MECH 6V29: Multiagent Robotic Systems- HW 4

Jonas Wagner

2022, April 6^{th}

Contents

Preliminary Notes

a) Definitions

Definition 1. *Graph $G(V, E)$ is constructed with vertex set*

$$V = \{v_1, v_2, \ldots, v_n\}$$

of n discrete vertices and edge set

$$E = \{e_1, \ldots, e_m\} \subseteq V \times V$$

consisting of m edges $e_{k=(i,j)} = (v_i, v_j) \forall_{k=1,\ldots,m}$ connecting vertices v_i and v_j.

Definition 2. *Let $V = \{v_1, v_2, \ldots, v_n\}$ be vertices. Δ-Disk Graphs are constructed for a particular Δ such that*

$$(v_i, v_j) \in E \iff \|v_i, v_j\| \leq \Delta$$

Definition 3. *Let $V = \{v_1, v_2, \ldots, v_n\}$ be vertices. A Gabriel Graph is defined as $G(V, E)$ in which*

$$\forall_{v_i, v_j \in V} (v_i, v_j) \in E \iff \forall_{v_k \in V} v_k \notin D(v_i, v_j)$$

where $D(a, b)$ is the closed disc with diameter between (a, b). In other words, a disk constructed from two adjacent vertices should not contain any other vertices.

Definition 4. *Let graph $G(V, E)$ with $V = \{v_1, \ldots, v_n\}$ and $E \subseteq V \times V$.*

a. $G(V, E)$ is considered undirected if

$$(v_i, v_j) \in E \iff (v_j, v_i) \in E$$

otherwise, $G(V, E)$ is considered directed.

b. An undirected graph $G(V, E)$ is considered connected if there exists a path between any two vertices.

c. A directed graph $G(V, E)$ is considered strongly connected if there exists a directed path between any two vertices.

d. A directed graph $G(V, E)$ is considered weakly connected if the corresponding undirected graph is connected.

Definition 5. *Let graph $G(V, E)$ with $V = \{v_1, \ldots, v_n\}$ and $E \subseteq V \times V$.*

a. The degree matrix $\Delta \in \mathbb{R}^{n \times n}$ is a diagonal matrix defined as

$$\Delta := \begin{bmatrix} \deg(v_1) & & & \\ & \deg(v_2) & & \\ & & \ddots & \\ & & & \deg(v_n) \end{bmatrix}$$

b. The adjacency matrix $A \in \mathbb{R}^{n \times n}$ is a symmetric matrix ($A = A^T$) defined s.t.

$$A = [a_{ij}] \ : \ a_{ij} \begin{cases} 1 & (v_i, v_j) \in V \\ 0 & (v_i, v_j) \notin V \end{cases}$$

c. *The* <u>incidence matrix</u> $D \in \mathbb{R}^{n \times m}$ *is defined as*

$$D = [d_{ij}] \ : \ d_{ij} \begin{cases} 1 & (v_i, -) \in e_j \\ -1 & (-, v_i) \in e_j \\ 0 & otherwise \end{cases}$$

d. *The* <u>Laplacian matrix</u> $L \in \mathbb{R}^{n \times n}$ *is a symmetric* ($L = L^T$) *and strictly semi-positive definite* ($L \succeq 0$) *is defined as*

$$L := \Delta - A = DD^T$$

and

$$L = \begin{bmatrix} \deg(v_1) & -a_{12} & -a_{13} & \cdots & -a_{1n} \\ -a_{21} & \deg(v_2) & -a_{23} & \cdots & -a_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -a_{n1} & -a_{n2} & -a_{n3} & \cdots & \deg(v_n) \end{bmatrix}$$

e. *For a weighted graph* $G(V, E, W)$*, the diagonal weighted matrix* $W \in \mathbb{R}^{m \times m}$ *is defined as*

$$W = [w_{ij}] \forall_{ij \in E}$$

were w_{ij} *are the corresponding weights for* $e_{ij} = (v_i, v_j)$.

Definition 6. *Consider a collection of* N <u>robots</u>*.* <u>Formation graph</u> $G(V, E_f, \omega)$ *consists of vertex set*

$$V = \{v_1, v_2, \ldots, v_N\}$$

of N *vertices* v_i *associated with robot* i*, edge set* E_f

$$E_f = \{e_1, \ldots, e_m\} \subseteq V \times V$$

of m *edges* $e_{k=(i,j)} = (v_i, v_j)$ *that indicate knowledge of the distance between robots* v_i *and* v_j*, and* $\omega : E_f \to \mathbb{R}_+$ *which associates a feasible desired inter-agent distance to each pair in* E_f.

Definition 7. *Consider a collection of* N *robots connected in formation graph* $G(V, E_f, \omega)$*. Let each robot be located at* <u>position</u> $P_i \in \mathbb{R}^d$ *within euclidean space* $(\mathbb{R}^p, \|\cdot\|_2)$.

a. *The* <u>formation position set</u> *is defined as the collection of associated robot positions*

$$P = \{P_1, \ldots, P_N\} \subseteq \mathbb{R}^{p \times p}$$

b. *The set of* <u>pair-wise inter-robot distances</u> *is defined by*

$$D = \{d_{ij} \geq 0 \ : \ d_{ij} = d_{ji}, \forall_{i,j \in i, \ldots, N}\}$$

where d_{ij} *is the distance* $\|P_i - P_j\|$.

c. D *is considered* <u>feasible</u> *if*

$$\exists_{P_1, \ldots, P_N \in \mathbb{R}^d} \ : \ \|P_i - P_j\| = d_{ij} \forall_{i,j \in \{1, \ldots, N\}}$$

d. *A* <u>framework</u> (G_f, P) *is a combination of a formation graph* G_f *and a set of feasible points* P.

e. *Framework* (G_f, P) *is considered* *generic* *if* P *is algebraically independent over* $\mathbb{Q}$. *(i.e. that the points are not collinear in 2-D or co-planer in 3-D)*

Definition 8. *Consider frameworks* (G, P_0) *and* (G, P_1).

a. (G, P_0) *and* (G, P_1) *are* *equivalent* *if*

$$\|P_0(i) - P_0(j)\| = \|P_1(i) - P_1(j)\| \forall_{(i,j)\in E_f}$$

meaning $d_{ij}^{(0)} = d_{ij}^{(1)}$ *for the vertices that are neighbors.*

b. (G, P_0) *and* (G, P_1) *are* *congruent* *if*

$$\|P_0(i) - P_0(j)\| = \|P_1(i) - P_1(j)\| \forall_{(i,j)\in V\times V}$$

meaning $d_{ij}^{(0)} = d_{ij}^{(1)}$ *for every vertex.*

c. (G, P_0) *is* *Globally Rigid* *if*

$$(G, P_1) \text{ equivalent } (G, P_0) \implies (G, P_1) \text{ congruent } (G, P_0)$$

d. (G, P_0) *is* *rigid* *if*

$$\exists_{\epsilon>0}\forall_{P_1} \ : \ (G, P_1) \text{ equivalent } (G, P_0) \wedge \forall_{i\in V}\|P_0(i) - P_1(i)\| < \epsilon \implies (G, P_1) \text{ congruent } (G, P_1)$$

Remark: *A framework being rigidity is equivalent to saying that every continuous motion maintaining distances where edges exist also maintains the distances between all other vertex pairs.*

Definition 9. *Let* (G, P_0) *be a d-dimensional generic framework.*

a. *The* *Rigidity Matrix* *is defined by the equations*

$$(x_i - x_j)^T(\dot{x}_i - \dot{x}_j) = 0 \ \forall_{i,j\in E}$$

resulting in a matrix $R(P_0)$ *so that* $R(P_0)\dot{P} = 0$.

b. ***Rigidity Test:*** (G, P_0) *is rigid if and only if*

$$\text{rank}(R(P_0)) = \begin{cases} 2N-3 & d=2 \\ 3N-6 & d=3 \end{cases}$$

Additionally, the rank of the rigidity matrix remains the same for all generic realizations, thus G *can be called* *Generically Rigid* *if any feasible generic realization is rigid.*

Definition 10. G_f *is considered* *minimally rigid* *if it is rigid and the removal of any single edge renders it not rigid.*

Additionally, G_f *is minimally rigid if and only if it is rigid and contains*

$$\begin{cases} 2N-3 \text{ edges} & d=2 \\ 3N-6 \text{ edges} & d=3 \end{cases}$$

Problem 1

State a summary of **Notes 13 - 15**, (which Include the topics of persistence, combinatorial coverage and graph grammars) preferably by creating a concept map diagram (flow diagram). The whole purpose is to make sure that we are clear about the bigger picture, and reiterate why are we doing and discussing the specific topics in the class. Do not merely write the topics, instead create connections between topics to clarify the flow of information.

a) Big Picture Chart

TO DO: Update all the graph to this time...

Problem 2

Preliminaries

Definition 11.

Definition 12. *Consider directed graph G_F.*

a. *G_F is Rigid if certain interagent distances are maintained then all interagent distances are maintained when the formation moves smoothly.*

b. *G_F is Constraint Consistent if the directed graph is able to maintain the specified interagent distances.*

c. *G_F is considered persistent if and only if G_F is Rigid and Constraint Consistent.*

A MATLAB Code:

All code I write in this course can be found on my GitHub repository:
https://github.com/jonaswagner2826/MECH6V29_MultiagentRoboticSystems