrm{d}t} \qquad T = \mathrm{I}\,\frac{\mathrm{d}\omega}{\mathrm{d}t}$$

Second law of motion

$$F = M\,g$$

Weight

Building Blocks Mechanics

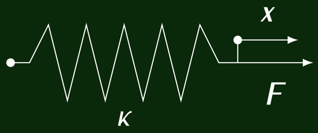


$$F = M \frac{\mathrm{d}v}{\mathrm{d}t} \qquad T = I \frac{\mathrm{d}\omega}{\mathrm{d}t}$$

Second law of motion

$$F = M\, g$$

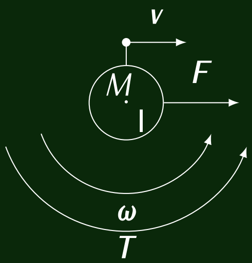
Weight



$$F = -\kappa \,(x - x_0)$$

Spring force

Building Blocks Mechanics

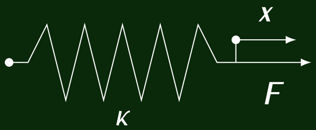


$$F = M \frac{\mathrm{d}v}{\mathrm{d}t} \qquad T = \mathrm{I} \frac{\mathrm{d}\omega}{\mathrm{d}t}$$

Second law of motion

$$F = M\,g$$

Weight

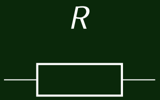


$$F = -\kappa\,(x - x_0)$$

Spring force

$$F = -b\,v$$

Viscous damping



$$V = R\,i$$

Resistor

Building Blocks Electric

R

$$V = R\,i$$

Resistor

L

$$V = L\,\frac{\mathrm{d}i}{\mathrm{d}t}$$

Inductance

Building Blocks Electric

R

$$V = R\, i$$

Resistor

L

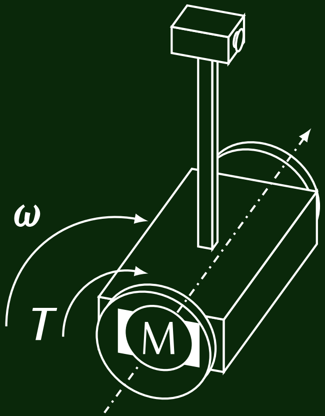
$$V = L\,\frac{\mathrm{d}i}{\mathrm{d}t}$$

Inductance

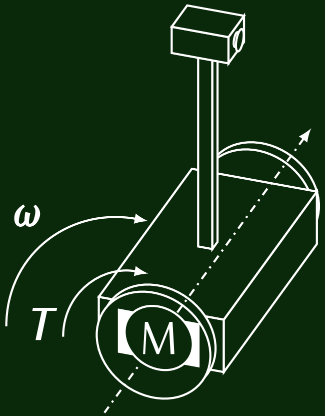
C

$$i = C\,\frac{\mathrm{d}V}{\mathrm{d}t}$$

Capacitor



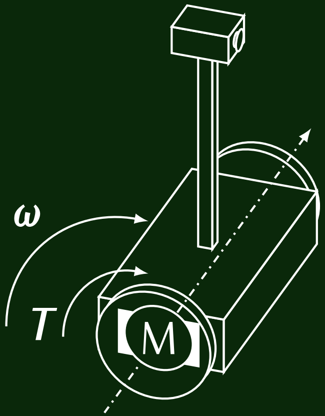
ω
T
M



$$T = K_t\, i$$

Motor

Building Blocks Electromechanics



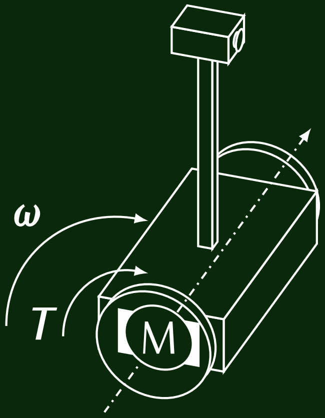
$$T = K_t \, i$$

Motor

$$V = K_v \, \omega$$

Generator

Building Blocks Electromechanics



$$T = K_t\, i$$

Motor

$$V = K_v\, \omega$$

Generator

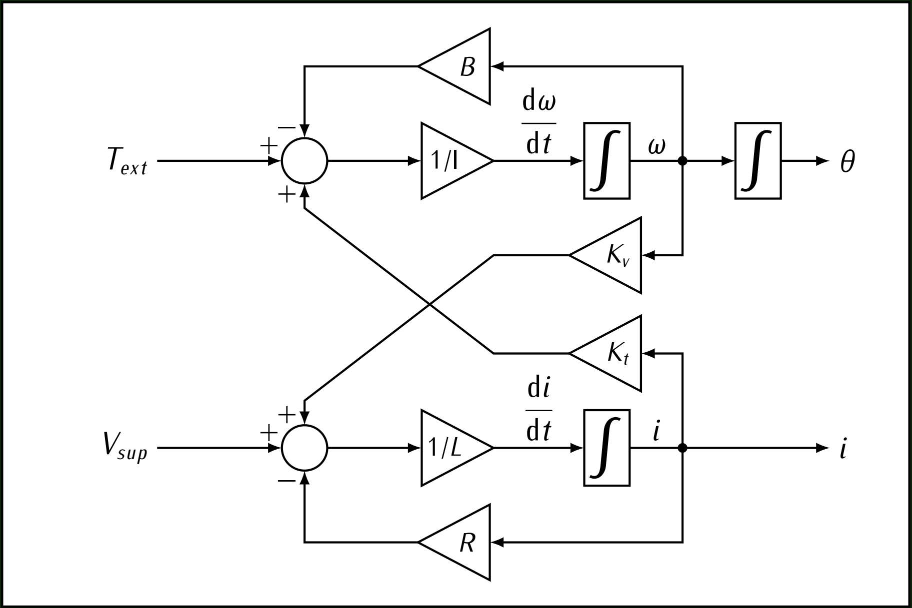
$$V = R\, i + L\,\frac{\mathrm{d}i}{\mathrm{d}t} + K_v\, \omega$$

$$I\,\frac{\mathrm{d}\omega}{\mathrm{d}t} = K_t\, i - b\, \omega$$

Electric motor

Tips & Tricks

1. Sampling Period (T_s): min. 100x faster than system time constant
2. Block Diagram: helps to keep overview
3. Adapt the model to your needs: different questions might need different models
4. Specialized Tools (SciPy, OpenModelica/OMEdit, Scilab/XCos):
    - for complex models or as reference
    - better differential equation solving (BDF, Runge-Kutta, etc.)
    - efficient through variable time-step
    - nice data logging and visualization tools



Motor Model Block Diagram

Background & Further Reading (Wikipedia)

- Scientific modeling
- Ordinary differential equation
- Numerical methods for ordinary differential equations
  - Euler Method
  - Runge-Kutta
  - Backward differentiation formula (BDF)
- Discrete time and continuous time
- State-space representation