PELEG BAR SAPIR

PHYSICS SIMULATIONS (LECTURE NOTES)

1

Mechanics

1.1 *Preface*

Text text text

1.2 *Pendulum*

1.2.1 *Simple pendulum*

Text text text.

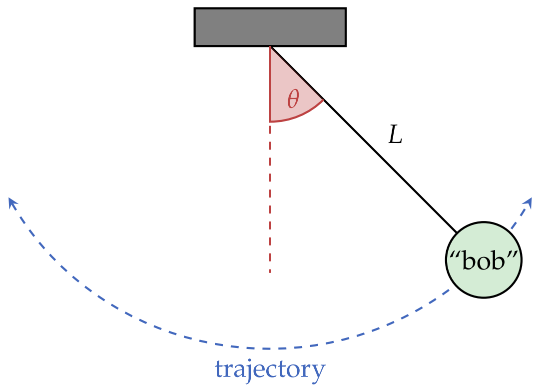

Figure 1.1: A simple pendulum. TBD: add more info.

Using force analysis we can derive an equation of motion for the bob (see Figure 1.2): since the rod can't change its length (it's always L), the only variable quantity is the angle θ, and the bob's trajectory is a circle. Any force acting in a radial direction to the trajectory must be counter-balanced (otherwise there will be some acceleration - and therefore motion - in that direction). We are therefore left with only a tangental force, with magnitude

$$F = -mg \sin(\theta), \tag{1.1}$$

the minus sign here is chosen to represent that gravity acts in the negative y direction (i.e. "down").

Applying Newton's second law we get that

$$F = ma = -mg\sin(\theta), \tag{1.2}$$

i.e.

$$a = -g\sin(\theta). \tag{1.3}$$

Now we see that the minus sign also makes sense physically, as it shows that the acceleration is always in the opposite direction to the angle (which is negative to the left and positive to the right).

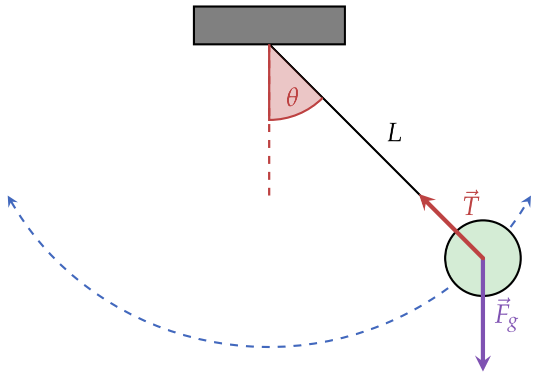

Figure 1.2: Forces acting on a simple pendulum. TBD: Force compnents.

The tangental position s of the bob can be calculated from the angle θ by

$$s = L\theta \tag{1.4}$$

(recall that θ is given in radians), and therefore the tangental velocity is

$$v = \dot{s} = L\dot{\theta}, \tag{1.5}$$

and the acceleration is therefore

$$a = \dot{v} = \ddot{s} = L\ddot{\theta}. \tag{1.6}$$

Since we know that $a = -g\sin(\theta)$, we get

$$L\ddot{\theta} = -g\sin(\theta), \tag{1.7}$$

and by moving the rhs term to the left and divide by l we get

$$\ddot{\theta} + \frac{g}{L}\sin(\theta) = 0. \tag{1.8}$$

This is a differential equation without analytical solution. We will therefore take two approaches: (1) use an approximation to yield an analytical solution, and (2) solve the equation numerically.

1.2.2 *Small-angle approximation*

The Taylor series expansion of $\sin(x)$ around $x = 0$ is

$$\sin(x) = x - \frac{1}{3!}x^3 + \frac{1}{5!}x^5 - \frac{1}{7!}x^7 + \ldots \tag{1.9}$$

We can therefore approximate $\sin(x)$ as x for small values of x:

$$\sin(x) \approx x. \tag{1.10}$$

This is known as the "small-angle approximation" of the sine function. By using this approximation we get that the (analytically) unsolvable Equation 1.8 reduces to

$$\ddot{\theta} + \frac{g}{L}\theta = 0, \tag{1.11}$$

for which we have an exact solution:

$$\theta(t) = A\cos\left(\omega t + \phi\right), \tag{1.12}$$

where $\omega = \sqrt{\frac{g}{L}}$. The parameters A and ϕ depend on the initial conditions (i.e. angle and tangental velocity).

TBW: resonance freq, how it looks like (+phase space), python plot?

1.2.3 *Numerical integratiob*

Another approach to solve Equation 1.8 is doing so numerically, i.e. essentially running a computer simulation. While computers can carry out many calculations per second (in the order of billions, in fact) - they are limited to performing descrete calculations. That is to say, a numerical calculation is also an approximation. However, unlike the small-angle approximation, in principle we can improve the approximation indefinitely, although in practice this is of course impossible.

Let us use a rather naive approach to numerically approximating Equation 1.8: instead of viewing θ as a continuous function of time t, we instead define t to only have equally spaced discrete values $t_0, t_1, t_2, \ldots$ such that

$$t_n = t_0 + n\Delta t, \tag{1.13}$$

or phrased differently: we look at the values t_i of t at intervals Δt starting from t_0 (see Figure 1.3).

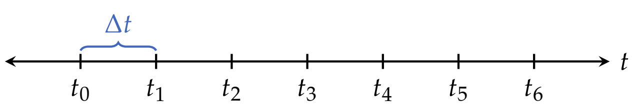

Figure 1.3: Discrete time steps.

In this discrete view, a function of t can also only be evaulated at discrete points. For example, the angle θ of the bob as a function of time is a discrete function taking the values $\theta_0, \theta_1, \theta_2, \ldots$, where each θ_i corresponds to the time t_i.

How does the discrete angular velocity look like in this scheme? It's worth looking at the definition of angular velocity in the continuous case:

$$\omega(t) = \dot{\theta}(t) = \frac{\mathrm{d}\theta(t)}{\mathrm{d}t} = \lim_{\Delta t \to 0} \frac{\theta\left(t + \Delta t\right) - \theta(t)}{\Delta t}. \tag{1.14}$$

To descritize, we can replace $\theta(t)$ with $\theta_{t_i} = \theta_i$, and therefore $\theta(t+\Delta t)$ with $\theta_{t_i+\Delta t} = \theta_{i+1}$. The limit $\lim\limits_{\Delta t \to 0}$ is pretty meaningless, since the smallest Δt possible in the discrete case is the Δt we chose to discretize out time steps. Therefore, we get that the discrete version of $\omega(t)$ is

$$\omega_i = \frac{\theta_{i+1} - \theta_i}{\Delta t}, \tag{1.15}$$

which is nothing more than saying that the angular velocity at time t_i is simply the difference between the angle at time t_{i+1} and the angle at time t_i, divided by the time difference Δt.

Equation 1.15 gives us a powerful tool: if we only know the current angle θ_i of the bob and its current velocity ω_i, then its next position θ_{i+1} is a simple rearrangement of the equation:

$$\theta_{i+1} = \theta_i + \omega_i \Delta t. \tag{1.16}$$

The same logic can be applied to the angular acceleration $\alpha(t)$: in the continuous version its given by

$$\alpha(t) = \dot{\omega(t)} = \frac{\mathrm{d}\omega(t)}{\mathrm{d}t} = \lim_{\Delta t \to 0} \frac{\omega(t+\Delta t) - \omega(t)}{\Delta t}, \tag{1.17}$$

and thus the discrete version is

$$\alpha_i = \frac{\omega_{i+1} - \omega_i}{\Delta t}, \tag{1.18}$$

and recovering the angular velocity ω_{i+1} from the angular velocity ω_i and the angular acceleration α_i is done by

$$\omega_{i+1} = \omega_i + \alpha_i \Delta t. \tag{1.19}$$

Since we use the inverse of the differentiation operation (in the discrete sense) to recover the quantity we're after, this scheme is known as a **numerical integration**. Specifically, this kind of numerical integration is called the **Forward-Euler** method, and it is generally considered unfavourable due to the fast rate with which it gains errors, and is rarely used in practice.

However, for the sake of simplicity of writing our first simulations, we will not discuss these issues now, nor will we generalize the method and present better ones - both of which we will do later in the course. Instead, for now we will continue with using this method to devise a numerical integration scheme for a simple pendulum.

In the case of the pendulum, recall that the acceleration at time t is $\ddot{\theta}(t) = -\frac{g}{L}\sin(\theta)$ (Equation 1.8), and therefore we can discretize it as

$$\alpha_{i+1} = -\frac{g}{L}\sin(\theta_i). \tag{1.20}$$

We then use the forward Euler method to get the angular velocity and angle, as seen in Equation 1.19 and Equation 1.16, respectively.

1.2.4 *Damped oscillation*

A slightly more realistic model of a pendulum also considers the way the pendulum loses energy over time (e.g. via friction). This can be modelled by adding a force which resists the angular velocity (see also Figure 1.4):

$$F_d = -bL\dot{\theta}, \tag{1.21}$$

where b is simply a parameter which adjusts how strong the damping is.

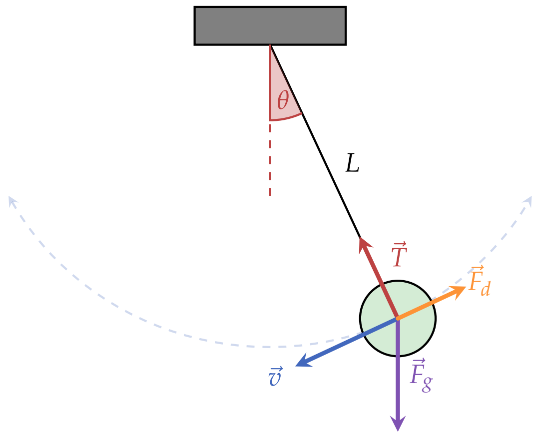

Figure 1.4: Forces on a pendulum including a damping force.

Using Newton's second law, recalling that $a = L\ddot{\theta}$ (Equation 1.6), we get

$$mL\ddot{\theta} = F_g + F_d = -mg\sin(\theta) - bL\dot{\theta}. \tag{1.22}$$

It's common to use $\beta = \frac{b}{2m}$ and of course $\omega_0 = \sqrt{\frac{g}{L}}$, which gives us

$$\ddot{\theta} = -2\beta\dot{\theta} - \omega_0^2\sin(\theta) = 0. \tag{1.23}$$

1.2.5 *Double pendulum*