Lab #2

1. *Creating a new function in CVX.* We consider the convex, monotone function f with values $f(x) = 2x + 3x^{1.2} + 4.1x^{2.3}$, and domain $\mathbf{R}_+$. Let g be the inverse function of f: for a given $y > 0$, $g(y)$ is the (unique) value of t such that $f(t) = y$. There is no closed form expression for g.

   (a) Show that, for a given $y \in \mathbf{R}$, $g(y)$ is the optimal value of the problem

   $$\max_t \ t \ : \ f(t) \leq y,$$

   with the standard convention that the optimal value is $-\infty$ if the problem is not feasible, that is, $y < 0$.

   (b) Show that g is concave, monotone increasing, and with domain $\mathbf{R}_+$. Write a CVX code that implements g.

   (c) Write a CVX code that solves the following problem:

   $$\min_{x,y} \ \frac{x^2}{y} + 4x + 5y \ : \ g(y) + 2g(y) \geq 2.$$

2. *Exploring nearly optimal points.* An optimization algorithm will find *an* optimal point for a problem, provided the problem is feasible. It is often useful to explore the set of nearly optimal points. When a problem has a 'strong minimum', the set of nearly optimal points is small; all such points are close to the original optimal point found. At the other extreme, a problem can have a 'soft minimum', which means that there are many points, some quite far from the original optimal point found, that are feasible and have nearly optimal objective value. In this problem you will use a typical method to explore the set of nearly optimal points.

   We start by finding the optimal value $p^\star$ of the given problem

   $$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p, \end{array}$$

   as well as an optimal point $x^\star \in \mathbf{R}^n$. We then pick a small positive number ϵ, and a vector $c \in \mathbf{R}^n$, and solve the problem

   $$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p \\ & f_0(x) \leq p^\star + \epsilon. \end{array}$$

Note that any feasible point for this problem is ϵ-suboptimal for the original problem. Solving this problem multiple times, with different c's, will generate (perhaps different) ϵ-suboptimal points. If the problem has a strong minimum, these points will all be close to each other; if the problem has a weak minimum, they can be quite different.

There are different strategies for choosing c in these experiments. The simplest is to choose the c's randomly; another method is to choose c to have the form $\pm e_i$, for $i = 1, \ldots, n$. (This method gives the 'range' of each component of x, over the ϵ-suboptimal set.)

You will carry out this method for the following problem, to determine whether it has a strong minimum or a weak minimum. You can generate the vectors c randomly, with enough samples for you to come to your conclusion. You can pick $\epsilon = 0.01p^\star$, which means that we are considering the set of 1% suboptimal points.

The problem is a minimum fuel optimal control problem for a vehicle moving in $\mathbf{R}^2$. The position at time kh is given by $p(k) \in \mathbf{R}^2$, and the velocity by $v(k) \in \mathbf{R}^2$, for $k = 1, \ldots, K$. Here $h > 0$ is the sampling period. These are related by the equations

$$p(k+1) = p(k) + hv(k), \quad v(k+1) = (1-\alpha)v(k) + (h/m)f(k), \quad k = 1, \ldots, K-1,$$

where $f(k) \in \mathbf{R}^2$ is the force applied to the vehicle at time kh, $m > 0$ is the vehicle mass, and $\alpha \in (0,1)$ models drag on the vehicle; in the absense of any other force, the vehicle velocity decreases by the factor $1-\alpha$ in each discretized time interval. (These formulas are approximations of more accurate formulas that involve matrix exponentials.)

The force comes from two thrusters, and from gravity:

$$f(k) = \begin{bmatrix} \cos\theta_1 \\ \sin\theta_1 \end{bmatrix} u_1(k) + \begin{bmatrix} \cos\theta_2 \\ \sin\theta_2 \end{bmatrix} u_2(k) + \begin{bmatrix} 0 \\ -mg \end{bmatrix}, \quad k = 1, \ldots, K-1.$$

Here $u_1(k) \in \mathbf{R}$ and $u_2(k) \in \mathbf{R}$ are the (nonnegative) thruster force magnitudes, θ_1 and θ_2 are the directions of the thrust forces, and $g = 10$ is the constant acceleration due to gravity.

The total fuel use is

$$F = \sum_{k=1}^{K-1} \left(u_1(k) + u_2(k)\right).$$

(Recall that $u_1(k) \geq 0$, $u_2(k) \geq 0$.)

The problem is to minimize fuel use subject to the initial condition $p(1) = 0$, $v(1) = 0$, and the way-point constraints

$$p(k_i) = w_i, \quad i = 1, \ldots, M.$$

(These state that at the time hk_i, the vehicle must pass through the location $w_i \in \mathbf{R}^2$.) In addition, we require that the vehicle should remain in a square operating region,

$$\|p(k)\|_\infty \leq P^{\max}, \quad k = 1, \ldots, K.$$

Both parts of this problem concern the specific problem instance with data given in `thrusters_data.m`.

(a) Find an optimal trajectory, and the associated minimum fuel use $p^\star$. Plot the trajectory $p(k)$ in $\mathbf{R}^2$ (*i.e.*, in the p_1, p_2 plane). Verify that it passes through the way-points.

(b) Generate several 1% suboptimal trajectories using the general method described above, and plot the associated trajectories in $\mathbf{R}^2$. Would you say this problem has a strong minimum, or a weak minimum?