Coordination and Control of Multi-Agent Systems (0867030)
Assignment #1

Due on Wednesday, November 12, 2025

1 Graph Theory

Problem 1.1 (3 pts)

For each of the following statements, find and draw a graph with the required property, and write the associated adjacency matrix, incidence matrix (with an arbitrary orientation), and Laplacian matrix.

i) A bipartite graph of order at least 5 where each node has at least degree 3.

ii) A *planar graph* is a graph that can be drawn in the plane so that none of its edges cross each other. Find and draw a planar 3-regular graph of order 6, and write its corresponding adjacency matrix, incidence matrix (with arbitrary orientation), and Laplacian matrix.

iii) A k*-colorable* graph is a graph where each node can be assigned a color, and there are no edges connecting nodes of a similar color. For example, a bipartite graph is always 2-colorable. Draw a 3-colorable graph and clearly identify the colors assigned to each node.

Note: The order of a graph is the number of its nodes, i.e., $|\mathcal{V}|$.

Problem 1.2 (3 pts)

The *degree sequence* of a graph $\mathcal{G}$ is a listing of the degrees of its nodes. For example, the complete graph on 3 nodes, K_3, has the degree sequence $(2, 2, 2)$. Sequences of integers are called *graphic* if we can construct a graph having the sequence as its degree sequence. Are the following sequences graphic? If yes, draw the graph, and if not, explain why!

i) $(3, 3, 2, 1, 1, 0)$

ii) $(4, 3, 2, 1)$

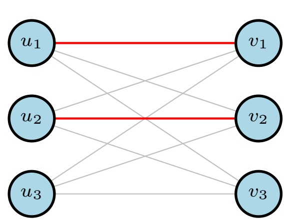
iv) $(5, 3, 3, 3, 2, 2)$

v) $(2, 2, 2, 2, 3, 3, 3, 3)$ - find 2 different graphs

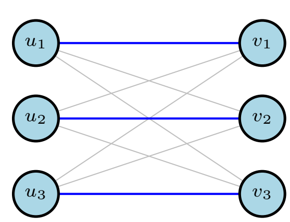
Problem 1.3 (6 pts)

A graph $\mathcal{G} = (L \cup R, \mathcal{E})$ is *bipartite* if its vertices are partitioned into two disjoint sets L and R such that all edges in $\mathcal{E}$ connect a vertex in L to a vertex in R. For a subset $S \subseteq L$, let $\mathcal{N}(S) \subseteq R$ denote the set of neighbors of the vertices in S. A *matching* is a set of pairwise edge-disjoint edges, meaning no two edges share a common vertex. A *perfect matching* is a matching in which every vertex in the graph is incident to exactly one edge of the matching, saturating both L and R when $|L| = |R|$.

Figure 1 illustrates these concepts using the complete bipartite graph $K_{3,3}$, which has partitions $L = \{u_1, u_2, u_3\}$ and $R = \{v_1, v_2, v_3\}$, with edges connecting every vertex in L to every vertex in R. Subfigure 1a shows a matching consisting of two edges, u_1, v_1 and u_2, v_2, highlighted in red, where only four vertices are matched, leaving u_3 and v_3 unmatched. Subfigure 1b depicts a perfect matching with three edges, u_1, v_1, u_2, v_2, and u_3, v_3, highlighted in blue, saturating all vertices in both L and R. In both subfigures, the full edge set of $K_{3,3}$ is shown in light gray for context.



(a) Matching in $K_{3,3}$ with 2 edges.



(b) Perfect matching in $K_{3,3}$ with 3 edges.

Figure 1: Bipartite graph $K_{3,3}$ with a matching and a perfect matching.

Theorem 1 (Hall's Marriage Theorem)**.** *A bipartite graph $\mathcal{G} = (L \cup R, \mathcal{E})$ has a perfect matching that saturates L if and only if*

$$|\mathcal{N}(S)| \;\geq\; |S| \qquad \textit{for every } S \subseteq L.$$

i) Let $L = \{u_1, u_2, u_3, u_4\}$, $R = \{v_1, v_2, v_3, v_4\}$ with edges

$$\mathcal{E} = \{\{u_1, v_1\},\ \{u_1, v_2\},\ \{u_2, v_2\},\ \{u_2, v_3\},\ \{u_3, v_3\},\ \{u_3, v_4\},\ \{u_4, v_1\},\ \{u_4, v_4\}\}\,.$$

Draw $\mathcal{G}$, verify Hall's condition, and identify a perfect matching.

ii) Let $L = \{a, b, c, d\}$, $R = \{1, 2, 3, 4\}$ with edges

$$\mathcal{E} = \{\{a, 1\}, \{b, 1\}, \{b, 2\}, \{c, 2\}, \{d, 2\}, \{d, 3\}\}.$$

Draw $\mathcal{G}$, find a subset $S \subseteq L$ with $|N(S)| < |S|$, and explain why no perfect matching exists.

iii) Prove the necessity of Hall's Theorem: If $\mathcal{G}$ has a perfect matching saturating L, then $|N(S)| \geq |S|$ for all $S \subseteq L$.

Problem 1.4 (2pts)

Recall that $\mathbf{1}$ is the vector of all ones, and $J = \mathbf{1}\mathbf{1}^T$ is a matrix of all ones. Using this notation, find an expression for the adjacency matrix of the complete graph, $A(K_n)$, and the Laplacian, $L(K_n)$. What are the eigenvalues of $A(K_n)$ and $L(K_n)$?

Problem 1.5 (4pts)

Consider the matrices

$$A_1 = \frac{1}{2}\begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix} \quad \texttt{and} \quad \frac{1}{2}\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}.$$

i) For each matrix, draw the graph associated with it.

ii) Show that both A_1 and A_2 are irreducible using *both* algebraic and graph theoretic approaches.

2 Justifying the Single-Integrator Model for Robots

Problem 2.1 (5pts)

Consider a unicycle robot with dynamics

$$\begin{aligned} \dot{x} &= v\cos(\theta) \\ \dot{y} &= v\sin(\theta) \\ \dot{\theta} &= \omega, \end{aligned}$$

where $(x, y) \in \mathbb{R}^2$ is the position of the robot and $\theta \in \mathbb{S}^1$ is the heading of the robot. The speed of the robot is controlled by the input $v \in \mathbb{R}$, and the heading is controlled by the input $\omega \in \mathbb{R}$.

Design a feedback controller that drives the robot from an initial position and heading to some desired point in $\mathbb{R}^2$ (with any heading angle). In particular, let

$$v = k_p e_p,$$

where e_p is the *distance* between the robot and the desired position, and k_p a positive constant. The heading input has the form

$$\omega = k_\theta e_\theta,$$

where $e_\theta = \theta_d - \theta$ is heading error of the robot. The “desired” orientation θ_d is the orientation (in a common global coordinate frame) that points the robot to the target position. The control gain k_θ is also a positive constant.

Simulate the controller for the initial position $(x(0), y(0), \theta(0)) = (0, 0, \pi)$ and desired position $(2, 2) \in \mathbb{R}^2$ (the heading can be arbitrary). Use $k_\theta = 1$ and simulate the results for different values of k_p (with $0 \leq k_p \leq 1$). Comment on the resulting robot trajectories (be sure to submit the plots!).

Problem 2.2 (6 pts)

Feedback linearization is a control technique used for nonlinear systems. The idea is to find a transformation of the nonlinear system into an equivalent linear system using a change of variables and a well designed control input. In this exercise, we will show that the unicycle model can be feedback linearized to an integrator point robot. As with the previous problem, we begin with the unicycle model,

$$
\begin{aligned}
\dot{x} &= v\cos(\theta) \\
\dot{y} &= v\sin(\theta) \\
\dot{\theta} &= \omega,
\end{aligned}
$$

where $(x, y) \in \mathbb{R}^2$ is the position of the robot and $\theta \in \mathbb{S}^1$ is the heading of the robot. The controls are the velocity $v \in \mathbb{R}$ and the angular velocity $\omega \in \mathbb{R}$.

i) Consider a point p that is a distance r from the center of the robot. That is,

$$
p = \begin{bmatrix} p_x \\ p_y \end{bmatrix} = \begin{bmatrix} x + r\cos(\theta) \\ y + r\sin(\theta) \end{bmatrix}
$$

Draw a picture of a two-wheeled robot and indicate its position and heading in a global reference frame, and the point p.

ii) Show that

$$
\dot{p} = \begin{bmatrix} \cos(\theta) & -r\sin(\theta) \\ \sin(\theta) & r\cos(\theta) \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}.
$$

Now, let

$$
u = \begin{bmatrix} \cos(\theta) & -r\sin(\theta) \\ \sin(\theta) & r\cos(\theta) \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix},
$$

and we see that $\dot{p} = u$ is nothing but integrator dynamics - that is, u controls the velocity of the offset point p.

iii) Given a control signal u, determine what the unicycle inputs v and ω should be to drive the unicycle.

iv) Using feedback linearization, design a control that drives the unicycle to a point in space (i.e., the go-to-point control). The control can drive the point p or the center of mass to the desired point (your choice!).

3 Cyclic Pursuit for the Unicycle Model

Problem 3.1 (10pts)

Consider a group of 5 unicycle robot with identical dynamics

$$
\begin{aligned}
\dot{x}_i &= v_i\cos(\theta_i) \\
\dot{y}_i &= v_i\sin(\theta_i) \\
\dot{\theta}_i &= \omega_i,
\end{aligned}
$$

for $i = 1, 2, 3, 4, 5$, where $(x_i, y_i) \in \mathbb{R}^2$ is the position of the ith robot and $\theta_i \in \mathbb{S}^1$ is the heading of the ith robot. The speed of the robot is controlled by the input $v_i \in \mathbb{R}$, and the heading is controlled by the input $\omega_i \in \mathbb{R}$.

Using the 'move-to-a-point' feedback controller from problem 2.1, implement a cyclic pursuit algorithm for the 5 robots. In particular, assume that robot i has access to the position position of robot $i+1$ (modulo 5), i.e., via communication. The desired point that robot i should drive to is the position of robot $i+1$. Chose $k_p = 0.1$ and $k_\theta = 0.5$ with random initial conditions for each robot.

Simulate the system for a variety of initial conditions and different values for k_p and k_θ and comment on the results.

Repeat the experiment using the feedback linearization technique.

Hint: *When computing the heading error, $e_\theta = \theta_d - \theta$, ensure that it is computed on the interval $[0\, 2\pi)$. For example, use the function `mod` in MATLAB or Simulink.*