Algorithms and Uncertainty, Winter 2017/18 Lecture 1 (4 pages)

Online Bipartite Matching

Instructor: Thomas Kesselheim

One topic of this class will by online algorithms. An online algorithm has to solve an optimization task that it is revealed to it only over time. The difficulty is that it has to make decisions before it has seen the entire instance.

1 Greedy Algorithm

As our first online problem, we will consider online maximum bipartite matching. Consider a bipartite graph $G = (L \cup R, E)$. There are no edges within L or within R. A matching M is a subset of the edges E such that for each vertex at most one incident edge is included. Let OPT be a maximum matching, that is, a matching that contains the maximum number of edges. There are algorithms to compute OPT is polynomial time. For example, there is an easy reduction to the maximum-flow problem.

In the online variant of the problem, we do not know the graph from the start but it is revealed to us over time as follows. We know the set L from the start. Now, in each round one vertex r from R is revealed to us including its incident edges. We now have to decide immediately and irrevocably if we want to select one of these edges. If we decide to leave r unmatched, we cannot match it later. If we decide to match it to some $\ell \in L$, then we can neither match any other $r' \in R$ to ℓ nor match r to some other $\ell' \in L$ later on.

Let us consider the following simple greedy algorithm for this problem. Index the vertices in L arbitrarily from 1 to $|L|$. Whenever a vertex $r \in R$ is revealed, if there is still an unmatched neighbor, match it to the unmatched neighbor of smallest index.

Theorem 1.1. *Consider any graph bipartite graph G and any arrival order of R. Let ALG be the set of edges chosen by the greedy algorithm. Then $|ALG| \geq \frac{1}{2}|OPT|$.*

Proof. For $\ell \in L$, $r \in R$, we set

$$\alpha_\ell = \begin{cases} 1 & \text{if } \ell \text{ is matched in ALG} \\ 0 & \text{otherwise} \end{cases} \qquad \beta_r = \begin{cases} 1 & \text{if } r \text{ is matched in ALG} \\ 0 & \text{otherwise} \end{cases}$$

Observe that by this definition $\sum_{\ell \in L} \alpha_\ell + \sum_{r \in R} \beta_r = 2|\text{ALG}|$. So if we can show that $|\text{OPT}| \leq \sum_{\ell \in L} \alpha_\ell + \sum_{r \in R} \beta_r$, we are done.

To this end, observe that for any $(\ell, r) \in E$, we have $\alpha_\ell + \beta_r \geq 1$. Suppose otherwise, that is, $\alpha_\ell = 0$ and $\beta_r = 0$. This means that $r \in R$ does not get matched in ALG although $\ell \in L$ would be available. This is a contradiction to the greedy property.

Now we have:

$$|\text{OPT}| \leq \sum_{(\ell,r) \in \text{OPT}} (\alpha_\ell + \beta_r) \leq \sum_{\ell \in L} \alpha_\ell + \sum_{r \in R} \beta_r \enspace ,$$

where in the last step we use that each ℓ and r is matched at most once in OPT. □

2 Online Competitive Analysis

What have we done so far? We have devised an algorithm that only works online and we compared its performance to what we could have done offline.

In an abstract way, the algorithm operates on an input sequence $\sigma = (\sigma_1, \sigma_2, \dots, \sigma_n)$. In the t-th step request σ_t arrives and we have to process it only knowing $\sigma_1, \sigma_2, \dots, \sigma_t$ but not $\sigma_{t+1}, \sigma_{t+2}, \dots, \sigma_n$.

We say that a deterministic algorithm for a maximization problem is c-competitive if

$$v(\mathrm{ALG}(\sigma)) \geq c \cdot v(\mathrm{OPT}(\sigma)) - b \quad \text{for any sequence } \sigma \enspace , \tag{1}$$

where $v(\mathrm{ALG}(\sigma))$ denotes the value that the algorithm achieves on sequence σ and $v(\mathrm{OPT}(\sigma))$ denotes the value of the optimal offline solution. That is, it is the value that one could have achieved on σ with perfect knowledge and unlimited computational power.

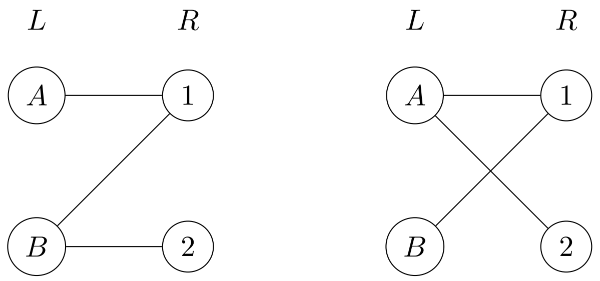
The constant $b \geq 0$ is useful to avoid corner cases. If Equation (1) holds with $b = 0$, then the algorithm is *strictly* c-competitive.

We can now restate Theorem by stating that the Greedy algorithm is strictly $\frac{1}{2}$-competitive. This is indeed the best guarantee that we can have for deterministic algorithms.

Theorem 1.2. *There is no deterministic algorithm for bipartite online matching that is strictly c-competitive for $c > \frac{1}{2}$.*

Proof. We will present the algorithm two different sequences σ and σ' for which both Equation (1) would have to hold. As $\sigma_1 = \sigma'_1$, the online algorithm cannot distinguish the two sequences in the first step and will make an error.

The two graphs look as follows:



If the algorithm decides to match vertex 1 to B, then it will perform poorly on the left graph. If it matches vertex 1 to A, then it will perform poorly on the right graph. If it does not match vertex 1 at all, this performs poorly on both graphs. $\square$

It is crucial for this perspective that the algorithm is deterministic. We will now turn to randomized algorithms.

We say that a randomized algorithm for a maximization problem is c-competitive if

$$\mathbf{E}\left[v(\mathrm{ALG}(\sigma))\right] \geq c \cdot v(\mathrm{OPT}(\sigma)) - b \quad \text{for any sequence } \sigma \enspace ,$$

where σ may not depend on the internal randomness of the algorithm.

3 Ranking Algorithm

We will now analyze a slight modification of the Greedy algorithm for which the guarantee will be better. This algorithm is called Ranking. It was introduced by Karp, Vazirani, and Vazirani in 1990. The analysis that we will consider was given by Devanur, Jain, and Kleinberg in 2013.

Ranking:

Index the vertices in L randomly from 1 to $|L|$. That is, draw one of the $n!$ permutations at random and indet L accordingly. Whenever a vertex $r \in R$ is revealed, if there is still an unmatched neighbor, match it to the unmatched neighbor of smallest index.

Theorem 1.3. Ranking *is strictly* $1 - \frac{1}{e}$*-competitive.*

Proof. We will use an alternative approach to define the indexing. For each $\ell \in L$, draw Y_ℓ independently uniformly from $[0, 1]$ and order L by increasing value of Y_ℓ. Observe that because the random variables $(Y_\ell)_{\ell \in L}$ are independent and identically distributed, each indexing now has the same probability.

We will follow the same approach as in the proof of Theorem 2. Which edges are selected now, of course, is random. Therefore also the values α_ℓ and β_r will be random variables now.

To define them, let us first fix arbitrary values of $(Y_\ell)_{\ell \in L}$. Whenever $\ell \in L$ is matched to $r \in R$, set

$$\alpha_\ell = g(Y_\ell)/F \quad \beta_r = (1 - g(Y_\ell))/F \ , \qquad \text{where } g(y) = \mathrm{e}^{y-1} \text{ and } F = 1 - \frac{1}{\mathrm{e}} \ .$$

If ℓ or r remain unmatched, set $\alpha_\ell = 0$ or $\beta_r = 0$ respectively.

Observe that for every $(\ell, r) \in \mathrm{ALG}$, we now have $\alpha_\ell + \beta_r = \frac{1}{F}$, that is $\sum_{\ell \in L} \alpha_\ell + \sum_{r \in R} \beta_r = |\mathrm{ALG}|/F$.

This hold *pointwise* for any random outcome. Therefore, we can take the expectation on both sides and get

$$\mathbf{E}\left[\sum_{\ell \in L} \alpha_\ell + \sum_{r \in R} \beta_r\right] = \mathbf{E}\left[|\mathrm{ALG}|/F\right] = \mathbf{E}\left[|\mathrm{ALG}|\right]/F \ .$$

Below, we will show that for any $(\ell, r) \in E$, we have $\mathbf{E}[\alpha_\ell] + \mathbf{E}[\beta_r] \geq 1$. This will then show

$$|\mathrm{OPT}| \leq \sum_{(\ell,r) \in \mathrm{OPT}} (\mathbf{E}[\alpha_\ell] + \mathbf{E}[\beta_r]) \leq \sum_{\ell \in L} \mathbf{E}[\alpha_\ell] + \sum_{r \in R} \mathbf{E}[\beta_r] = \mathbf{E}\left[\sum_{\ell \in L} \alpha_\ell + \sum_{r \in R} \beta_r\right]$$

and we are done. □

Lemma 1.4. *For each* $(\ell, r) \in E$*, we have* $\mathbf{E}[\alpha_\ell] + \mathbf{E}[\beta_r] \geq 1$*.*

Proof. We consider a fixed edge $(\ell, r) \in E$. The values of α_ℓ and β_r are determined by the outcomes of the random variables $(Y_{\ell'})_{\ell' \in L}$. We will keep all $Y_{\ell'}$ for $\ell' \neq \ell$ fixed to arbitrary values and only argue about the value of Y_ℓ. We can do this because the values are drawn independently.

Let us consider the execution of the algorithm on $G \setminus \{\ell\}$, that is, if we remove ℓ from the graph G. We define y^c as follows. If r gets matched in this execution to some ℓ', then set $y^c = Y_{\ell'}$. Otherwise, set $y^c = 1$.

Our first observation is that ℓ gets matched whenever $Y_\ell < y^c$. The reason is as follows. If $Y_\ell < y^c$, then by the time r arrives, ℓ could be already matched and we are done. Otherwise, all vertices from R up to this point are matched exactly the same way as if ℓ did not exist. Now, r would be matched to ℓ' if ℓ was not there as $Y_\ell < y^c = Y_{\ell'}$, r is matched to ℓ instead. So, again ℓ ends up being matched.

Consequently, we get

$$\mathbf{E}_{Y_\ell}[\alpha_\ell] = \int_0^1 g(y)/F \cdot \mathbf{1}_{\ell \text{ gets matched when } Y_\ell = y} dy = \int_0^{y^c} g(y)/F dy = \frac{1}{F}\left[\mathrm{e}^{y-1}\right]_{y=0}^{y^c} = \frac{1}{F}\left(\mathrm{e}^{y^c-1} - \mathrm{e}^{-1}\right) \ .$$

Now, let us turn to r. We claim that if $y^c < 1$ then r is always matched to some ℓ'' with $Y_{\ell''} \leq y^c$, even if vertex ℓ is around.

To this end, we compare the executions of the algorithm on G and on $G \setminus \{\ell\}$. More specifically, we compare which subset of the offline vertices L is still unmatched. Let $U_0, U_1, \ldots$ and $U'_0, U'_1, \ldots$ be the respective sets after the respective rounds. We claim that $U_t \supseteq U'_t$ for all t. Note that this proves our claim because it means that r can be matched in G if it is matched in $G \setminus \{\ell\}$ because this respective neighbor is free.

We apply a simple induction. For $t = 0$ this is trivial. Afterwards, the algorithm always takes the neighbor that comes first in the order by $(Y_{\ell'})_{\ell' \in L}$. If in both executions the algorithm uses the same ℓ', we are done because $U_{t+1} = U_t \setminus \{\ell'\} \supseteq U'_t \setminus \{\ell'\} = U'_{t+1}$. Otherwise, it uses in one execution an offline vertex that is contained in U_t but not in U'_t. So definitely $U_{t+1} = U_t \setminus \{\ell'\} \supseteq U'_t \supseteq U'_{t+1}$.

Consequently, irrespective of Y_ℓ, we have that $\beta_r \geq (1 - g(y^c))/F$

$$\mathbf{E}_{Y_\ell}\left[\beta_r\right] \geq \mathbf{E}_{Y_\ell}\left[(1 - g(y^c))/F\right] = (1 - g(y^c))/F = \frac{1}{F}(1 - \mathrm{e}^{y_c - 1}) \ .$$

In combination

$$\mathbf{E}_{Y_\ell}\left[\alpha_\ell\right] + \mathbf{E}_{Y_\ell}\left[\beta_r\right] \geq \frac{1}{F}\left(\mathrm{e}^{y^c - 1} - \mathrm{e}^{-1}\right) + \frac{1}{F}(1 - \mathrm{e}^{y_c - 1}) = \frac{1}{F}(1 - \mathrm{e}^{-1}) = 1 \ .$$

Recall that we kept $Y_{\ell'}$ fixed to arbitrary values for $\ell' \neq \ell$. We can now take the expectation over all these random variables on both sides and we are done. □

References

- R. Karp, U. Vazirani, and V. Vazirani, An Optimal Algorithm for On-line Bipartite Matching, STOC 1990 (original paper)
- N. Devanur, K. Jain, R. Kleinberg, Randomized Primal-Dual analysis of RANKING for Online BiPartite Matching, SODA 2013 (proof structure followed here)