Advanced topics in TCS

Exercise sheet 5. Weighted and unweighted matching

Christian Konrad

Question 1. Weighted Matching Algorithm from the Lecture

Give an example of an input stream on which the algorithm for weighted matching discussed in the lecture produces an approximation ratio close to $1/6$ (e.g., $< 1/5.9$ or similar). Such an example input stream demonstrates that our analysis is best possible.

Solution:

Figure 1 shows a hard instance that can easily be extended to yield an approximation ratio arbitrarily close to 6. In this example, a weight x^- means a value of $x - \epsilon$, for some arbitrarily small $\epsilon > 0$. Edges arrive in the following order: $1, 2^-, 2, 4^-, 4, 8^-, \dots, 64, 128^-, 128^-$. Observe that the algorithm outputs the edge with weight 64. The red edges form an optimal matching of weight 382. The algorithm therefore produces a $64/382 \approx 1/5.968$ approximation.

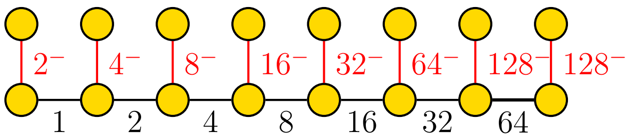

Figure 1: Hard example instance

Question 2. Greedy Matching on d-regular Graphs

We know that the Greedy matching algorithm has an approximation ratio of $1/2$ for the Maximum Matching problem on arbitrary graphs. Let $G = (V, E)$ be a d-regular graph, i.e., a graph where every vertex has degree d. Suppose we run Greedy on an arbitrarily ordered sequence of the edges of G. Give an improved bound (better than $1/2$) that depends on d on the approximation factor of Greedy when run on G. For example, the bound should yield $2/3$ for $d = 2$.

Solution:

Let M be the matching computed by Greedy, and let M^* denote a maximum matching. Whenever Greedy select an edge uv into M then at most $2d-1$ edges become ineligible for being added later on. This is because both vertices u and v are incident each to at most d edges, but they are both incident to the edge uv. Since every d-regular graph has $\frac{1}{2}dn$ edges, Greedy therefore selects at least

$$\frac{\frac{1}{2}dn}{2d-1}$$

edges into M, and since $|M^*| \leq \frac{1}{2}n$, we obtain the approximation factor:

$$\frac{|M|}{|M^*|} \geq \frac{\frac{\frac{1}{2}dn}{2d-1}}{\frac{1}{2}n} = \frac{d}{2d-1} > \frac{1}{2} \ .$$

Question 3. Weighted Matching with Restricted Edge Weights

Let $G = (V, E, w)$ be a weighted graph with $w : E \rightarrow \{1, 2\}$. Consider the following two algorithms, which can be implemented as semi-streaming algorithms, for computing matchings:

$\mathbf{A}_1$: Ignore the edge weights and use the Greedy matching algorithm to compute a maximal matching M. Return M with its edge weights.

$\mathbf{A}_2$: Run Greedy on the subgraph of edges of weight 1, which produces a matching M_1. In parallel, run Greedy on the subgraph of edges of weight 2, which produces a matching M_2. The output matching M is obtained by inserting every edge of M_1 into M_2 if possible.

1. What is the approximation guarantee of $\mathbf{A}_1$? Give a complete proof. Give a worst-case example that shows that your analysis is tight.

   Solution:

   Let M^* be a maximum matching in the input graph and M be the matching returned by $\mathbf{A}_1$. We know that Greedy has an approximation guarantee of $\frac{1}{2}$, so

$$|M| \geq \frac{1}{2}|M^*| \ .$$

Since each edge weight is in $\{1, 2\}$, we have:

$$w(M) \geq |M| \text{ , and}$$
$$|M^*| \geq \frac{1}{2} w(M^*) \text{ .}$$

Combining, we obtain:

$$w(M) \geq |M| \geq \frac{1}{2}|M^*| \geq \frac{1}{4} w(M^*) \text{ .}$$

See Figure 2 for a worst case example.

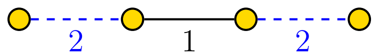

Figure 2: A worst case example of $\mathbf{A}_1$.

2. What is the approximation guarantee of $\mathbf{A}_2$? Give a complete proof. Give a worst-case example that shows that your analysis is tight.

Solution:

Let M^* be a maximum matching in the input graph. Let $M_1^* \subseteq M^*$ denote the subset of edges of weight 1, and let M_2^* denote the subset of edges of weight 2. For each edge $m \in M_2$, let $C(m)$ denote the set of at most 2 edges from M_1 that are incident to m. In other words, m is responsible for these at most two edges for not being added to the final output matching M.
Next, for any $i \in \{1, 2\}$, observe that since M_i is maximal, every edge $m \in M_i^*$ is either adjacent to an edge from M_i or is itself contained in M_i.
We will now charge the weights of the edges M^* to the edges in M, as follows:

- Let $m \in M_2^*$: We charge $w(m)$ to every edge that is incident to m in M_2. If there is no such edge, then $m \in M_2$, and we charge m by $w(m)$.
- Let $m \in M_1^*$: Let $N_1(m)$ denote the edges of M_1 that are incident to m, or, if there are no such edges (which implies $m \in M_1$), let $N_1(m) = m$. We now charge the weight $w(m)$ to every edge in $N_1(m)$. Then, if an edge in $N_1(m)$ is not included in the output matching, then we transfer its charge to the edge in M_2 that prevent it from being inserted.

Observe first that we inject at least $w(M^*)$ charge to the edges of the output matching. It remains to bound the maximum charge of an edge in M:

- Consider an edge $m \in M_1 \cap M$, i.e., m is included in the the output matching. Then m is charged at most $2w(m)$.
- Consider an edge $m \in M_2$. Then, $m \cup C(m)$ forms a path of length at most 3 and thus covers at most 4 vertices. This implies that $m \cup C(m)$

is incident to at most 4 edges from M^*. Since $m \cup C(m)$ contains only one edge from M_2 (i.e., the edge m), at most two of these 4 edges are from M_2^*. Hence, m has a charge of at most $2 \cdot 2 + 2 \cdot 1 = 3w(m)$ (since $w(m) = 2$).

Overall, an edge in the output matching receives a charge at most three times its own weight. Hence, $w(M^*) \leq \frac{1}{3}w(M)$. See Figure 3 for a worst case example.

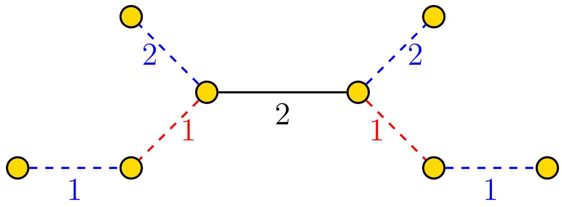

Figure 3: A worst case example of $\mathbf{A}_2$.