Network Flow

In studying network structure in first part of course, we focused on both nodes and edges.

In studying network dynamics we were largely concerned with state variables defined on nodes and their evolution.

In studying network flow, we will put our focus on edges and how the limitations of the edges [capacity, noise, etc.] limit the exchange of commodities.

Dating back to classic results of Kirchhoff, et al.

What are Kirchhoff's laws?

Kirchhoff's Current Law (KCL)

Principle of conservation of electric charge implies that at any node in electrical network, sum of currents flowing into node is equal to sum of currents flowing out of node.

\[ \sum_{k=1}^{n} I_k = 0 \]

Kirchhoff's Voltage Law (KVL)

Principle of conservation of energy implies that sum of voltages around any closed network is zero.

\[ \sum_{k=1}^{n} V_k = 0 \]
Laplacian matrix of graph is also called the Kirchhoff matrix

$$L = D - A$$

for simple graphs

where

$$L_{ij} = \begin{cases} 
\text{deg}(v_i) & \text{if } i \neq j \\
-1 & \text{if } i = j \text{ and } v_i \text{ adjacent to } v_j \\
0 & \text{otherwise}.
\end{cases}$$

$L$ is symmetric, positive semidefinite.

It has eigenvalues $0 = \lambda_0 \leq \lambda_1 \leq \ldots \leq \lambda_n$.

Kirchhoff's matrix tree theorem specifies that the number of spanning trees in a graph is:

$$t(G) = \frac{1}{n} \lambda_1 \lambda_2 \cdots \lambda_{n-1}$$

where a spanning tree $T$ is a subgraph of a graph $G$ that is a tree which includes all of the vertices of $G$ with minimal possible number of edges.

Spanning trees are often important subcomponents in solving network flow problems.

There are seemingly mathematical connections between dynamics and flow problems given the graph Laplacian arises in both places.

Network flow studies really took off in 1956 with statements of maximum flow theorems.

- largely studied in OR, CS, communication theory, etc.
- Not too much study in network science, so this part of course is a little distant from a typical physics-style network science core.
Networks with flow

Power networks have electrical flow
Trade networks have customer product flow
Transportation networks have vehicle flow
Communication networks have message flow.

Typically want flow to be as efficient as possible.

> Key focus on optimization algorithms for optimization problems.

1. **Shortest Path Problem**: what is the best way to traverse a network to get from one point to another as cheaply as possible.

2. **Maximum Flow Problem**: if a network has capacities on edges, how can we send as much flow as possible between two points in the network?

3. **Minimum Cost Flow Problem**: if we incur cost per unit flow on network with edge weights and need to send units of goods from one or more points, to other points, how do it with minimal cost?

**Minimum Cost Flow Problem**

> most fundamental problem.

Let \( G = (V, E) \) be directed network with

\((ij) \in E \) having cost \( c_{ij} \), for cost per unit flow.

Capacity \( u_{ij} \) limits on flow.

lower bound \( l_{ij} \)
Associate with each node \( i \in V \), an integer \( b(i) \) representing supply/demand:

\[
\begin{align*}
    b(i) > 0 & : \text{ supply node} \\
    b(i) < 0 & : \text{ demand node} \\
    b(i) = 0 & : \text{ transshipment node}
\end{align*}
\]

Decision variables in optimization problem on flows on \((i,j) \in E\), \(x_{ij}\).

Then we want

\[
\begin{align*}
    \min \sum_{(i,j) \in E} c_{ij} x_{ij} \\
    \text{s.t.:} \sum_{j : (i,j) \in E} x_{ij} - \sum_{j : (j,i) \in E} x_{ji} = b(i) \quad \text{for all } i \in V. \\
    l_{ij} \leq x_{ij} \leq u_{ij} \quad \text{for all } (i,j) \in E. \\
    \sum_{i=1}^{k} b(i) = 0.
\end{align*}
\]

Can also be written in matrix vector form as

\[
\begin{align*}
    \min \ c x \\
    \text{s.t.:} \ N x = b \\
    l \leq x \leq u
\end{align*}
\]

where \( N \) is an \( n \times m \) matrix called the node-edge incidence matrix:

Each column of \( N \) corresponds to a remaining \( k_{ij} \). The column \( N_{ij} \) has a +1 in the \( i \)th row, a -1 in the \( j \)th row, zero otherwise.
Shortest path problem

Source node $s$, Sink node $t$: want minimum cost path (don't really worry about flow constraints).

So how can we set up minimum cost flow problem to solve shortest path problem?

Set $b(s) = 1, b(t) = -1, b(i) = 0$ otherwise.

Maximum flow problem

Source node $s$, Sink node $t$: want maximum steady-state flow without cost constraints.

How to set up minimum cost flow to solve maximum flow.

Set $b(i) = 0$ for all $i \in V$, set $c_{ij} = 0$ for all $(i,j) \in E$.

Introduce additional edge $(t,s)$ with cost $c_{ts} = -1$ and flow bounds $u_{ts} = \infty$.

Then optimization maximizes flow on edge $(t,s)$.

But since any flow on $(t,s)$ must travel from node $s$ to node $t$ through edges in $E$ (since each $b(i) = 0$), solution will maximize flow from node $s$ to node $t$. 