Pendolo quadrifilare

Sommario

L'esperienza verte sullo studio del moto di un pendolo e della dipendenza del periodo dall'ampiezza dell'oscillazione.

Materiale a disposizione

- Un pendolo quadrifilare;
- metro a nastro (risoluzione 1 mm);
- un computer per acquisizione ed analisi dei dati.

Definizioni

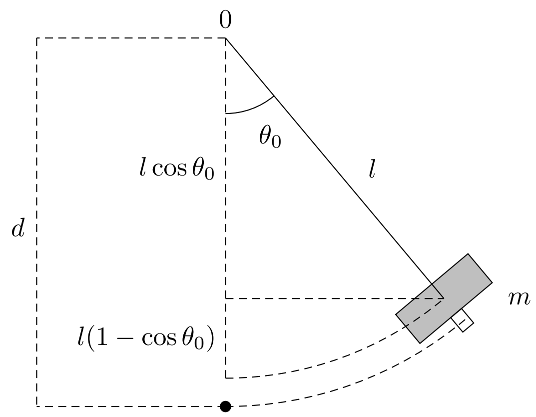

Figura 1: Schematizzazione dell'apparato sperimentale e definizioni di base.

Al di là delle definizioni ovvie, sottolineiamo la differenza tra la lunghezza l del pendolo e la distanza d tra il punto di sospensione ed il traguardo ottico che registra i passaggi della bandierina.

Ricordiamo inoltre che l'espressione per il periodo del pendolo si può sviluppare in serie come

$$T = 2\pi\sqrt{\frac{l}{g}}\left(1 + \frac{1}{16}\theta_0^2 + \frac{11}{3072}\theta_0^4 + \cdots\right). \qquad (1)$$

Misure da effettuare ed analisi

Nel momento in cui il pendolo si trova (con velocità nulla) nella posizione di massima altezza raggiunta dall'i-esima oscillazione la sua energia meccanica sarà soltanto energia potenziale. Al contrario, nel punto più basso della traiettoria (a patto di fissare opportunamente lo zero del potenziale), si avrà solamente energia cinetica.

Misure di velocità

Si costruisca il grafico della velocità v_0 nel punto più basso in funzione del tempo. Nel caso di smorzamento proporzionale alla velocità ci si aspetta un andamento esponenziale:

$$v_0(t) = v_0(0)e^{-\lambda t}. \qquad (2)$$

Si stimi, tramite un fit numerico o graficamente, la costante di smorzamento λ e il tempo di smorzamento associato $\tau = 1/\lambda$.

Per completezza, può essere interessante costruire il grafico cartesiano del periodo T in funzione del tempo. A causa della variazione dell'ampiezza di oscillazione, anch'esso diminuirà nel tempo.

Misure di smorzamento

L'ampiezza *(quasi) istantanea* di oscillazione si può ricavare, assumendo trascurabile la perdita di energia per attrito su un quarto di periodo, dal bilancio energetico

$$mgl(1-\cos\theta_0) = \frac{1}{2}mv_0^2, \qquad (3)$$

da cui si ricava banalmente

$$\theta_0 = \arccos\left(1 - \frac{v_0^2}{2gl}\right). \qquad (4)$$

Si costruisca il grafico cartesiano del periodo T in funzione dell'ampiezza θ_0 di oscillazione e si verifichi la (1). Quanti ordini dello sviluppo in serie si riescono a *vedere*?

Considerazioni pratiche

Il formato dei dati

Il programma di acquisizione registra il tempo di ogni transizione dello del traguardo ottico, e fornisce un *file* di uscita contenente tre colonne che rappresentano, rispettivamente:

1. un indice progressivo della transizione;
2. una flag del tipo di transizione: 0/1 se la bandierina entra/esce
3. il tempo assoluto della transizione. t_T del cartoncino nel traguardo ottico, mediato tra andata e ritorno.

Da questi dati si possono ricavare il periodo di oscillazione T, ed il tempo di transito t_T.

Detta w la larghezza della bandierina, la velocità del centro di massa del pendolo nel punto più basso dell'oscillazione si ottiene come

$$v_0 = \frac{w}{t_\mathrm{T}} \times \frac{l}{d} \qquad (5)$$

Un esempio di script per l'analisi dati di questa esperienza è mostrato nella pagina successiva.

Note sul programma di acquisizione

Una volta acceso il calcolatore, selezionare dal menù principale (in alto a sinistra) *Application* → *Education* → *plasduino*. Questo dovrebbe mostrare la finestra principale del programma di acquisizione. Per questa esperienza, tra la lista dei moduli, lanciate *Pendulum* (doppio click sulla linea corrispondente, oppure selezionate la linea stessa e premete *Open*).

Di norma al termine di ogni sessione di presa dati il programma vi chiede se volete salvare una copia del *file* dei dati in una cartella a vostra scelta (il che può essere comodo per l'analisi successiva). Se questa funzionalità dovesse essere disabilitata potete ri-abilitarla attraverso il menù di plasduino *Configuration* → *Change settings*: nella finestra che si apre selezionate il tab *daq* e abilitate l'opzione *prompt-save-dialog*.

```python
# Programma di esempio per l'analisi del file prodotto dal modulo pendulum  di plasduino
import numpy as np
import scipy
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
plt.ion() # interactive on

# Lettura del file di dati in ingresso (da modificare con il percorso al vostro file in ingresso).
ii, edge, tt = scipy.loadtxt('data/pendulum.txt', unpack = True)

# Calcolo dei tempi di transito Tr e del periodo T calcolato come la differenza tra
# il tempo di ingresso della bandierina e il tempo al passaggio successivo che completa 1 periodo
t  = []; T = []; Tr = [];
for i in range(3, len(tt)-1, 4):
    t.append(0.5*(tt[i]+tt[i-1]))
    Tr.append(tt[i] - tt[i-1])    # diff. tra uscita e ingresso della bandierina
    T.append((tt[i+1] - tt[i-3])) # si puo' calcolare in altri modi?
t  = np.array(t); T  = np.array(T); Tr = np.array(Tr)

# Definizione della geometria del pendolo [cm] (da modificare con le vostre misure).
w = 2.05; l = 113.5; d = 116.5
# Calcolo di velocita' e angolo
v = (w/Tr)*(l/d)
Theta = np.arccos(1. - (v**2)/(2*981.*l))

# Funzioni di fit per velocita' e angolo
def f_v(x, v0, tau):
    return v0*np.exp(-x/tau)

def f_Theta(x, p1, p2):
    return 2*np.pi*np.sqrt(l/981.)*(1 +p1*(x**2) +p2*(x**4))

# Fit di V vs t
popt_v, pcov_v = curve_fit(f_v, t, v, np.array([500., 100.]))
v0_fit,   tau_fit = popt_v
dv0_fit, dtau_fit = np.sqrt(pcov_v.diagonal())
print('v0  = %.2f +- %.2f cm/s' % (v0_fit,  dv0_fit))
print('tau = %.2f +- %.2f s' % (tau_fit, dtau_fit))
# Plot di V vs t
plt.figure(1)
plt.xlabel('Tempo [s]'); plt.ylabel('v [cm/s]'); plt.grid(color = 'gray')
plt.plot(t, v, '+', t, f_v(t, v0_fit, tau_fit))
plt.savefig('pendolo_VvsT.png')
# Plot di T vs Theta e dei residui rispetto alla funzione attesa
plt.figure(2)
plt.subplot(211)
plt.ylabel('Periodo [s]')
plt.plot(Theta, T, 'o', Theta, f_Theta(Theta, 1./16, 11./3072), 'r')
plt.subplot(212)
plt.xlabel('Angolo [rad]'); plt.ylabel('Periodo data - modello [ms]')
plt.plot(Theta, 1000*(T-f_Theta(Theta, 1./16, 11./3072)))
plt.savefig('pendolo_TvsTheta.png')
```