WP@ELAB training, the calculus day

Vojtech Svoboda, Pavel Kuriscak, Frantisek Lustig

November 16, 2021

Table of Contents

First of all ...

Companion website

http://buon.fjfi.cvut.cz/wp

- This presentation (in latex) .. to be reused/adapted for education.
- All used examples (ready to be used for education).
- Other relevant info.
- Resources.

Table of Contents

Motivation

Scientific problem

Theory, **Numerical simulation**, Experiment

Figure: Soberenia Pendulum

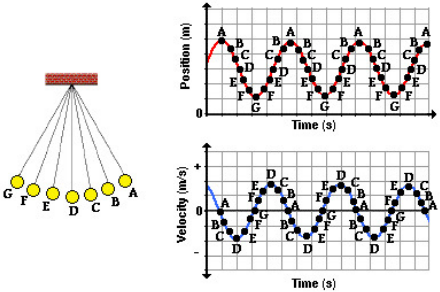
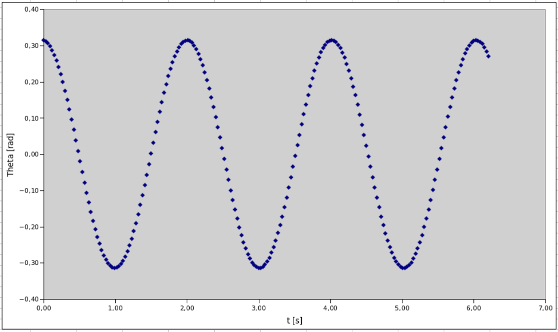


Figure: Pendulum analysis @ [Hen20]

Sreenshot: Pendulum basic @ spreadsheet

	A	B	C	D	E
1	t	F	eps	omega	theta
2	[s]	[N]	rad/s*2	rad/s	rad
3	0,00			0,00	0,31
4	0,03	−3,03	−3,03	−0,09	0,31
5	0,06	−3,01	−3,01	−0,18	0,31
6	0,09	−2,96	−2,96	−0,27	0,30
7	0,12	−2,88	−2,88	−0,36	0,29
8	0,15	−2,78	−2,78	−0,44	0,27
9	0,18	−2,66	−2,66	−0,52	0,26
10	0,21	−2,51	−2,51	−0,59	0,24
11	0,24	−2,34	−2,34	−0,66	0,22
12	0,27	−2,15	−2,15	−0,73	0,20
13	0,30	−1,94	−1,94	−0,79	0,18
14	0,33	−1,71	−1,71	−0,84	0,15
15	0,36	−1,47	−1,47	−0,88	0,12
16	0,39	−1,21	−1,21	−0,92	0,10
17	0,42	−0,94	−0,94	−0,95	0,07
18	0,45	−0,66	−0,66	−0,97	0,04
19	0,48	−0,38	−0,38	−0,98	0,01
20	0,51	−0,09	−0,09	−0,98	−0,02
21	0,54	0,20	0,20	−0,98	−0,05
22	0,57	0,49	0,49	−0,96	−0,08
23	0,60	0,77	0,77	−0,94	−0,11
24	0,63	1,04	1,04	−0,91	−0,13
25	0,66	1,31	1,31	−0,87	−0,16
26	0,69	1,56	1,56	−0,82	−0,18
27	0,72	1,80	1,80	−0,77	−0,21
28	0,75	2,02	2,02	−0,71	−0,23
29	0,78	2,22	2,22	−0,64	−0,25
30	0,81	2,40	2,40	−0,57	−0,26
31	0,84	2,57	2,57	−0,49	−0,28
32	0,87	2,71	2,71	−0,41	−0,29
33	0,90	2,82	2,82	−0,32	−0,30
34	0,93	2,91	2,91	−0,24	−0,31
35	0,96	2,98	2,98	−0,15	−0,31
36	0,99	3,02	3,02	−0,06	−0,31
37	1,02	3,04	3,04	0,03	−0,31
38	1,05	3,03	3,03	0,13	−0,31
39	1,08	2,99	2,99	0,21	−0,30

	I	J	K	L	M	N
1	m	1,00	kg	T	2,01	s
2	g	9,81	m/s**2			
3	dt	0,03	s			
4	l	1,00	m			



See example

Sreenshot: Pendulum basic @ processing

https://editor.p5js.org/vojtech.svob/sketches/vTEaAkgs

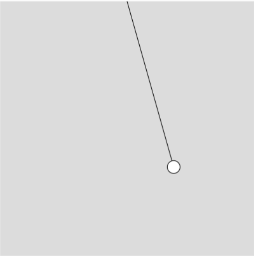
M C N Ggs TV@J Trelo Aktual KnowH GM Dg GW #0 GMrm Osobni Duše Galleries Viol YT BckgM Spánek WP

p5* File Edit Sketch Help

Auto-refresh Pendulum - basic version by vojtech.svob

sketch.js Saved: 3 minutes ago Preview

```
function setup() {
  createCanvas(400, 400);
  m=2
  l=2.705
  g=9.814
  dt=0.02
  t=0
  theta = 3.14/10; // Pendulum initial angle theta
  omega = 0; // Initial angular velocity
  C = 2; // Center point
}

function draw() {
  background(220);
  // physics
  t = t + dt;
  F=-m*g*sin(theta)
  epsilon = (F/m)/l; //angular acceleration
  omega = omega + epsilon * dt;
  theta = theta + omega * dt;
  xp = C - l * sin(theta); // X coordinate of pendulum ball
  yp = l * cos(theta); // Y coordinate of pendulum ball
  //draw it
  ppm=100 //scale it to the canvas (from meters to pixels)
  line(C*ppm, 0, xp*ppm, yp*ppm);
  ellipse(xp*ppm, yp*ppm, 20, 20);
}
```



▸ See example

Objectives

(World) Pendulum ... as a gate to physics

Numerical simulations point of view

- A comprehensive, as simple as possible numerical approach to the Pendulum problem using Euler scheme for solving ordinary differential equations (ODE) developed under various Computer Algebraic Systems:
  - spreadsheet (Excel, LibreOffice Calc, Google, gnumeric),
  - p5* processing,
  - jupyter notebook (python),
  - octave (matlab).
- Wide range of simple examples (ready to be used for education)
- Way to avoid the complex math problems (ODE) in the (early) physics education.

Outline of the talk

1. *Introduction*
   - Motivation
   - Vojtech Svoboda @ CTU
   - Euler method
2. *1D problem in cartesian coordinates: free fall*
   - Spreadsheet
   - Processing
   - Python
3. *1D problem in rotational system: pendulum*
   - Basic analysis (spreadsheet & processing & octave)
   - Pendulum with friction (spreadsheet & processing)
   - Pendulum - phase space (spreadsheet)
   - Pendulum - energy conservation (spreadsheet)
   - Pendulum - small angle approximation analysis (spreadsheet)
   - Two pendulums (processing)
4. *Numerical simulation versus experiment*
   - Prague
   - World pendulum
5. *Final remarks*
   - 2D problem in cartesian coordinates: horizontal launch
   - Runge Kutta
   - ODE solving with standard functions

Table of Contents

Faculty of Nuclear Sciences and Physical Engineering Czech Technical University in Prague

FNSPE main building in Prague

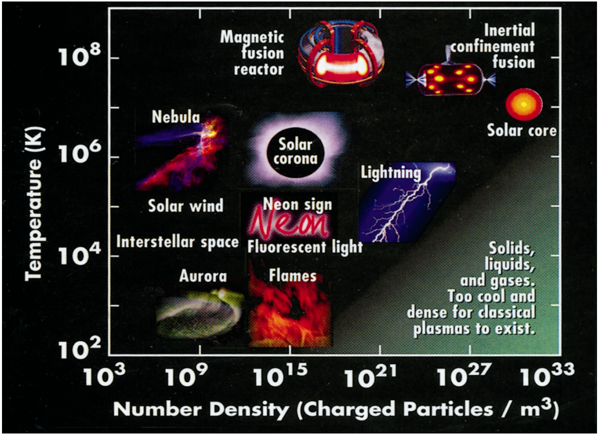
FNSPE insignia

CTU ceremony hall

- CTU founded in 1707 by the emperor Joseph I.
- CTU approximately 2200 staff members, 16000 undergraduate students, 9000 graduate and PhD students. ($\approx$ 2500 foreign students).
- FNSPE established in 1955 with the mission to train new experts for the emerging Czechoslovak nuclear programme.
- FNSPE currently a centre of education and research specialised in boundary fields between modern science and their applications in technologies, medicine, economy, biology, ecology, and other fields.

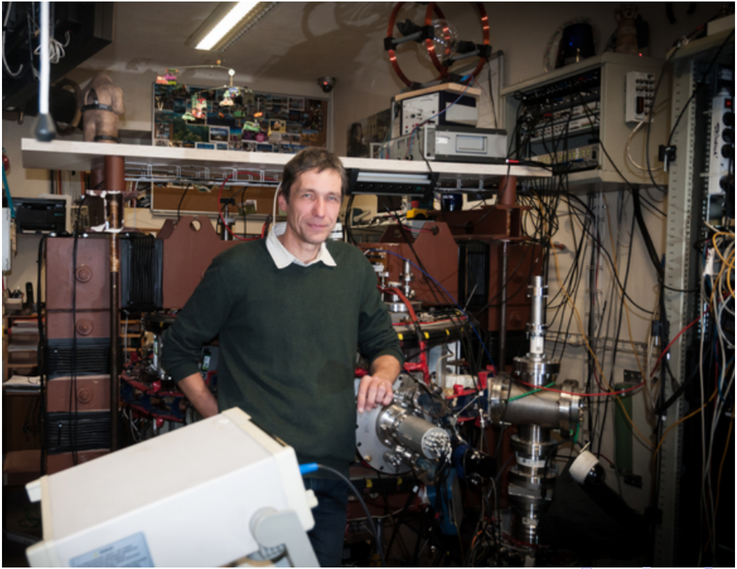
Scientific group/ education specialization

The Physics of Plasma and Thermonuclear fusion

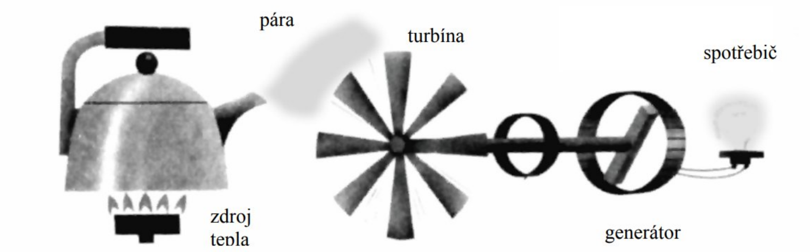


99.999 % Universe is in the Plasma state of matter

Tokamak GOLEM & Vojtěch Svoboda



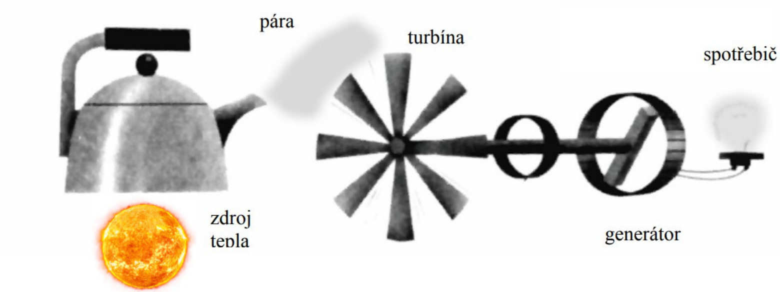
Thermal power plant - basic principle



The question:

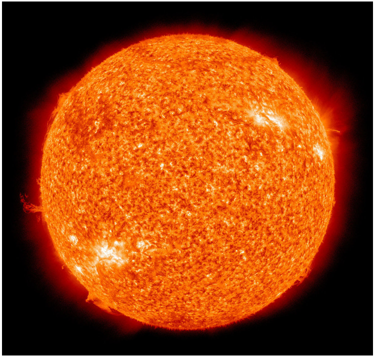
?? WHAT TO BURN ??

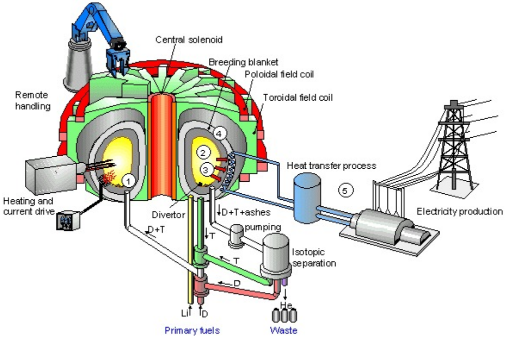
Small μSun in the terestriall conditions ??



The challenge

Tokamak mission: to create μSun in the terrestrial conditions





$$^2_1\mathbf{D}_{(10\mathrm{keV})} + ^3_1\mathbf{T}_{(10\mathrm{keV})} \Rightarrow ^4_2\mathbf{He}_{(3.5\mathrm{MeV})} + \mathbf{n}_{(14.1\mathrm{MeV})}$$

The task: to heat (up to 100 million degrees) DT fuel and confine it (up to 30 years) in the high temperature plasma state of matter to produce He & fusion energy.

Table of Contents

Initial value problem

Let's have a general force field $F(t, x, v)$ applying on an object of a mass m, having some initial conditions t_0, v_0, x_0:

- Differential solution: having dt time progress: $a = F/m$, then $v(t) = \int_{t_0}^{t} adt$, and $x(t) = \int_{t_0}^{t} vdt$
- Discrete solution: having Δt time progress, in principal, we are looking for a time series of object position $(t_0, x_0), (t_1, x_1), ..(t_n, x_n)$: $a_i = F_i/m$, then $v_{i+1} = v_i + a \cdot \Delta t$, and $x_{i+1} = x_i + v_i \cdot \Delta t$

Discrete solution - towards algorithmization

Recurring principle/algorithm

ideal for computer algebraic systems

Having Δt time progress, in principal, we are looking for a time series of object position $(t_0, x_0), (t_1, x_1), ..(t_n, x_n)$: $a_i = F_i/m$, then $v_{i+1} = v_i + a \cdot \Delta t$, and $x_{i+1} = x_i + v_i \cdot \Delta t$

time	$F(t, x, v)$	$a(t)$	$v(t)$ calculation	$x(t)$ calculation
t_0	$F_0 = F(t_0, x_0, v_0)$	$a_0 = F_0/m$	v_0 (initial cond.)	x_0 (initial cond.)
$t_1 = t_0 + \Delta t$	$F_1 = F(t_1, x_1, v_1)$	$a_1 = F_1/m$	$v_1 = v_0 + a_1\Delta t$	$x_1 = x_0 + v_1\Delta t$
$t_2 = t_1 + \Delta t$	$F_2 = F(t_2, x_2, v_2)$	$a_2 = F_2/m$	$v_2 = v_1 + a_2\Delta t$	$x_2 = x_1 + v_2\Delta t$
..	..	..	..	..
$t_n = t_{n-1} + \Delta t$	$F_n = F(t_n, x_n, v_n)$	$a_n = F_n/m$	$v_n = v_{n-1} + a_n\Delta t$	$x_n = x_{n-1} + v_n\Delta t$

Euler method solving ODE - the principle

Let an initial value problem be specified:

$$\dot{y} = f(t, y), \quad y(t_0) = y_0$$

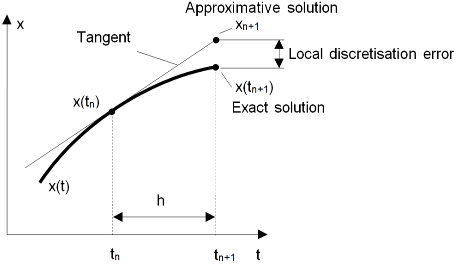


Figure: credit:[Sza14]

$$y_{n+1} = y_n + h\ f(t_n, y_n),$$
$$t_{n+1} = t_n + h$$

Euler method solving ODE - repetition (loop)

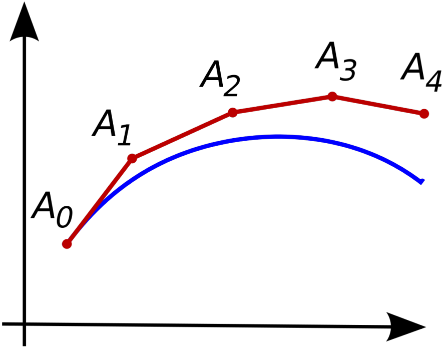


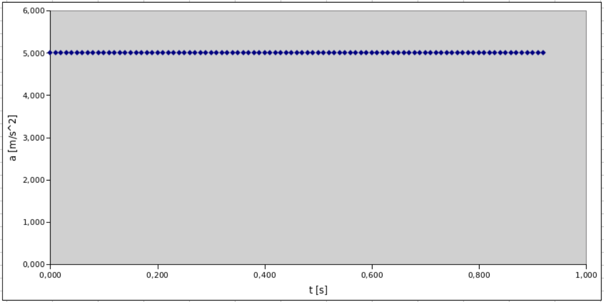
Figure: credit:[Wik20a]

Sreenshot: Let's dive into a problem

0^{th} order ODE: Constant force

$F_{ext} = k$

	A	B	C	I	J
1	t	F	a	dt	0,010
2	0,000	10,000	5,000	m	2,000
3	0,010	10,000	5,000	F_ext	10,000
4	0,020	10,000	5,000		
5	0,030	10,000	5,000		
6	0,040	10,000	5,000		
7	0,050	10,000	5,000		
8	0,060	10,000	5,000		
9	0,070	10,000	5,000		
10	0,080	10,000	5,000		
11	0,090	10,000	5,000		
12	0,100	10,000	5,000		
13	0,110	10,000	5,000		
14	0,120	10,000	5,000		
15	0,130	10,000	5,000		
16	0,140	10,000	5,000		
17	0,150	10,000	5,000		
18	0,160	10,000	5,000		
19	0,170	10,000	5,000		
20	0,180	10,000	5,000		
21	0,190	10,000	5,000		
22	0,200	10,000	5,000		
23	0,210	10,000	5,000		
24	0,220	10,000	5,000		
25	0,230	10,000	5,000		
26	0,240	10,000	5,000		
27	0,250	10,000	5,000		
28	0,260	10,000	5,000		
29	0,270	10,000	5,000		
30	0,280	10,000	5,000		
31	0,290	10,000	5,000		
32	0,300	10,000	5,000		
33	0,310	10,000	5,000		
34	0,320	10,000	5,000		
35	0,330	10,000	5,000		
36	0,340	10,000	5,000		
37	0,350	10,000	5,000		



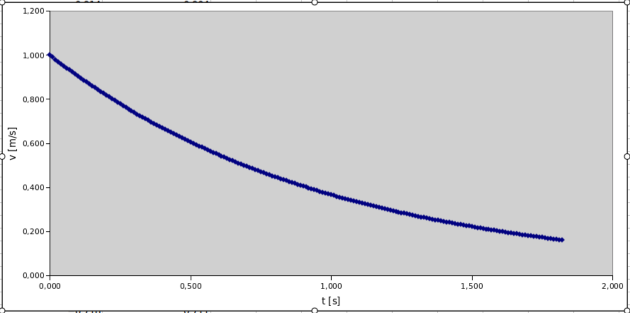
▸ See example

Sreenshot: Let's dive into a problem

1^{st} order ODE: Friction force

$$F_{ext} = -b \cdot v$$

	A	B	C	D	I	J
1	t	F	a	v	b	1,000
2	0,000			1,000	dt	0,010
3	0,010	−1,000	−1,000	0,990	m	1,000
4	0,020	−0,990	−0,990	0,980		
5	0,030	−0,980	−0,980	0,970		
6	0,040	−0,970	−0,970	0,961		
7	0,050	−0,961	−0,961	0,951		
8	0,060	−0,951	−0,951	0,941		
9	0,070	−0,941	−0,941	0,932		
10	0,080	−0,932	−0,932	0,923		
11	0,090	−0,923	−0,923	0,914		
12	0,100	−0,914	[illegible]	[illegible]		
13	0,110	−0,904				
14	0,120	−0,895				
15	0,130	−0,886				
16	0,140	−0,878				
17	0,150	−0,869				
18	0,160	−0,860				
19	0,170	−0,851				
20	0,180	−0,843				
21	0,190	−0,835				
22	0,200	−0,826				
23	0,210	−0,818				
24	0,220	−0,810				
25	0,230	−0,802				
26	0,240	−0,794				
27	0,250	−0,786				
28	0,260	−0,778				
29	0,270	−0,770				
30	0,280	−0,762				
31	0,290	−0,755				
32	0,300	−0,747				
33	0,310	−0,740				
34	0,320	−0,732				
35	0,330	−0,725				
36	0,340	−0,718	[illegible]	[illegible]		
37	0,350	−0,711	−0,711	0,703		
38	0,360	−0,703	−0,703	0,696		



See example

Table of Contents

Free fall set-up

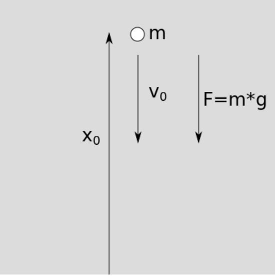


Figure: Experiment set-up

Equation of motion:

$$
\begin{aligned}
F_{ext} &= -mg, \\
a &= F_{ext}/m \\
dv/dt &= a \\
dx/dt &= v
\end{aligned}
$$

Table of Contents

A spreadsheet approach

time	$F(t,x,v)$	$a(t)$	$v(t)$ calculation	$x(t)$ calculation
t_0	$F_0 = F(t_0, x_0, v_0)$	$a_0 = F_0/m$	v_0 (initial cond.)	x_0 (initial cond.)
$t_1 = t_0 + \Delta t$	$F_1 = F(t_1, x_1, v_1)$	$a_1 = F_1/m$	$v_1 = v_0 + a_1 \Delta t$	$x_1 = x_0 + v_1 \Delta t$
$t_2 = t_1 + \Delta t$	$F_2 = F(t_2, x_2, v_2)$	$a_2 = F_2/m$	$v_2 = v_1 + a_2 \Delta t$	$x_2 = x_1 + v_2 \Delta t$
..	..	..	..	..
$t_n = t_{n-1} + \Delta t$	$F_n = F(t_n, x_n, v_n)$	$a_n = F_n/m$	$v_n = v_{n-1} + a_n \Delta t$	$x_n = x_{n-1} + v_n \Delta t$

Let us have a force in a cell L2, object mass in a cell I2, time advance in a cell I4, initial height in a cell E4 and initial velocity in a cell D4, then

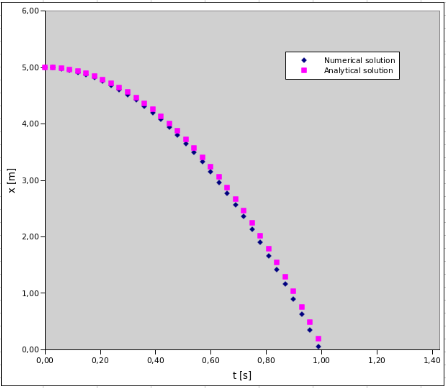
row	column A	column B	column C	column D	column E
4	0	-L2	B4/I2	any number (v_0 initial cond.)	any number (x_0 initial cond.)
5	A4+I4	-L2	B5/I2	D4+C5*I4	E4+D5*I4
6	A5+I4	-L2	B6/I2	D5+C6*I4	E5+D6*I4
7..N-1	..	..	..	..	..
N	A(N-1)+I4	-L2	BN/I2	D(N-1)+CN*I4	E(N-1)+DN*I4

So it is possible to specify only row #5 and then use copy row #5 and paste special to the consequent rows from #6 to #N.

See example

Sreenshot: Free fall (numerical and analytical comparison)

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Discrete solution					Analytical solution							
2	t	F	a	v	x			m	1,00	kg	F	9,81	N
3	[s]	[N]	[m/s/s]	[m/s]	m			g	9,81	m/s/s	t_fall	1,01	s
4	0,00			0,00	5,00	5,00		dt	0,03	s			
5	0,03	−9,81	−9,81	−0,29	4,99	5,00							
6	0,06	−9,81	−9,81	−0,59	4,97	4,98							
7	0,09	−9,81	−9,81	−0,88	4,95	4,96							
8	0,12	−9,81	−9,81	−1,18	4,91	4,93							
9	0,15	−9,81	−9,81	−1,47	4,87	4,89							
10	0,18	−9,81	−9,81	−1,77	4,81	4,84							
11	0,21	−9,81	−9,81	−2,06	4,75	4,78							
12	0,24	−9,81	−9,81	−2,35	4,68	4,72							
13	0,27	−9,81	−9,81	−2,65	4,60	4,64							
14	0,30	−9,81	−9,81	−2,94	4,51	4,56							
15	0,33	−9,81	−9,81	−3,24	4,42	4,47							
16	0,36	−9,81	−9,81	−3,53	4,31	4,36							
17	0,39	−9,81	−9,81	−3,83	4,20	4,25							
18	0,42	−9,81	−9,81	−4,12	4,07	4,13							
19	0,45	−9,81	−9,81	−4,41	3,94	4,01							
20	0,48	−9,81	−9,81	−4,71	3,80	3,87							
21	0,51	−9,81	−9,81	−5,00	3,65	3,72							
22	0,54	−9,81	−9,81	−5,30	3,49	3,57							
23	0,57	−9,81	−9,81	−5,59	3,32	3,41							
24	0,60	−9,81	−9,81	−5,89	3,15	3,23							
25	0,63	−9,81	−9,81	−6,18	2,96	3,05							
26	0,66	−9,81	−9,81	−6,47	2,77	2,86							
27	0,69	−9,81	−9,81	−6,77	2,56	2,66							
28	0,72	−9,81	−9,81	−7,06	2,35	2,46							
29	0,75	−9,81	−9,81	−7,36	2,13	2,24							
30	0,78	−9,81	−9,81	−7,65	1,90	2,02							
31	0,81	−9,81	−9,81	−7,95	1,66	1,78							
32	0,84	−9,81	−9,81	−8,24	1,42	1,54							
33	0,87	−9,81	−9,81	−8,53	1,16	1,29							
34	0,90	−9,81	−9,81	−8,83	0,89	1,03							
35	0,93	−9,81	−9,81	−9,12	0,62	0,76							
36	0,96	−9,81	−9,81	−9,42	0,34	0,48							
37	0,99	−9,81	−9,81	−9,71	0,05	0,19							
38	1,02	−9,81	−9,81	−10,01	−0,25	−0,10							
39	1,05	−9,81	−9,81	−10,30	−0,56	−0,41							



See example

A spreadsheet approach cont.

row	column A	column B	column C	column D	column E
4	0	-L2	B4/I2	any number (v_0 initial cond.)	any number (x_0 initial cond.)
5	A4+I4	-L2	B5/I2	D4+C5*I4	E4+D5*I4
6	A5+I4	-L2	B6/I2	D5+C6*I4	E5+D6*I4
7..N-1	..	..	..	..	..
N	A(N-1)+I4	-L2	BN/I2	D(N-1)+CN*I4	E(N-1)+DN*I4

A more convenient way is to name basic parameters, e.g. Let us have a force in a cell L2 named F, object mass in a cell I2 named m, time advance in a cell I4 named dt, initial height in a cell E4 and initial velocity in a cell D4, then

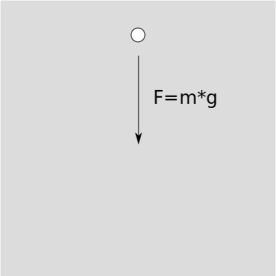
row	column A	column B	column C	column D	column E
4	0	$-F$	$B4/m$	any number (v_0 initial cond.)	any number (x_0 initial cond.)
5	$A4+dt$	$-F$	$B5/m$	$D4+C5*dt$	$E4+D5*dt$
6	$A5+dt$	$-F$	$B6/m$	$D5+C6*dt$	$E5+D6*dt$
7..N-1	..	..	..	..	..
N	$A(N-1)+dt$	$-F$	BN/m	$D(N-1)+CN*dt$	$E(N-1)+DN*dt$

See example

Table of Contents

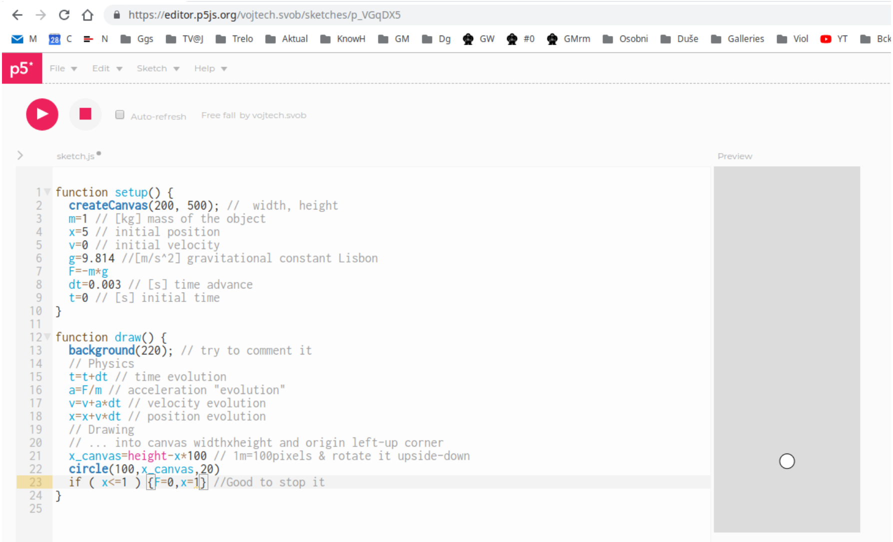
A processing approach

```
function setup() {
  createCanvas(200, 500); //  width, height
  m=1 // [kg] mass of the object
  x=5 // initial position
  v=0 // initial velocity
  g=9.814 //[m/s^2] gravitational constant Lisbon
  F=-m*g
  dt=0.003 // [s] time advance
  t=0 // [s] initial time
}

function draw() {
  background(220); // try to comment it
  // Physics
  t=t+dt // time evolution
  a=F/m // acceleration "evolution"
  v=v+a*dt // velocity evolution
  x=x+v*dt // position evolution
  // Drawing
  // ... into canvas widthxheight and origin left-up corner
  x_canvas=height-x*100 // 1m=100pixels & rotate it upside-down
  circle(100,x_canvas,20)
  if ( x<=0 ) {F=0,x=0} //Good to stop it
}
```



See example

Sreenshot: Free fall

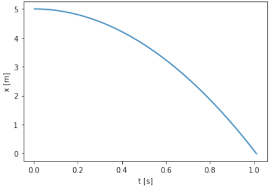


▸ See example

Table of Contents

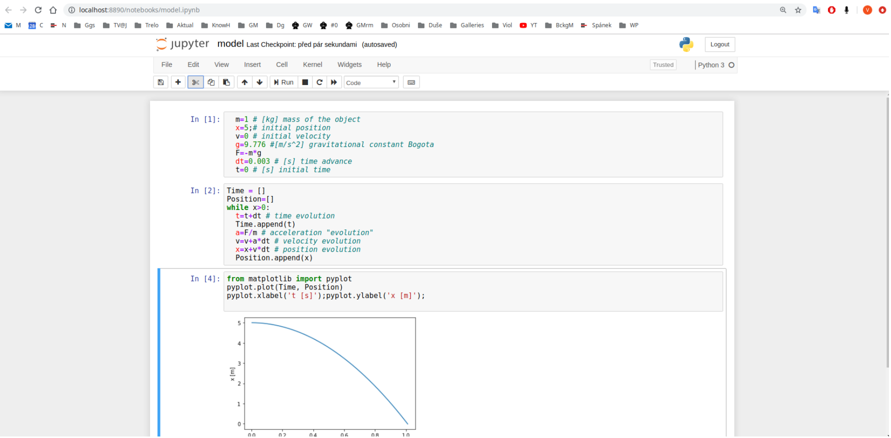
A python@Jupyter notebook approach

```
m=1 # [kg] mass of the object
x=5;# initial position
v=0 # initial velocity
g=9.814 #[m/s^2] gravitational constant Lisbon
F=-m*g
dt=0.003 # [s] time advance
t=0 # [s] initial time

Time = []
Position=[]
while x>0:
   t=t+dt # time evolution
   Time.append(t)
   a=F/m # acceleration "evolution"
   v=v+a*dt # velocity evolution
   x=x+v*dt # position evolution
   Position.append(x)

from matplotlib import pyplot
pyplot.plot(Time, Position)
pyplot.xlabel('t [s]');pyplot.ylabel('x [m]');
```



See example

Sreenshot: Free fall



See example

Table of Contents

Pendulum set-up

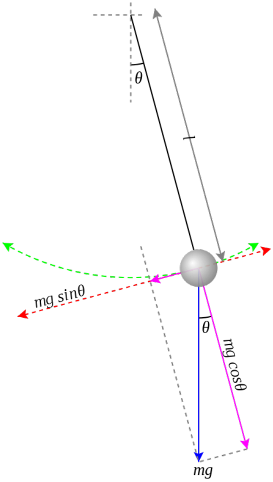


Figure: Pendulum setup.
credit:[Wik20c]

Equation of motion:

$$
\begin{aligned}
F &= -mg\sin\theta = ma, \\
a &= -g\sin\theta \\
a &= \frac{d^2 s}{dt^2} = \ell\frac{d^2\theta}{dt^2} = \ell\epsilon,
\end{aligned}
$$

$$\frac{d^2\theta}{dt^2} + \frac{g}{\ell}\sin\theta = 0,$$

$$\frac{d^2\theta}{dt^2} + \frac{g}{\ell}\theta = 0 \quad \text{(small angle approx.)}.$$

Table of Contents

A spreadsheet approach modification from translational to rotational system

time	$F(t, \theta, \omega)$	$\epsilon(t)$	$\omega(t)$ calculation	$\theta(t)$ calculation
t_0	$F_0 = F(t_0, \theta_0, \omega_0)$	$\epsilon_0 = F_0/m$	ω_0 (initial cond.)	θ_0 (initial cond.)
$t_1 = t_0 + \Delta t$	$F_1 = F(t_1, \theta_1, \omega_1)$	$\epsilon_1 = F_1/m$	$\omega_1 = \omega_0 + \epsilon_1 \Delta t$	$\theta_1 = \theta_0 + \omega_1 \Delta t$
$t_2 = t_1 + \Delta t$	$F_2 = F(t_2, \theta_2, \omega_2)$	$\epsilon_2 = F_2/m$	$\omega_2 = \omega_1 + \epsilon_2 \Delta t$	$\theta_2 = \theta_1 + \omega_2 \Delta t$
..	..	..	..	..
$t_n = t_{n-1} + \Delta t$	$F_n = F(t_n, \theta_n, \omega_n)$	$\epsilon_n = F_n/m$	$\omega_n = \omega_{n-1} + \epsilon_n \Delta t$	$\theta_n = \theta_{n-1} + \omega_n \Delta t$

Let's specify and name basic parameters: object mass in a cell J1 named m, time advance in a cell J3 named dt, length of the pendulum in J4 named l, gravitational constant in J2 named g, initial angle in a cell E4 and initial velocity in a cell D4, then

row	column A	column B	column C	column D	column E
4	0		B4/m	any number (ω_0 initial cond.)	any number (θ_0 initial cond.)
5	A4+dt	$-m \cdot g \cdot \sin(E4)$	B5/m	D4+C5*dt	E4+D5*dt
6	A5+dt	$-m \cdot g \cdot \sin(E5)$	B6/m	D5+C6*dt	E5+D6*dt
7..N-1	..	..	..	..	..
N	A(N-1)+dt	$-m \cdot g \cdot \sin(E(N-1))$	BN/m	D(N-1)+CN*dt	E(N-1)+DN*dt

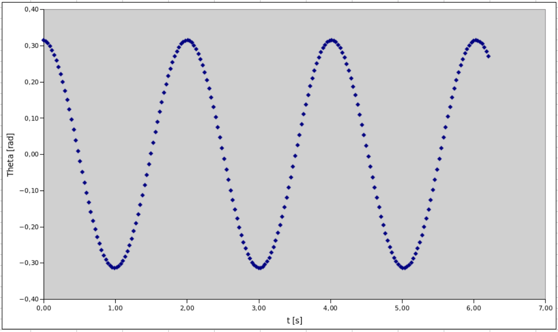
See example

Sreenshot: Pendulum basic @ spreadsheet

t	F	eps	omega	theta
[s]	[N]	rad/s*2	rad/s	rad
0,00			0,00	0,31
0,03	−3,03	−3,03	−0,09	0,31
0,06	−3,01	−3,01	−0,18	0,31
0,09	−2,96	−2,96	−0,27	0,30
0,12	−2,88	−2,88	−0,36	0,29
0,15	−2,78	−2,78	−0,44	0,27
0,18	−2,66	−2,66	−0,52	0,26
0,21	−2,51	−2,51	−0,59	0,24
0,24	−2,34	−2,34	−0,66	0,22
0,27	−2,15	−2,15	−0,73	0,20
0,30	−1,94	−1,94	−0,79	0,18
0,33	−1,71	−1,71	−0,84	0,15
0,36	−1,47	−1,47	−0,88	0,12
0,39	−1,21	−1,21	−0,92	0,10
0,42	−0,94	−0,94	−0,95	0,07
0,45	−0,66	−0,66	−0,97	0,04
0,48	−0,38	−0,38	−0,98	0,01
0,51	−0,09	−0,09	−0,98	−0,02
0,54	0,20	0,20	−0,98	−0,05
0,57	0,49	0,49	−0,96	−0,08
0,60	0,77	0,77	−0,94	−0,11
0,63	1,04	1,04	−0,91	−0,13
0,66	1,31	1,31	−0,87	−0,16
0,69	1,56	1,56	−0,82	−0,18
0,72	1,80	1,80	−0,77	−0,21
0,75	2,02	2,02	−0,71	−0,23
0,78	2,22	2,22	−0,64	−0,25
0,81	2,40	2,40	−0,57	−0,26
0,84	2,57	2,57	−0,49	−0,28
0,87	2,71	2,71	−0,41	−0,29
0,90	2,82	2,82	−0,32	−0,30
0,93	2,91	2,91	−0,24	−0,31
0,96	2,98	2,98	−0,15	−0,31
0,99	3,02	3,02	−0,06	−0,31
1,02	3,04	3,04	0,03	−0,31
1,05	3,03	3,03	0,13	−0,31
1,08	2,99	2,99	0,21	−0,30

m	1,00	kg
g	9,81	m/s**2
dt	0,03	s
l	1,00	m

T	2,01	s



See example

Sreenshot: Pendulum basic @ processing

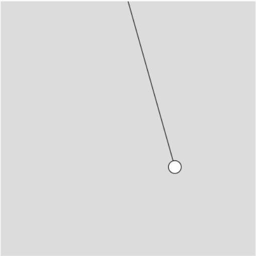
https://editor.p5js.org/vojtech.svob/sketches/vTEaAkgs

Pendulum - basic version by vojtech.svob

sketch.js Saved: 3 minutes ago

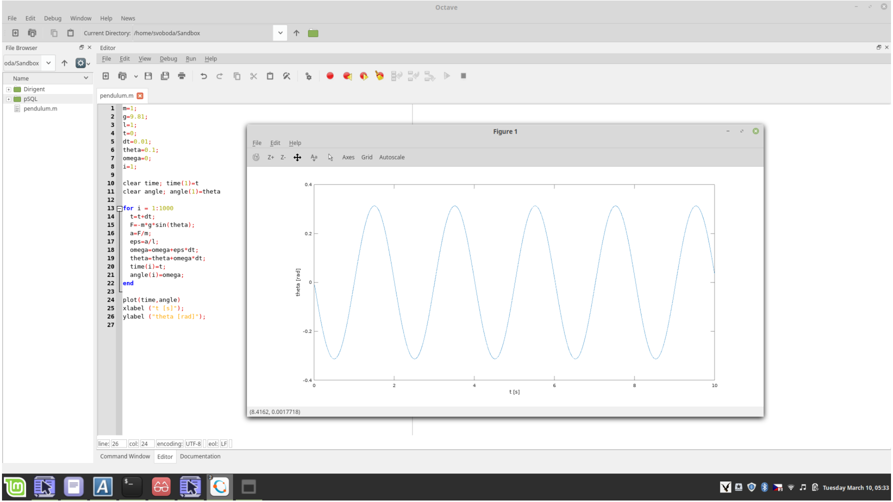
```
function setup() {
  createCanvas(400, 400);
  m=2
  l=2.705
  g=9.814
  dt=0.02
  t=0
  theta = 3.14/10; // Pendulum initial angle theta
  omega = 0; // Initial angular velocity
  C = 2; // Center point
}

function draw() {
  background(220);
  // physics
  t = t + dt;
  F=-m*g*sin(theta)
  epsilon = (F/m)/l; //angular acceleration
  omega = omega + epsilon * dt;
  theta = theta + omega * dt;
  xp = C - l * sin(theta); // X coordinate of pendulum ball
  yp = l * cos(theta); // Y coordinate of pendulum ball
  //draw it
  ppm=100 //scale it to the canvas (from meters to pixels)
  line(C*ppm, 0, xp*ppm, yp*ppm);
  ellipse(xp*ppm, yp*ppm, 20, 20);
}
```

Preview



See example

Sreenshot: Pendulum basic @ octave (matlab)

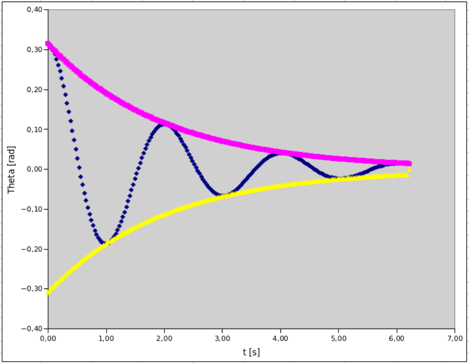


▸ See example

Table of Contents

Sreenshot: Pendulum with friction

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	t	F	eps	omega	theta	Envelope			m	1,00	kg	T	2,01	s
2	[s]	[N]	rad/s*2	rad/s	rad				g	9,81	m/s**2			
3	0,00			0,00	0,31	0,31	−0,31		dt	0,03	s			
4	0,03	−3,03	−3,03	−0,09	0,31	0,31	−0,30		l	1,00	m			
5	0,06	−2,92	−2,92	−0,18	0,31	0,30	−0,30		b	1,00	kg/s	d	0,50	1/s
6	0,09	−2,78	−2,78	−0,26	0,30	0,30	−0,30							
7	0,12	−2,62	−2,62	−0,34	0,29	0,30	−0,29							
8	0,15	−2,45	−2,45	−0,41	0,28	0,29	−0,29							
9	0,18	−2,26	−2,26	−0,48	0,26	0,29	−0,28							
10	0,21	−2,05	−2,05	−0,54	0,24	0,28	−0,28							
11	0,24	−1,84	−1,84	−0,60	0,23	0,28	−0,27							
12	0,27	−1,61	−1,61	−0,65	0,21	0,27	−0,27							
13	0,30	−1,38	−1,38	−0,69	0,19	0,27	−0,27							
14	0,33	−1,14	−1,14	−0,72	0,17	0,27	−0,26							
15	0,36	−0,89	−0,89	−0,75	0,14	0,26	−0,26							
16	0,39	−0,65	−0,65	−0,77	0,12	0,26	−0,25							
17	0,42	−0,40	−0,40	−0,78	0,10	0,25	−0,25							
18	0,45	−0,16	−0,16	−0,79	0,07	0,25	−0,25							
19	0,48	0,07	0,07	−0,78	0,05	0,25	−0,24							
20	0,51	0,30	0,30	−0,77	0,03	0,24	−0,24							
21	0,54	0,52	0,52	−0,76	0,00	0,24	−0,24							
22	0,57	0,73	0,73	−0,74	−0,02	0,24	−0,23							
23	0,60	0,92	0,92	−0,71	−0,04	0,23	−0,23							
24	0,63	1,10	1,10	−0,68	−0,06	0,23	−0,23							
25	0,66	1,27	1,27	−0,64	−0,08	0,23	−0,22							
26	0,69	1,42	1,42	−0,60	−0,10	0,22	−0,22							
27	0,72	1,55	1,55	−0,55	−0,11	0,22	−0,22							
28	0,75	1,66	1,66	−0,50	−0,13	0,22	−0,21							
29	0,78	1,76	1,76	−0,45	−0,14	0,21	−0,21							
30	0,81	1,84	1,84	−0,39	−0,15	0,21	−0,21							
31	0,84	1,90	1,90	−0,33	−0,16	0,21	−0,20							
32	0,87	1,94	1,94	−0,28	−0,17	0,20	−0,20							
33	0,90	1,96	1,96	−0,22	−0,18	0,20	−0,20							
34	0,93	1,96	1,96	−0,16	−0,18	0,20	−0,19							
35	0,96	1,95	1,95	−0,10	−0,19	0,19	−0,19							
36	0,99	1,92	1,92	−0,04	−0,19	0,19	−0,19							
37	1,02	1,87	1,87	0,01	−0,19	0,19	−0,19							
38	1,05	1,81	1,81	0,07	−0,19	0,19	−0,18							
39	1,08	1,74	1,74	0,12	−0,18	0,18	−0,18							



See example

Sreenshot: Pendulum with friction @ processing

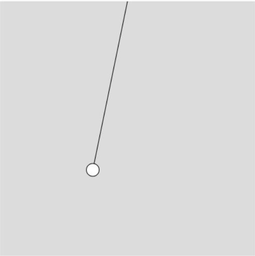
p5* File Edit Sketch Help

Auto-refresh Pendulum with friction

sketch.js

```
function setup() {
  createCanvas(400, 400);
  m=2
  l=2.705
  g=9.814
  dt=0.02
  b=0.1 //friction coefficient
  t=0
  theta = 3.14/10; // Pendulum initial angle theta
  omega = 0; // Initial angular velocity
  C = 2; // Center point
}

function draw() {
  background(220);
  // physics
  t = t + dt;
  F=-m*g*sin(theta)-b*(l*omega)
  epsilon = (F/m)/l; //angular acceleration
  omega = omega + epsilon * dt;
  theta = theta + omega * dt;
  xp = C - l * sin(theta); // X coordinate of pendulum ball
  yp = l * cos(theta); // Y coordinate of pendulum ball
  //draw it
  ppm=100 //scale it to the canvas (from meters to pixels)
  line(C*ppm, 0, xp*ppm, yp*ppm);
  ellipse(xp*ppm, yp*ppm, 20, 20);
}
```

Preview



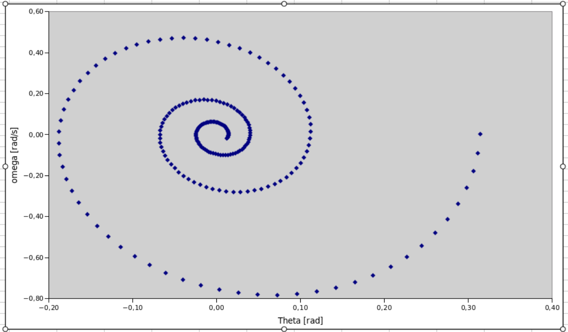
▸ See example

Table of Contents

Sreenshot: Pendulum with friction - phase space

	A	B	C	D	E
1	t	F	eps	omega	theta
2	[s]	[N]	rad/s*2	rad/s	rad
3	0,00			0,00	0,31
4	0,03	−3,03	−3,03	−0,09	0,31
5	0,06	−2,92	−2,92	−0,18	0,31
6	0,09	−2,78	−2,78	−0,26	0,30
7	0,12	−2,62	−2,62	−0,34	0,29
8	0,15	−2,45	−2,45	−0,41	0,28
9	0,18	−2,26	−2,26	−0,48	0,26
10	0,21	−2,05	−2,05	−0,54	0,24
11	0,24	−1,84	−1,84	−0,60	0,23
12	0,27	−1,61	−1,61	−0,65	0,21
13	0,30	−1,38	−1,38	−0,69	0,19
14	0,33	−1,14	−1,14	−0,72	0,17
15	0,36	−0,89	−0,89	−0,75	0,14
16	0,39	−0,65	−0,65	−0,77	0,12
17	0,42	−0,40	−0,40	−0,78	0,10
18	0,45	−0,16	−0,16	−0,79	0,07
19	0,48	0,07	0,07	−0,78	0,05
20	0,51	0,30	0,30	−0,77	0,03
21	0,54	0,52	0,52	−0,76	0,00
22	0,57	0,73	0,73	−0,74	−0,02
23	0,60	0,92	0,92	−0,71	−0,04
24	0,63	1,10	1,10	−0,68	−0,06
25	0,66	1,27	1,27	−0,64	−0,08
26	0,69	1,42	1,42	−0,60	−0,10
27	0,72	1,55	1,55	−0,55	−0,11
28	0,75	1,66	1,66	−0,50	−0,13
29	0,78	1,76	1,76	−0,45	−0,14
30	0,81	1,84	1,84	−0,39	−0,15
31	0,84	1,90	1,90	−0,33	−0,16
32	0,87	1,94	1,94	−0,28	−0,17
33	0,90	1,96	1,96	−0,22	−0,18
34	0,93	1,96	1,96	−0,16	−0,18
35	0,96	1,95	1,95	−0,10	−0,19
36	0,99	1,92	1,92	−0,04	−0,19
37	1,02	1,87	1,87	0,01	−0,19
38	1,05	1,81	1,81	0,07	−0,19

	I	J	K	L	M	N
1	m	1,00	kg	T	2,01	s
2	g	9,81	m/s**2			
3	dt	0,03	s			
4	l	1,00	m			
5	b	1,00	kg/s	d	0,50	1/s



▸ See example

Table of Contents

Energy of the Pendulum

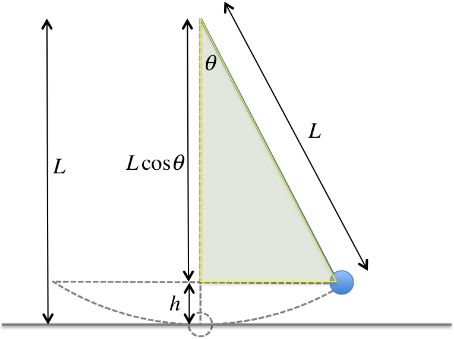
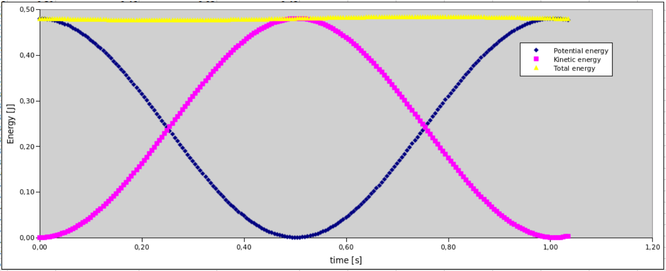


Figure: credit:[Lee20]

Sreenshot: Pendulum - energy conservation analysis

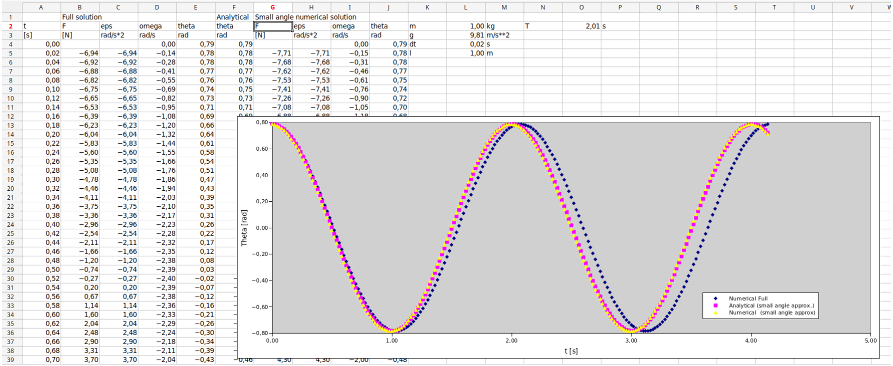
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	t	F	eps	omega	theta	Potential energy	Kinetic energy	Total energy	m	1,00	kg	T	2,01	s		
2	[s]	[N]	rad/s*2	rad/s	rad	J	J	J	g	9,81	m/s**2					
3	0,00			0,00	0,31	0,48	0,00	0,48	dt	0,01	s					
4	0,01	−3,03	−3,03	−0,02	0,31	0,48	0,00	0,48	l	1,00	m					
5	0,01	−3,03	−3,03	−0,03	0,31	0,48	0,00	0,48								
6	0,01	−3,03	−3,03	−0,05	0,31	0,48	0,00	0,48								
7	0,02	−3,03	−3,03	−0,06	0,31	0,48	0,00	0,48								
8	0,03	−3,03	−3,03	−0,08	0,31	0,48	0,00	0,48								
9	0,03	−3,02	−3,02	−0,09	0,31	0,48	0,00	0,48								
10	0,04	−3,02	−3,02	−0,11	0,31	0,47	0,01	0,48								
11	0,04	−3,01	−3,01	−0,12	0,31	0,47	0,01	0,48								
12	0,04	−3,01	−3,01	−0,14	0,31	0,47	0,01	0,48								
13	0,05	−3,00	−3,00	−0,15	0,31	0,47	0,01	0,48								
14	0,05	−2,99	−2,99	−0,17	0,31	0,47	0,01	0,48								
15	0,06	−2,99	−2,99	−0,1												
16	0,06	−2,98	−2,98	−0,2												
17	0,07	−2,97	−2,97	−0,2												
18	0,07	−2,96	−2,96	−0,2												
19	0,08	−2,95	−2,95	−0,2												
20	0,09	−2,94	−2,94	−0,2												
21	0,09	−2,92	−2,92	−0,2												
22	0,10	−2,91	−2,91	−0,2												
23	0,10	−2,90	−2,90	−0,3												
24	0,11	−2,88	−2,88	−0,3												
25	0,11	−2,87	−2,87	−0,3												
26	0,12	−2,85	−2,85	−0,3												
27	0,12	−2,84	−2,84	−0,3												
28	0,13	−2,82	−2,82	−0,3												
29	0,13	−2,80	−2,80	−0,3												
30	0,14	−2,79	−2,79	−0,4												
31	0,14	−2,77	−2,77	−0,4												
32	0,15	−2,75	−2,75	−0,4												
33	0,15	−2,73	−2,73	−0,4												
34	0,16	−2,71	−2,71	−0,4												
35	0,16	−2,69	−2,69	−0,4												
36	0,17	−2,66	−2,66	−0,4												
37	0,17	−2,64	−2,64	−0,4												
38	0,18	−2,62	−2,62	−0,5												
39	0,18	−2,59	−2,59	−0,52	0,26	0,34	0,13	0,48								



See example

Table of Contents

Sreenshot: Pendulum - small angle approximation analysis



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1		Full solution				Analytical	Small angle numerical solution									
2	t	F	eps	omega	theta	theta	F	eps	omega	theta	m	1,00	kg	T	2,01	s
3	[s]	[N]	rad/s*2	rad/s	rad	rad	[N]	rad/s*2	rad/s	rad	g	9,81	m/s**2			
4	0,00			0,00	0,79	0,79			0,00	0,79	dt	0,02	s			
5	0,02	-6,94	-6,94	-0,14	0,78	0,78	-7,71	-7,71	-0,15	0,78	l	1,00	m			
6	0,04	-6,92	-6,92	-0,28	0,78	0,78	-7,68	-7,68	-0,31	0,78						
7	0,06	-6,88	-6,88	-0,41	0,77	0,77	-7,62	-7,62	-0,46	0,77						
8	0,08	-6,82	-6,82	-0,55	0,76	0,76	-7,53	-7,53	-0,61	0,75						
9	0,10	-6,75	-6,75	-0,69	0,74	0,75	-7,41	-7,41	-0,76	0,74						
10	0,12	-6,65	-6,65	-0,82	0,73	0,73	-7,26	-7,26	-0,90	0,72						
11	0,14	-6,53	-6,53	-0,95	0,71	0,71	-7,08	-7,08	-1,05	0,70						
12	0,16	-6,39	-6,39	-1,08	0,69											
13	0,18	-6,23	-6,23	-1,20	0,66											
14	0,20	-6,04	-6,04	-1,32	0,64											
15	0,22	-5,83	-5,83	-1,44	0,61											
16	0,24	-5,60	-5,60	-1,55	0,58											
17	0,26	-5,35	-5,35	-1,66	0,54											
18	0,28	-5,08	-5,08	-1,76	0,51											
19	0,30	-4,78	-4,78	-1,86	0,47											
20	0,32	-4,46	-4,46	-1,94	0,43											
21	0,34	-4,11	-4,11	-2,03	0,39											
22	0,36	-3,75	-3,75	-2,10	0,35											
23	0,38	-3,36	-3,36	-2,17	0,31											
24	0,40	-2,96	-2,96	-2,23	0,26											
25	0,42	-2,54	-2,54	-2,28	0,22											
26	0,44	-2,11	-2,11	-2,32	0,17											
27	0,46	-1,66	-1,66	-2,35	0,12											
28	0,48	-1,20	-1,20	-2,38	0,08											
29	0,50	-0,74	-0,74	-2,39	0,03											
30	0,52	-0,27	-0,27	-2,40	-0,02											
31	0,54	0,20	0,20	-2,39	-0,07											
32	0,56	0,67	0,67	-2,38	-0,12											
33	0,58	1,14	1,14	-2,36	-0,16											
34	0,60	1,60	1,60	-2,33	-0,21											
35	0,62	2,04	2,04	-2,29	-0,26											
36	0,64	2,48	2,48	-2,24	-0,30											
37	0,66	2,90	2,90	-2,18	-0,34											
38	0,68	3,31	3,31	-2,11	-0,39											
39	0,70	3,70	3,70	-2,04	-0,43											

▸ See example

Table of Contents

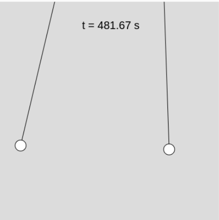
Sreenshot: Two pendulums

p5* File Edit Sketch Help

Auto-refresh Two pendulums by vojtech.svob

sketch.js Saved: just now Preview

```
function setup() {
  createCanvas(400, 400);
  m=2
  l=2.705
  //https://en.wikipedia.org/wiki/Gravity_of_Earth
  g1=9.78 //equator T1=3.304 s
  g2=9.832 //pole T2=3.296 s
  dt=0.02
  t=0
  theta1 = theta2 = 3.14/10; // Pendulum initial angle theta
  omega1 = omega2= 0; // Initial angular velocity
  C1 = 1; C2 = 3;// Center points}

function draw() {
  background(220);
  t = t + dt;
  // First pendulum
  F1=-m*g1*sin(theta1)
  epsilon1 = (F1/m)/l; //angular acceleration
  omega1 = omega1 + epsilon1 * dt;
  theta1 = theta1 + omega1 * dt;
  xp1 = C1 - l * sin(theta1); // X coordinate of pendulum ball
  yp1 = l * cos(theta1); // Y coordinate of pendulum ball
  //draw it
  ppm=100 //scale it to the canvas (from meters to pixels)
  line(C1*ppm, 0, xp1*ppm, yp1*ppm);
  ellipse(xp1*ppm, yp1*ppm, 20, 20);
    // Second pendulum
  F2=-m*g2*sin(theta2)
  epsilon2 = (F2/m)/l; //angular acceleration
  omega2 = omega2 + epsilon2 * dt;
  theta2 = theta2 + omega2 * dt;
  xp2 = C2 - l * sin(theta2); // X coordinate of pendulum ball
  yp2 = l * cos(theta2); // Y coordinate of pendulum ball
  //draw it
  ppm=100 //scale it to the canvas (from meters to pixels)
  line(C2*ppm, 0, xp2*ppm, yp2*ppm);
  ellipse(xp2*ppm, yp2*ppm, 20, 20);

  textSize(20);  text("t = "+nf(t,0,2)+" s", 150,50);

}
```



▸ See example

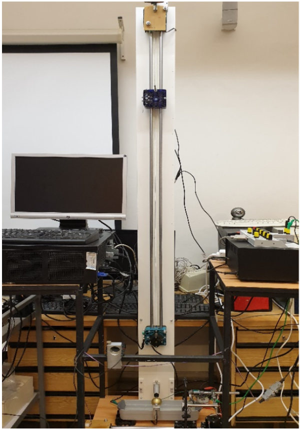
Table of Contents

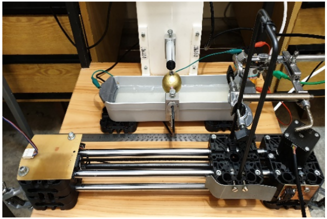
Table of Contents

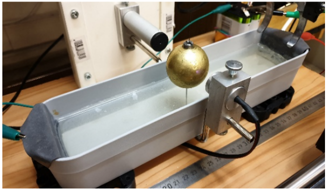
Pendulum in Prague

Parameters:

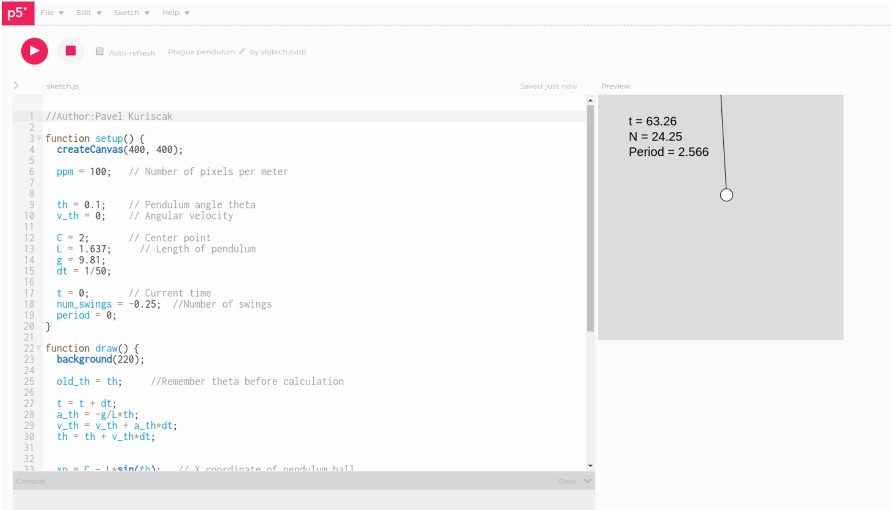
$l = 1.637$ m, $g = 9.810$ (Charles Univ.) or 9.834 (Wolfram) or 9.813 (Wiki) m/s^2





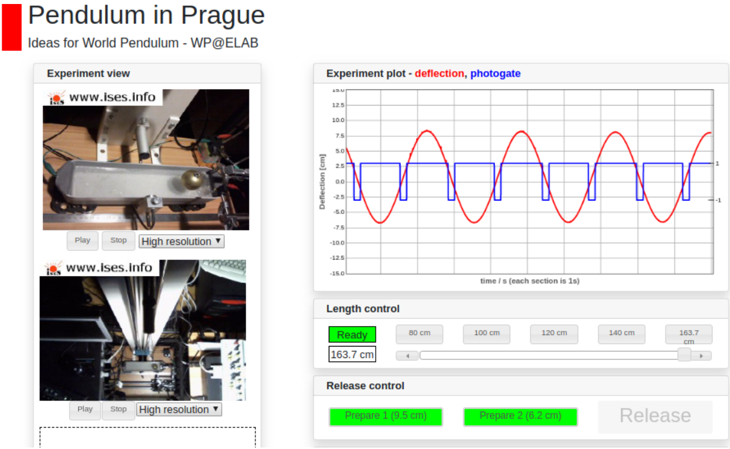


Sreenshot: Pendulum "advanced" @ processing



See example

Sreenshot: Pendulum in Prague

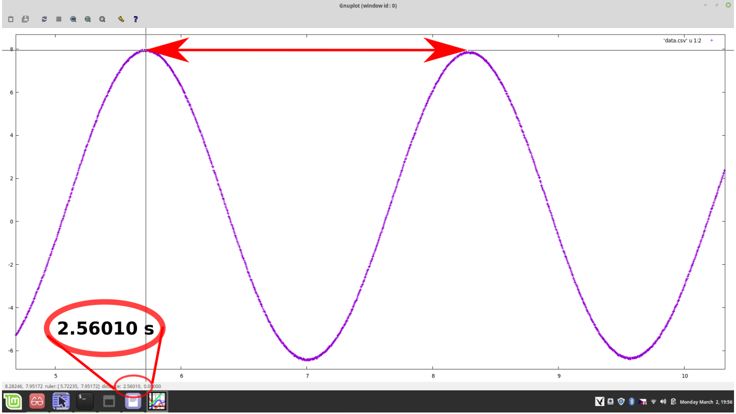


See example

Period

via Gnuplot

set datafile separator ',';plot 'data.csv' u 1:2



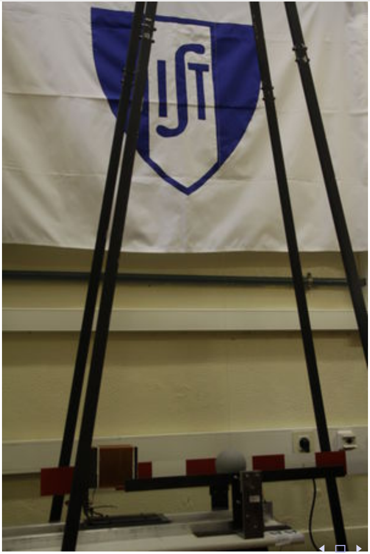
data.csv

Table of Contents

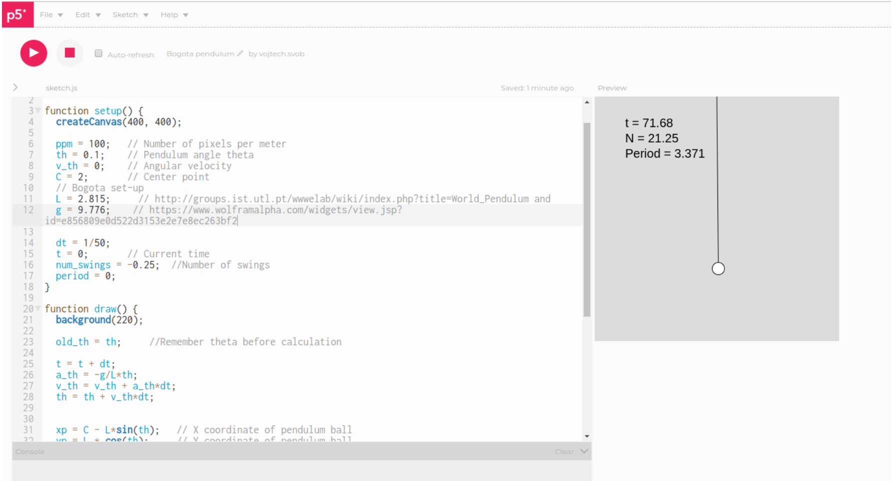
World Pendulum

Parameters:

$l \approx 2.81$ m, $g \approx 9.8$ m/s^2



Sreenshot: Pendulum "advanced" @ processing

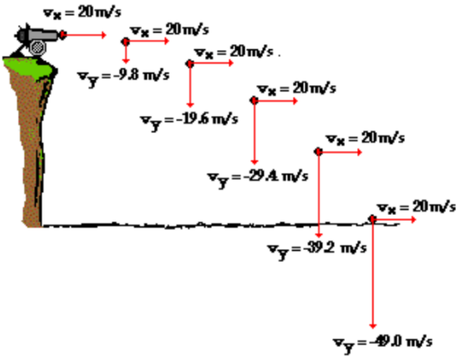


See example

Table of Contents

Table of Contents

Sreenshot: Experiment setup (credit:The Physics clasroom)

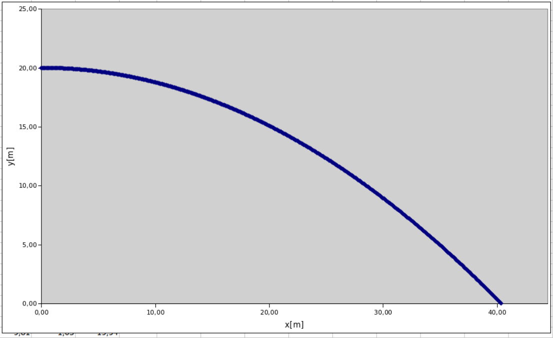


▸ See example

Sreenshot: Spreadsheet approach

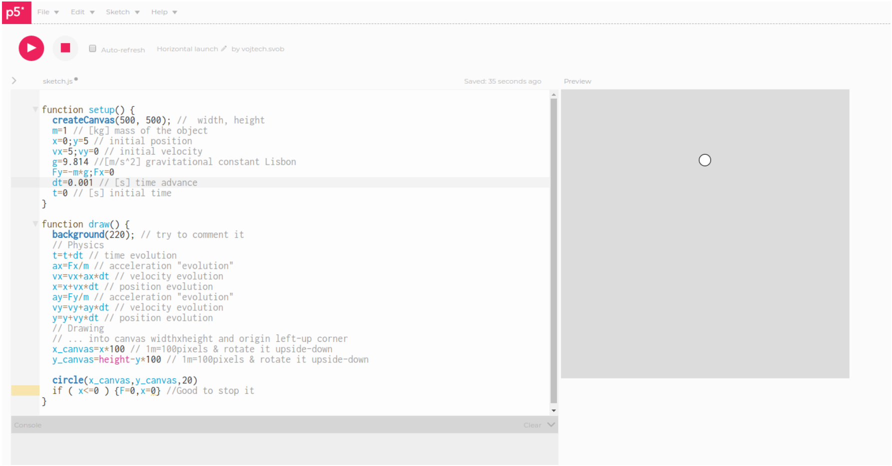
	A	B	C	D	E	F	G	H	I
1									
2	t	Fx	ax	vx	x	Fy	ay	vy	y
3	[s]	[N]	[m/s/s]	[m/s]	m	[N]	[m/s/s]	[m/s]	m
4	0,00	0,00	0,00	20,00	0,00	−9,81	−9,81	0,00	20,00
5	0,00	0,00	0,00	20,00	0,06	−9,81	−9,81	−0,03	20,00
6	0,01	0,00	0,00	20,00	0,12	−9,81	−9,81	−0,06	20,00
7	0,01	0,00	0,00	20,00	0,18	−9,81	−9,81	−0,09	20,00
8	0,01	0,00	0,00	20,00	0,24	−9,81			
9	0,01	0,00	0,00	20,00	0,30	−9,81			
10	0,02	0,00	0,00	20,00	0,36	−9,81			
11	0,02	0,00	0,00	20,00	0,42	−9,81			
12	0,02	0,00	0,00	20,00	0,48	−9,81			
13	0,03	0,00	0,00	20,00	0,54	−9,81			
14	0,03	0,00	0,00	20,00	0,60	−9,81			
15	0,03	0,00	0,00	20,00	0,66	−9,81			
16	0,04	0,00	0,00	20,00	0,72	−9,81			
17	0,04	0,00	0,00	20,00	0,78	−9,81			
18	0,04	0,00	0,00	20,00	0,84	−9,81			
19	0,05	0,00	0,00	20,00	0,90	−9,81			
20	0,05	0,00	0,00	20,00	0,96	−9,81			
21	0,05	0,00	0,00	20,00	1,02	−9,81			
22	0,05	0,00	0,00	20,00	1,08	−9,81			
23	0,06	0,00	0,00	20,00	1,14	−9,81			
24	0,06	0,00	0,00	20,00	1,20	−9,81			
25	0,06	0,00	0,00	20,00	1,26	−9,81			
26	0,07	0,00	0,00	20,00	1,32	−9,81			
27	0,07	0,00	0,00	20,00	1,38	−9,81			
28	0,07	0,00	0,00	20,00	1,44	−9,81			
29	0,08	0,00	0,00	20,00	1,50	−9,81			
30	0,08	0,00	0,00	20,00	1,56	−9,81			
31	0,08	0,00	0,00	20,00	1,62	−9,81			
32	0,08	0,00	0,00	20,00	1,68	−9,81			
33	0,09	0,00	0,00	20,00	1,74	−9,81			
34	0,09	0,00	0,00	20,00	1,80	−9,81			
35	0,09	0,00	0,00	20,00	1,86	−9,81			
36	0,10	0,00	0,00	20,00	1,92	−9,81			
37	0,10	0,00	0,00	20,00	1,98	−9,81			
38	0,10	0,00	0,00	20,00	2,04	−9,81			
39	0,11	0,00	0,00	20,00	2,10	−9,81			

	K	L	M	N	O	P	R	S
1	Parameters							
2	m	1,000	kg	Fx	0		T_fall	2,02
3	g	9,810	m/s/s		Fy	−9,81	x_fall	40,39
4	dt	0,003	s					44,42



▸ See example

Sreenshot: Processing approach



▸ See example

Table of Contents

Runge Kutta method

Let an initial value problem be specified:

$$\dot{y} = f(t, y), \quad y(t_0) = y_0$$

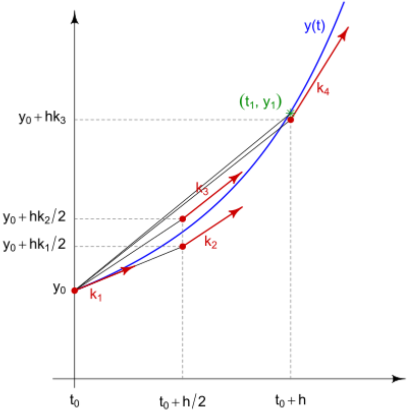


Figure: Slopes used by the classical Runge-Kutta method [Wik20e]

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

$$k_1 = h\ f(t_n, y_n),$$
$$k_2 = h\ f\left(t_n + \frac{h}{2}, y_n + \frac{k_1}{2}\right),$$
$$k_3 = h\ f\left(t_n + \frac{h}{2}, y_n + \frac{k_2}{2}\right),$$
$$k_4 = h\ f\left(t_n + h, y_n + k_3\right).$$

Runge-Kutta versus Euler method

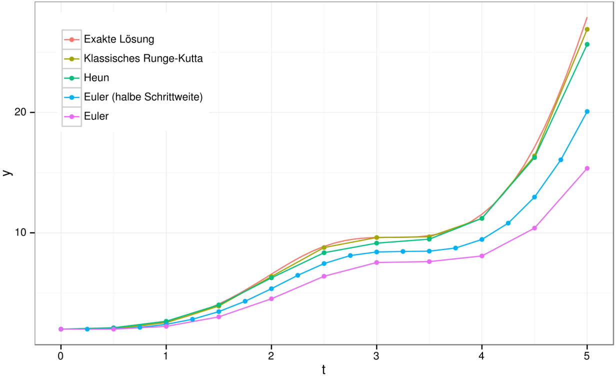


Figure: Runge-Kutta methods for the differential equation $y' = sin(t)^2 \cdot y$ [Wik20e]

Table of Contents

Sreenshot: odeint: Python solver

jupyter model (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

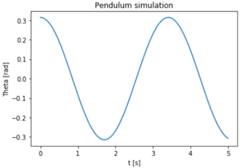
Run Code

In [1]:
```python
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import odeint
```

In [10]:
```python
g=9.81
l=2.85
b=0.0 #With friction
def dTheta_dt(Theta, t):
    return [Theta[1], -b*Theta[1] -g/l*np.sin(Theta[0])]
Theta0 = [np.pi/10, 0]
t = np.linspace(0, 5, 200)
ThetaSolution = odeint(dTheta_dt, [np.pi/10, 0], t)
ThetaDraw = ThetaSolution[:,0]
T=2*np.pi*np.sqrt(l/g);print("T=%2.2f s"%T)
```

```
T=3.39 s
```

In [11]:
```python
plt.xlabel("t [s]")
plt.ylabel("Theta [rad]")
plt.title("Pendulum simulation")
plt.plot(t,ThetaDraw);
```



▸ See example

Table of Contents

Foucalt pendulum

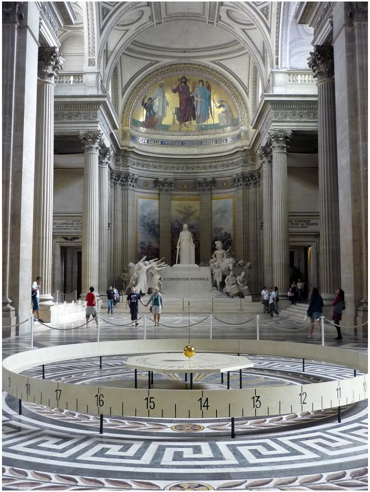


Figure: [Wik20b]

Foucault pendulum - dynamic equations

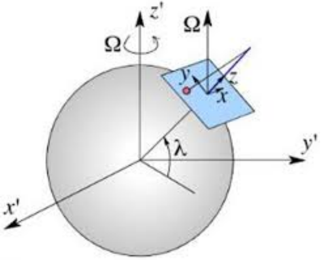


Figure: Foucault pendulum - setup

Coriolis force:

$$F_{c,x} = 2m\Omega \frac{dy}{dt} \sin\varphi$$
$$F_{c,y} = -2m\Omega \frac{dx}{dt} \sin\varphi$$

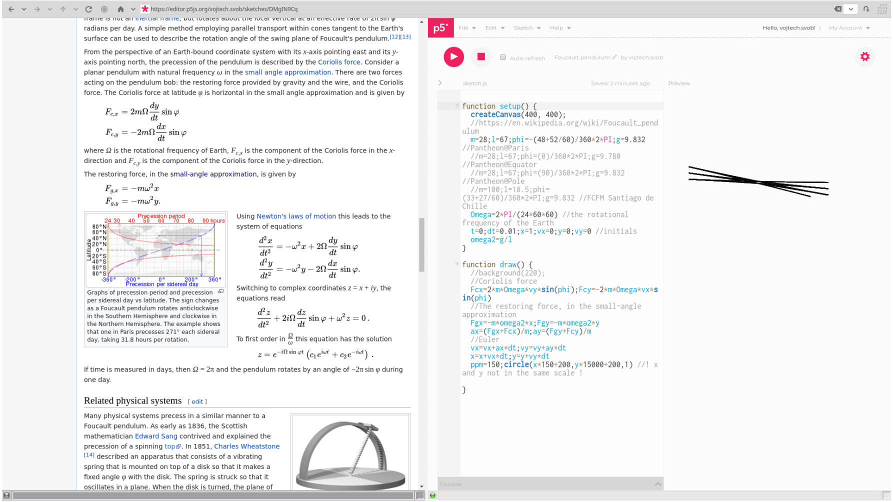
Restoring force (small angle approximation):

$$F_{g,x} = -m\omega^2 x$$
$$F_{g,y} = -m\omega^2 y.$$

Then dynamic equations:

$$\frac{d^2x}{dt^2} = -\omega^2 x + 2\Omega \frac{dy}{dt} \sin\varphi$$
$$\frac{d^2y}{dt^2} = -\omega^2 y - 2\Omega \frac{dx}{dt} \sin\varphi.$$

Sreenshot: Foucault pendulum @ processing



See example

Table of Contents

Sreenshot: Satellite motion @ processing

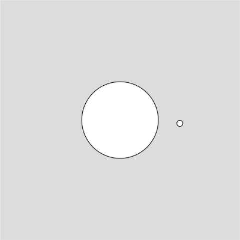
p5* File Edit Sketch Help

Auto-refresh Satellite motion by vojtech.svob

sketch.js Saved: 25 seconds ago

```
function setup() {
  createCanvas(400, 400);
  kappa=6.672E-11
  m=1
  M=5.972E24
  //Initial conditions
  t=0;dt=100;x=10E6;y=0;vx=0;vy=7.5E3
}

function draw() {
  background(220);
  r=sqrt(sq(x)+sq(y))
  Fg=kappa*m*M/(sq(r))
  Fx=-Fg*(x/r);Fy=-Fg*(y/r)
  ax=Fx/m;ay=Fy/m
  vx=vx+ax*dt;vy=vy+ay*dt
  x=x+vx*dt;y=y+vy*dt
  t=t+dt

  mpp=100000
  circle(200,200,2*6.378E6/mpp)
  if (r>6.378E6) {circle(200+x/mpp,200+y/mpp,10)}
  //circle(200+x/mpp,200+y/mpp,10)

}
```

Preview



See example

Table of Contents

1. *Introduction*
   - Motivation
   - Vojtech Svoboda @ CTU
   - Euler method
2. *1D problem in cartesian coordinates: free fall*
   - Spreadsheet
   - Processing
   - Python
3. *1D problem in rotational system: pendulum*
   - Basic analysis (spreadsheet & processing & octave)
   - Pendulum with friction (spreadsheet & processing)
   - Pendulum - phase space (spreadsheet)
   - Pendulum - energy conservation (spreadsheet)
   - Pendulum - small angle approximation analysis (spreadsheet)
   - Two pendulums (processing)
4. *Numerical simulation versus experiment*
   - Prague
   - World pendulum
5. *Final remarks*

To be continued..

Thank you

for your attention

Tom Henderson.
The physics classroom: Pendulum motion.
https://www.physicsclassroom.com/class/waves/Lesson-0/Pendulum-Motion, 2020.

Hok Kong (Wilfred) Lee.
Conservation of energy.
http://dept.swccd.edu/hlee/content/phys-170/lecture-web-07/, 2020.
[Online; accessed 14-March-2020].

Mike Stubna and Wendy McCullough.
Euler's method tutorial.
https://sites.esm.psu.edu/courses/emch12/IntDyn/course-docs/Euler-tutorial/.

Tamás Dr. Szabó.
Mechatronical Modelling.
2014.

Wikipedia contributors.
Euler method — Wikipedia, the free encyclopedia.
https://en.wikipedia.org/w/index.php?title=Euler_method&oldid=942478767, 2020.
[Online; accessed 9-March-2020].

Wikipedia contributors.
Foucault pendulum — Wikipedia, the free encyclopedia.
https://en.wikipedia.org/w/index.php?title=Foucault_pendulum&oldid=934467185, 2020.
[Online; accessed 14-March-2020].

Wikipedia contributors.
Pendulum (mathematics) — Wikipedia, the free encyclopedia.
https://en.wikipedia.org/w/index.php?title=Pendulum_(mathematics)&oldid=942104313, 2020.
[Online; accessed 1-March-2020].

Wikipedia contributors.

Projectile motion — Wikipedia, the free encyclopedia.
https://en.wikipedia.org/w/index.php?title=Projectile_motion&oldid=941891568, 2020.
[Online; accessed 3-March-2020].

Wikipedia contributors.
Runge–kutta methods — Wikipedia, the free encyclopedia.
https://en.wikipedia.org/w/index.php?title=Runge%E2%80%93Kutta_methods&oldid=944202380, 2020.
[Online; accessed 14-March-2020].