CS602 Applied Algorithms **Jul-Nov 2023**

Homework (No submission)

Matching

1. Describe an algorithm that finds a walk alternating between matching and non-matching edges, starting with a given source vertex and ending with a given destination vertex. A walk is a sequence of vertices where any two consecutive vertices are connected by an edge and repeating vertices is allowed.

2. Consider the problem of finding maximum weight matching in bipartite graphs. Suppose we have a matching of size i that has maximum weight among all matchings of size i. Suppose we find a minimum weight augmenting path (where weight of an augmenting path is defined as weight of matching edges minus the weight of non-matching edges). Prove that after swapping the edges, we will get a matching of size $i+1$, which has maximum weight among all matchings of size $i+1$.

   Use this idea to design an algorithm for maximum weight matching. How will you find a maximum weight augmenting path (in the bipartite case)? Is it a concern that some edges have negative weights? There are algorithms for shortest path when there are no negative weight cycles. Can we guarantee this here?

   If we try to apply this idea in the non-bipartite case with Edmonds Blossom idea, where will it fail?

3. **(Difficult)** Consider an undirected graph with edges colored either red or blue. We want to find a path from a given source to a given destination that alternates between red and blue edges. You can also specify the colors of starting and ending edges, that does not change the problem much. By definition, a path does not have repeated vertices. Design a polynomial time algorithm. Either you can try to modify Edmonds' algorithm or you can try to reduce this problem to finding maximum matching in a graph.

4. Consider the above red-blue path problem in directed graphs. Prove that it is NP-hard. Some related NP-hard problems, which may be useful, are – longest path, hamiltonian path, shortest path with negative and positive edge weights.

Linear Programming and Integer Linear Programming

5. Consider the following integer linear program (ILP) for a graph $G(V, E)$.

$$\begin{aligned} \max \sum_{e\in E} x_e \text{ subject to} & \\ x_e \geq 0 \text{ for all } e \in E & \\ \sum_{e \text{ incident on } v} x_e \leq 1 \text{ for all } v \in V & \\ x_e \in \mathbb{Z} \text{ for all } e \in E. & \end{aligned}$$

   Prove that every feasible solution for the above constraints corresponds to a matching in G. Prove that every matching in G corresponds to an feasible solution for the above constraints.

6. Consider the following linear program (LP) for a graph $G(V, E)$.

$$\max \sum_{e \in E} x_e \text{ subject to}$$
$$x_e \geq 0 \text{ for all } e \in E$$
$$\sum_{e \text{ incident on } v} x_e \leq 1 \text{ for all } v \in V$$

Prove that if graph G is bipartite then the optimal value of this LP is equal to the maximum matching size.

Hint: If the optimal solution of this LP is integral, then it must correspond to a matching. Consider the case when an optimal solution is non-integral. Then argue that the edges where the optimal solution has fractional values must contain a cycle. Argue that the optimal solution can modified on this cycle edges, while maintaining feasibility of all constraints, such that the objective value increases.

7. Let S be a set of elements, where element $i \in S$ has weight w_i. Consider the problem of selecting an odd size subset that maximizes the total weight. Write an integer linear program for this problem, where the number of variables is $|S|$.

Hint: you will need exponentially many constraints, one for every even set.

August 11 Lecture 4

8. Prove that the set $\{x \in \mathbb{R}^n : Ax \leq b\}$ is a convex set, for any given $k \times n$ matrix A and $b \in \mathbb{R}^k$.

9. Suppose S is a convex set and we are maximizing a linear function $w^T x$ over it. If a point $x^* \in S$ locally maximizes the function, then it maximizes the function over all S. Locally maximizes means the following: there exists an $\epsilon > 0$ such that for all points $y \in S$ within distance ϵ from x^*, we have $w^T x^* \geq w^T y$. You will need to prove such an inequality for all points y in S.

10. Let a polyhedron P be given by k constraints

$$\{x \in \mathbb{R}^n : a_1^T x \leq b_1, a_2^T x \leq b_2, \ldots, a_k^T x \leq b_k.\}$$

Consider a linear function defined by

$$w = a_1 + a_2 + \cdots + a_\ell$$

for some $1 \leq \ell \leq k$. Then prove that $w^T x$ is maximized at a face that is defined by first ℓ constraints being equalities, that is,

$$\{x \in \mathbb{R}^n : a_1^T x = b_1, a_2^T x = b_2, \ldots, a_\ell^T x = b_\ell, a_{\ell+1}^T x \leq b_{\ell+1}, \ldots, a_k^T x \leq b_k\}.$$

August 18 Lecture 5

Notation: For a set of points S, $\text{conv}(S)$ denotes the convex hull of the points in S.

11. Let $p, q, r, s, t \in \mathbb{R}^n$ be points such that $s \in \text{conv}(p, q)$ and $t \in \text{conv}(r, s)$. Show that $t \in \text{conv}(p, q, r)$.

12. Let S be a set points in $\mathbb{R}^n$ and let $f(x) = w^T x$ be a linear function. Then

$$\max f(x) \text{ over } S = \max f(x) \text{ over } \text{conv}(S).$$

In fact, this also holds true for any convex function $f(x)$.

August 22 Lecture 6

13. Use Fourier Motzkin procedure to compute the linear inequalities in variables x_1, x_2 which describe the polytope $\text{conv}\{(0,0), (0,1), (1,0)\}$.

14. Consider the polytope described by

$$\begin{aligned} x_3 &\geq 0 \\ x_1 + x_3 &\geq -1 \\ x_3 - x_1 &\geq -1 \\ x_2 + x_3 &\leq 2 \\ -x_2 + x_3 &\leq 2. \end{aligned}$$

If we project this polytope on the (x_1, x_2)-plane, what will be the linear constraints describing the projection polytope.

15. We are given a system S of linear constraints in variables $y, x_1, x_2, \ldots, x_n$. We construct another system S' of linear constraints in variables $x_1, x_2, \ldots, x_n$ by eliminating variable y as described by Fourier Motzkin elimination. Prove that

$(\alpha_1, \alpha_2, \ldots, \alpha_n)$ is feasible for S'

if and only if

there exists $\beta \in \mathbb{R}$ such that $(\beta, \alpha_1, \alpha_2, \ldots, \alpha_n)$ is feasible for S.

16. Let P be the convex hull of points $q_1, q_2, \ldots, q_r \in \{0,1\}^n$. Show that each q_i is a corner of P.

17. For an LP with bounded optimal value, show that the optimal value is bounded exponentially in number of variables, number of constraints, and the bit size of coefficients in the constraints.

18. Suppose we have an optimization oracle that outputs the LP optimal value for a given feasible system of constraints and an objective function. If the given system is infeasible than it outputs an arbitrary answer. Using this optimization oracle, design an algorithm for the LP feasibility problem.

19. For any polytope described by a system of constraints, show that any of its corners can be described using only polynomially many bits (polynomial in number of variables, number of constraints, and the bit size of coefficients in the constraints).

    August 25 Lecture 7

20. Assume the form of Farkas' Lemma proved in the class

$$Ax = b, x \geq 0 \text{ is not feasible } \implies \exists y \text{ s.t. } y^T A \geq 0, y^T b = -1.$$

And prove the following forms of Farkas' Lemma.

- $Ax \leq b$ is not feasible $\implies \exists y \geq 0$ s.t. $y^T A = 0, y^T b = -1$.
- $Ax \leq b, x \geq 0$ is not feasible $\implies \exists y \geq 0$ s.t. $y^T A \geq 0, y^T b = -1$.
- $Ax \geq b$ is not feasible $\implies \exists y \geq 0$ s.t. $y^T A = 0, y^T b = 1$.

21. Prove that the cone generated by a set of given vectors $v_1, v_2, \ldots, v_k \in \mathbb{R}^n$

$$\{\lambda_1 v_1 + \lambda_2 v_2 + \cdots + \lambda_k v_k : \lambda_1, \lambda_2, \ldots, \lambda_k \geq 0\}$$

is convex.

    August 29 Lecture 8

22. Write the dual LP for the following linear program. Find the primal optimal solution and the dual optimal solution. What are the optimal values of the two LPs.

$$\begin{array}{rcl} \min x_1 - x_2 + 2x_3 & & \text{subject to} \\ x_1 + x_2 - x_3 & = & 1 \\ x_1 + x_3 & \geq & -1 \\ 2x_2 - x_3 & \leq & 2 \\ x_2 & \geq & 0 \end{array}$$

23. Below are given some LPs in various forms and their dual LPs. For each form,

(1) Prove that for any dual feasible solution y and any primal feasible solution x, we have $f(x) \leq g(y)$.

(2) Prove that if the primal LP has an optimal solution x^*, then the dual LP has an optimal solution y^* such that $f(x^*) = g(y^*)$.

Primal LP	**Dual LP**
$\max w^T x$ subject to $Ax = b$, $x \geq 0$	$\min b^T y$ subject to $A^T y \geq w$
$\max w^T x$ subject to $Ax \leq b$, $x \geq 0$	$\min b^T y$ subject to $A^T y \geq w$, $y \geq 0$

24. Prove that if the primal LP has unbounded optimum value then the dual LP is infeasible. Prove that if the dual LP has unbounded optimum then the primal LP is infeasible.

25. Find an example, where both primal and dual LPs are infeasible.

Sep 1 Lecture 9

26. Write an LP for the max flow problem (input: directed graph with edge capacities, a source and a sink). Write the dual LP for this. What combinatorial problem does the dual LP represent?

27. Write an LP for maximum size bipartite matching problem. Write the dual LP. Does the dual LP correspond to a natural combinatorial problem?

28. Primal dual approach can be used for a general LP. Let the primal LP be

$$\min w^T x \text{ subject to } Ax = b, x \geq 0,$$

and its dual LP be

$$\max b^T y \text{ subject to } A^T y \leq w.$$

Let β be a dual feasible solution. Suppose there is no primal feasible solution α for which α and β satisfy complementary slackness conditions. We want to argue that in this case there exists a way to improve the dual feasible solution.

Let the matrix A have entries $(a_{i,j})$. Complementary slackness condition would say that

$$\text{for } 1 \leq j \leq n, \sum_i a_{i,j} y_i < w_j \implies x_j = 0.$$

Without loss of generality let us say that

$$\text{for } 1 \leq j \leq n_1, \sum_i a_{i,j} \beta_i = w_j,$$

$$\text{and for } n_1 < j \leq n, \sum_i a_{i,j} \beta_i < w_j.$$

That means the primal feasible solution α we are looking for must satisfy

$$\text{for } n_1 < j \leq n, \alpha_j = 0$$

together with $A\alpha = b, \alpha \geq 0$. Suppose there is no such α. Use one of the forms of Farkas' Lemma and show that there is a dual feasible solution β' with $b^T \beta' > b^T \beta$.

Finding such a β' would mean solving a simpler linear program.

Linear Programming to solve combinatorial optimization problems

Sep 5 Lecture 10

29. Suppose a bipartite graph H has n vertices on both sides. Suppose L and R are two sets of nodes. Suppose there is no matching of size n in H. Then we want to compute a subset $S \subseteq L$ such that $|N(S)| < |S|$, that is, the number of neighbors of S is less than cardinality of S.

    Suppose M is a maximum matching in H, which can be computed using the augmenting path algorithm. Let $U \subseteq L$ be the set of unmatched vertices (in M) on the left side. Consider all the left side vertices which are reachable from U via alternating paths (where edges alternate between non-matching edge and matching edge). Does this give you S?

30. There is a better analysis for the number of rounds needed in the bipartite matching primal dual algorithm. Suppose you have computed a matching M (not necessarily perfect) among the tight edges in the last round. Let U be the set of unmatched vertices on the left side. Let V_ℓ and V_r be set of vertices on the left side and right side, respectively, that are reachable from U via alternating paths (using only tight edges). Argue that

    - If V_r has an unmatched vertex then you have an augmenting path, and hence matching size can be increased.
    - If V_r has no unmatched vertices then note that V_r is the neighboring set of V_ℓ and $|V_\ell| > |V_r|$.
    - Let us update the dual variables as follows: increase by ϵ for all vertices in V_ℓ and decrease by ϵ for all vertices in V_r. We choose ϵ to be the maximum possible value while maintaining dual constraints feasible. After the update, some new edges will become tight and also some edges will become non-tight. Prove that the number of vertices reachable from U via alternating paths will definitely increase.
    - How many times can this number increase before you get an augmenting path.
    - Give an upper bound on the overall running time of the primal dual algorithm, which is independent of the weights, just a function of number of vertices and edges.

Sep 8 Lecture 11

31. Consider the following linear program for the minimum weight spanning tree problem.

$$\min \sum_{e \in E} w_e x_e \text{ subject to}$$
$$\sum_{e \in E} x_e = n - 1$$
$$\sum_{e \in E(S, \overline{S})} x_e \geq 1 \text{ for all non-empty } S \subset V.$$

    Construct an example where the LP optimal value is different from the minimum weight of a spanning tree.

Sep 12 Lecture 12

32. Define a notion of approximate complementary slackness. And argue that any primal feasible and dual feasible solutions satisfying approximate complementary slackness will be approximately optimal.

33. Prove that the following algorithm gives 2-approximation for minimize **size** vertex cover:

    (a) Initialize S as empty set.

    (b) Consider any edge which is not covered by vertices in S, include both its endpoints in S.

(c) Mark all the edges covered by newly included vertices as covered.

(d) If there is any edge remaining uncovered, go to (b). Otherwise return S.

34. We saw primal dual algorithm for minimum weight vertex cover in the class. If you want to interpret the algorithm as a greedy algorithm, what would be the greedy step? Below is a formal description of the algorithm we discussed.

(a) Initialize S as empty set.

(b) Consider any edge $e = (a, b)$ which is not covered by vertices in S.

(c) Increase the dual variable y_e till the point that the dual constraint for vertex a or the dual constraint for vertex b (or both) becomes tight.

(d) Include whichever vertex becomes tight in S. Include both vertices if both become tight.

(e) Mark all the edges covered by newly included vertices as covered (observe that the dual variable for any covered edge cannot be increased further).

(f) If there is any edge remaining uncovered, go to (b). Otherwise return S.

Sep 16 Lecture 13

35. Let Δ_n be the n-dimensional probability simplex, i.e.,

$$\Delta_n = \{\lambda \in \mathbb{R}^n : \sum_{i=1}^{n} \lambda_i = 1 \text{ and } \lambda_i \geq 0 \text{ for each } 1 \leq i \leq n\}.$$

Let R be a $m \times n$ real matrix (payoff matrix). Let $R_{i,j}$ be the (i, j)-th entry of R. Recall that $R_{i,j}$ is the payoff of the red player, when the red player goes with strategy i and blue player goes with strategy j.

If the red player goes with a mixed strategy $y \in \Delta_m$ and the blue player goes with a mixed strategy $x \in \Delta_n$ then the expected payoff for the red player will be $y^T Rx$.

Consider a situation where the red player first decides the strategy, say y. And the blue player is given this information. Then the blue player will naturally try to minimize the payoff for the red player. Hence, the payoff for the red player will be

$$h(y) = \min_{x \in \Delta_n} y^T Rx$$

Show that

$$h(y) = \min \left\{ \sum_{i=1}^{m} y_i R_{i,1}, \sum_{i=1}^{m} y_i R_{i,2}, \ldots, \sum_{i=1}^{m} y_i R_{i,n} \right\}$$

Naturally, the red player will choose y so as to maximize $h(y)$. Show that $\max_{y \in \Delta_m} h(y)$ is same as the optimal value of the below LP.

$$
\begin{aligned}
&\max z \\
&\text{subject to} \\
&\sum_{i=1}^{m} y_i R_{i,1} \geq z \\
&\sum_{i=1}^{m} y_i R_{i,2} \geq z \\
&\vdots \\
&\sum_{i=1}^{m} y_i R_{i,n} \geq z \\
&\sum_{i=1}^{m} y_i = 1 \\
&y_i \geq 0 \text{ for each } 1 \leq i \leq m.
\end{aligned}
$$

36. Using strong duality show that the optimal value of the above LP is same as the optimal value of the below LP.

$$
\begin{aligned}
&\min z \\
&\text{subject to} \\
&\sum_{j=1}^{n} R_{1,j} x_j \leq z \\
&\sum_{j=1}^{n} R_{2,j} x_j \leq z \\
&\vdots \\
&\sum_{j=1}^{n} R_{m,j} x_j \leq z \\
&\sum_{j=1}^{n} x_j = 1 \\
&x_j \geq 0 \text{ for each } 1 \leq i \leq n.
\end{aligned}
$$

37. Observe that the optimal value of the above LP is same as $\min_{x \in \Delta_n} g(x)$, where $g(x) = \max_{y \in \Delta_m} y^T R x$.

38. Finally conclude that

$$\min_{x \in \Delta_n} \max_{y \in \Delta_m} x^T R y = \max_{y \in \Delta_m} \min_{x \in \Delta_n} x^T R y.$$