autorob.org

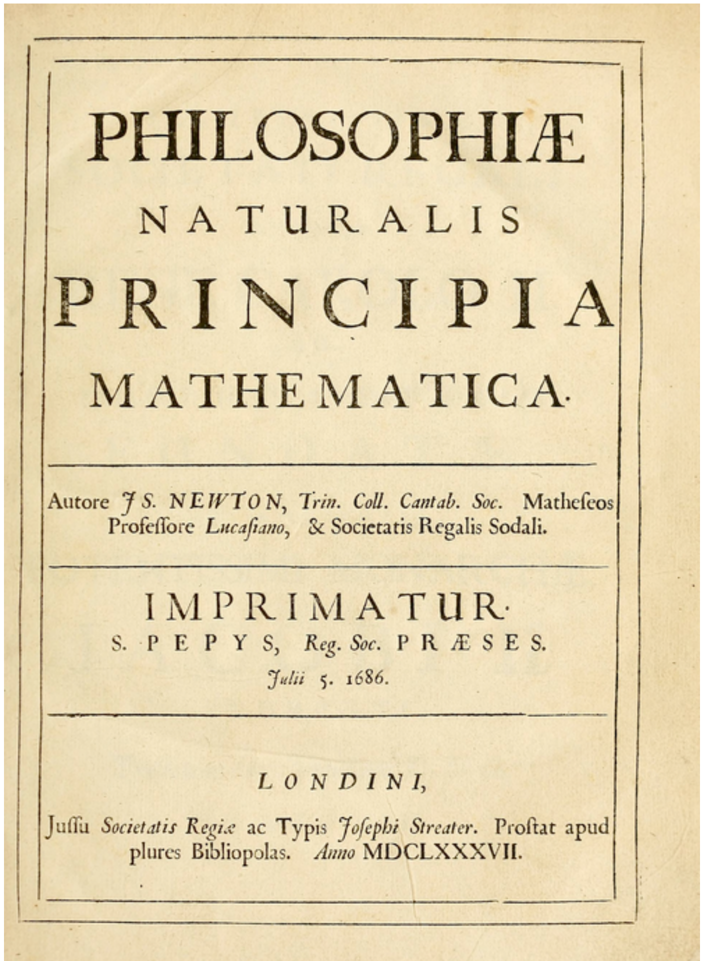
Dynamics and Numerical Integration

EECS 367
Intro. to Autonomous Robotics

ROB 511
Robot Operating Systems

Fall 2020

1687


PHILOSOPHIÆ
NATURALIS
PRINCIPIA
MATHEMATICA.

Autore J S. NEWTON, Trin. Coll. Cantab. Soc. Mathefeos Profeffore Lucafiano, & Societatis Regalis Sodali.

IMPRIMATUR.
S. PEPYS, Reg. Soc. PRÆSES.
Julii 5. 1686.

LONDINI,

Juffu Societatis Regiæ ac Typis Jofephi Streater. Proftat apud plures Bibliopolas. Anno MDCLXXXVII.

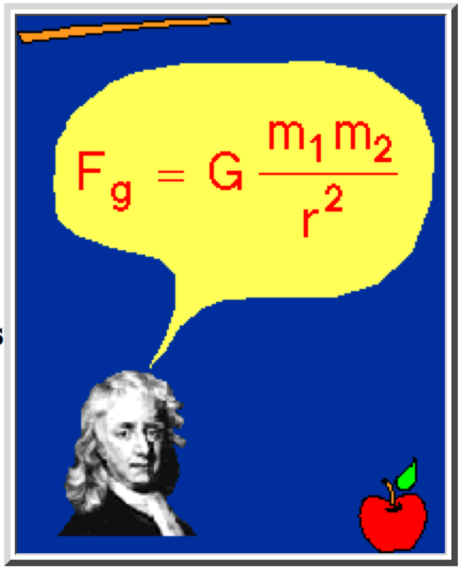
2012

http://drawception.com/viewgame/azF19y6c8s/edible-science/

There is a popular story that Newton was sitting under an apple tree, an apple fell on his head, and he suddenly thought of the Universal Law of Gravitation. As in all such legends, this is almost certainly not true in its details, but the story contains elements of what actually happened.

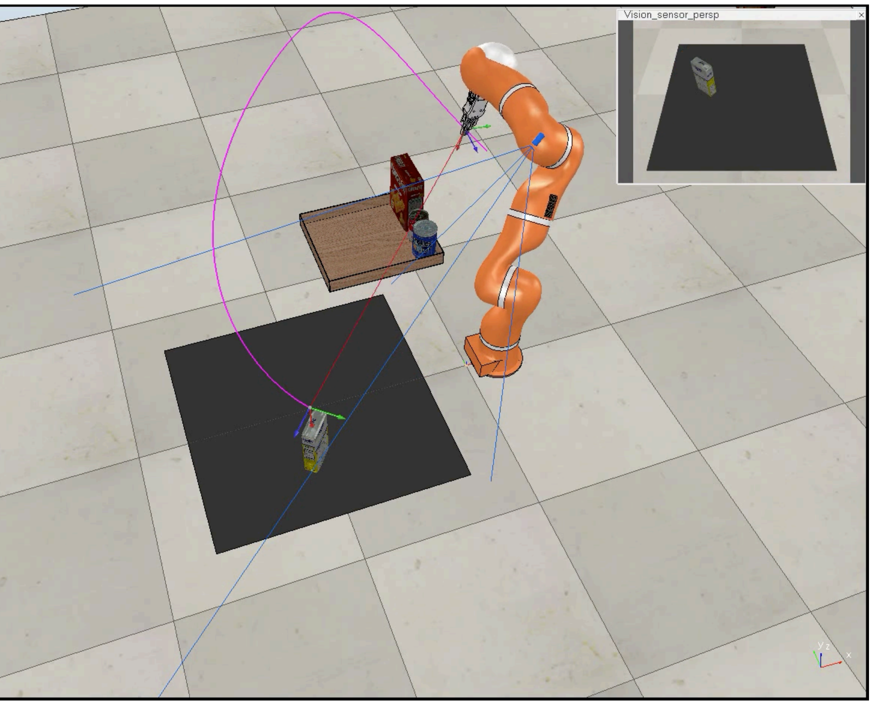
What Really Happened with the Apple?

Probably the more correct version of the story is that Newton, upon observing an apple fall from a tree, began to think along the following lines: The apple is accelerated, since its velocity changes from zero as it is hanging on the tree and moves toward the ground. Thus, by Newton's 2nd Law there must be a force that acts on the apple to cause this acceleration. Let's call this force "gravity", and the associated acceleration the "accleration due to gravity". Then imagine the apple tree is twice as high. Again, we expect the apple to be accelerated toward the ground, so this suggests that this force that we call gravity reaches to the top of the tallest apple tree.



http://csep10.phys.utk.edu/astr161/lect/history/newtongrav.html

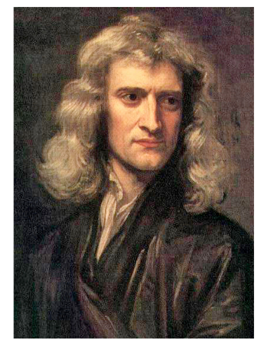
Dynamical Simulation



KUKA Lightweight Arm
in V-REP simulator

Rigid Body Equations of Motion enable simulation of robots on computers

$$\mathbf{F} = m\mathbf{a}_{\mathrm{cm}}$$



Issac Newton
(1643-1727)

$$\mathbf{M} = \mathbf{r}_{\mathrm{cm}} \times \mathbf{a}_{\mathrm{cm}} m + I\boldsymbol{\alpha}$$

Leonhard Euler
(1707-1783)

Users

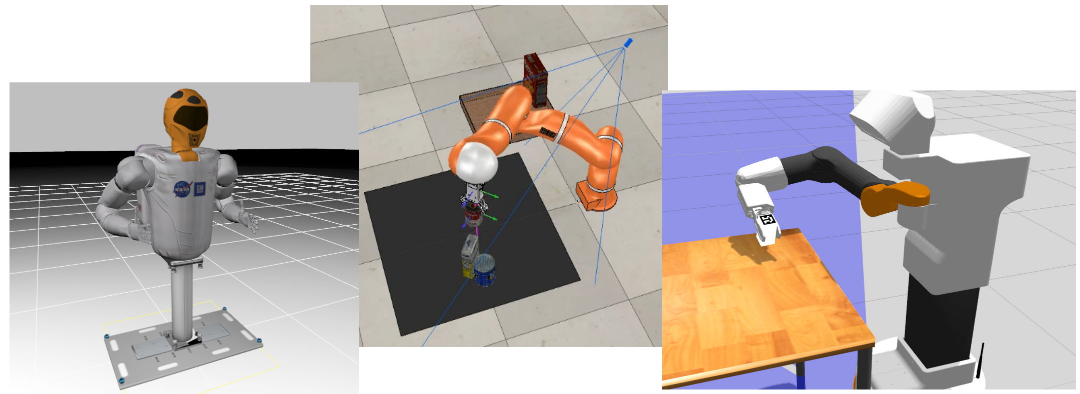
Robot Applications

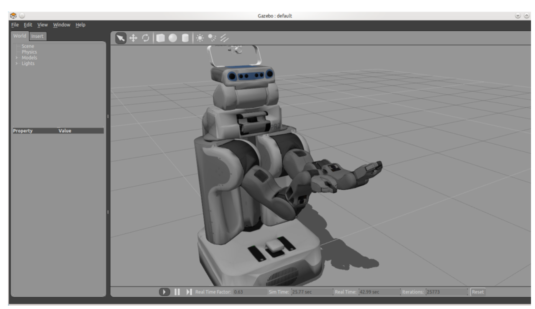
Robot Operating System

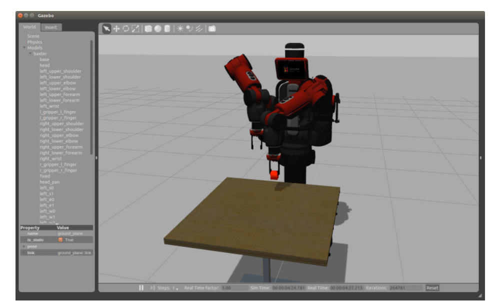
Operating System

Dynamical Simulation

$$\mathbf{F} = m\mathbf{a}_{\mathrm{cm}} \qquad \mathbf{M} = \mathbf{r}_{\mathrm{cm}} \times \mathbf{a}_{\mathrm{cm}} m + I\alpha$$







Users

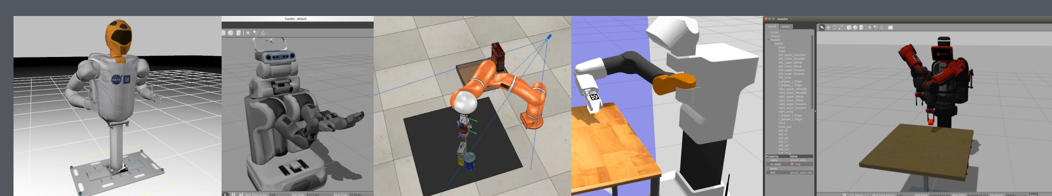
Robot Applications

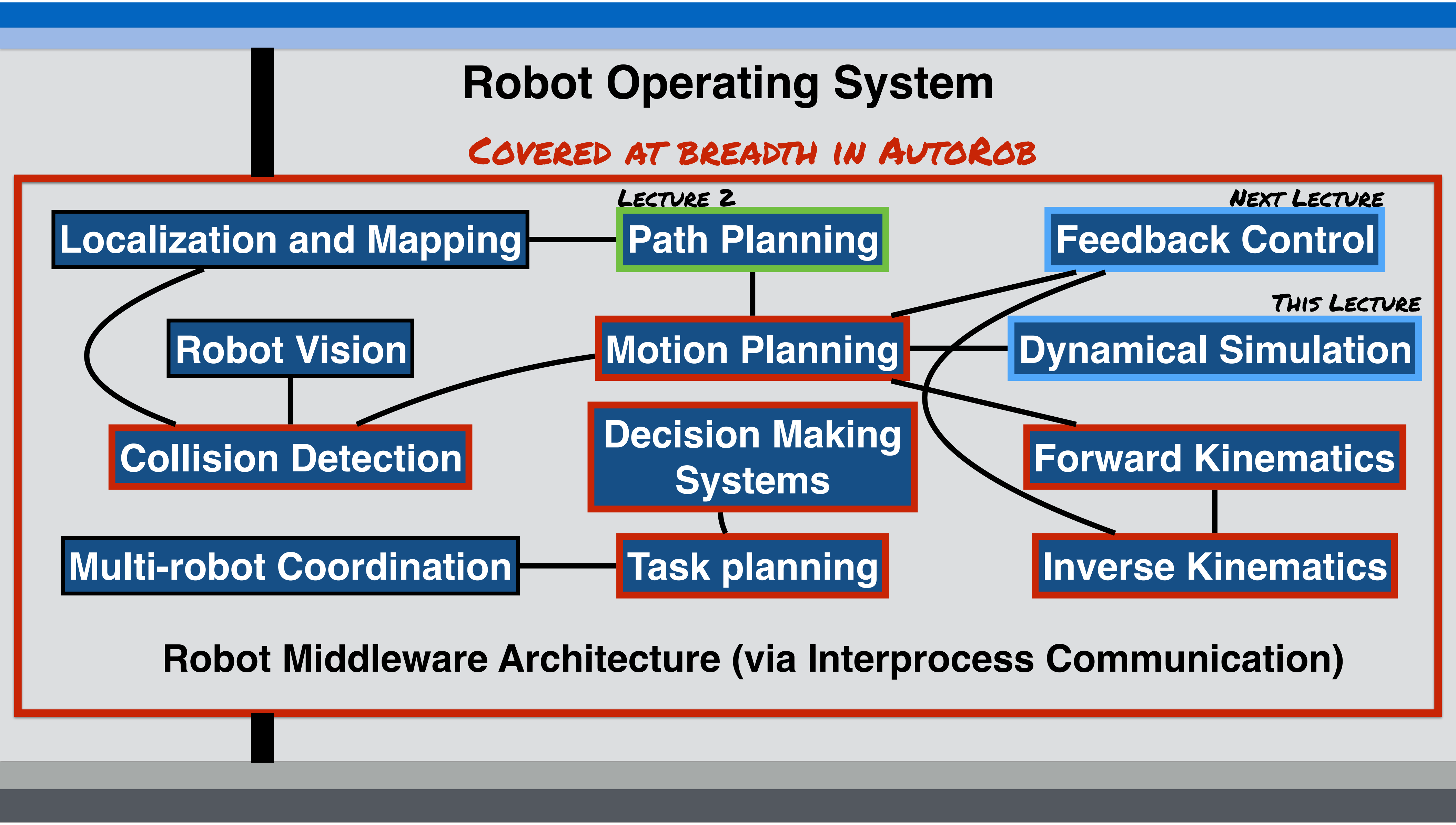
Robot Operating System

Operating System

Dynamical Simulation

Simulated Hardware




Robot Operating System
Covered at breadth in AutoRob
Lecture 2
Next Lecture
This Lecture
Localization and Mapping
Path Planning
Feedback Control
Robot Vision
Motion Planning
Dynamical Simulation
Collision Detection
Decision Making Systems
Forward Kinematics
Multi-robot Coordination
Task planning
Inverse Kinematics
Robot Middleware Architecture (via Interprocess Communication)

physical laws describing the motion of bodies under the action of a system of forces.

Classical mechanics

From Wikipedia, the free encyclopedia

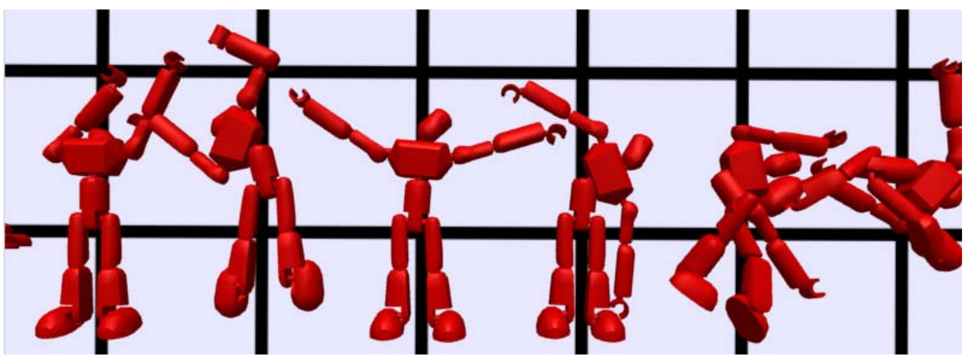
In physics, **classical mechanics** and quantum mechanics are the two major sub-fields of mechanics. Classical mechanics is concerned with the set of physical laws describing the motion of bodies under the action of a system of forces. The study of the motion of bodies is an ancient one, making classical mechanics one of the oldest and largest subjects in science, engineering and technology. It is also widely known as **Newtonian mechanics**.

Classical mechanics

physical laws describing the motion of bodies under the action of a system of forces

Kinematics

kin•e•mat•ics | ˌkinəˈmatiks |
plural noun [usu. treated as sing.]
the branch of mechanics concerned with the motion of objects without reference to the forces that cause the motion. Compare with



Dynamics

dy•nam•ics | dīˈnamiks |
plural noun
1 [treated as sing.] the branch of mechanics concerned with the motion of bodies under the action of forces. Compare with STATICS.
• [usu. with modifier] the branch of any science in which forces or changes are considered: *chemical dynamics*.

$$F = ma$$

Classical mechanics

physical laws describing the motion of bodies under the action of a system of forces

Kinematics

"the geometry of motion"

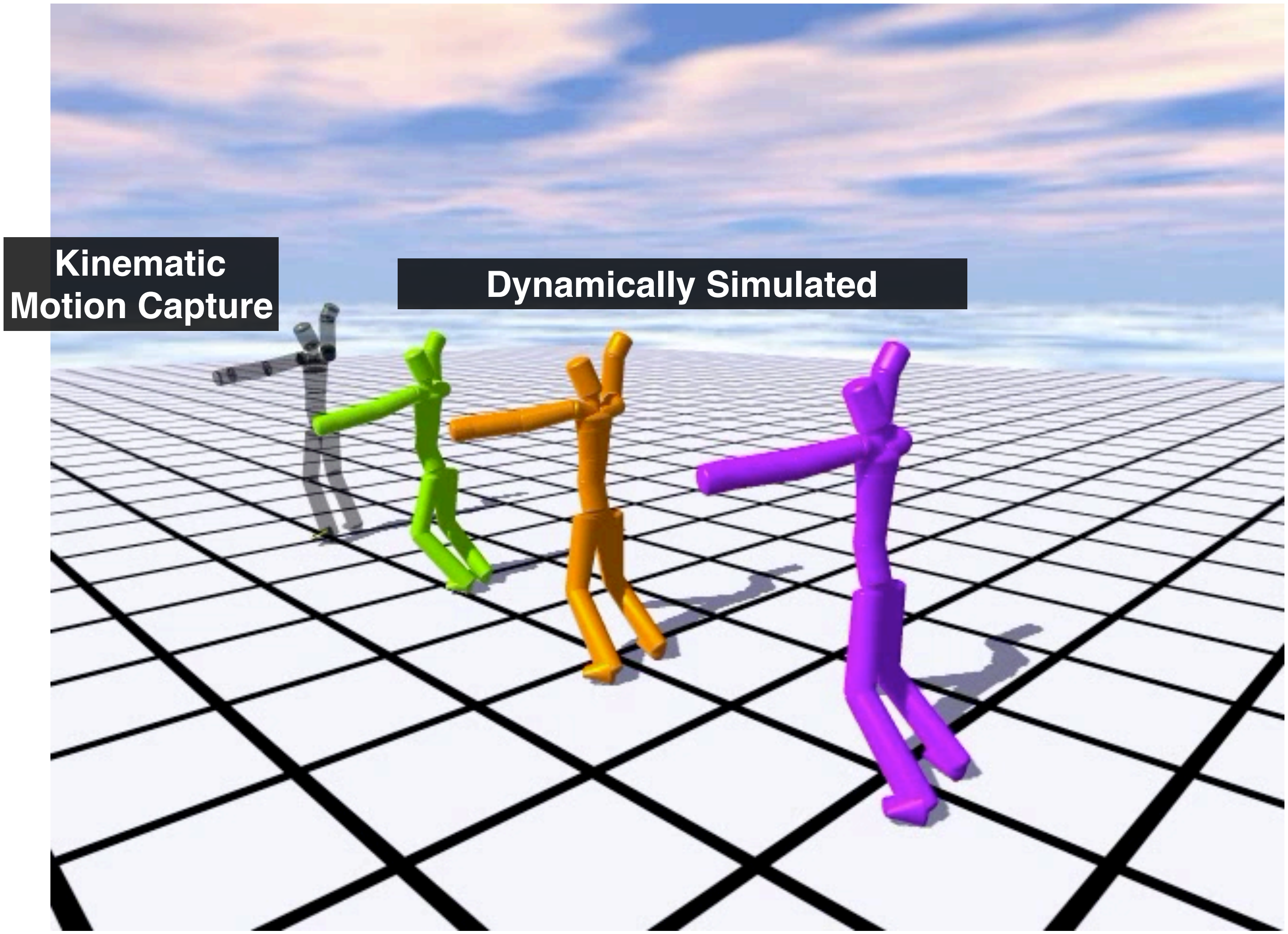
expresses range of possible motion
(or states of the robot)

Dynamics

"physical motion over time"

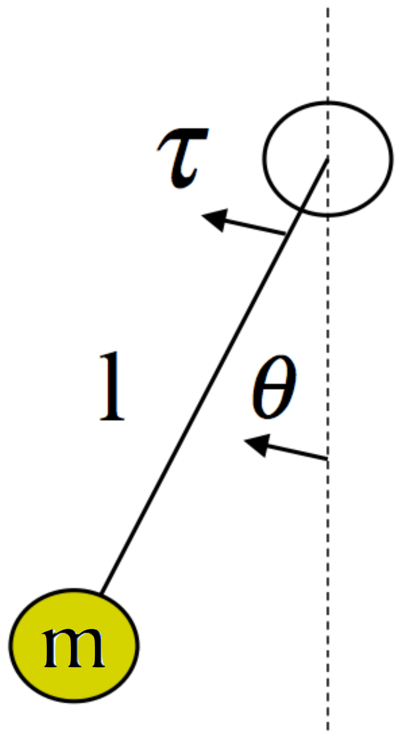
expresses motion over time with
respect to Newton's laws
(evolution of state over time)

Dynamo - Wrotek, McGuire, Jenkins 2006 - https://youtu.be/5nuhY6kJdv4

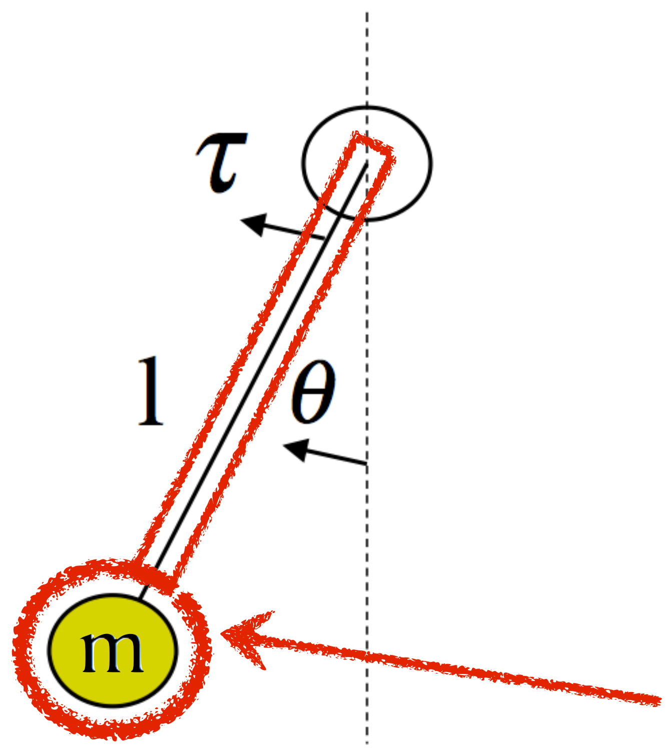


Let’s start with a simple example

Example: Motorized Pendulum

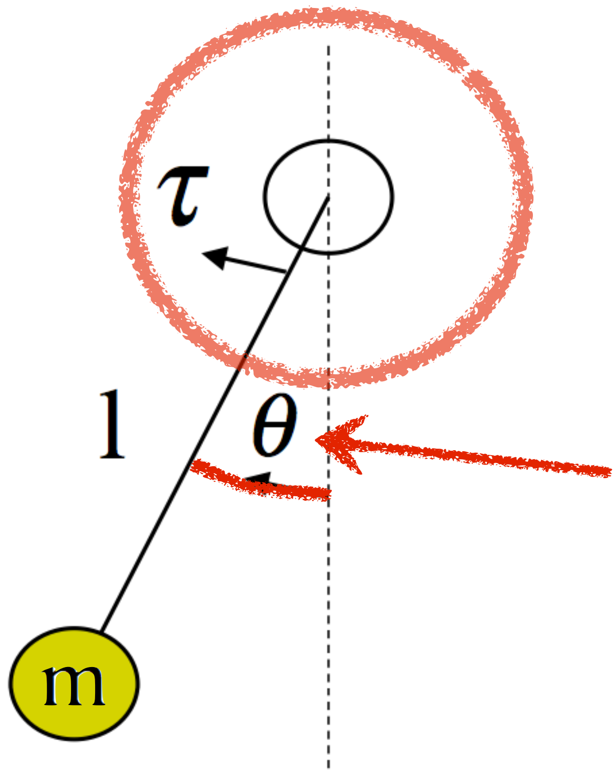


Example: Motorized Pendulum



Pendulum of length *l* with point mass *m*

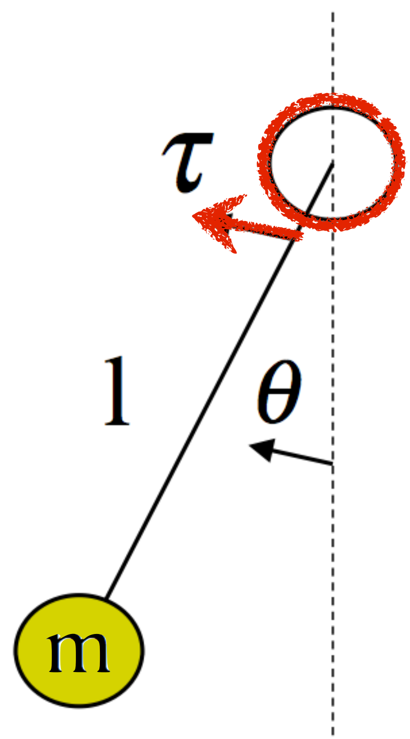
Example: Motorized Pendulum



Angle expresses pendulum range of motion

Pendulum of length *l* with point mass *m*

Example: Motorized Pendulum



Motor produces torque (angular force)

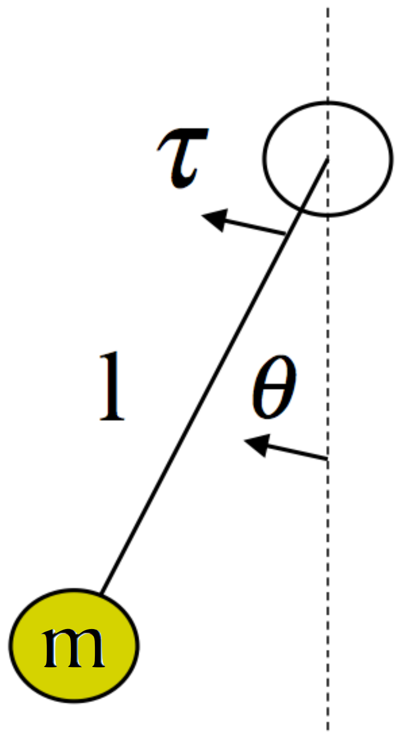
Angle expresses pendulum range of motion

Pendulum of length *l* with point mass *m*

Example: Motorized Pendulum

Equation of motion
(with rotational inertia *I*)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$



Motor produces torque
(angular force)

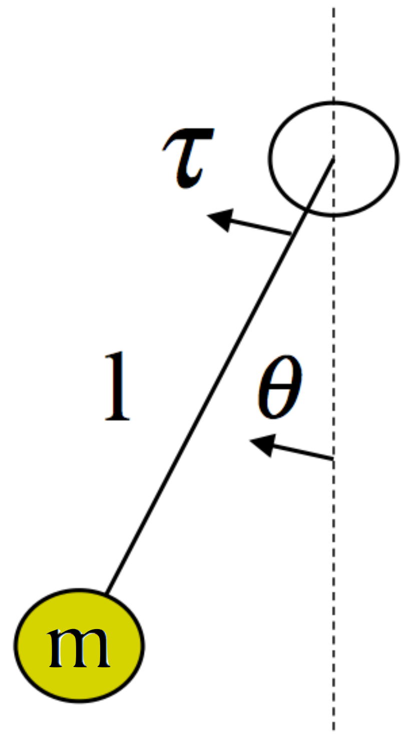
Angle expresses pendulum
range of motion

Pendulum of length *l* with
point mass *m*

Dynamics

Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$



Controls

Motor produces torque
(angular force)

State (or Configuration)

Angle expresses pendulum
range of motion

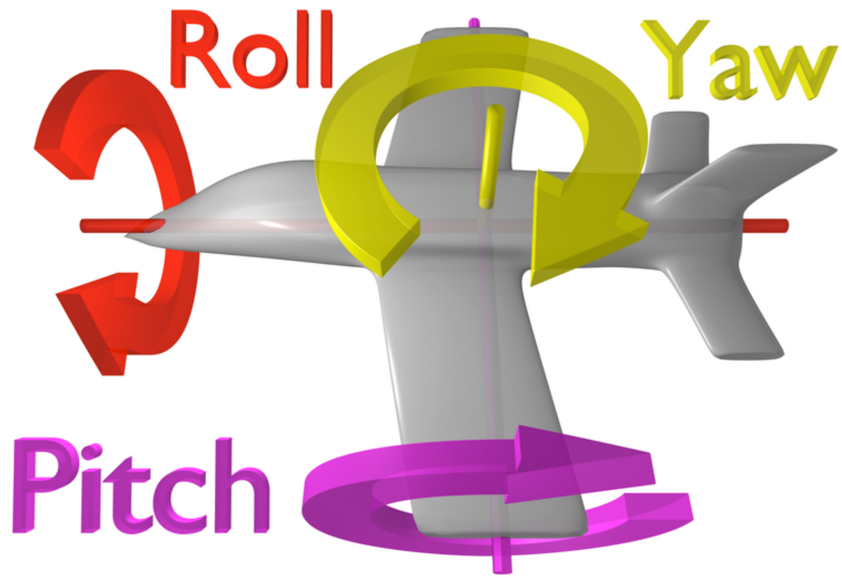
System

Pendulum of length *l* with
point mass *m*

Defining State

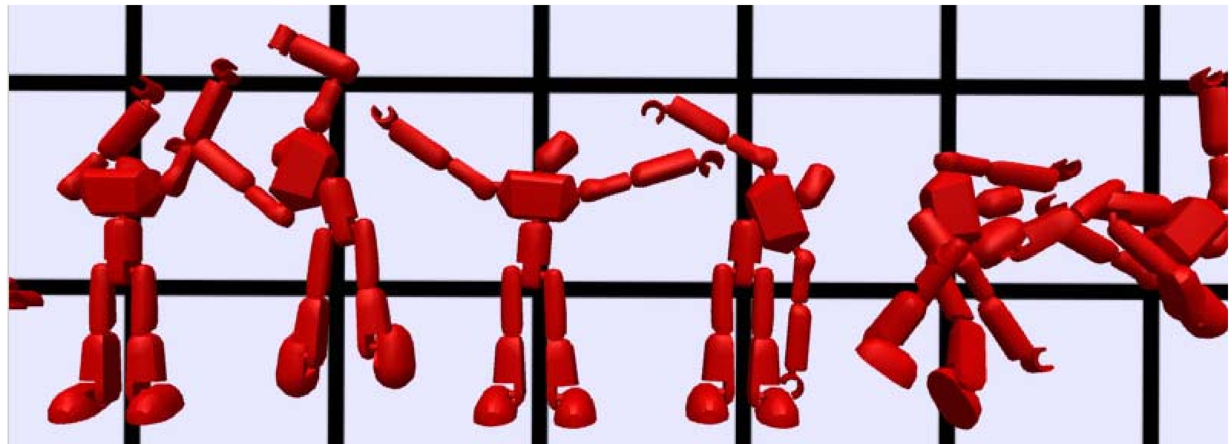
- State comprised of <u>degrees-of-freedom (DOFs)</u>
- DOFs describe translational and rotational axes for the motion about robot joints

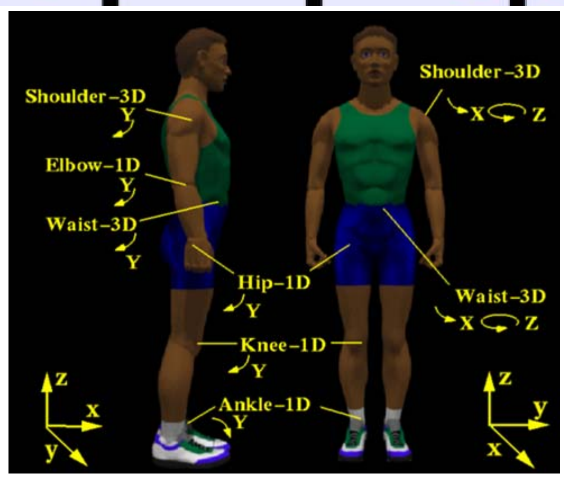
- How many DOFs in the example pendulum?
- Airplane DOFs?

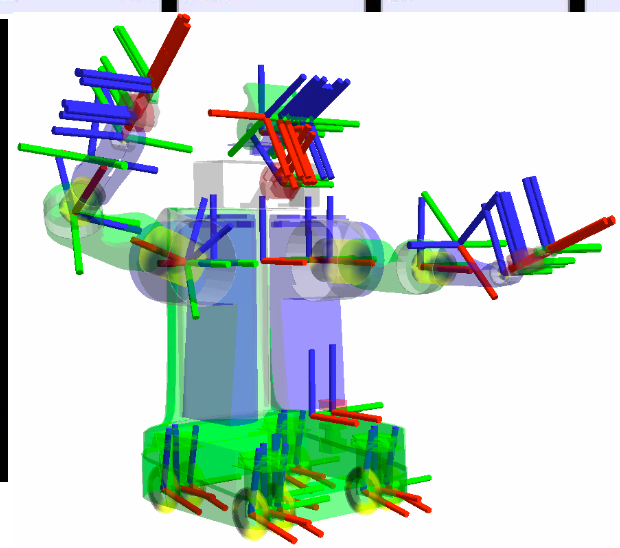


Defining State

- State comprised of <u>degrees-of-freedom (DOFs)</u>
- DOFs describe translational and rotational axes for the motion about robot joints
- <u>Humanoid DOFs?</u>
  - joint angles
  - global positioning



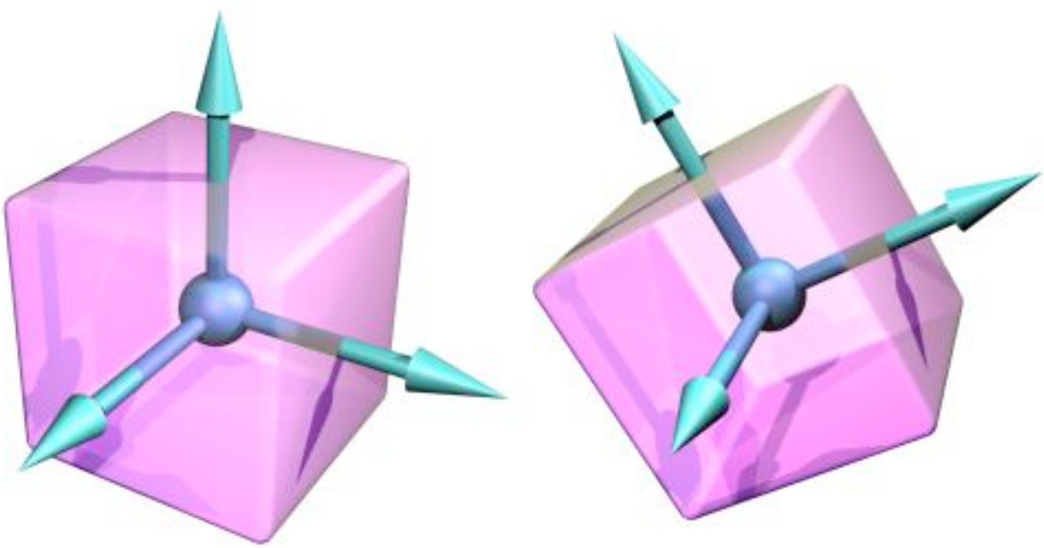




DOFs and Coordinate Spaces

- Each body has its own coordinate system

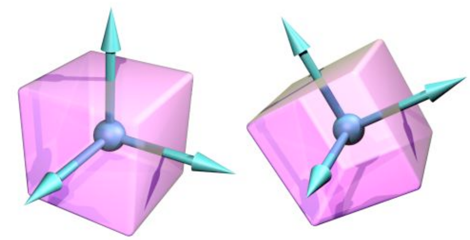
LINK FRAME



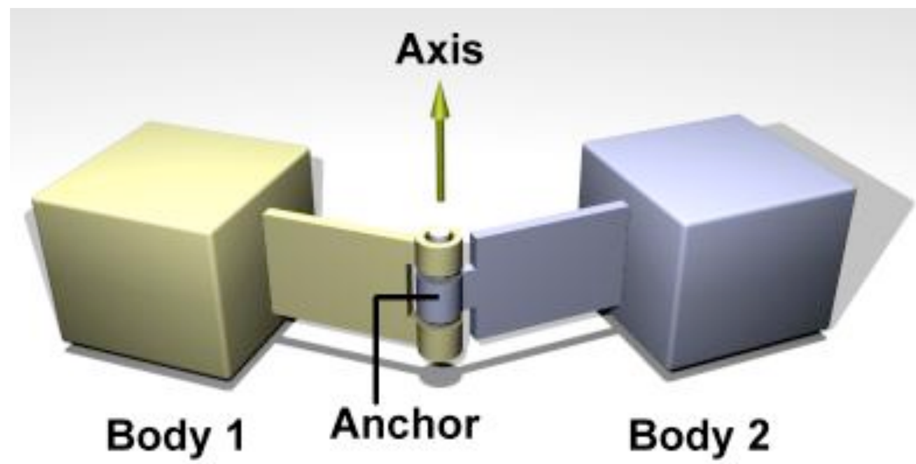
DOFs and Coordinate Spaces

- Each <u>body</u> has its own <u>coordinate system</u>

LINK FRAME



- Joints connect two links (rigid bodies)
  - Hinge (1 rotational DOF)

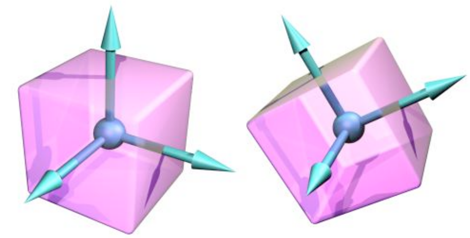


HINGE JOINT

DOFs and Coordinate Spaces

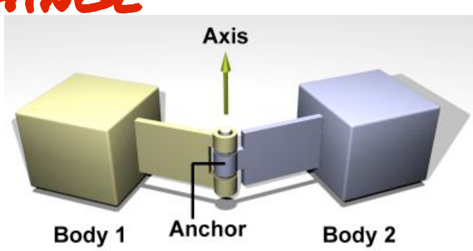
- Each body has its own coordinate system

LINK FRAME

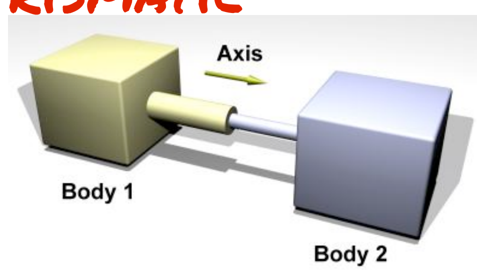


- Joints connect two links (rigid bodies)
  - Hinge (1 rotational DOF)
  - Prismatic (1 translational DOF)
  - Ball-socket (3 DOFs)

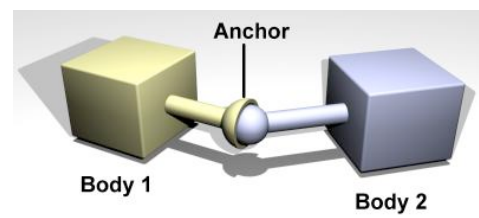
HINGE



PRISMATIC



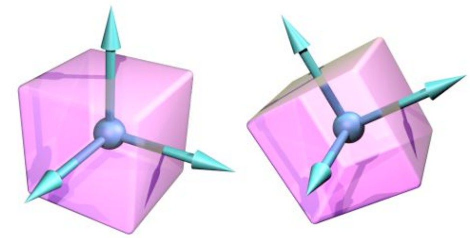
BALL



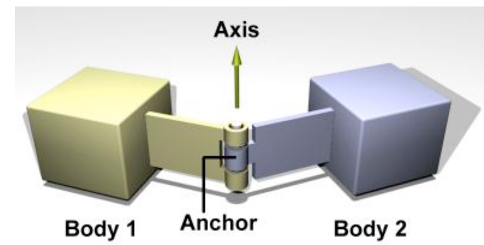
DOFs and Coordinate Spaces

- Each <u>body</u> has its own <u>coordinate system</u>

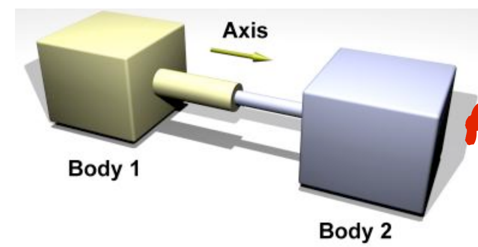
LINK FRAME



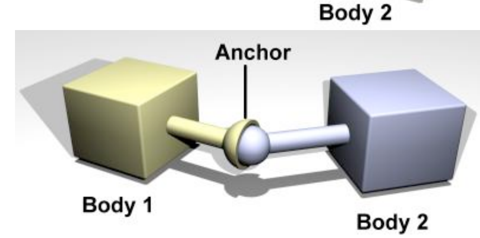
- Joints connect two links (rigid bodies)
  - Hinge (1 rotational DOF)
  - Prismatic (1 translational DOF)
  - Ball-socket (3 DOFs)
- A motor exerts force along a DOF axis



HINGE



PRISMATIC

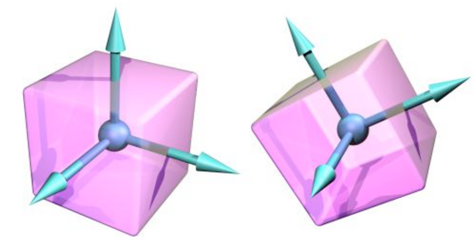


BALL

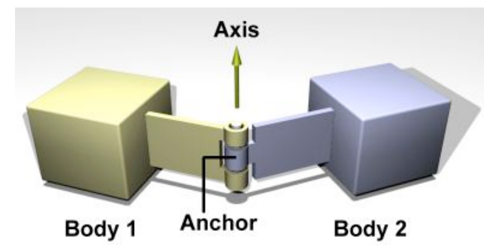
DOFs and Coordinate Spaces

- Each <u>body</u> has its own <u>coordinate system</u>

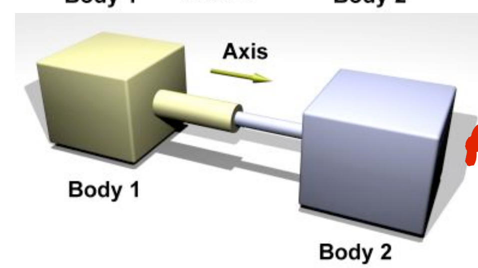
LINK FRAME



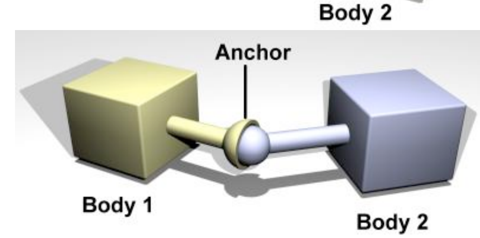
- Joints connect two links (rigid bodies)
  - Hinge (1 rotational DOF)
  - Prismatic (1 translational DOF)
  - Ball-socket (3 DOFs)
- A motor exerts force along a DOF axis
- Linear transformations used to relate coordinate frames of robot links and joints
- Spatial geometry attached to each link, but does not affect the body's coordinate frame



HINGE



PRISMATIC



BALL

Robotic machines are comprised of
N joints and *N+1* links

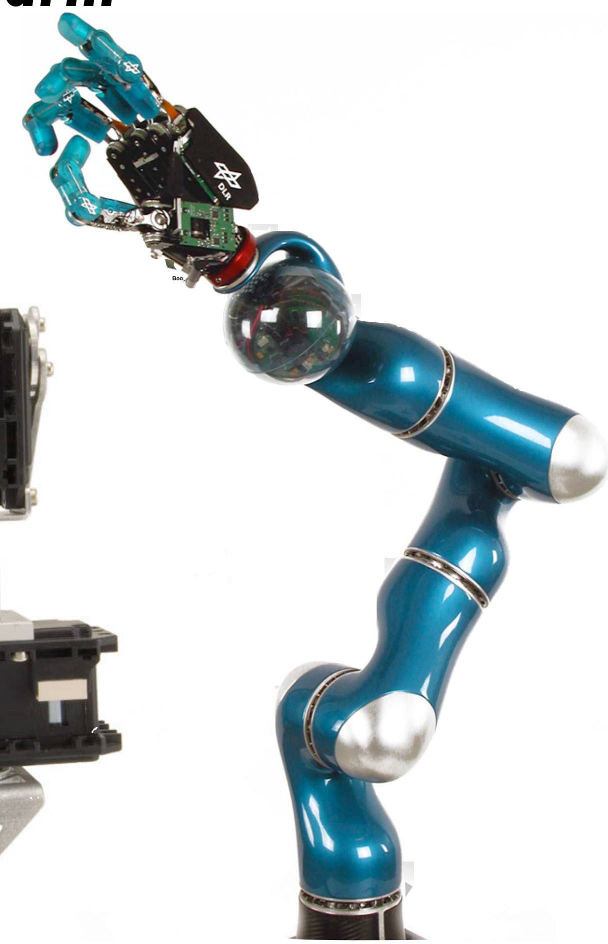
Joints and links form a tree hierarchy of articulated motion as an "open chain"

The "base" is the root link of this hierarchy

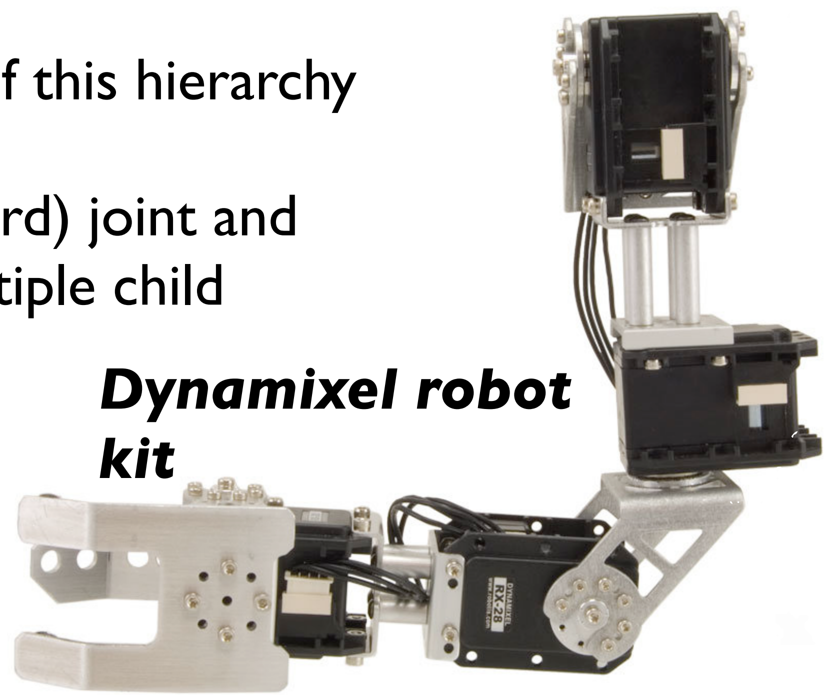
A link has one parent (inboard) joint and potentially zero, one, or multiple child (outboard) joints

A "serial chain" is a robot where every link has only one child joint

DLR Lightweight arm



Dynamixel robot kit



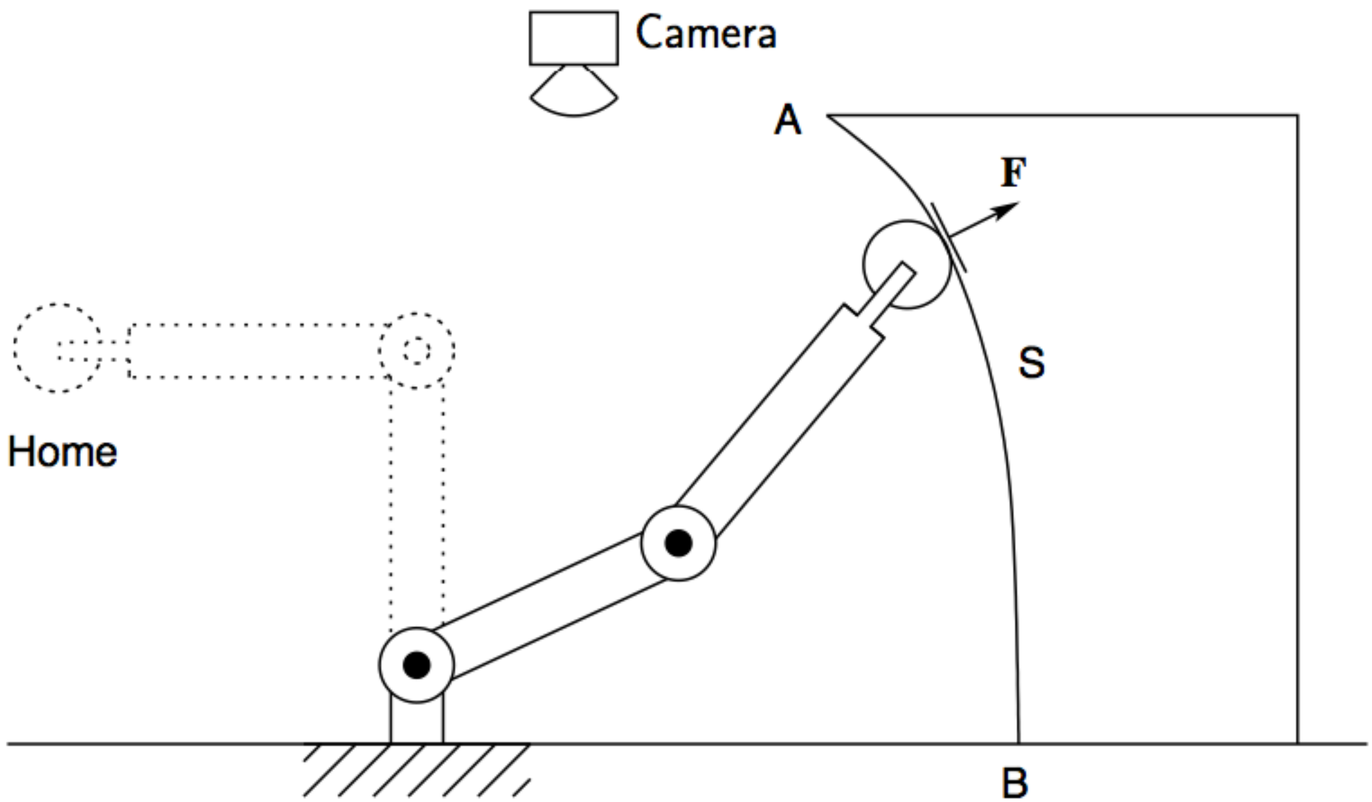
Consider some examples of robots…

Planar Arm

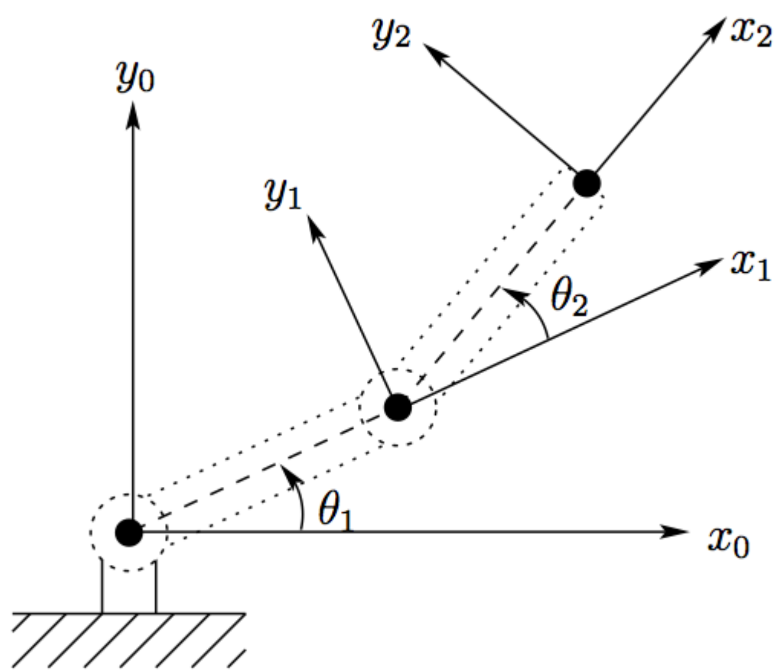
https://
www.nowhereelse.fr/
aikon-project-robot-
artiste-dessin-
portraits-29397/

AIKON-II Project - https://youtu.be/bbdQbyff_Sk

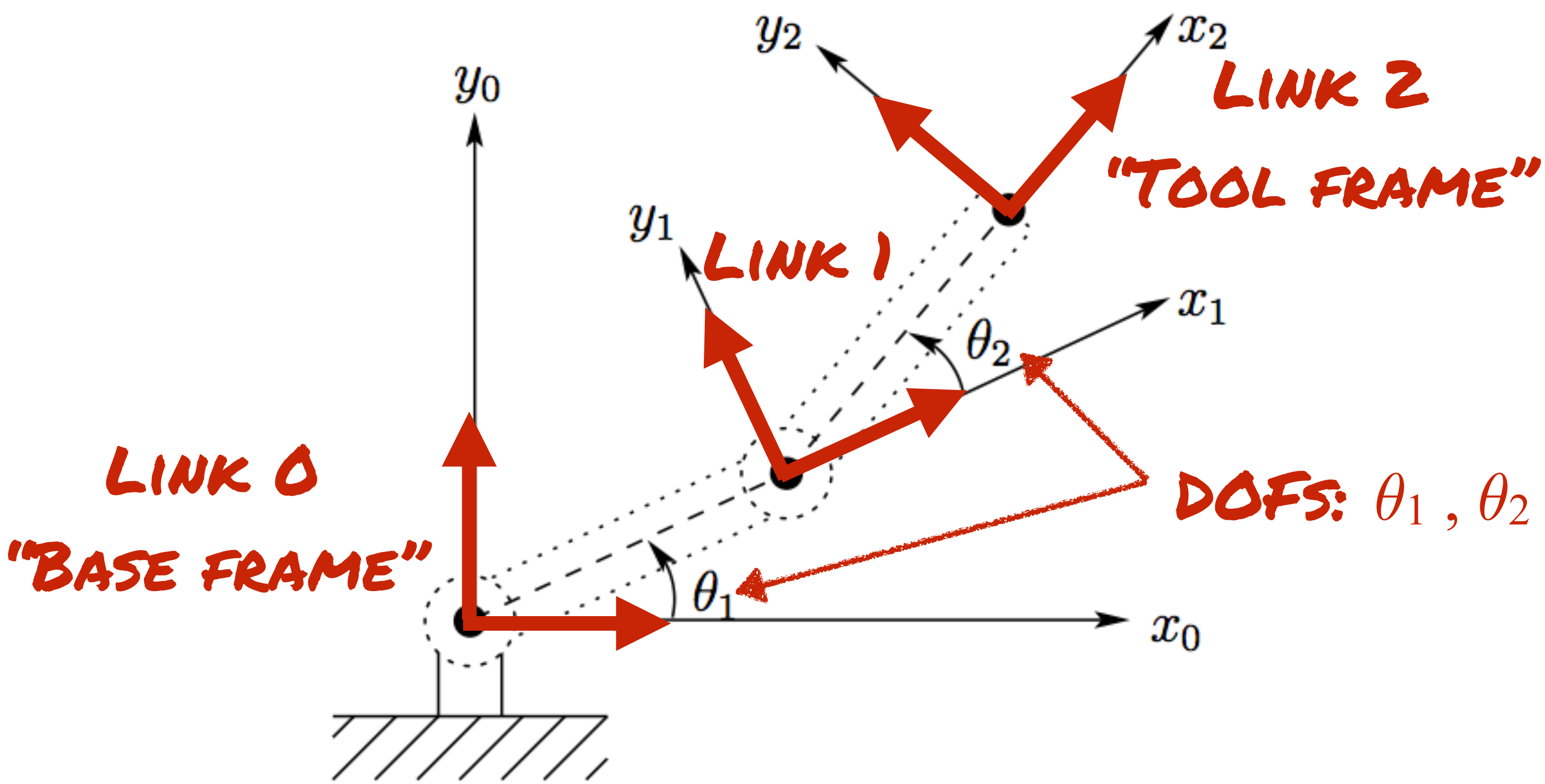
Planar 2-DOF 2-link Arm



Planar 2-DOF 2-link Arm

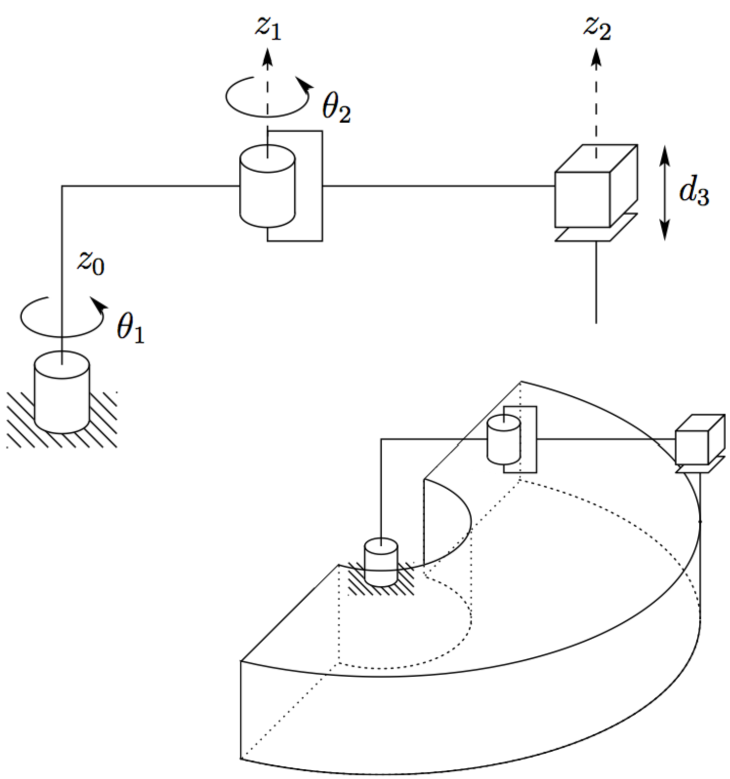


Planar 2-DOF 2-link Arm

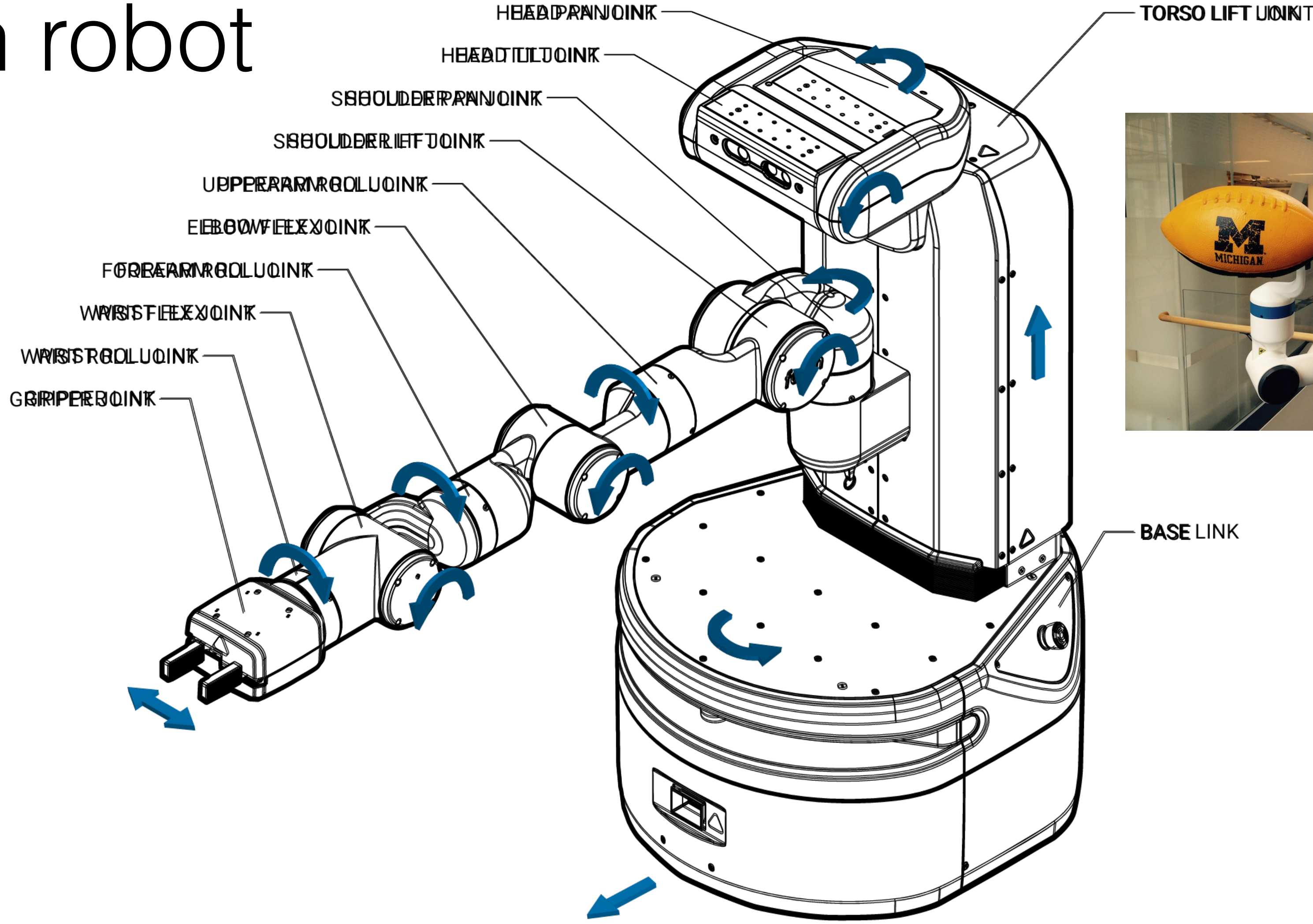


SCARA Arm

Selective Compliance Assembly Robot Arm



Fetch robot

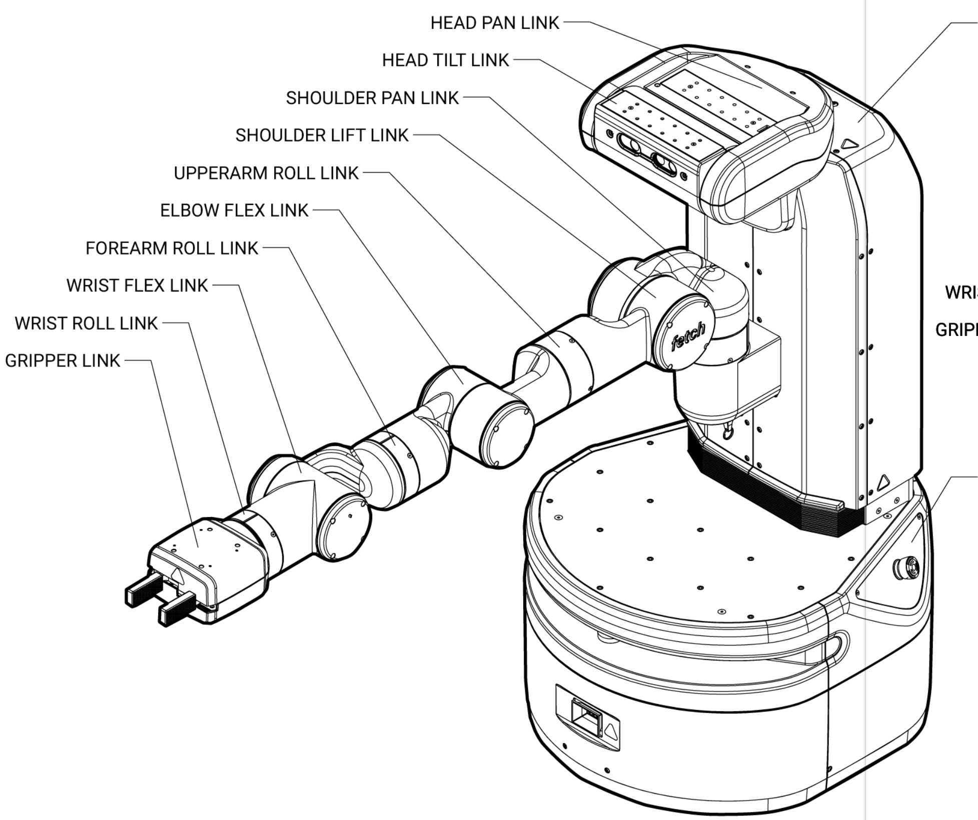


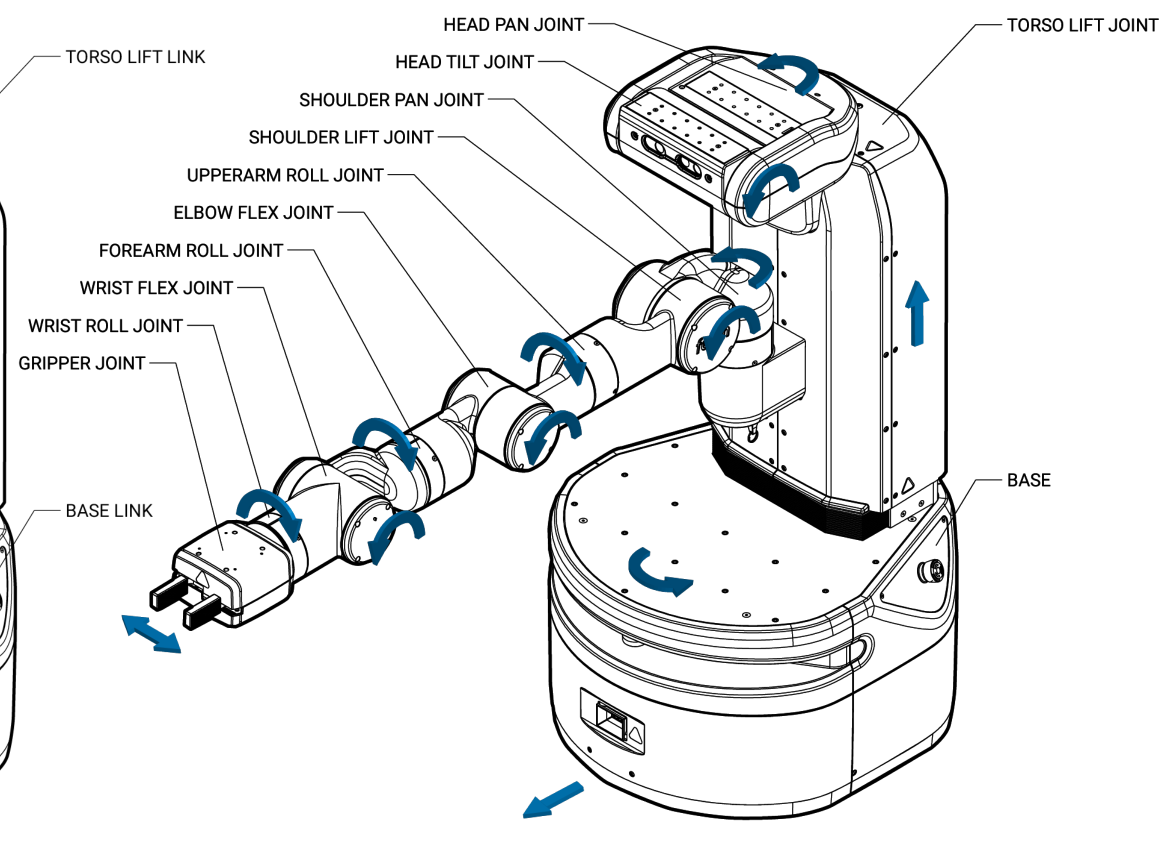
HEAD PAN JOINT
HEAD TILT JOINT
SHOULDER PAN JOINT
SHOULDER LIFT JOINT
UPPERARM ROLL JOINT
ELBOW FLEX JOINT
FOREARM ROLL JOINT
WRIST FLEX JOINT
WRIST ROLL JOINT
GRIPPER JOINT
TORSO LIFT JOINT
BASE LINK

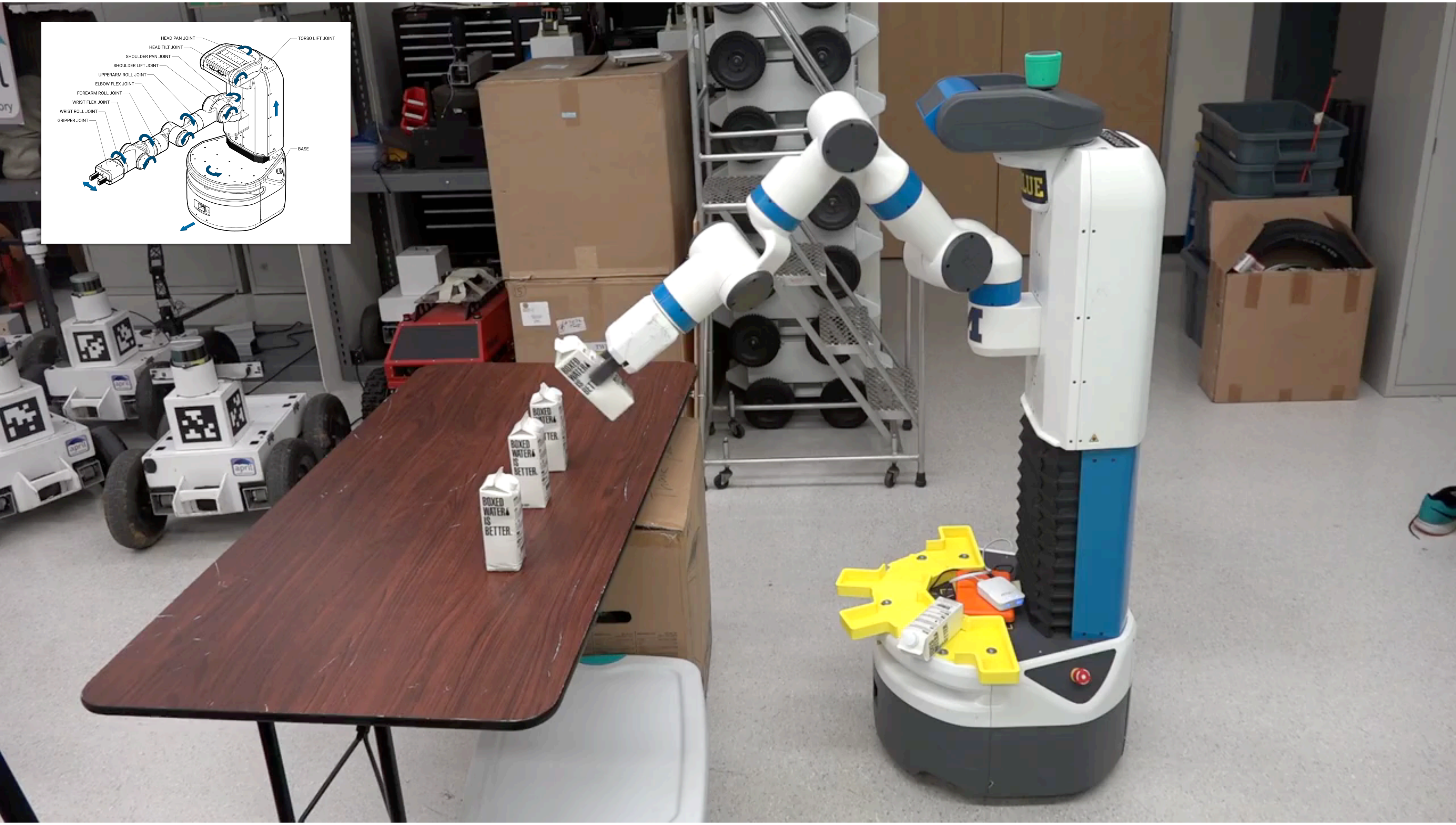
MICHIGAN
fetch

Fetch links

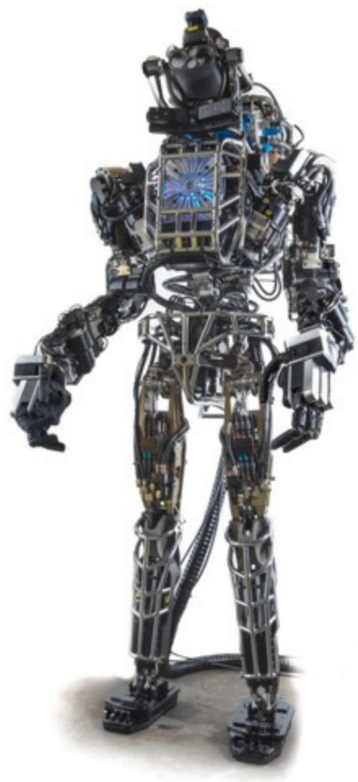
Fetch joints

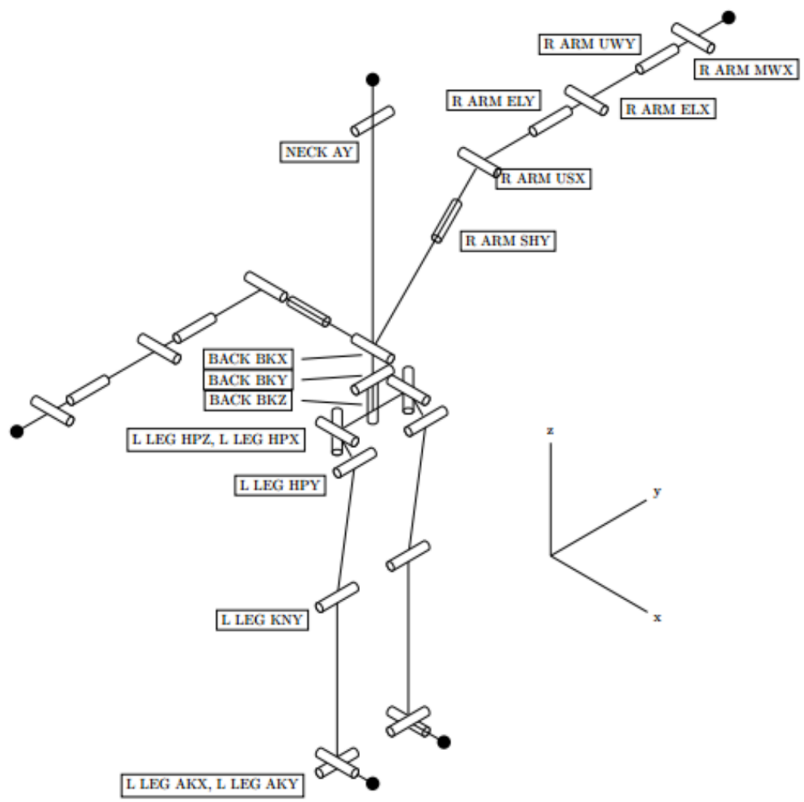





HEAD PAN JOINT
HEAD TILT JOINT
SHOULDER PAN JOINT
SHOULDER LIFT JOINT
UPPERARM ROLL JOINT
ELBOW FLEX JOINT
FOREARM ROLL JOINT
WRIST FLEX JOINT
WRIST ROLL JOINT
GRIPPER JOINT
TORSO LIFT JOINT
BASE
BOXED WATER IS BETTER.

Atlas robot





WPI

Physical Simulation (Gazebo)

Why simulate robots?

Why simulate robots?

- **Real robots are expensive**
  - Improper controllers can physically break robots
  - Inexpensive to experiment and test robot controllers in simulation
- **Predictive model of dynamics**
  - Necessary for some types of control (e.g., optimal control)

$$J = \Phi\left[\mathbf{x}(t_0), t_0, \mathbf{x}(t_f), t_f\right] + \boxed{\int_{t_0}^{t_f} \mathcal{L}\left[\mathbf{x}(t), \mathbf{u}(t), t\right] \mathrm{d}\,t}$$

Why simulate robots?

- **Robot simulation packages are readily available**

CoppeliaSim
from the creators of V-REP
Create. Compose. Simulate.
Any Robot.

GAZEBO

DART
Dynamic Animation and Robotics Toolkit

Webots
robot simulation

- **Built on physics engines that numerically integrate over dynamics**

SD/FAST
(THE OG)

vortex
studio

newton
DYNAMICS

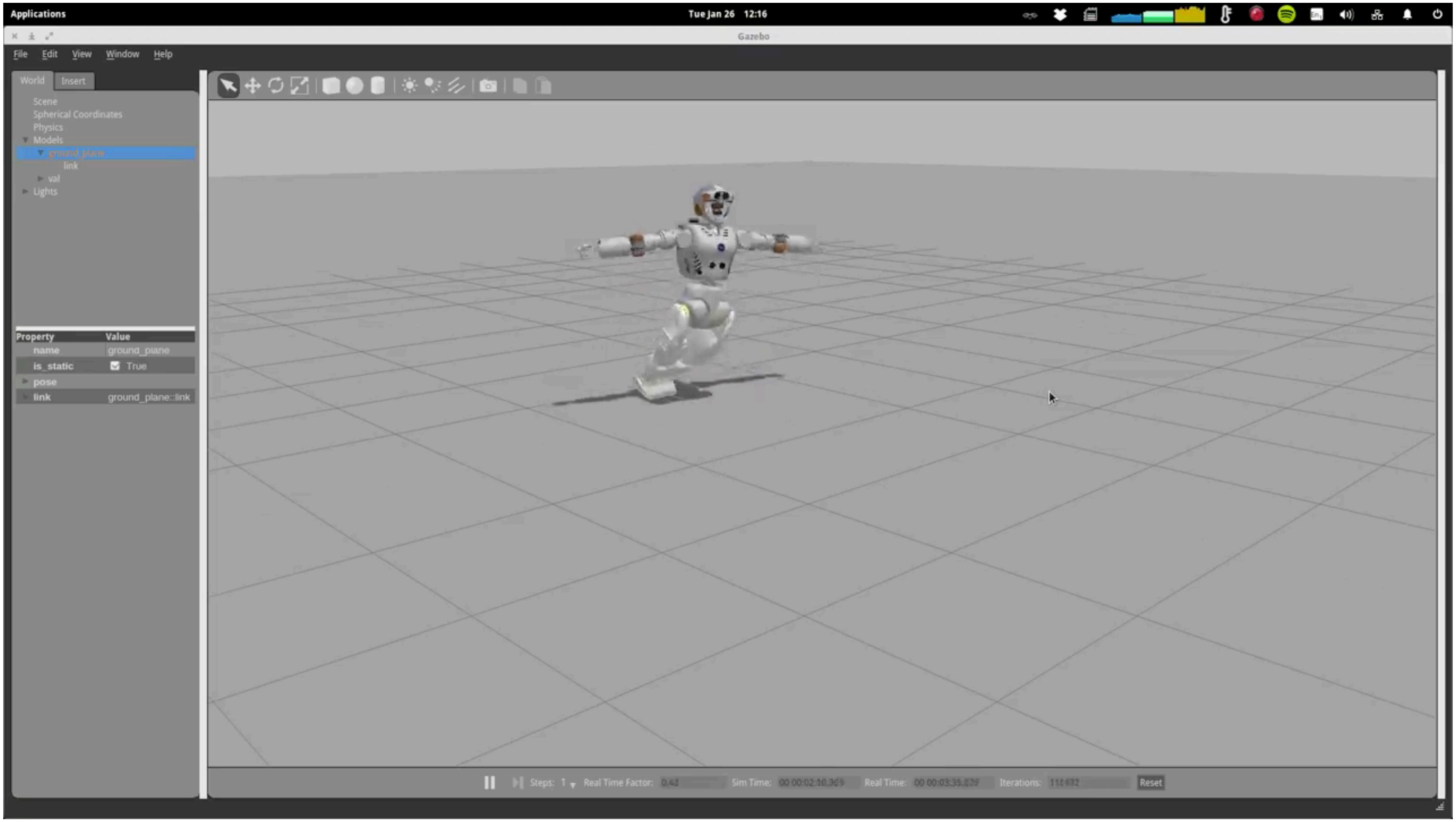
OPEN DYNAMICS ENGINE™

BULLET
PHYSICS LIBRARY

For example,
NASA Valkyrie…

Valkyrie for real - https://youtu.be/5Ee5u2ekE8c

NASA Valkyrie…
development in simulation



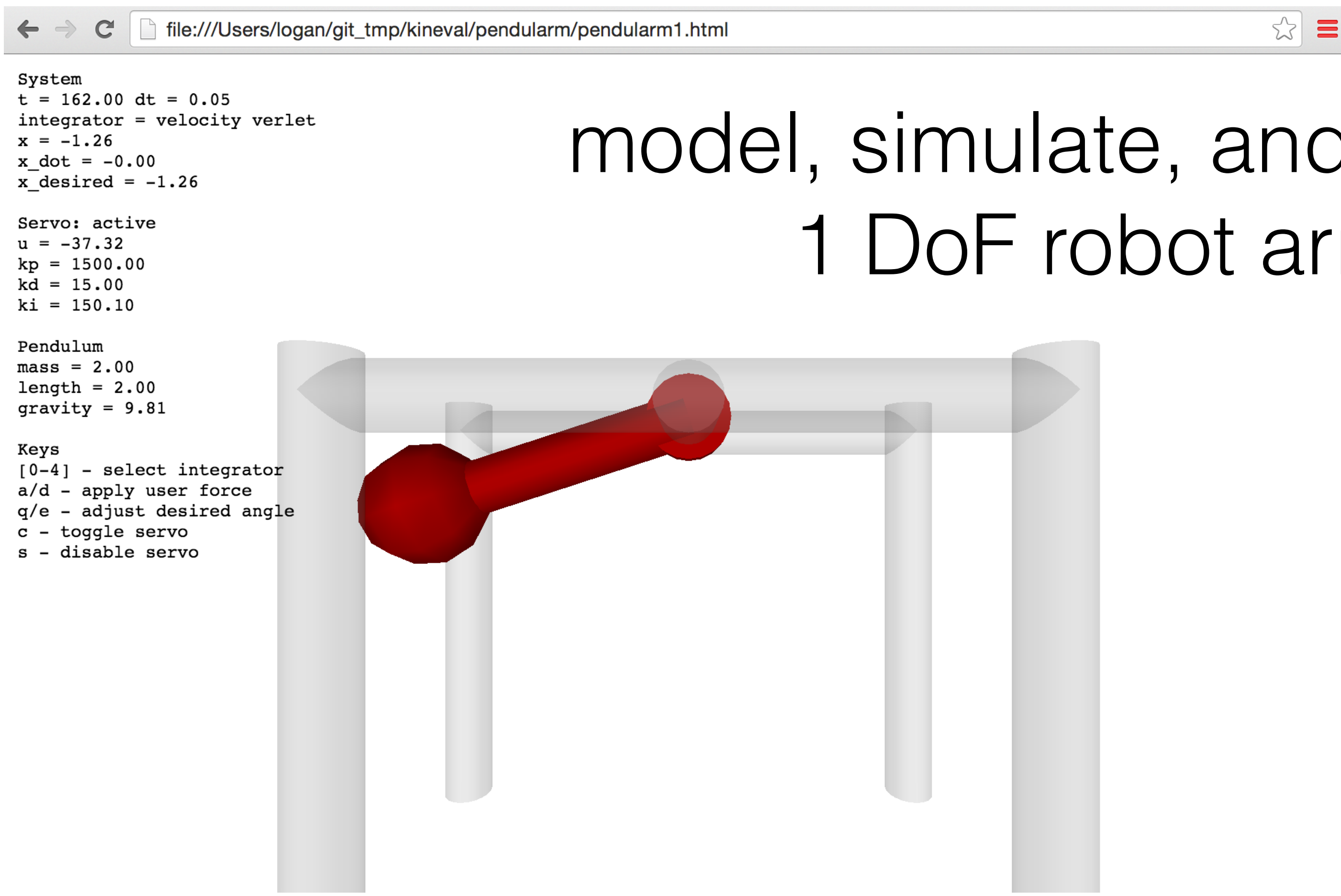
Valkyrie in Gazebo simulation - https://youtu.be/tLCpJvqgtRQ

Our first robot:

Pendularm!

Project 2: 1 DOF Pendularm

model, simulate, and control
1 DoF robot arm

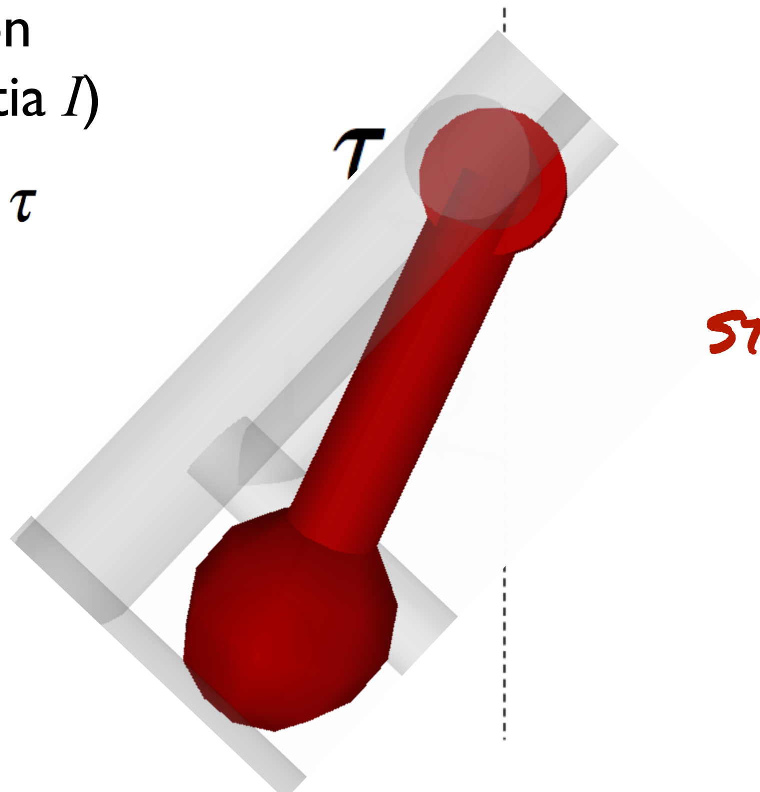


Example: Pendulum

Dynamics

Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$



Controls

Motor produces torque
(angular force)

State (or Configuration)

Angle expresses pendulum
range of motion

System

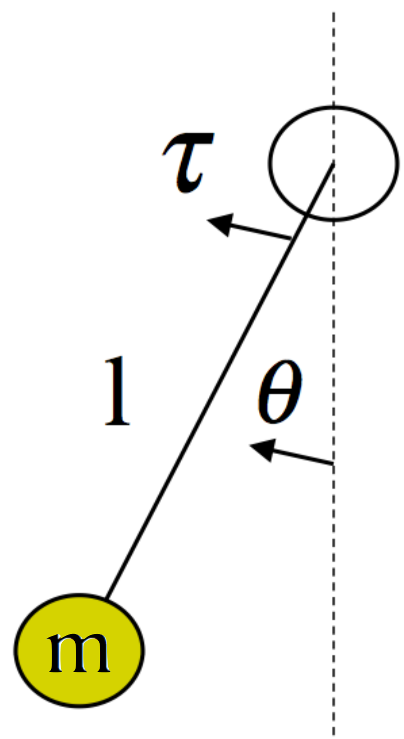
Pendulum of length *l* with
point mass *m*

Example: Pendulum

DYNAMICS

Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$



CONTROLS

Motor produces torque
(angular force)

STATE (OR CONFIGURATION)

Angle expresses pendulum
range of motion

SYSTEM

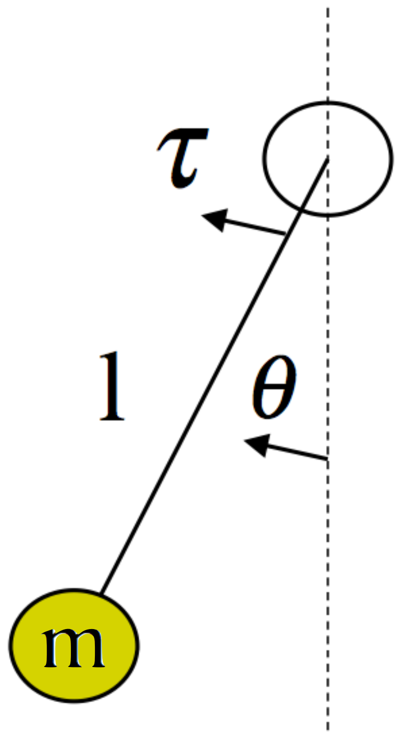
Pendulum of length *l* with
point mass *m*

Example: Pendulum

Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$

MOTOR TORQUE



Motor produces torque (angular force)

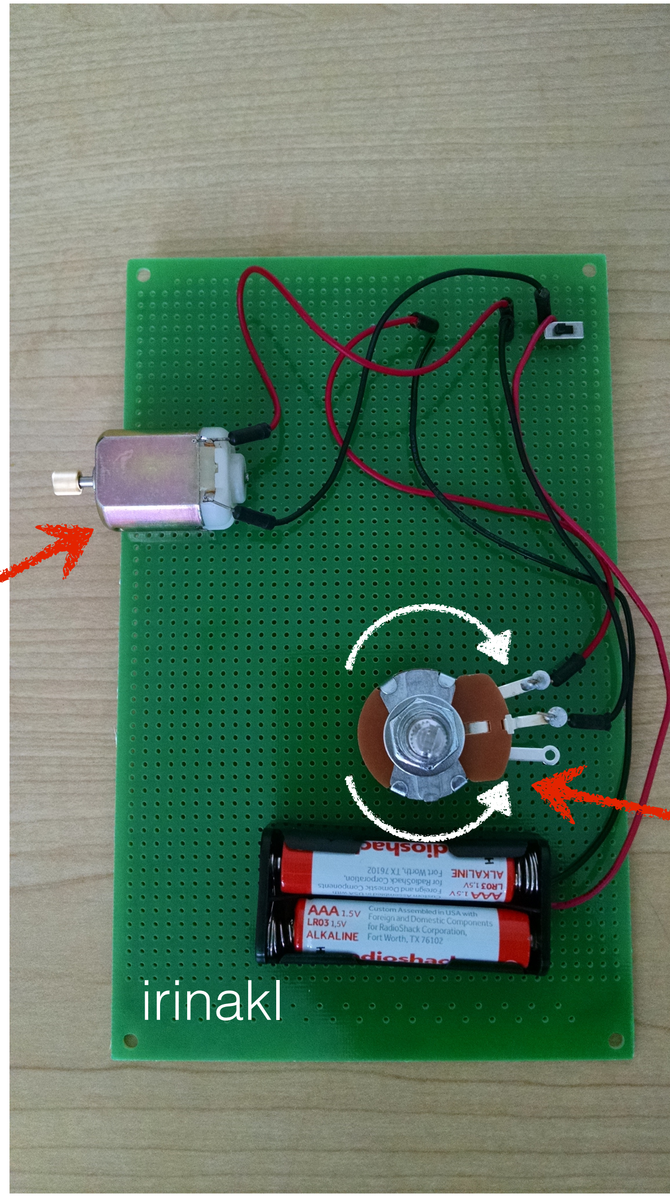
Angle expresses pendulum range of motion

Pendulum of length l with point mass m

Example device

Control torque of motor as a proportion of current applied

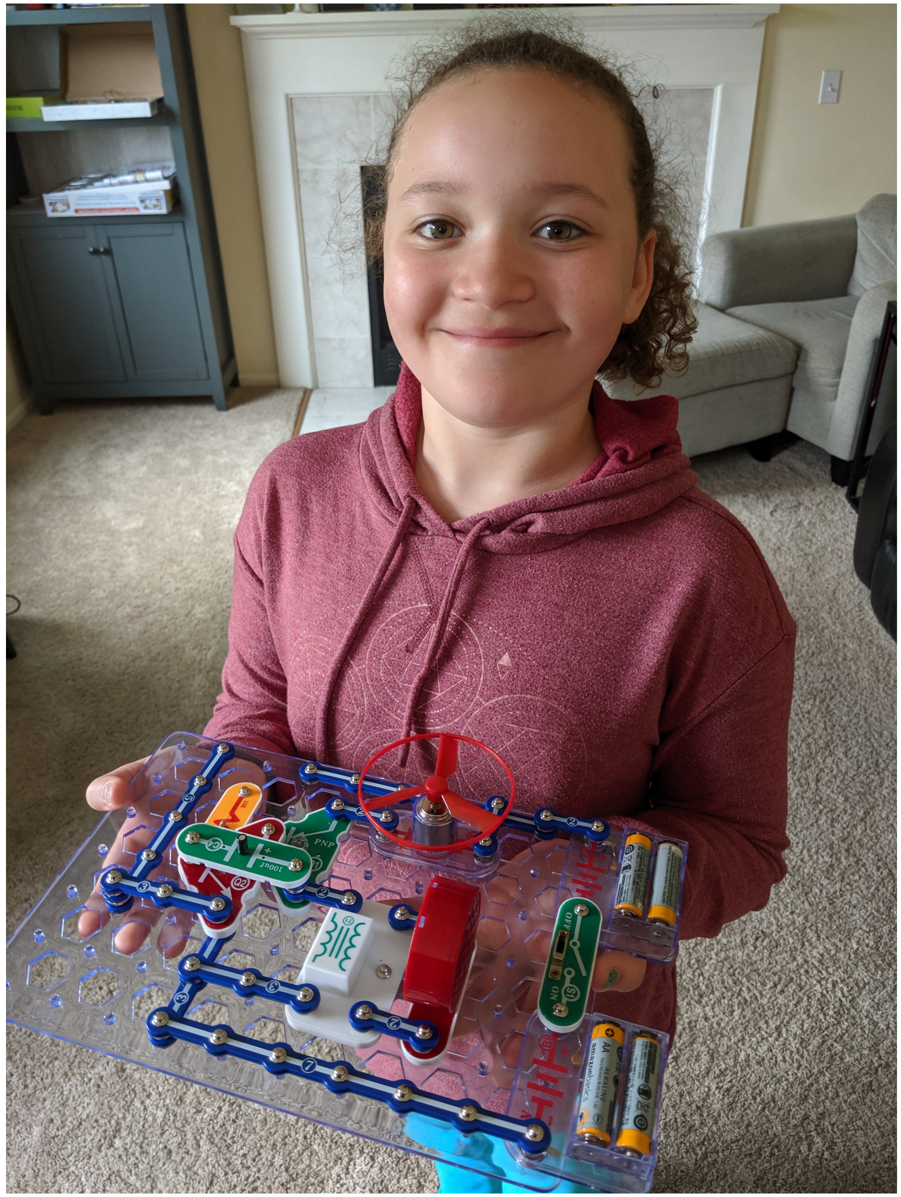
DC MOTOR

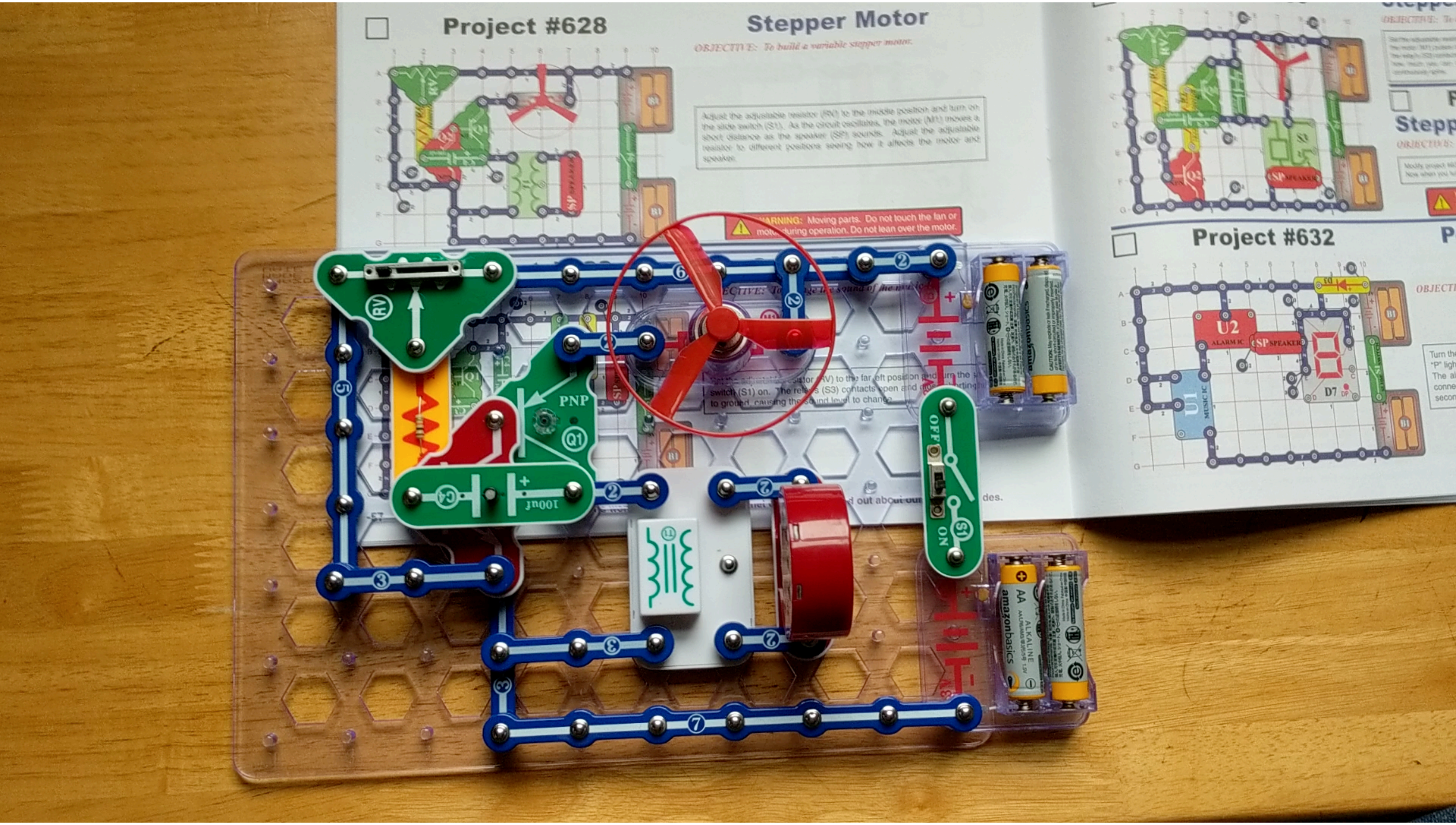


Decrease resistance to increase force

POTENTIOMETER

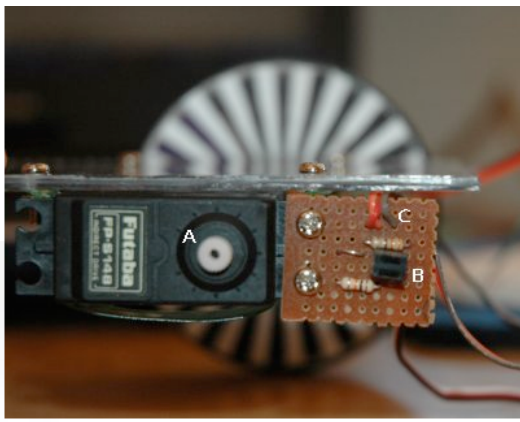
Increase resistance to decrease force



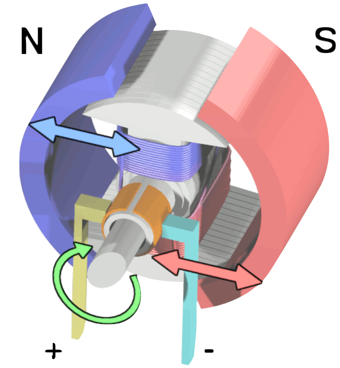

Project #628
Stepper Motor
OBJECTIVE: To build a variable stepper motor.
Adjust the adjustable resistor (RV) to the middle position and turn on the slide switch (S1). As the circuit oscillates, the motor (M1) moves a short distance as the speaker (SP) sounds. Adjust the adjustable resistor to different positions seeing how it affects the motor and speaker.
Project #632
RV
PNP
Q1
C4
100uF
OFF
ON
S1
AA
amazonbasics
ALKALINE

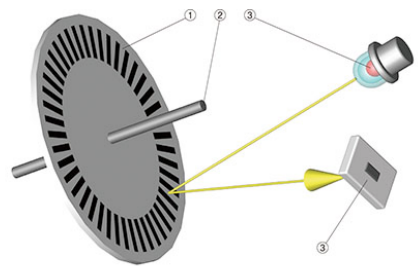
- actuators to produce motion
- proprioception to sense pose
- controller to regulate current to motor

Servo



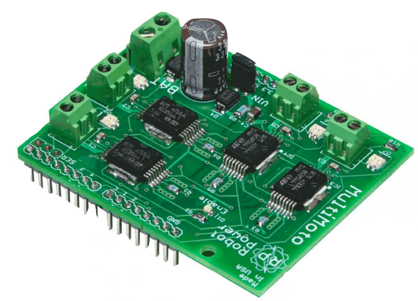
Actuator
(electric)





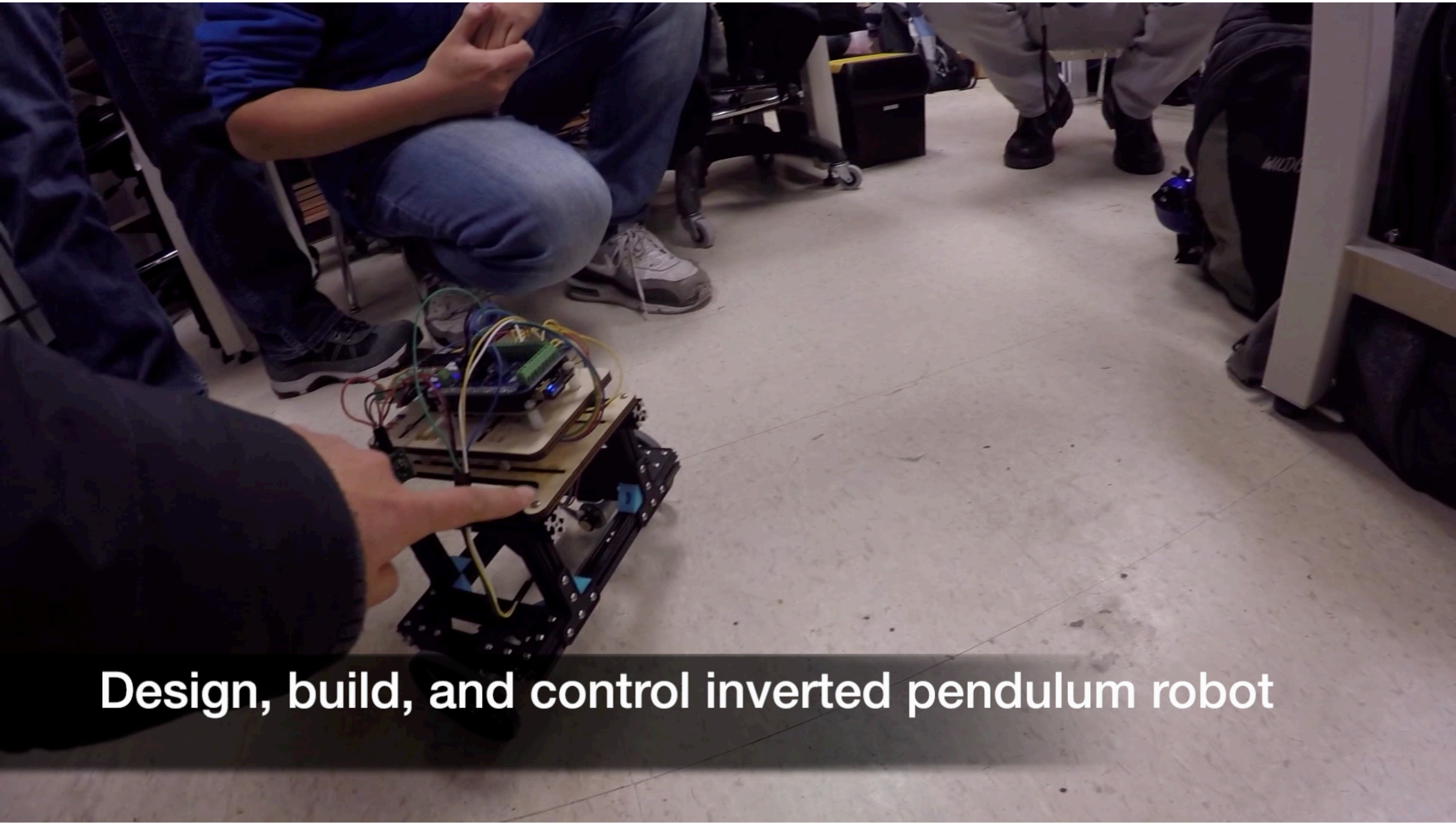
Proprioception
(optical encoder)

Controller
(4 channel H-Bridge)



Consider servos on wheels of an inverted pendulum

ROB 550 Fall 2016 - https://youtu.be/9QUSexcsSZk


Design, build, and control inverted pendulum robot

What is an equation of motion?

Example: Pendulum

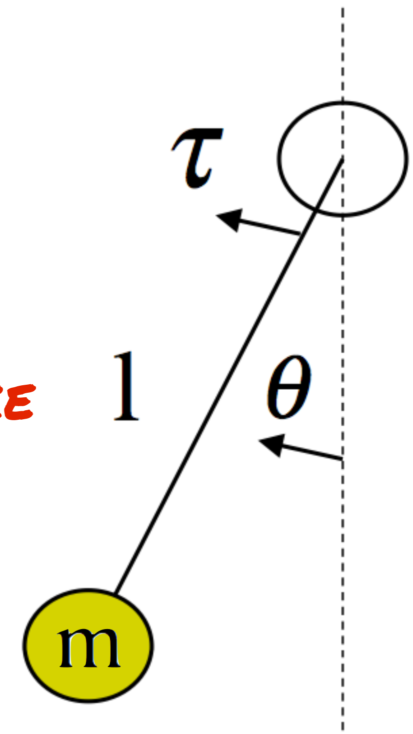
Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = \underline{-mgl\sin(\theta) + \tau}$$

NET FORCE

INERTIA

ACCELERATION



Motor produces torque
(angular force)

Angle expresses pendulum
range of motion

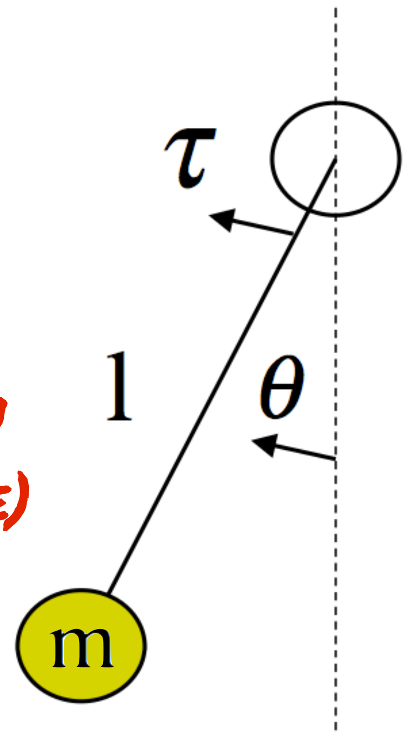
Pendulum of length *l* with
point mass *m*

Example: Pendulum

Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$

ANGULAR ACCELERATION
(SECOND TIME DERIVATIVE)



Motor produces torque (angular force)

Angle expresses pendulum range of motion

Pendulum of length *l* with point mass *m*

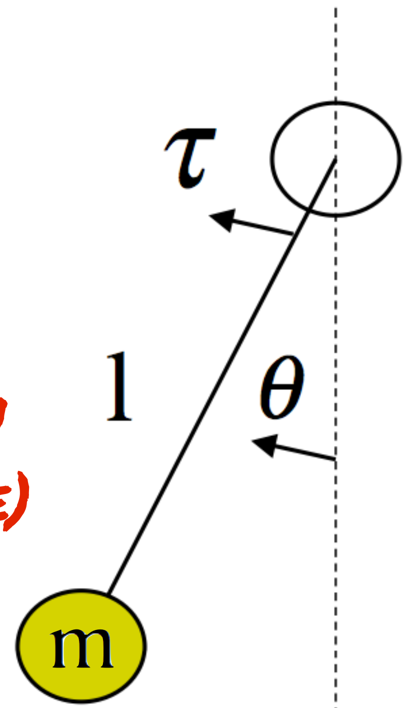
Example: Pendulum

Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$

ANGULAR ACCELERATION
(SECOND TIME DERIVATIVE)

or $\frac{d\dot{\theta}}{dt}$ as the rate of change of the velocity of the pendulum angle



Motor produces torque (angular force)

Angle expresses pendulum range of motion

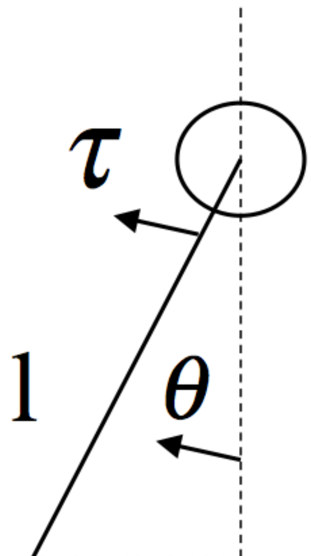
Pendulum of length l with point mass m

Example: Pendulum

Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$

ANGULAR ACCELERATION
(SECOND TIME DERIVATIVE)



or $\dfrac{d\dot{\theta}}{dt}$ as the rate of change of the velocity of the pendulum angle

or $\dfrac{d^2\theta}{dt^2}$ as the rate of change of the rate of change of the pendulum angle

Motor produces torque (angular force)

Angle expresses pendulum range of motion

Pendulum of length *l* with point mass *m*

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$

How was the
equation of motion derived?

$$\frac{d^2\theta}{dt^2} \qquad \frac{d\dot{\theta}}{dt}$$

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$

We are going to need some math

$$\frac{d^2\theta}{dt^2} \qquad \frac{d\dot{\theta}}{dt}$$

Derivative

From Wikipedia, the free encyclopedia

The **derivative** of a function of a real variable measures the sensitivity to change of a quantity (a function or dependent variable) which is determined by another quantity (the independent variable).

$$\frac{d\theta}{dt}$$

is the rate of change of the pendulum angle θ with respect to the variable for time t

$$\theta = f(t)$$

assume that the pendulum angle is a function over time

Derivative

From Wikipedia, the free encyclopedia

The **derivative** of a function of a real variable measures the sensitivity to change of a quantity (a function or dependent variable) which is determined by another quantity (the independent variable).

$$\frac{df}{da}$$

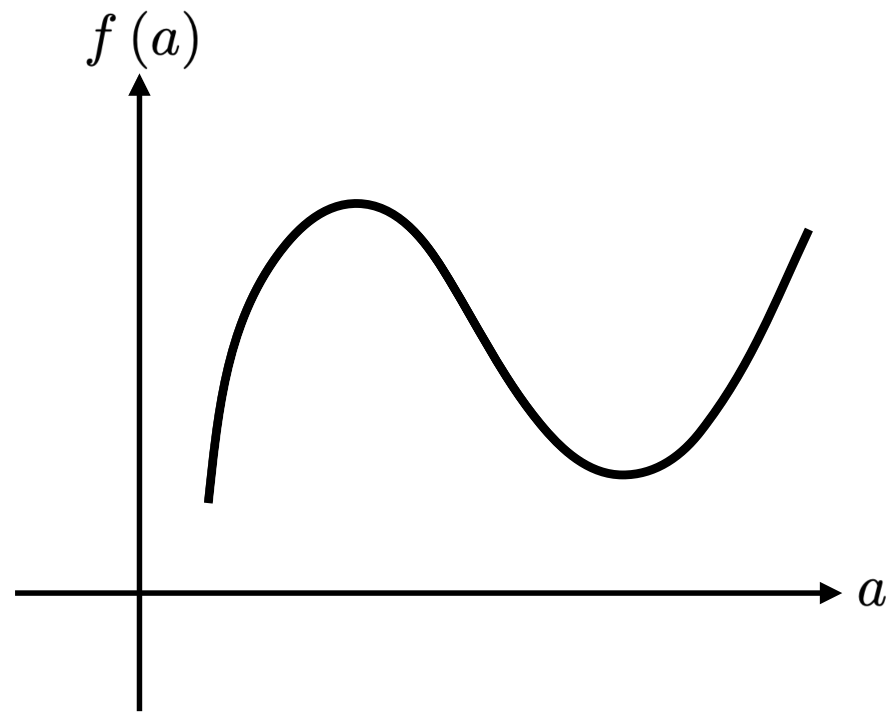
is the rate of change of some function $f(a)$ with respect to some variable a

Derivative

From Wikipedia, the free encyclopedia

The **derivative** of a function of a real variable measures the sensitivity to change of a quantity (a function or dependent variable) which is determined by another quantity (the independent variable).

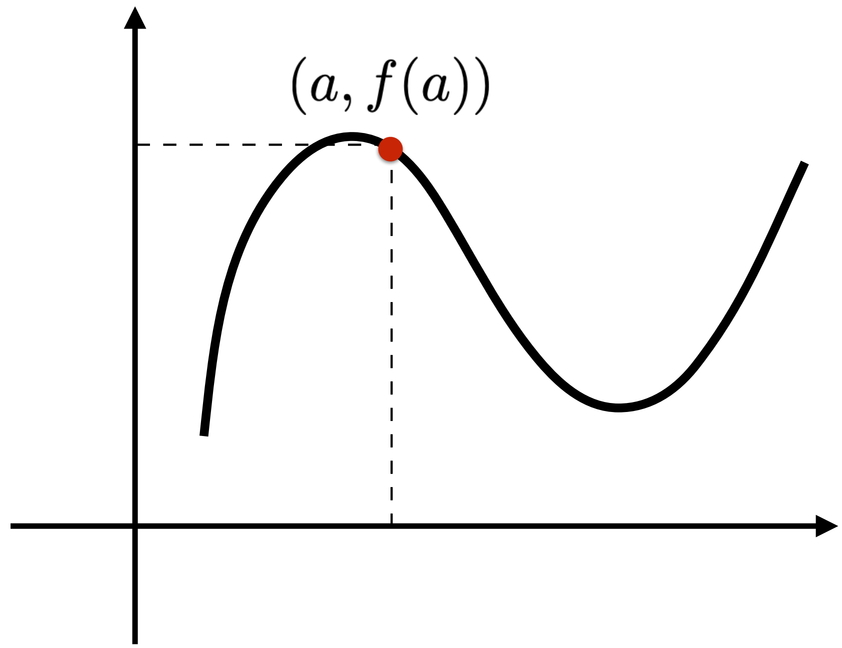
Consider a function $f(a)$ over variable a



Derivative

Consider a function $f(a)$ over variable a

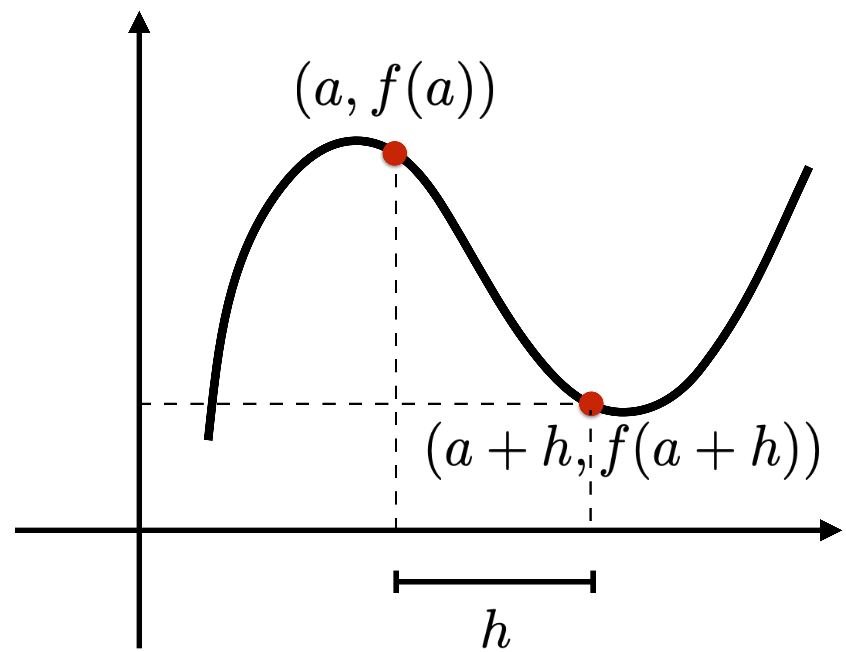
Evaluate $f(a)$ at a



Derivative

Consider a function $f\left(a\right)$ over variable a

Evaluate $f\left(a\right)$ at a and a nearby point $a + h$



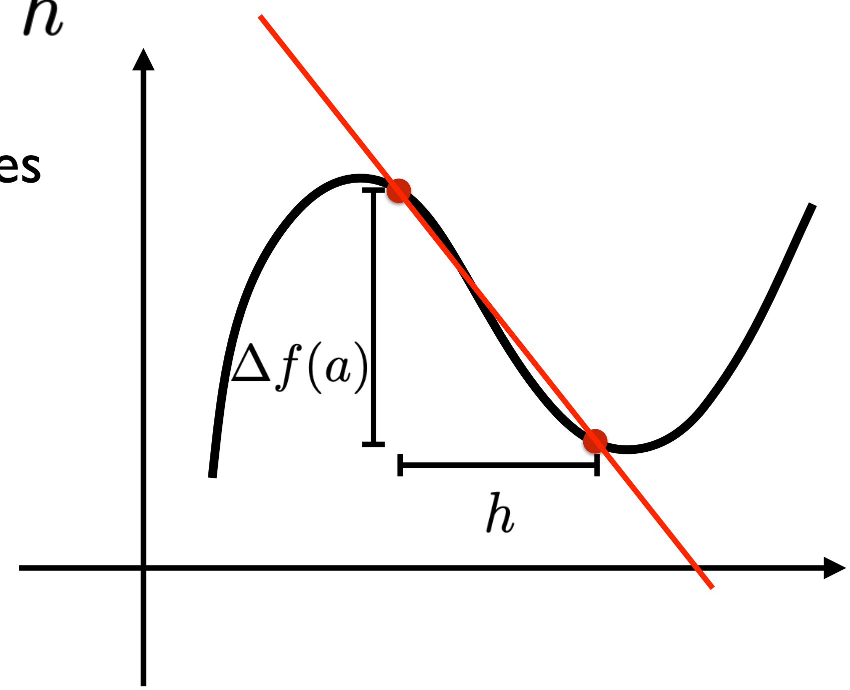
Derivative

Consider a function $f\left(a\right)$ over variable a

Evaluate $f\left(a\right)$ at a and a nearby point $a+h$

Slope of line between these points approximates the rate of change of the function

$$\frac{\Delta f(a)}{\Delta a} = \frac{f\left(a+h\right) - f\left(a\right)}{h}$$



Derivative

Consider a function $f\left(a\right)$ over variable a

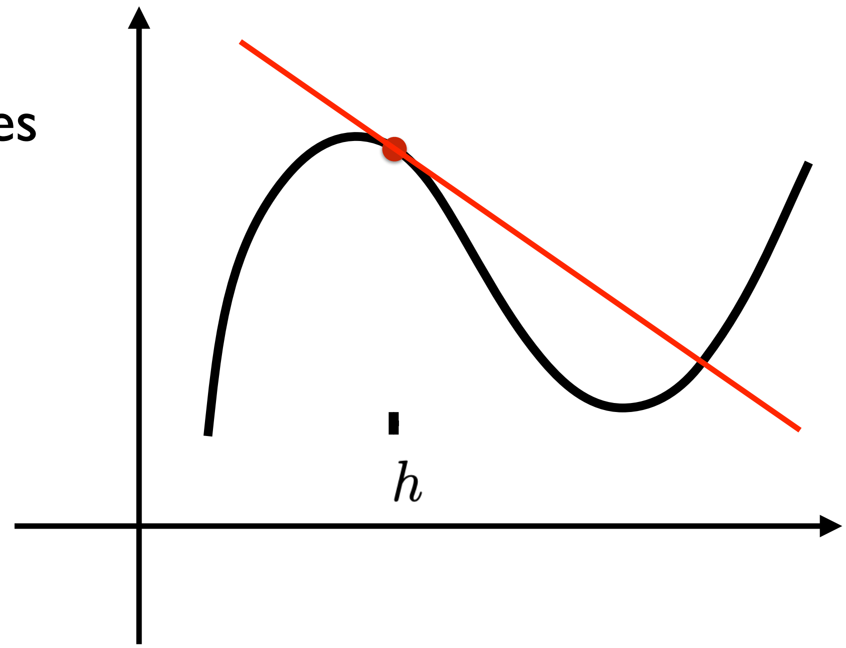
Evaluate $f\left(a\right)$ at a and a nearby point $a+h$

Slope of line between these points approximates the rate of change of the function

$$\frac{\Delta f(a)}{\Delta a} = \frac{f\left(a+h\right) - f\left(a\right)}{h}$$

Derivative is this slope as $a+h$ gets as close as possible to a

$$f'(a) = \lim_{h\to 0} \frac{f\left(a+h\right) - f\left(a\right)}{h}$$



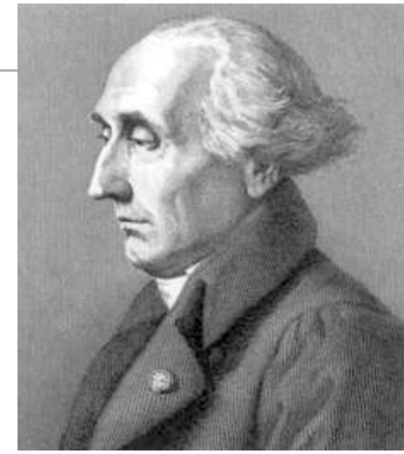
Derivative

Newton's Notation
(assuming *a* is time)

Leibniz's Notation
(assuming function *y* = *f(x)*)

Lagrange's
Notation

Gottfried Wilhelm
Leibniz
(1646-1716)



Joseph-Louis
Lagrange
(1736-1813)

$$\dot{y} = \frac{dy}{dx} = f'(a) = \lim_{h \to 0} \frac{f(a+h) - f(a)}{h}$$

```
$\dot{y} = \frac{dy}{dx} = f'(a)=\lim_{h\to 0}\frac{f(a+h)-f(a)}{h}$
```

will produce this expression in LaTeX math mode

$$\dot{y} = \frac{dy}{dx} = f'(a) = \lim_{h\to 0}\frac{f(a+h)-f(a)}{h}$$

First derivative $\dot{y}$ or $\frac{dy}{dx}$ or f'

$$\frac{\Delta f(a)}{\Delta a} = \frac{f(a+h) - f(a)}{h}$$

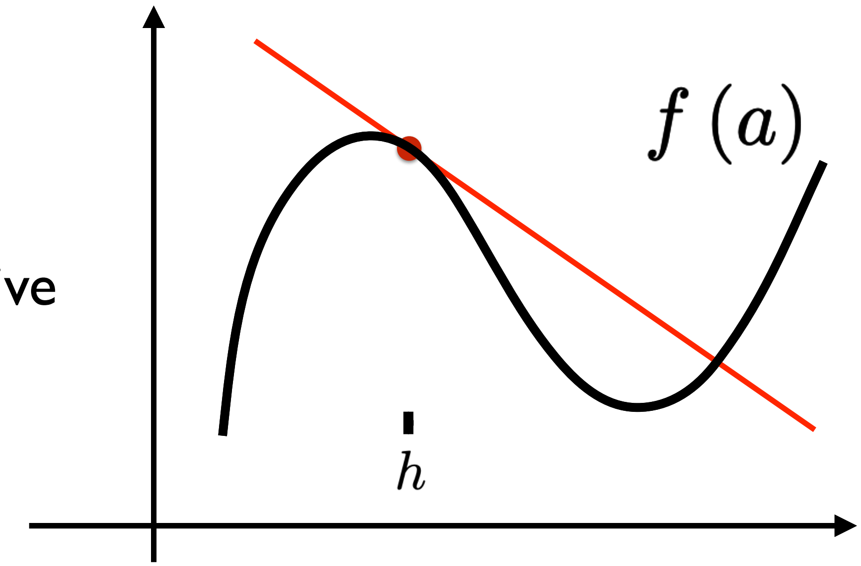
First derivative $\dot{y}$ or $\dfrac{dy}{dx}$ or $f\,'$

$$\frac{\Delta f(a)}{\Delta a} = \frac{f\,(a+h) - f\,(a)}{h}$$

Second derivative $\ddot{y}$ or $\dfrac{d^2y}{dx^2}$ or f''

$$\frac{\Delta^2 f(a)}{\Delta a^2} = \frac{\frac{f(a+h)-f(a)}{h} - \frac{f(a)-f(a-h)}{h}}{h}$$

Differentiation: the process of finding a derivative



For functions expressed in closed form, there are rules for differentiation such as:

Power rule

$$\frac{d}{dx}x^n = nx^{n-1},$$

$$n \neq 0.$$

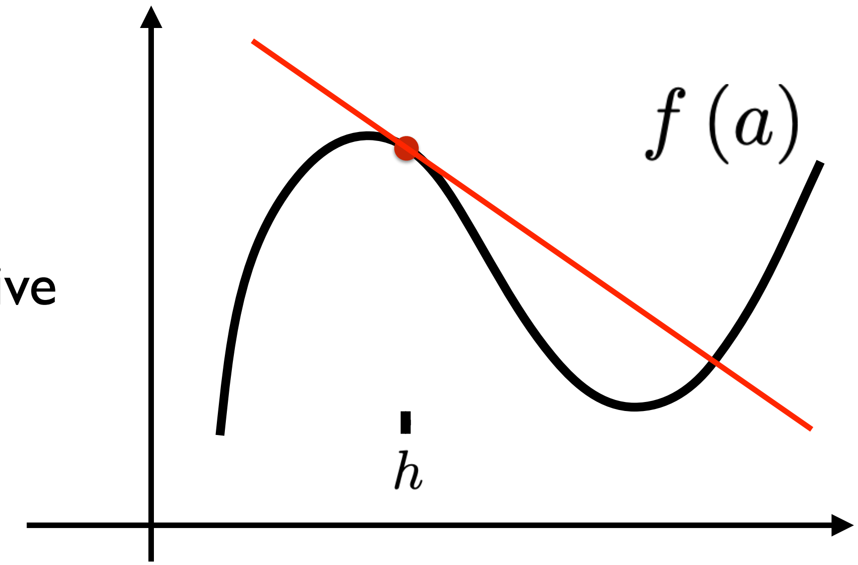
Product rule

$$\frac{d}{dx}(u \cdot v) = u \cdot \frac{dv}{dx} + v \cdot \frac{du}{dx}$$

Chain rule

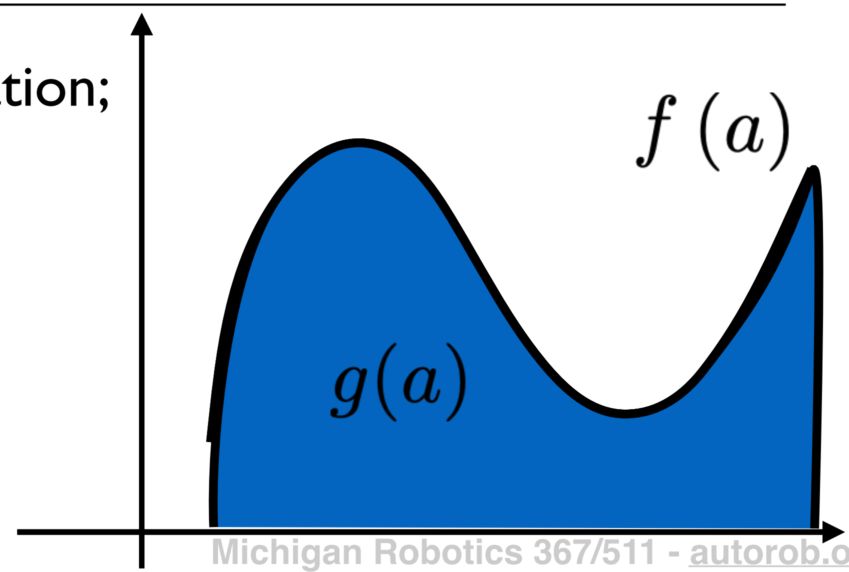
$$[f(g(x))]' = f'(g(x))g'(x)$$

Differentiation: the process of finding a derivative



Integration is the inverse operation of differentiation; finding the area under the function

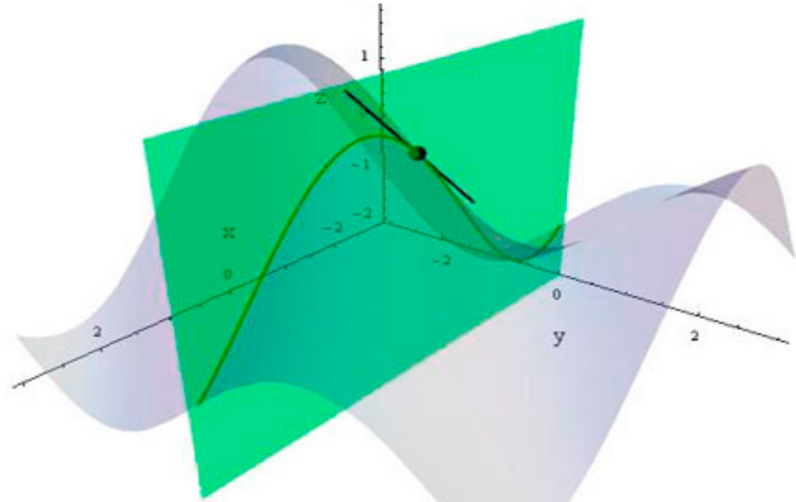
$$\frac{d}{dx}f(x) = g(x), \int g(x)\ dx = f(x) + C$$



Differential equations addresses differentiation for functions of multiple variables

Consider $z = f(x, y, t)$

and $x = x(t)\,, y = y(t)$



https://math.stackexchange.com/questions/607942/what-is-the-best-way-to-think-about-partial-derivatives

---

Partial derivative of z with respect to t is

$$\frac{\partial f}{\partial t}$$

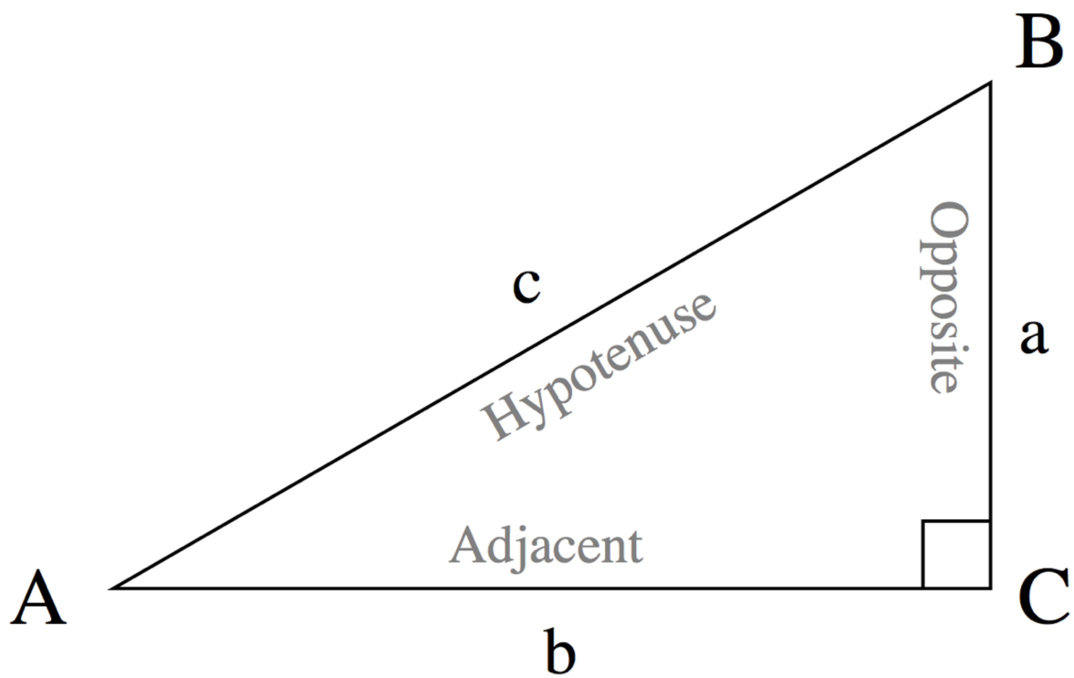
x AND y TREATED AS CONSTANTS

Total derivative of z with respect to t is

$$\frac{\mathrm{d}f}{\mathrm{d}t} = \frac{\partial f}{\partial t}\frac{\mathrm{d}t}{\mathrm{d}t} + \frac{\partial f}{\partial x}\frac{\mathrm{d}x}{\mathrm{d}t} + \frac{\partial f}{\partial y}\frac{\mathrm{d}y}{\mathrm{d}t}$$

VIA THE CHAIN RULE

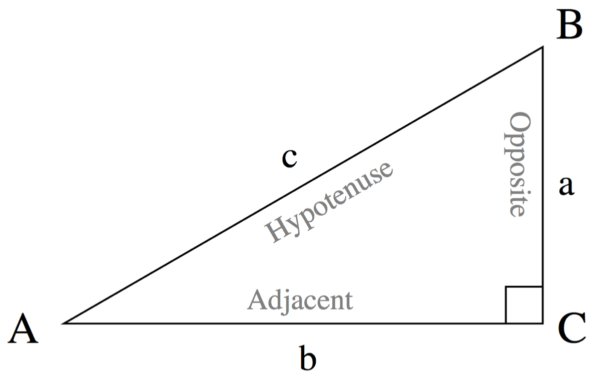
Don't forget trig



Trigonometry

From Wikipedia, the free encyclopedia

Trigonometry (from Greek *trigōnon*, "triangle" and *metron*, "measure"[1]) is a branch of mathematics that studies relationships involving lengths and angles of triangles. The field emerged during the 3rd century BC from applications of geometry to astronomical studies.[2]



`Math.sin(A)`

$$\sin A = \frac{\text{opposite}}{\text{hypotenuse}} = \frac{a}{c}.$$

`Math.cos(A)`

$$\cos A = \frac{\text{adjacent}}{\text{hypotenuse}} = \frac{b}{c}.$$

`Math.tan(A)`

$$\tan A = \frac{\text{opposite}}{\text{adjacent}} = \frac{a}{b} = \frac{\sin A}{\cos A}$$

`Math.atan2(a,b)`

$$\operatorname{atan2}(y, x) = \begin{cases} \arctan \frac{y}{x} & x > 0 \\ \arctan \frac{y}{x} + \pi & y \geq 0, x < 0 \\ \arctan \frac{y}{x} - \pi & y < 0, x < 0 \\ +\frac{\pi}{2} & y > 0, x = 0 \\ -\frac{\pi}{2} & y < 0, x = 0 \\ \text{undefined} & y = 0, x = 0 \end{cases}$$

Coming back …

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$

How was the
equation of motion derived?

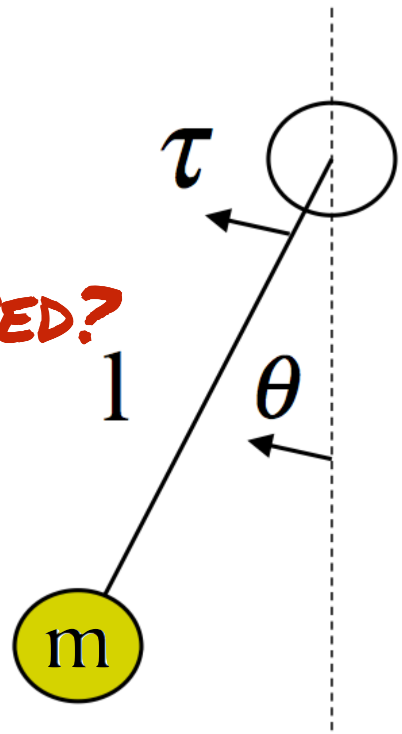
$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$

Example: Pendulum

Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$

How was this derived?



Motor produces torque (angular force)

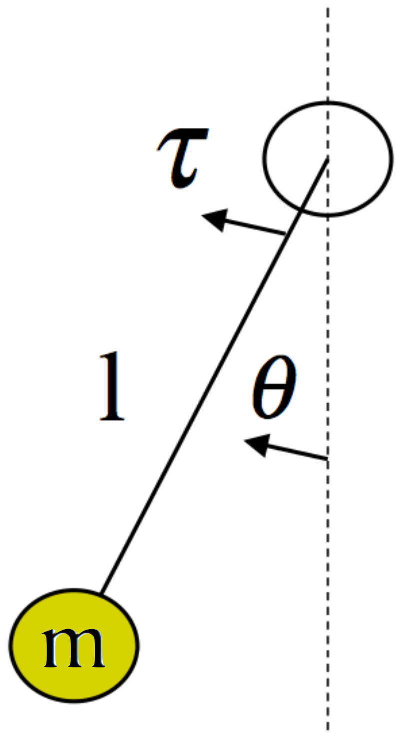
Angle expresses pendulum range of motion

Pendulum of length l with point mass m

Lagrangian Dynamics

Equation of motion
(with rotational inertia *I*)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$



Lagrangian is kinetic energy minus potential energy

$$L = T - U$$

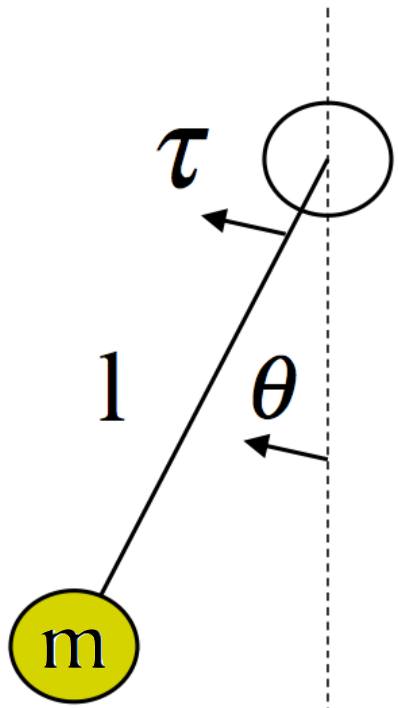
and used to generate equation of motion as

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_i} - \frac{\partial L}{\partial \theta_i} = \tau_i$$

Lagrangian Dynamics

Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$



Lagrangian is kinetic energy minus potential energy

$$L = T - U$$

and used to generate equation of motion as

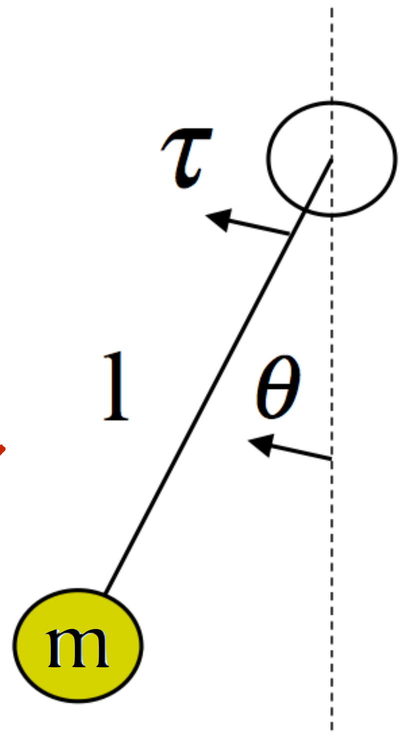
$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_i} - \frac{\partial L}{\partial \theta_i} = \tau_i$$

$\frac{\partial L}{\partial \theta_i}$ is the rate of change of the Lagrangian with respect to only the i^{th} degree of freedom

Lagrangian Dynamics

Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$



Lagrangian is kinetic energy minus potential energy

$$L = T - U$$

and used to generate equation of motion as

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_i} - \frac{\partial L}{\partial \theta_i} = \tau_i$$

WHAT IS THE KINETIC ENERGY OF A PARTICLE?

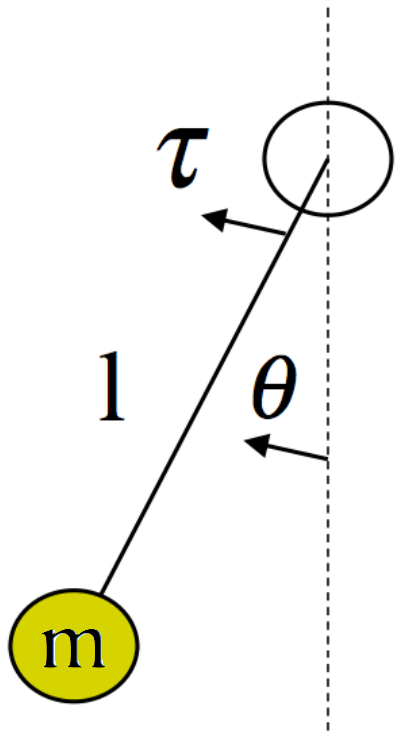
WHAT IS ITS POTENTIAL ENERGY?

Lagrangian Dynamics

Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$

$$(1/2)mv^2$$



$$mgh$$

Lagrangian is kinetic energy minus potential energy

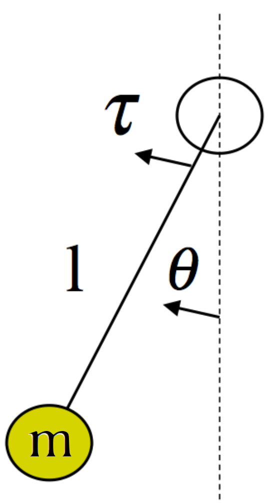
$$L = T - U$$

and used to generate equation of motion as

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_i} - \frac{\partial L}{\partial \theta_i} = \tau_i$$

Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$



Lagrangian Dynamics for Pendulum

- Kinetic Energy $T = \frac{1}{2} I\dot{\theta}^2$

  $(1/2)mv^2$

- Potential Energy $U = mgl(1-\cos\theta)$

  mgh

- Lagrangian $\frac{1}{2} I\dot{\theta}^2 - mgl(1-\cos\theta)$

  $L = T - U$

- Equation of motion $I\ddot{\theta} = -mgl\sin(\theta) + \tau$

  $$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_i} - \frac{\partial L}{\partial \theta_i} = \tau_i$$

- Equation of motion

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_i} - \boxed{\frac{\partial L}{\partial \theta_i}} = \tau_i$$

$$L = \frac{1}{2} I \dot{\theta}^2 - mgl(1 - \cos\theta)$$

- Equation of motion

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_i} - \frac{\partial L}{\partial \theta_i} = \tau_i$$

$$L = \frac{1}{2} I\dot{\theta}^2 - mgl(1 - \cos\theta)$$

- Partial derivative of Lagrangian wrt. pendulum angle

$$\frac{\partial L}{\partial \theta_i} = \frac{\partial}{\partial \theta_i} \frac{1}{2} I\dot{\theta}^2 - mgl(1 - \cos\theta)$$

- Equation of motion

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_i} - \frac{\partial L}{\partial \theta_i} = \tau_i$$

$$\boxed{L = \frac{1}{2} I\dot{\theta}^2 - mgl(1-\cos\theta)}$$

- Partial derivative of Lagrangian wrt. pendulum angle

$$\frac{\partial L}{\partial \theta_i} = \frac{\partial}{\partial \theta_i} \overset{0}{\cancel{\frac{1}{2} I\dot{\theta}^2}} - mgl(1-\cos\theta)$$

$\frac{1}{2} I\dot{\theta}^2$ First term is constant wrt. θ

differentiates to zero

- Equation of motion

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_i} - \frac{\partial L}{\partial \theta_i} = \tau_i$$

$$\boxed{L = \frac{1}{2} I\dot{\theta}^2 - mgl(1-\cos\theta)}$$

- Partial derivative of Lagrangian wrt. pendulum angle

$$\frac{\partial L}{\partial \theta_i} = \frac{\partial}{\partial \theta_i} \overset{0}{\cancel{\frac{1}{2} I\dot{\theta}^2}} - mgl(1-\cos\theta)$$

$\frac{1}{2} I\dot{\theta}^2$ First term is constant wrt. θ

differentiates to zero

Second term: $-mgl(1-\cos\theta)$

distribute multiplication: $-mgl + mgl\cos(\theta)$

cosine identity for differentiation: $\frac{d}{dx}\cos(x) = -\sin(x)$

yields: $-mgl\sin(\theta)$

- Equation of motion

$$\frac{d}{dt}\boxed{\frac{\partial L}{\partial \dot{\theta}_i}} - \frac{\partial L}{\partial \theta_i} = \tau_i$$

$$L = \frac{1}{2} I \dot{\theta}^2 - mgl(1 - \cos\theta)$$

- Partial derivative of Lagrangian wrt. pendulum angle

$$\frac{\partial L}{\partial \theta_i} = -mgl\sin(\theta)$$

- Equation of motion

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_i} - \frac{\partial L}{\partial \theta_i} = \tau_i$$

$$\boxed{L = \frac{1}{2}I\dot{\theta}^2 - mgl(1-\cos\theta)}$$

- Partial derivative of Lagrangian wrt. pendulum angle

$$\frac{\partial L}{\partial \theta_i} = -mgl\sin(\theta)$$

- Partial derivative of Lagrangian wrt. pendulum velocity

$$\frac{\partial L}{\partial \dot{\theta}_i} = \frac{\partial}{\partial \dot{\theta}_i} \frac{1}{2}I\dot{\theta}^2 - mgl(1-\cos\theta)$$

- Equation of motion

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_i} - \frac{\partial L}{\partial \theta_i} = \tau_i$$

$$\boxed{L = \frac{1}{2}I\dot{\theta}^2 - mgl(1-\cos\theta)}$$

- Partial derivative of Lagrangian wrt. pendulum angle

$$\frac{\partial L}{\partial \theta_i} = -mgl\sin(\theta)$$

- Partial derivative of Lagrangian wrt. pendulum velocity

$$\frac{\partial L}{\partial \dot{\theta}_i} = \frac{\partial}{\partial \dot{\theta}_i}\frac{1}{2}I\dot{\theta}^2 - \overset{0}{\cancel{mgl(1-\cos\theta)}}$$

Second term is constant wrt. $\dot{\theta}$

- Equation of motion

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_i} - \frac{\partial L}{\partial \theta_i} = \tau_i$$

$$L = \frac{1}{2} I\dot{\theta}^2 - mgl(1-\cos\theta)$$

- Partial derivative of Lagrangian wrt. pendulum angle

$$\frac{\partial L}{\partial \theta_i} = -mgl\sin(\theta)$$

- Partial derivative of Lagrangian wrt. pendulum velocity

$$\frac{\partial L}{\partial \dot{\theta}_i} = \frac{\partial}{\partial \dot{\theta}_i}\frac{1}{2} I\dot{\theta}^2 - \overset{0}{\cancel{mgl(1-\cos\theta)}}$$

Second term is constant wrt. $\dot{\theta}$

Apply power rule to first term: $f'(x) = rx^{r-1}$

differentiates to: $I\dot{\theta} = \frac{\partial}{\partial \dot{\theta}_i}\frac{1}{2} I\dot{\theta}^2$

- Equation of motion

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_i} - \frac{\partial L}{\partial \theta_i} = \tau_i$$

$$L = \frac{1}{2} I\dot{\theta}^2 - mgl(1 - \cos\theta)$$

- Partial derivative of Lagrangian wrt. pendulum angle

$$\frac{\partial L}{\partial \theta_i} = -mgl\sin(\theta)$$

- Partial derivative of Lagrangian wrt. pendulum velocity

$$\frac{\partial L}{\partial \dot{\theta}_i} = I\dot{\theta}$$

- Time derivative of Partial derivative of Lagrangian wrt. pendulum velocity

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_i} = \frac{d}{dt} I\dot{\theta} = I\ddot{\theta}$$

inertia remains constant

- Equation of motion

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_i} - \frac{\partial L}{\partial \theta_i} = \tau_i$$

$$L = \frac{1}{2} I\dot{\theta}^2 - mgl(1-\cos\theta)$$

- Partial derivative of Lagrangian wrt. pendulum angle

$$\frac{\partial L}{\partial \theta_i} = -mgl\sin(\theta)$$

- Partial derivative of Lagrangian wrt. pendulum velocity

$$\frac{\partial L}{\partial \dot{\theta}_i} = I\dot{\theta}$$

- Time derivative of Partial derivative of Lagrangian wrt. pendulum velocity

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_i} = \frac{d}{dt} I\dot{\theta} = I\ddot{\theta}$$

- Equation of motion

$$I\ddot{\theta} + mgl\sin(\theta) = \tau_i$$

$$L = \frac{1}{2}I\dot{\theta}^2 - mgl(1-\cos\theta)$$

- Partial derivative of Lagrangian wrt. pendulum angle

$$\frac{\partial L}{\partial \theta_i} =$$

- Partial derivative of Lagrangian wrt. pendulum velocity

$$\frac{\partial L}{\partial \dot{\theta}_i} = I\dot{\theta}$$

- Time derivative of Partial derivative of Lagrangian wrt. pendulum velocity

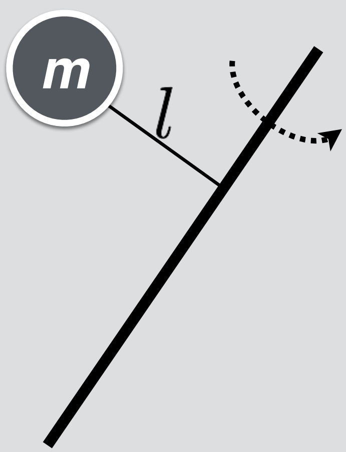
$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_i} = \frac{d}{dt} I\dot{\theta} =$$

- Equation of motion $I\ddot{\theta} + mgl\sin(\theta) = \tau_i$

$$L = \frac{1}{2} I\dot{\theta}^2 - mgl(1 - \cos\theta)$$

- Parallel axis theorem $I = ml^2$

Inertia grows quadratically as mass moves further from its axis of rotation



- Equation of motion

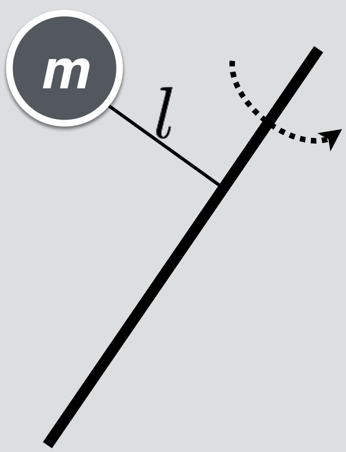
$$I\ddot{\theta} + mgl\sin(\theta) = \tau_i$$

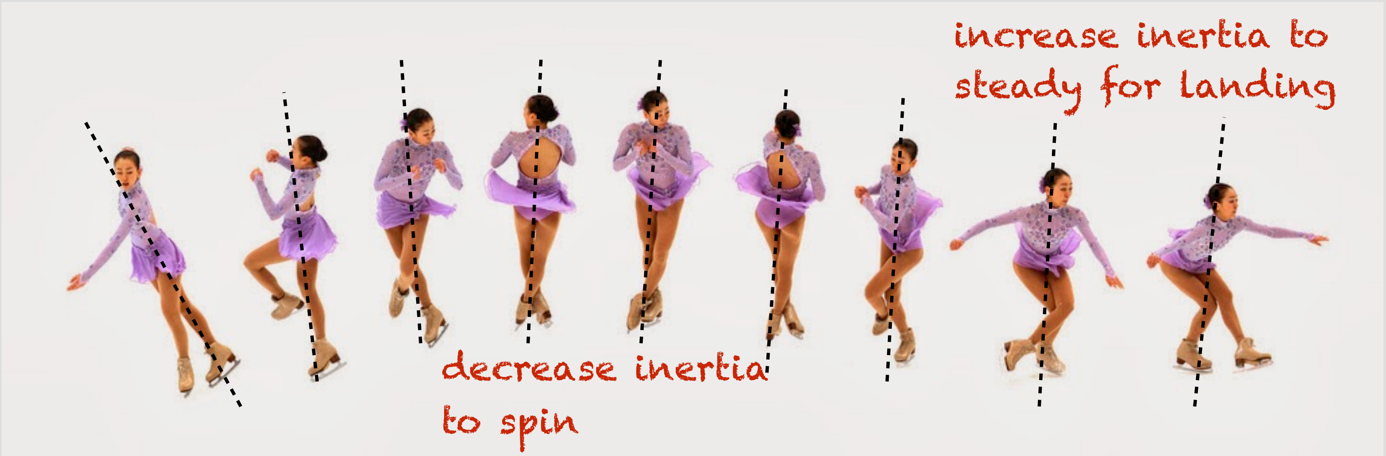
$$L = \frac{1}{2}I\dot{\theta}^2 - mgl(1-\cos\theta)$$

- Parallel axis theorem

$$I = ml^2$$

Inertia grows quadratically as mass moves further from its axis of rotation





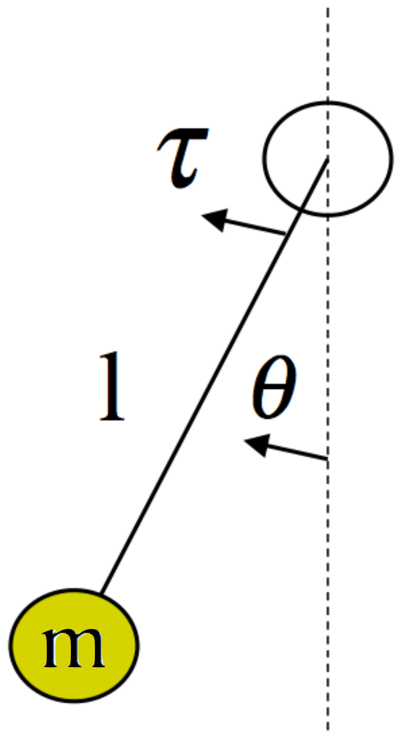
Example: Pendulum

Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$

with Parallel Axis Theorem ($I = ml^2$)

$$I\ddot{\theta} + mgl\sin(\theta) = \tau_i$$



Motor produces torque
(angular force)

Angle expresses pendulum
range of motion

Pendulum of length *l* with
point mass *m*

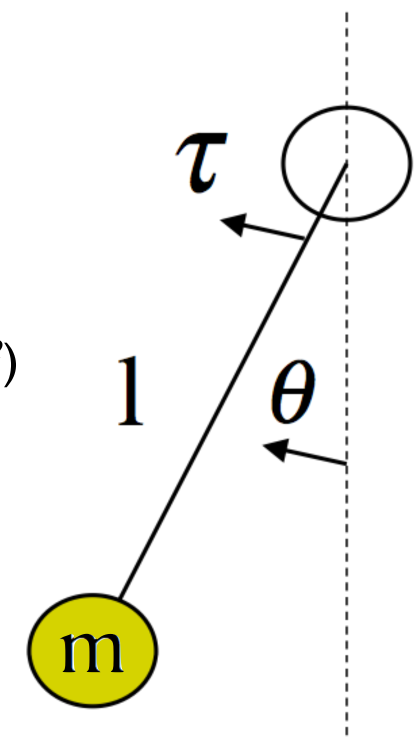
Example: Pendulum

Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$

with Parallel Axis Theorem ($I=ml^2$)

$$I\ddot{\theta} + mgl\sin(\theta) = \tau_i$$



Motor produces torque (angular force)

Angle expresses pendulum range of motion

Pendulum of length *l* with point mass *m*

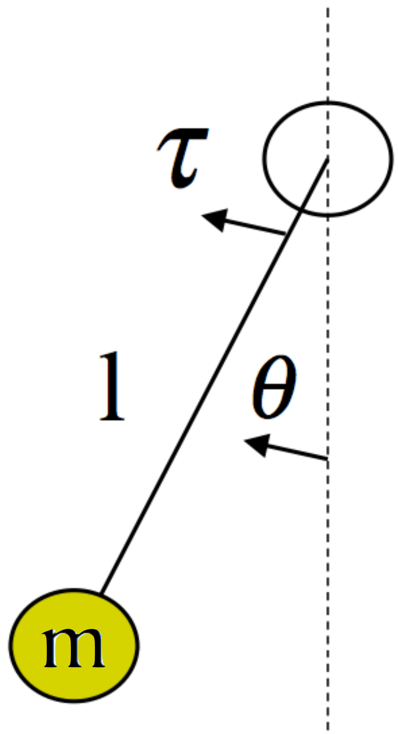
Example: Pendulum

Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$

with Parallel Axis Theorem ($I=ml^2$)

$$\ddot{\theta} = -\frac{g}{l}\sin(\theta) + \frac{\tau}{ml^2}$$



Motor produces torque
(angular force)

Angle expresses pendulum
range of motion

Pendulum of length *l* with
point mass *m*

Example: Pendulum

Equation of motion
(with rotational inertia I)

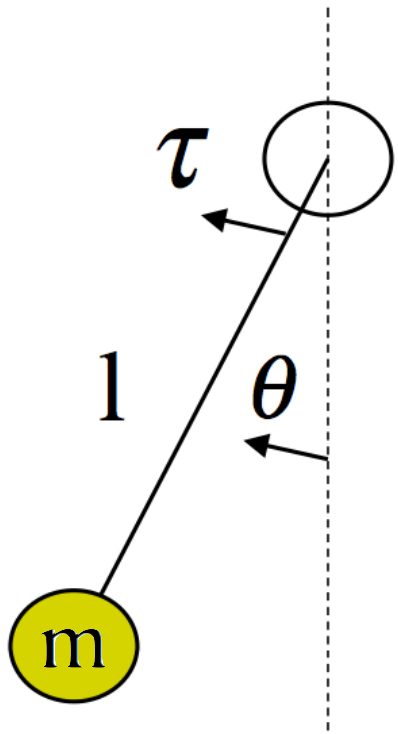
$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$

with Parallel Axis Theorem ($I=ml^2$)

$$\ddot{\theta} = -\frac{g}{l}\sin(\theta) + \frac{\tau}{ml^2}$$

GRAVITY TERM

MOTOR TERM



Motor produces torque (angular force)

Angle expresses pendulum range of motion

Pendulum of length *l* with point mass *m*

Example: Pendulum

Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$

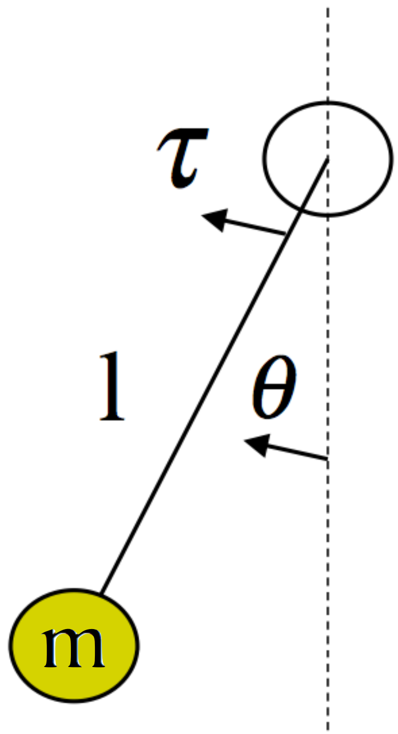
with Parallel Axis Theorem ($I=ml^2$)

$$\ddot{\theta} = -\frac{g}{l}\sin(\theta) + \frac{\tau}{ml^2}$$

Numerical integration over time

$$\theta_{t+\Delta\theta} = \theta_t + \dot{\theta}_t \Delta\theta$$

$$\dot{\theta}_{t+\Delta\theta} = \dot{\theta}_t + \ddot{\theta}_t \Delta\theta$$



Motor produces torque (angular force)

Angle expresses pendulum range of motion

Pendulum of length *l* with point mass *m*

Example: Pendulum

Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$

with Parallel Axis Theorem ($I=ml^2$)

$$\ddot{\theta} = -\frac{g}{l}\sin(\theta) + \frac{\tau}{ml^2}$$

What is this?

Numerical integration over time

$$\theta_{t+\Delta\theta} = \theta_t + \dot{\theta}_t \Delta\theta$$

$$\dot{\theta}_{t+\Delta\theta} = \dot{\theta}_t + \ddot{\theta}_t \Delta\theta$$

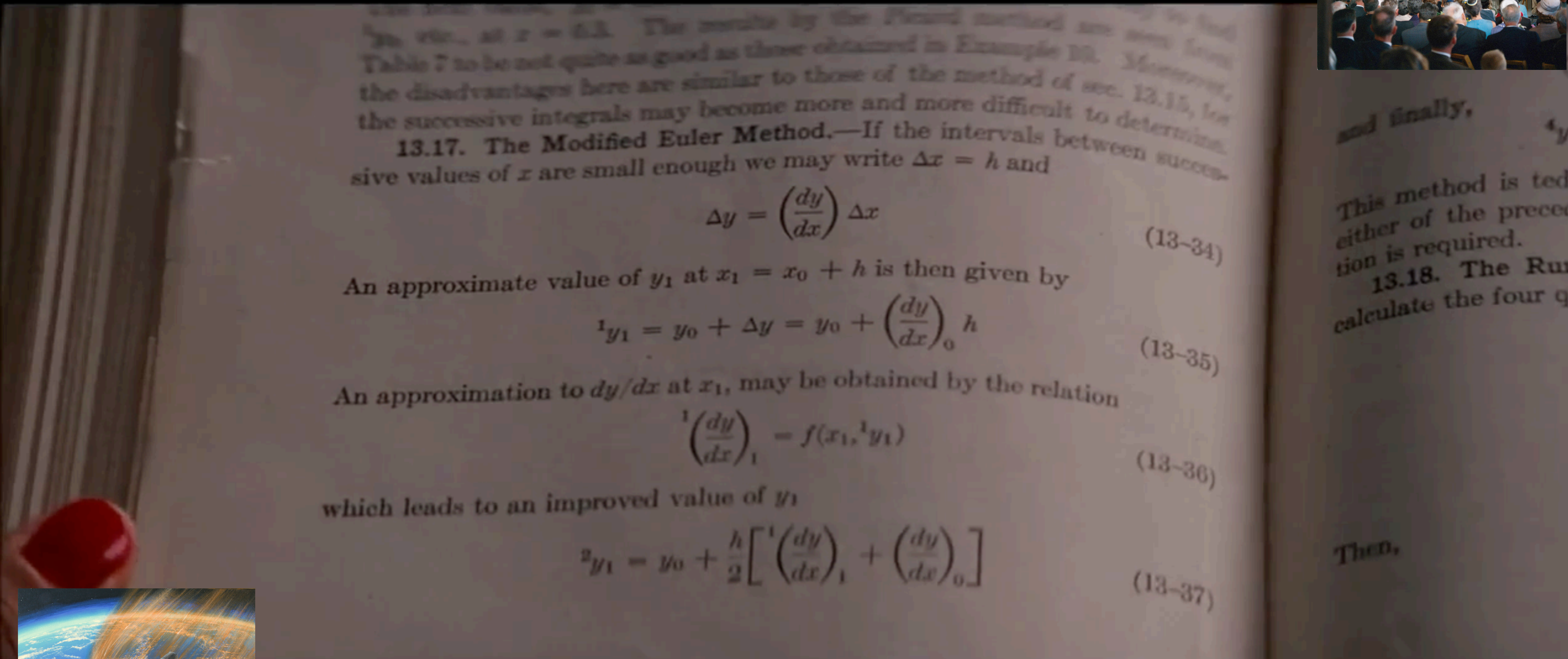
roduces torque
ular force)

resses pendulum
e of motion

Hint

Pendulum of length l with
point mass m


Euler integration
Hidden Figures (2017)



Euler integration

Hidden Figures (2017)

Example: Pendulum

Equation of motion
(with rotational inertia I)

$$I\ddot{\theta} = -mgl\sin(\theta) + \tau$$

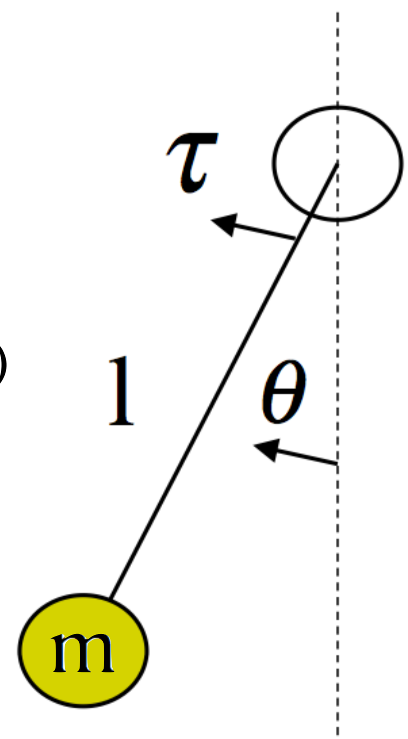
with Parallel Axis Theorem ($I=ml^2$)

$$\ddot{\theta} = -\frac{g}{l}\sin(\theta) + \frac{\tau}{ml^2}$$

Numerical integration over time

$$\theta_{t+\Delta\theta} = \theta_t + \dot{\theta}_t \Delta\theta$$

$$\dot{\theta}_{t+\Delta\theta} = \dot{\theta}_t + \ddot{\theta}_t \Delta\theta$$



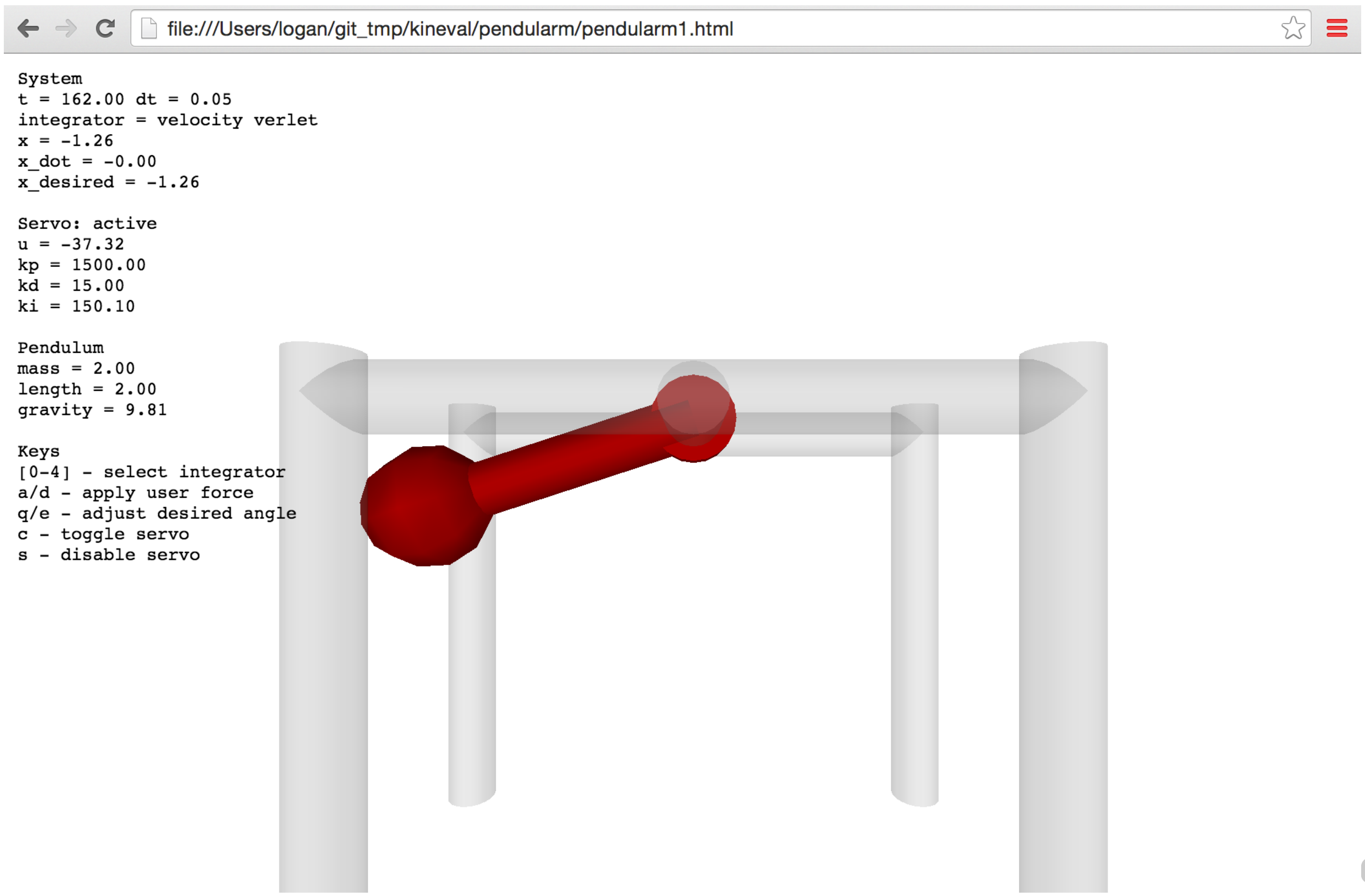
EULER INTEGRATION

Motor produces torque
(angular force)

NOTE: STATE IS BOTH PENDULUM ANGLE AND VELOCITY

Pendulum of length *l* with point mass *m*

Let's see what happens



Euler integration would kill this man

Walter Lewin - https://commons.wikimedia.org/wiki/File:Physicsworks.ogg

Why is Euler Integration not the best choice?

Why is Euler Integration not the best choice?

Think about it as an initial value problem

Initial value problem

From Wikipedia, the free encyclopedia

In mathematics, in the field of differential equations, an **initial value problem** (also called **the Cauchy problem** by some authors) is an ordinary differential equation together with a specified value, called the **initial condition**, of the unknown function at a given point in the domain of the solution. In physics or other sciences, modeling a system frequently amounts to solving an initial value problem; in this context, the differential equation is an evolution equation specifying how, given initial conditions, the system will evolve with time.

Initial value problem

From Wikipedia, the free encyclopedia

Given initial condition (t_0, y_0) where $y_0 = f(t_0)$

TIME

CURRENT STATE

Initial value problem

From Wikipedia, the free encyclopedia

Given initial condition (t_0, y_0) where $y_0 = f(t_0)$ and differential equation $y' = \frac{dy}{dt}$ in the form

$$y'(t) = f(t, y(t))$$

SYSTEM VELOCITY SYSTEM DYNAMICS

Initial value problem

From Wikipedia, the free encyclopedia

Given initial condition (t_0, y_0) where $y_0 = f(t_0)$ and

differential equation $y' = \frac{dy}{dt}$ in the form $y'(t) = f(t, y(t))$

How to estimate future state $y(t_0 + h)$?

NEXT

STATE

Initial value problem

From Wikipedia, the free encyclopedia

Given initial condition (t_0, y_0) where $y_0 = f(t_0)$ and differential equation $y' = \frac{dy}{dt}$ in the form $y'(t) = f(t, y(t))$

How to estimate future state $y(t_0 + h)$?

NUMERICAL INTEGRATION OVER TIMESTEP DURATION

$$y(t_0 + h) - y(t_0) = \int_{t_0}^{t_0+h} f(t, y(t))dt$$

NEXT STATE

CURRENT STATE

INTEGRATION OVER TIMESTEP

Euler Integration

Integral over timestep approximated as $\int_{t_0}^{t_0+h} f(t, y(t))dt \approx \boldsymbol{hf(t_0, y(t_0))}$

Discrete step from iteration *n* to iteration *n+1* computed in two assignments

Advance state

$$y_{n+1} = y_n + hf(t_n, y_n)$$

NEXT STATE

CURRENT STATE

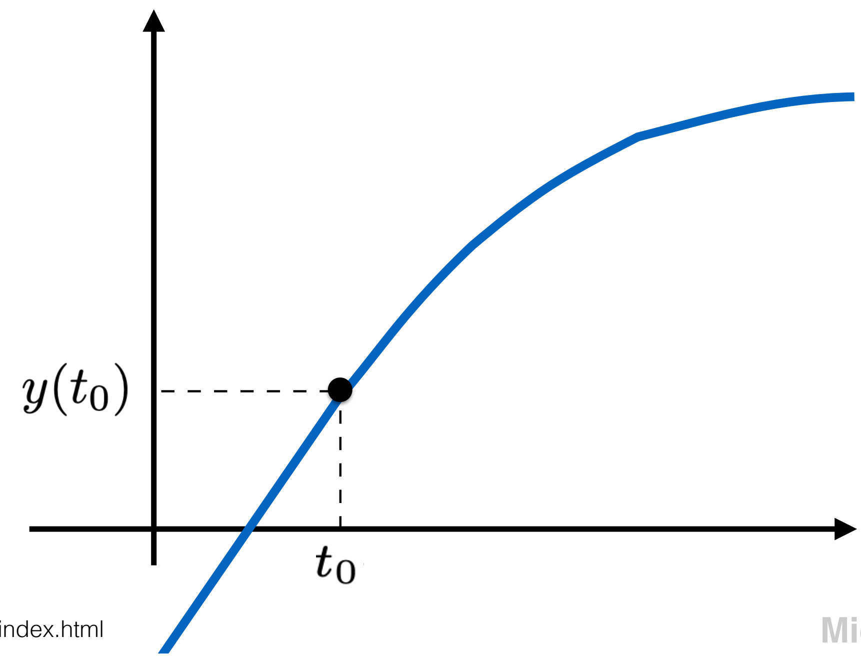
TIMESTEP DURATION * INITIAL RATE OF CHANGE

Advance time

$$t_{n+1} = t_n + h$$

Euler Integration

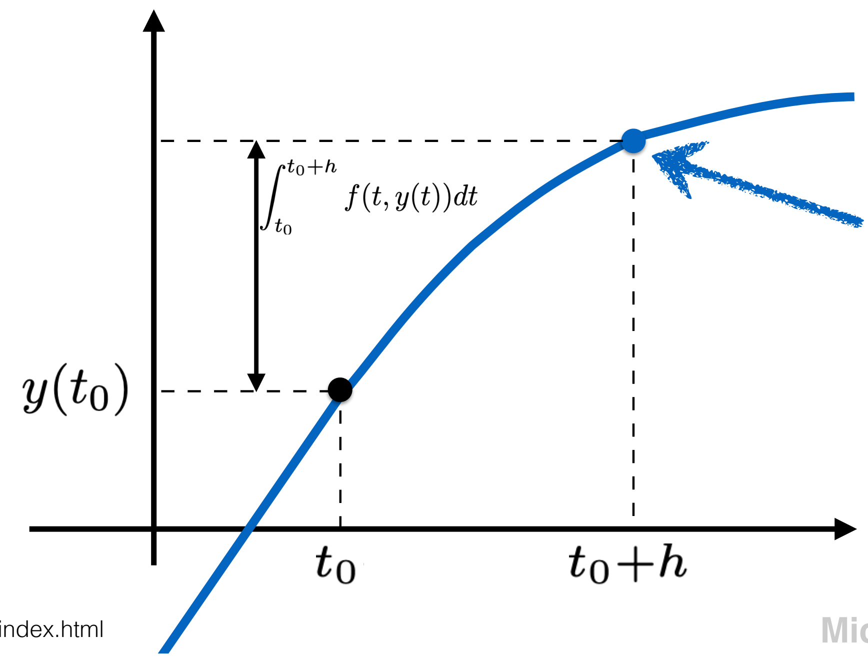
Integral over timestep approximated as $\int_{t_0}^{t_0+h} f(t, y(t))dt \approx hf(t_0, y(t_0))$



http://astro.physics.ncsu.edu/urca/course_files/Lesson15/index.html

Euler Integration

Integral over timestep approximated as $$\int_{t_0}^{t_0+h} f(t, y(t))dt \approx hf(t_0, y(t_0))$$

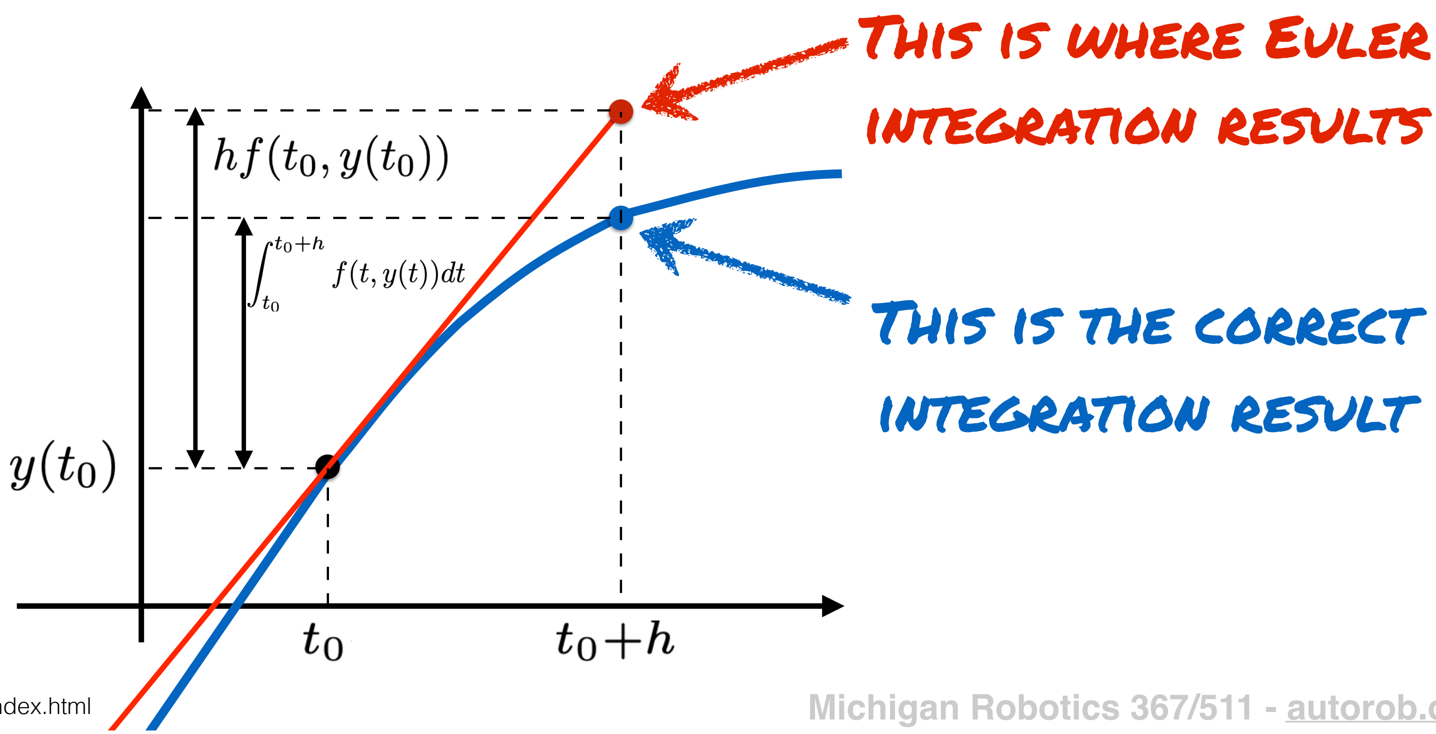


THIS IS THE CORRECT INTEGRATION RESULT

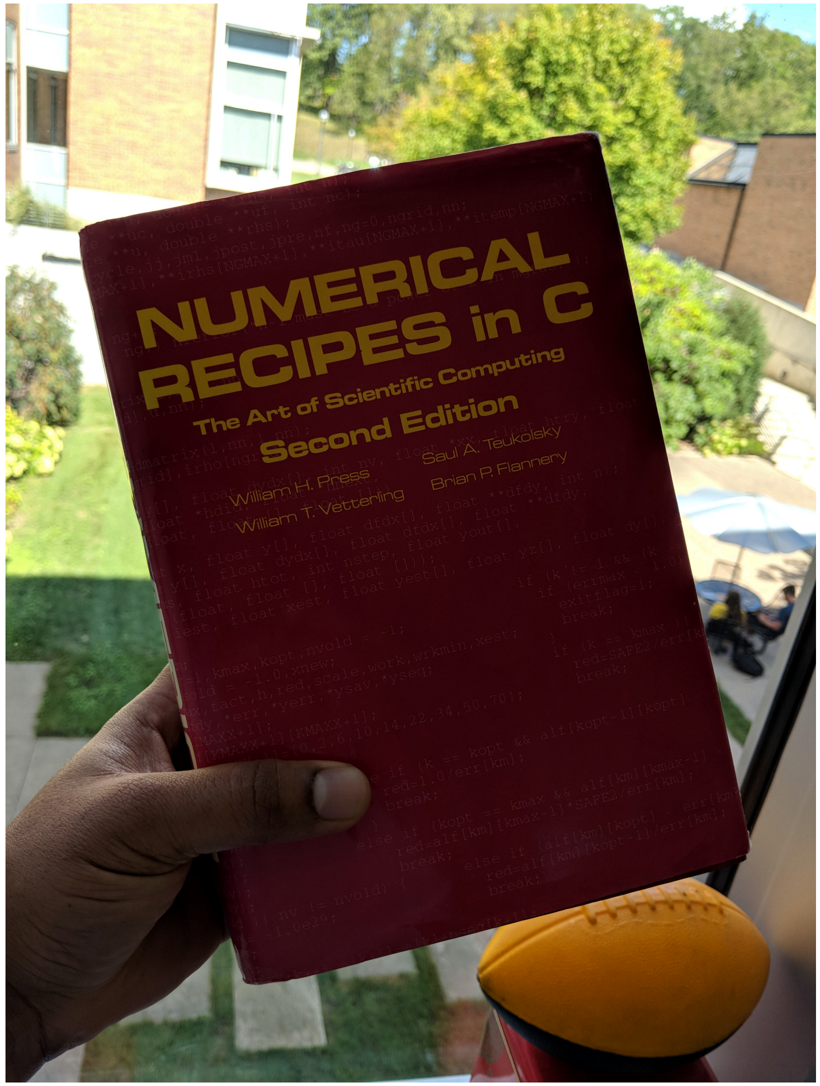
http://astro.physics.ncsu.edu/urca/course_files/Lesson15/index.html

Euler Integration

Integral over timestep approximated as $$\int_{t_0}^{t_0+h} f(t, y(t))dt \approx hf(t_0, y(t_0))$$

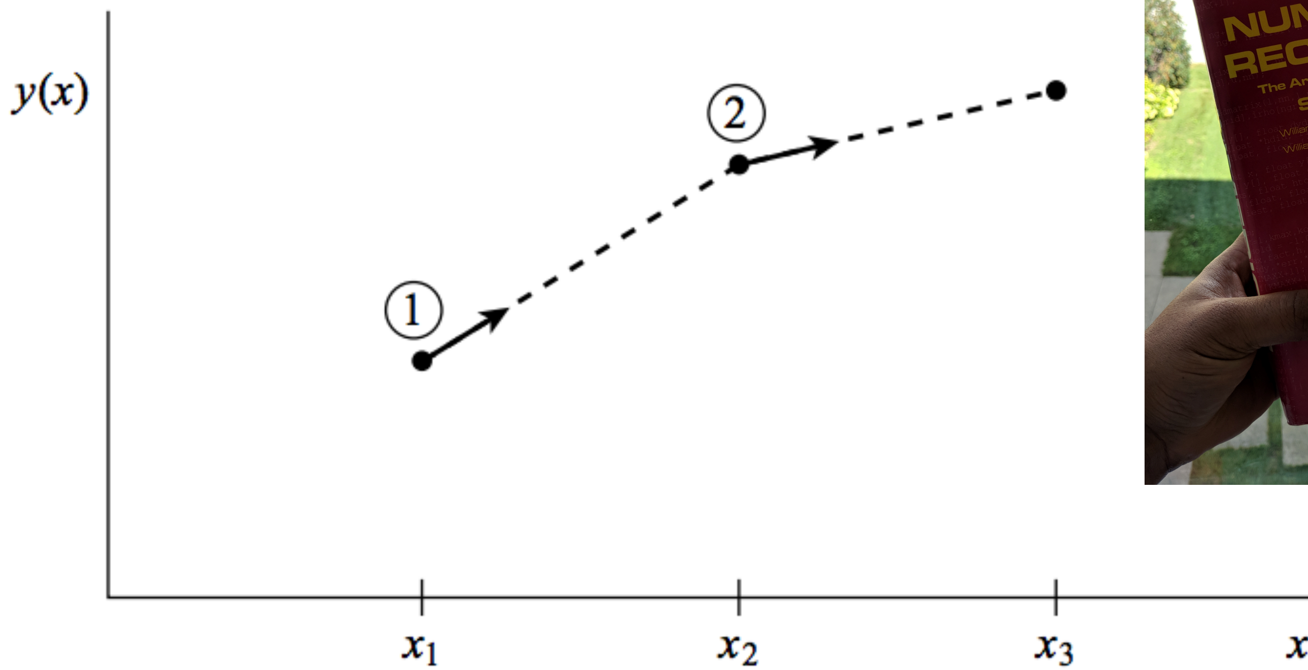


http://astro.physics.ncsu.edu/urca/course_files/Lesson15/index.html



The unofficial textbook
of this lecture

Example Euler integration of
2D point over 2 time steps



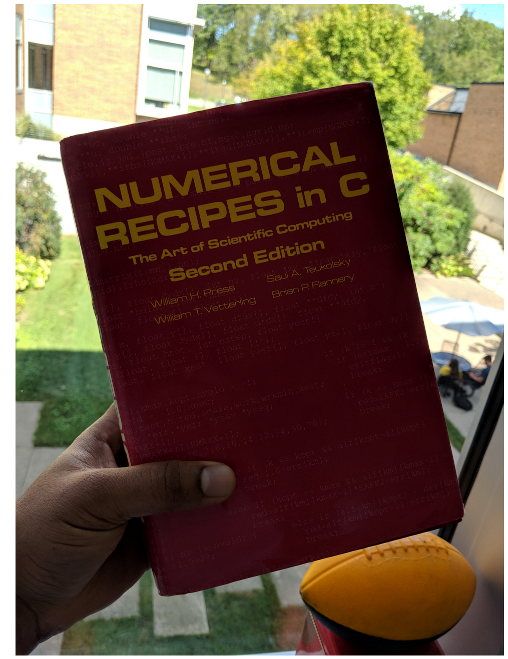


Figure 16.1.1. Euler's method. In this simplest (and least accurate) method for integrating an ODE, the derivative at the starting point of each interval is extrapolated to find the next function value. The method has first-order accuracy.

Second-order state

Second-order state

Reminder:

- State in Newtonian physics has both position (θ) and velocity ($\dot{\theta}$)

$$\boxed{\begin{aligned}\theta_{t+\Delta\theta} &= \theta_t + \dot{\theta}_t \Delta\theta \\ \dot{\theta}_{t+\Delta\theta} &= \dot{\theta}_t + \ddot{\theta}_t \Delta\theta\end{aligned}}$$

Euler's method

Advance position using velocity

$$y_{n+1} = y_n + \dot{y}_n \Delta t$$

Advance velocity using acceleration

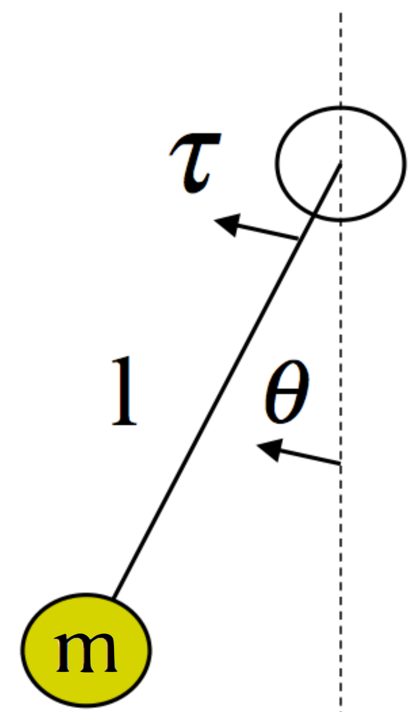
$$\dot{y}_{n+1} = \dot{y}_n + \ddot{y}_n \Delta t$$

$a(y_n)$

ACCELERATION

Advance time

$$t_{n+1} = t_n + \Delta t$$



$$\ddot{\theta} = -\frac{g}{l}\sin(\theta) + \frac{\tau}{ml^2}$$

assumes second order system f(y,y',y'') treats velocity
y' = f(y) and acceleration y'' = f'(y') individually

Euler's method

Advance position using velocity

$$y_{n+1} = y_n + \dot{y}_n \Delta t$$

Advance velocity using acceleration

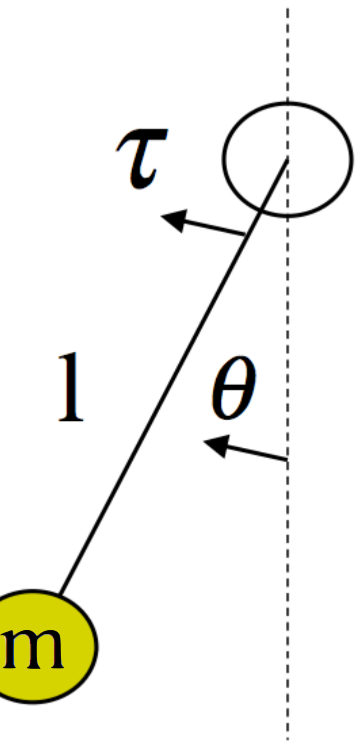
$$\dot{y}_{n+1} = \dot{y}_n + \ddot{y}_n \Delta t$$

$a(y_n)$

ACCELERATION

Advance time

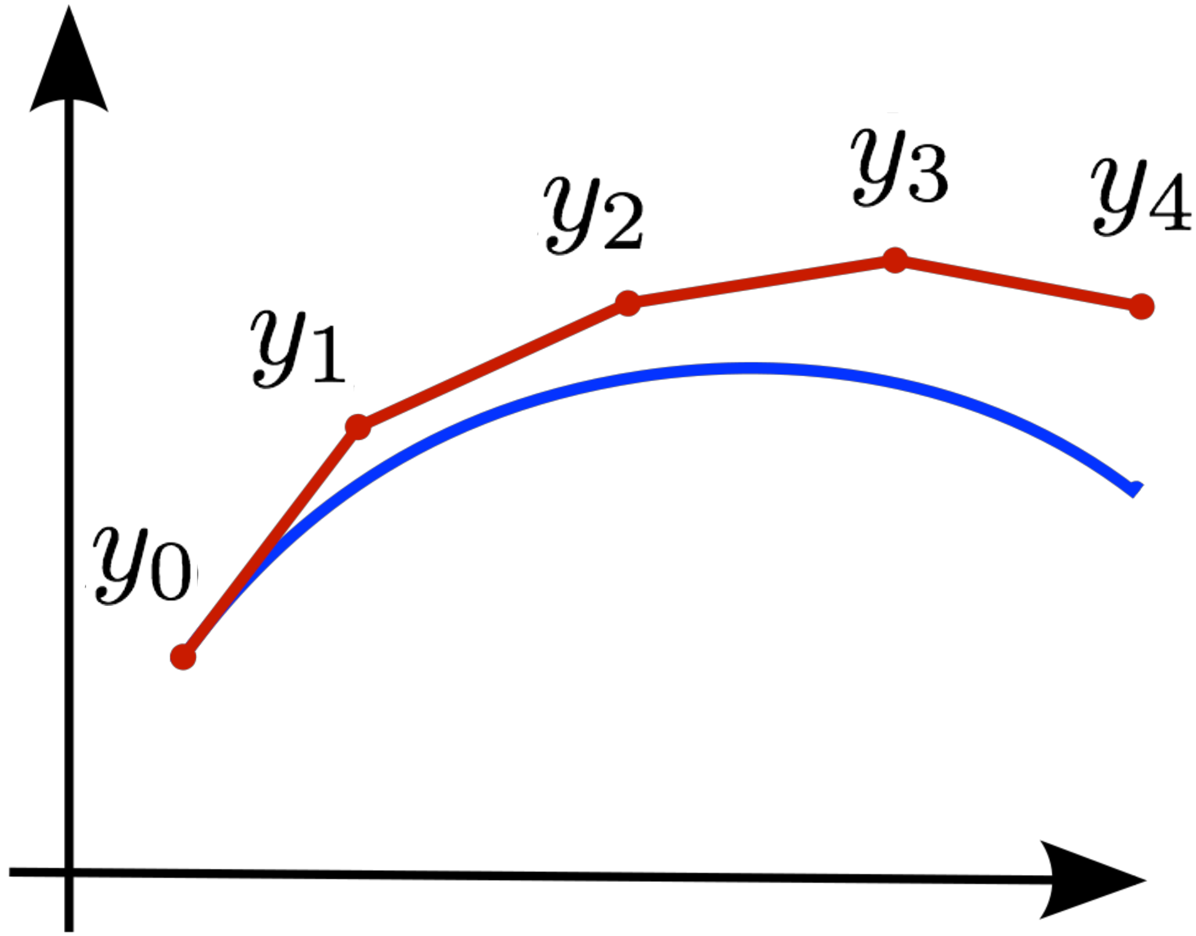
$$t_{n+1} = t_n + \Delta t$$



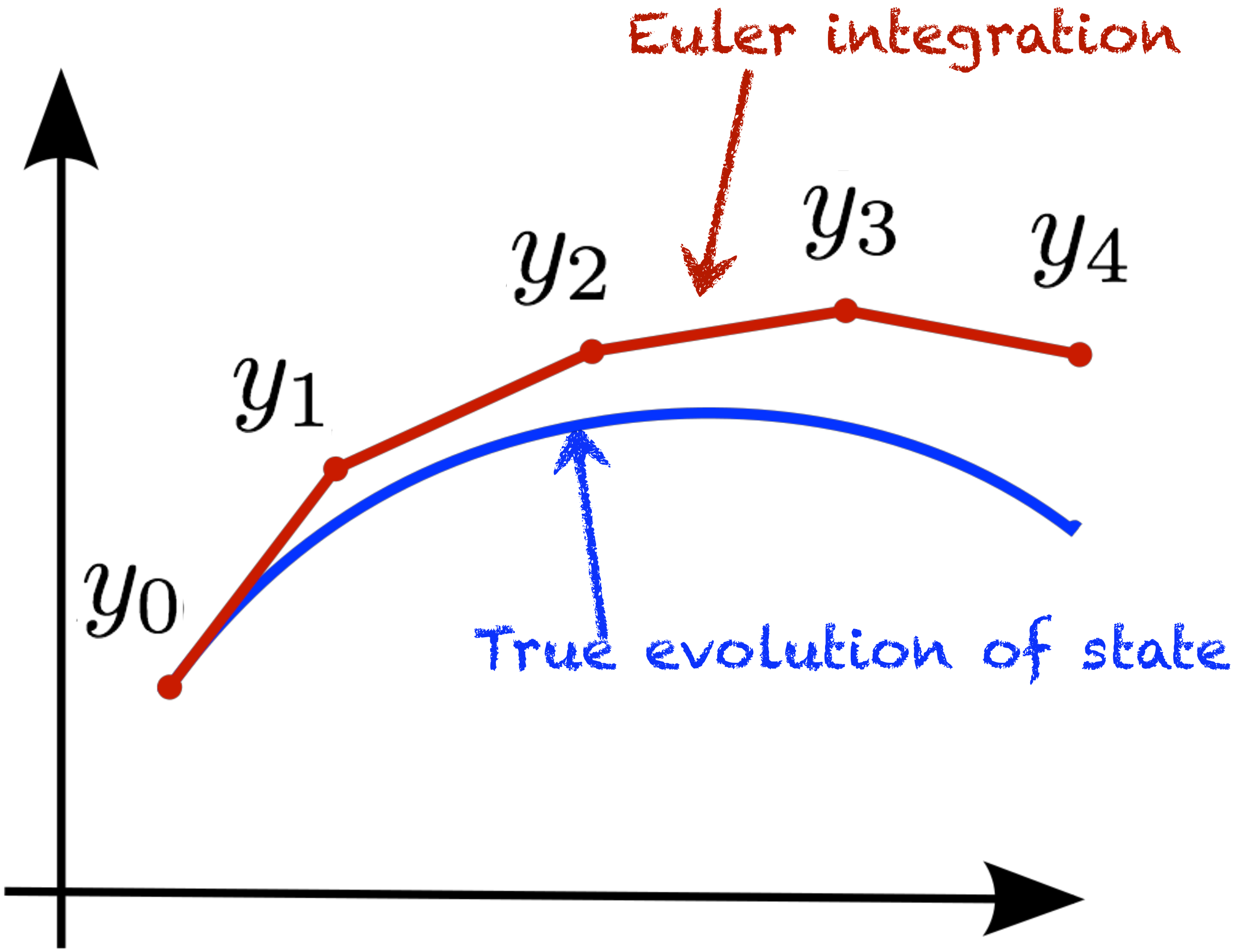
$$\ddot{\theta} = -\frac{g}{l}\sin(\theta) + \frac{\tau}{ml^2}$$

Why is the pendulum going unstable?

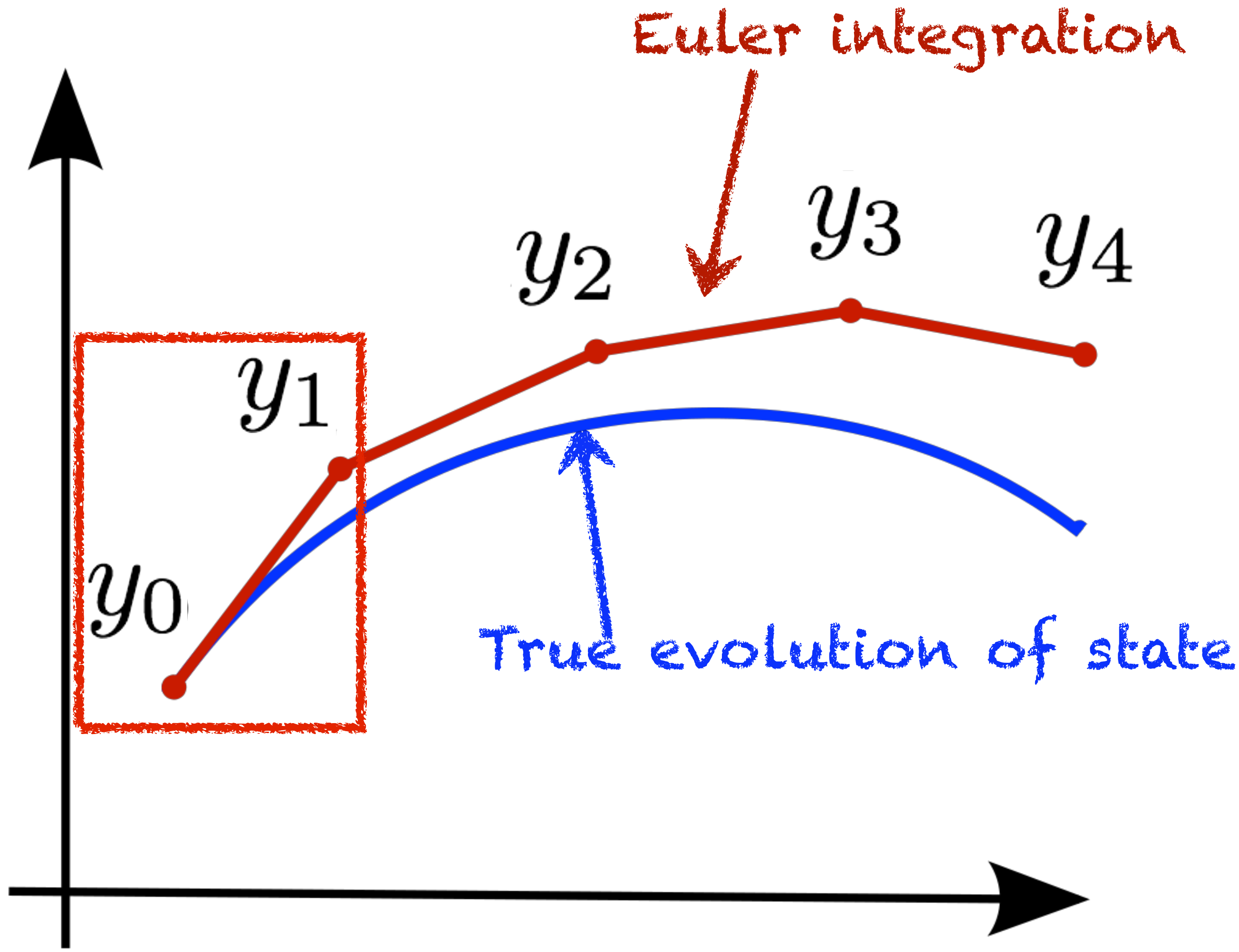
Example Euler integration of 2D point over 4 time steps

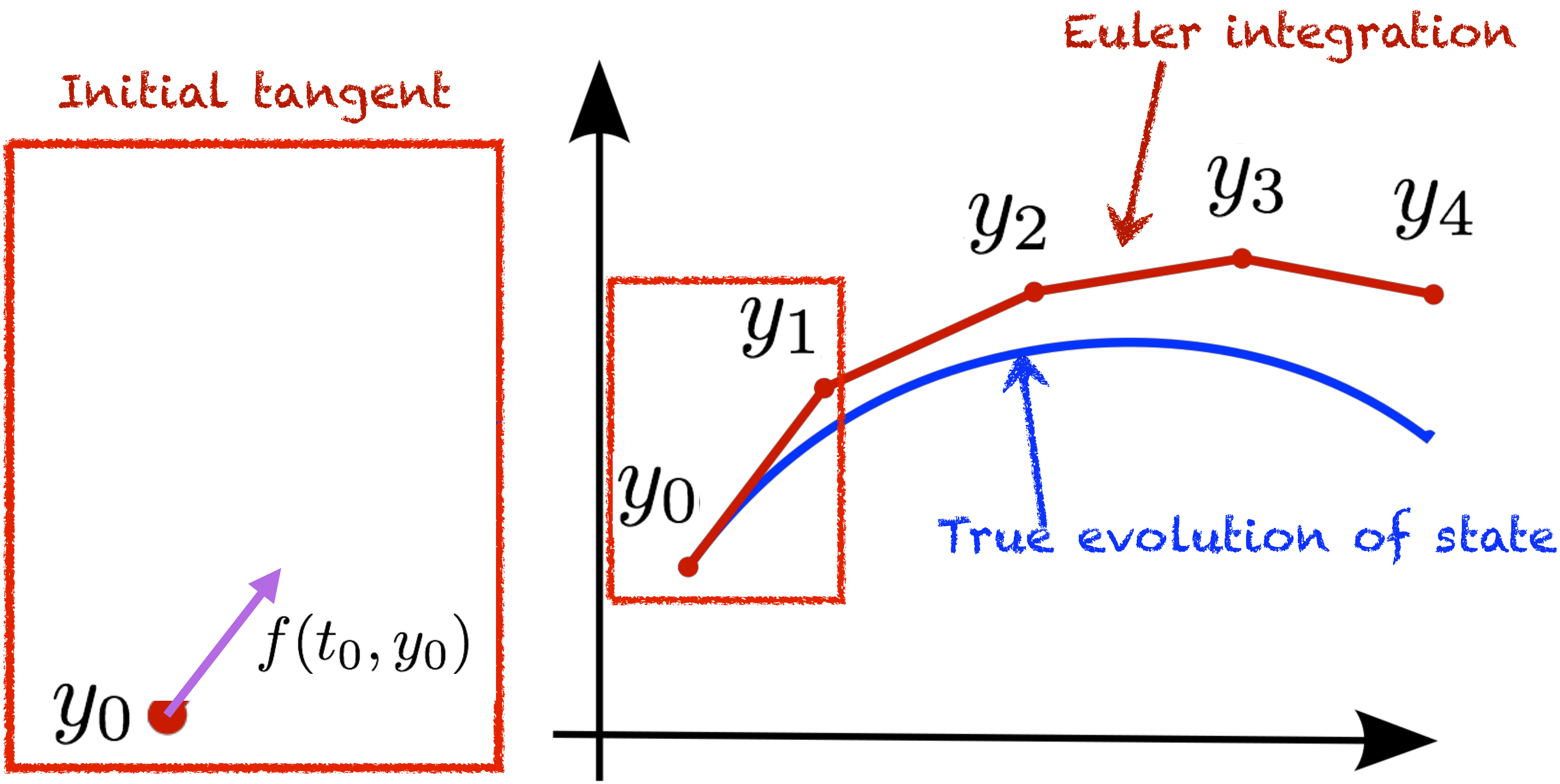


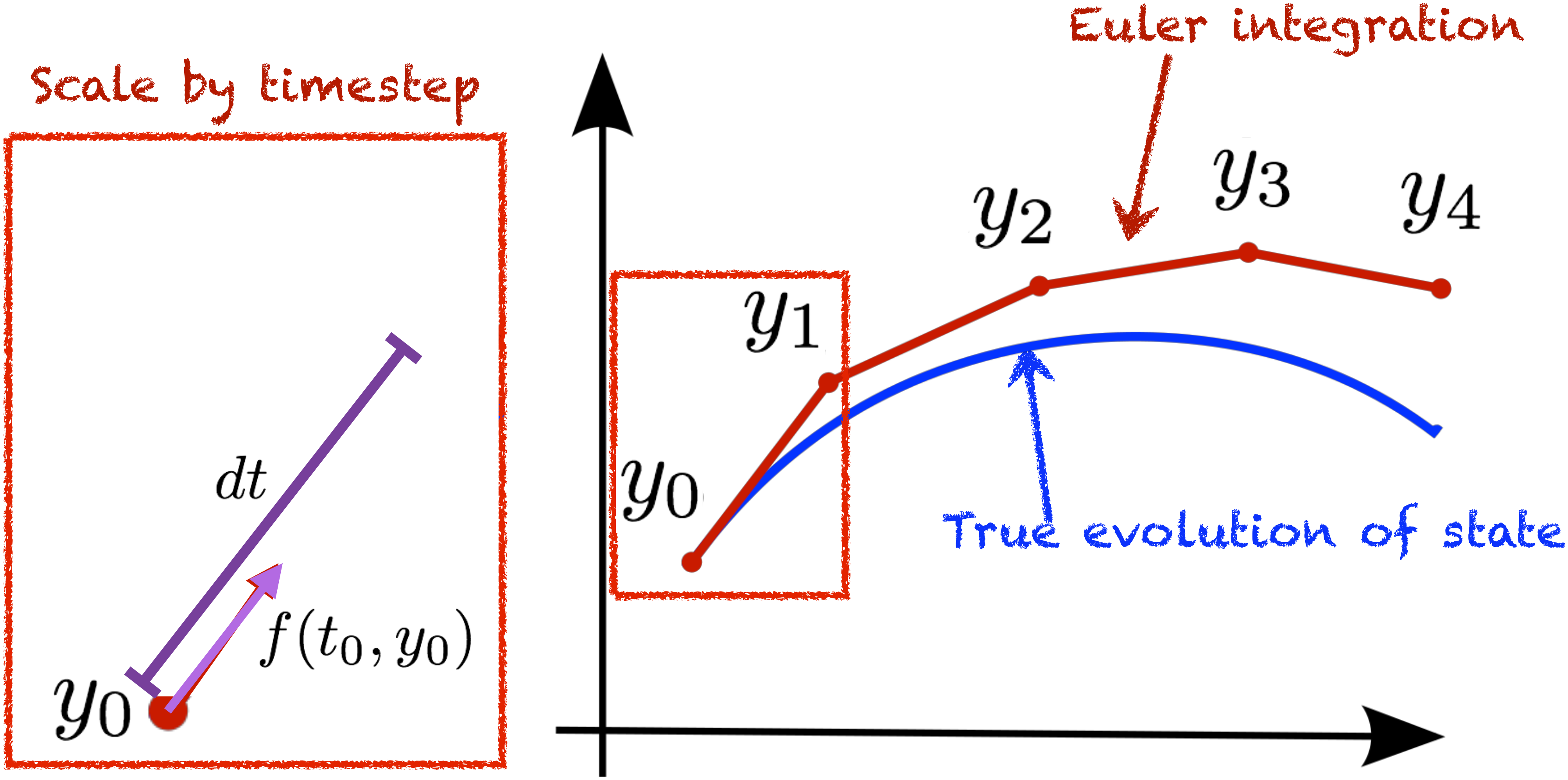
Example Euler integration of 2D point over 4 time steps



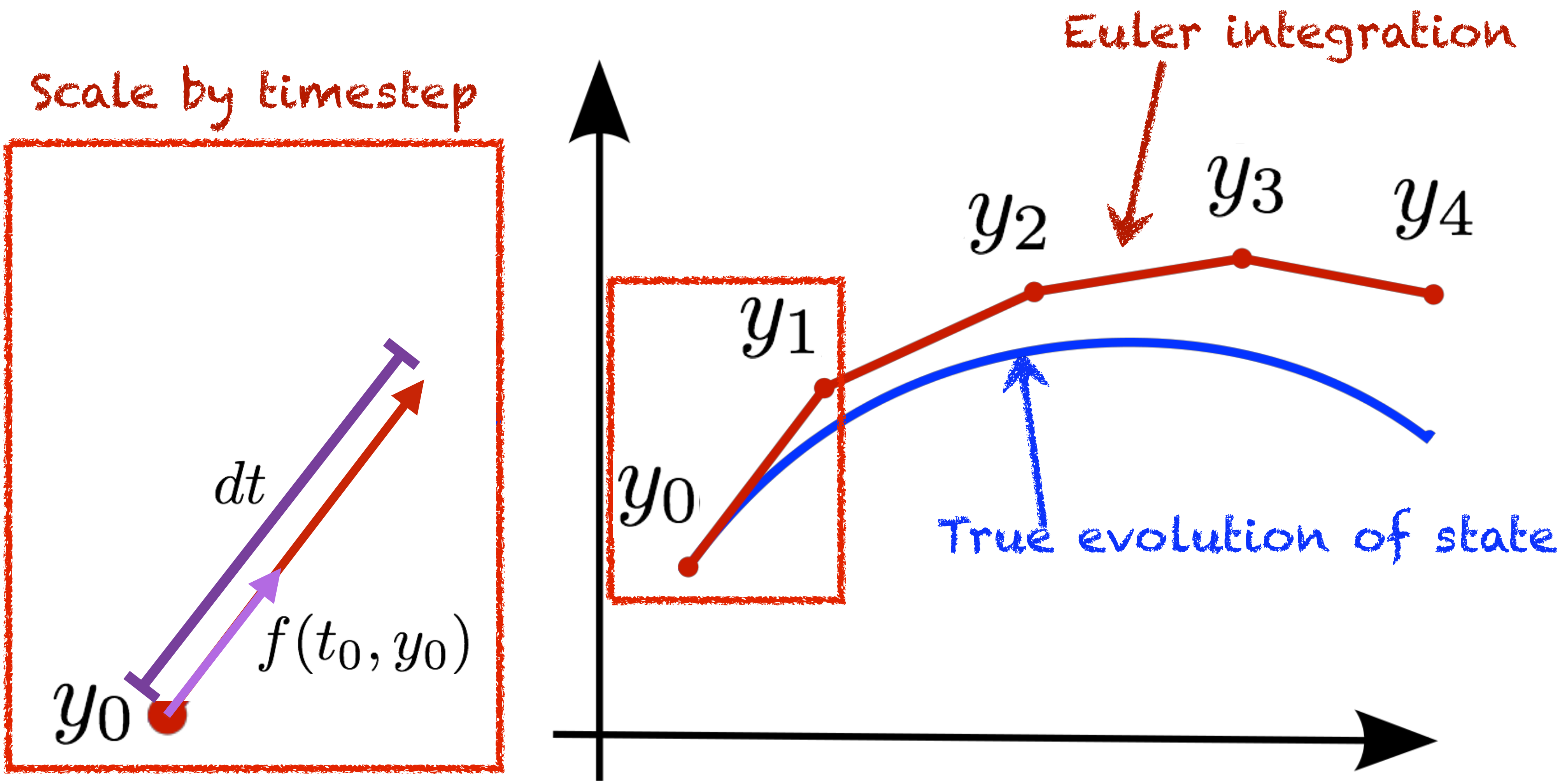
Example Euler integration of 2D point over 4 time steps

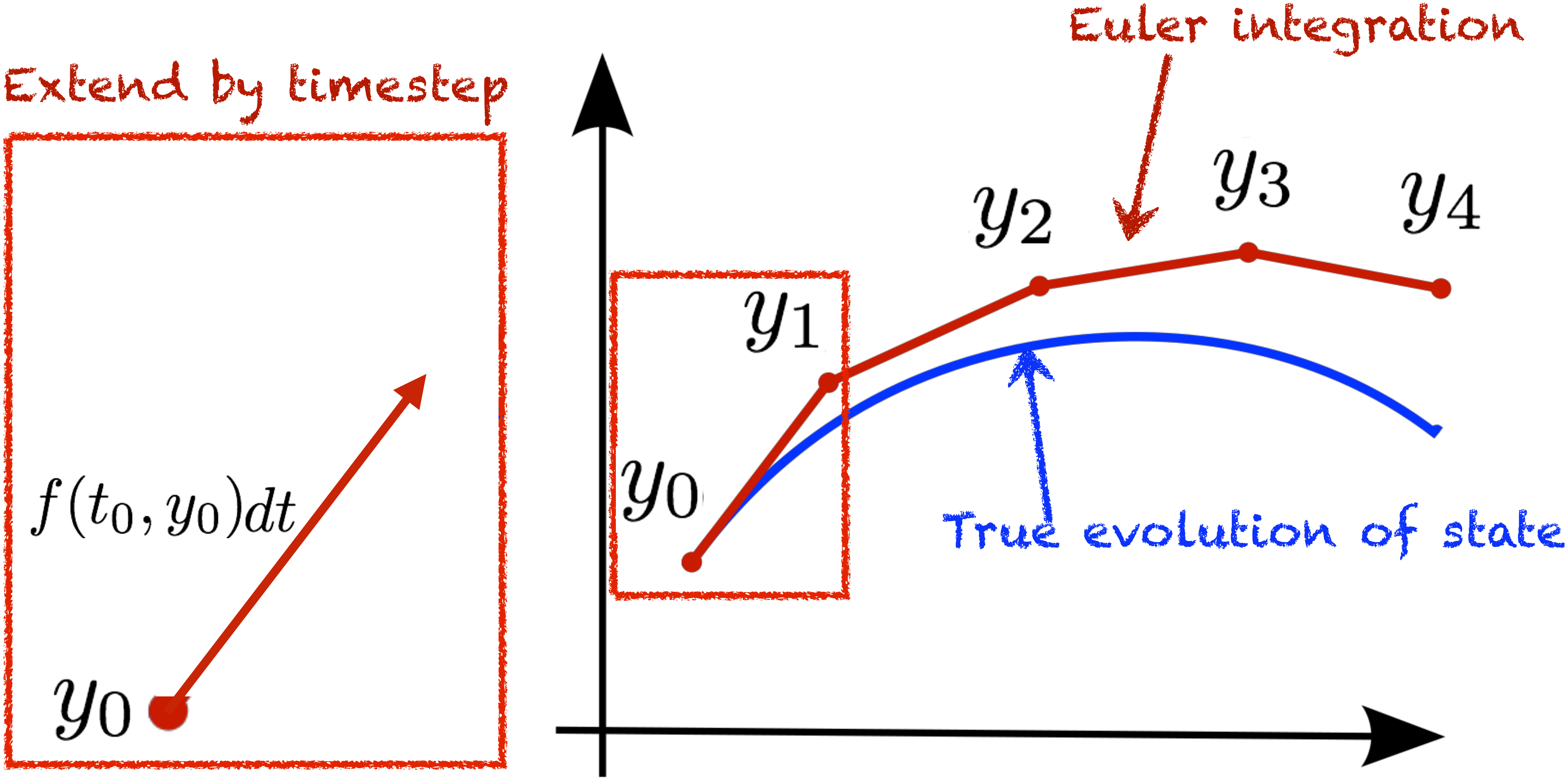


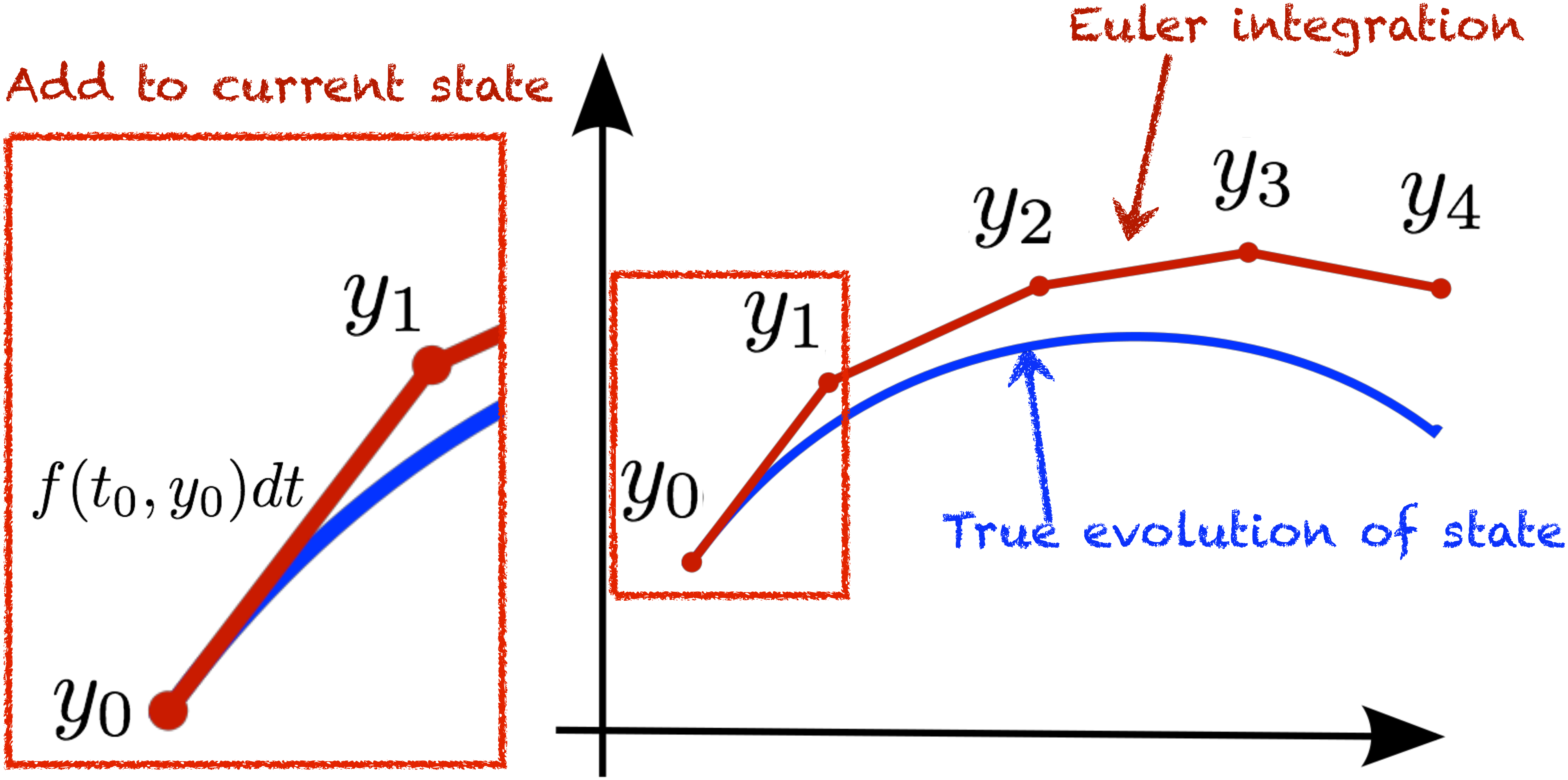

Initial tangent
y_0
$f(t_0, y_0)$
Euler integration
y_0
y_1
y_2
y_3
y_4
True evolution of state

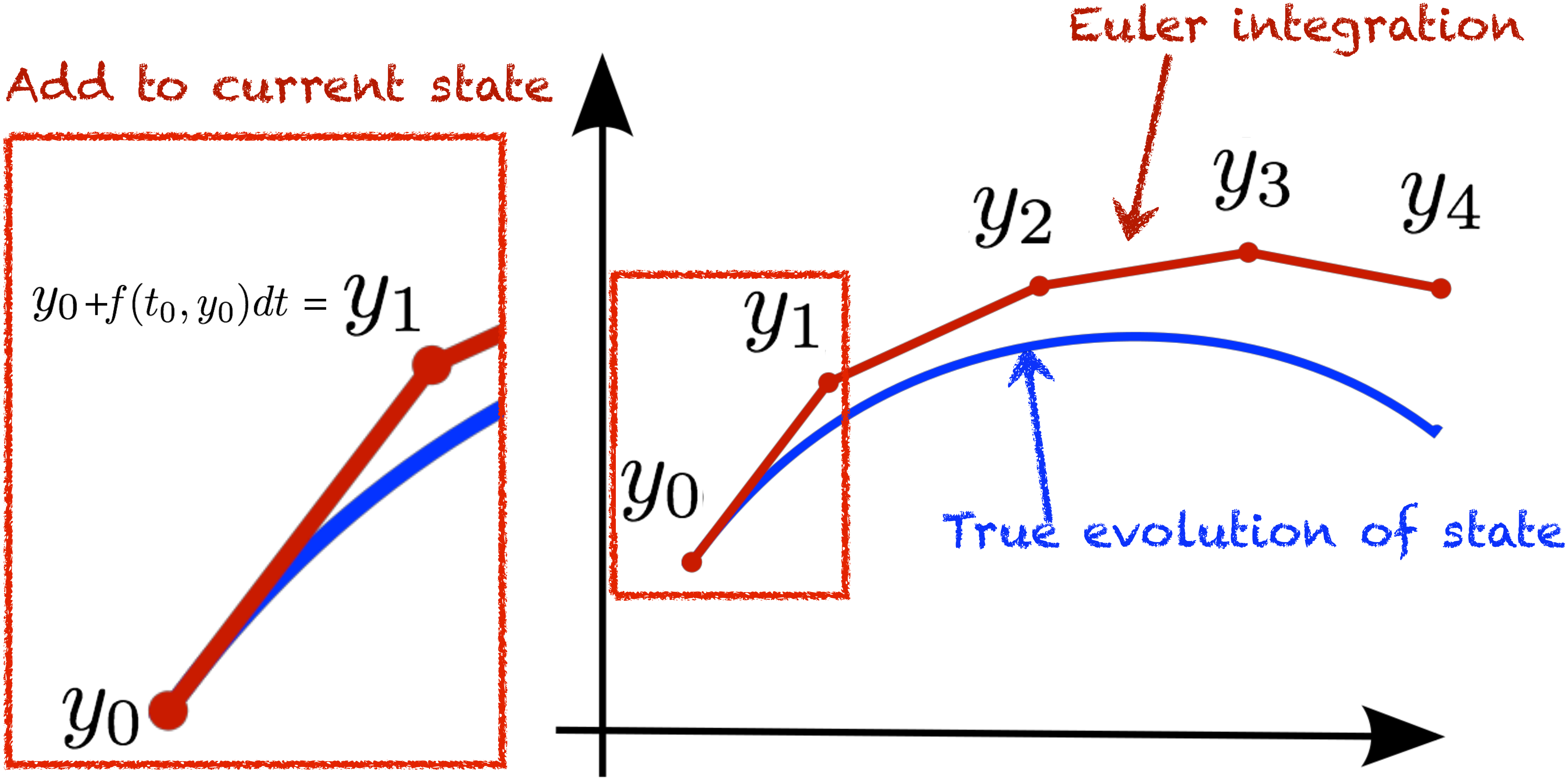


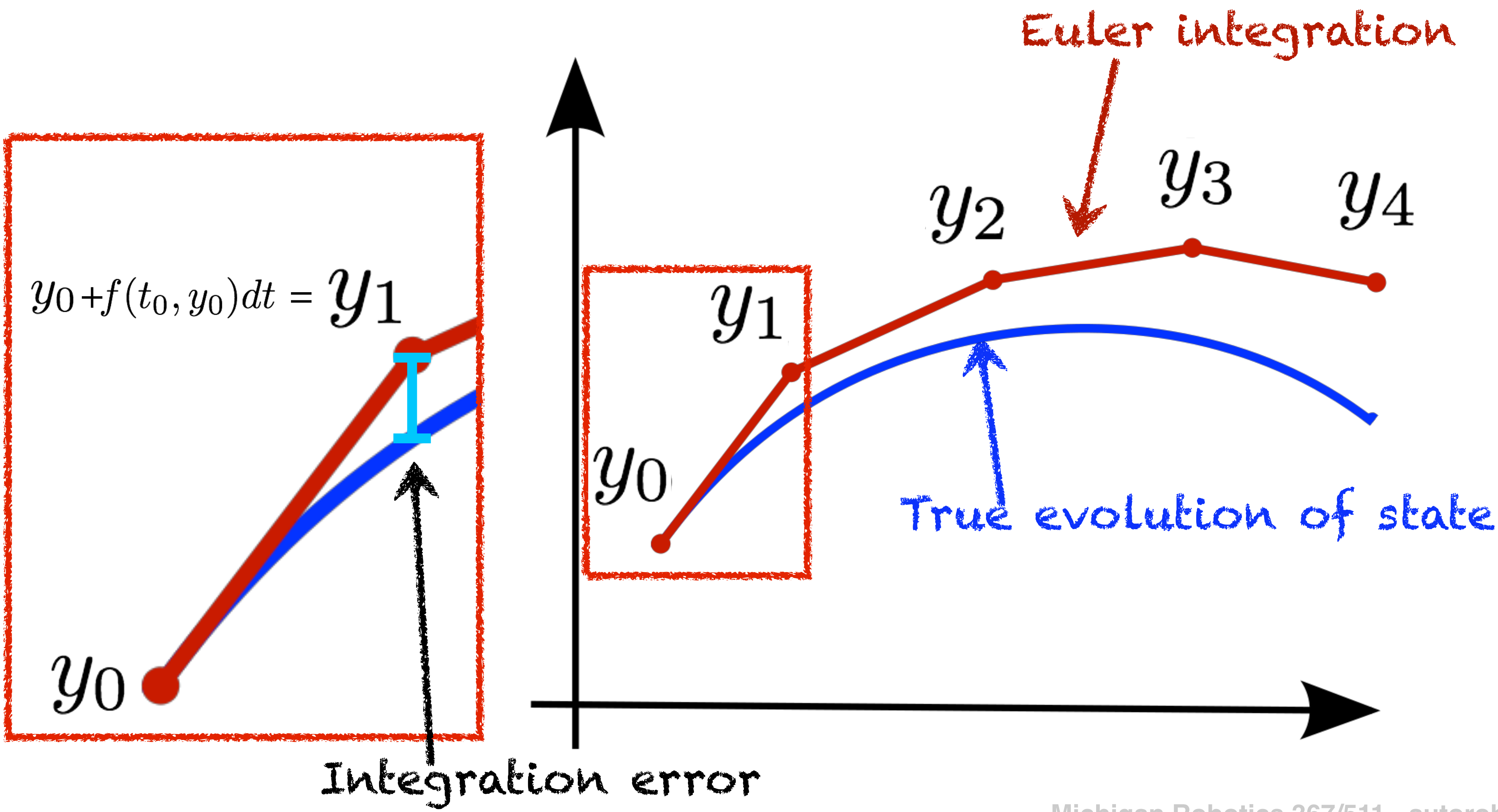
Scale by timestep
dt
$f(t_0, y_0)$
y_0
Euler integration
y_1
y_2
y_3
y_4
y_0
True evolution of state

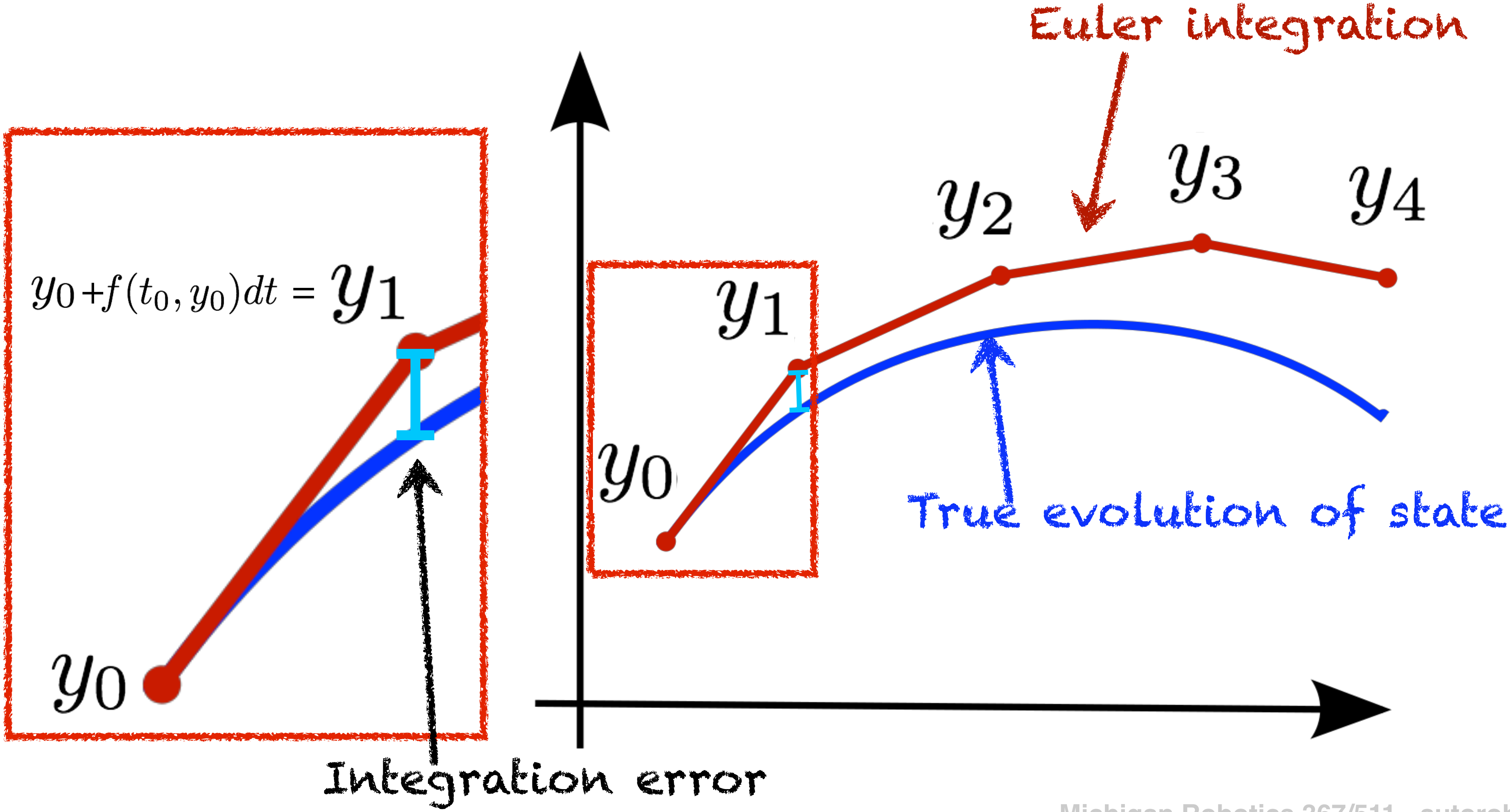

Scale by timestep
dt
$f(t_0, y_0)$
y_0
Euler integration
y_0
y_1
y_2
y_3
y_4
True evolution of state

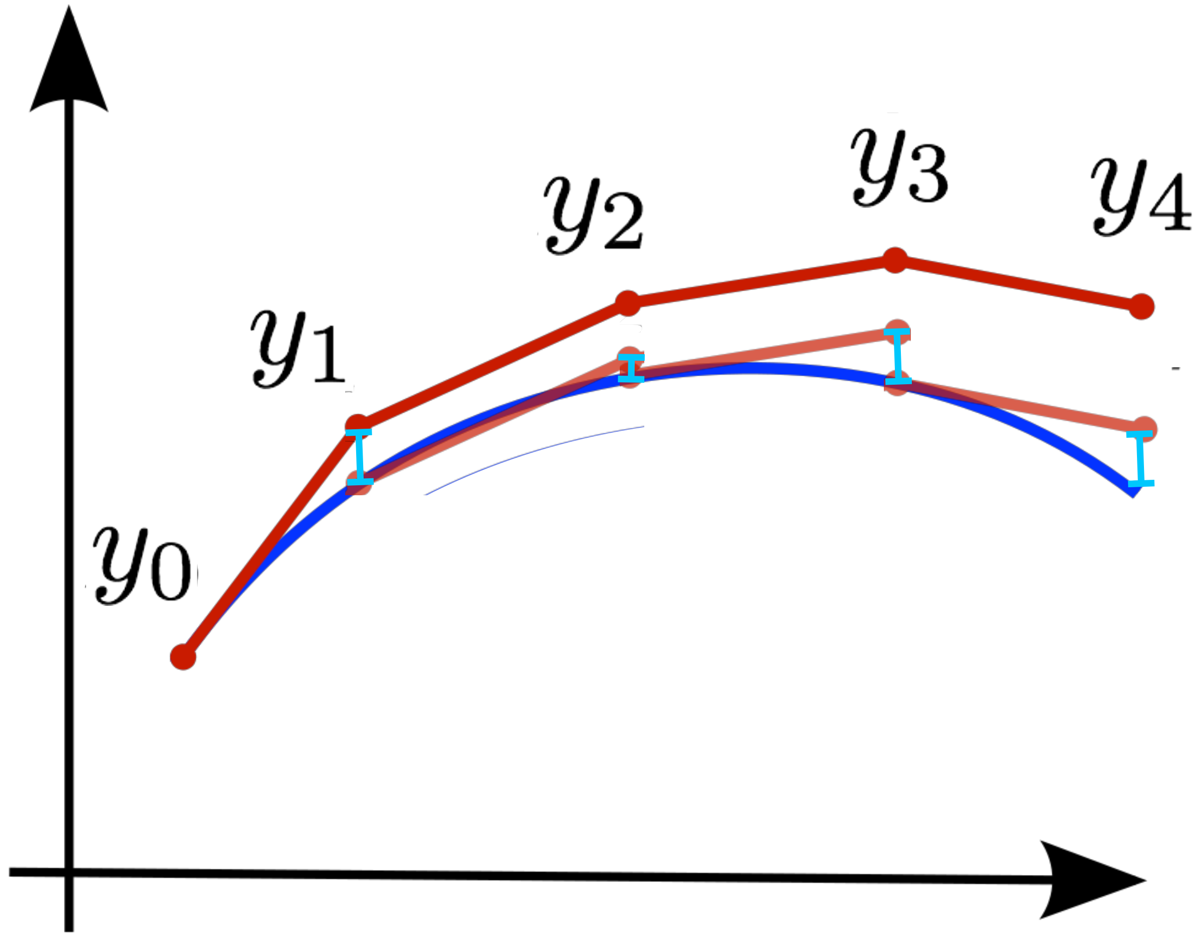

Extend by timestep
$f(t_0, y_0)dt$
y_0
Euler integration
y_0
y_1
y_2
y_3
y_4
True evolution of state


Add to current state
y_1
$f(t_0, y_0)dt$
y_0
Euler integration
y_2
y_3
y_4
y_1
y_0
True evolution of state

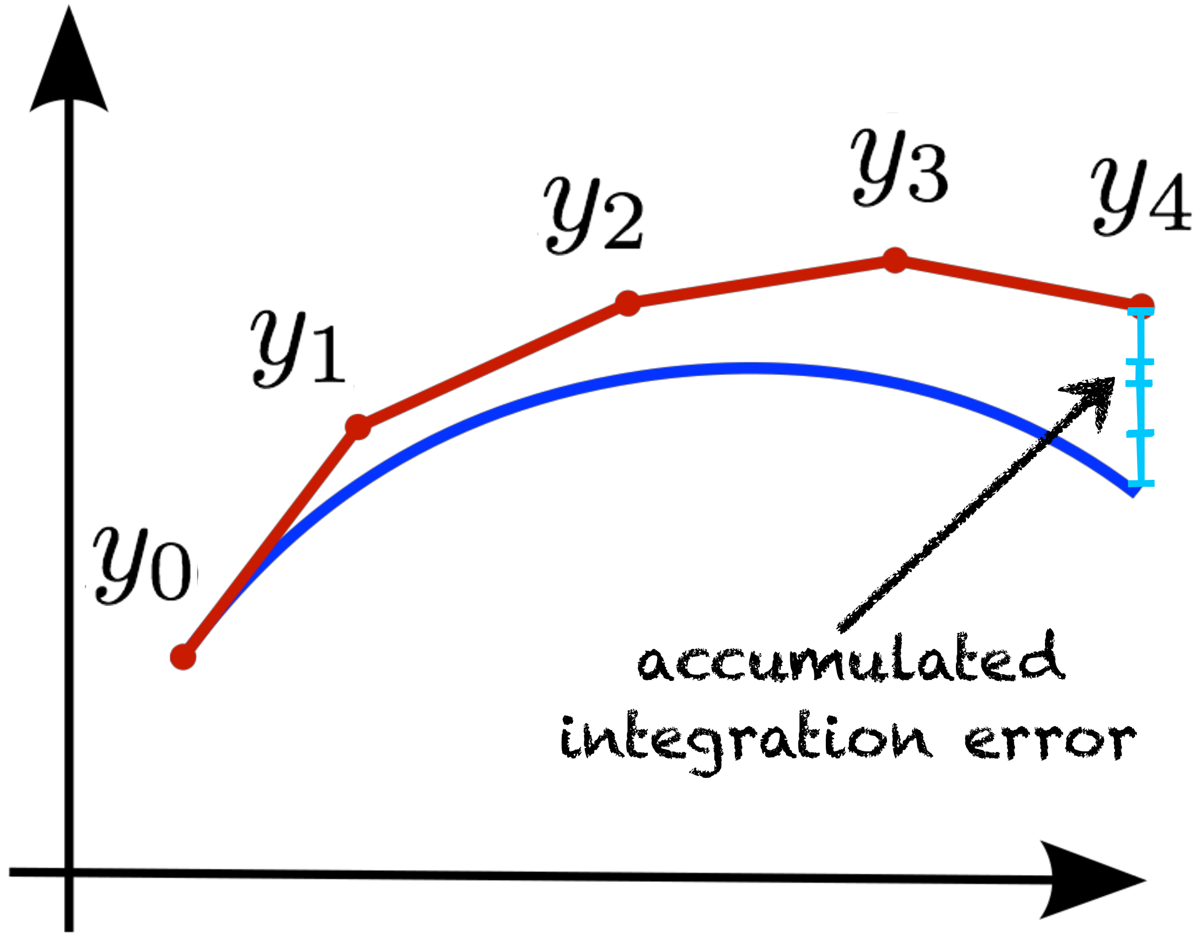

Add to current state
$y_0 + f(t_0, y_0)dt = y_1$
y_0
Euler integration
y_0
y_1
y_2
y_3
y_4
True evolution of state


Euler integration
$y_0 + f(t_0, y_0)dt = y_1$
y_0
y_1
y_2
y_3
y_4
True evolution of state
Integration error


Euler integration
$y_0 + f(t_0, y_0)dt = y_1$
y_1
y_2
y_3
y_4
y_0
y_0
True evolution of state
Integration error

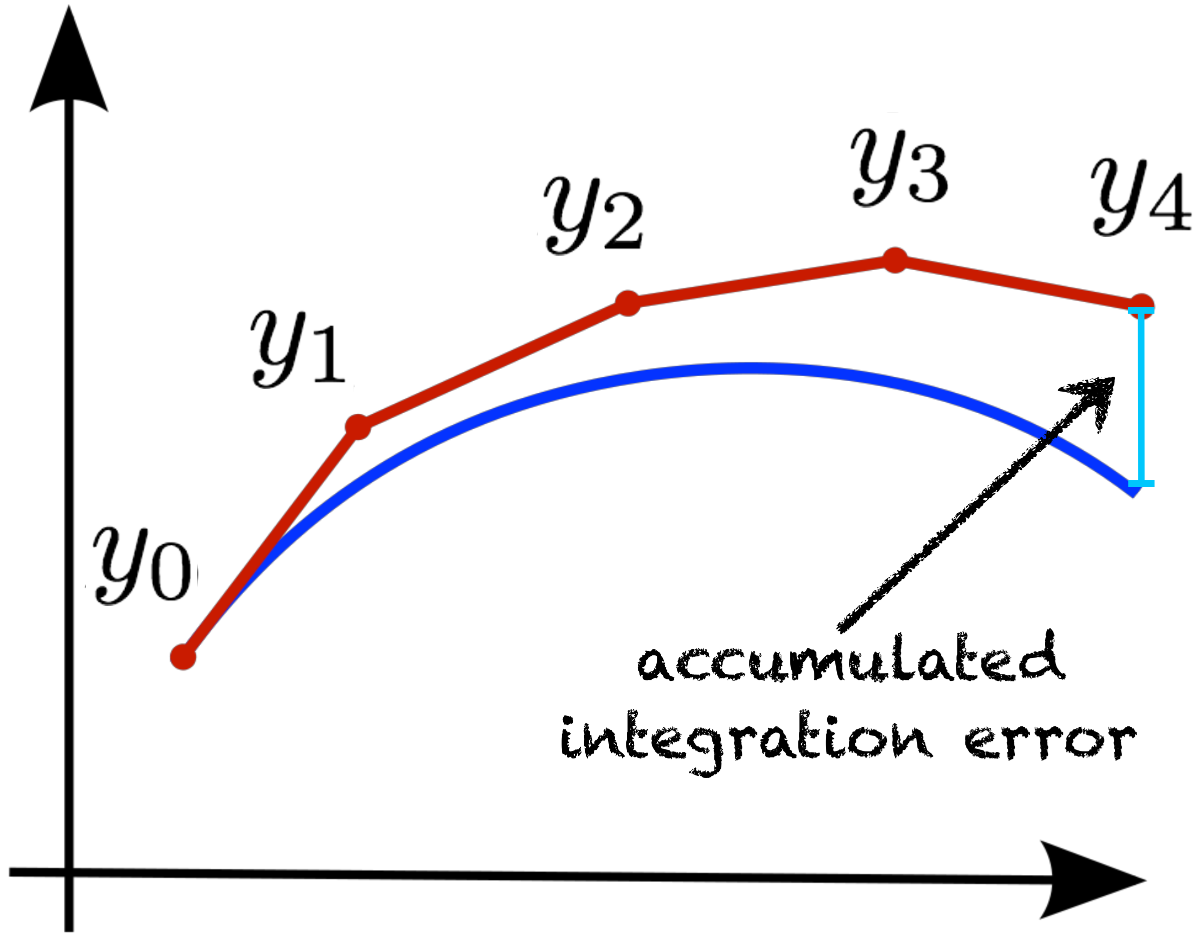


y_0
y_1
y_2
y_3
y_4



y_0
y_1
y_2
y_3
y_4
accumulated
integration error

Can we improve this integration over time?



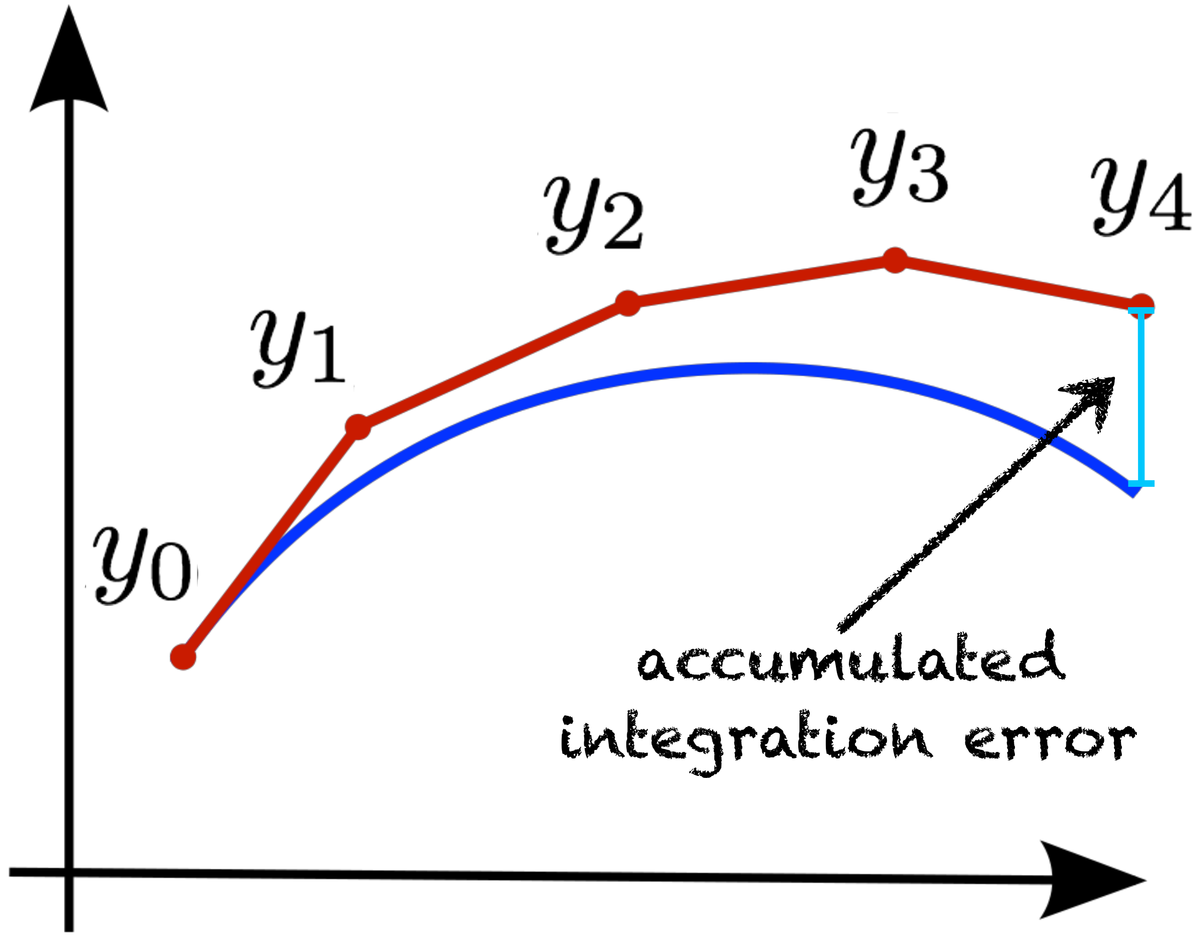
Can we improve this integration over time?

Option 1:

[redacted]

Option 2:

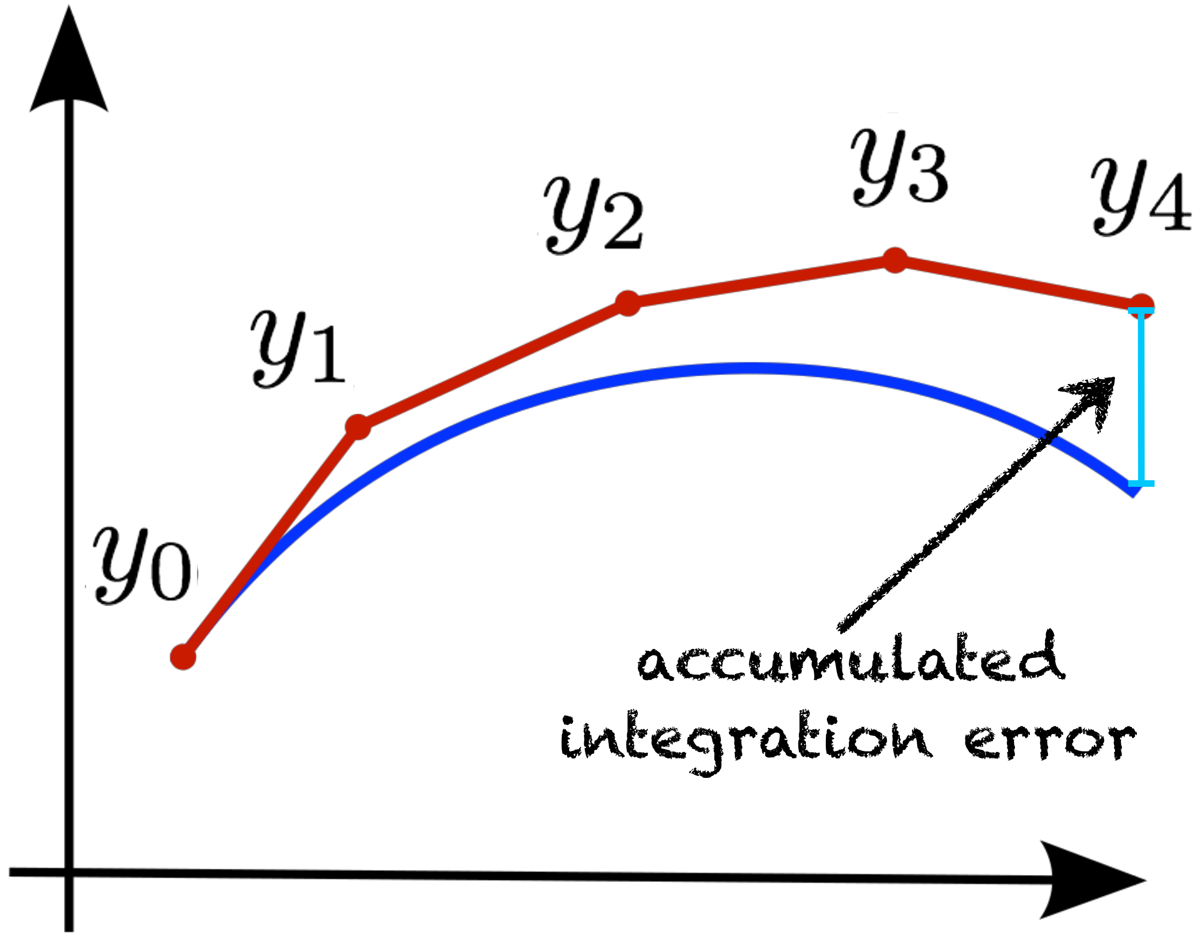
[redacted]



Can we improve this integration over time?

Option 1: Reduce timestep

Option 2: Use a better integrator



Verlet Integration

Verlet integration

For a differential equation of second order of the type $\ddot{\vec{x}}(t) = A(\vec{x}(t))$ with initial conditions $\vec{x}(t_0) = \vec{x}_0$ and $\dot{\vec{x}}(t_0) = \vec{v}_0$, an approximate numerical solution $\vec{x}_n \approx \vec{x}(t_n)$ at the times $t_n = t_0 + n\,\Delta t$ with step size $\Delta t > 0$ can be obtained by the following method:

Advance position $$y_{n+1} = 2y_n - y_{n-1} + a(y_n)h^2$$

Verlet Integration

$$a(y_n) = \frac{\Delta^2 y_n}{\Delta t^2}$$

START WITH DISCRETE TIME APPROXIMATION OF ACCELERATION AT TIME N

Verlet Integration

$$a(y_n) = \frac{\Delta^2 y_n}{\Delta t^2}$$

START WITH DISCRETE TIME APPROXIMATION OF ACCELERATION AT TIME N

$$\approx \frac{\dot{y}_n - \dot{y}_{n-1}}{\Delta t}$$

BREAKDOWN INTO VELOCITIES

Verlet Integration

$$a(y_n) = \frac{\Delta^2 y_n}{\Delta t^2}$$

START WITH DISCRETE TIME APPROXIMATION OF ACCELERATION AT TIME N

$$\approx \frac{\dot{y}_n - \dot{y}_{n-1}}{\Delta t}$$

BREAKDOWN INTO VELOCITIES

$$\approx \frac{\frac{y_{n+1} - y_n}{\Delta t} - \frac{y_n - y_{n-1}}{\Delta t}}{\Delta t}$$

AND THEN POSITIONS

Verlet Integration

$$a(y_n) = \frac{\Delta^2 y_n}{\Delta t^2}$$

START WITH DISCRETE TIME APPROXIMATION OF ACCELERATION AT TIME N

$$\approx \frac{\dot{y}_n - \dot{y}_{n-1}}{\Delta t}$$

BREAKDOWN INTO VELOCITIES

$$\approx \frac{\frac{y_{n+1} - y_n}{\Delta t} - \frac{y_n - y_{n-1}}{\Delta t}}{\Delta t}$$

AND THEN POSITIONS

$$= \frac{y_{n+1} - 2y_n + y_{n-1}}{\Delta t^2}$$

DO SOME ALGEBRA

Verlet Integration

$$a(y_n) = \frac{\Delta^2 y_n}{\Delta t^2}$$ START WITH DISCRETE TIME APPROXIMATION OF ACCELERATION AT TIME N

$$\approx \frac{\dot{y}_n - \dot{y}_{n-1}}{\Delta t}$$ BREAKDOWN INTO VELOCITIES

$$\approx \frac{\frac{y_{n+1} - y_n}{\Delta t} - \frac{y_n - y_{n-1}}{\Delta t}}{\Delta t}$$ AND THEN POSITIONS

$$= \frac{y_{n+1} - 2y_n + y_{n-1}}{\Delta t^2}$$ DO SOME ALGEBRA

SOLVE FOR NEXT STATE $$\Rightarrow \boxed{y_{n+1} \approx 2y_n - y_{n-1} + a(y_n)\Delta t^2}$$

Verlet Integration

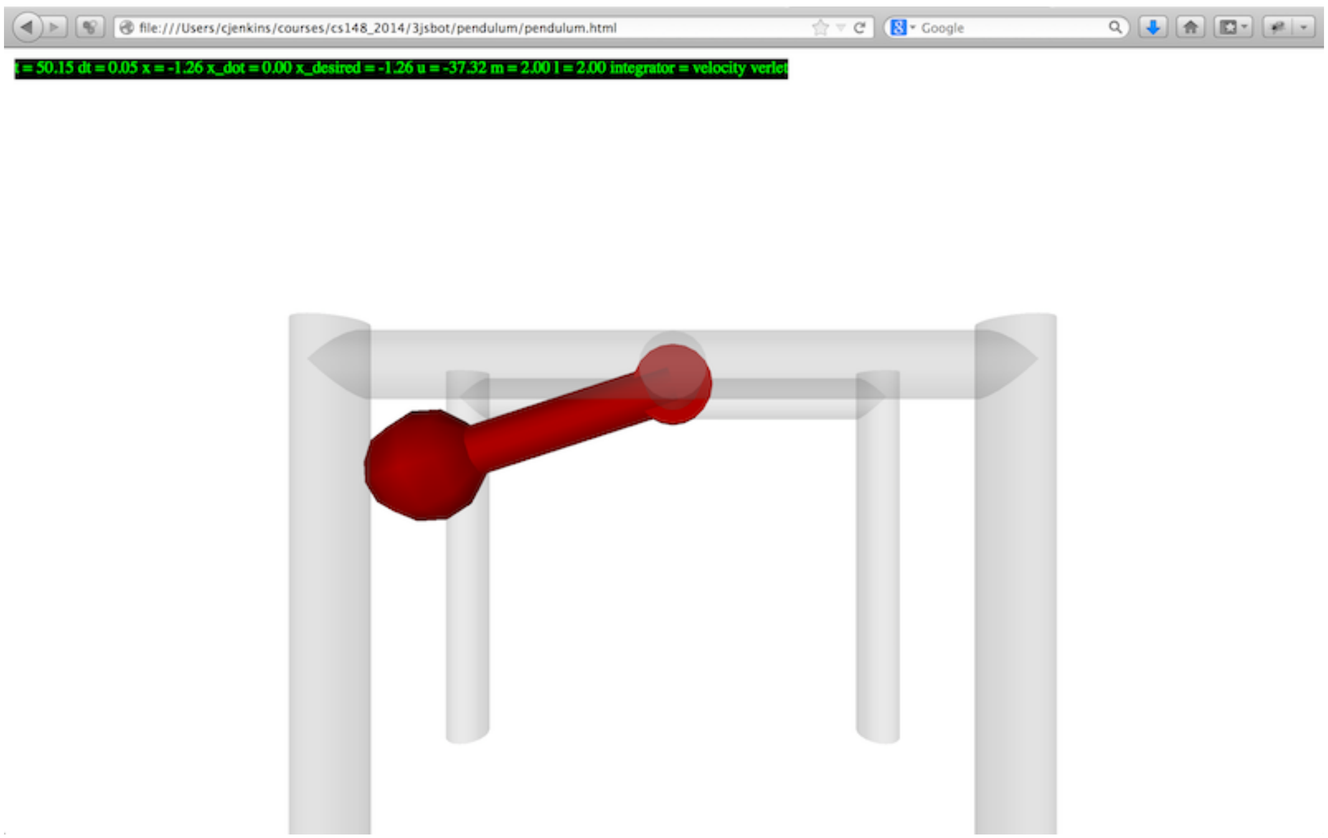
For a differential equation of second order of the type $\ddot{\vec{x}}(t) = A(\vec{x}(t))$ with initial conditions $\vec{x}(t_0) = \vec{x}_0$ and $\dot{\vec{x}}(t_0) = \vec{v}_0$, an approximate numerical solution $\vec{x}_n \approx \vec{x}(t_n)$ at the times $t_n = t_0 + n\,\Delta t$ with step size $\Delta t > 0$ can be obtained by the following method:

Initialize $$y_1 = y_0 + h\dot{y}_0 + h^2\frac{1}{2}a(y_0)$$

Advance position $$y_{n+1} = 2y_n - y_{n-1} + a(y_n)h^2$$

Do not forget to initialize

Let's see what happens



How does Verlet integrate velocity?

Verlet Integration

For a differential equation of second order of the type $\ddot{\vec{x}}(t) = A(\vec{x}(t))$ with initial conditions $\vec{x}(t_0) = \vec{x}_0$ and $\dot{\vec{x}}(t_0) = \vec{v}_0$, an approximate numerical solution $\vec{x}_n \approx \vec{x}(t_n)$ at the times $t_n = t_0 + n\,\Delta t$ with step size $\Delta t > 0$ can be obtained by the following method:

Initialize

$$y_1 = y_0 + h\dot{y}_0 + h^2 \frac{1}{2} a(y_0)$$

Advance position

$$y_{n+1} = 2y_n - y_{n-1} + a(y_n)h^2$$

Advance velocity (optional)

$$\dot{y}_n = \frac{y(t+h) - y(t-h)}{2h} + \mathcal{O}(h^2)$$

Verlet Integration

For a differential equation of second order of the type $\ddot{\vec{x}}(t) = A(\vec{x}(t))$ with initial conditions $\vec{x}(t_0) = \vec{x}_0$ and $\dot{\vec{x}}(t_0) = \vec{v}_0$, an approximate numerical solution $\vec{x}_n \approx \vec{x}(t_n)$ at the times $t_n = t_0 + n\,\Delta t$ with step size $\Delta t > 0$ can be obtained by the following method:

IS THERE A CLEANER ALTERNATIVE?

Initialize $$y_1 = y_0 + h\dot{y}_0 + h^2\frac{1}{2}a(y_0)$$

Advance position $$y_{n+1} = 2y_n - y_{n-1} + a(y_n)h^2$$

Advance velocity (optional) $$\dot{y}_n = \frac{y(t+h) - y(t-h)}{2h} + \mathcal{O}(h^2)$$

Velocity Verlet

$$y(t+\Delta t) = y(t) + \dot{y}(t)\Delta t + \frac{1}{2}a(t)\Delta t^2$$

$$\dot{y}(t+\Delta t) = \ddot{y}(t) + \frac{a(t) + a(t+\Delta t)}{2}\Delta t$$

ASSUMES THAT ACCELERATION $a(t+\Delta t)$
ONLY DEPENDS ON POSITION $y(t+\Delta t)$

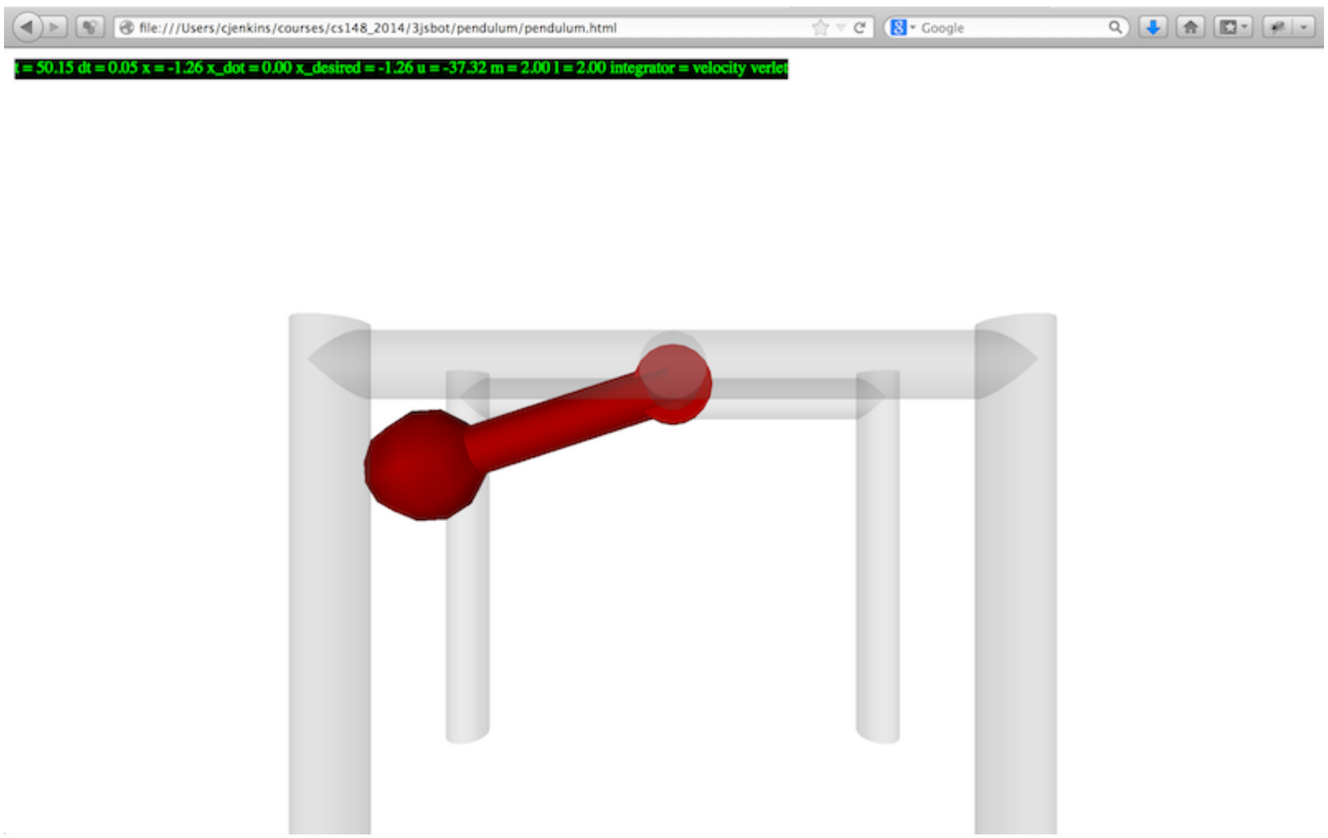
Velocity Verlet

$$x(t + \Delta t) = x(t) + v(t)\Delta t + \frac{1}{2}a(t)\Delta t^2$$

$$v(t + \Delta t) = v(t) + \frac{a(t) + a(t + \Delta t)}{2}\Delta t$$

ASSUMES THAT ACCELERATION $a(t + \Delta t)$
ONLY DEPENDS ON POSITION $x(t + \Delta t)$

Let's see what happens



Runge-Kutta

Runge–Kutta methods

From Wikipedia, the free encyclopedia

In numerical analysis, the **Runge–Kutta methods** are a family of implicit and explicit iterative methods, which include the well-known routine called the Euler Method, used in temporal discretization for the approximate solutions of ordinary differential equations.[1] These methods were developed around 1900 by the German mathematicians Carl Runge and Wilhelm Kutta.

Describes integrators that take the form of

$$y_{n+1} = y_n + \sum_{i=1}^{s} b_i k_i,$$

Runge-Kutta

$$y_{n+1} = y_n + \sum_{i=1}^{s} b_i k_i,$$

think of f() as callable function for dy/dt

where

$$\begin{aligned}
k_1 &= hf(t_n, y_n),\\
k_2 &= hf(t_n + c_2 h, y_n + a_{21}k_1),\\
k_3 &= hf(t_n + c_3 h, y_n + a_{31}k_1 + a_{32}k_2),\\
&\vdots\\
k_s &= hf(t_n + c_s h, y_n + a_{s1}k_1 + a_{s2}k_2 + \cdots + a_{s,s-1}k_{s-1})
\end{aligned}$$

Runge-Kutta

Butcher tableau

$$y_{n+1} = y_n + \sum_{i=1}^{s} b_i k_i,$$

where

$$\begin{aligned}
k_1 &= hf(t_n, y_n), \\
k_2 &= hf(t_n + c_2 h, y_n + a_{21} k_1), \\
k_3 &= hf(t_n + c_3 h, y_n + a_{31} k_1 + a_{32} k_2), \\
&\vdots \\
k_s &= hf(t_n + c_s h, y_n + a_{s1} k_1 + a_{s2} k_2 + \cdots + a_{s,s-1} k_{s-1})
\end{aligned}$$

$$\boxed{\begin{array}{c|cccc}
0 & & & & \\
c_2 & a_{21} & & & \\
c_3 & a_{31} & a_{32} & & \\
\vdots & \vdots & & \ddots & \\
c_s & a_{s1} & a_{s2} & \ldots & a_{s,s-1} \\
\hline
 & b_1 & b_2 & \ldots & b_{s-1} \quad b_s
\end{array}}$$

parameterizes family of RK methods

Runge-Kutta

$$y_{n+1} = y_n + \sum_{i=1}^{s} b_i k_i,$$

where

$$
\begin{aligned}
k_1 &= hf(t_n, y_n),\\
k_2 &= hf(t_n + c_2 h, y_n + a_{21}k_1),\\
k_3 &= hf(t_n + c_3 h, y_n + a_{31}k_1 + a_{32}k_2),\\
&\vdots\\
k_s &= hf(t_n + c_s h, y_n + a_{s1}k_1 + a_{s2}k_2 + \cdots + a_{s,s-1}k_{s-1})
\end{aligned}
$$

s : order of RK formula

Butcher tableau

$$
\boxed{
\begin{array}{c|ccccc}
0 & & & & & \\
c_2 & a_{21} & & & & \\
c_3 & a_{31} & a_{32} & & & \\
\vdots & \vdots & & \ddots & & \\
c_s & a_{s1} & a_{s2} & \cdots & a_{s,s-1} & \\
\hline
 & b_1 & b_2 & \cdots & b_{s-1} & b_s
\end{array}
}
$$

parameterizes family of RK methods

Runge-Kutta 1 (RK1)

$$y_{n+1} = y_n + \sum_{i=1}^{s} b_i k_i,$$

where

$$k_1 = hf(t_n, y_n),$$

Euler's Method is first order RK

$$\boxed{\begin{array}{c|c} 0 & \\ \hline & 1 \end{array}}$$

parameters
$s=1$ and $b_1=1$

The Midpoint Method

(one possible RK2)

$$y_{n+1} = y_n + \sum_{i=1}^{s} b_i k_i,$$

where

$s=2$

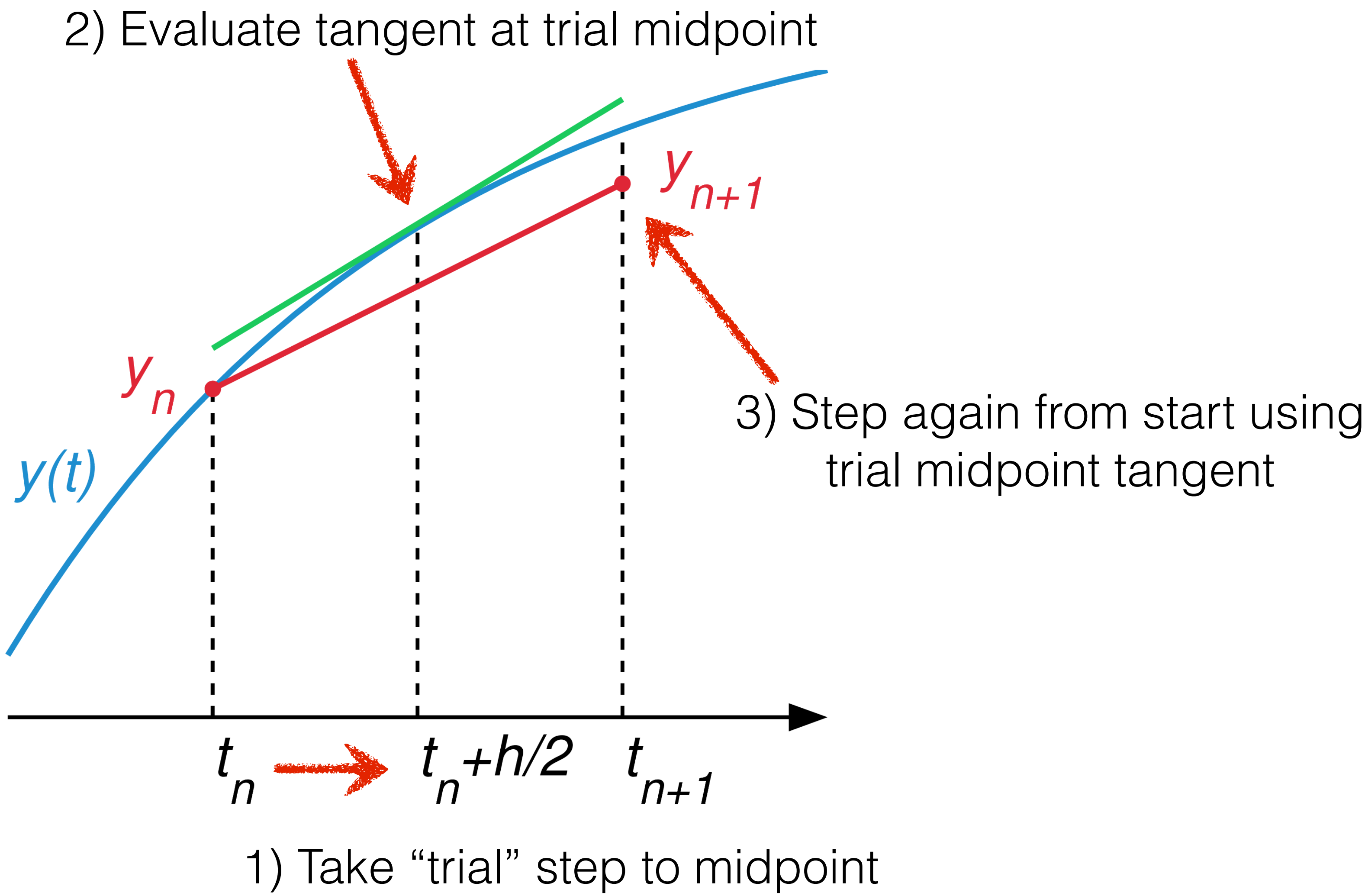
$$k_1 = hf(t_n, y_n), \quad b_1=0$$

$$k_2 = hf(t_n + c_2 h, y_n + a_{21} k_1), \quad b_2=1$$

$c_2=0.5$ $\quad$ $a_{21}=0.5$

Midpoint Method

$$\boxed{\begin{array}{c c|c c}
c_1 & 0 & & \\
 & & & a_{21} \\
c_2 & 1/2 & 1/2 & \\
\hline
 & & 0 & 1 \\
 & & b_1 & b_2
\end{array}}$$


2) Evaluate tangent at trial midpoint
y(t)
y n
y n+1
3) Step again from start using trial midpoint tangent
t n
t n+h/2
t n+1
1) Take “trial” step to midpoint

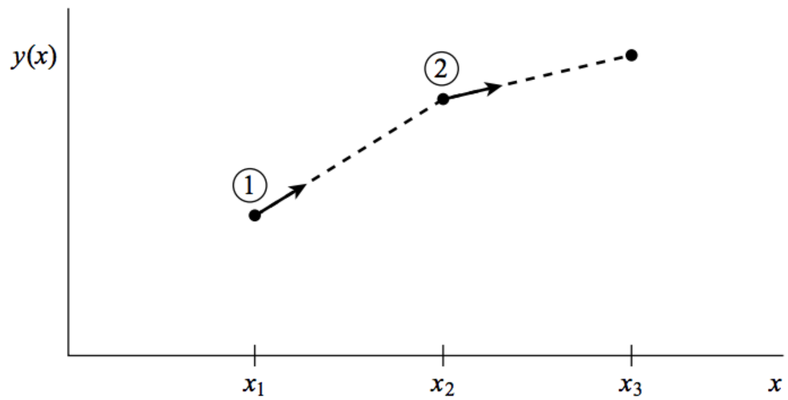
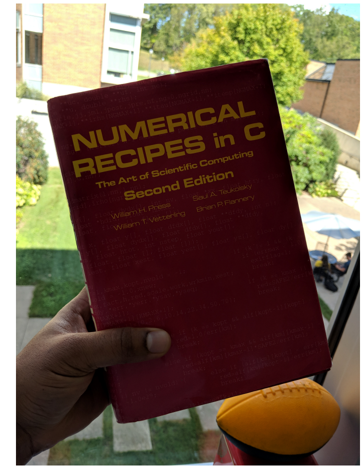


Figure 16.1.1. Euler's method. In this simplest (and least accurate) method for integrating an ODE, the derivative at the starting point of each interval is extrapolated to find the next function value. The method has first-order accuracy.



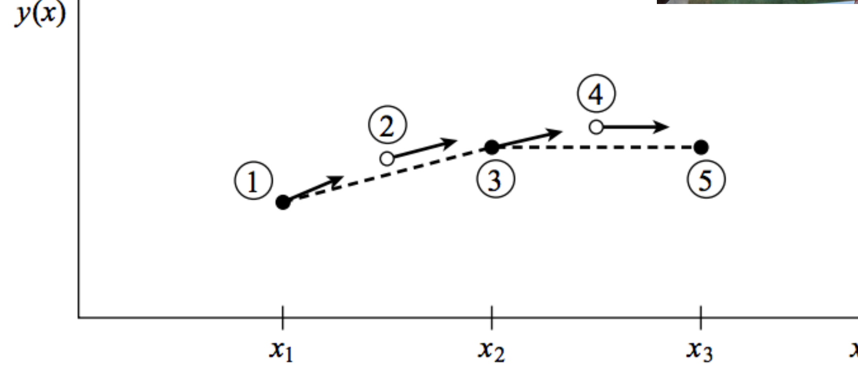
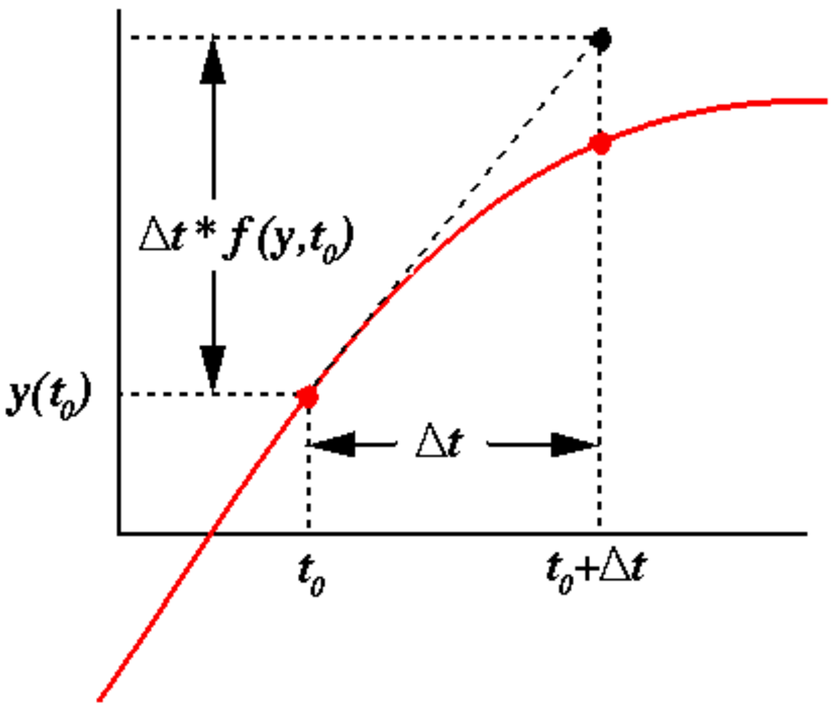
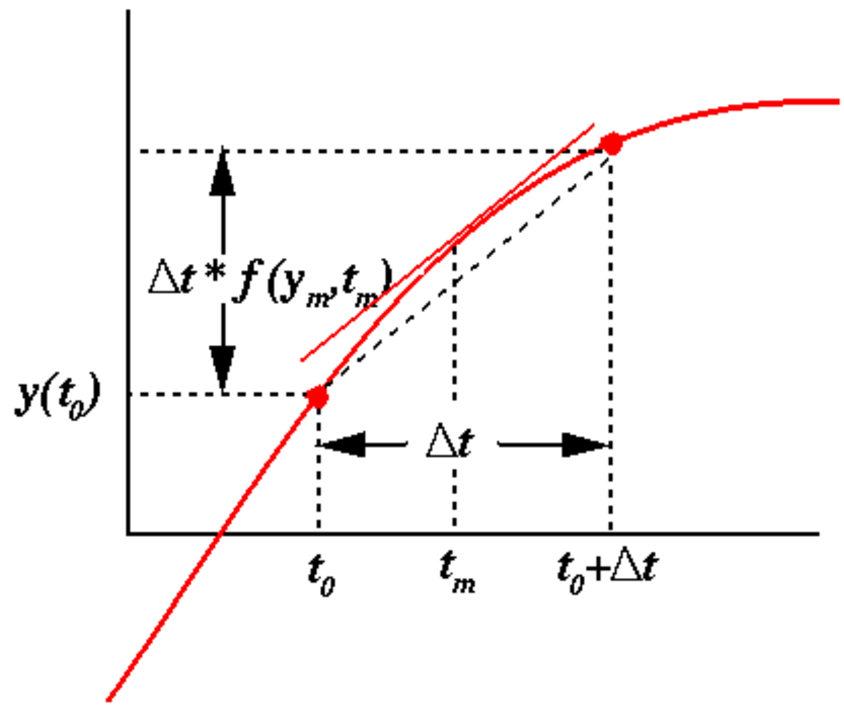


Figure 16.1.2. Midpoint method. Second-order accuracy is obtained by using the initial derivative at each step to find a point halfway across the interval, then using the midpoint derivative across the full width of the interval. In the figure, filled dots represent final function values, while open dots represent function values that are discarded once their derivatives have been calculated and used.

Derivative at midpoint leads to a better approximation of the integration



Euler



Midpoint

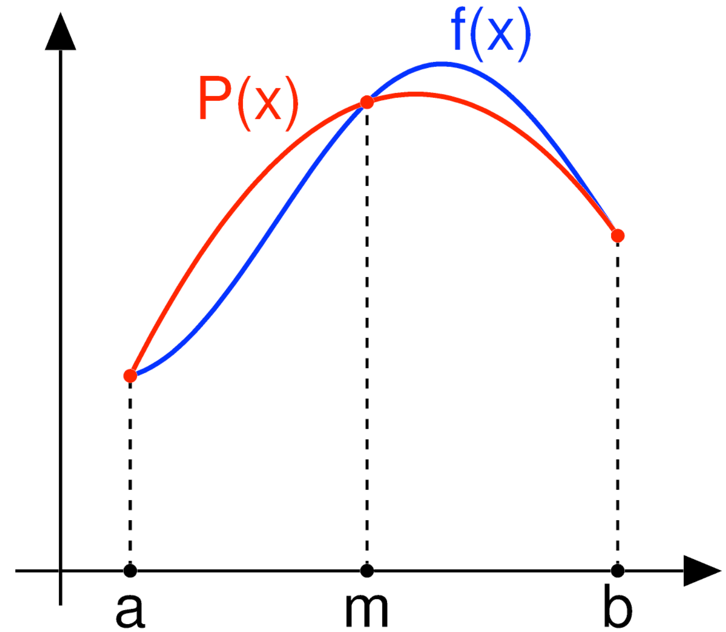
http://astro.physics.ncsu.edu/urca/course_files/Lesson15/index.html

The “RK4”

The "RK4": Simpson's Rule

a widely used version of fourth order Runge-Kutta (known as "RK4") implements Simpson's rule:

$$\int_a^b f(x)\,dx \approx \frac{b-a}{6}\left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right]$$



numerical approximation of definite integrals by evaluating integral extents and midpoint

RK4 parameters: s=4
$a_{21}=1/2$, $a_{32}=1/2$, $a_{43}=1$,
$c_1=0$, $c_2=1/2$, $c_3=1/2$, $c_4=1$
$b_1=1/6$, $b_2=1/3$, $b_3=1/3$, $b_4=1/6$

$$y_{n+1} = y_n + \sum_{i=1}^{s} b_i k_i,$$

$$\begin{array}{c|cccc} 0 & & & & \\ 1/2 & 1/2 & & & \\ 1/2 & 0 & 1/2 & & \\ 1 & 0 & 0 & 1 & \\ \hline & 1/6 & 1/3 & 1/3 & 1/6 \end{array}$$

where

$$\begin{aligned} k_1 &= hf(t_n, y_n), \\ k_2 &= hf(t_n + c_2 h, y_n + a_{21} k_1), \\ k_3 &= hf(t_n + c_3 h, y_n + a_{31} k_1 + a_{32} k_2), \\ &\vdots \\ k_s &= hf(t_n + c_s h, y_n + a_{s1} k_1 + a_{s2} k_2 + \cdots + a_{s,s-1} k_{s-1}) \end{aligned}$$

Runge-Kutta: RK4

Advance state

$$y_{n+1} = y_n + \tfrac{1}{6}h\left(k_1 + 2k_2 + 2k_3 + k_4\right)$$

Advance time

$$t_{n+1} = t_n + h$$

Runge-Kutta: RK4

Advance state: combination of integration trials at points with different tangents

$$y_{n+1} = y_n + \tfrac{1}{6}h\left(k_1 + 2k_2 + 2k_3 + k_4\right)$$

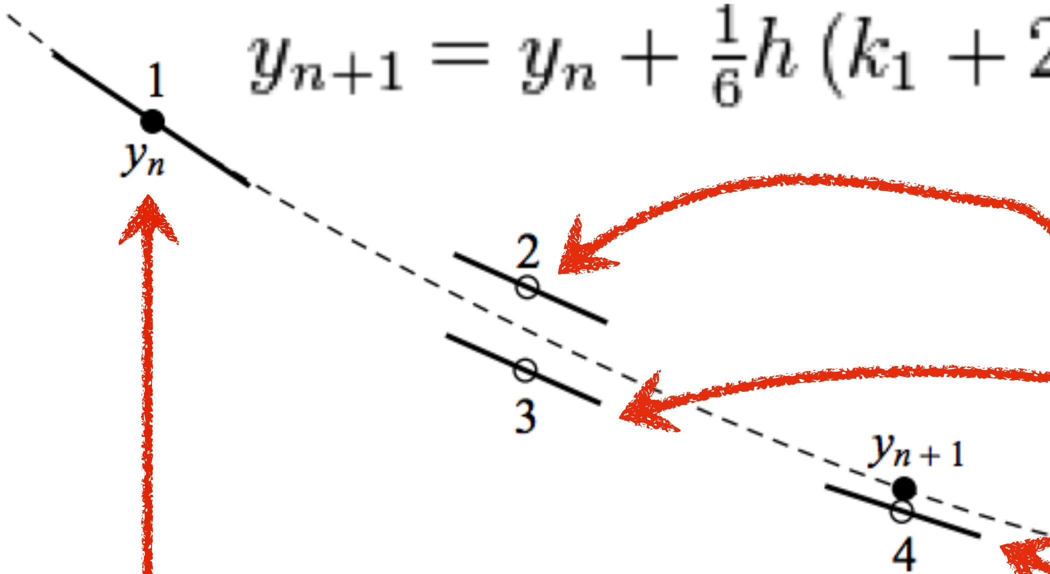
$k_1 = f(t_n, y_n)$ Tangent at beginning of interval

$k_2 = f(t_n + \tfrac{1}{2}h, y_n + \tfrac{h}{2}k_1)$ Tangent at trial midpoint $hk_1/2$

$k_3 = f(t_n + \tfrac{1}{2}h, y_n + \tfrac{h}{2}k_2)$ Tangent at trial midpoint $hk_2/2$

$k_4 = f(t_n + h, y_n + hk_3)$ Tangent at trial end of interval

$$y_{n+1} = y_n + \frac{1}{6}h\left(k_1 + 2k_2 + 2k_3 + k_4\right)$$



$k_1 = f(t_n, y_n)$ Tangent at beginning of interval

$k_2 = f(t_n + \frac{1}{2}h, y_n + \frac{h}{2}k_1)$ Tangent at trial midpoint $hk_1/2$

$k_3 = f(t_n + \frac{1}{2}h, y_n + \frac{h}{2}k_2)$ Tangent at trial midpoint $hk_2/2$

$k_4 = f(t_n + h, y_n + hk_3)$ Tangent at trial end of interval

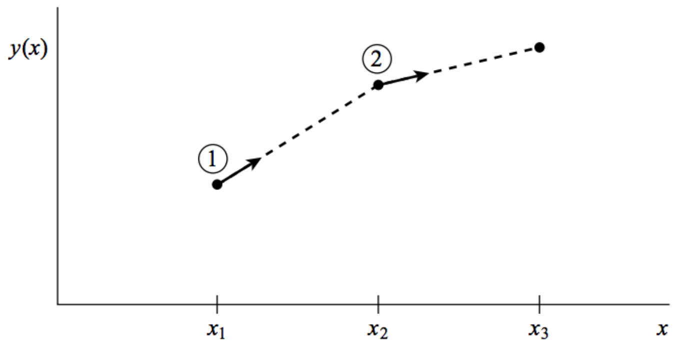


Figure 16.1.1. Euler's method. In this simplest (and least accurate) method for integrating an ODE, the derivative at the starting point of each interval is extrapolated to find the next function value. The method has first-order accuracy.

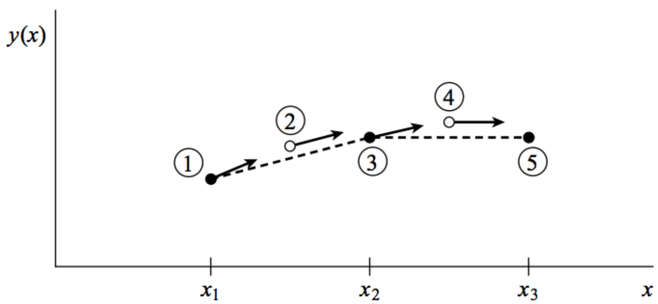
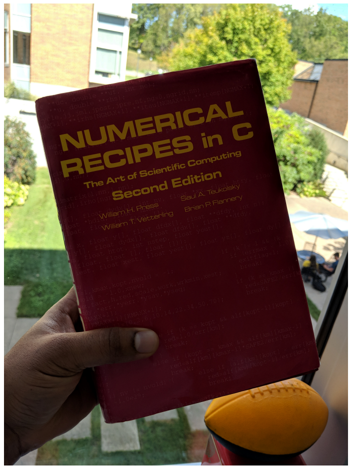


Figure 16.1.2. Midpoint method. Second-order accuracy is obtained by using the initial derivative at each step to find a point halfway across the interval, then using the midpoint derivative across the full width of the interval. In the figure, filled dots represent final function values, while open dots represent function values that are discarded once their derivatives have been calculated and used.



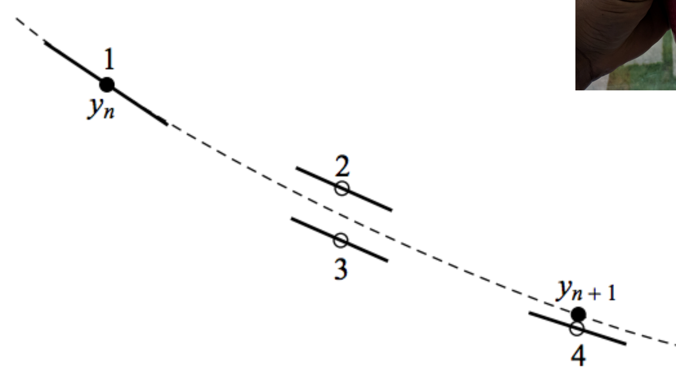


Figure 16.1.3. Fourth-order Runge-Kutta method. In each step the derivative is evaluated four times: once at the initial point, twice at trial midpoints, and once at a trial endpoint. From these derivatives the final function value (shown as a filled dot) is calculated. (See text for details.)

$k_{(x1)} = x_t + 0$

$k_{(v1)} = v_t + 0$

$c_1 = 0$

remember: 2^{nd} order ODE can represented as 1^{st} order

$$y_{n+1} = y_n + \dot{y}_n \Delta t$$

$$\dot{y}_{n+1} = \dot{y}_n + \ddot{y}_n \Delta t$$

$$k_{(x1)} = x_t + 0$$

$$k_{(v1)} = v_t + 0$$

$$k_{(x2)} = k_{(x1)} + (a_{21} * k_{(v1)} * h)$$

integrate position from t to trial midpoint at $t+c_2h$ using $k_{(v1)}$ ($c_2=1/2$)

$a_{21}=1/2$

$k_{(x1)} = x_t + 0$

$k_{(v1)} = v_t + 0$

$k_{(x2)} = k_{(x1)} + (a_{21} * k_{(v1)} * h)$

$k_{(v2)} = k_{(v1)} + (a_{21} * \text{acceleration}(k_{(x1)}) * h)$

integrate velocity from t to trial midpoint at $t+c_2h$

$a_{21}=1/2$

acceleration depends on system state (i.e., position of pendulum)

$k_{(x1)} = x_t + 0$

$k_{(v1)} = v_t + 0$

$k_{(x2)} = k_{(x1)} + (a_{21} * k_{(v1)} * h)$

$k_{(v2)} = k_{(v1)} + (a_{21} * \text{acceleration}(k_{(x1)}) * h)$

$k_{(x3)} = k_{(x1)} + (a_{32} * k_{(v2)} * h)$

$a_{32}=1/2$

integrate position from t to second trial midpoint at $t+c_3h$ using $k_{(v2)}$ $(c_3=1/2)$

$k_{(x1)} = x_t + 0$

$k_{(v1)} = v_t + 0$

$k_{(x2)} = k_{(x1)} + (a_{21} * k_{(v1)} * h)$

$k_{(v2)} = k_{(v1)} + (a_{21} * \text{acceleration}(k_{(x1)}) * h)$

$k_{(x3)} = k_{(x1)} + (a_{32} * k_{(v2)} * h)$

$k_{(v3)} = k_{(v1)} + (a_{32} * \text{acceleration}(k_{(x2)}) * h)$

acceleration depends on system state

$$k_{(x1)} = x_t + 0$$

$$k_{(v1)} = v_t + 0$$

$$k_{(x2)} = k_{(x1)} + (a_{21} * k_{(v1)} * h)$$

$$k_{(v2)} = k_{(v1)} + (a_{21} * \text{acceleration}(k_{(x1)}) * h)$$

$$k_{(x3)} = k_{(x1)} + (a_{32} * k_{(v2)} * h)$$

$$k_{(v3)} = k_{(v1)} + (a_{32} * \text{acceleration}(k_{(x2)}) * h)$$

$$k_{(x4)} = k_{(x1)} + (a_{43} * k_{(v3)} * h)$$

$a_{43}=1$

integrate position from t to trial endpoint at $t+c_4h$ using $k_{(v3)}$ ($c_4=1$)

$$k_{(x1)} = x_t + 0$$
$$k_{(v1)} = v_t + 0$$
$$k_{(x2)} = k_{(x1)} + (a_{21} * k_{(v1)} * h)$$
$$k_{(v2)} = k_{(v1)} + (a_{21} * \text{acceleration}(k_{(x1)}) * h)$$
$$k_{(x3)} = k_{(x1)} + (a_{32} * k_{(v2)} * h)$$
$$k_{(v3)} = k_{(v1)} + (a_{32} * \text{acceleration}(k_{(x2)}) * h)$$
$$k_{(x4)} = k_{(x1)} + (a_{43} * k_{(v3)} * h)$$
$$k_{(v4)} = k_{(v1)} + (a_{43} * \text{acceleration}(k_{(x3)}) * h)$$

position update is weighted sum of velocities
$b_1=1/6$, $b_2=1/3$, $b_3=1/3$, $b_4=1/6$

$k_{(x1)} = x_t + 0$

$k_{(v1)} = v_t + 0$

$k_{(x2)} = k_{(x1)} + (a_{21} * k_{(v1)} * h)$

$k_{(v2)} = k_{(v1)} + (a_{21} * \text{acceleration}(k_{(x1)}) * h)$

$k_{(x3)} = k_{(x1)} + (a_{32} * k_{(v2)} * h)$

$k_{(v3)} = k_{(v1)} + (a_{32} * \text{acceleration}(k_{(x2)}) * h)$

$k_{(x4)} = k_{(x1)} + (a_{43} * k_{(v3)} * h)$

$k_{(v4)} = k_{(v1)} + (a_{43} * \text{acceleration}(k_{(x3)}) * h)$

$$\boxed{x_{t+h} = x_t + h*(b_1*k_{(v1)}+b_2*k_{(v2)}+b_3*k_{(v3)}+b_4*k_{(v4)})}$$

velocity update is weighted sum of accelerations

$b_1=1/6,\ b_2=1/3,\ b_3=1/3,\ b_4=1/6$

$k_{(x1)} = x_t + 0$

$k_{(v1)} = v_t + 0$

$k_{(x2)} = k_{(x1)} + (a_{21} * k_{(v1)} * h)$

$k_{(v2)} = k_{(v1)} + (a_{21} * \text{acceleration}(k_{(x1)}) * h)$

$k_{(x3)} = k_{(x1)} + (a_{32} * k_{(v2)} * h)$

$k_{(v3)} = k_{(v1)} + (a_{32} * \text{acceleration}(k_{(x2)}) * h)$

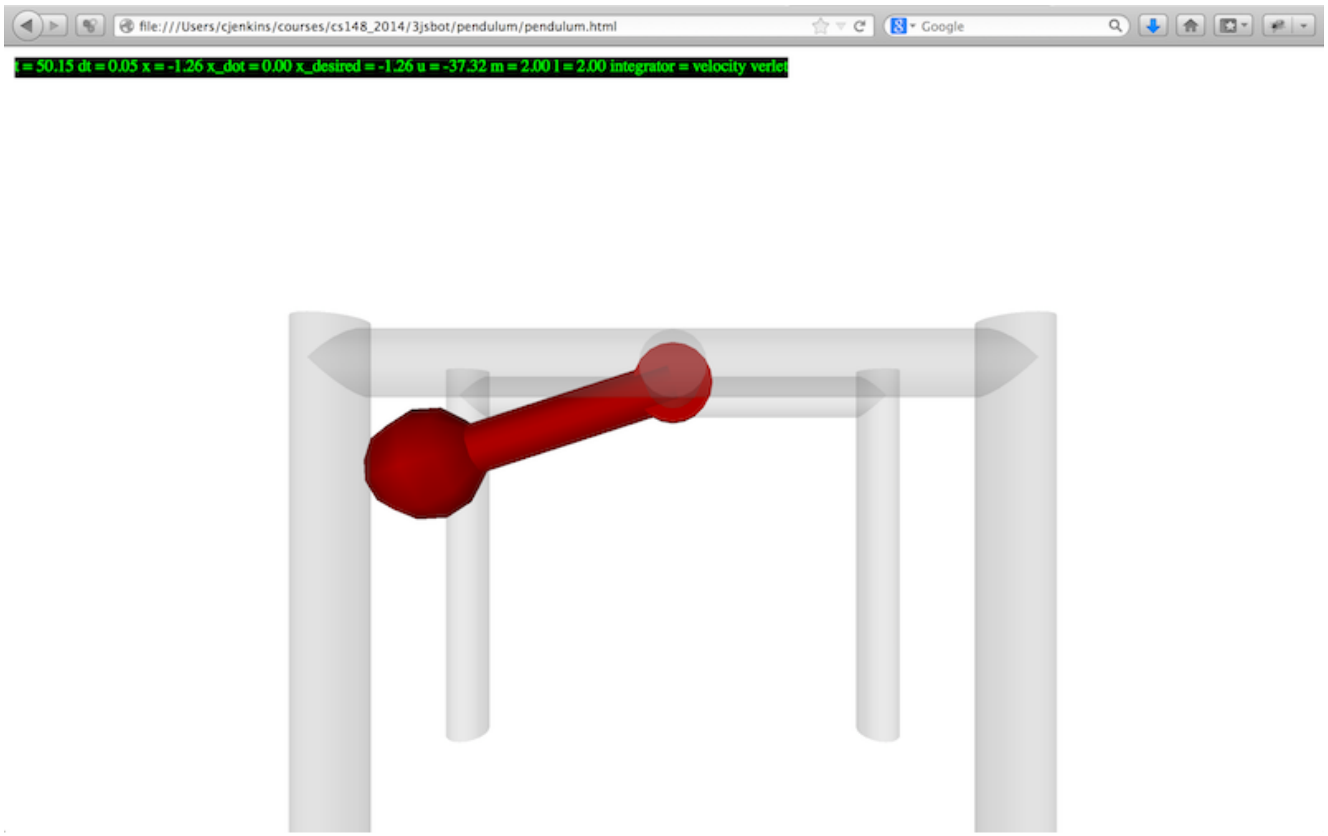
$k_{(x4)} = k_{(x1)} + (a_{43} * k_{(v3)} * h)$

$k_{(v4)} = k_{(v1)} + (a_{43} * \text{acceleration}(k_{(x3)}) * h)$

$x_{t+h} = x_t + h*(b_1*k_{(v1)}+b_2*k_{(v2)}+b_3*k_{(v3)}+b_4*k_{(v4)})$

$v_{t+h} = v_t + h*[b_1*a(k_{(x1)})+b_2*a(k_{(x2)})+b_3*a(k_{(x3)})+b_4*a(k_{(x4)})]$

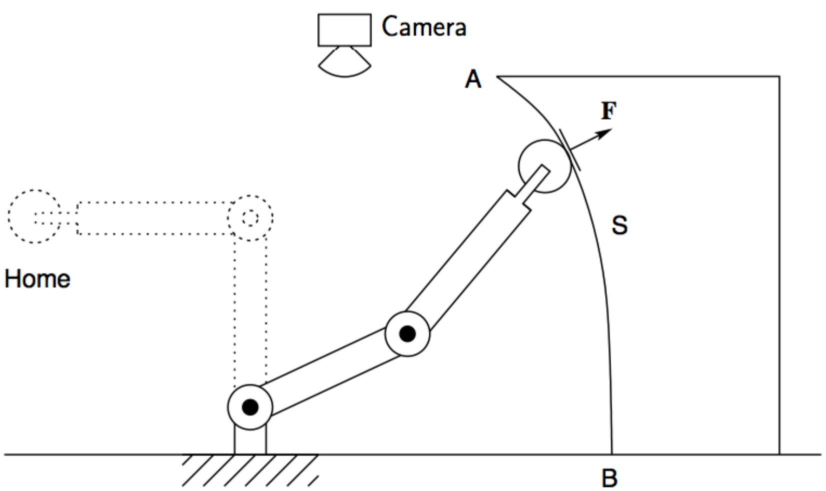
Let’s see what happens

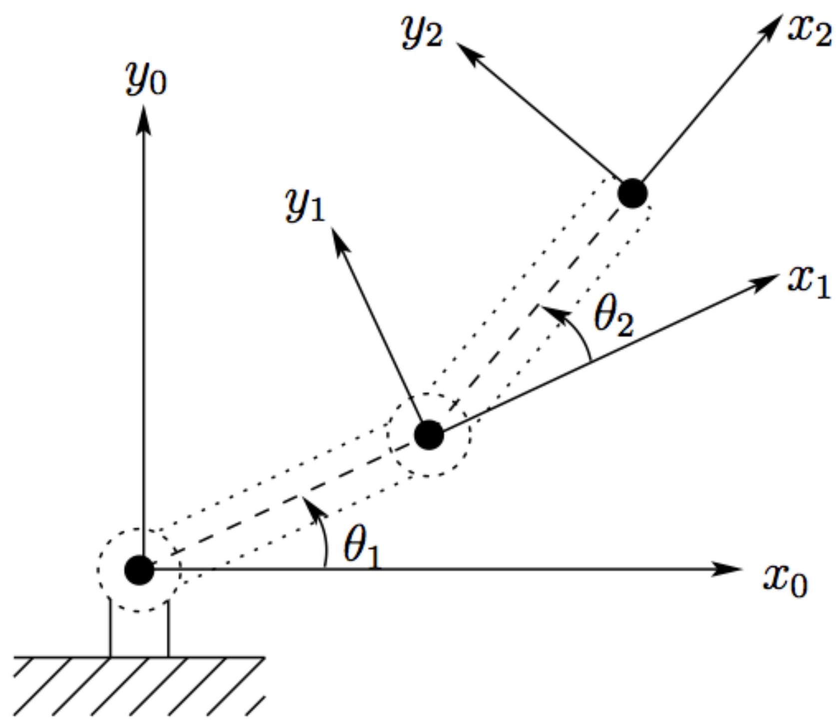


Let’s revisit the

Planar 2-DOF 2-link Arm

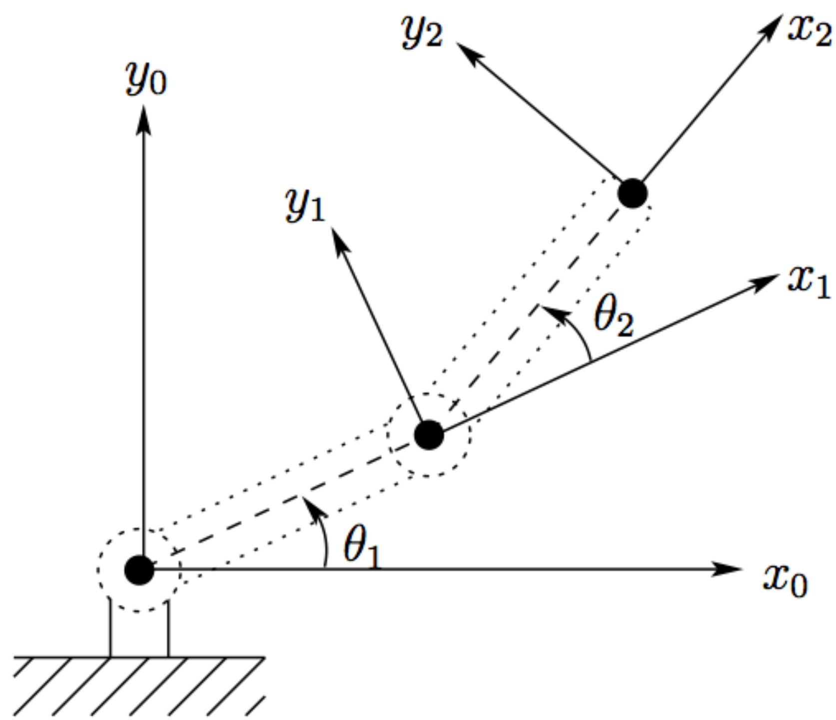
Planar 2-DOF 2-link Arm





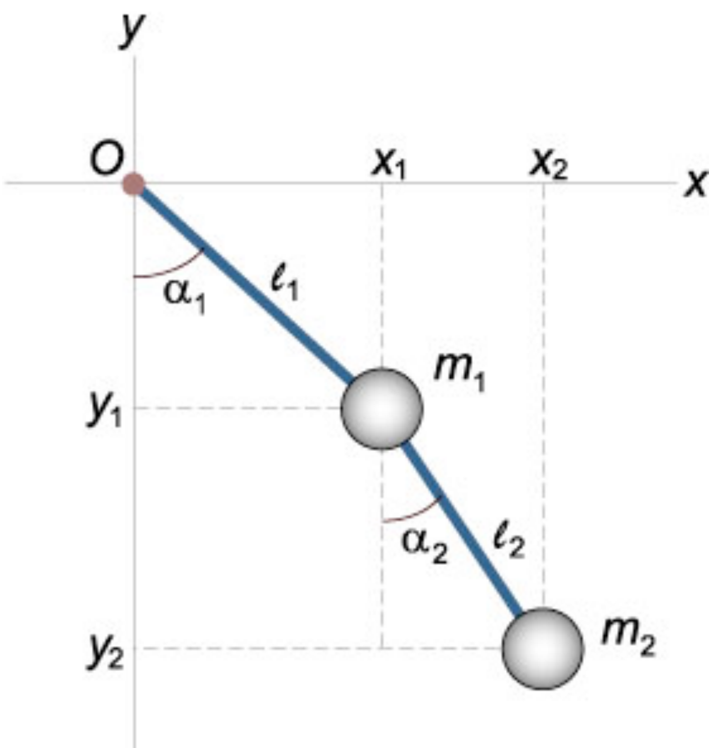
Planar 2-DOF 2-link Arm

Beyster 3753
Wed Sep 12 2018




For Coffee Supplies
Please See Yolonda in
Room 3820

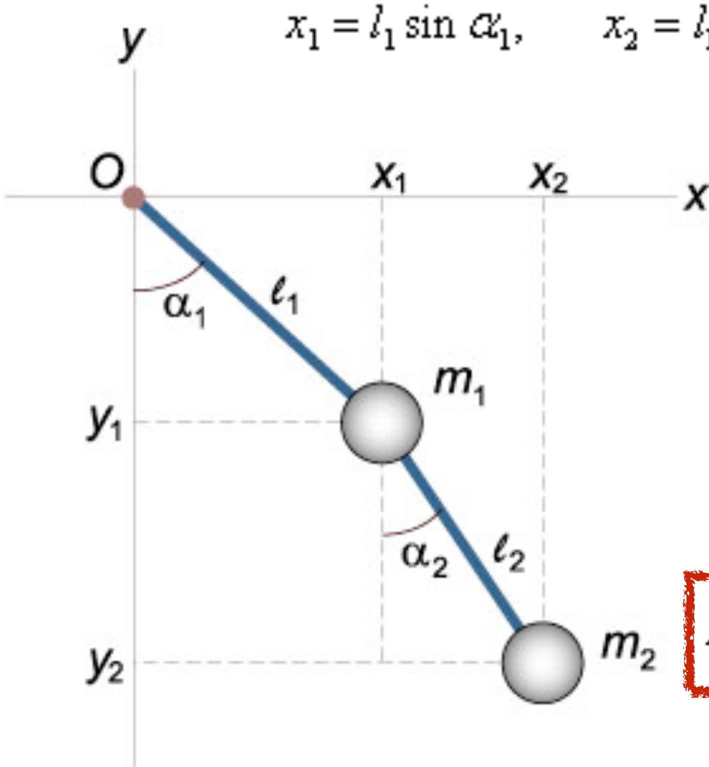
Can we add another link?



- Double pendulum

Locations of pendulum bobs

$$x_1 = l_1 \sin \alpha_1, \qquad x_2 = l_1 \sin \alpha_1 + l_2 \sin \alpha_2, \qquad y_1 = -l_1 \cos \alpha_1, \qquad y_2 = -l_1 \cos \alpha_1 - l_2 \cos \alpha_2.$$



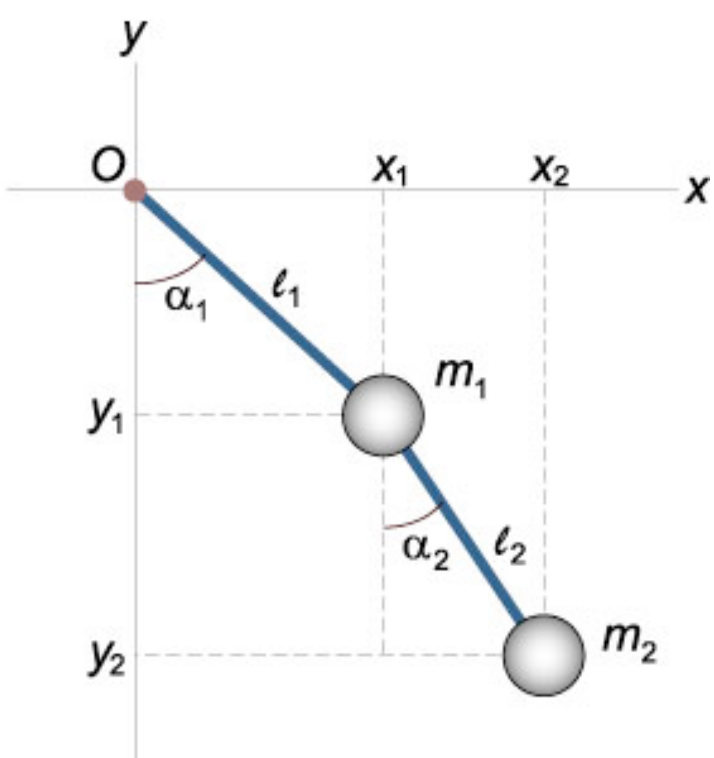
Lagrangian of pendulum bob positions

$$T = \frac{m_1 v_1^2}{2} + \frac{m_2 v_2^2}{2} = \frac{m_1\left(\dot{x}_1^2 + \dot{y}_1^2\right)}{2} + \frac{m_2\left(\dot{x}_2^2 + \dot{y}_2^2\right)}{2}, \qquad V = m_1 g y_1 + m_2 g y_2.$$

$$\boxed{L = T - V = T_1 + T_2 - (V_1 + V_2) = \frac{m_1}{2}\left(\dot{x}_1^2 + \dot{y}_1^2\right) + \frac{m_2}{2}\left(\dot{x}_2^2 + \dot{y}_2^2\right) - m_1 g y_1 - m_2 g y_2.}$$

http://www.math24.net/double-pendulum.html

Lagrangian in generalized coordinates (joint angles)



http://www.math24.net/double-pendulum.html

$$\dot{x}_1 = l_1 \cos\alpha_1 \cdot \dot{\alpha}_1, \qquad \dot{x}_2 = l_1 \cos\alpha_1 \cdot \dot{\alpha}_1 + l_2 \cos\alpha_2 \cdot \dot{\alpha}_2,$$

$$\dot{y}_1 = l_1 \sin\alpha_1 \cdot \dot{\alpha}_1, \qquad \dot{y}_2 = l_1 \sin\alpha_1 \cdot \dot{\alpha}_1 + l_2 \sin\alpha_2 \cdot \dot{\alpha}_2.$$

$$T_1 = \frac{m_1}{2}\left(\dot{x}_1^2 + \dot{y}_1^2\right) = \frac{m_1}{2}\left(l_1^2\dot{\alpha}_1^2\cos^2\alpha_1 + l_1^2\dot{\alpha}_1^2\sin^2\alpha_1\right) = \frac{m_1}{2}l_1^2\dot{\alpha}_1^2,$$

$$T_2 = \frac{m_2}{2}\left(\dot{x}_2^2 + \dot{y}_2^2\right) = \frac{m_2}{2}\left[\left(l_1\dot{\alpha}_1\cos\alpha_1 + l_2\dot{\alpha}_2\cos\alpha_2\right)^2 + \left(l_1\dot{\alpha}_1\sin\alpha_1 + l_2\dot{\alpha}_2\sin\alpha_2\right)^2\right]$$

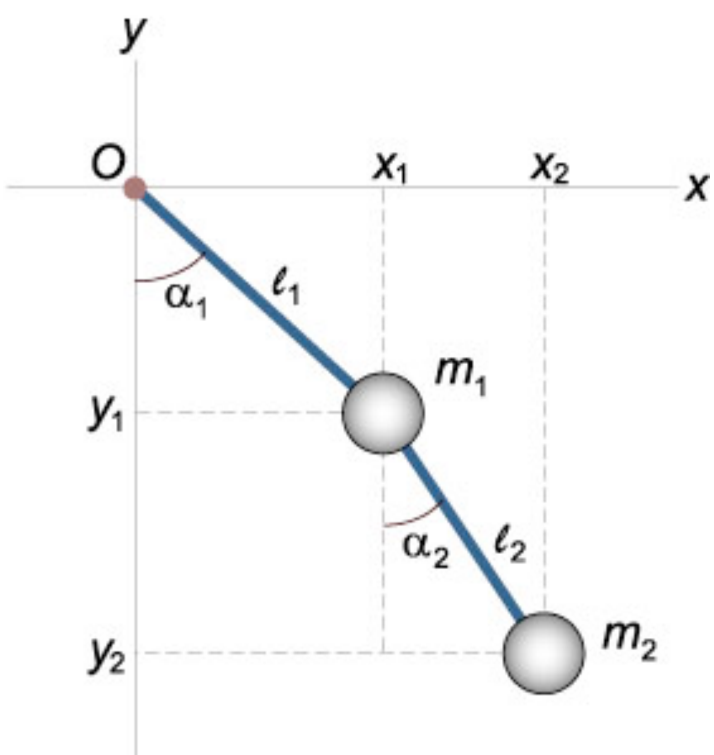
$$= \frac{m_2}{2}\left[l_1^2\dot{\alpha}_1^2\cos^2\alpha_1 + l_2^2\dot{\alpha}_2^2\cos^2\alpha_2 + 2l_1l_2\dot{\alpha}_1\dot{\alpha}_2\cos\alpha_1\cos\alpha_2 + l_1^2\dot{\alpha}_1^2\sin^2\alpha_1 + l_2^2\dot{\alpha}_2^2\sin^2\alpha_2 + 2l_1l_2\dot{\alpha}_1\dot{\alpha}_2\sin\alpha_1\sin\alpha_2\right]$$

$$= \frac{m_2}{2}\left[l_1^2\dot{\alpha}_1^2 + l_2^2\dot{\alpha}_2^2 + 2l_1l_2\dot{\alpha}_1\dot{\alpha}_2\cos\left(\alpha_1 - \alpha_2\right)\right],$$

$$V_1 = m_1 g y_1 = -m_1 g l_1 \cos\alpha_1,$$

$$V_2 = m_2 g y_2 = -m_2 g\left(l_1\cos\alpha_1 + l_2\cos\alpha_2\right).$$

$$\boxed{\begin{aligned} L &= T - V = T_1 + T_2 - \left(V_1 + V_2\right) = \\ &= \left(\frac{m_1}{2} + \frac{m_2}{2}\right)l_1^2\dot{\alpha}_1^2 + \frac{m_2}{2}l_2^2\dot{\alpha}_2^2 + m_2l_1l_2\dot{\alpha}_1\dot{\alpha}_2\cos\left(\alpha_1 - \alpha_2\right) + \left(m_1 + m_2\right)gl_1\cos\alpha_1 + m_2gl_2\cos\alpha_2. \end{aligned}}$$



http://www.math24.net/double-pendulum.html

Lagrangian equations of motion

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\alpha}_i} - \frac{\partial L}{\partial \alpha_i} = 0, \qquad i = 1,2.$$

$$\frac{\partial L}{\partial \dot{\alpha}_1} = (m_1+m_2) l_1^2 \dot{\alpha}_1 + m_2 l_1 l_2 \dot{\alpha}_2 \cos(\alpha_1 - \alpha_2), \qquad \frac{\partial L}{\partial \alpha_1} = -m_2 l_1 l_2 \dot{\alpha}_1 \dot{\alpha}_2 \sin(\alpha_1 - \alpha_2) - (m_1+m_2) g l_1 \sin\alpha_1,$$

$$\frac{\partial L}{\partial \dot{\alpha}_2} = m_2 l_2^2 \dot{\alpha}_2 + m_2 l_1 l_2 \dot{\alpha}_1 \cos(\alpha_1 - \alpha_2), \qquad \frac{\partial L}{\partial \alpha_2} = m_2 l_1 l_2 \dot{\alpha}_1 \dot{\alpha}_2 \sin(\alpha_1 - \alpha_2) - m_2 g l_2 \sin\alpha_2.$$

Lagrangian EOM for first DOF (*i*=*1*)

$$\frac{d}{dt}\left[(m_1+m_2) l_1^2 \dot{\alpha}_1 + m_2 l_1 l_2 \dot{\alpha}_2 \cos(\alpha_1 - \alpha_2)\right] + m_2 l_1 l_2 \dot{\alpha}_1 \dot{\alpha}_2 \sin(\alpha_1 - \alpha_2) + (m_1+m_2) g l_1 \sin\alpha_1 = 0,$$

$$\Rightarrow (m_1+m_2) l_1^2 \ddot{\alpha}_1 + m_2 l_1 l_2 \left[\ddot{\alpha}_2 \cos(\alpha_1 - \alpha_2) - \dot{\alpha}_2 \sin(\alpha_1 - \alpha_2) \cdot (\dot{\alpha}_1 - \dot{\alpha}_2)\right]$$

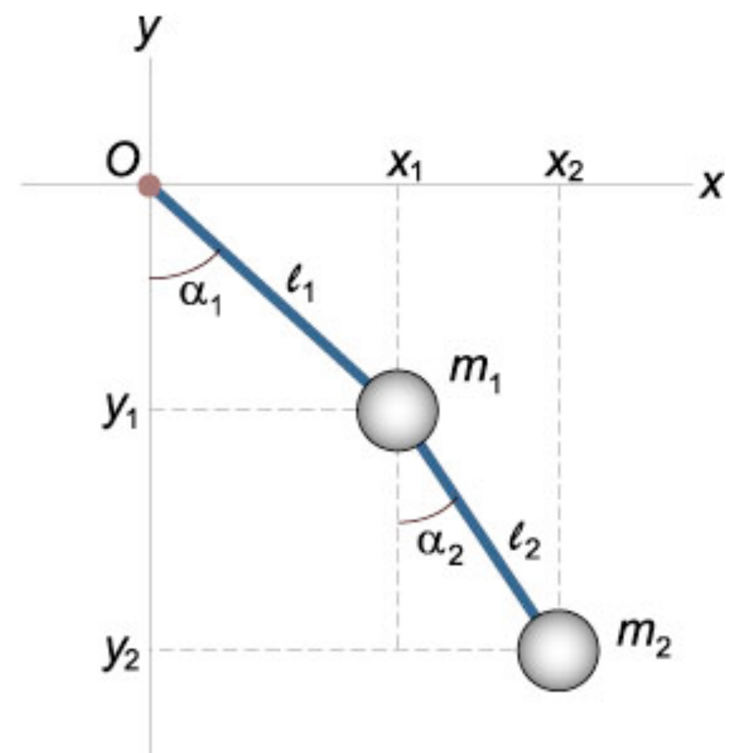
$$+ m_2 l_1 l_2 \dot{\alpha}_1 \dot{\alpha}_2 \sin(\alpha_1 - \alpha_2) + (m_1+m_2) g l_1 \sin\alpha_1 = 0,$$

$$\Rightarrow (m_1+m_2) l_1^2 \ddot{\alpha}_1 + m_2 l_1 l_2 \ddot{\alpha}_2 \cos(\alpha_1 - \alpha_2) - \cancel{m_2 l_1 l_2 \dot{\alpha}_1 \dot{\alpha}_2 \sin(\alpha_1 - \alpha_2)} + m_2 l_1 l_2 \dot{\alpha}_2^2 \sin(\alpha_1 - \alpha_2)$$

$$+ \cancel{m_2 l_1 l_2 \dot{\alpha}_1 \dot{\alpha}_2 \sin(\alpha_1 - \alpha_2)} + (m_1+m_2) g l_1 \sin\alpha_1 = 0,$$

$$\Rightarrow (m_1+m_2) l_1^2 \ddot{\alpha}_1 + m_2 l_1 l_2 \ddot{\alpha}_2 \cos(\alpha_1 - \alpha_2) + m_2 l_1 l_2 \dot{\alpha}_2^2 \sin(\alpha_1 - \alpha_2) + (m_1+m_2) g l_1 \sin\alpha_1 = 0.$$

$$\boxed{(m_1+m_2) l_1 \ddot{\alpha}_1 + m_2 l_2 \ddot{\alpha}_2 \cos(\alpha_1 - \alpha_2) + m_2 l_2 \dot{\alpha}_2^2 \sin(\alpha_1 - \alpha_2) + (m_1+m_2) g \sin\alpha_1 = 0.}$$



http://www.math24.net/double-pendulum.html

Lagrangian equations of motion

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\alpha}_i} - \frac{\partial L}{\partial \alpha_i} = 0, \qquad i = 1,2.$$

$$\frac{\partial L}{\partial \dot{\alpha}_1} = (m_1+m_2) l_1^2 \dot{\alpha}_1 + m_2 l_1 l_2 \dot{\alpha}_2 \cos(\alpha_1-\alpha_2), \qquad \frac{\partial L}{\partial \alpha_1} = -m_2 l_1 l_2 \dot{\alpha}_1 \dot{\alpha}_2 \sin(\alpha_1-\alpha_2) - (m_1+m_2) g l_1 \sin\alpha_1,$$

$$\frac{\partial L}{\partial \dot{\alpha}_2} = m_2 l_2^2 \dot{\alpha}_2 + m_2 l_1 l_2 \dot{\alpha}_1 \cos(\alpha_1-\alpha_2), \qquad \frac{\partial L}{\partial \alpha_2} = m_2 l_1 l_2 \dot{\alpha}_1 \dot{\alpha}_2 \sin(\alpha_1-\alpha_2) - m_2 g l_2 \sin\alpha_2.$$

Lagrangian EOM for second DOF (*i*=2)

$$\frac{d}{dt}\left[m_2 l_2^2 \dot{\alpha}_2 + m_2 l_1 l_2 \dot{\alpha}_1 \cos(\alpha_1-\alpha_2)\right] - m_2 l_1 l_2 \dot{\alpha}_1 \dot{\alpha}_2 \sin(\alpha_1-\alpha_2) + m_2 g l_2 \sin\alpha_2 = 0,$$

$$\Rightarrow m_2 l_2^2 \ddot{\alpha}_2 + m_2 l_1 l_2 \ddot{\alpha}_1 \cos(\alpha_1-\alpha_2) - m_2 l_1 l_2 \dot{\alpha}_1 \sin(\alpha_1-\alpha_2)\cdot(\dot{\alpha}_1-\dot{\alpha}_2) - m_2 l_1 l_2 \dot{\alpha}_1 \dot{\alpha}_2 \sin(\alpha_1-\alpha_2)$$

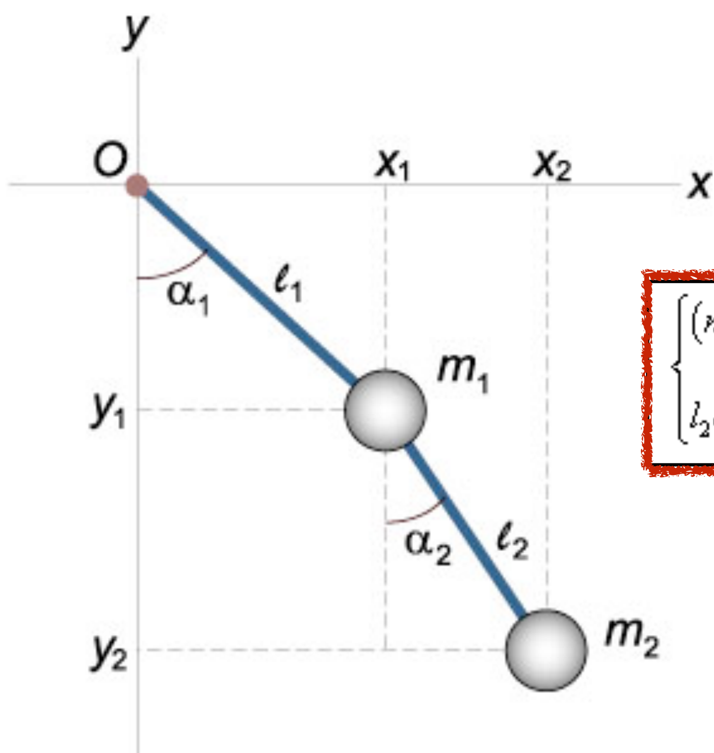
$$+ m_2 g l_2 \sin\alpha_2 = 0,$$

$$\Rightarrow m_2 l_2^2 \ddot{\alpha}_2 + m_2 l_1 l_2 \ddot{\alpha}_1 \cos(\alpha_1-\alpha_2) - m_2 l_1 l_2 \dot{\alpha}_1^2 \sin(\alpha_1-\alpha_2) + \cancel{m_2 l_1 l_2 \dot{\alpha}_1 \dot{\alpha}_2 \sin(\alpha_1-\alpha_2)}$$

$$- \cancel{m_2 l_1 l_2 \dot{\alpha}_1 \dot{\alpha}_2 \sin(\alpha_1-\alpha_2)} + m_2 g l_2 \sin\alpha_2 = 0,$$

$$\Rightarrow m_2 l_2^2 \ddot{\alpha}_2 + m_2 l_1 l_2 \ddot{\alpha}_1 \cos(\alpha_1-\alpha_2) - m_2 l_1 l_2 \dot{\alpha}_1^2 \sin(\alpha_1-\alpha_2) + m_2 g l_2 \sin\alpha_2 = 0.$$

$$\boxed{l_2 \ddot{\alpha}_2 + l_1 \ddot{\alpha}_1 \cos(\alpha_1-\alpha_2) - l_1 \dot{\alpha}_1^2 \sin(\alpha_1-\alpha_2) + g \sin\alpha_2 = 0.}$$



http://www.math24.net/double-pendulum.html

The 2 equations of motion for double pendulum

$$\begin{cases} (m_1+m_2)l_1\ddot{\alpha}_1+m_2l_2\ddot{\alpha}_2\cos(\alpha_1-\alpha_2)+m_2l_2\dot{\alpha}_2^2\sin(\alpha_1-\alpha_2)+(m_1+m_2)g\sin\alpha_1=0 \\ l_2\ddot{\alpha}_2+l_1\ddot{\alpha}_1\cos(\alpha_1-\alpha_2)-l_1\dot{\alpha}_1^2\sin(\alpha_1-\alpha_2)+g\sin\alpha_2=0 \end{cases}$$

Substitute equations into each other to solve for $\ddot{\alpha}_1$ and $\ddot{\alpha}_2$

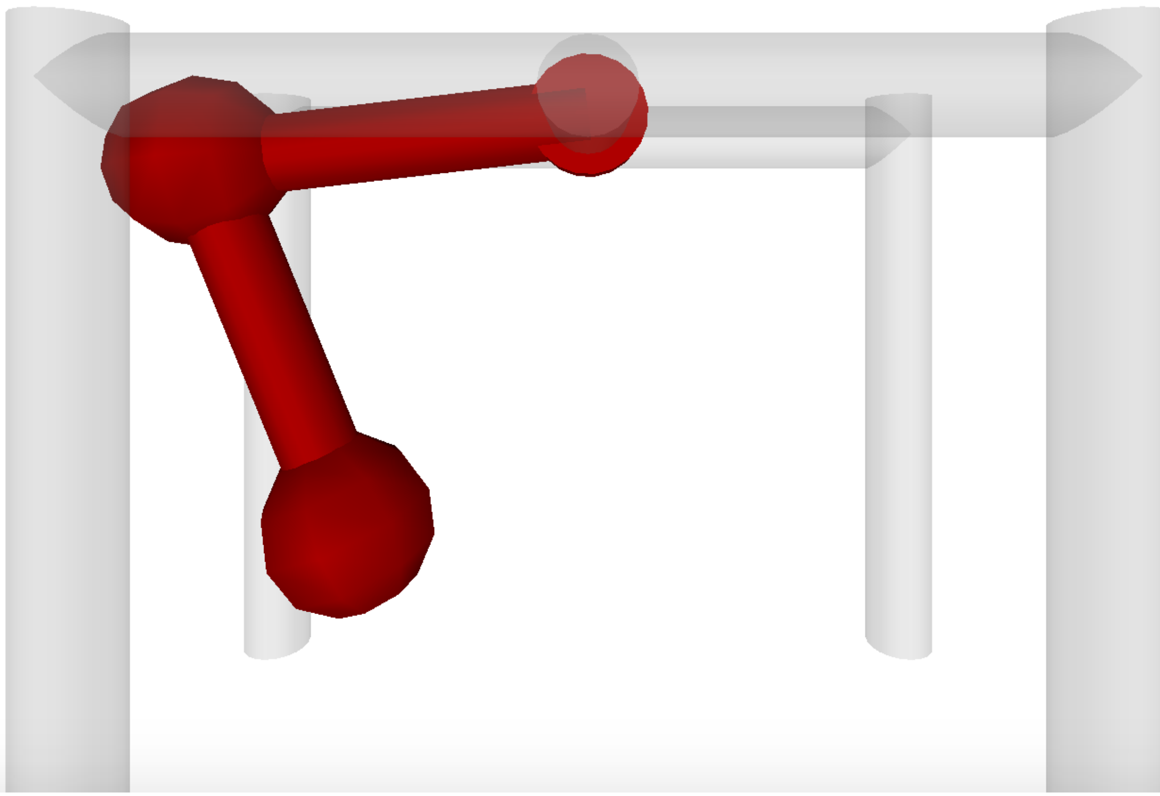
What happened to the torques?

Pendularm2 by oharib

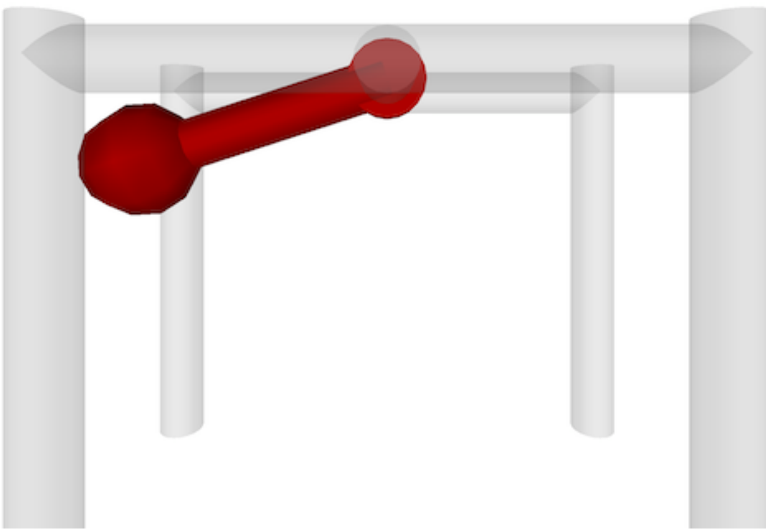
System
t = 6119.20 dt = 0.05
integrator = runge-kutta
x1 = -1.46
x1_dot = -0.00
x2 = 1.84
x2_dot = -0.00
x1_desired = -1.46
x2_desired = 1.84

Pendulum
mass = 2.00
length = 2.00
gravity = 9.81

Keys
a/d - apply user force
1/2 - adjust desired angle 1
3/4 - adjust desired angle 2
c|x - toggle servo
s - disable servo



Project 2: Pendularm



https://raw.githubusercontent.com/autorob/kineval-stencil/master/project_pendularm/pendularm1.html

```
function init() {...
   pendulum = { // pendulum object
        length:2.0,
        mass:2.0,
        angle:Math.PI/2,
        angle_dot:0.0};

    gravity = 9.81;  // Earth gravity
    t = 0; dt = 0.05; // init time
    pendulum.control = 0; // motor

    // next lecture: PID control
    pendulum.desired = -Math.PI/2.5;
    pendulum.desired_dot = 0;
    pendulum.servo =
        {kp:0, kd:0, ki:0};
...}
```

```
function animate() { ...
if (numerical_integrator === "euler") {
    // STENCIL: Euler integrator }
    else if (numerical_integrator === "verlet") {
    // STENCIL: basic Verlet integration }
    else if (numerical_integrator === "velocity verlet") {
    // STENCIL: velocity Verlet }
    else if (numerical_integrator === "runge-kutta") {
    // STENCIL: Runge-Kutta 4 integrator }
    else {   }
    // set the angle of the pendulum
    pendulum.geom.rotation.y = pendulum.angle;
    t = t + dt;  // advance time
...}

function pendulum_acceleration(p,g) {
    // STENCIL: return acceleration from equations of motion }
```

Next lecture:
Motion Control
and PID

https://youtu.be/1H8t_F638Mo