

Lecture19 ShortestPaths3

Tuesday, October 20, 2020 9:20 PM

Linear Programming

Let A be an $m \times n$ matrix, b be an m -vector, and c be an n -vector. Find an n -vector x that maximizes $c^T x$ subject to $Ax \leq b$, or determine that no such solution exists.



Algorithms for the general problem

- Simplex methods – practical, but worst case exponential time
- Interior-point methods – polynomial time and competes with simplex.

Feasibility problem: No optimization criterion.

Just find x such that $Ax \leq b$.

- In general, just as hard as or harder than LP.

Solving a system of difference constraints

Linear programming where each row of A contains exactly one 1, one -1 , and the rest 0's.

Example:

$$\begin{cases} x_1 - x_2 \leq 3 \\ x_2 - x_3 \leq 1 \\ x_1 - x_3 \leq 2 \end{cases}$$

Solution:

$$\begin{cases} x_1 = 3 \\ x_2 = 0 \\ x_3 = 2 \end{cases}$$

Constraint graph:



- For each unknown create a vertex.
- For each inequality / constraint put an edge from the one with negative sign to the one with positive sign. The weight of edge will be the right hand side of the inequality.

Unsatisfiable constraints

Theorem. If the constraint graph contains a negative-weight cycle, then the system of differences is unsatisfiable.

Proof. Suppose that the negative-weight cycle is $v_1 \rightarrow v_2 \rightarrow \dots \rightarrow v_k \rightarrow v_1$. Then, we have

$$\begin{aligned} x_2 - x_1 &\leq w_{12} \\ x_3 - x_2 &\leq w_{23} \\ &\vdots \\ x_k - x_{k-1} &\leq w_{k-1,k} \\ x_1 - x_k &\leq w_{k1} \end{aligned}$$

$$\frac{0}{0} \leq \text{weight of cycle} < 0$$

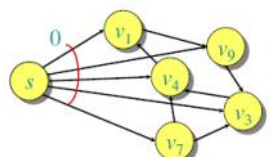
Therefore, no values for the x_i can satisfy the constraints. \square

- Right side: negative number \rightarrow contradiction

Satisfying the constraints

Theorem. Suppose no negative-weight cycle exists in the constraint graph. Then, the constraints are satisfiable.

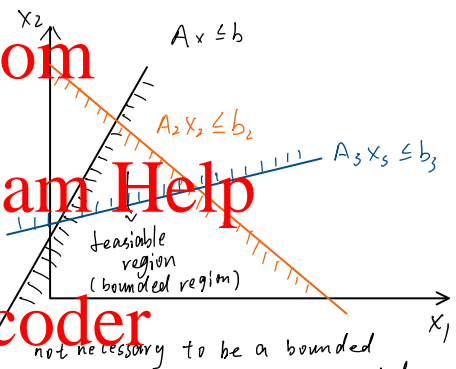
Proof. Add a new vertex s to V with a 0-weight edge to each vertex $v_i \in V$.



Note: No negative-weight cycles introduced \Rightarrow shortest paths exist.

- s : dummy variable, a vertex that connect to all vertices.

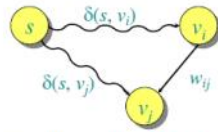
subject to $Ax \leq b$
max $c^T x$



Optimal could be one of the corner of the convex region

Claim: The assignment $x_i = \delta(s, v_i)$ solves the constraints.

Consider any constraint $x_j - x_i \leq w_{ij}$, and consider the shortest paths from s to v_j and v_i :



The triangle inequality gives us $\delta(s, v_j) \leq \delta(s, v_i) + w_{ij}$. Since $x_i = \delta(s, v_i)$ and $x_j = \delta(s, v_j)$, the constraint $x_j - x_i \leq w_{ij}$ is satisfied. \square

Bellman-Ford and linear programming

Corollary. The Bellman-Ford algorithm can solve a system of m difference constraints on n variables in $O(mn)$ time.

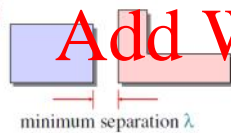
Single-source shortest paths is a simple LP problem.

In fact, Bellman-Ford maximizes $x_1 + x_2 + \dots + x_n$ subject to the constraints $x_j - x_i \leq w_{ij}$ and $x_i \leq 0$ (exercise).

Bellman-Ford also minimizes $\max_i \{x_i\} - \min_i \{x_i\}$ (exercise).

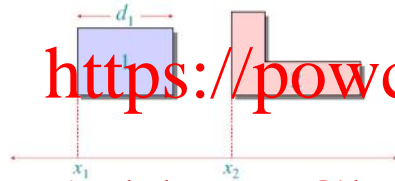
Application to VLSI layout compaction

Integrated
-circuit
features:



Problem: Compact (in one dimension) the space between features in a VLSI layout without bringing any features too close together.

- Subject to the second optimization function above



Constraint: $x_2 - x_1 \geq d_1$

Bellman-Ford minimizes $\max_i \{x_i\} - \min_i \{x_i\}$, which compacts the layout in the x -dimension.