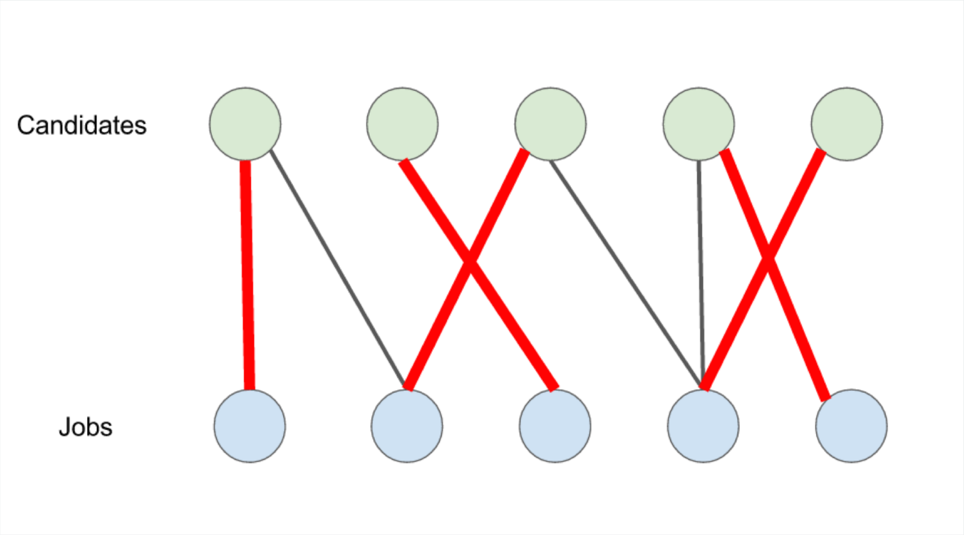
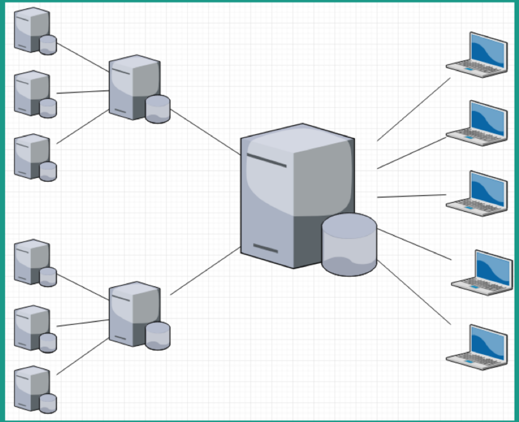
Online Min Cost Matching Overview

Koosha Jaferian
Winter 2020
Instructor: Prof. Allan Borodin



General Problem's Definition

- Complete bipartite graph $G = (R \cup S, E)$
- Positive edge cost c_e for each $e \in E$
- R = requests, and S = servers: $|R| \leq |S|$
- Goal: Match all the requests to a server with the minimum cost.
- $c_M = \sum_{e \in M} c_e$



Polynomial Time Algorithm

- Hungarian Method (Kuhn - Mawr 1955)
- Uses Adjacency Matrix of the graph.
- Makes the smallest number in each row and column zero.
- Draw the fewest number of lines to cover all zeroes, make the smallest numbers covered by them zero.
- Choose 3 zeroes not in the same column or row, and restore values!

Gives $O(|V|^3)$

108	125	150
150	135	175
122	148	250

0	17	42
15	0	40
0	26	128

0	17	2
15	0	0
0	26	88

0	17	2
15	0	0
0	26	88

-2	15	0
15	0	0
-2	24	86

0	15	0
17	0	0
0	24	86

0	15	0
17	0	0
0	24	86

0	15	0
17	0	0
0	24	86

108	125	150
150	135	175
122	148	250

Online Instance Definition

- Requests are unknown and they arrive one by one.
- They had to be matched to an unmatched server, and the decisions are irrevocable.

★ For arbitrary edge costs no optimal assignment in online instance.

★ α-competitive algorithm: $c(M) \leq \alpha \times OPT$

No α-competitive algorithm!

- Two requests
- First request's cost is 1 for both servers.
- Second request's cost is 1 for the first server and ∞ for the second.

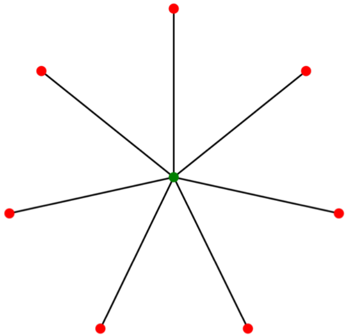
- We don't know anything about the second request when we serve the first.

- Therefore, the competitive ratio is unbounded!
- Same even for randomized algorithms!

First Relaxation: Metric

- We suppose that the triangle inequality holds for all the vertices, and costs.
- Kalyanasundaram and Pruhs (1993) and independently Khuller, Mitchell, and Vazirani (1994)
- $2n - 1$ competitive algorithm ($2n$ is the number of nodes)

- Gradually building Minimum partial matchings, and then using ⊕ operator and permutation.
- No competitive ratio better than $2n - 1$



Second Relaxation: Matching on the line

- Edge costs are induced by line metrics.
  - TSP Tour: the shortest path for a person to take to visit a list of destinations.
  - Diameter: Longest "shortest path" between any two vertices.
  - μ (G) = maximum ratio of minimum TSP tour / Diameter

- Raghvendra (2016): Competitive ratio of $O(\mu(G) log^2(n))$ for general problem.
- Linear Graph → μ (G) = 2 ⇒ $O(log^2(n))$
- Raghvendra (2018): Improved to $\theta(log(n))$: Approximation of the nearest free server

Big Open Problem

- Is there a constant-factor or O(1) Competitive algorithm for min cost matching on the line?

More Relaxations to achieve constant factor

- Able to change or revoke small number of decisions made.
- Match each request k number of times.

- (Megow, Nolke 2020): (1 + ε)-competitive if we can reassign any request at most $O(\epsilon^{-1.001})$

- Gupta et al (2019): $O(log(log(log(n)))^2)$- competitive factor for online known i.i.d arrivals.

Other related problems: k-server problem

- K-server problem.
- We have k servers on the line, or a metric space. The requests will come one by one, and we should move the servers on the line to get to the request.
- K = 2 we will have 2-competitive algorithm.
- k - competitive algorithm for all metric spaces with k+1 points.

★ Conjecture 1: For every metric space with more than k points, the competitive ratio of k-server problem is exactly k. Current positive result 2k-1

★ Conjecture 2: For every metric space, there is a randomized algorithm which gives the competitive ratio of $O(log(k))$

  - More evidence: Computer analysis for k = 3 and prove for the line and tree. (Chrobak 1991)

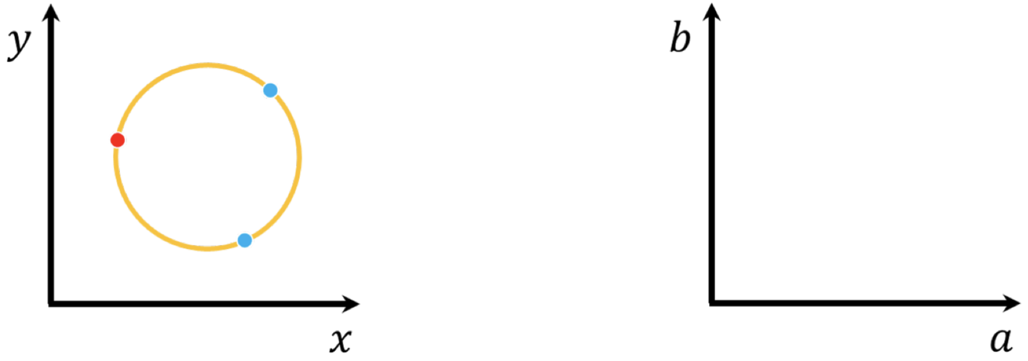
Worst Case (Adversary) vs. Average Case (Random order of arrivals)

- These analysis were for the worst case.
  - Adversary will give you some order to make you perform as bad as you can.

- Some analysis are for completely random orders for max weight matching.
  - (Kesselheim et al. 2013): (1/e - competitive) deterministic algorithm.

So much room for future work

- Many instances!
  - Line
  - Circle
  - Tree
  - Two-dimensional space
  - Three-dimensional space
  - Metric
  - General



- Many Open problems and conjectures.

OPENCONJECTURE

ML Advice Relaxation

- A prediction or advice derived from Machine Learning algorithms.

- $p^* = (p_1^*, \ldots, p_m^*)$ Vector of predictions (m = number of requests)
- Predicts edge weight adjacent to $r \in R$
- There is an optimal bipartite matching that r is adjacent to an edge with weight p_r^*

- Threshold greedy methods → There is a polynomial time algorithm that for some constants $0 < \alpha, \beta < 1$ so that it is α-competitive with α > 1/e if the error is small enough, and ß-competitive with ß < 1/e regardless of the error.

Different types of ML Advice

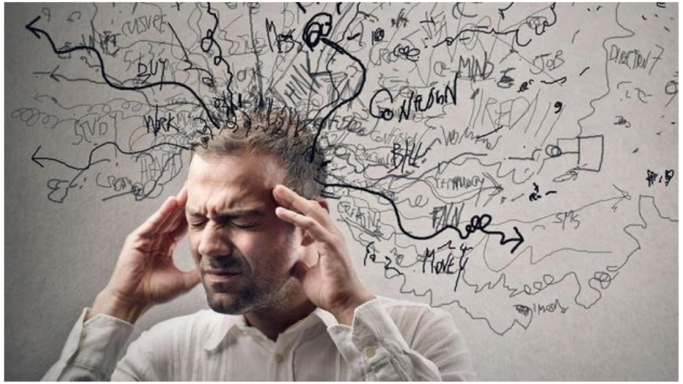
- Deterministic (Trusted)
  - We know the maximum possible difference between the advice and the optimal solution.

- Randomized (Untrusted)
  - With some probability the advice will not be within the range of the optimal solution.
    - Harder to derive results.
    - Most ML Algorithms give Untrusted advice to some extent.

So Much to Work on ...

- Different Types of spaces
  - Metric
  - Tree
  - Line
- What is online?
  - Requests (nodes) arrive online with all their adjacent edges.
  - Edges arrive online one by one.

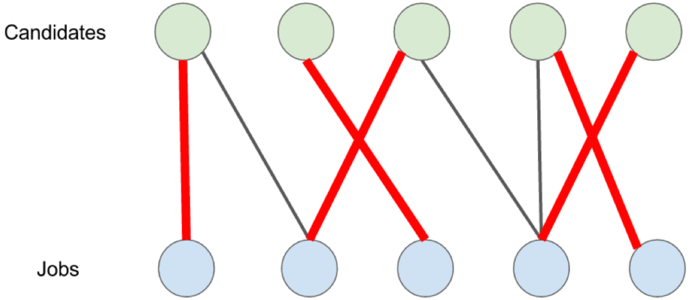
➔ Do we have any advice?
  - ◆ Untrusted
  - ◆ Trusted
- Different instances?
  - K - server problem
  - etc.



Applications (Matching in General)

- Servers and requests in networking
- Stable marriage problem
- Scheduling and planning
- Chemistry (Modeling Bonds)
- Networks and Packet switches
- Graph Coloring
- Neural nets
- etc.

Conclusion



Any Questions?

Thanks for Your attention!

References

- https://onlinelibrary.wiley.com/doi/epdf/10.1002/nav.3800020109
- https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.54.2873&rep=rep1&type=pdf
- https://www.sciencedirect.com/science/article/pii/0304397594900426
- https://drops.dagstuhl.de/opus/volltexte/2016/6641/pdf/LIPIcs-APPROX-RANDOM-2016-18.pdf
- http://www.cs.ox.ac.uk/people/elias.koutsoupias/Personal/Papers/paper-kou09.pdf
- https://arxiv.org/pdf/2006.01026.pdf
- https://arxiv.org/pdf/1803.07206.pdf
- http://www.thomas-kesselheim.de/science/onlinematching.pdf
- https://www.uncg.edu/cmp/faculty/srtate/papers/cowpath.pdf
- https://link.springer.com/chapter/10.1007/3-540-60692-0_38