MDM2 Group 09

03 February 2025 15:03

metronome.py

```python
if __name__ == "__main__":
    x = np.array([0])
    mnlist = list()
    lines  = list()
    for i in range(mncount):
        mnlist.append(Metronome(i, randint(0, 42), mnlist))
    plt.ion()
    fig = plt.figure()
    plt.show()
    ax = fig.add_subplot(111)
    for mn in mnlist:
        mn.line, = ax.plot(x, mn.y, label="Metronome {}".format(mn.id))
    ax.legend()
    fig.canvas.draw()
        Dave McCulloch, 5 years ago • Initial Commit …
    for i in np.arange(0+inc, xlim, inc):
        x = np.append(x, [i])
        for mn in mnlist:
            mn.updateoffset()
            mn.calcyval(i)
            mn.line.set_xdata(x)
            mn.line.set_ydata(mn.y)
        for mn in mnlist:
            mn.oldoffset = mn.offset
        i += inc
        ax.relim()
        ax.autoscale_view(True, True, True)
        fig.canvas.draw()
        plt.pause(0.001)
```

how many metronomes on cart

initial offset

→ Every metronome has a list of all metronomes before it

For every metronome

Every metronome calculates its own y-value as sin (0 + offset)

For every x-value

In this case, x represents time rather than physical cart location

→ Add new x-value

→ Old offset + multiplier × $\sum$ sin (offset (i) – old offset)

↳ sum of all metronomes before it

→ Update offset data

For 3 metronomes:

Met. 1 : For every timestep, calculate new offset as old + multiplier × sin (0)

→ offset stays same

→ All metronomes sync to this one

Met. 2: For every timestep, calculate new offset as old + multiplier × sin (offset met. 1 - old)

→ should reduce offset from met 1 slowly as old → offset met. 1, offset (met 2) → offset (met. 1)

Met. 3: For every timestep, calculate new offset as old + multiplier × (sin (offset met. 1 – old) + sin (offset met. 2 – old))

Met. 1: $\theta_{1,new} = \theta_{1,old}$

Met. 2: $\theta_{2,new} = \theta_{2,old} + k\sin(\theta_{1,old} - \theta_{2,old})$

Met. n: $\theta_{n,new} = \theta_{n,old} + k\sum_{i=1}^{n} \theta_{i,old} - \theta_{n,old}$

metronomes.py

variables: mass of each bob, m_n

length of each pendulum, l_n

mass of cart, M

gravity, g

stiffness of cart, M_k

damping of cart, M_d

escapement damping coefficient, a_{damp}

escapement angle coefficient, a_{esc}

Functions:

$$\ddot{x} = \frac{\left(\frac{m_1}{M}\right) g\sin\theta_1 + \left(\frac{m_1}{M}\right) l_1 \dot{\theta}_1^2 \sin\theta_1 + \left(\frac{m_2}{M}\right) g\sin\theta_2 + \left(\frac{m_2}{M}\right) l_2 \dot{\theta}_2 \sin\theta_2}{1 - \left(\frac{m_1}{M}\right)\cos\theta_1 - \left(\frac{m_2}{M}\right)\cos\theta_2}$$

$$\ddot{\theta}_1 = -\left(\frac{g}{l_1}\right)\sin\theta_1 - \frac{\ddot{x}}{l_1}\cos\theta_1 - a_{damp} \times \theta_1\left(\left(\frac{\theta_1}{a_{esc}}\right)^2 - 1\right)$$

$$\ddot{\theta}_2 = -\left(\frac{g}{l_2}\right)\sin\theta_2 - \frac{\ddot{x}}{l_2}\cos\theta_2 - a_{damp} \times \theta_2\left(\left(\frac{\theta_2}{a_{esc}}\right)^2 - 1\right)$$

integrated w.r.t time by odeint.

initial: $\theta_1 =$ uniform(-30,30) rads

$\theta_2 =$ uniform (-30,30) rads

$\dot{\theta}_1 =$ uniform (-360,360) rads

$\dot{\theta}_2 =$ uniform (-360,360) rads

$x = 0$

$\dot{x} = 0$