Jan 16 Notes

1 Continuing with the Budget Allocation Problem

1.1 Algorithm

Let $g(x) = e^{x-1}$ and $y_i = \sum_j \frac{b_{ij} x_{ij}}{B_i}$.
When node $j \in R$ "arrives" repetitively match dx of j to $\text{argmax}_{i:i\sim j} \{b_{ij}(1 - g(y_i))\}$ until either
1. $\sum_i x_{ij} = 1$, i.e., node j is exhausted or
2. $\min_{i:i\sim j}\{y_i\} = 1$, i.e., the budgets of nodes connected to j are exhausted.

1.2 Charging Policy

For each dx of j which is matched to i then increment α_i by $b_{ij} g(y_i) dx$ and β_j by $b_{ij}(1 - g(y_i))\, dx$.

1.3 Analysis of Competitive Ratio

The change in y_i, when dx arrives along edge $j \to i$, is $dy_i = \frac{b_{ij} dx}{B_i}$, or equivalently

$$b_{ij} dx = B_i dy_i.$$

Consequently, the change in α_i is

$$d\alpha_i = b_{ij} g(y_i) dx = B_i g(y_i) dy_i.$$

From the analysis of the the water level algorithm we have that $\int_0^{y_i^f} g(y_i) dy_i = g(y_i^f) - \frac{1}{e}$. The final α_i value is thus

$$\alpha_i \quad = \quad \int_0^{y_i^f} B_i g(y_i) dy_i = B_i \int_0^{y_i^f} g(y_i) dy_i = B_i \left(g(y_i^f) - \frac{1}{e} \right).$$

At the end of the algorithm if node i has exhausted its budget, i.e., $y_i^f = 1$ then $\alpha_i = B_i \left(g(1) - \frac{1}{e} \right) = B_i \left(1 - \frac{1}{e} \right)$. The alternative is that there is still budget remaining at node i and therefore $y_i^f < 1$. In this case, an adjacent node j will have $\beta_j \geq b_{ij} \left(1 - g(y_i^f) \right)$, as j will always assign to a node i', say, with $y_{i'} \leq y_i^f$. Consider the optimal set of edge assignment x_{ij}^*. For those edges adjacent to i, the value gathered from these edges is $\sum_j b_{ij} x_{ij}^*$, which is at most B_i, due to the budget constraint on i. The value gathered from the proposed algorithm is α_i from node i and a selected contribution of $\sum_j \beta_j x_{ij}^*$ from the nodes adjacent to i. From the bounds on α_i and β_j above,

$$\begin{aligned}
\alpha_i + \sum_j \beta_j x_{ij}^* \quad &\geq \quad B_i \left(g(y_i^f) - \frac{1}{e} \right) + \sum_j b_{ij} x_{ij}^* \left(1 - g(y_i^f) \right) \\
&\geq \quad \sum_j b_{ij} x_{ij}^* \left(g(y_i^f) - \frac{1}{e} \right) + \sum_j b_{ij} x_{ij}^* \left(1 - g(y_i^f) \right) \text{ (from the constraints } \sum_j b_{ij} x_{ij}^* \leq B_i) \\
&\geq \quad \sum_j b_{ij} x_{ij}^* \left(g(y_i^f) - \frac{1}{e} + 1 - g(y_i^f) \right) \\
&\geq \quad \sum_j b_{ij} x_{ij}^* \left(1 - \frac{1}{e} \right).
\end{aligned}$$

Therefore extending the bounds over all nodes i, the competitive ratio is at least $1 - \frac{1}{e}$.
Exercise: *Convert the above analysis of the competitive ratio to a primal-dual proof.*

2 Ranking Algorithm

2.1 Algorithm

Consider a new online matching problem where a permutation $\pi : [n] \to [n]$ is picked uniformly at random, where n is the number of nodes on the left. When j on the right arrives match it to

$$\text{argmin}_{i:i\sim j} \{\pi(i) : i \text{ is not yet matched}\}.$$

An equivalent description is for each node i on the left to pick a $Y_i \sim [0,1]$, uniformly at random. When j on the right arrives match it to

$$\text{argmin}_{i:i\sim j} \{Y_i : i \text{ is not yet matched}\}.$$

We consider the Y_i's as "ranking" the nodes on the left from 1 to n, i.e., $\forall i, i'$ if $Y_i \leq Y_{i'}$ then $M(i) \leq M(i')$, where $M : [n] \to [n]$ is the "rank".

2.2 Charging Policy

For each j if j is matched to i then set $\alpha_i = g(Y_i)$ and $\beta_j = 1 - g(Y_i)$.

2.3 Analysis of Competitive Ratio

Fix the randomly assigned value of all nodes $i' \neq i$, i.e., all $Y_{i'}$ is fixed for $i' \neq i$. Consider the effect of the matching on i for Y_i taken from 0 to 1. As Y_i increases from 0 to 1 then its rank against the other nodes will increase. Let y_i^f be the maximum Y_i value such that any larger Y_i doesn't change i's rank, and therefore its match. Thus, the expected value of α_i over all Y_i is

$$\mathbb{E}_{y_i}[\alpha_i] = \int_0^{y_i^f} g(y)dy = g(y_i^f) - \frac{1}{e}.$$

If j is matched to node i when $Y_i = y_i^f$ then decreasing Y_i will serve to improve the rank of i (decrease $M(i)$). Consequently, j may be matched to say i' which was previously higher in the ranking ($M(i') \leq M(i)$) with a $Y_{i'} \leq y_i^f$. Therefore, β_j can only get larger i.e., $\beta_j \geq 1 - g(y_i^f)$.
It follows that $\mathbb{E}_{y_i}[\alpha_i + \beta_j] \geq 1 - \frac{1}{e}$ and the competitive ratio is at least $1 - \frac{1}{e}$.

3 Vertex Weight Bipartite Matching Problem

Consider an adaptation of the ranking algorithm problem by assigned a matching profit of $v_i \in \mathbb{R}_+$, when a node i on the left is matched. The objective of the primal problem is consequently

$$\max \sum_{\forall \text{i matched}} v_i.$$

3.1 Charging Policy

For each j if j is matched to i then set $\alpha_i = g(Y_i)v_i$ and $\beta_j = (1 - g(Y_i))\, v_i$.

Exercise: *Extend the algorithm and analysis of the ranking algorithm to the vertex weighted bipartite matching problem.*