Technical Perspective:
Bipartite Matching: What to do in the Real World When Computing Assignment Costs Dominates Finding the Optimal Assignment

Nikos Mamoulis
University of Ioannina, Greece
nikos@cs.uoi.gr

The optimal assignment problem is a classic combinatorial optimization problem. Given a set of n agents A, a set T of m tasks, and an $n \times m$ *cost matrix* $\mathbf{C}$, the objective is to find the *matching* between A and T, which minimizes or maximizes an aggregate cost of the assigned agent-task pairs. In its standard definition, $n = m$ and we are looking for the 1-to-1 matching with the minimum total cost. From a graph theory perspective, this is a weighted bipartite graph matching problem. A classic algorithm for solving the assignment problem is the Hungarian algorithm (a.k.a. Kuhn–Munkres algorithm) [3], which bears a $O(n^3)$ computational cost (assuming that $n = m$); this is the best run-time of any strongly polynomial algorithm for this problem. There are many variants of the assignment problem, which differ in the optimization objective (i.e., minimize/maximize an aggregate cost, achieve a stable matching, maximize the number of agents matched which their top preferences, etc.) and in whether there are constraints on the number of matches for each agent or task.

Assignment problems have gained interest recently, due to the advent of applications, such as ride-hailing services, that demand fast assignment decisions. For example, given a set of available drivers and a set of passengers that request a ride, the objective is to find the matching of drivers to passengers that minimizes the overall wait time (or the maximum wait time of any passenger) [1]. Other examples include assigning mobile devices to wireless access points or cars to parking spaces based on spatial distance and capacity [4], and assigning reviewers to papers [2]. In most applications, it is assumed that the matrix $\mathbf{C}$ is readily available before running the algorithm, or that each element c_{ij} of $\mathbf{C}$ can be computed very fast (e.g., c_{ij} is Euclidean distance in [4]); hence, the bottleneck is in the assignment algorithm.

However, as the following paper unveils, there are problem instances, where the elements of the cost matrix are unknown and expensive to compute. This is especially true in real-world applications, such as ride-hailing services, where an assignment needs to be computed on-demand for dynamically generated agents and tasks in real-time. The service may have to assign thousands of available drivers to requesting passengers within limited time. As the paper shows, computing the shortest path distances between all pairs of drivers and passengers (i.e., matrix $\mathbf{C}$) well exceeds the cost of finding the optimal assignment using $\mathbf{C}$.

As also observed in previous work [4], it is not necessary to compute the assignment costs between all drivers and passengers (i.e., the entire $\mathbf{C}$), since the pairs in the optimal matching tend to be spatially close to each other. Based on this fact, the paper proposes an extension of the Hungarian algorithm that computes the matrix elements in order of their likelihood to be part of the optimal matching. In particular, a set of vertices, called landmarks, are selected in the road network graph and the distances from all other vertices to them are precomputed. The extended assignment algorithm uses the precomputed distances to landmarks and the triangle inequality, to compute effective lower bounds for the cost matrix elements. In addition, the authors propose two refinement rules, which determine when it is necessary to compute the exact matching cost of a pair, during the execution of the assignment algorithm.

The experimental results exhibit an impressive performance for the proposed algorithm. It can find the assignment between thousands of drivers and passengers by only computing less than 5% of the matrix elements and it can achieve a throughput of about 1500 assigned pairs every 15 seconds for the Grab ride-hailing service in the city of Singapore. The strong and generalizable results of this paper open the road for additional work on extending assignment algorithms with different objectives (e.g., stable matching), for problems that involve hard-to-compute pairing costs and need be solved in real time.

1. REFERENCES

[1] G. Gao, M. Xiao, and Z. Zhao. Optimal multi-taxi dispatch for mobile taxi-hailing systems. In *ICPP*, pages 294–303. IEEE Computer Society, 2016.

[2] N. M. Kou, L. H. U, N. Mamoulis, and Z. Gong. Weighted coverage based reviewer assignment. In *ACM SIGMOD*, pages 2031–2046. ACM, 2015.

[3] H. Kuhn. The Hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[4] L. H. U, K. Mouratidis, M. L. Yiu, and N. Mamoulis. Optimal matching between spatial datasets under capacity constraints. *ACM Trans. Database Syst.*, 35(2):9:1–9:44, 2010.