

# Homework #5

AA 597: Networked Dynamics Systems

Prof. Mehran Mesbahi

Due: Feb 16, 2024 11:59pm

Soowhan Yi

All the codes are available at the end of the documents or here. <https://github.com/SoowhanYi94/ME597>

P1. 7.1 This chapter mainly dealt with  $\Delta$ -disk graphs, that is, proximity graph  $(V, E)$  such that  $v_i, v_j \in E$  if and only if  $\|x_i - x_j\| \leq \Delta$ . where  $x_i \in R^p, i = 1, \dots, n$ , is the state of robot  $i$ . In this exercise, we will be exploring another type of proximity graph, namely the wedge graph. Assume that instead of single integrator dynamics, the agents' dynamics are defined as unicycle robots, that is,

$$\begin{aligned}\dot{x}_i(t) &= v_i(t) \cos \phi_i(t) \\ \dot{y}_i(t) &= v_i(t) \sin \phi_i(t) \\ \dot{\phi}_i(t) &= \omega_i(t)\end{aligned}$$

Here  $[x_i, y_i]^T$  is the position of robot  $i$ , while,  $\phi_i$  denotes its orientation. Moreover,  $v_i$  and  $\omega_i$  are the translational and rotational velocities, which are the controlled inputs. Now, assume that such a robot is equipped with a rigidly mounted camera, facing in the forward direction. This gives rise to a directed wedge graph, as seen in the figure. For such a setup, if robot  $j$  is visible from robot  $i$ , the available information is  $d_{ij} = \|[x_i, y_i]^T - [x_j, y_j]^T\|$  (distance between agents) and  $\delta\phi_{ij}$  (relative interagent angle) as per the figure below. Explain how you would solve the rendezvous (agreement) problem for such a system.

In order to solve this agreement problem, we need to control the distance between two agents, especially between the leaf nodes ( $v_j$  and  $v_k$  in the example) and the center node ( $v_i$ ), and angle between them. We have a sector form of sensing area, and we need to keep them inside in order to stay connected. So we construct a potential function of  $V_i = \sum_{j \in N(i)} \left( \frac{1}{(\delta\phi_{ij} - \frac{\Delta\psi}{2})^2} + \frac{d_{ij}^2}{\rho - d_{ij}} \right)$ . Then,

$$\begin{aligned}v_i(t) = \dot{r}_i(t) &= \sum_{j \in N(i)} -\frac{dV_i}{d(d_{ij})} = - \sum_{j \in N(i)} \frac{2\rho - \|d_{ij}\|}{(\rho - \|d_{ij}\|)^2} (r_i - r_j) \\ \dot{x}_i(t) &= - \sum_{j \in N(i)} \frac{2\rho - \|d_{ij}\|}{(\rho - \|d_{ij}\|)^2} (r_i - r_j) \cos(\delta\phi_{ij}) \\ \dot{y}_i(t) &= - \sum_{j \in N(i)} \frac{2\rho - \|d_{ij}\|}{(\rho - \|d_{ij}\|)^2} (r_i - r_j) \sin(\delta\phi_{ij})\end{aligned}$$

where this  $i$  corresponds to those robot agents. Lemma 7.1 from textbook proves that  $\Omega(\rho, r_0) = \{r | V(\rho, r) \leq V(\rho, r_0)\}$  is invariant set under the above control law, where  $r_0 \in D_{G, \rho}^\epsilon$  for  $\epsilon \in (0, \rho)$ . Also the theorem 7.2 from textbook proves that this control law asymptotically converges to the static centroid.

Now we need to keep the angles between them to be in certain range, and I would use the same logic that was used in above equation.

$$\omega_i = \delta\dot{\phi}_{ij}(t) = - \sum_{j \in N(i)} \frac{-2}{(\delta\phi_{ij} - \frac{\Delta\psi}{2})^3}$$

With above equations, agents are guaranteed to stay in the sector shaped sensing area. Since we have  $\delta\phi_{ij}^0 \leq \frac{\Delta\psi}{2}$ , this  $\delta\phi_{ij}(t)$  would never increase. Therefore angle between them would never increase and distance would converge to their static centroid.

P2. Show that if a  $\delta$ -disk proximity graph with 4 agents starts connected, then, using the linear agreement protocol, it stays connected for all times.

With theorem 3.4 in the textbook, we know that the linear agreement protocol converges to the agreement set. Therefore the length of edges between two vertices would never increase. Even with the case where 2 nodes (node 1 and node 2) are located in one point  $\Delta$  away from node 3, and node 4 is located  $\Delta$  away from node 3 and  $2\Delta$  away from node 1 and 2, we proved that they do converge in one point with 4 agents from last homework(4.8). As we have started with each lengths of edges being less than or equal to  $\Delta$ , those lengths would stay less than  $\Delta$  in the progress of the linear agreement protocol and until it reaches the agreement.

P3.

$$\dot{x}_i(t) = - \sum_{j \in N_{G_d}(i)} \frac{2(\Delta - \|d_{ij}\|) - \|l_{ij} - d_{ij}\|}{(\Delta - \|d_{ij}\| - \|l_{ij} - d_{ij}\|)^2} (x_i(t) - x_j(t) - d_{ij})$$

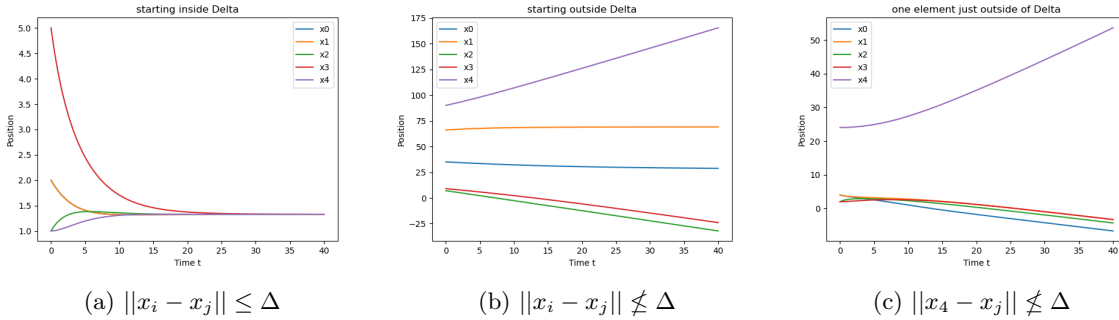
Lemma 7.5

Given an initial conditions  $x_0$  such that  $y_0 = (x_0 - \tau_0) \in D_{G_d, \Delta - \|d\|}^\epsilon$ , with  $G_d$  a connected spanning graph of  $G(x_0)$ , the group of autonomous mobile agents adopting the decentralized control law (the equation above) are guaranteed to satisfy

$$\|x_i(t) - x_j(t)\| = \|l_{ij}(t)\| \leq \Delta \text{ for all } t > 0 \text{ and } v_i, v_j \in E_d$$

Theorem 7.6 Under the same assumptions as in Lemma 7.5, for all  $i, j$ , the pairwise relative distances  $\|l_{ij}\| = \|x_i(t) - x_j(t)\|$  asymptotically converge to  $\|d_{ij}\|$  for  $v_i, v_j \in E_d$

Implement the algorithm above and explore how the conditions of Lemma 7.5/Thm 7.6 are required for the algorithm to work as proposed.



For above graphs, the desired lengths between each nodes are set to be 0 for simplicity. As we can see from above results, when those initial conditions satisfy those conditions of Lemma 7.5 and theorem 7.6, above algorithm asymptotically converges to edge length of 0. Also, those that are not satisfying those conditions ( $x_0, x_3, x_4$  in (b), and  $x_4$  in (c)) are diverging and those that are satisfying conditions ( $x_1, x_2$  in (b), and  $x_0, x_1, x_2, x_3$  in (c)) are converging. Also, when  $\|x_i - x_j\| = \Delta$ , above algorithm does not compute because we would be getting infinite number when the desired length for edges are 0. So we need this threshold  $\epsilon$  for computation. Therefore this algorithm, in order to work properly, those conditions of Lemma 7.5 and theorem 7.6 are required.

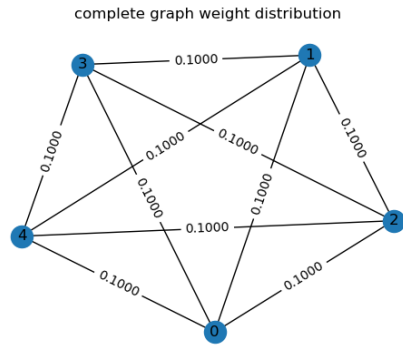
P4. 11.2 Show that if  $\lambda_2(G)$  has an algebraic multiplicity  $m$  in the graph  $G$ , then adding up to  $m$  edges will not improve it.

If  $\lambda_2(G)$  has an algebraic multiplicity of  $m$ , there are  $m$  linearly independent eigenvectors associated with this  $\lambda_2(G)$ . Also the characteristic equation for this eigenvalues is

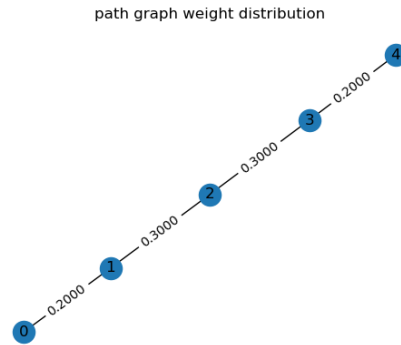
$$f(\lambda) = (\lambda_1 - \lambda)(\lambda_2 - \lambda) \cdots (\lambda_{m+1} - \lambda) \cdots (\lambda_n - \lambda) = 0$$

where  $\lambda_2 = \lambda_3 = \cdots = \lambda_{m+1}$ . If single edge is added, the connectivity increases, and increases the value of  $\lambda_2$ . However, the eigenvectors associated with those  $\lambda$ s are linearly independent, and therefore the value of  $\lambda_3, \lambda_4 \cdots \lambda_m$  remains the same, making the second smallest eigenvalue to become  $\lambda_3, \lambda_4 \cdots \lambda_{m+1}$ . Therefore adding edges upto  $m$  edges would not change the value of second smallest eigenvalue.

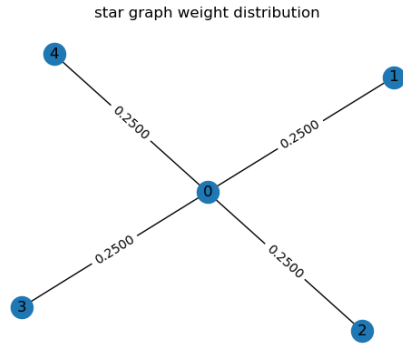
P5. 11.10 Use the approach of section 11.6 and a semidefinite programming solver (such as the one mentioned in notes and references), to maximize  $\lambda_2(G)$  for the weighted versions of  $K_5$ ,  $P_5$ , and  $S_5$ , subject to a normalization on the sum of the weights. Comment on any observed patterns for the optimal weight distribution.



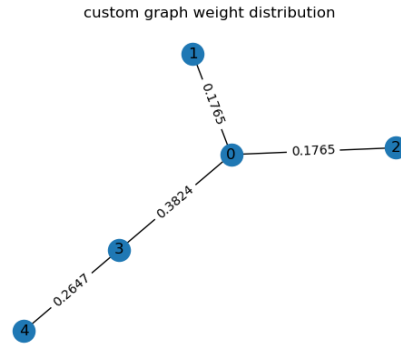
(a)  $K_5$  weight distribution



(b)  $P_5$  weight distribution



(c)  $S_5$  weight distribution



(d) Textbook example

We see that those weights are summed to 1 for all graphs and they are symmetric respect to their edges, in case of  $K_5$ ,  $P_5$ ,  $S_5$ . Of course the custom graph (textbook example) would not have symmetric weights for edges that are not symmetric. But the edge connecting (0,1) and (0,2) are symmetric and their weights are the same.

Code