$ Supplement Note 17: Further Topics on Numerical Integration

$1 Simple Pendulum

The simple plane pendulum (Fig. 1) consists of a point mass m at the end of a weightless rod of length l which swings back and forth in a vertical plane. The movement of the angel θ is governed by the tangential component of the gravitation force mg from *Newton's Second Law of Motion*:

$$\frac{d}{dt}\left(ml\,\frac{d\theta}{dt}\right) = -mg\sin\theta\ . \tag{1}$$

This leads to a *second order differential equation* for the angle θ

$$\frac{d^2\theta}{dt^2} = -\,\frac{g}{l}\ \sin\theta = -\,\omega_0^2\ \sin\theta \tag{2}$$

where

$$\omega_0 \equiv \sqrt{\frac{g}{l}} \tag{3}$$

For oscillations with small angles, $|\theta| \ll 1$, the sine function may be approximated by keeping only the leading linear term in the series expansion:

$$\sin\theta = \theta - \frac{\theta^3}{3!} + \cdots \approx \theta\ . \tag{4}$$

Eq.(2) is reduced to an equation for a *simple harmonic oscillator*

$$\frac{d^2\theta}{dt^2} = -\,\omega_0^2\,\theta \tag{5}$$

with general solutions given by

$$\theta(t) = a\cos\omega_0 t + b\sin\omega_0 t = \theta_0\ \cos(\omega_0 t - \alpha)\ ,\ a, b, \theta_0, \alpha \text{ constants}\ . \tag{6}$$

In this limit, the period of a simple pendulum T is equal to that of a simple harmonic oscillator T_0 and is given by

$$\lim_{\theta_0\to 0} T = T_0 \equiv \frac{2\pi}{\omega_0} = 2\pi\sqrt{\frac{l}{g}}\ . \tag{7}$$

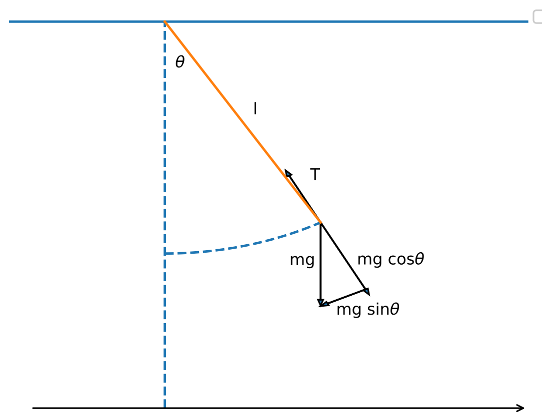

Figure 1: Simple plane pendulum.

$2 Numerical Solutions for the Equations of Simple Pendulums and and Simple Harmonic Oscillations

The first step to obtain numerical solutions for a second order differential equation

$$\frac{d^2\theta}{dt^2} = f(\frac{d\theta}{dt^2}, \theta, t) \tag{8}$$

is by introducing a new variable $\Omega \equiv \frac{d\theta}{dt}$ and rewriting it as a set of simultaneous first order differential equations of the variables θ, Ω:

$$\frac{d\Omega}{dt} \equiv \frac{d^2\theta}{dt^2} = f(\theta, \Omega, t) \tag{9a}$$

$$\frac{d\theta}{dt} = \Omega \tag{9b}$$

Numerical results for such *first order differential equations* may then be obtained by using the *Euler method*. It is found that the Euler method fails for oscillatory motions. As was discussed in **Supplement Note 8**, a slight modification, the *Euler-Cromer Method* can produce satisfactory results.

In practice, it is convenient to use the *scipy.integrate.odeint* sub-package for performing such jobs. In the codes listed in the *Pendulum.py* are examples using the sub-package *scipy.integrate.odeint* to solve both simple harmonic oscillator problem Eq.(5) and the pendulum problem Eq.(2) and the results ploted on the top parts and the lower parts of Figure 2 respectively. Using two initial conditions for the initial angles ($\theta(0) = 1.05$ and $\theta(0) = 1.05$), the simulated results for $\theta(t)$ and $\Omega(t)$ are plotted on the left side and on the right side of Figure 1 respectively. Some important observations can be obtained:

1. stable amplitueds are obtained for all the cases involved, in contrast to the results from using the original simple Euler method.

2. the amplitudes of the simple harmonic oscilator Eq.(5) (upper part of Figure 2) are independent of the initial angles θ_0.

3. the amplitueds the pendulum problem Eq.(2) (lower part of Figure 2) grows as the initial angle θ_0 increases.

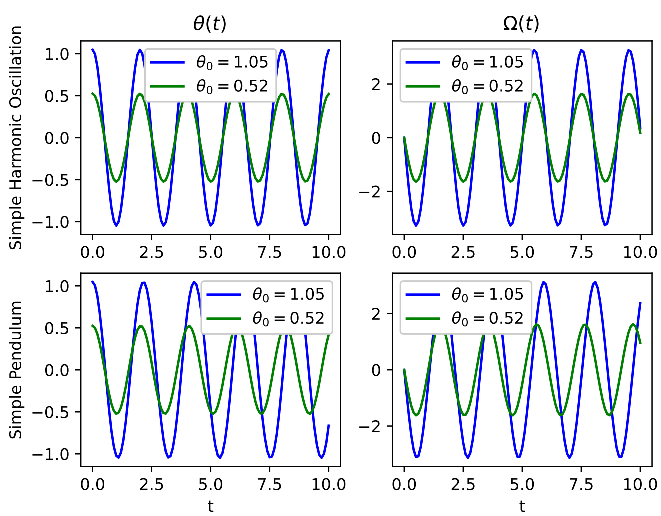

Figure 2: Simple Pendulum

```
# Pendulum.py
# simulation of Pendulum and Simple Harmonic Oscillation
```

```
import numpy as np
from scipy.integrate import odeint
import matplotlib.pyplot as plt
g, l, pi3= 9.8, 1, np.pi/3
def pend(y,t,c, index):
    theta, Omega = y
    if index == 1: dydt = [Omega, -c* theta] # simple harmonic oscillator
    else: dydt = [Omega, -c* np.sin(theta)]  # simple pendulum
    return dydt
tta, theta0 = [], pi3
for i in range(2):
    tta.append(theta0)
    theta0 /=2
c = g/l
t = np.linspace(0,10,101)
yy1,yy2,zz1,zz2 = [],[],[],[]
for i in range(2):
    theta0 = tta[i]
    y0= [theta0, 0]
    sol= odeint(pend,y0,t,args=(c,1))   # simple harmonic oscillator
    y1,y2 =sol[:,0],sol[:,1]
    yy1.append(y1)
    yy2.append(y2)
    sol2 = odeint(pend,y0,t,args=(c,2)) # simple pendulum
    z1,z2 =sol2[:,0],sol2[:,1]
    zz1.append(z1)
    zz2.append(z2)
gr=['b','g']
plt.figure()
plt.subplot(2,2,1)
plt.title(r'$\theta(t)$')
plt.ylabel('Simple Harmonic Oscillation')
for i,cc in enumerate(gr):
```

```
    theta0 = tta[i]
    plt.plot(t,yy1[i],cc,label=r'$\theta_0={:4.2f}$'.format(theta0))
plt.legend(loc='best')
plt.subplot(2,2,2)
plt.title(r'$\Omega(t)$')
for i,cc in enumerate(gr):
    theta0 = tta[i]
    plt.plot(t,yy2[i],cc,label=r'$\theta_0={:4.2f}$'.format(theta0))
plt.legend(loc='best')
plt.legend()
plt.subplot(2,2,3)
plt.ylabel('Simple Pendulum')
plt.legend(loc='best')
for i,cc in enumerate(gr):
    theta0 = tta[i]
    plt.plot(t,zz1[i],cc,label=r'$\theta_0={:4.2f}$'.format(theta0))
plt.legend(loc='best')
plt.xlabel('t')
plt.subplot(2,2,4)
#for i in range(2):
for i,cc in enumerate(gr):
    theta0 = tta[i]
    plt.plot(t,zz2[i],cc,label=r'$\theta_0={:4.2f}$'.format(theta0))
plt.xlabel('t')
plt.legend(loc='best')
fig = plt.gcf()
fig.savefig('SimplePendulum.eps', format='eps')
plt.show()
```