1 OPTIONAL: Optimal Control Problem

Consider an astronaut flying a rocket ship in outer space. Treat the dynamic system as a point mass with states $x = [p_x, v_x, p_y, v_y]$ where $p_x$ is $x$-position, $v_x$ is $x$-velocity, $p_y$ is $y$-position, and $v_y$ is $y$-velocity. The system has control inputs $u = [t_x, t_y]$ where $t_x$ is thrust in $x$, and $t_y$ is thrust in $y$.

Assuming the mass of the rocket is 1 for simplicity, the equations of motion are:

$$\frac{d}{dt}\begin{pmatrix} p_x \\ v_x \\ p_y \\ v_y \end{pmatrix} = \begin{pmatrix} v_x \\ t_x \\ v_y \\ t_y \end{pmatrix}$$

or

$$\frac{d}{dt}\begin{pmatrix} p_x \\ v_x \\ p_y \\ v_y \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}\begin{pmatrix} p_x \\ v_x \\ p_y \\ v_y \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}\begin{pmatrix} t_x \\ t_y \end{pmatrix}$$

or

$$\frac{d}{dt}x = Ax + Bu$$

Let’s convert this to a discrete problem, with timestep $\Delta t = 0.1$. In general, forward Euler integration is a poor choice, but for simplicity we will use it here.

$$x_{k+1} = x_k + \frac{dx_k}{dt} \Delta t$$

$$x_{k+1} = x_k + (Ax_k + Bu_k)\Delta t$$

$$x_{k+1} = (I + A\Delta t)x_k + B\Delta t u_k$$

### 1.1 Objective function

The astronaut has to be at work at time $k = 61$, at space coordinates $[140, 0]$, but he has to do a few errands on the way. At time $k = 20$, he has to be at coordinates $[20, 10]$, and at time $k = 40$ he has to be at coordinates $[80, -30]$. 
The astronaut also wants to save fuel, which costs \( \sqrt{t_x^2 + t_y^2 + 1} - 1 \) fuel units per timestep. The objective is:

\[
f = \| (p_{x,20}, p_{y,20}) - (20, 10) \|_2^2 + \| (p_{x,40}, p_{y,40}) - (80, -30) \|_2^2 + \| (p_{x,61}, p_{y,61}) - (140, 0) \|_2^2 \\
+ \| (v_{x,61}, v_{y,61}) - (0, 0) \|_2^2 + 10^{-6} \sum_{k=1}^{60} (\sqrt{t_{x,k}^2 + t_{y,k}^2} - 1)
\]

The astronaut starts at \( x_1 = [0, 0, 0, 0] \), and must choose \([u_1, u_2...u_{60}]\) to minimize \( f \). This objective is smooth, and surprisingly it is also convex, so the minimizer you wrote in Exercise 3 should be able to find the global optimal solution.

Tasks:

1. Implement a function \( X = \text{integrate}\_\text{controls}(U) \) where \( U \) is the vector \( U = [t_{x,1}, t_{x,2}...t_{x,60}, t_{y,1}, t_{y,2}...t_{y,60}] \), and \( X \) is the matrix \([x_1, x_2...x_{60}, x_{61}]\).

2. Using your \text{integrate}\_\text{controls} function, implement a function \( f = \text{astro}\_\text{objective}(U) \) which returns the objective function.

3. Minimize this function using the optimizer you wrote. You may want to put a print statement inside your minimizer’s main loop, so that you can see the scheme converging (or not converging). I suggest \text{fprintf}(’%4d %.3g\n’,k,norm(J))

4. Plot the solution, and verify that it looks something like this: