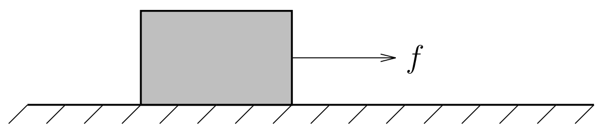
UNIVERSITY OF CALIFORNIA, SANTA CRUZ
BOARD OF STUDIES IN COMPUTER ENGINEERING

CMPE 240: INTRODUCTION TO LINEAR DYNAMICAL SYSTEMS

PROBLEM SET 2

---

1. *Mass/force example.* Find the matrix A for the mass/force example in the lecture notes.



   - unit mass, zero position/velocity at $t = 0$, subject to force $f(t)$ for $0 \leq t \leq n$
   - $f(t) = x_j$ for $j - 1 \leq t < j$, $j = 1, \ldots, n$
     (x is the sequence of applied forces, constant in each interval)
   - y_1, y_2 are final position and velocity

   For $n = 4$, find a specific input force sequence x that moves the mass to final position 1 and final velocity zero.

2. *Undirected graph.* Consider an undirected graph with n nodes, and no self loops (*i.e.*, all branches connect two different nodes). Let $A \in \mathbf{R}^{n \times n}$ be the *node adjacency matrix*, defined as

$$A_{ij} = \begin{cases} 1 & \text{if there is a branch from node } i \text{ to node } j \\ 0 & \text{if there is no branch from node } i \text{ to node } j \end{cases}$$

   Note that $A = A^T$, and $A_{ii} = 0$ since there are no self loops. We can intrepret A_{ij} (which is either zero or one) as the number of branches that connect node i to node j.

   Let $B = A^k$, where $k \in \mathbf{Z}$, $k \geq 1$. Give a simple interpretation of B_{ij} in terms of the original graph. (You might need to use the concept of a *path* of length m from node p to node q.)

3. *Counting sequences in a language or code.* We consider a language or code with an alphabet of n symbols $1, 2, \ldots, n$. A sentence is a finite sequence of symbols, $k_1, \ldots, k_m$ where $k_i \in \{1, \ldots, n\}$. A language or code consists of a set of sequences, which we will call the *allowable sequences.*

   A language is called *Markov* if the allowed sequences can be described by giving the allowable transitions between consecutive symbols. For each symbol we give a set of symbols which are allowed to follow the symbol. As a simple example, consider a Markov language with three symbols $1, 2, 3$. Symbol 1 can be followed by 1 or 3; symbol 2 must be followed by 3; and symbol 3 can be followed by 1 or 2. The sentence 1132313 is allowable (*i.e.*, in the language); the sentence 1132312 is not allowable (*i.e.*, not in the language).

To describe the allowed symbol transitions we can define a matrix $A \in \mathbf{R}^{n \times n}$ by

$$A_{ij} = \begin{cases} 1 & \text{if symbol } i \text{ is allowed to follow symbol } j \\ 0 & \text{if symbol } i \text{ is not allowed to follow symbol } j \end{cases}.$$

(a) Let $B = A^k$. Give an interpretation of B_{ij} in terms of the language.

(b) Consider the Markov language with five symbols $1, 2, 3, 4, 5$, and the following transition rules:

- 1 must be followed by 2 or 3
- 2 must be followed by 2 or 5
- 3 must be followed by 1
- 4 must be followed by 4 or 2 or 5
- 5 must be followed by 1 or 3

Find the total number of allowed sentences of length 10. Compare this number to the simple code that consists of all sequences from the alphabet (*i.e.*, all symbol transitions are allowed).

In addition to giving the answer, you must explain how you solve the problem. Do not hesitate to use Matlab.

4. *Price elasticity of demand.* The demand for n different goods as a function of their prices is described by a function $f(\cdot)$ from $\mathbf{R}^n$ to $\mathbf{R}^n$:

$$q = f(p),$$

where p is the price vector, and q is the demand vector. Linear models of the demand function are often used to analyze the effects of small price changes. Denoting the current price and current demand vectors by p^* and q^*, we have that $q^* = f(p^*)$, and the linear approximation is:

$$q^* + \delta q \approx f(p^*) + \left.\frac{df}{dp}\right|_{p^*} \delta p.$$

This is usually rewritten in term of the *elasticity matrix* E, with entries

$$e_{ij} = \left.\frac{df_i}{dp_j}\right|_{p_j^*} \frac{1/q_i^*}{1/p_j^*}$$

(*i.e.*, relative change in demand per relative change in price.) Define the vector y of relative demand changes, and the vector x of relative price changes,

$$y_i = \frac{\delta q_i}{q_i^*}, \quad x_j = \frac{\delta p_j}{p_j^*},$$

and, finally, we have the linear model $y = Ex$. Here are the questions:

(a) What is a reasonable assumption about the diagonal elements e_{ii} of the elasticity matrix?

(b) Consider two goods. The off-diagonal elements of E describe how the demand for one good is affected by changes in the price of the other good. Assume $e_{11} = e_{22} = -1$ and $e_{12} = e_{21}$, that is,

$$E = \begin{bmatrix} -1 & e_{12} \\ e_{12} & -1 \end{bmatrix}.$$

Two goods are called *substitutes* if they provide a similar service or other satisfaction (for example: train tickets and bus tickets, cake and pie, etc.) Two goods are called *complements* if they tend to be used together (for example: automobiles and gasoline, left and right shoes, etc.) For each of these two generic situations, what can you say about e_{12}?

(c) Suppose the price elasticity of demand matrix is

$$E = \begin{bmatrix} -1 & -1 \\ -1 & -1 \end{bmatrix}.$$

Describe the *nullspace* of E, and give an interpretation (in one or two sentences.) What kind of goods could have such an elasticity matrix?

5. *Color perception.* Human color perception is based on the responses of three different types of color light receptors, called *cones.* The three types of cones have different spectral response characteristics and are called L, M, and, S because they respond mainly to long, medium, and short wavelengths, respectively. In this problem we will divide the visible spectrum into 20 bands, and model the cones' response as follows:

$$L_{\text{cone}} = \sum_{i=1}^{20} l_i p_i, \qquad M_{\text{cone}} = \sum_{i=1}^{20} m_i p_i, \qquad S_{\text{cone}} = \sum_{i=1}^{20} s_i p_i,$$

where p_i is the incident power in the ith wavelength band, and l_i, m_i and s_i are nonnegative constants that describe the spectral response of the different cones. The perceived color is a complex function of the three cone responses, *i.e.*, the vector $(L_{\text{cone}}, M_{\text{cone}}, S_{\text{cone}})$, with different cone response vectors perceived as different colors. (Actual color perception is a bit more complicated than this, but the basic idea is right.)

(a) *Metamers.* When are two light spectra, p and $\tilde{p}$, visually indistinguishable? (Visually identical lights with different spectral power compositions are called *metamers.*)

(b) *Visual color matching.* In a color matching problem, an observer is shown a test light and is asked to change the intensities of three primary lights until the sum of the primary lights looks like the test light. In other words, the observer is asked the find a spectrum of the form

$$p_{\text{match}} = a_1 u + a_2 v + a_3 w,$$

where $u,\ v,\ w$ are the spectra of the primary lights, and a_i are the (nonnegative) intensities to be found, that is visually indistinguishable from a given test light spectrum p_{test}. Can this always be done? Discuss briefly.

(c) *Visual matching with phosphors.* A computer monitor has three phosphors, R, G, and B. It is desired to adjust the phosphor intensities to create a color that looks like a reference test light. Find weights that achieve the match or explain why no such weights exist.

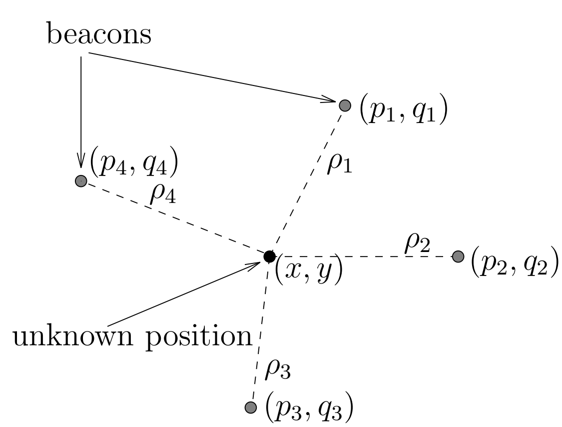
The data for this problem is available from the class web page, by following the link to Matlab datasets, in an m-file named `color_perception.m`. Copy this file to your working directory and type `color_perception` from within Matlab. This will define and plot the vectors `wavelength`, `B_phosphor`, `G_phosphor`, `R_phosphor`, `L_coefficients`, `M_coefficients`, `S_coefficients`, and `test_light`.

(d) *Effects of illumination.* An object's surface can be characterized by its reflectance (*i.e.*, the fraction of light it reflects) for each band of wavelengths. If the object is illuminated with a light spectrum characterized by I_i, and the reflectance of the object is r_i (which is between 0 and 1), then the reflected light spectrum is given by $I_i r_i$, where $i = 1, \ldots, 20$ denotes the wavelength band.

Now consider two objects illuminated (at different times) by two different light sources, say an incandescent bulb and sunlight. Sally argues that if the two objects look identical when illuminated by a tungsten bulb, they will look identical when illuminated by sunlight.

Beth disagrees: she says that two objects can appear identical when illuminated by a tungsten bulb, but look different when lit by sunlight. Who is right? If Sally is right, explain why. If Beth is right give an example of two objects that appear identical under one light source and different under another. You can use the vectors `sunlight` and `tungsten` defined in `color_perception.m` as the light sources.

6. Consider the linearized navigation equations on from the lecture notes.



- linearize around (x_0, y_0): $\delta\rho \approx A \begin{bmatrix} \delta x \\ \delta y \end{bmatrix}$, where

$$a_{i1} = \frac{(x_0 - p_i)}{\sqrt{(x_0 - p_i)^2 + (y_0 - q_i)^2}}, \quad a_{i2} = \frac{(y_0 - q_i)}{\sqrt{(x_0 - p_i)^2 + (y_0 - q_i)^2}}$$

Find the conditions under which A has full rank. Describe the conditions geometrically (*i.e.*, in terms of the relative positions of the unknown coordinates and the beacons).

7. *Proof of Cauchy-Schwarz inequality.* You will prove the Cauchy-Schwarz inequality.

   (a) Suppose $a \geq 0$, $c \geq 0$, and for all $\lambda \in \mathbf{R}$, $a + 2b\lambda + c\lambda^2 \geq 0$. Show that $|b| \leq \sqrt{ac}$.

   (b) Given v, $w \in \mathbf{R}^n$ explain why $(v + \lambda w)^T(v + \lambda w) \geq 0$ for all $\lambda \in \mathbf{R}$.

   (c) Apply (a) to the quadratic resulting when the expression in (b) is expanded, to get the Cauchy-Schwarz inequality:

   $$|v^T w| \leq \sqrt{v^T v}\sqrt{w^T w}.$$

   (d) When does equality hold?

8. *Vector spaces over the Boolean field.* In this course the *scalar field*, *i.e.*, the components of vectors, will usually be the real numbers, and sometimes the complex numbers. It is also possible to consider vector spaces over other fields, for example $\mathbf{Z}_2$, which consists of the two numbers 0 and 1, with Boolean addition and multiplication (*i.e.*, $1+1 = 0$). Unlike $\mathbf{R}$ or $\mathbf{C}$, the field $\mathbf{Z}_2$ is finite, indeed, has only two elements.

   A vector in $\mathbf{Z}_2^n$ is called a *Boolean vector*. Much of the linear algebra for $\mathbf{R}^n$ and $\mathbf{C}^n$ carries over to $\mathbf{Z}_2^n$. For example, we define a function $f : \mathbf{Z}_2^n \to \mathbf{Z}_2^m$ to be linear (over $\mathbf{Z}_2$) if $f(x+y) = f(x)+f(y)$ and $f(\alpha x) = \alpha f(x)$ for every x, $y \in \mathbf{Z}_2^n$ and $\alpha \in \mathbf{Z}_2$. It is easy to show that every linear function can be expressed as matrix multiplication, *i.e.*, $f(x) = Ax$, where $A \in \mathbf{Z}_2^{m\times n}$ is a Boolean matrix, and all the operations in the matrix multiplication are Boolean, *i.e.*, in $\mathbf{Z}_2$. Concepts like nullspace, range, independence and rank are all defined in the obvious way for vector spaces over $\mathbf{Z}_2$.

   Although we won't consider them in this course, there are many important applications of vector spaces and linear dynamical systems over $\mathbf{Z}_2$. In this problem you will explore one simple example: block codes.

   Linear block codes. Suppose $x \in \mathbf{Z}_2^n$ is a Boolean vector we wish to transmit over an unreliable channel. In a linear block code, the vector $y = Gx$ is formed, where $G \in \mathbf{Z}_2^{m\times n}$ is the *coding matrix*, and $m > n$. Note that the vector y is 'redundant'; roughly speaking we have *coded* an n-bit vector as a (larger) m-bit vector. This is called an (n, m) code. The coded vector y is transmitted over the channel; the received signal $\hat{y}$ is given by

   $$\hat{y} = y + v,$$

   where v is a noise vector (which usually is zero). This means that when $v_i = 0$, the ith bit is transmitted correctly; when $v_i = 1$, the ith bit is changed during transmission.

In a *linear decoder*, the received signal is multiplied by another matrix: $\hat{x} = H\hat{y}$, where $H \in \mathbf{Z}_2^{n \times m}$. One reasonable requirement is that if the transmission is perfect, *i.e.*, $v = 0$, then the decoding is perfect, *i.e.*, $\hat{x} = x$. This holds if and only if H is a left inverse of G, *i.e.*, $HG = I_n$, which we assume to be the case.

(a) What is the practical significance of $\mathcal{R}(G)$?

(b) What is the practical significance of $\mathcal{N}(H)$?

(c) A one-bit *error correcting code* has the property that for any noise v with one component equal to one, we still have $\hat{x} = x$. Consider $n = 3$. Either design a one-bit error correcting linear block code with the smallest possible m, or explain why it cannot be done. (By design we mean, give G and H explicitly and verify that they have the required properties.)

Remark: linear decoders are never used in practice; there are far better nonlinear ones.