Auction algorithm

Zachary Campbell

February 28, 2018

We now look at a cool application of weighted bipartite matching. Consider an auction scenario, where there is a set B of bidders, a set G of goods, and costs c_{ij} that quantifies the amount that bidder $i \in B$ is willing to pay for good $j \in G$. This can easily be modeled with a weighted bipartite graph. Our goal is to maximize the total amount earned in the auction – i.e. to maximize $\sum_{(i,j)} c_{ij}$, under the constraint that no bidder gets more than one good, and no good is purchased by more than one bidder.
The algorithm presented here is an alternative to the alternating path algorithms we have discussed.

Alg 1

For each $j \in G$, set $p_j := 0$ and $owner_j := null$.
Queue $Q := B$.
Set $\delta = 1/(|G| + 1)$.
while $Q \neq \emptyset$
    $i = Q.deque()$
    Find $j \in G$ that maximizes $w_{ij} - p_j$
    if $w_{ij} - p_j \geq 0$
        $Q.enqueue(owner_j)$
        $owner_j = i$
        $p_j = p_j + \delta$
return $(owner_j, j)$ for all j.