CS 570 - HW 6

Due Thursday, April 29, 2021 (by 4 AM Pacific Time)

1 Integer Programming (20 Points)

In Linear Programming, variables are allowed to be real numbers. Consider that you are restricting variables to be only integers, keeping everything else the same. This is called Integer Programming. Integer Programming is nothing but a Linear Programming with the added constraint that variables be integers. Prove that integer programming is NP-Hard by reduction from SAT.

Solution + Rubric:
Proof by reduction from Satisfiability. Any SAT instance has boolean variables and clauses. Our Integer Programming problem will have twice as many variables, one for each variable and its compliment, as well as the following inequalities: $0 \leq v_i \leq 1$ and $0 \leq v_i \leq 1$
$1 \leq v_i + \neg v_i \leq 1$ for each clause $C = v_1, \neg v_2, ...v_i : v_1 + \neg v_2 + \ldots + v_i \geq 1$

2 HALF-SAT (20 points)

We know that the SAT problem is NP-complete. Consider another variant of the SAT problem: given a CNF formula F, does there exist a satisfying assignment in which exactly half the variables are true? Let us call this problem HALF-SAT. Prove that HALF-SAT is NP-complete.

Solution + Rubric:
Verifying NP is trivial.
For NP hardness:
For each variable X_i, introduce a new variable Y_i and add clauses $(X_i \vee Y_i)$ and $(\neg X_i \vee \neg Y_i)$. Let this new formula be G. These new clauses ensure that for G to be satisfied, X_i and Y_i must always have opposite assignments, i.e., exactly

one of them is True.
If G is satisfied, it always has half the variables true. A satisfying assignment of G (with just considering the X_i's) satisfies all the clauses in F.
On the other hand, given a satisfying assignment of F and setting $Y_i = \neg X_i$, gives a satisfying assignment of G with exactly half the variables true.

3 Taking courses (20 points)

There are N courses in USC, each of them requires multiple disjoint time intervals. For example, a course may require the time from 9am to 11am and 2pm to 3pm and 4pm to 5pm (you can assume the number of intervals of a course is at least 1, at most N). You want to know, given a number K, if it's possible to take at least K courses. You cannot choose any two overlapping courses. Prove that the problem is NP-complete, which means that choosing courses is indeed a difficult thing in our life. Use a reduction from the Independent set problem.

Solution + Rubric:

1. (Showing Problem in NP, 5 points) The solution of the problem can be verified in polynomial time (just check the number of the courses in the solution is larger or equal to K, and they don't have time overlap), thus it is in NP.

2. (Showing Problem in NP-Hard, 15 points) Given an independent set problem, suppose the graph has n nodes and asks if it has an independent set of size at least M. Establish an injection $f : V \times V \to \mathbb{N}$ s.t.

$$f(v_i, v_j) = \begin{cases} i * n + j, & \text{if } i \le j, \\ f(v_j, v_i), & \text{otherwise.} \end{cases}$$

   Now we construct an instance of the course choosing problem, each course corresponds to a vertex of the graph, and if there exists an edge (v_i, v_j) in the original graph, we let the i-th courses require the $f(v_i, v_j)$-th hour. The problem is to determine whether we can choose M courses. Notice that, if there exists an edge (v_i, v_j) in the original graph, then the i-th course and the j-th course will jointly require the $f(v_i, v_j)$-th hour, which means that we can't choose these two courses at the same time. You can verify that any other courses (except i-th and j-th course) will not require $f(v_i, v_j)$-th hour. If the Independent problem is a "yes" instance (has an independent set of size at least M), then we can choose the corresponding courses, and they don't overlap. For the other direction, if we can choose the corresponding courses, then it follows that the independent set problem is a “yes” instance. Thus we can reduce the independent set problem to the course choosing problem in polynomial time. Since independent set problem is NP-Complete, so the course choosing problem is in NP-Hard.

Thus the course choosing problem is NP-Complete.

4 Approximation 1 (20 points)

It is well-known that planar graphs are 4-colorable. However finding a vertex cover on planar graphs is NP-hard. Design an approximation algorithm to solve the vertex cover problem on planar graph. Prove your algorithm approximation ratio.

Assumption: The minimum vertex-cover is of size at least $|V|/2$

Solution + Rubric:

Proof using Assumption In case the assumption of “The minimum vertex-cover is of size at least $|V|/2$” is used, the following proof should be presented:

Propose the Algorithm and show output is a Vertex Cover (13 points)

Algorithm (10 points)

Take a planar graph $G = (V, E)$, and color it in 4 colors via a polynomial run-time algorithm. This will split all vertices in four color groups C_1, C_2, C_3 and C_4. The idea is to use only vertices from the vertices not in the most-frequent color group to construct our vertex-cover. Let the most frequent color be C_4. Thus, we will consider all vertices in $VC = C_1 \bigcup C_2 \bigcup C_3$ as our vertex cover. Clearly, this is polynomial time.

Correctness (only for reference)

To see VC is indeed a vertex cover, consider an edge $(u, v) \in E$. If both $u, v \notin VC$ then $u, v \in C_4$. However, nodes belonging to the same color cannot have an edge. Hence, $\forall (u, v) \in E$, either u or v is covered by VC. Hence, VC is indeed a vertex cover.

Find the Approximation Ratio (10 points)

Note that the most frequent colors will be of size at least $|V|/4$. If not, then all colors will be of size $< |V|/4$ and as a result, total will be less than $|V|$.

This further implies that the vertex cover is of size at most $3|V|/4$.

We already know the lower bound is $|V|/2$.

Hence, the approximation ratio is $\frac{3|V|/4}{|V|/2} = \frac{3}{2}$.

6 points for mentioning most frequent class is of size at least $|V|/4$ and hence size of remaining color vertices is of size at most $3|V|/4$. Arriving at the ratio of $3/2$ is worth 4 points.

Detailed Rubric:

1. Algorithm of choosing all vertices except those of most frequent class (or any variation of this statement) should be awarded 10 points.
2. If the algorithm fails to mention "most frequent class" deduct 3 points.
3. Correctness is for reference only. No points should be deducted even if correctness is not shown.
4. Approximation Ratio: The lower-bound has been provided as an assumption via piazza note 809. For upper-bound, the important argument is that that most frequent color is of size at least $|V|/4$. The upper bound on the vertex-cover would thus be $3|V|/4$. If this is stated, award 6 points. Dividing with lower bound on vertex-cover gives $3/2$. Arriving at the final ratio is another 4 points.

Proof without Assumption In case the assumption of "The minimum vertex-cover is of size at least $|V|/2$" is not used, the following proof should be presented:

Propose the Algorithm and show output is a Vertex Cover (10 points)

Take a planar graph $G = (V, E)$, and color it in 4 colors. This will split all vertices in four color groups C_1, C_2, C_3 and C_4. The idea is to use only vertices from the vertices not in the most-frequent color group to construct our vertex-cover.

We consider the LP formulation of the vertex-cover:

$$\begin{array}{lll} \text{minimize} & z = \sum_{v \in V} x_v & \\ \text{subject to} & x_u + x_v \geq 1 & \forall (u, v) \in E \\ & 0 \leq x_v \leq 1 & \forall v \in V \end{array}$$

Note that in the solution to the LP, the values of x_u can be made to be one of $\{0, 1/2, 1\}$, i.e. every vertex of the LP relaxation of VC is half-integral.

Let the set of vertices with values $\{0, 1/2, 1\}$ be denoted by $V_0, V_{1/2}, V_1$. In the usual vertex-cover, we use the sets $V_{1/2}, V_1$. Since, we have a 4-color graph, we can discard the vertices belonging to the most frequent class from $V_{1/2}$. Without loss of generality, let the most frequent class be C_4. That is, our resulting vertex-cover VC is $V_1 + V'_{1/2}$ where $V'_{1/2} = V_{1/2} - C_4$

Complete Algorithm is worth 10 points. Stating LP formulation is 4 points and removing most frequent class from $V_{1/2}$ is worth 6 points. If most frequent class is removed from $V_1 + V_{1/2}$ award 1 point.

Correctness

To see VC is indeed a vertex cover, consider an edge $(u, v) \in E$. If $x_u = 0$ then $x_v = 1$ as it is a solution to the linear program, so it is covered. If $x_u = 1/2$ then x_v is either $1/2$ or 1. If it is the latter, then again the edge is covered. The only way it is not covered is if both $x_u = x_v = 1/2$ and both belong to C_4. However, vertices belonging to the same color cannot have an edge between them. Thus, all edges are covered in VC and thus it is indeed a vertex cover.

Correctness proof is only for reference.

Find the Approximation Ratio (10 points)

After solving the LP, let the optimal value of the LP be $z^* = \frac{1}{2}V_{1/2} + V_1$. This is the lower-bound for the optimal-vertex-cover. That is, if VC^* is the optimal vertex cover then $|VC^*| \geq z^*$

Note that size of $|VC| = |V_1| + |V'_{1/2}|$. Let $V^i_{1/2}$ denote the vertices in $V_{1/2}$ belonging to the color-group i for $i \in \{1, 2, 3, 4\}$ and $i = 4$ is the most frequent group. Thus, $|V^4_{1/2}| \geq |V_{1/2}|/4$. As a result, $|V'_{1/2}| \leq 3|V_{1/2}|/4$

Thus, we have $|VC| \leq |V_1| + \frac{3}{4}|V_{1/2}| \leq |V_1| + \frac{3}{4}(2(z^* - |V_1|)) \leq \frac{3}{2}z^* \leq \frac{3}{2}|VC^*|$

This gives an approximation ratio of 3/2.

Approximation computation is worth 10 points. Using the output of relaxed-LP as lower bound is worth 2 points. Computing output of relaxed LP in the form of $|V_1|$ and $|V_{1/2}|$ and upper-bounding $|V_{1/2} - C_4| \leq 3|V_{1/2}|/4$ is worth 6 points. Arriving at the final approximation ratio is worth 2 points.

Detailed Rubric:

1. If Algorithm is via ILP, give 10 points.
2. For Approximation Ratio, it is important to consider $|V_1|$ separately and not together with $V_{1/2}$. If both are considered together, i.e. most frequent color is removed from the entire $V_1 + V_{1/2}$, then award 5 points.
3. If the approximation ratio is not $3/2$, but lower bound and upper bound arguments are correct, award 4 points for the approximation ratio part.

5 Approximation 2 (20 points)

Consider the following heuristic to compute a minimum vertex cover of a connected graph G. Pick an arbitrary vertex as the root and perform depth first search. Output the set of non-leaf vertices in the resulting depth first search tree.

1. Show that the output is indeed a vertex cover for G.
2. How good is this approximation? That is, upper bound the ratio of the number of vertices in the output to the number of vertices in a minimum vertex

Solution + Rubric:

Let the dfs tree be T. L denote the set of leaf vertices in the depth first search tree and N - the set of non-leaf vertices.

Proof that output is vertex cover (5 points)

Suppose that N is not a vertex-cover. Then, $\exists (u, v) \in E$ such that $u \notin N$ and $v \notin N$. This implies, both $u \in L$ and $v \in L$, i.e. both u, v are leaves. However, leaves cannot have an edge, hence this is a contradiction.

Therefore, N is indeed a vertex-cover.

Approximation Ratio (15 points)

For approximation ratio, we need to lower bound the optimal vertex cover which is the size of the maximum matching for N.

Above statement or similar restatement is worth 2 points

We show two proofs, either proof is fine.

1. **Proof by Constructing Matching:**

   We consider two particular matching for the tree T. In the first matching (M_1), we assume at least one of the leaf node is present in the matching, and in the second matching (M_2), we assume none of the leaf-nodes are present.

   (a) Consider the nodes at one level before the leaf node, and call this set of nodes L_{-1}. From each node of $l_{-1} \in L_{-1}$, we select one of its children (which is a leaf) and add this edge to our matching set. Our partial matching set now has a size of $|L_{-1}|$.

   (b) Now, note that we cannot add any edge to the matching which already contains the nodes at L_{-1}.

   (c) Consider the tree T_1 in which all leaf nodes L and their parents L_{-1} are removed, i.e., $T_1 = T - L_{-1} - L$. For T_1, the leaf nodes are the parents of the nodes in L_{-1}, which we call L_{-2}. Following the same procedure as for T, we can add edges connecting nodes in L_{-3} to those in L_{-2}. So we have added another $|L_{-3}|$ to our partial matching set.

   (d) We can continue this process till L_{-k} for some k is not empty.

   (e) The final matching set, which we call M_1 would contain edges from (x_1, x_2) such that $x_1 \in L_{-k_o}, x_2 \in L_{-k_o+1}$ where k_o is an odd number and $|L_{k_o}| > 0$. As a result, $|M_1| = |L_{-1}| + |L_{-3}| + |L_{-5}| + \ldots$.

   Now, we again start from T. But this time, we don't include any matching containing the leaf node. So we first create $T^{'} = T - L$. On $T^{'}$, we apply the same steps 1-4 as we did for T. This matching M_2 would be using edges from (x_1, x_2) such that $x_1 \in L_{-k_e}, x_2 \in L_{-k_e+1}$ where k_e is even and $|L_{-k_e}| > 0$. As a result M_2 on Tree $T^{'}$ would be of size $|M_2| = |L_{-2}| + |L_{-4}| + \ldots$.

   Note that M_2 on the whole tree T would include some edges from the L_{-1} to L as well. As a result, $|M_2| \geq |L_{-2}| + |L_{-4}| + \ldots$

   Note that $|M_1| + |M_2| \geq |L_{-1}| + |L_{-2}| + \ldots \geq |N|$. Thus, $max(|M_1|, |M_2|) \geq |N|/2$. That is, the tree T has at least one matching which is of size more than half the number of internal nodes.

   Alternative way to see the matching:

   Essentially, M_1 and M_2 can be thought of as edge connections from the parent nodes to their descendants at the odd-levels and even-levels (levels computed with respect to the leaves). So an argument suggesting to use the larger of M_1 or M_2 is equally valid.

   Rubric for Construction Proof:

(a) Constructing the matching is 7 points. Argument with odd, even levels is equally correct.

(b) Noting that $|M_1| + |M_2| \geq |N|$ and therefore $max(|M_1|, |M_2|) \geq |N|/2$ is worth 3 points.

(c) Arriving at the final approximation ratio is worth 3 points.

2. **Proof via Induction:**

   An alternate proof is via Induction on the size of the tree. We claim that there exists a maximal matching size of a tree is at least $|f(T)|/2$ where $f(T)$ denotes the internal nodes of a tree.

   This is trivially true for tree of size 1.

   Suppose this is true for size of tree k. That is, a tree T_k of size k has a maximal matching size of $|f(T_k)|/2$

   Now, for tree of size $k+1$, we can create one matching as follows: include one edge joining the root (call it r) and one of its children (call it r_1). Let the sub-trees of r be $[T_1, \ldots, T_n]$, the sub-trees of T_1 be $[T_{11}, \ldots, T_{1m}]$. Then, we can create a matching of size

$$1+\frac{1}{2}(\sum_{i=2}^{n}|f(T_i)|+\sum_{j=1}^{m}|f(T_{1j}|) = 1+(|f(T_{k+1})|-2)/2 = |f(T_{k+1})|/2 = f(k+1)/2$$

   Hence, proved.

Thus, the optimal vertex cover say VC^* is lower-bounded by $|N|/2$. The vertex-cover generated by our approximation algorithm is N. Thus, the approximation ratio is 2.

Arriving at the final approximation ratio is worth 3 points

Detailed Rubric:

1. Any similar arguments such as choosing the parent of the leaf-node covers the leaf-node is also fine and should be awarded 5 points for the part 1.

2. For approximation ratio, one has to make the argument that the minimum vertex cover is lower-bounded by the maximal matching and that there exists a maximal matching of size at least $N/2$. Then, this statement has to proved. The argument alone (without following proof) is worth 2 points.

3. The proof to show there exists a maximal matching of size at least $N/2$ is worth 10 points. Two proofs, one by constructing two matching and other by induction are provided. Either proof is valid. The rubrics for the proofs are present in their relevant section.

4. Giving the final approximation ratio as 2 is 3 points. If no proof is given but only the approximation ratio is mentioned award 2 points. If final approximation ratio is incorrect, award 0/3 points for this part.