

PELEG BAR SAPIR

# PHYSICS SIMULATIONS (LECTURE NOTES)



# 1

## Introduction

### 1.1 *Why are Simulations Used?*

Text

### 1.2 *Python*

Text

### 1.3 *A Bit About Git*

Text

### 1.4 *Some Mathematical Background*

Text

### 1.5 *Harmonic Oscillator*

#### 1.5.1 *Simple Harmonic Oscillator*

Many systems in physics present a simple, periodic (repeating) motion. One such system is a simple mass-less spring connected to a mass  $m$  and allowed to move in a single dimension only. If we ignore the effects of gravity, the only force acting on the mass arises from the spring itself: the more we pull or push the spring, the stronger it will resist to that change. This resistant force is given by

$$F = -kx, \tag{1.1}$$

where  $k$  is the **spring constant**, and  $x$  is the amount by which the spring contracts or expands relative to its rest position  $x_0$ . In Figure 1.1 we show three cases: when the mass is at the springs rest position,  $x_m = x_0$ , the spring applies no force on it. When the mass is displaced by a positive amount  $\Delta x > 0$ , the spring applied a *negative* force on it:  $F = -k\Delta x < 0$  (see note 1.1). And when the mass

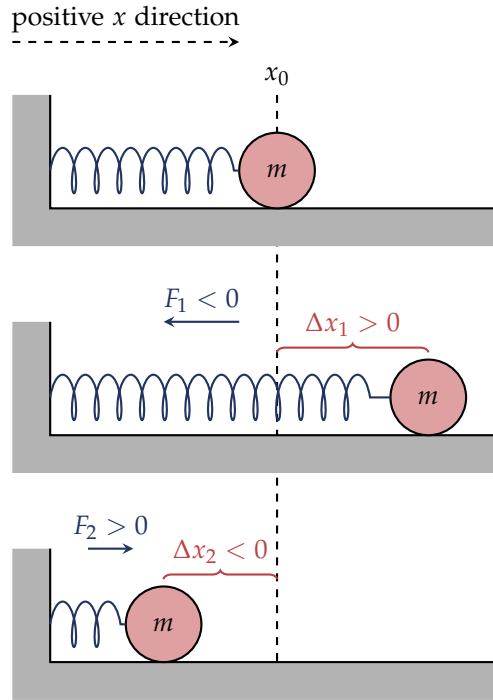
is displaced such that it contracts the spring,  $\Delta x < 0$  and thus the force applied by the spring is positive,  $F = k\Delta x > 0$ .

**Note 1.1 Negative and positive forces**

Recall that in this context, negative force means a force in the negative  $x$  direction.



Figure 1.1: A simple spring-mass system with spring constant  $k$  and a mass  $m$ . The top figure shows the spring at rest - i.e. when the mass is located at position  $x_0$  the spring applies no force on the mass (since  $\Delta x = x_m - x_0 = 0$ ). The middle figure shows the spring being at a *positive* displacement  $\Delta x_1 > 0$ , causing the spring to pull back with a negative force  $F_1 = -k\Delta x_1$ . The bottom picture shows the spring contracting by  $\Delta x_2 < 0$ , causing the spring to apply a positive force  $F_2 = -k\Delta x_2$  on the mass.



Using Newton's second law of motion (REF?) with  $x$  being the displacement from the rest position  $x_0$ , we get the relation

$$a = -kx, \quad (1.2)$$

but since  $a = \frac{dv}{dt} = \frac{d^2x}{dt^2}$ , we can re-write Equation 1.2 as

$$\frac{d^2x}{dt^2} = -kx, \quad (1.3)$$

or even more succinctly as

$$\ddot{x} = -kx. \quad (1.4)$$

Equation 1.4 is one of the simplest possible 2nd-order *ordinary* differential equations. Its solution is a combination of the two basic trigonometric equations:

$$x(t) = c_1 \sin(\alpha t) + c_2 \cos(\beta t), \quad (1.5)$$

where  $c_1$  and  $c_2$  are constants which we can find using the *starting conditions* (see note 1.2), and  $\alpha, \beta$  are parameters of the motion. Since function arguments must be unitless, these two parameters also cause the total quantity inside the trigonometric functions to

be unitless by having units corresponding to  $\left[\frac{1}{\text{time}}\right]$ . For example, if we measure the time in  $[\text{s}]$ , then the units of  $\alpha$  and  $\beta$  are  $[\text{s}^{-1}] = [\text{Hz}]$ .

**Note 1.2 Starting conditions for solving ODEs**

Recall that in order to completely solve an ordinary differential equation of order  $n$  we must have  $n$  starting conditions.



In the case where the initial position of the mass is  $x_0$  and the initial velocity is  $\dot{x}_0 = 0$  we get the simplified solution

$$x(t) = x_0 \cos(\omega t), \quad (1.6)$$

where  $\omega = \sqrt{\frac{k}{m}}$ . The plot of the position  $x(t)$  of the mass is given in Figure 1.2. Note how, when displayed side-by-side using the same time axes, the position plot is “lagging behind” the velocity plot by  $\phi = \frac{\pi}{2}$ : when we start the motion, the velocity is 0 and then starts to increase in the negative direction (i.e. the mass is moving to the left). The position is at its maximum at that time, and decreases from  $x_{\max}$  at  $t = 0$  to  $x = 0$  at  $t = \frac{\pi}{2}$ , while still staying positive. At  $t = \frac{\pi}{2}$  the velocity reaches its maximum negative value,  $\dot{x} = -v_{\max}$  (which is 1 in units of  $\frac{1}{v_{\max}}$ ), and the position becomes negative (since it is to the left of the rest position of the spring).

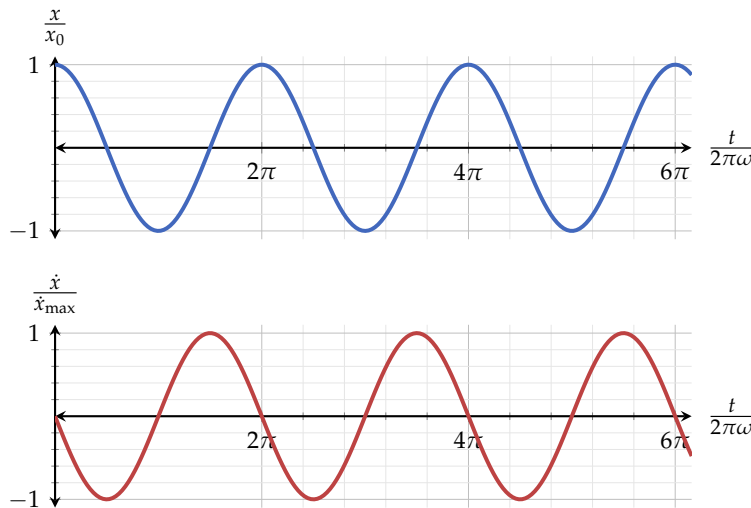


Figure 1.2: Simple harmonic oscillator, top (blue): position  $x$  vs. time  $t$ . The axes units are such that a full period of the oscillation takes  $\Delta t = 2\pi$ , and that the minimum and maximum values of the position are  $\pm 1$ , respectively. Bottom (red): velocity vs. time on the same time axes, and a velocity axis which is scaled such that  $v_{\max} = 1$ .

We can now plot the position of the mass vs. its momentum  $p = m\dot{x}$  (Figure 1.3). This is called a **phase space** plot. MORE TEXT ON THE PHASE SPACE.

### 1.5.2 Damped Harmonic Oscillator

We can make the harmonic oscillator model a bit more realistic if we add a damping force proportional to the velocity of the mass. This corresponds e.g. to introducing friction into the model. The form of the damping force is  $F_{\text{damp}}(t) = -c v(t) = -c \dot{x}$ , for some

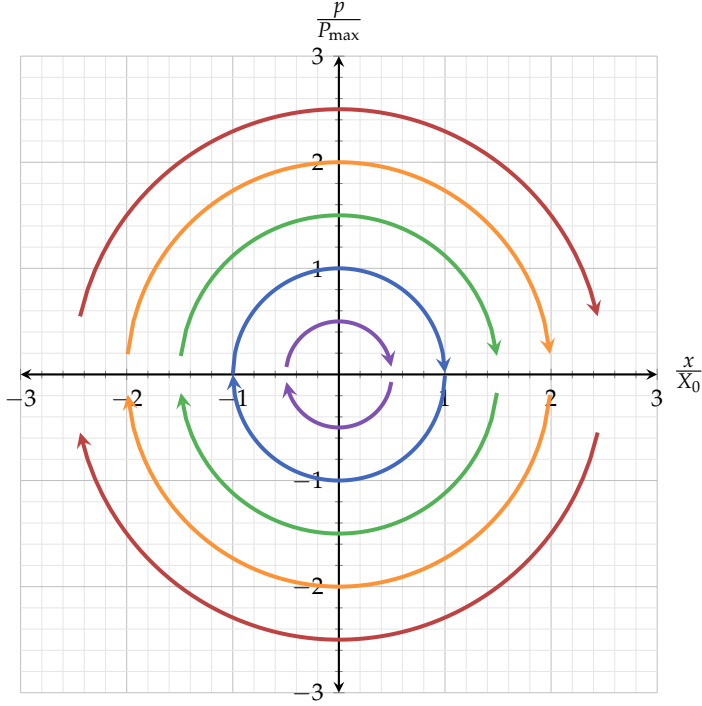


Figure 1.3: Phase space plot of simple harmonic oscillators with different momenta (either their masses are different, or their initial distance are different). The axes are scaled such that their units are the initial position  $X_0$  and maximum momentum  $P_0$ , respectively, of the second oscillator (trajectory drawn in blue).

real damping coefficient  $c \geq 0$  (in the case where  $c = 0$  we get back the simple harmonic oscillator model). Thus, the overall Newton 2nd law equation of the system has the following form:

$$m\ddot{x} = -c\dot{x} - kx. \quad (1.7)$$

The term  $-c\dot{x}$  always acts in the opposite direction to the velocity due to the minus sign. Re-arranging Equation 1.7 we get the more “canonical” form

$$m\ddot{x} + c\dot{x} + kx = 0, \quad (1.8)$$

We then use the solution  $x(t) = e^{\lambda t}$ , substituting it into Equation 1.8:

$$F_{\text{total}}(t) = \lambda^2 m e^{\lambda t} + c e^{\lambda t} + k e^{\lambda t} \quad (1.9)$$

$$= e^{\lambda t} (m\lambda^2 + c\lambda + k) \quad (1.10)$$

$$= 0. \quad (1.11)$$

Since Equation 1.11 is true for all  $t$  (and in any case  $e^{\lambda t} \neq 0$  for all  $t$ ), for the equation to be true it must be that the quadratic equation in  $\lambda$  equals zero. Using the quadratic formula we get that

$$\lambda_{1,2} = \frac{-c \pm \sqrt{c^2 - 4km}}{2m}. \quad (1.12)$$

For the solutions to  $\lambda$  to be real numbers,  $c^2 \geq 4km$  - otherwise the term in the square root is negative. When  $c^2 > 4km$  we call the system **overdamped**, and the case where  $c^2 = 4km$  is the **critical**

**damping** value. The overall position vs. time relationship of the damped system is given by

$$x(t) = \quad (1.13)$$

### 1.5.3 *Simulating Harmonic Oscillators in Python*

Text