ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institute of Theoretical Computer Science
Bernd Gärtner, Rasmus Kyng, Angelika Steger, David Steurer, Emo Welzl

Algorithms, Probability, and Computing **Solutions for SPA 2** **HS22**

This is a draft version. After we have graded your assignments, we will publish an improved version based on the feedback that we have obtained from what you have written.

Solution 1

(a) Each distribution network is a feasible solution of the IP, as for each cut with d on one side, there must be an edge going over the cut (away from d), as otherwise the vertices on the other side of the cut would not be reachable from d. On the other hand, a feasible solution of the IP is a distribution network, as if some vertex v would not be reachable from d, there would be a $d - v$ cut with the value of the variables corresponding to the edges going over the cut all having value 0, as the maximum possible flow from d to v (with capacities set as the x_e's) would be zero.
As we have established that the set of feasible solutions is exactly the set of distribution networks, and the objective function is exactly the value of the corresponding distribution network, this equality also holds for the set of optimal solutions.

(b) We give a polynomial separation oracle, which given a proposed solution to the relaxed LP either determines that it is a valid solution, or finds a violated constraint. Obviously, whether a constraint $x_a \geq 0$ or $x_a \leq 1$ is violated can be determined in linear time. We thus assume these constraints all hold. We give each arc (u, v) the capacity $x_{(u,v)}$, and compute the maximum $d - w$ flow for all $w \in V$. If any of these flows is less than 1, there is a violated constraint. Given a maximum $d - w$ flow, a minimum $d - w$ cut C (thus of value < 1) can be found by taking all vertices reachable from d in the residual network. This cut is thus a violated constraint, which can be returned.
With this separation oracle, and the fact that the solution polytope is constrained to the hypercube $[0, 1]^A$, we can decide whether the LP is feasible using the Ellipsoid method in polynomial time. Furthermore, we can also perform a binary search on the optimization function to find an optimal solution (we only have to go as far until our confidence interval of the optimum value is smaller than the smallest edge weight, which takes polynomial time). With repeated calls of the Ellipsoid method, we can decide for each edge e whether there exists an optimal solution which has $x_e = 1$. If yes, this variable is from now on set to 1, otherwise it is set to 0. After $|E|$ oracle calls, all variables are set, and we have found an optimal integral basic feasible solution of the LP.

Solution 2

(a) The set of integral feasible solutions consists of variables which are either 0 or 1 (larger values are not possible, as otherwise an endpoint of e with $x_e > 1$ would already be over-saturated). Taking the edges with $x_e = 1$ must yield a matching, as every vertex can only be incident to at most one such edge, otherwise the constraint summing up the values of its incident edges

would be violated. Thus each integral solution corresponds to a matching.
The optimum integral solution is thus at least as good as the maximum matching. Furthermore, each matching is also an integral solution, thus the optimal integral solutions are exactly the maximum matchings.

(b) Let C be such a cycle. As the graph is bipartite, C consists of an even number of vertices and edges. We split the edges of C into the set E_1 and E_2, such that E_1 and E_2 are the two maximum matchings on C (thus E_1 and E_2 alternate along C). We take the value

$$\epsilon = \min(\min_{e \in E_1}(x_e), \min_{e \in E_2}(x_e), \min_{e \in E_1}(1 - x_e), \min_{e \in E_2}(1 - x_e)).$$

Then, we create two alternate solutions x^- and x^+. x^- is equal to x, but all variables belonging to edges in E_1 are reduced by ϵ, and all variables belonging to edges in E_2 are increased by ϵ. x^+ is obtained by doing the opposite.
We now check that x^- and x^+ are distinct feasible solutions. By the way we picked ϵ, $\epsilon > 0$, and the constraints $x_e^- \geq 0$ and $x_e^+ \geq 0$ can not be violated. As for each vertex on the cycle C, we increase one incident edge by ϵ and decrease one by ϵ, and the other vertices are not impacted, we have

$$\sum_{e \in d(v)} x_e = \sum_{e \in d(v)} x_e^- = \sum_{e \in d(v)} x_e^+ \leq 1.$$

Thus, x^- and x^+ are basic feasible solutions, and their average is clearly x, showing that x cannot be a basic feasible solution.

(c) We consider a path P between two leaves of F. Again, we split up the edges of P into E_1 and E_2 alternatingly along P. We define ϵ as in subtask (b). We also define x^+ and x^- the same way. Again, by the definition of ϵ, all non-negativity constraints still hold in both of these solutions. Actually, all x_e are also at most 1. The matching constraints hold for all internal vertices of the path, as the effects cancel out. We only need to consider the endpoints of the path. Let p be one endpoint of P. As p is a leaf of F, the edge e' of P was the only fractional incident edge. Thus, all other incident edges must be 0. Thus, $\sum_{e \in d(p)} x_e = x_{e'} \leq 1$, and all constraints hold for x^+ and x^-. We conclude that x is the average of two feasible solutions x^+ and x^-, and is thus not a BFS.

Solution 3

(a) The path on 2 vertices is the easiest example.

(b) The graphs that contain no nice cycles. If G contains no nice cycles, every orientation is trivially Pfaffian. On the other hand, if G contains at least one nice cycle, there exists at least one orientation which makes this cycle not oddly oriented (for example any orientation that orients the edges on the cycle all in the same direction), showing that the probability of Pfaffian is less than one.

(c) From (b) we know that a graph where this probability is strictly less than 1 would need to contain at least one nice cycle. Let e be an arbitrary edge of an arbitrary nice cycle C. Pair up all orientations which agree on all edges except e. In each such pair, exactly one of the two orientations makes C oddly oriented. Thus, at least one of every pair of orientations is not Pfaffian. Thus, the probability of being Pfaffian is at most 1/2, if it is not 1. We conclude that such a graph can not exist.

(d) We define the graph $G^{(k)}$ which consists of k disjoint 4-cycles. Each of these cycles is nice, as there is a perfect matching in the remaining graph after removing the cycle (which is $G^{(k-1)}$). The probability of each cycle being oddly oriented is $1/2$ (by a similar argument as in (c)), and these probabilities are independent as the cycles do not share an edge. We conclude that the probability of all nice cycles being oddly oriented, and thus the orientation being Pfaffian is $1/2^k$.

(e) If an edge e is not part of any perfect matching, adding it cannot create or destroy any nice cycles. Furthermore, it is not part of a nice cycle itself, thus its orientation does not matter for the definition of Pfaffian. Thus, any orientation of $G \setminus \{e\}$ is Pfaffian if and only if the same orientation of G (with e oriented arbitrarily) is Pfaffian.

(f) If $G \setminus \{e\}$ has no Pfaffian orientation, G clearly cannot have one either, as adding more even cycles only makes the condition of Pfaffian stronger. If $G \setminus \{e\}$ has a Pfaffian orientation, we can add e to that orientation in two ways, one of which way will make the even cycle containing e oddly oriented. This orientation is then also Pfaffian, as all other nice cycles were already oddly oriented, and the only possible new nice cycle is oddly oriented (whether or not it is nice is irrelevant).

Solution 4

(a) We look at A, the bipartite adjacency matrix of G with variable entries, as defined in the lecture notes. As mentioned in the proof of Lemma 5.2, each non-zero monomial in $\det(A)$ corresponds to a perfect matching. Thus, if we compute $\det(A) - sign(\sigma_M) \prod_{(a_i, b_j \in M)} x_{ij}$, the result is either the zero polynomial, in which case there is no other perfect matching, or it is non-zero, and there must be another perfect matching providing a non-zero monomial. We can test this by applying Schwartz-Zippel over $GF(p)$ for $2n < p < 4n$. For this, we need to be able to efficiently compute our modified polynomial, which we can do easily: We just compute $\det(A)$, and then subtract the fixed monomial (which can be evaluated in $O(n)$). Runtime and success probability follows from the analysis in chapter 5.3.

(b) We now look at the Tutte matrix T of G. We again subtract a monomial corresponding to our given matching M: The one which is given by the permutation σ which maps each i to its matching partner in M (thus always two elements point at each other, i.e., σ consists of only cycles of length 2). As in this case it no longer holds that each non-zero monomial of $\det(T)$ corresponds to a perfect matching, we need to analyze a bit more to figure out why the modified determinant polynomial is zero if and only if there is no other perfect matching.
For the first direction: If there is another perfect matching, that perfect matching contributes a non-zero monomial which is not cancelled out (just like the one we subtracted), thus making the modified determinant polynomial non-zero too.
For the other direction: We assume the modified determinant polynomial is non-zero. Then there is a non-zero monomial which is not cancelled out, and which thus consists of only even cycles. If it consists of only cycles of length 2, it corresponds to a matching, and we are done. Otherwise, we can find two perfect matchings in each even cycle, which can be combined in all possible ways, thus leading to 2^k perfect matchings for k being the number of even cycles of length ≥ 4. In both cases, we have found another perfect matching M'.
We conclude that we can find out whether there exists another perfect matching by using Schwartz-Zippel on the modified determinant polynomial, which can again be evaluated efficiently, as we only need to subtract one fixed monomial from the determinant.

(c) Here, we need to subtract multiple monomials: The two that correspond to M_1 and M_2, and the monomial that corresponds to the union of M_1 and M_2. This union must consist of only even cycles. In fact, it must consist of a single even cycle (apart from the cycles of length two), as otherwise we could find another matching directly, by flipping some but not all cycles of the union in M_1. The monomial of this union has to be subtracted twice, as it occurs twice in the determinant, due to the cycle being orientable in both directions. Any non-zero monomial left must now correspond to a union of one or more new matchings M', and similarly, as long as there is another matching M', it contributes a non-zero monomial.