CS880: Approximations Algorithms

Scribe: Dave Andrzejewski **Lecturer:** Shuchi Chawla

Topic: Randomized routing, LP-Duality **Date:** 3/1/07

The first part of the lecture shows how a randomized rounding scheme can be used to transform an optimal LP solution to a valid solution of the original problem. The specific problem used for this example is the min-congestion routing problem.

The second part of the lecture introduces the concept of LP-Duality, and the primal-dual interpretation of the maxflow-mincut problem.

12.1 Randomized rounding

12.1.1 Min congestion routing problem

GIVEN: a graph $G = (V, E)$ and k pairs of demand vertices (s_i, t_i).
DO: find a single path p_i from s_i to t_i for every i, trying to minimize the congestion C. The congestion is defined as the number of paths going through the most-used edge in the graph.

$$C = \max_{e \in E} |\{i | e \in p_i\}|$$

We then cast the problem as an LP where x_{ie} is defined as the amount of path i flow being sent through edge e.

$$\begin{aligned}
&\min t && \text{obj fcn} \\
&\sum_{e \in \delta^+(v)} x_{ie} = \sum_{e \in \delta^-(v)} x_{ie} && \forall i, \forall v \neq s_i, t_i \\
&\sum_{e \in \delta^-(s_i)} x_{ie} = \sum_{e \in \delta^+(t_i)} x_{ie} = 1 && \forall i \\
&\sum_i x_{ie} \leq t && \forall e \\
&x_{ie} \geq 0 && \forall i, \forall e
\end{aligned}$$

The sets $\delta^+(v)$ and $\delta^-(v)$ correspond to the incoming and outgoing flow, respectively, for vertex v. The summation constraints then enforce flow conservation, and source/sink assignments.

Since the objective function is to minimize t, which is constrained to be an upper bound for the flow across any edge, t will give us the (possibly fractional) congestion for a solution point of this LP.

In order to recover an integral/unsplittable flow solution from the LP solution, we will consider an equivalent formulation of the LP.

In this alternative formulation x_p refers to the flow along path p, and P_i is the set of all possible paths from s_i to t_i.

$$
\begin{aligned}
&\min t && \text{obj fcn} \\
&\sum_{p \in P_i} x_p = 1 && \forall i \\
&\sum_{\{p|e \in p\}} x_p \leq t && \forall e \\
&x_p \geq 0 && \forall p
\end{aligned}
$$

These formulations are equivalent in the sense that any solution to one can be converted into a solution of the other.

To convert from the first to the second, find all non-zero x_{ie} for a given i. Then, considering only these edges, perform DFS from s_i to find a path to t_i. For this path p, set x_p to the minimum x_{ie} value on the path, then subtract that amount from all x_{ie} on the path. That will effectively remove the minimum edge from our edge set. Repeat this procedure until there are no edges remaining, building a set of x_p values as you go.

To convert from the second LP to the first, find all non-zero x_p for a given P_i, then simply add that much flow to the edge flow x_{ie} for each $e \in p$.

It is important to note that the second formulation is impractical to solve directly, as the number of possible paths in the P_i sets will lead to exponentially many constraints. However the x_p quantities will come in useful for our randomized rounding scheme, as we will see.

12.1.2 Randomized rounding transformation

As on previous problems, the optimal objective function value of our LP forms a lower bound on the true optimal solution of the original problem. That is, $LP^* \leq OPT$. However, we need a way to do rounding from the flows in our LP solution to legal unsplittable flows for the original problem.

Our approach will be to solve the LP in the first formulation, convert the solution to the second formulation, and then treat the x_p values as path selection probabilities. For a given path set P_i, the constraints that all x_p must be ≥ 0 and sum to 1 ensure that this is a valid probability distribution. The algorithm is then relatively simple.

1. solve original formulation LP,
2. convert solution LP^* to second formulation
3. for each i, pick a $p \in P_i$ with probability x_p

Obviously, this algorithm will select exactly one path for each (s_i, t_i) pair, yielding a valid solution. What will the congestion be?

For every edge, total traffic in LP^* is $\leq t$. The traffic corresponds to the x_p values, which also then correspond to the path selection probabilities.

For each edge define a set of indicator random variables X_i.

$$X_i = \begin{cases} 1 & \text{if algo picks edge } e \text{ for commodity } i \\ 0 & \text{else} \end{cases}$$

Then define the expectation of X_i.

$$E(X_i) = \mu_i = \sum_{\{p \in P_i | e \in p\}} x_p$$

Then for any edge e and set of random variable values $X_i = x_i$, define its congestion to be $C_e = \sum_i x_i$. The expectation of the edge congestion can then be computed using our μ_i values and the linearity of expectation.

$$E(C_e) = E(\sum_i X_i) = \sum_i E(X_i) = \sum_i \mu_i \leq t$$

For our approximation factor, we want the sum of all X_i for every edge to be small. As shown above, we know the expectation for each edge e to be $\leq t$. More specifically, we want to show that for an appropriate value of λ, $Pr[C_e \geq \lambda E(C_e)]$ is small, for all e.

This can be accomplished through the use of a concentration bound result, specifically Chernoff's bound [1]. This bound assumes that the individual X_i are independent variables which can take on the values $\{0, 1\}$, and then uses Markov's inequality applied to a certain function to get the bound. For $\lambda \in [0, 1]$, the bound is:

$$Pr[X \notin (1 \pm \lambda)E(X)] \leq \exp \frac{-\lambda^2 E(X)}{3}$$

We customize the bound to our specific situation, using the fact that $\sum_i \mu_i \leq t$ to get the following inequality.

$$Pr[C_e \geq (1 + \lambda)t] \leq \exp \frac{-\lambda^2 \sum_i \mu_i}{3}$$

Note that $\sum_i \mu_i$ may be quite small, meaning that no value of $\lambda \leq 1$ will give a small probability of error. So in order to obtain a smaller bound we must use a more general formulation of Chernoff's bound that also holds for $\lambda > 1$.

$$P(C_e \geq (1 + \lambda)t) \leq \left(\frac{\exp \lambda}{(1 + \lambda)^{1+\lambda}} \right)^{\sum_i \mu_i}$$

We then manipulate the right-hand side.

$$\begin{aligned}
\left(\frac{e^\lambda}{(1+\lambda)^{1+\lambda}}\right)^{\sum_i \mu_i} &\leq \left(\frac{e^\lambda}{\lambda^\lambda}\right)^{\sum_i \mu_i} \\
&\leq \left(\frac{1}{(\lambda/e)^\lambda}\right)^{\sum_i \mu_i}
\end{aligned}$$

We now pick a value of λ for which the term λ^λ in the denominator becomes $n^O(1)$. Specifically we set $\lambda = O(\frac{\log n}{\log\log n})$. We then substitute in our definition of λ.

$$\begin{aligned}
\lambda^\lambda &= \exp(\lambda \log \lambda) \\
&= \exp\left(\frac{\log n}{\log\log n}(\log\log n - \log\log\log n\right) \\
&= \Theta(n^c)
\end{aligned}$$

where c is a factor determined by the actual terms in the $O()$ equation used to calculate λ. Setting λ appropriately to get $c = 3$, we plug this back into the original bound and end up with the result that for any edge e:

$$Pr[C_e > (1 + \left(\frac{\log n}{\log\log n}\right))t] < 1/n^3$$

We take the union of this bound over all $\leq n^2$ edges to get a total probability bound.

$$P(\exists e | C_e > (1 + \left(\frac{\log n}{\log\log n}\right))t) < 1/n$$

By repeatedly applying our algorithm, we can then achieve an arbitrarily low probability of exceeding an $(1 + \frac{\log n}{\log\log n})$-approximation to OPT.

12.2 LP-Duality

12.2.1 Definition/Derivation of LP-Duality

Consider the following example linear program. (For more discussion of this example, see [1].)

$$\begin{aligned}
\min x + 4y& \\
x + 2y &\geq 5 \\
2x + y &\geq 4 \\
x, y &\geq 0
\end{aligned}$$

How could we obtain a lower bound on the true optimal objective function value for this LP (without actually solving it, that is)?

We can take a non-negative linear combination of the constraint equations. Since $x, y \geq 0$, if the x coefficient in our combination is $\leq$ the x coefficient in the objective function, and the same holds true for the y coefficients, our linear combination of constraints must also be $\leq$ the objective function for any legal x, y.

When taking linear combinations of the constraints, we will also end up a linear combination of the right-hand side of the constraints. Since it is a lower bound on the left-hand side, the linear combination of the right-hand sides of the constraints is also a lower bound on the objective function.

To see what we mean, call the linear combination coefficients u, v.

$$\begin{aligned} &\min x + 4y \\ &(x + 2y)u \geq 5u \\ &(2x + y)v \geq 4v \\ &x, y, u, v \geq 0 \end{aligned}$$

Enforcing that the x, y coefficients in the linear combination are $\leq$ than the x, y coefficients in the original objective function gives us some constraints on u, v. Since we want the tightest lower bound possible, we then want to maximize the right-hand size of the constraint linear combination. This mix of constraints and objective function give us a new LP.

$$\begin{aligned} &\max 5u + 4v \\ &u + 2v \leq 1 \\ &2u + v \leq 4 \\ &u, v \geq 0 \end{aligned}$$

This LP is known as the dual of the original LP, which is called the primal. It is interesting to note that the objective function coefficients of the primal have become the constraint bounds in the dual, while the constraint bounds of the primal have become the objective function coefficients of the dual. Also, the u, v constraint coefficient matrix is the transpose of the x, y constraint coefficient matrix.

The relationship between the primal and dual LPs is very special and useful. The specifics will be spelled out in series of lemmas. For notation, $Val_P(x, y)$ and $Val_D(u, v)$ are the objective function values of the primal and dual LPs, respectively, with an $*$ denoting the optimal objective function value.

Theorem 12.2.1 *Let (x, y) be any feasible primal solution. Let (u, v) be any feasible dual solution. Then $Val_P(x, y) \geq Val_D(u, v)$.*

This result follows from manipulation of the constraints in the definition of the dual presented above, and is known as the Weak LP-Duality Theorem.

Theorem 12.2.2 *Val_P^* is finite iff Val_D^* is finite.*

Theorem 12.2.3 *If both the primal and the dual have feasible solutions, then $Val_P^* = Val_D^*$.*

These results are known as the Strong LP-Duality Theorem.

What if there are no feasible solutions for one of the versions of the problem, or if one of the problems is unbounded? It turns out that the dual has no feasible solutions iff the primal is unbounded below. Likewise, the primal has no feasible solution iff the dual is unbounded above.

Finally, it is interesting to note what happens when we take the dual of the dual.

Lemma 12.2.4 *The dual of the dual is the original primal.*

12.2.2 Applications of LP-Duality

12.2.2.1 General motivation

Why do we care about LP-Duality? For one thing, some of the optimization algorithms for actually finding LP solutions rely heavily on LP-Duality and its consequences.

Aside from that, it may be that the primal formulation is unwieldy, or that the rounding transformation for the dual solution may be more favorable. The dual formulation also may yield useful combinatorial insight into problem structure.

Finally, we can use the structure of the primal and dual to guide a purely combinatorial approximation algorithm for the underlying optimization problem. This technique is known as the primal-dual method and will be sketched out more fully in the next lecture.

12.2.2.2 Mincut-Maxflow

Consider the following LP formulation of the standard maxflow problem. Let x_p be the flow on path p, which is a path from source s to sink t.

$$
\begin{aligned}
\max \sum_p x_p & \qquad \text{obj fcn} \\
\sum_{\{p|e\in p\}} x_p \le c_e & \qquad \forall e \\
x_p \ge 0 & \qquad \forall p
\end{aligned}
$$

Then take the dual.

$$
\begin{aligned}
\min \sum_{e\in E} y_e c_e & \qquad \text{obj fcn} \\
\sum_{e\in p} y_e \ge 1 & \qquad \forall p \\
y_e \ge 0 & \qquad \forall p
\end{aligned}
$$

Note that since all $c_e \geq 0$ and we are trying to minimize the sum over e, in a solution no y_e will be assigned a value greater than 1.

What is the intuitive interpretation of the dual of our maxflow LP? Each y_e corresponds to 'choosing' an edge. The constraints state that we must choose ≥ 1 edge from every path, and the objective function tells us to minimize the sum of $y_e c_e$ over all e.

Ignoring fractional y_e, this means that our optimal LP solution to this dual needs to choose edges so that every path contains one of the chosen edges, and also choose the edges with the smallest total capacity. The smallest capacity set of edges which interesects every path from s to t is, by definition, the minimum weight separating cut between s and t.

As it turns out, all basic points of the primal and dual LPs are integral. This, along with the Strong LP Duality theorem (Theorem 12.2.3), implies the Max-Flow Min-Cut Theorem. The following lemma and theorem formalize this notion.

Lemma 12.2.5 *The Mincut LP has integral basic points.*

Corollary 12.2.6 *Maxflow-Mincut Theorem: The maximum possible flow between s and t is equal to the capacity of the minimum cut separating s and t.*

In this way LP-Duality allows us to clearly see the duality of the maxflow and mincut problems.

References

[1] V. Vazirani. Approximation Algorithms. Springer, 2001.