The Minimum Cost-to-Time Ratio Cycle Problem

By: Nick Rogers

Background

- Consider a min-cost flow problem where the arcs have profits p_{ij} and traversal times τ_{ij}
- We want to find the best ratio of cost versus time to traverse a cycle
- Motivation: the tramp steamer problem
    - Also has applications in data storage (see Ch. 19 in Orlin)

Goal: Find Minimum Cycle

- We set $c_{ij} = -p_{ij}$ to transform this into a minimization problem

$$\mu(W) = \frac{\sum_{(i,j)\epsilon W} c_{ij}}{\sum_{(i,j)\epsilon W} \tau_{ij}}$$

- We assume all data is integral
- $\tau_{ij} \geq 0$ for every arc $(i, j) \in W$
  - So, the sum of arc traversal time will be at least 0.

Searching for μ^*

- Use a negative cycle detection algorithm to find μ^*, the optimal solution

- Set $l_{ij} = c_{ij} - \mu\tau_{ij}$ for an initial guess μ

- Three cases:
  - G contains a negative cycle
  - G contains a zero cycle and no negative cycles
  - Every directed cycle in G has a positive length

Case 1: G Contains a Negative Cycle

- $\sum_{(i,j)\in W} l_{ij} < 0$

- $\mu > \frac{\sum_{(i,j)\in W} c_{ij}}{\sum_{(i,j)\in W} \tau_{ij}}$ for every cycle in G

- μ is a strict upper bound on μ^*

Case 2: G Contains a Zero-Cost Cycle and No Negative Cycles

- $\sum_{(i,j)\in W} l_{ij} \geq 0$ for all directed cycles in G
- $\mu \leq \frac{\sum_{(i,j)\in W} c_{ij}}{\sum_{(i,j)\in W} \tau_{ij}}$ for all directed cycles in G
- $\mu = \frac{\sum_{(i,j)\in W^*} c_{ij}}{\sum_{(i,j)\in W^*} \tau_{ij}}$ for some directed cycle W^* in G
- $\mu = \mu^*$, and we have found the value of a minimum cost-to-time ratio cycle.

Case 3: G Contains Only Positive Length Cycles

- $\sum_{(i,j)\in W} l_{ij} > 0$

- $\mu < \frac{\sum_{(i,j)\in W} c_{ij}}{\sum_{(i,j)\in W} \tau_{ij}}$ for every cycle in G

- μ is a strict lower bound on μ^*

Two Algorithms to Find μ

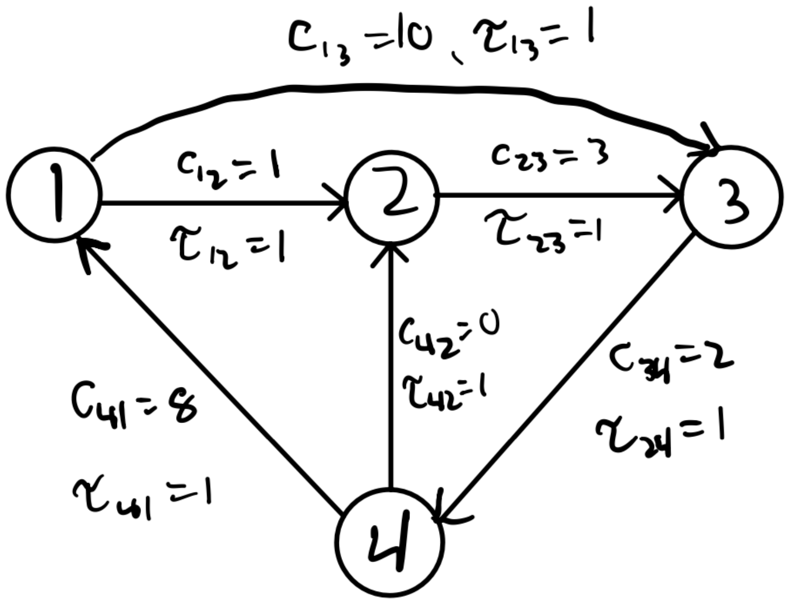
- Sequential Search

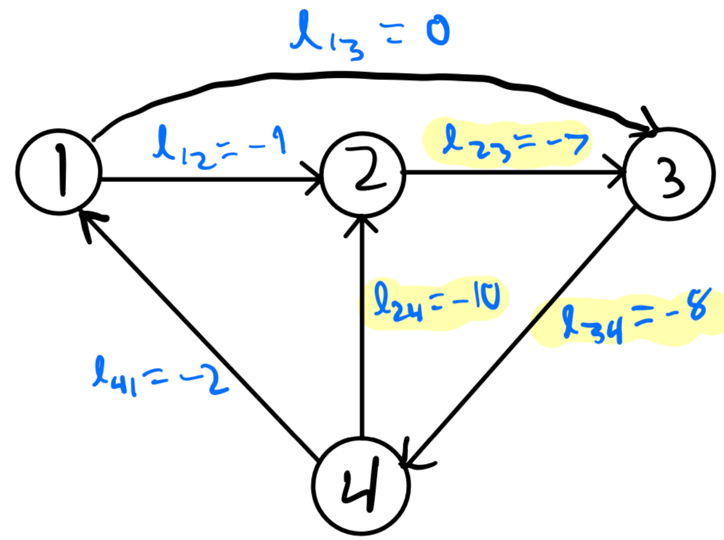
- Binary Search

Sequential Search

- Start with a known upper bound on μ^*, call it μ^0
- Compute $l_{ij} = c_{ij} - \mu^0 \tau_{ij}$ and check for cycles
- If we have a zero-length cycle and no negative cycles, we're done
- If we have negative cycles, set $\mu^1 = \frac{\sum_{(i,j)\in W^*} c_{ij}}{\sum_{(i,j)\in W^*} \tau_{ij}}$ and repeat
- This runs in pseudo-polynomial time

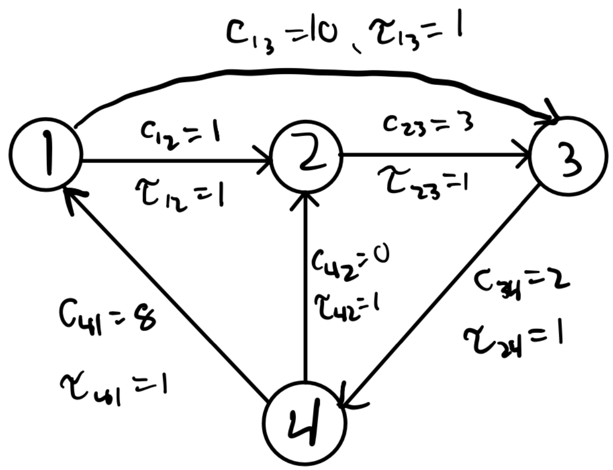
Example: $\mu^0 = 10$

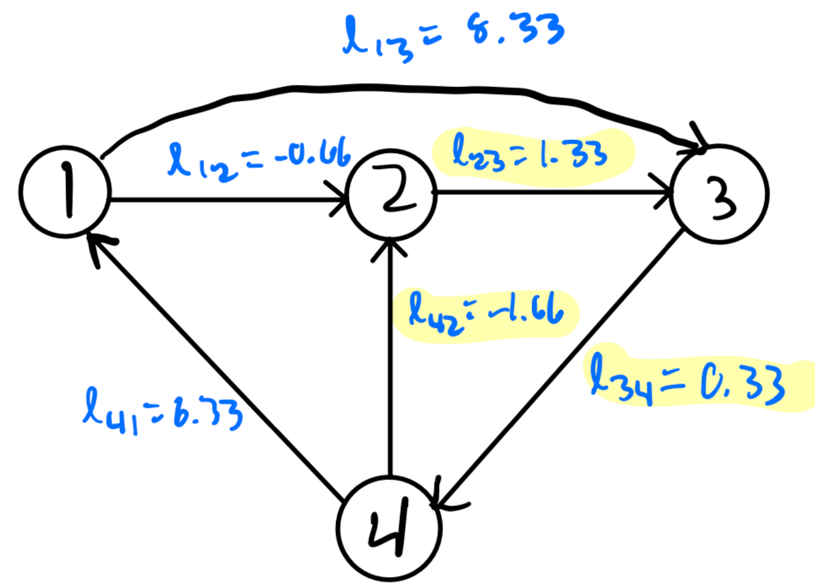




cycle: $\frac{-25}{3}$, $\mu^1 = \frac{-25}{3}$

Example: $\mu^k = \frac{5}{3}$





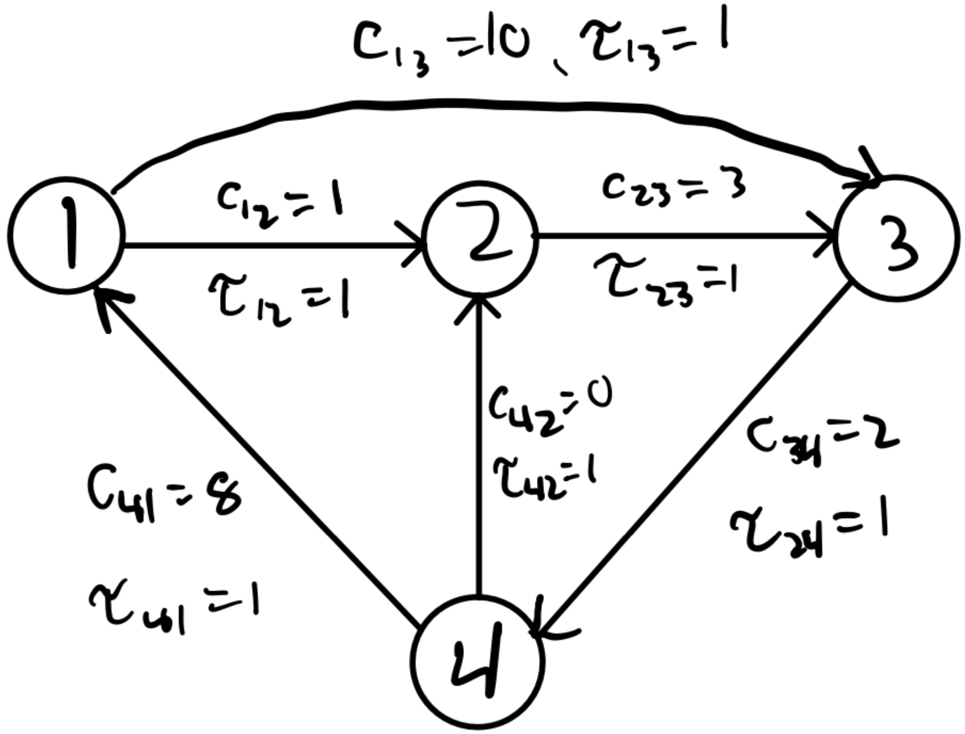
zero cycle, no negative cycles

$\mu^k = \mu^* = \frac{5}{3}$

Binary Search

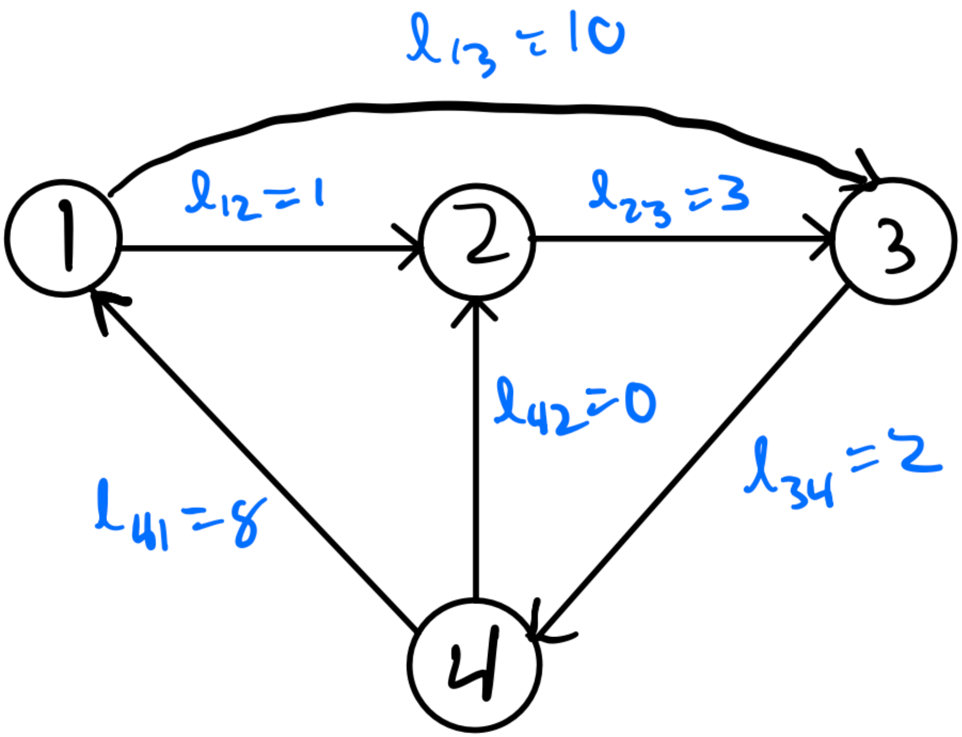
- Start with interval $[\underline{\mu}, \bar{\mu}]$ that contains μ^*
- Let $\mu^0 = \left(\frac{\bar{\mu}+\underline{\mu}}{2}\right)$
- Compute $l_{ij} = c_{ij} - \mu^0\tau_{ij}$ and check for cycles
- If a negative cycle exists, μ^0 is a strict upper bound on μ^* and set interval to $[\underline{\mu}, \mu^0]$
- If no negative or zero cycles exist, μ^0 is a strict lower bound on μ^* and set interval to $[\mu^0, \bar{\mu}]$
- Repeat until interval contains only one potential value
- Runs in $O(\log(\tau_o C))$, where τ_o is the largest traversal time and C is the largest arc cost in G.

Example



- Here, $[\underline{\mu}, \overline{\mu}]$ = [-10, 10]
- $\left(\frac{\overline{\mu}+\underline{\mu}}{2}\right) = \mu^0 = 0$

Example



No negative nor zero cycles exist, so we know 0 is a strict lower bound on μ^*

We now set $[\underline{\mu}, \overline{\mu}]$ = [0, 10] and continue running

Special Case: Minimum Mean Cycle Problem

- Special case of minimum cost-to-time ratio cycle problem
- Assume $\tau_{ij} = 1$ for all arcs $(i, j) \in A$
- Assume graph is strongly connected
  - If not, we add extra arcs with sufficiently large cost where they're missing
  - These new arcs will not be used in the min mean cycle
- Minimize $\frac{\sum_{(i,j)\in W} c_{ij}}{|W|}$

Theorem 5.8

- Let $d^k(j)$ denote the shortest path from a node s to a node j using exactly k arcs.
- We start by getting these shortest path distances for all nodes in N. We then claim:

$$\mu^* = \min_{j \in N} \max_{1 \le k \le n-1} \left[\frac{d^n(j) - d^k(j)}{n-k} \right]$$

- We will prove this in two cases: $\mu^* = 0$ and $\mu^* \neq 0$

Case 1: $\mu^* = 0$

- We have no negative cycles, but we do have a zero cycle W^*
- Compute the shortest path distances from node s to each node j. Call this distance $d(j)$
- Calculate the reduced arc costs with ${c_{ij}}^d = c_{ij} + d(i) - \ d(j)$
- All arcs are now nonnegative integers
  - Any arc on shortest path or W^* will be zero
- Compute $\overline{d}(j)$, the new shortest path distance using the reduced cost arcs

Case 1: $\mu^* = 0$ continued

- We now get that $\max_{1 \le k \le n-1} \left[\overline{d}^{n}(j) - \overline{d}^{k}(j) \right] \ge 0$ for all nodes j.

- Now, for a shortest path from s to j, the length will be 0 and it will have $1 \le k \le n-1$ arcs

- We will extend this walk to contain n arcs, continuing to walk along the cycle W^*

- Call the node we end up at p

Case 1: $\mu^* = 0$ continued

- We know the walk to node p has cost zero, and any arc along the cycle W^* has cost zero, so the walk from s to p has total cost zero

- We also have a shortest path from s to p that has total cost zero

- So, $d^n(p) = d^k(p) = 0$

- Thus, $\mu^* = \max_{1 \le k \le n-1} \left[\frac{d^n(p) - d^k(p)}{n-k} \right] = 0$, as desired

Case 2: $\mu^* \neq 0$

- Choose a number Δ and reduce the cost of all arcs by Δ
- $d^k(j)$ is reduced by $\mathrm{k} * \Delta$
- μ^* is reduced by Δ, and so is the ratio $\frac{d^n(j) - d^k(j)}{n-k}$
- If we chose Δ well, $\mu^* = 0$ and we can apply case 1

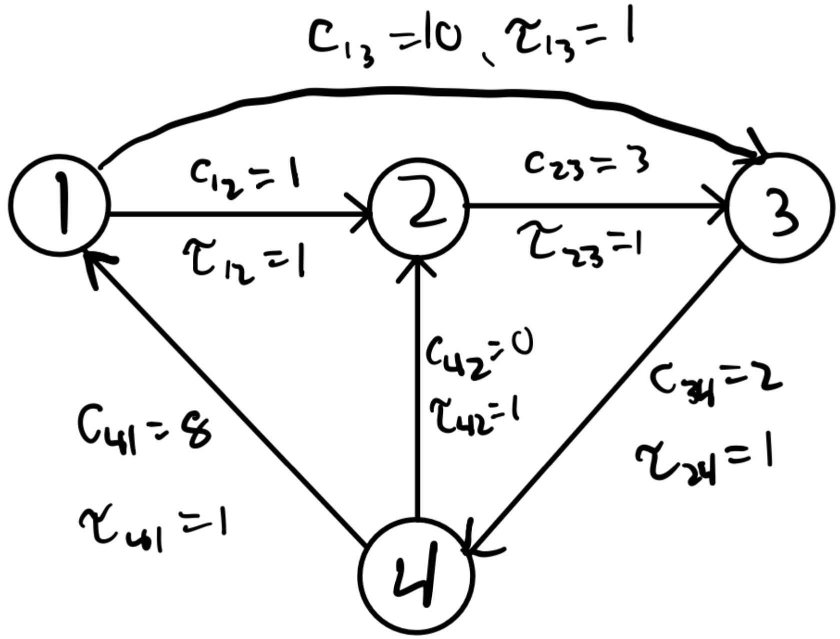
Problem 5.55: prove that the reduced cost shortest path distances can be used to get the min mean cycle

- We are given $d^k(j)$ for all nodes in N and for all $1 \leq k \leq n - 1$
  - Can be found in $O(nm)$ time using a recursive formula

- We can just calculate $\min_{j \in N} \max_{1 \leq k \leq n-1} \left[\frac{d^n(j) - d^k(j)}{n-k}\right]$ from the table
  - This takes $O(n^2)$ time

- This is known as "Karp's Algorithm", and in total, it takes $O(nm + n^2)$ time

Example (same as before)



Node	$d^0(j)$	$d^1(j)$	$d^2(j)$	$d^3(j)$	$d^4(j)$
1 (s)	0	∞	∞	20	16
2	∞	1	∞	12	6
3	∞	10	4	∞	15
4	∞	∞	12	6	∞

Example (same as before)

- We know from before that $\mu^* = \frac{5}{3}$

Node	$\boldsymbol{d^0(j)}$	$\boldsymbol{d^1(j)}$	$\boldsymbol{d^2(j)}$	$\boldsymbol{d^3(j)}$	$\boldsymbol{d^4(j)}$	$\max_{1 \le k \le n-1} \left[\frac{d^n(j) - d^k(j)}{n-k} \right]$
1 (s)	0	∞	∞	20	16	$\frac{14-0}{4-0} = \frac{7}{2}$
2	∞	1	∞	12	6	$\frac{6-1}{4-1} = \frac{5}{3}$
3	∞	10	4	∞	15	$\frac{15-4}{4-2} = \frac{11}{2}$
4	∞	∞	12	6	∞	∞

Summary

- There are many ways to find the value of a Minimum Cost-to-Time Ratio Cycle
- Sequential Search runs in pseudo-polynomial time
- Binary Search runs in $O(\log(\tau_o C))$
- Karp's Algorithm runs in $O(nm + n^2)$, but can only be used for the special Minimum Mean Cycle problem

References

- Ravinda K. Ahuja, Thomas L. Magnanti, James B. Orlin. Network Flows. 1993.
- Karp, R. M. (1978). A characterization of the minimum cycle mean in a digraph. In Discrete Mathematics (Vol. 23, Issue 3, pp. 309–311). Elsevier BV. https://doi.org/10.1016/0012-365x(78)90011-0