Homework 6

August 13, 2017

1. **Polynomial trajectories between waypoints.** Christmas disaster strikes as we have found that Rudolph and his merry reindeer have oversleep and do not want to work anymore. It is up to you to save the holidays! You activate the *Naughty and Nice-Completely Automated Trajectory* machine (N.a.N.-C.A.T), a high level path planing system, that outputs a set of way-points $\{w_1, ..., w_n\}$, $w_j \in \mathbb{R}^2$ and corresponding target times $\{k_1, ..., k_n\}$. Your goal is to once again plan a path for your magical flying animals (llamas, obviously). Let $p(t) \in \mathbb{R}^2$ be the path that the llama will take, *i.e.*

$$p(t) = \begin{bmatrix} \text{pos}_x(t) \\ \text{pos}_y(t) \end{bmatrix}.$$

Our job is to design a path given a set of constraints. The path needs to go through all the way-points at the corresponding target times. In addition, we also need the position, velocity and acceleration of the llama to be continuous along the path $p(t)$. This is so the path can be *dynamically feasible.* In order to achieve this, we approximate the path $p(t)$ by a piecewise cubic polynomial function, where each cubic polynomial joins a set of 2 way-points. Mathematically we have:

$$p(t) = p_i(t) \quad \text{for} \quad k_i \le t \le k_{i+1}, \quad i = 1, 2, \cdots, (n-1),$$

where

$$p_i(t) = \begin{bmatrix} a_{ix} \\ a_{iy} \end{bmatrix} t^3 + \begin{bmatrix} b_{ix} \\ b_{iy} \end{bmatrix} t^2 + \begin{bmatrix} c_{ix} \\ c_{iy} \end{bmatrix} t + \begin{bmatrix} d_{ix} \\ d_{iy} \end{bmatrix}.$$

Let H_i be the ith set of 2 adjacent way-points, $i = 1, 2, \cdots, (n-1)$. Each set shares one way-point with the adjacent set, *i.e.*

$$H_1 = \{(w_1, k_1), (w_2, k_2)\} \quad H_2 = \{(w_2, k_2), (w_3, k_3)\}$$

and so on. To ensure the position $p(t)$, velocity $\dot{p}(t)$ and acceleration $\ddot{p}(t)$ of the llama are continuous, for each set H_i we should have:

$$p_i(k_j) = w_j \quad \forall (w_j, k_j) \in H_i.$$

We also need:

$$\dot{p}_i(k_s) = \dot{p}_{i+1}(k_s) \qquad\qquad \ddot{p}_i(k_s) = \ddot{p}_{i+1}(k_s),$$

where k_s is the shared way-point time between sets H_i and H_{i+1}.

Lastly, we also want to minimize the RMS jerk, or third derivative of the path the llama will take (sharp turns are hard on its legs):

$$J = \left(\frac{1}{k_n - k_1} \int_{k_1}^{k_n} \|\dddot{p}_x(t)\|^2 + \|\dddot{p}_y(t)\|^2 dt \right)^{1/2},$$

where triple dots denotes the third derivative. Finally we get to the problem. You need to find the polynomial coefficients for $p_i(t)$'s such that the cost function J is minimized and all the constraints are satisfied.

a) Cast this problem as an optimization problem with a cost function and constraints. Your cost function should **not** be in terms of any integrals. We advise you to develop a single matrix equation for the constraints.

b) Using the setup of part (a), implement your solution with the data in the file `cities.m` (If you want to know where these points came from see `cities.txt`.). Each column of `w` gives the $(\text{pos}_x, \text{pos}_y)$ position waypoint for the corresponding time in `k`. Report the following plots:

   - Each polynomial path (trajectory) overlayed on the waypoints
   - The velocity and acceleration over time (they should be continuous)

   Also report your final minimized RMS jerk value J.

2. Optimal control for maximum asymptotic growth. We consider the controllable linear system

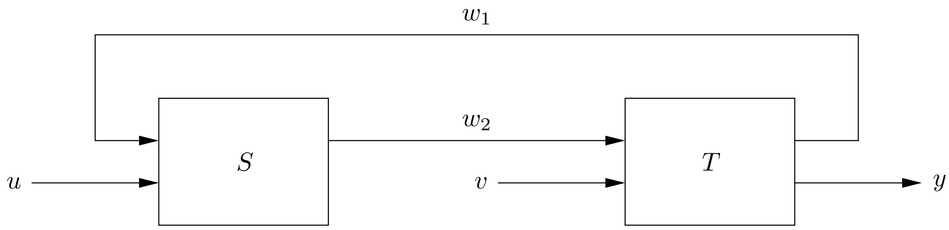
$$x(t+1) = Ax(t) + Bu(t), \qquad x(0) = 0,$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$. You can assume that A is diagonalizable, and that it has a single dominant eigenvalue (which here, means that there is one eigenvalue with largest magnitude). An input $u(0), \dots, u(T-1)$ is applied over time period $0, 1, \dots, T-1$; for $t \geq T$, we have $u(t) = 0$. The input is subject to a total energy constraint:

$$\|u(0)\|^2 + \cdots + \|u(T-1)\|^2 \leq 1.$$

The goal is to choose the inputs $u(0), \dots, u(T-1)$ that maximize the norm of the state for large t. To be more precise, we're searching for $u(0), \dots, u(T-1)$, that satisfies the total energy constraint, and, for any other input sequence $\tilde{u}(0), \dots, \tilde{u}(T-1)$ that satisfies the total energy constraint, satisfies $\|x(t)\| \geq \|\tilde{x}(t)\|$ for t large enough. Explain how to do this. You can use any of the ideas from the class, *e.g.*, eigenvector decomposition, SVD, pseudo-inverse, etc. Be sure to summarize your final description of how to solve the problem. Unless you have to, you should *not* use limits in your solution. For example you cannot explain how to make $\|x(t)\|$ as large as possible (for a specific value of t), and then say, "Take the limit as $t \to \infty$" or "Now take t to be really large".

3. **Interconnection of linear systems.** Often a linear system is described in terms of a block diagram showing the interconnections between components or subsystems, which are themselves linear systems. In this problem you consider the specific interconnection shown below:



Here, there are two subsystems S and T. Subsystem S is characterized by

$$\dot{x} = Ax + B_1 u + B_2 w_1, \qquad w_2 = Cx + D_1 u + D_2 w_1,$$

and subsystem T is characterized by

$$\dot{z} = Fz + G_1 v + G_2 w_2, \qquad w_1 = H_1 z, \qquad y = H_2 z + J w_2.$$

We don't specify the dimensions of the signals (which can be vectors) or matrices here. You can assume all the matrices are the correct (*i.e.*, compatible) dimensions. Note that the subscripts in the matrices above, as in B_1 and B_2, refer to different matrices. Now the problem. Express the overall system as a single linear dynamical system with input, state, and output given by

$$\begin{bmatrix} u \\ v \end{bmatrix}, \qquad \begin{bmatrix} x \\ z \end{bmatrix}, \qquad y,$$

respectively. Be sure to explicitly give the input, dynamics, output, and feedthrough matrices of the overall system. If you need to make any assumptions about the rank or invertibility of any matrix you encounter in your derivations, go ahead. But be sure to let us know what assumptions you are making.

4. **Worst and best direction of excitation for a suspension system.** A suspension system is connected at one end to a base (that can move or vibrate) and at the other to the load (that it is supposed to isolate from vibration of the base). Suitably discretized, the system is described by

$$x(t+1) = Ax(t) + Bu(t), \quad y(t) = Cx(t), \quad x(0) = 0,$$

where $u(t) \in \mathbb{R}^3$ represents the (x-, y-, and z- coordinates of the) displacement of base, and $y(t) \in \mathbb{R}^3$ represents the (x-, y-, and z- coordinates of the) displacement of the load. The input u has the form $u(t) = qv(t)$, where $q \in \mathbb{R}^3$ is a (constant) vector with $\|q\| = 1$, and $v(t) \in \mathbb{R}$ gives the displacement amplitude versus time. In other words, the driving displacement u is always in the direction q, with amplitude given by the (scalar) signal v. The response of the system is judged by the RMS deviation of the load over a 100 sample interval, *i.e.*,

$$D = \left(\frac{1}{100} \sum_{t=1}^{100} \|y(t)\|^2 \right)^{1/2}.$$

The data A, B, C, $v(0), \ldots, v(99)$ are known (and available in the JSON file `worst_susp_data.json` on the course web site). The problem is to find the direction $q_{\max} \in \mathbb{R}^3$ that maximizes D, and the direction $q_{\min} \in \mathbb{R}^3$ that minimizes D. Give the directions and the associated values of D ($D_{\max}$ and $D_{\min}$, respectively).

5. **Some optimization problems.** Solve the following optimization problems (find the optimal $t \in \mathbb{R}$). Your solution must be in terms of given matrices, and as concise as possible. Assume that the given matrices are with appropriate size, including the identity matrix I.

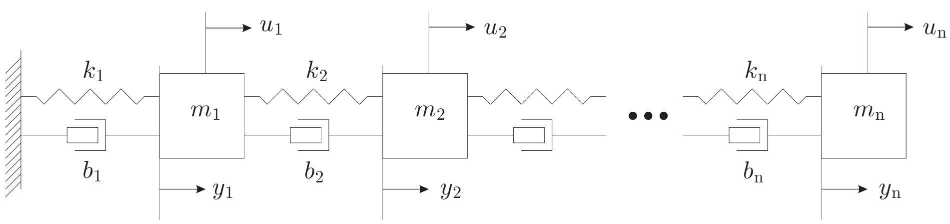
   a) Given symmetric X, find
   $$\min \{t \mid tI - X \geq 0\}$$

   b) Given Y, find
   $$\min \left\{t \mid tI - Y^T Y \geq 0\right\}$$

   c) Given symmetric W and symmetric positive definite Z, find
   $$\min \{t \mid tZ - W \geq 0\}$$

   d) Given A and symmetric positive definite P, find
   $$\max \left\{t \mid A^T P + PA + 2tP \leq 0\right\}$$

6. **An analog signal generator..** We have 7 masses and springs arranged in the chain below.



The masses have mass 1, the spring constants are all 0.8, and the damping constants are all 0.32.

We'd like to use it to generate a sawtooth function for t on $[0, 10]$,

$$y_{\text{saw}}(t) = \begin{cases} \frac{t}{2} & \text{if } t \leq 2 \\ 1 - \frac{t-2}{2} & \text{if } 2 < t \leq 4 \\ 0 & \text{otherwise} \end{cases}$$

In particular, we want the position of the first mass to follow y_{saw} as closely as possible.

The equations of motion for the system are

$$M\ddot{q}(t) + D\dot{q}(t) + Kq(t) = f(t)$$

where $q(t)$ is the vector of displacements at time t and $f(t)$ is the force applied.

   a) Using sample period $h = 0.1$, construct a discretized linear dynamical system in state-space form for this system. When doing this, let $u(k)$ be the constant force applied during the time interval $[kh, (k+1)h)$ and let $y(k)$ be $q_1(kh)$, the position of the first mass sampled at $t = kh$.

b) Our first plan is to displace the masses from equilibrium, and then let them go without applying any external force to the masses, *i.e.*, $u = 0$. By picking the initial displacements appropriately, we'd like the position of the first mass to follow y_{saw} as closely as possible. Find the initial displacements which minimize the sum of the squares of the error

$$J_{\text{out}} = \sum_{k=0}^{100} |q_1(kh) - y_{\text{saw}}(kh)|^2$$

with the initial velocities zero. Plot the desired trajectory and the trajectory you achieve for $q_1(t)$ on the same plot. Note that for this problem you can allow the masses to pass through each other.

c) Our second plan is to apply external forces to the masses so that the position of the first mass follows the trajectory y_{saw} as closely as possible. The initial positions and velocities of the masses are all zeros, *i.e.*, $q(0) = 0$ and $\dot{q}(0) = 0$. In order to do this, we define $J = J_{\text{out}} + \mu J_{\text{in}}$, where

$$J_{\text{in}} = \sum_{k=0}^{100} \|u(k)\|^2.$$

Using `logspace(-3,-1,10)` in Matlab, let μ have 10 different values. Pick the best value of μ among 10 values to find the minimum norm input force sequence u_{opt} minimizing J_{in} such that J_{out} is less than or equal to the achieved value of J_{out} in part (b). Plot the desired trajectory y_{saw} and the trajectory you achieve for $q_1(t)$ using the minimum norm input force sequence u_{opt} that you found. Note that for this problem you can allow the masses to pass through each other.