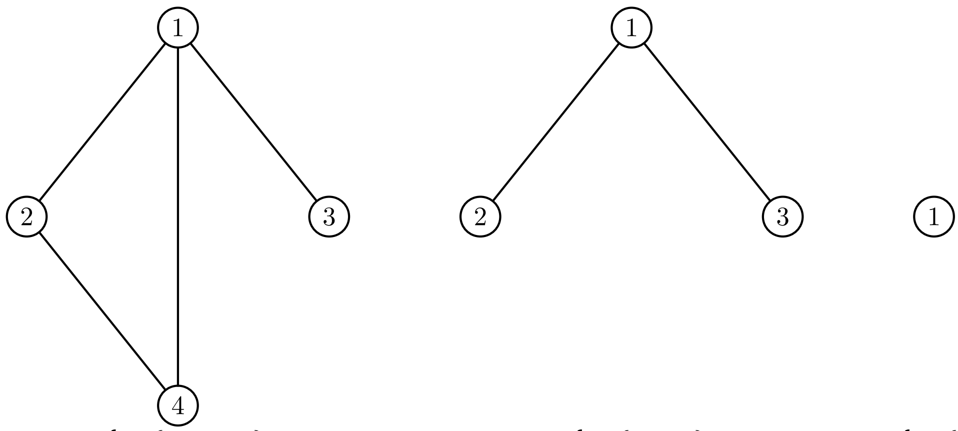
Lecture 11 — Basic Graph Theory

Much of the discussion that follows is taken from Bertsekas and Gallager section 5.2. For those who already know graph theory, we are talking about undirected graphs with no arcs incident to the same node (later we will move on to talking about directed graphs).

Definition 1. A graph $G = (\mathcal{N}, \mathcal{A})$ is a finite set of $\mathcal{N}$ nodes and a set $\mathcal{A}$ of unordered pairs (i, j) where $i, j \in \mathcal{N} : i \neq j$ (known as arcs).

Definition 2. If n_1 and n_2 are nodes and (n_1, n_2) (where $n_1 \neq n_2$) is an arc then this arc is said to be *incident* on n_1 and n_2.



$\mathcal{N} = \{1, 2, 3, 4\}$
$\mathcal{A} = \{(1, 2), (1, 3), (1, 4), (2, 4)\}$

$\mathcal{N} = \{1, 2, 3\}$
$\mathcal{A} = \{(1, 2), (1, 3)\}$

$\mathcal{N} = \{1\}$
$\mathcal{A} = \{\}$

Definition 3. A *walk* in a graph G is a sequence of nodes in a graph $(n_1, n_2, \ldots, n_l)$ such that each adjacent pair $(n_1, n_2), (n_2, n_3), \ldots (n_{l-1}, n_l)$ are arcs in G.

Definition 4. A *path* is a walk with no repeated nodes.

Definition 5. A *cycle* is a walk with $n_1 = n_l$ where $l > 3$ and with no other repeated nodes.

Definition 6. A *connected graph* is a graph where for any two nodes i and j we can find a walk which begins at i and ends at j.

Lemma 1. *Let $G = (\mathcal{N}, \mathcal{A})$ be a connected graph. For any S such that $S \subset \mathcal{N}$ and $S \neq \varnothing$ we can find at least one arc (i, j) in $\mathcal{A}$ such that $i \in S$ and $j \notin S$.*

Proof. The proof is almost immediate. If G is connected then we can find a walk $(n_1, n_2, \ldots, n_l)$ where $n_1 \in S$ and $n_l \notin S$. However, since $n_1 \in S$ then $n_2 \in S$ since no arc (n_1, n_2) exists with $n_1 \in S$ and $n_2 \notin S$. By a similar argument $n_3 \in S$ and so on until we find that $n_l \in S$ which is a contradiction since $n_l \notin S$. Thus no such walk can exist. $\square$

Definition 7. A graph $G' = (\mathcal{N}', \mathcal{A}')$ is a subgraph of a graph $G = (\mathcal{N}, \mathcal{A})$ if G' is a graph, $\mathcal{N}' \subset \mathcal{N}$ and $\mathcal{A}' \subset \mathcal{A}$.

Definition 8. A tree is a connencted graph with no cycles. A spanning tree of a graph G is a subgraph of G which is a tree and includes all the nodes of G.

Given a connected graph $G = (\mathcal{N}, \mathcal{A})$ we can trivially construct a subgraph $G' = (\mathcal{N}', \mathcal{A}')$ which is a spanning tree using the following simple algorithm:

- Choose n an arbitrary node in $\mathcal{N}$. Let $\mathcal{N}' = n$ and let $\mathcal{A}' = \varnothing$.
- If $\mathcal{N}' = \mathcal{N}$ then our graph is a spanning tree. Stop here.
- Let $(i, j) \in \mathcal{A}$ be an arc such that $i \in \mathcal{A}'$ and $j \in \mathcal{N} - \mathcal{N}'$. Now update $\mathcal{N}'$ and $\mathcal{A}'$ by adding elements and updating $\mathcal{N}'$ and $\mathcal{A}'$ so that $\mathcal{N}' = \mathcal{N}' \cup \{j\}$ and $\mathcal{A}' = \mathcal{A}' \cup \{(i, j\}$.
- Go to step 2.

Proposition 2. *The above algorithm will generate a spanning tree.*

Proof. Our initial graph $G' = (\{n\}, \varnothing)$ is trivially a tree. Lemma 1 ensures that we can find an arc $(i, j) \in \mathcal{A}$ such that $i \in \mathcal{N}'$ and $j \in \mathcal{N} - \mathcal{N}'$ if we consider the set S to be the nodes currently in G'. Since G is connected then there must exist an arc (i, j) with the property declared. Our new graph can contain no cycles since each arc we add cannot be part of a cycle since one end of it is only incident to j and no other parts of the graph G'. Hence, we must be able to increase the number of nodes in our graph by one each time without forming any cycles and eventually we will have a spanning tree. □

This leads immediately to the following proposition:

Proposition 3. *If G is a connected graph with N nodes and A arcs then:*

- *G contains a spanning tree.*
- *$A \geq N - 1$.*
- *G is a tree if and only if $A = N - 1$*

Proof. The first part follows immediately from our algorithm which shows how to construct such a spanning tree. The second part follows from the fact that, on the first iteration of our algorithm the number of nodes in $\mathcal{N}'$ is 1 and the number of arcs in $\mathcal{A}'$ is 0. At each iteration the number of nodes in $\mathcal{N}'$ and the number of arcs in $\mathcal{A}'$ is increased by exactly one. By the final iteration, all the nodes in $\mathcal{N}$ are in $\mathcal{N}'$. The number of arcs in $\mathcal{A}'$ is the number of nodes in $\mathcal{N} - 1$. Since $G' \subseteq G$ then the number of arcs in $\mathcal{A}$ is at least the number in $\mathcal{A}'$.

The final part can be seen by the fact that if we add an arc to a tree then it will cease to be a tree since it must contain a cycle (since the arc added must be between two nodes which are already connected by some other path). Therefore, G is only a tree if $\mathcal{A} = \mathcal{A}'$. □