

¿Cómo calculamos
movimientos con
la compu?
(Integración numérica)

1) Un poco de programación

Práctica Computacional

Página de la Materia

Tutoriales para programar en Google Colab

1. Tutorial de Google Colab (plataforma de programación en Python)
2. Tutorial de Programación en Python



Códigos para simular movimientos

1. Tutorial Integrador
2. Péndulo
3. 2 Masas unidas por un resorte
4. Problema de Kepler



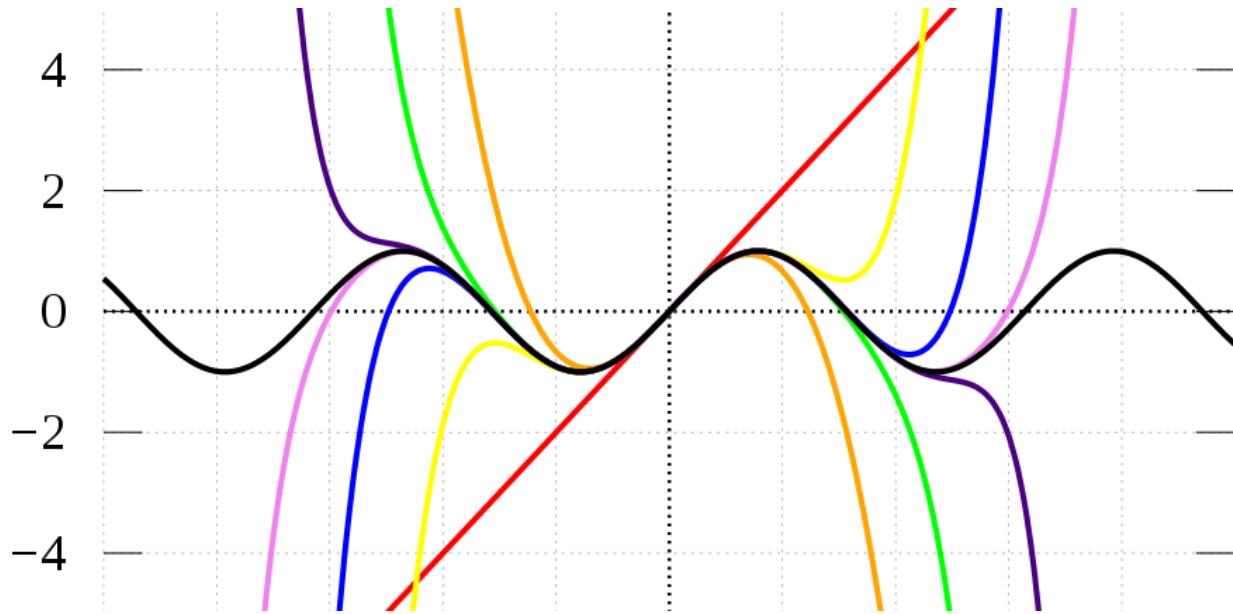
*Tutoriales para
ir aprendiendo
a programar*

Guía de ejercicios

Presentación

2) Un poco de matemática

Teorema de Taylor: puedo aproximar $f(x)$ alrededor de un x_0 mediante un polinomio



$$r(t + dt) = r(t) + \dot{r}(t)dt + \frac{\ddot{r}(t)}{2!} dt^2 + \frac{\dddot{r}(t)}{3!} dt^3 + \dots$$

$$r(t + dt) \approx r(t) + \dot{r}(t)dt + \frac{\ddot{r}(t)}{2} dt^2 + \frac{\ddot{r}(t)}{3!} dt^3$$
$$r(t - dt) \approx r(t) - \dot{r}(t)dt + \frac{\ddot{r}(t)}{2} dt^2 - \frac{\ddot{r}(t)}{3!} dt^3$$

} +

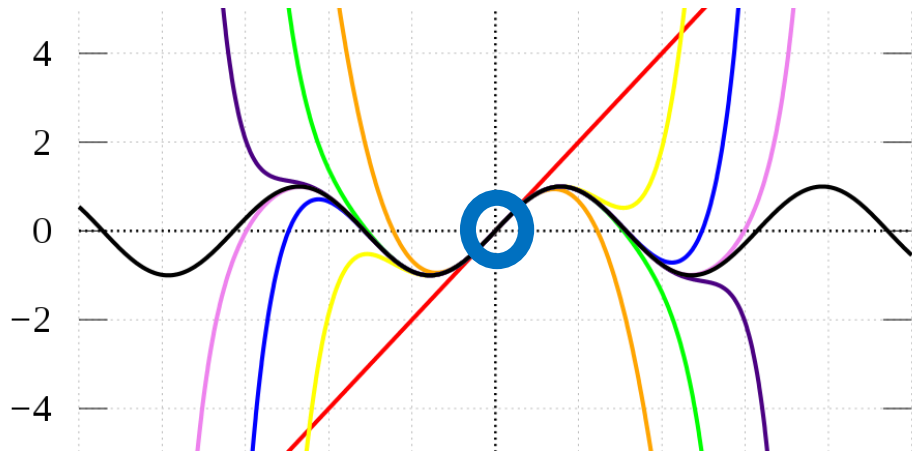
$$r(t + dt) + r(t - dt) \approx 2r(t) + \ddot{r}(t)dt^2$$

$$r(t + dt) \approx 2r(t) - r(t - dt) + \ddot{r}(t)dt^2$$

$$r(t + dt) \approx 2r(t) - r(t - dt) + \frac{\sum F(r(t), t)}{m} dt^2$$

3) Y que se haga la física

$$r(t + dt) \approx 2r(t) - r(t - dt) + \frac{\sum F(r(t), t)}{m} dt^2$$

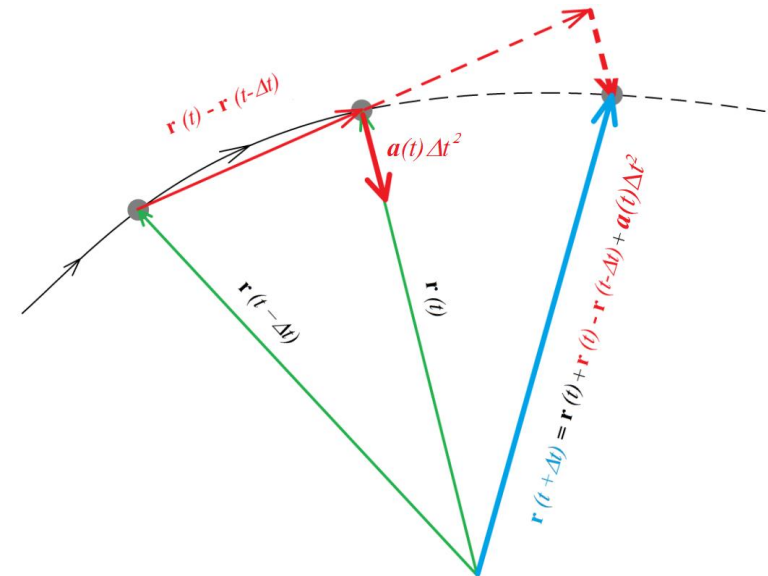


Algoritmo de Verlet:

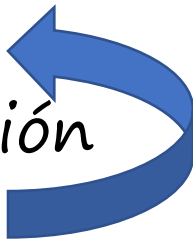
- 1) Elegimos una posición
- 2) Calculamos la fuerza y la aceleración
- 3) Calculamos la posición siguiente



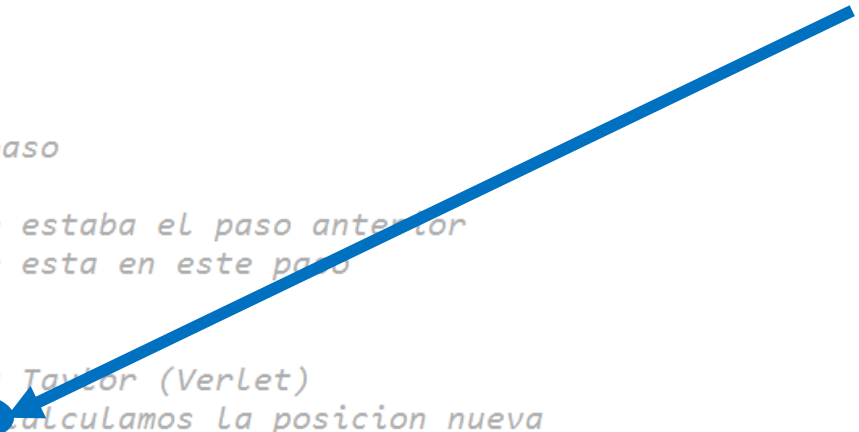
La idea es tomar un dt pequeño donde valga la aproximación de Taylor y además que $F(r(t), t) = cte$



Algoritmo de Verlet:

- 1) Elegimos una posición
 - 2) Calculamos la fuerza y la aceleración
 - 3) Calculamos la posición siguiente
- 

```
1 # importamos las librerías necesarias
2 import matplotlib.pyplot as plt
3
4 tiempos=[] # definimos lista para los tiempos
5 posiciones=[] # definimos lista para las posiciones
6
7 dt=0.1 # definimos el paso del tiempo
8 numero_pasos=100 # definimos la cantidad total de pasos
9
10 t0=0 # definimos el tiempo inicial
11 x0=1 # definimos la posición inicial
12 v0=0 # definimos la velocidad inicial
13
14 t1=t0+dt # definimos el tiempo siguiente para Verlet
15 x1=x0+v0*dt # definimos la posición siguiente para Verlet
16
17 k=1 # definimos la constante elástica del resorte
18 m=1 # definimos la masa
19
20 tiempos.append(t0)
21 tiempos.append(t1)
22 posiciones.append(x0)
23 posiciones.append(x1)
24
25 t=t1
26 for i in range(1, numero_pasos - 1):
27     t=t+dt # avanzamos el tiempo en cada paso
28
29     x_antes=posiciones[-2] # tomamos donde estaba el paso anterior
30     x_ahora=posiciones[-1] # tomamos donde está en este paso
31     a=-k*x_ahora/m
32
33     # esta es la predicción justita que usa Taylor (Verlet)
34     x_despues=2*x_ahora-x_antes+a*dt**2 # calculamos la posición nueva
35
36     tiempos.append(t) # guardamos el tiempo nuevo en la lista de tiempos
37     posiciones.append(x_despues) # guardamos la posición nueva en la lista
```

$$r(t + dt) \approx 2r(t) - r(t - dt) + \frac{\sum F(r(t), t)}{m} dt^2$$


¿Qué hacemos?

Hay 3 programas:

1) 2 masas y un resorte

2) Péndulo (barra)

3) Kepler (órbita de la tierra)

Jueguen

Resuelvan la guía

Resuelvan la guía

¿Qué hacemos?

Hay 3 programas:

1) 2 masas y un resorte

Jueguen

2) Péndulo (barra)

Resuelvan la guía

3) Kepler (órbita de la tierra)

Resuelvan la guía

A jugar