Final Exam Solutions

This is a 15 hour take-home final with 4 problems. Please submit your solutions to Gradescope at most 15 after you receive the exam.

- You may use any books, notes, or computer programs (e.g., MATLAB), but you may not discuss the exam with others until Dec. 14, after everyone has taken the exam and the solutions are posted. The only exception is that you can ask the course staff for clarification, by emailing to the staff email address ee263-fall1718-staff@lists.stanford.edu. We have tried pretty hard to make the exam unambiguous and clear, so we’re unlikely to say much. **Please do not post any exam related questions on Piazza.**

- Since you have 15 hours, we expect your solutions to be legible, neat, and clear. Do not submit your rough notes, and please try to simplify your solutions as much as you can. We will deduct points from solutions that are technically correct, but much more complicated than they need to be.

- Please check your email a few times during the exam, just in case we need to send out a clarification or other announcements. It’s unlikely we will need to do this, but you never know.

- If a problem asks for some specific answers, make sure they are obvious in your solutions. You might put a box around the answers, so they stand out from the surrounding discussion, justification, plots, etc.

- When a problem involves some computation (say, using MATLAB), we do not want just the final answers. We want a clear discussion and justification of exactly what you did, the well-commented MATLAB source code that produces the result, and the final numerical result or plots. Be sure to show us your verification that your computed solution satisfies whatever properties it is supposed to, at least up to numerical precision. For example, if you compute a vector $x$ that is supposed to satisfy $Ax = b$ (say), show us the matlab code that checks this, and the result. (This might be done by the matlab code \texttt{norm(A*x-b)}; be sure to show us the result, which should be very small.) **We will not check your numerical solutions for you, in cases where there is more than one solution.**

- In the portion of your solutions where you explain the mathematical approach, you cannot refer to MATLAB operators, such as the backslash operator. (You can, of course, refer to inverses of matrices, or any other standard mathematical construct.)
• Some of the problems are described in a practical setting, such as heat diffusion or Markov chains. You do not need to understand anything about the application area to solve these problems. We have taken special care to make sure all the information and math needed to solve the problem is given in the problem description.

• Some of the problems require you to download and run a matlab file to generate the data needed. These files can be found at the URL

   http://ee263.stanford.edu/exams/ilove263finals/final17data.zip

• Please respect the honor code. Although we encourage you to work on homework assignments in small groups, you cannot discuss the midterm with anyone, with the exception of EE263 course staff, until Dec. 14, when everyone has taken it and the solutions are posted online.
1. **Stereo-vision calibration [15 points].** A stereo-vision system consists of two cameras that view a three-dimensional scene from slightly different positions. A small object located at a position in three-dimensional space is projected onto each camera’s two-dimensional image plane. The position of the object on the image plane of the first camera is \( p \in \mathbb{R}^2 \), and the position of the object on the image plane of the second camera is \( q \in \mathbb{R}^2 \). An analysis of the geometry of ideal camera imaging reveals that \( p \) and \( q \) satisfy the equation:
\[
\begin{bmatrix}
  p^T \\
  1
\end{bmatrix} F 
\begin{bmatrix}
  q^T \\
  1
\end{bmatrix} = 0,
\]
where \( F \in \mathbb{R}^{3 \times 3} \) is a nonzero matrix called the fundamental matrix associated with the stereo-vision system. We can multiply \( F \) by any nonzero constant, and the equation above will still hold. Therefore, we normalize \( F \) by assuming that \( \sum_{i=1}^{3} \sum_{j=1}^{3} F_{ij}^2 = 1 \), and \( F_{11} > 0 \). (We ignore the possibility that \( F_{11} = 0 \).) The fundamental matrix \( F \) can be determined through careful analysis of the positions, orientations, and optical properties of the two cameras. An alternate approach is calibration, which is used in this problem.

During calibration both cameras view \( K \) labeled objects. For the \( k \)th object, we record the positions \( p_k \) and \( q_k \) of the object in the image planes of the two cameras. We then estimate \( F \) from the calibration data by finding the matrix that minimizes the mean squared residual:
\[
J = \frac{1}{K} \sum_{k=1}^{K} \left( \begin{bmatrix}
  p_k^T \\
  1
\end{bmatrix} F 
\begin{bmatrix}
  q_k^T \\
  1
\end{bmatrix} \right)^2,
\]
subject to the normalization constraints \( \sum_{i=1}^{3} \sum_{j=1}^{3} F_{ij}^2 = 1 \), and \( F_{11} > 0 \). (If the calibration measurements were exact, and the camera optics had no distortion or imperfections, then we would get \( J = 0 \) for the true fundamental matrix.)

a) Given the calibration data \( p_k \) and \( q_k \) for \( k = 1, \ldots, K \), explain how to find the matrix \( F_{ls} \) that minimizes \( J \) subject to the normalization constraints. State any assumptions that are needed for your method to work.

b) Apply your method to the data given in `stereovision_calibration_data.m`. The calibration data are given in the \( 2 \times K \) matrices \( P \) and \( Q \), where \( p_k \) and \( q_k \) are the \( k \)th columns of \( P \) and \( Q \), respectively. Report your estimate \( F_{ls} \), and the corresponding value of \( J \).

c) Now suppose we are given \( \tilde{p}_k \) and \( \tilde{q}_k \) for \( k = 1, \ldots, N \), where \( \tilde{p}_k \) and \( \tilde{q}_{\sigma(k)} \) are the positions of the \( k \)th object in the image planes of the two cameras, and \( \sigma(1), \ldots, \sigma(N) \) is a permutation of \( 1, \ldots, N \). The correspondence problem is to guess the permutation \( \sigma(1), \ldots, \sigma(N) \) given the fundamental matrix \( F \), and the image data \( \tilde{p}_k \) and \( \tilde{q}_k \) for \( k = 1, \ldots, N \). Give a simple heuristic for the correspondence problem. (Your method will not be infallible, but it will tend to work well.
if the vision system is close to being ideal, and $N$ is not too large.) Apply your method to the data given in `stereovision_calibration_data.m` using the estimate of the fundamental matrix that you found above. Report the permutation that you find.

**Solution.**

a) **[7 points]** Define the vectors $x_k, y_k \in \mathbb{R}^3$ such that

$$x_k = \begin{bmatrix} p_k \\ 1 \end{bmatrix} \quad \text{and} \quad y_k = \begin{bmatrix} q_k \\ 1 \end{bmatrix}.$$  

We can write the mean squared residual as

$$J = \frac{1}{K} \sum_{k=1}^{K} (x_k^T F y_k)^2 = \frac{1}{K} \left\| \begin{bmatrix} x_1^T F y_1 \\ \vdots \\ x_K^T F y_K \end{bmatrix} \right\|^2 = \frac{1}{K} \left\| \begin{bmatrix} x_1^T \left[ F_{*1} \ F_{*2} \ F_{*3} \right] y_1 \\ \vdots \\ x_K^T \left[ F_{*1} \ F_{*2} \ F_{*3} \right] y_K \end{bmatrix} \right\|^2 = \frac{1}{K} \left\| \begin{bmatrix} (y_1)_1 x_1^T F_{*1} + (y_1)_2 x_1^T F_{*2} + (y_1)_3 x_1^T F_{*3} \\ \vdots \\ (y_K)_1 x_K^T F_{*1} + (y_K)_2 x_K^T F_{*2} + (y_K)_3 x_K^T F_{*3} \end{bmatrix} \right\|^2 = \frac{1}{K} \left\| \begin{bmatrix} (y_1)_1 x_1^T F_{*1} \\ \vdots \\ (y_K)_1 x_K^T F_{*1} \end{bmatrix} \right\| = \| M f \|^2,$$

where we define

$$f = \begin{bmatrix} F_{*1} \\ F_{*2} \\ F_{*3} \end{bmatrix} \in \mathbb{R}^9 \quad \text{and} \quad M = \frac{1}{\sqrt{K}} \begin{bmatrix} (y_1)_1 x_1^T \\ \vdots \\ (y_K)_1 x_K^T \end{bmatrix} \in \mathbb{R}^{K \times 9}.$$  

We can write the normalization constraints as

$$\sum_{i=1}^{3} \sum_{j=1}^{3} F_{ij}^2 = \sum_{j=1}^{3} \| F_{*j} \|^2 = \left\| \begin{bmatrix} F_{*1} \\ F_{*2} \\ F_{*3} \end{bmatrix} \right\|^2 = \| f \|^2 = 1,$$

4
and \( f_1 > 0 \). Thus, we want to solve the following optimization:

\[
\begin{align*}
\text{minimize} & \quad \|Mf\| \\
\text{subject to} & \quad \|f\| = 1 \quad f_1 > 0.
\end{align*}
\]

We can solve this problem as follows. Let \( v_9 \) be the last right singular vector of \( M \), and let

\[
f = \text{sign}((v_9)_1)v_9.
\]

Assuming \((v_9)_1 \neq 0\) (which is equivalent to the given assumption that \( F_{11} \neq 0 \)), this solves the optimization problem above. Our estimate of the fundamental matrix \( F \) is obtained by stacking the entries of the vector \( f \) into a matrix:

\[
F = \begin{bmatrix}
f_1 & f_4 & f_7 \\
f_2 & f_5 & f_8 \\
f_3 & f_6 & f_9
\end{bmatrix}.
\]

The corresponding value of \( J \) is \( \sigma_9 \), the last singular value of \( M \).

b) [4 points] We find that

\[
F = \begin{bmatrix}
0.5847 & -0.3432 & 0.1567 \\
-0.3357 & 0.2357 & -0.3821 \\
0.1518 & 0.3861 & -0.1715
\end{bmatrix},
\]

and \( J = 2.73e-7 \).

c) [4 points] For \( k = 1, \ldots, N \), define

\[
\tilde{x}_k = \begin{bmatrix} \hat{p}_k \\ 1 \end{bmatrix} \quad \text{and} \quad \tilde{y}_k = \begin{bmatrix} \hat{q}_k \\ 1 \end{bmatrix},
\]

and choose

\[
\hat{\sigma}(k) \in \arg\min_{\kappa = 1, \ldots, N} (\tilde{x}_k^T F \tilde{y}_\kappa)^2.
\]

This heuristic makes sense because if the image locations were measured exactly, and the vision system were ideal, we would have that \( \tilde{x}_k^T F \tilde{y}_{\sigma(k)} = 0 \). The solution of the correspondence problem for the specific problem instance is given in Table 1.

---

Table 1: solution of the correspondence problem

<table>
<thead>
<tr>
<th>( \hat{\sigma}(k) )</th>
<th>2</th>
<th>1</th>
<th>4</th>
<th>3</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>( k )</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>5</td>
</tr>
</tbody>
</table>


```matlab
% clean up the workplace, and load the data
clear all; close all; clc
stereovision_calibration_data;

% (b) stereo-vision calibration
X = [P ; ones(1,K)];
Y = [Q ; ones(1,K)];
for k = 1:K
    M(k,:) = kron(Y(:,k)', X(:,k)') / sqrt(K);
end
[~, S, V] = svd(M);
F = reshape(V(:,end), 3, 3) * sign(V(1,9))
J = S(9,9)^2

% (c) correspondence
Xtilde = [Ptilde ; ones(1,N)];
Ytilde = [Qtilde ; ones(1,N)];
[~, sigma] = min((Xtilde' * F * Ytilde').^2)
```
2. You can’t take the sky from me [30 points]. In this problem we are going to be exploring a particular generalization of stability. Previously, we mentioned that a dynamical system is stable if it met a certain eigenvalue criterion, which is known as asymptotic stability. Under a different type of stability a system is also said to be stable about a point $x_{eq}$ if for any $\epsilon > 0$ there exists a $\delta > 0$ such that

$$||x(0) - x_{eq}|| < \delta \text{ then } ||x(t) - x_{eq}|| < \epsilon \; \forall t > 0$$

In order to study this type of stability we will use a quadratic form (rather than a linear function). The reasons for this are numerous, but one reason is that a quadratic form returns a scalar, which is easier to work with. Suppose we have an autonomous linear dynamical system in the form

$$\dot{x} = Ax.$$ 

We define a quadratic form

$$V : \mathbb{R}^n \rightarrow \mathbb{R} \quad V(x) = x^T Px.$$ 

Now some preliminaries.

a) Show that $\dot{V}(x) = -x^T Q x$, where $\dot{V}(x) = \frac{d}{dt} V(x)$. Express $Q$ as a simple function of $P$ and $A$. Formulate an equation in the form of $L(A, P, Q) = 0$, where $L(A, P, Q)$ is a very simple function of $A$, $P$ and $Q$ and has nothing to do with a Lagrangian!

**Hint:** Use the chain rule/product rule to find a relationship between $Q$, $A$ and $P$. $L(A, P, Q) = 0$ is just a simple reformulation of this relationship (not any trivial function like $L(A, P, Q) = Q - Q = 0$). The reason for this is so that we can succinctly reference that relationship later. It is nothing fancy, just notation!

b) Show that if $P > 0$ and $Q > 0$ (i.e. $P$ and $Q$ are positive definite) then there exists some $a \in \mathbb{R}$ such that $\dot{V}(x) < a V(x)$. Find the smallest possible $a$ that works for the bound as a function of eigenvalues of $P$ and $Q$.

**Note:** you can write the $a$ as a function of things like $\lambda_{max}(Q)$.

Now that we have characterizations of $Q$ and $P$ we will explore how to find and use them. We can think of $V$ as a quantity representing the *generalized* energy of the system, where $P$ is an appropriate positive definite matrix (energy $> 0$ for non-zero states). $\dot{V}$ has a similar analogy as generalized energy dissipation. One way to use this is to specify, by picking $Q > 0$, how we want the system to dissipate its energy such that $\dot{V} = -x^T Q x < 0, \; \forall x \neq 0$. Given a $Q$ and $A$ we wish to find the $P$ that solves $L(A, P, Q) = 0$ such that our target dissipation is met.

c) Suppose that the LDS described by $A$ is stable. Verify that

$$P = \int_0^\infty e^{tA^T} Q e^{tA} dt$$
is a valid solution of \( L(A, P, Q) = 0 \)

We will now use this method to verify the stability of two systems, one with which you are familiar, and another one with which you may not be familiar. The purpose for the \( V(x) \) function is to serve as a proxy for more complex system behaviors and to allow us to analyze the system as a scalar-valued differential equation. It can be shown that if we can find some positive definite function \( V(x) \) that also is monotonically decreasing, then our system will tend towards an equilibrium point \( x_{eq} \) (but might not reach it) and it will satisfy the \( \epsilon, \delta \) property defined above. Furthermore, we do not have to explicitly solve the dynamics equation of the system. Thus if we can find such a function \( V(x) \), then the system can be regarded as stable (at least under this definition of stability).

Consider a damped harmonic oscillator

\[
M \ddot{z} + C \dot{z} + Kz = 0, \quad M, C, K, z \in \mathbb{R}.
\]

d) Given \( M, C, K > 0 \), reformulate the damped harmonic oscillator as an autonomous LDS \( \dot{x} = Ax \) for \( x \in \mathbb{R}^2 \). Verify that the eigenvalues of the dynamics matrix \( A \) have non-positive real parts.

e) Suppose

\[
Q = \begin{bmatrix}
\alpha K & \alpha \frac{C}{2} \\
\alpha \frac{C}{2} & C - \alpha M
\end{bmatrix}
\]

Find conditions on \( \alpha \) that ensure \( Q \) is positive definite. It is OK to give your conditions as inequalities involving some functions of \( \alpha \).

**Hint:** The polynomial \( \chi(x) = x^2 + ax + b \) has negative roots if and only if \( a, b > 0 \).

f) Consider the system with \( M = 2, C = 1, K = 1 \). Find the corresponding \( P \) for \( \alpha = 0.1 \). Simple naive numerical integration for calculating \( P \) is fine, you may need to use a small \((10^{-4})\) time step. Plot trajectory simulations of the system for two different initial conditions of your choice. We are looking for three plots for each each initial condition:

- \( x_1 \) vs. \( x_2 \) (phase plot)
- \( x_1 \) and \( x_2 \) vs. \( t \)
- \( V(x) \) and \( \|x\| \) vs. \( t \)

Adjust initial condition to very large or very small values, comment briefly on what you see with respect to the \( \epsilon, \delta \) we mentioned before.

Verify that the \( P \) matrix you calculate solves \( L(A, P, Q) = 0 \). You can use `ode45` function if you want to simulate the trajectory, or just calculate \( e^{tA} \) for some length of time.
Suddenly you realize you are captain of a spaceship called *Tranquility*. Your ship’s main systems are failing and you need to lock in a safe landing trajectory. The dynamics are

\[
\begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2
\end{bmatrix} = A x + g(x) = \begin{bmatrix}
-2 & 3 - k \\
3 & -2
\end{bmatrix} \begin{bmatrix}
x_1 \\
x_2
\end{bmatrix} + \begin{bmatrix}
0 \\
0
\end{bmatrix}
\]

(1)

Before your ship’s AI went off-line, it suggested you need a pole angle (which is the angle the complex eigenvalue makes with respect to the real axis) of \(\theta = 56.309^\circ\), i.e. \(\tan(56.309^\circ) = \frac{|\text{Im}(\lambda)|}{|\text{Re}(\lambda)|}\). Your goal is to find the \(k^*\) that meets this criteria and analyze how it affects the stability of the system.

g) Plot the real and imaginary parts of eigenvalues of \(A\) vs \(k\) and find the value of \(k^*\) such that the pole angle constraint is met and \(\text{Re}(\lambda) < 0\).

Now that we have a candidate \(k = k^*\), We need to determine how far from \(x_{eq}\) we can initially be such that we safely reach the origin after some time. In order to do this we need to determine the region of attraction, which in this case we will define to be the largest ball about equilibrium point \(||x(0) - x_{eq}|| < \alpha\) such that any point inside the ball will tend to the equilibrium point \(x_{eq}\) which in our case is the origin. One way to do this is to first determine a function \(V(x) = x^T P x\) that acts as our systems energy, which we want to be constantly dissipating. To do this, first ignore the nonlinear part of the dynamics and determine \(P\) like you did in part (f). Use \(Q = 4I_2\). (\(I_2\) is the \(2 \times 2\) identity matrix.)

h) Determine the region where \(\dot{V}(x) < 0\) (make sure to use the full system dynamics given by Eq. (1) this time). Find the largest \(\alpha\) such that if \(||x - x_{eq}|| < \alpha\) then \(\dot{V}(x) < 0\). Report \(P\) and \(\alpha\).

**Hint:** It might help to first plot the boundary of the region where \(\dot{V}(x) < 0\) (i.e. the set \(\{x|\dot{V}(x) = 0\}\)), and then determine the largest \(\alpha\) based on the plot.

i) Pick thee initial points inside and three points outside the \(||x - x_{eq}|| < \alpha\) region and plot their trajectories (i.e. \(x_1\) vs \(x_2\)). Explain your observations in connection with the \(\epsilon,\delta\) criterion.

**Solution.** The type of stability in the problem is called *Lyapunov Stability*

a) [4 points]

\[
\dot{V}(x) = \dot{x}^T P x + x^T P \dot{x} = x^T A^T P x + x^T P A x = x^T (A^T P + PA)x
\]

Thus \(Q = -(A^T P + PA)\) and

\[
L(A, P, Q) = A^T P + PA + Q = 0
\]

Which is known as the Lyapunov Equation.
b) [3 points]
We need the fact that \( \lambda_{\text{min}}(P)x^T x \leq x^T Px \leq \lambda_{\text{max}}(P)x^T x \) If \( P > 0 \) then it follows that \( \frac{x^TPx}{\lambda_{\text{max}}(P)} < x^T x \)

Since \( Q > 0 \)
\[ \dot{V}(x) = -x^T Q x < -\lambda_{\text{min}}(Q)x^T x < -\frac{\lambda_{\text{min}}(Q)}{\lambda_{\text{max}}(P)} x^T Px < -\lambda_{\text{min}}(Q) \]
where \( a = -\frac{\lambda_{\text{min}}(Q)}{\lambda_{\text{max}}(P)} \)

Since \( \dot{V}(x) < 0 \) the function is always decreasing, so \( V(x(t)) < V(x(0)) \)

c) [3 points] Since A is stable \( \lim_{t \to \infty} e^{tA} = 0 \)
\[ A^T P + PA + Q = A^T \int_0^\infty e^{tA} Q e^{tA} dt + \int_0^\infty e^{tA} Q e^{tA} dt A + Q \]
\[ = \int_0^\infty A^T e^{tA} Q e^{tA} + e^{tA} Q e^{tA} A dt + Q \]
\[ = \int_0^\infty \frac{d}{dt} e^{tA} Q e^{tA} dt + Q \]
\[ = e^{tA} Q e^{tA} \bigg|_0^\infty + Q = 0 - e^0 Q e^0 + Q = -Q + Q = 0 \]

d) [4 points]
This system has the form
\[ \frac{d}{dt} \begin{bmatrix} \dot{x} \\ x \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{C}{M} & -\frac{K}{M} \end{bmatrix} \begin{bmatrix} \dot{x} \\ x \end{bmatrix} \]

With eigenvalues \( \lambda = \frac{-C \pm \sqrt{(C^2 - 4KM)}}{2M} \). These eigenvalues are always non-positive because the discriminate has maximum value at \( C^2 \), so the numerator will never be positive.

e) [3 points] Finding \( Q \) to be positive definite, is the same as finding \(-Q\) to be negative definite. With the characteristic polynomial we can use the criteria (Routh Hurwitz criteria) to find restrictions on \( M, C, K, \alpha \) such that the roots are negative, and thus the eigenvalues of \(-Q\) are negative making \( Q \) a positive definite matrix.
\[ \chi(\lambda) = \det(-Q - \lambda I) = \det \begin{bmatrix} -\alpha K - \lambda & \alpha \frac{C}{2} \\ \alpha \frac{C}{2} & -C + \alpha M - \lambda \end{bmatrix} \]
\[ \chi(\lambda) = \lambda^2 + (\alpha K + C - \alpha M)\lambda + (\alpha CK - \alpha^2 MK - \frac{\alpha^2 C^2}{4}) \]
We simply need \( \alpha K + C - \alpha M > 0 \) and \( \alpha CK - \alpha^2 MK - \frac{\alpha^2 C^2}{4} > 0 \)
f) [4 points]

\[ P = \begin{bmatrix} 0.50005 & 0.100025 \\ 0.100025 & 1.0004 \end{bmatrix} \]

for the two initial conditions \([-1, 1]\) and \([2, -2]\)
Notice that $V(x)$ is always monotonically decreasing (by design), while $\|x\|$ is not.

\( g) [3 \text{ points}] \)

\( k^* = 6 \)
h) [3 points] After finding $P$ calculate $\dot{V}(x)$

$$
\dot{V}(x) = x^T Q x + 2 g(x)^T P x = -4x_1^2 - 4x_2^2 + 2x_3^3 = -4x_1^2 + 2x_2^2(x_2 - 2)
$$

We are interested in the region where $-2x_1^2 + x_2^2(x_2 - 2) < 0$. To find this region we plot the contour of the $\dot{V}(x) = 0$ level set, given by

$$
x_1 = \pm \sqrt{x_2^2(x_2 - 2)/2}
$$

Note: the domain of this function is $x_2 > 2$. Next we observe that the largest circle centered about the origin contained in the region has radius $r = 2$

\[ P=1.0 -0.0 \]
\[ -0.0 1.0 \]
\[ \alpha=0.5 \]

i) [3 points]
We see that trajectories that enter the circle still stay there, but outside of this the distance to the equilibrium point does not really tell us too much. For example the yellow trajectory starts further from the origin, at \([-3, 4]\), and yet still converges to the equilibrium point. However, the cyan trajectory which starts closer \([2, 3]\) does not converge to the origin. We can say that if \(\|x(0) - x_{eq}\| < \delta = 2\) then \(\|x(t) - x_{eq}\| < \epsilon = \delta = 2\) for all \(t > 0\), but for larger \(\epsilon\) we might not be able to find such a \(\delta\) because there might be some point inside in \(\delta\) ball that still goes off to infinity.
3. Heat transfer with incomplete information [30 points]. In this problem you will discover some interesting properties of the heat equation, in connection to the gradient descent algorithm, and you will apply standard optimization techniques to solve a real-life problem.

Let \( f : \mathbb{R}^n \to \mathbb{R} \) denote an arbitrary smooth function and consider the dynamical system given by

\[
\dot{x} = -\nabla_x f(x(t)), \quad t > 0,
\]

with initial condition given by \( x(0) = x_0 \). Here \( \nabla_x f(x) = \left[ \frac{\partial f}{\partial x_1}, \ldots, \frac{\partial f}{\partial x_n} \right]^T \) denotes the standard gradient vector and \( x \in \mathbb{R}^n \). Note that throughout this problem the dot notation (e.g. \( \dot{x} \)) always means the derivative with respect to time.

Consider the function \( g(t) = f(x(t)) \). Using the Chain Rule and the definition of \( x \), we find that

\[
\dot{g} = \nabla_x f(x(t))^T \dot{x} = -\|\nabla f(x(t))\|_2^2 \leq 0.
\]

Since \( \dot{g}(t) \leq 0 \) for all \( t > 0 \), \( g \) is a non-increasing function. Roughly, this means that starting at \( x_0 \), the dynamical system 2 can be used to minimize the function \( f \). This procedure is referred to as gradient descent.

In fact, it is known that if \( f \) is a smooth convex function which has a unique minimum at \( x = x^* \), then \( x(t) \to x^* \) as \( t \to \infty \).

Now let \( C^2(\mathbb{D}) \) denote the set of twice continuously differentiable functions \( u : \mathbb{D} \to \mathbb{R} \), where \( \mathbb{D} = \{ x \in \mathbb{R}^n ||x||^2 < 1 \} \) is the open unit ball in \( \mathbb{R}^n \), and define

\[
V = \{ u \in C^2(\mathbb{D}) | u(x) = 0, \text{ if } x \in \partial \mathbb{D} \},
\]

where \( \partial \mathbb{D} = \{ x \in \mathbb{R}^n ||x||^2 = 1 \} \) denotes the boundary of \( \mathbb{D} \). Note that the \( \partial \) sign in \( \partial \mathbb{D} \) has nothing to do with a derivative.

a) Verify that \( V \) is a subspace of the known vector space \( C^2(\mathbb{D}) \). Since \( V \) is a subset of \( C^2(\mathbb{D}) \), it would suffice to show that \( V \) contains the special element 0, and is closed under addition and scalar multiplication. Note that in this case, the 0 \( \in C^2(\mathbb{D}) \) corresponds to the function \( 0 : \mathbb{D} \to \mathbb{R} \) such that \( 0(x) = 0 \) for all \( x \in \mathbb{D} \), i.e. the function which maps the unit disk to the real number 0.

Now define \( \|\cdot\|_V : V \to \mathbb{R} \) such that

\[
\|u\|_V = \left( \int_{\mathbb{D}} \|\nabla_x u(x)\|_2^2 \, dx \right)^{\frac{1}{2}}.
\]

Note: since \( u \) is sufficiently smooth, you may think of the integral defining \( \|u\|_V \) as the iterated integral:

\[
\int \cdots \int_{\mathbb{D}} \|\nabla_x u(x_1, \ldots, x_n)\|_2^2 \, dx_1 \cdots dx_n.
\]
b) Show that $\|\cdot\|_V$ defines a norm on $V$. Recall that, in general, a norm is a function which satisfies:

i. (positive definiteness) For all $u \in V$, $\|u\|_V \geq 0$ and $\|u\|_V = 0$ if and only if $u = 0$;

ii. (homogeneity) For any $\alpha \in \mathbb{R}$, $\|\alpha u\|_V = |\alpha|\|u\|_V$;

iii. (triangle inequality) For any $u, v \in V$, $\|u + v\|_V \leq \|u\|_V + \|v\|_V$.

**Hint:** You may use Minkowski’s Inequality, which states that every pair of integrable functions $f$ and $g$ satisfy

$$\left( \int_D \|f + g\|^2 dx \right)^{\frac{1}{2}} \leq \left( \int_D \|f\|^2 dx \right)^{\frac{1}{2}} + \left( \int_D \|g\|^2 dx \right)^{\frac{1}{2}}.$$

Now define the mapping $E : V \to \mathbb{R}$ such that $E(u) = \frac{1}{2}\|u\|_V^2$. We can define the derivative $E'(u) = \frac{d}{du} E(u)$ as the function $g$ which satisfies

$$\frac{d}{ds} E(u + sv) \bigg|_{s=0} = \int_D g(x)v(x)dx$$

for every fixed $v \in V$. In particular, $E'$ is called the Frechet derivative of $E$. Using integration by parts, it can be shown that $g(x) = -\Delta_x u$, where $\Delta_x u = \nabla_x \cdot \nabla_x u = \frac{\partial^2 u}{\partial x_1^2} + \cdots + \frac{\partial^2 u}{\partial x_n^2}$ denotes the Laplacian of $u$.

c) Relate the heat equation

$$\begin{cases}
\dot{u} = \alpha \Delta_x u, & x \in \mathbb{D}, \alpha \in \mathbb{R}_+,
\quad u = 0, & x \in \partial \mathbb{D},
\quad u(0, x) = f(x), & x \in \mathbb{D},
\end{cases}$$

to the gradient descent method for $E(u)$ starting at $f$. In particular, your answer should describe in words how and why the idea of gradient descent can be applied to solve the heat equation.

**Hint:** Consider the dynamical system given by Eq. (2) and the definition of $E'$.

Physically, the function $u(t, x)$ satisfying the above equations gives the temperature profile over time of a thin circular plate with thermal diffusivity $\alpha$, having the initial temperature distribution described by $f(x)$.

As you showed above, we may think of the trajectory of the LDS induced by the heat equation as the path of minimal energy $E$, or equivalently, as the path which minimizes
the norm $\|\cdot\|_V$ over time. We apply this fact in a real-life setting to aid in the solution of a one-dimensional heat transfer problem.

Suppose we are migrating a server in the Packard building to a server in the Gates building. Due to the massive volume of data we need to transfer, we decide to connect the servers via a long cable. The parts of the cable inside the buildings are kept cool at a constant temperature. However, the part of the cable outside the buildings is exposed to a sizzling Stanford day. This part of the cable becomes very hot and it is rendered useless during the day. We would like to find out how long we have to wait after the sun sets so that the cable cools below its maximum operating temperature $T$.

To be more precise, suppose there are $M + 1$ equally spaced thermometers on the cable, with one at each of the points at which the cable enters a building. If the cable’s temperature at any point exceeds the maximum operating temperature $T$, the cable cannot be used to transfer data. Without loss of generality, we shift our temperature scale so that the cool ends of the cable are kept at temperature $0$, and we suppose that the cable has unit length. We also shift at time reference such that $t = 0$ at sunset.

It is experimentally discovered that the temperature $u(t, x)$ at time $t$ and position $x$ along the cable satisfies the heat equation, as given above, where in this case we let $n = 1$ (so that $D = (-1, 1)$, where $(a, b) \subset \mathbb{R}$ denotes the interval containing every real number $x$ such that $a < x < b$, and $\partial D = \{-1, 1\}$). For simplicity, we shift and scale the domain, and consider the equation defined on the interval $(0, 1)$. That is, we suppose that

$$
\begin{aligned}
\dot{u} &= \alpha \frac{\partial^2 u}{\partial x^2}, \quad x \in (0, 1), \alpha \in \mathbb{R}_+, t > 0, \\
\{ u(t, 0) = u(t, 1) = 0, \quad t > 0, \\
\{ u(0, x) = f(x), \quad x \in (0, 1),
\end{aligned}
$$

where $f$ denotes the temperature distribution of the cable at sunset and $\alpha$ is the thermal diffusivity.

Now the initial temperature at sunset can only be measured at $M + 1$ points, where the cable has sensors. Therefore, we do not know the complete initial profile $f$, which prevents us from solving for the temperature distribution at later times. In order to avoid this problem, we decide to consider a discrete version of the heat equation, and enforce it only at the points $x_j = jh$, with $h = 1/M$, along the cable at which the sensors are located.

We also choose a (small) finite time step $k$ and define $U_j^n = u(nk, jh)$ to be the temperature evaluated at our chosen discrete points. Motivated by the approximations

$$
\begin{aligned}
\frac{d}{dt}u(nk, jh) &\approx \frac{U^n_{j+1} - U^n_j}{k}, \\
\frac{d^2}{dx^2}u(nk, jh) &\approx \frac{U^n_{j+1} - 2U^n_j + U^n_{j-1}}{h^2},
\end{aligned}
$$

17
we replace the equations given above by the following discretized version:

\[
\begin{align*}
\frac{U_{n+1}^j - U_n^j}{k} &= \alpha \frac{U_{n+1}^{j+1} - 2U_n^{j+1} + U_{n+1}^{j+1}}{h^2}, & j = 1, 2, \ldots, M - 1, \quad n \geq 0 \\
U_n^0 &= U_M^n = 0, & n \geq 0, \\
U_0^j &= f(jh), & j = 0, 1, \ldots, M + 1.
\end{align*}
\]

Suppose we measure \( F = \left[ f(h) \cdots f((M-1)h) \right]^T \) and define \( U^n = \left[ U_1^n \cdots U_{M-1}^n \right]^T \).

d) Letting \( \lambda = \alpha k / h^2 \), write the above system of linear equations as \( BU^{n+1} = U^n \), where \( B \) is an \((M-1) \times (M-1)\) matrix involving \( \lambda \).

Note that, given \( U^0 = F \), we can iteratively solve the system of equations derived in part (d) to explicitly obtain \( U^n \), the temperature profile of the cable at each of the sensor locations at time \( t = nk \), for any \( n \geq 1 \).

Regardless, life is not perfect. Some of the sensors along the cable are damaged due to excess heat exposure and we cannot use them to measure the initial temperature \( F \). We are given index set \( \mathcal{I} \) of broken sensors, such that \( F_j = f(jh) \) is unknown for each \( j \in \mathcal{I} \). Since \( F \) is not completely known, we cannot simply iteratively solve the system \( BU^{n+1} = U^n \) in order to obtain the temperature distribution at positive times. In fact, we cannot even obtain \( U^1 \! \)!

We develop a heuristic to approximately overcome this issue. The main idea is to use the fact that the trajectory of the heat equation minimizes the function \( E \) defined above. We will approximate \( U^1 \) (and more generally \( U^n \)) by minimizing a quadratic objective subject to linear constraints.

In the theory part we learned that for each fixed point \( x \in (0, 1) \), the time evolution of \( u(t, x) \) minimizes the function \( E(u(\cdot, x)) \). As a heuristic, we will approximate the \( u \) which achieves the minimum over the entire space \( V \) by a function \( v \) which is close to \( u \) and lives in a finite-dimensional subspace \( S \subset V \) of our choosing. We will do so by first finding a basis of \( S \) and then finding the linear combination \( v \) of the basis elements of \( S \) which achieves the smallest value of \( E(v(\cdot, x)) \).

In this case we consider the subspace \( S \) of \( V \) spanned by the hat functions \( \Phi_1, \ldots, \Phi_{M-1} \), where

\[
\Phi_j(x) = \begin{cases} 
\frac{x - x_{j-1}}{h}, & \text{if } x_{j-1} \leq x < x_j, \\
\frac{x_{j+1} - x}{h}, & \text{if } x_j \leq x < x_{j+1} \\
0, & \text{otherwise}. 
\end{cases}
\]

It is known that the hat functions are linearly independent, which makes them a basis of \( S \).
It turns out that since $S$ is finite dimensional, we can write the restriction of $\|\cdot\|_V$ to $S$ as a quadratic form. This means that there exists a matrix $A \in \mathbb{R}^{(M-1) \times (M-1)}$ such that for every $v(x) = \sum_{j=1}^{M-1} v_j \Phi_j(x) \in S$ we have $\|v\|_V = \sqrt{v^T AV}$, where $V = [v_1 \cdots v_{M-1}]^T \in \mathbb{R}^{M-1}$. Note that such $A$ satisfies $E(v) = \frac{1}{2} v^T A v$.

It is known that $A$ is given by

$$A_{ij} = \int_{D} \nabla_x \Phi_i(x)^T \nabla_x \Phi_j(x) dx.$$  

e) Recall that a matrix $C$ is tridiagonal if $C_{ij} = 0$ whenever $|i - j| > 1$. Using the definition of $\Phi_j$, verify that $A$ is symmetric and tridiagonal. In addition, show that $A_{ii} = 2/h$ and $A_{i,i+1} = -1/h$.

Note that since $x \in \mathbb{R}$ in our problem, you can treat the gradient as a simple derivative with respect to $x$ and you can ignore the transposition in the expression above.

f) Now let $\tilde{U}^n$ denote the temperature vector $U^n$ without the entries corresponding to the broken sensors, and let $\tilde{B}$ be such that $\tilde{B} U^1 = \tilde{U}^0$.

Let $v^{(1)}(x) = \sum_{j=1}^{M-1} U^1_j \Phi_j(x)$ be the element of $S$ obtained via linear interpolation between the coefficients of $U^1$. Note that every element of $S$ can be written in this form, since the hat functions form a basis of $S$. Therefore, we can minimize $E$ over $S$ by minimizing an appropriate quadratic form.

Use the method of Lagrange multipliers to find the $U^1$ which minimizes $E(v^{(1)})$ over $S$ subject to the problem’s equality constraints. Note that $\tilde{U}^0 = \tilde{F}$ is known.

g) Given $M, k, I$, and $\tilde{F}$ as defined in the file temp_data.m, extend the method delineated above to find the first time at which every sensor in the cable reports a temperature lower than $T = 23$.

At each time step $t = nk$, first approximate $U^{n+1}$ from $\tilde{U}^n$ by solving the minimization problem delineated in part (f). If the maximum approximated temperature exceeds $T$, then discard the approximations for the broken sensors at time...
\[ t + k \ (i.e. \ form \ \tilde{U}^{n+1} \ from \ \tilde{U}^{n+1}) \), \ and \ use \ \tilde{U}^{n+1} \ to \ solve \ for \ \tilde{U}^{n+2}. \ \text{Continue \ until} \ \\
the \ approximated \ temperature \ everywhere \ across \ the \ cable \ is \ less \ than \ T. \]

**Hint:** You may find the `gallery` function in MATLAB useful when setting up your matrices. The `spy` command might be useful to inspect the structure of your matrices.

**Note:** Read the comments in the m-file for variable definitions.

**Solution.**

(a) **[3 points]** Since \( V \) is contained in the known vector space \( C^2(D) \), we need only show that \( V \) contains the 0 vector and that it is closed under linear combination.

In this case the 0 vector is the function \( Z : D \rightarrow \mathbb{R} \) such that \( Z(x) = 0 \) for every \( x \in D \). Since \( Z \) is smooth, we have that \( Z = 0 \) on \( \partial D \) as well, which implies \( Z \in V \).

Next, note that if \( f, g \in V \) and \( \alpha, \beta \in \mathbb{R} \), we have that
\[ (\alpha f + \beta g)(x) = \alpha f(x) + \beta g(x) = 0 \]
for every \( x \in \partial D \) because \( f, g = 0 \) on \( \partial D \). Since \( C^2(D) \) is a vector space, \( \alpha f + \beta g \in C^2(D) \) and therefore \( \alpha f + \beta g \in V \).

(b) **[4 points]** We must show that \( \| \cdot \|_V \) satisfies positive definiteness, homogeneity, and the triangle inequality. Consider \( u, v \in V \).

i. Note that since \( \| \nabla_x u \|^2 \geq 0 \), we have that \( \| u \|_V \geq 0 \). Further, if \( \| u \|_V = 0 \), then
\[ \int_D \| \nabla_x u \|^2 \, dx = 0. \]

If there exists a point at which the non-negative function \( \| \nabla_x u \|^2 \) is positive, then there is in fact a small region in \( D \) on which the function is positive, which implies that the integral is positive. However, the integral is in fact zero, which implies that \( \| \nabla_x u \|^2 = 0 \). Therefore \( \nabla_x u = 0 \) by definiteness of the Euclidean norm. Since \( \nabla_x u = 0 \), we must have \( u = c \) for some constant.

Since \( u \) is smooth and \( u = 0 \) on \( \partial D \), we must have \( u = 0 \) for all \( x \in D \).

(Technical version: If there exists \( x_0 \) such that \( g(x_0) = \| \nabla u(x_0) \|^2 = r > 0 \), the continuity of \( g \) implies there exists \( \delta > 0 \) such that \( \| g(x) - g(x_0) \| < r/2 \) whenever \( \| x - x_0 \| < \delta \). Therefore \( g(x) > r/2 \) for all \( x \in B(x_0, \delta) \), where \( B(x_0, \delta) \) denotes the open ball of radius \( \delta \) centered at \( x_0 \). This implies that
\[ \int_D \| \nabla_x u \|^2 \, dx = \int_D g(x) \, dx \geq \int_{B(x_0, \delta)} g(x) \, dx > \frac{r}{2} \text{vol}(B(x_0, \delta)) > 0, \]
a contradiction.)
ii. Next note that for any $\alpha \in \mathbb{R}$ we have
\[
\|\alpha u\|_V = \left( \int_D \|\nabla_x \alpha u\|^2 dx \right)^{\frac{1}{2}} = \left( \int_D \alpha^2 \|\nabla_x u\|^2 dx \right)^{\frac{1}{2}} = |\alpha| \left( \int_D \|\nabla_x u\|^2 dx \right)^{\frac{1}{2}} = |\alpha| \|u\|_V.
\]

iii. Finally, we use Minkowski’s inequality (applied to $\nabla u, \nabla v$) to conclude that
\[
\|u + v\|_V = \left( \int_D \|\nabla_x (u + v)\|^2 dx \right)^{\frac{1}{2}} \leq \left( \int_D \|\nabla_x u\|^2 dx \right)^{\frac{1}{2}} + \left( \int_D \|\nabla_x v\|^2 dx \right)^{\frac{1}{2}} = \|u\|_V + \|v\|_V.
\]

(c) [2 points] The definition of $E'(u)$ implies that $E'(u) = -\Delta u$. Since the gradient descent method follows the path of the dynamical system $u_t = -E'(u) = \Delta u$, we can say that the solution to the heat equation follows the trajectory of the gradient descent method applied to the energy functional $E$, starting at the point $u(0, x) = f$.

(d) [5 points] We consider the $(M-1)$-vector $U^n$. We can express the equations as
\[
(1 + 2\lambda)U_{j+1}^n - \lambda(U_{j+1}^n + U_{j-1}^n) = U_j^n, \quad j = 1, \ldots, M - 1,
\]
or equivalently, $BU^{n+1} = U^n$, with
\[
B = \begin{bmatrix}
1 + 2\lambda & -\lambda & 0 \\
-\lambda & 1 + 2\lambda & -\lambda \\
& \ddots & \ddots & \ddots \\
& & -\lambda & 1 + 2\lambda & -\lambda \\
& & & 0 & -\lambda & 1 + 2\lambda
\end{bmatrix}.
\]

Using MATLAB notation, we can express $B$ as
\[
B = \text{eye}(M-1) + \lambda \cdot \text{gallery}('tridiag', M-1, -1, 2, -1).
\]

(e) [5 points] First note that if $|i - j| > 1$ then $\Phi_i(x)\Phi_j(x) = 0$ for every $x$, which implies that $A$ is tridiagonal. Further, note that $A$ is clearly symmetric. Therefore, we need only compute the $A_{ii}$ and $A_{i,i+1}$. By direct calculation, we find that
\[
\Phi'_i(x) = \begin{cases}
\frac{1}{h}, & \text{if } x_{i-1} \leq x < x_i \\
-\frac{1}{h}, & \text{if } x_i \leq x < x_{i+1}, \\
0, & \text{otherwise.}
\end{cases}
\]

Therefore, $(\Phi'(x))^2 = 1/h^2$ if $x_{i-1} \leq x < x_{i+1}$ and it is zero otherwise. Thus
\[
A_{ii} = \int_0^1 \Phi'_i(x)\Phi'_i(x) dx = 2h \cdot \frac{1}{h^2} = \frac{2}{h}.
\]
Similarly, note that \( \Phi_i'(x)\Phi_{i+1}'(x) = -1/h^2 \) if \( x_i \leq x < x_{i+1} \) and it is zero otherwise. Thus
\[
A_{i,i+1} = \int_0^1 \Phi_i'(x)\Phi_{i+1}'(x) \, dx = h \cdot -\frac{1}{h^2} = -\frac{1}{h}.
\]

(f) **5 points** The Lagrangian for this problem is given by \( L(x, \lambda) = (x^T Ax)/2 + \lambda^T (Bx - \tilde{U}^0) \), and its partials are
\[
\nabla_x L = Ax + \lambda^T \tilde{B} \\
\nabla_\lambda L = \tilde{B}x - \tilde{U}^0.
\]

The normal equations are therefore given by
\[
\begin{bmatrix}
A & \tilde{B}^T \\
\tilde{B} & 0
\end{bmatrix}
\begin{bmatrix}
x \\
\lambda
\end{bmatrix} = \begin{bmatrix}
0 \\
\tilde{U}^0
\end{bmatrix}.
\]

Solving the last system allows us to recover the optimal \( x^* = U_1 \).

(g) **6 points** The desired time \( t^* \) is given by the smallest \( n \) such that
\[
m_n \leq T,
\]
where \( m_n = \max_j U_j^n \). We find \( t^* \) via a line search: we iteratively compute \( m_n \) by solving the optimization problem delineated in part (f) and comparing it to \( T \). We stop the procedure when we find an \( n \) such that \( m_n \leq T \).

Here’s the code to solve the problem:

```matlab
%temp_data;
%Define lambda
lambda = alpha*k / h^2;

%Construct the constraint matrix B
B = eye(M-1) + lambda*gallery('tridiag', M-1);

%Remove rows from B corresponding to broken sensors in order
%to construct B_tilde
%First find the working indices
all_idx = 0:M;
working_idx = [];
for idx = 1:length(all_idx)
    loc = find(broken_idx == all_idx(idx));
    if isempty(loc)
        working_idx(end + 1) = all_idx(idx);
    end
end
working_idx = working_idx(2:end-1);
```
%Select working indices from B
B_tilde = B(working_idx, :);

%Define the Gram matrix A
A = M/2*gallery('tridiag', M-1);

%Solve normal equations to solve obtain an approximation for U^-1
N = [A B_tilde'; B_tilde zeros(length(working_idx))];
U_tilde = F_tilde;
sol = N \ [zeros(M-1,1); U_tilde];
U_approx = sol(1:M-1);

%Find smallest time at which M+1 points along the cable
%report temperature below T = 23
n = 1;
while max(U_approx) > T
    sol = N \ [zeros(M-1,1); U_tilde];
    U_approx = sol(1:M-1);
    U_tilde = sol(working_idx);
    n = n + 1;
end

%Report desired time t^*
T = n * k

%Plot the broken-sensors approximated U and the U obtained if
%all sensors were working on the same set of axes
x = linspace(0, 1, M + 1);
figure
hold on
plot(x, [0; U_approx; 0])

%Compute the U that would be obtained if all sensors were working
%Define U^0 using initial distribution from which F_tilde was drawn
%(part of solution key)
U_0 = 1.5*(7*normpdf(x, .3, .1) + 2*normpdf(x, .5, .15) + 5*normpdf(x, .8, .05))';

%Verify that U^0 and F_tilde agree on working indices (internal test of
%consistency
for i=1:length(working_idx)
    U_working = U_0(working_idx);
    assert(abs(U_working(i) - F_tilde(i)) < 1e-2)
end
U = U_0(2:M);

%Compute U^{t^*}
for k = 1:n
    U = B \ U;
end
The calculated time is approximately $t^* = 860.27s$.

Note that our approximation is pretty close to the solution obtained if we had no broken sensors, even though we are missing 10 sensors!
4. **Stochastic matrices with restart [15 points + 5 extra credit points]**. Consider a discrete time dynamical system with \( n \) modes. An example of such a system is a computer CPU where the modes could be off, sleep, idle, active, etc. Let the mode of the system at time \( t \) be \( y(t) \in \{1, \ldots, n\} \). Also, let the probability of being in mode \( i \) at time \( t \) be \( x_i(t) = \Pr\{y(t) = i\} \). The probability of transitioning from mode \( i \) to mode \( j \) is \( P_{ij} = \Pr\{y(t+1) = j \mid y(t) = i\} \). Let \( P \in \mathbb{R}^{n \times n} \) be the matrix with elements \( P_{ij} \). A system with such properties is called a *Markov chain*. We can cast the Markov chain as an autonomous linear dynamical system, where \( x(t) \in \mathbb{R}^n \) is the state vector and the dynamics equation is given by

\[
x(t + 1) = P^T x(t).
\]

From the basics of Probability theory, we can easily see that the elements of \( P \) are non-negative and the summation of each row is equal to 1, that is 
\[
\sum_{j=1}^{n} P_{ij} = 1.
\]
Any matrix with these two properties is called a stochastic matrix.

In some applications of Markov chains, at each time step the system might be forced to transition to a specific mode \( k \), with probability \( 1 - p \). In this case the dynamics will be described by a new matrix \( \tilde{P} \) defined as follows:

\[
\tilde{P} = pP + (1 - p)E_k,
\]

where \( 0 \leq p \leq 1 \), \( k \in \{1, \ldots, n\} \) is known and \( E_k \in \mathbb{R}^{n \times n} \) is a matrix with all zero elements except for the \( k \)th column that is all ones. In this question we will study the matrix \( \tilde{P} \).

a) As a warm-up, prove that the magnitude of all the eigenvalues of any stochastic matrix \( P \) is less than or equal to 1. What is the right eigenvector corresponding to eigenvalue 1? (You don’t need to and may not use the Perron-Frobenius theorem.)

**Remark:** In fact, for any stochastic matrix, the largest eigenvalue in magnitude is 1 and the rest of eigenvalues are strictly less than 1 in magnitude. However, we only ask you to prove the weaker statement above.

b) **Extra Credit:** Assume that \( \lambda_1, \ldots, \lambda_n \) are the eigenvalues of \( P \) such that \( 1 = \lambda_1 > |\lambda_2| \geq \cdots \geq |\lambda_n| \). How do the magnitude of the eigenvalues of \( \tilde{P} \) relate to the magnitude of the eigenvalues of \( P \).

**Hint:** You may find writing the characteristic polynomial of \( \tilde{P} \) useful. Also, you can use the following property of determinant:

\[
\det \left( \begin{bmatrix} a_1 + b_1 & a_2 & \cdots & a_n \end{bmatrix} \right) = \det \left( \begin{bmatrix} a_1 & a_2 & \cdots & a_n \end{bmatrix} \right) + \det \left( \begin{bmatrix} b_1 & a_2 & \cdots & a_n \end{bmatrix} \right).
\]

c) Show that the state equation

\[
x(t + 1) = \tilde{P}^T x(t)
\]
can be written as a linear dynamical system with input:

$$x(t + 1) = pP^T x(t) + (1 - p)u(t)$$

for a suitable choice of $u(t)$.

d) A vector $\pi \in \mathbb{R}^n$ with non-negative elements is called the stationary distribution of the stochastic matrix $P$ if

$$P^T \pi = \pi, \quad 1^T \pi = 1.$$ 

It’s easy to see that $\pi$ is the left eigenvector of matrix $P$ corresponding to eigenvalue 1. It’s a well-known fact that if $P$ meets certain criteria, which we assume is the case in our problem, then, for any initial state $x(0)$, $x_i(0) \geq 0$ for each $i$, and $1^T x(0) = 1$ we will have

$$\lim_{t \to \infty} x(t) = \pi.$$ 

Let $v_1, \ldots, v_n$ and $w_1, \ldots, w_n$ be the right and left eigenvectors of $P$, respectively, with eigenvalues $\lambda_1, \ldots, \lambda_n$ such that if $V = [v_1 \ldots v_n]$, and $W^T = [w_1 \ldots w_n]$ then, $V^{-1} = W$. Using part c, derive the simplest expression possible for $\tilde{\pi}$, the stationary distribution of $\tilde{P}$, in terms of $v_i$’s, $w_i$’s, $\lambda_i$’s, $p$, and $k$.

**Solution.**

a) [5 points] First note that, the vector of ones $1$ is an eigenvector for $P$ with eigenvalue $\lambda_1 = 1$ because each row of this matrix add up to 1. By contradiction, we will show that having an eigenvalue with magnitude $|\lambda| > 1$ is impossible. Consider an eigenvector $x \in \mathbb{C}^n$, and put $y = Px = \lambda x$. For any $i$, one may have

$$|y_i| = \left| \sum_{j=1}^n p_{ij} x_j \right| \leq \sum_{j=1}^n p_{ij} \left| x_j \right| \leq \max_j \left| x_j \right|.$$ 

If $|\lambda| > 1$ then we should have $\max_i |y_i| > \max_i |x_i|$, which is a contradiction.

b) [5 points, extra] Without loss of generality, let $k = 1$. Given a stochastic matrix $P$, with eigenvalues $\lambda_1 = 1 \geq |\lambda_2| \geq \ldots \geq |\lambda_n|$, we show that the matrix $\tilde{P} = pP + (1 - p)E_k$, corresponding to a random walk starting form node $k$ with restart probability $p$ has eigenvalues $\tilde{\lambda}_1 = 1 \geq |\tilde{\lambda}_2| = p|\lambda_2| \geq \ldots \geq |\tilde{\lambda}_n| = p|\lambda_n|$.

To show this, note that the characteristic polynomial of $P$ is $\chi_P(s) = (s - 1)(s - \lambda_2)\ldots(s - \lambda_n)$. Furthermore one can see that all the other eigenvalues of $E_k$ (except $\lambda_1$) are zero. We want to show that the characteristic polynomial of $\tilde{P}$ is $\chi_{\tilde{P}}(s) = (s - 1)(s - p\lambda_2)\ldots(s - p\lambda_n)$. Without loss of generality, assume $k = 1$. We have:
\[
\tilde{P} - sI = \begin{bmatrix}
    pP_{11} + (1 - p) - s & pP_{12} & \ldots & pP_{1n} \\
    pP_{21} + (1 - p) & pP_{22} - s & \ldots & pP_{2n} \\
    \vdots & \vdots & \ddots & \vdots \\
    pP_{n1} + (1 - p) & pP_{n2} & \ldots & pP_{nn} - s
\end{bmatrix}
\]

One can see that \( \det(\tilde{P} - sI) = \det(A) + \det(B) \) such that:

\[
A = \begin{bmatrix}
    pP_{11} - s & pP_{12} & \ldots & pP_{1n} \\
    pP_{21} & pP_{22} - s & \ldots & pP_{2n} \\
    \vdots & \vdots & \ddots & \vdots \\
    pP_{n1} & pP_{n2} & \ldots & pP_{nn} - s
\end{bmatrix}
\]

\[
B = \begin{bmatrix}
    (1 - p) & pP_{12} & \ldots & pP_{1n} \\
    (1 - p) & pP_{22} - s & \ldots & pP_{2n} \\
    \vdots & \vdots & \ddots & \vdots \\
    (1 - p) & pP_{n2} & \ldots & pP_{nn} - s
\end{bmatrix}
\]

Note that the determinant of \( A \) is \( \chi_{pP}(s) = (s - p)(s - p\lambda_2)\ldots(s - p\lambda_n) \). For \( B \) we have:

\[
\det B = \frac{1 - p}{p - s} \det \begin{bmatrix}
    p - s & pP_{12} & \ldots & pP_{1n} \\
    p - s & pP_{22} - s & \ldots & pP_{2n} \\
    \vdots & \vdots & \ddots & \vdots \\
    p - s & pP_{n2} & \ldots & pP_{nn} - s
\end{bmatrix} = \frac{1 - p}{p - s} \det A
\]

Therefore \( \det(\tilde{P} - sI) = \det A + \frac{1 - p}{p - s} \det A \) and thus the characteristic function of \( \tilde{P} \), i.e. \( \chi_{\tilde{P}}(s) = (s - 1)(s - p\lambda_2)\ldots(s - p\lambda_n) \).

**c) [2 points]** It suffices to put \( u(t) = e_k \).

**d) [8 points]** Fix some initial state \( x(0) \). As it is explicitly mentioned in the problem statement we have

\[
\lim_{t \to \infty} x(t) = \tilde{\pi}.
\]

Also, from part (c), we know we can use the following system with input

\[
x(t + 1) = pP^T x(t) + (1 - p)e_k.
\]

From the course we know the solution of this linear system is

\[
x(t) = (pP^T)^t x(0) + (1 - p) \sum_{\tau=0}^{t-1} (pP^T)^\tau e_k.
\]
Note that from part (a), we know that the all the eigenvalues of $pP^T$ are less than 1 in magnitude as a result

$$\lim_{t \to \infty} x(t) = \lim_{t \to \infty} (1 - p) \sum_{\tau=0}^{t-1} (pP^T)^\tau e_k = (1 - p) \sum_{\tau=0}^{\infty} (pP^T)^\tau e_k.$$ 

Finally, note that for any $\tau$, the eigenvectors of $(pP^T)^\tau$ are the same and the we have

$$\lambda_i \left( (1 - p) \sum_{\tau=0}^{\infty} (pP^T)^\tau \right) = (1 - p) \sum_{\tau=0}^{\infty} p^\tau \lambda_i^\tau = \frac{1 - p}{1 - p\lambda_i},$$

which suggest the following decomposition

$$\tilde{\pi} = \lim_{t \to \infty} x(t) = W \text{diag}(\frac{1 - p}{1 - p\lambda_1}, \ldots, \frac{1 - p}{1 - p\lambda_n})V^T e_k = \sum_{i=1}^{n} \frac{(1 - p)(v_i)_k w_i}. $$