CSC373S Lecture 21

- We now begin our last topic of the term, integer programming IP and linear programming LP. Mainly we will be discussing the LP relaxation of IPs and rounding such LPs to obtain IP solutions.

  We will start with some examples and then briefly discuss some LP theory.

  While there are problems which are directly represented by LPs (eg max flow), I will focus on NP hard problems which are (in most cases) naturally represented by IPs. Indeed solving IPs is an NP-hard problem although there are many heuristics and special cases that are solvable in practice. LPs are efficiently solvable, both in practice and theoretically although they do not tend to be as efficient as simpler combinatorial methods.

  Now lets consider some NP hard problems which we will first express as an IP (often as a {0,1} IP which is natural when one is considering problems where solutions are {0,1} vectors) and then relax to an LP. Having done that, there are two ways to obtain integral solutions (to the given problem): 1) by "rounding the LP solution" and 2) by using duality (ie dual inequalities) to guide in the setting of primal variables. The first approach is in some sense "conceptually simpler" but is not as efficient as the second approach (which avoids solving the LP). Neither approach is guaranteed to give a good approximation in general and the approach is intuitively limited by the "integrality gap" which measures the relative difference between a rational (LP) optimal solution and an integral (IP) optimal solution. We will only have time to discuss the first (rounding the LP solution) approach.

- What is the computational significance of the difference between IP and LP? As stated above, IP is an NP-hard problem whereas LP is polynomial time. In practice IP is sometimes optimally solved for some large applications but in general it is NP-hard and we do not know how to solve IPs efficiently. Linear programming has been used in practice since the late 1940s (mainly by "the simplex method template"). Here (similar to saying thre Ford Fulkerson template for max flow) we have to specify the "pivoting rule" to obtain a specific simplex method. Theoretically, no known pivting rule is poly time and most (if not all) known methods in practice have been shown to have bad examples which can take exponential time to terminate.

- The first problem we will consider is the (weighted) vertex cover problem (a minimization problem) See section 11.6 in text. This is the standard example of a simple rounding procedure.

  A vertex cover of $G = (V, E)$ is a subset $V' \subseteq V$ such that for every edge $(u, v) \in E$, either u or v (or both) are in V'. Given G and $w : V \rightarrow \Re^{\geq 0}$, the goal is to ind a min weight vertex cover V'. In the unweighted case, the goal is to minimize $|V'|$.

Here is how this problem is expressed as a {0,1} IP.

minimize $\sum_i w_i x_i$

subj to $x_u + x_v \geq 1$ for all $(u, v) \in E$ covering constraint

$x_i \in \{0, 1\}$ for all $v_i \in V$ integrality constraint

Here the intended meaning of x_i is that it is 1 iff vertex v_i is in the VC.

NOTE: There can be many IPs for a given problem.

The LP relaxation replaces the integrality constraint by $0 \leq x_i \leq 1$ for all $v_i \in V$. It is not hard to see that this can be simplified to $x_i \geq 0$ for all $v_i \in V$ since an optimum solution cannot set a variable $x_i > 1$.

- A second problem is a generalization of vertex cover. In the next lecture, we will consider the following (weighted) set cover problem.

  Given a collection of sets $\mathcal{C} = S_1, ..., S_n$ over a universe $U = \{u_1, ..., u_m\}$ with weight function $w : \mathcal{C} \rightarrow \Re^{\geq 0}$, find a subcollection $\mathcal{C}'$ such that $\cup_{i:S_i \in \mathcal{C}'} S_i = U$

  We can formulate this as a $\{0, 1\}$ IP :

  minimize $\sum_i w_i x_i$

  subj to $\sum_{i:u_j \in S_i} x_i \geq 1$ for all $u_j \in U$

  IP: $x_i \in \{0, 1\}$

  LP relaxtion : $x_i \geq 0$

  Vertex cover can be seen as a special case of set cover, namely:

  A vertex v is a set $S_v = \{e_j : v$ is adjacent to edge $e_j\}$ so that the universe is the set of edges E and the sets in the collection are the vertices V.

  Note that this collection of sets has the property that each universe element appears in exactly two sets. This leads to what is called the f-frequency set cover problem where each element occurs in at most f sets. Note that vertex cover is a special case of 2-frequency set cover (since 2-frequency can also have have universe elements which only occur once.

- So now lets assume we have optimally solved the LP relaxation of the stated IP for vertex cover.

The deterministic rounding for vertex cover:

Let $\{x_i^* | 1 \leq i \leq n\}$ be an optimal fractional solution (ie optimal solution to the relaxed LP). Then set $\bar{x}_i = 1$ iff $x_i^* \geq 1/2$, else set to 0.

Claim: $\{\bar{x}_i | 1 \leq i \leq n\}$ is an IP solution (i.e. a vertex cover)

Proof: Suppose $(u, v) \in E$, then $x_u^* + x_v^* \geq 1$ since the $\{x_i^*\}$ values are an LP solution. This implies that either $x_u^* \geq 1/2$ or $x_v^* \geq 1/2$ (or both) which implies that either $\bar{x}_u = 1$ or $\bar{x}_v = 1$ (or both).

Claim: This rounded integral solution is a 2-approximation.

Proof:

$\sum_i \bar{x}_i \leq \sum_i 2(x_i^*) = 2LP-OPT \leq 2OPT$

This is a really quick way to see that the weighted vertex cover problem has an "efficient" (given that we can solve an LP) approximation algorithm. It is easy to see that the unweighted VC problem had a simple greedy 2 approximation using the idea of a maximal matching. There are efficient greedy algs for the weighted VC problem but surprisingly, perhaps the most obvious greedy approach doesn't work ;i.e. is not an O(1) approx but rather an H_n approx where $H_n = \sum_{i=1}^{n} \frac{1}{i}$. Namely, the natural greedy algorithm works as follows: at any iteration while there are still uncovered edges, choose that vertex v_i so as to minimize $w_i/(\text{current degree of } v_i)$. This alg will (only be an H_n approx (or really H_d if $d =$ max degree) even for the unweighted case. Currently after much research there is no known poly time $(2-\epsilon)$ approx algorithm (for any $\epsilon > 0$) for vertex cover and subject to some (debatable) conjectures there cannot be such an approximation alg.

Set cover can be computed by a greedy algorithm (the natural greedy algorithm above for the weighted vertex cover is a special case) with approx ratio H_d where $d = max_i|S_i|$. Well accepted complexity assumptions say that this is the best possible approximation for the set cover problem.

- For a minimization problem, the ratio $max_I \frac{IP-OPT(I)}{LP-OPT(I)} = \frac{OPT(I)}{LP-OPT(I)}$ is called the integrality gap. For a maximization problem, there is the same confusion (as for approximation ratios, locality gaps) as to expressing integrality gaps being ≥ 1 or ≤ 1.

  Since $LP(I) \leq OPT(I)$ (for a minimization problem), it is often claimed that the locality gap for a particular IP/LP formulation of a problem is a bound on the best possible approximation realizable by this particular IP/LP (for any rounding procedure).

  In practice this may be how best to view the impact of an established locality gap but technically, it doesn't follow as far as I can see that there might not be a rounding procedure which on every instance I converts the $LP(I)$ solution to a feasible solution which is close to $OPT(I)$.

  For vertex cover, the locality gap of this IP/LP is a meaningful lower bound on the best approximation possible since we know that the rounding procedure must take the "$n/2$ fractional solution" for the complete graph to a solution with at least cost $n-1$. So any rounding that is independent of the input instance (which I would call "oblivious rounding") can double the value of the solution. On the other hand, for an n node cycle $I, OPT(I) = \lceil (n/2) \rceil$ whereas the fractional opt is $n/2$ so that on the cycle an input independent rounding can cause a solution (at least) twice from the optimal.

In any case, most people take the integrality ratio as a limitation on the usefulness of a particular IP/LP formulation.

- Sometimes there is a rather natural IP formulation and natural LP relaxation and also sometimes a natural (oblivious) rounding. This is what we observed in the case of the vertex cover problem. But in general IP/LP rounding can be very non trivial. "Rounding" just means converting the optimal fractional solution to an integral solution and does not necessarily mean rounding in the traditional numerical sense of the word (ie rounding up or down to nearest integer).

- For vertex cover, one can think of various reasonable ways to add constraints to the IP formulation. For example, for every triangle (v_i, v_j, v_k) (with all three pairwise edges) in the graph, add the constraint that $x_i + x_j + x_k \geq 2$. More generally, we can add exponentially many constraints that enforce the fact that every length k odd cycle requires $\lceil k/2) \rceil$ vertices to be covered. It turns out that such constraints can be added by a more general procedure but in the case of vertex cover, the integrality gap remains $2 - o(1)$.