CS 466/666 Algorithm Design and Analysis . Spring 2020

Lecture 18 : Matching polytopes

We will use a rank argument to show that the natural LP for bipartite matching is integral.

Then, we will see how the technique can be extended to design approximation algorithms for two related NP-hard problems, the 3-dimensional matching problem and the general assignment problem.

Finally, we briefly discuss Edmonds' LP for matchings in general graphs.

Bipartite matching

Consider the maximum bipartite matching problem.

For a subset of edges $F \subseteq E$, we use the shorthand $x(F)$ for $\sum_{e \in F} x_e$.

The natural LP relaxation is :

$$\max \sum_{e \in E} w_e \cdot x_e$$

$$x(\delta(v)) \leq 1 \quad \text{for } v \in V$$

$$x_e \geq 0 \quad \text{for } e \in E.$$

Outline

We will prove that there is an optimal integral solution for this LP using a rank argument.

Recall that there is always an optimal basic solution, which is defined by $|E|$ linearly independent tight constraints.

Note that there are only $|V|$ degree constraints, so if $|E| > |V|$ then there must be edges with $x_e = 0$.

We will show that if there is an edge e with $x_e \in \{0,1\}$, then we can reduce the problem into a smaller one.

Suppose we could not reduce the problem anymore. i.e. $0 < x_e < 1$ for all $e \in E$ in the remaining graph.

We will use a rank argument to prove that x is not a basic solution, a contradiction.

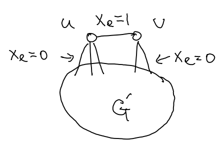
Therefore, there must be an edge e with $x_e \in \{0,1\}$ in any basic solution, and we can conclude that there is an optimal integral solution.

Reduction

The induction hypothesis is that the LP is integral for all graphs with fewer edges.

① If $x_e = 0$ for some edge e, we simply delete this edge from the graph. It doesn't change the LP value and the graph is smaller, so by induction there is an optimal integral solution for the LP.

② If $x_e = 1$ for some edge $e = uv$, we remove u and v from the graph and all the incident edges on u,v.



Call the remaining graph $G' = G - \{u,v\}$.

Note that the LP solution restricted to G' is a feasible solution with value $opt(G) - w_e$, where $opt(G)$ is the objective value of the LP in G.

By induction, there is a matching M' (i.e. an integral solution) in G' with objective value $opt(G) - w_e$.

Then, clearly, $M = M' + uv$ is a matching in G with objective value $opt(G)$.

To summarize, the reduction is very simple. We just follow the LP to choose an edge with $x_e = 1$ and to remove an edge with $x_e = 0$.

Rank argument

Now we prove that the simple reductions above would go all the way.

Lemma There exists an edge e with $x_e \in \{0,1\}$ in any basic solution.

Proof Suppose, by contradiction, that $0 < x_e < 1$ for all $e \in E$.

Recall that there are $|E|$ linearly independent tight constraints for a basic solution x.

Since $x_e > 0$ for all $e \in E$, all the tight constraints are degree constraints.

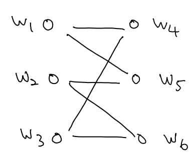
Call W be the set of vertices with tight degree constraints, i.e. $x(\delta(w)) = 1 \quad \forall w \in W$.

Then, we must have $|W| \geq |E|$.

For any $w \in W$, we must have $deg(w) \geq 2$ since $x(\delta(w)) = 1$ but $x_e < 1 \quad \forall e \in E$.

So, $|W| \geq |E| = \frac{1}{2} \sum_{v \in V} deg(v) = \frac{1}{2}\left(\sum_{v \in W} deg(w) + \sum_{v \notin W} deg(v) \right) \geq \frac{1}{2}\left(\sum_{v \in W} 2 + \sum_{v \notin W} 0 \right) = |W|$.

Thus, the inequalities must hold as equalities.



In particular, we have $deg(v) = 2 \quad \forall v \in W$ and $deg(v) = 0 \quad \forall v \notin W$.

We will show that the constraints in W are not linearly independent.

But this implies that the number of tight linearly independent constraints $\leq |W| - 1 = |E| - 1$, contradicting x is a basic solution.

Why the constraints are not linearly independent? Let's do it slowly.

Each constraint is a vector in $\mathbb{R}^{|E|}$.

For a subset of edges $F \subseteq E$, call χ_F the characteristic vector of F if $\chi_F(e) = \begin{cases} 1 & \text{if } e \in F \\ 0 & \text{if } e \notin F \end{cases}$.

Note that each tight constraint in the bipartite matching LP is just $\langle \chi_{\delta(v)}, x \rangle = 1$ for some $v \in V$.

Let W_1 be the subset of W on one side of the bipartite graph and $W_2 \subseteq W$ be on the other side.

Since $deg(v) = 0 \quad \forall v \notin W$, all the edges have exactly one endpoint in W_1, and thus $\sum_{w \in W_1} \chi_{\delta(w)} = \chi_E$.

By the same argument, $\sum_{w \in W_2} \chi_{\delta(w)} = \chi_E$.

Therefore, we have $\sum_{w \in W_1} \chi_{\delta(w)} = \sum_{w \in W_2} \chi_{\delta(w)}$, showing that the tight constraints in W are linearly dependent. $\square$

Approximation algorithms

We will extend the above rank argument to design approximation algorithms for NP-hard variants of bipartite matching.

The high level idea is to show that the basic solutions (although not integral) are "close" to integral:

- Sometimes it means that there are variables with value close to one;
- Sometimes it means that there are constraints almost satisfied by integral solutions.

We will see an example for each of the above "closeness" to integrality.

LP-rounding

A standard approach to design approximation algorithms is to use LP/SDP relaxations (SDP is a generalization of LP).

For a minimization problem, we say an algorithm is an α-approximation algorithm ($\alpha \geq 1$) if $\frac{\text{SoL}}{\text{OPT}} \leq \alpha$,

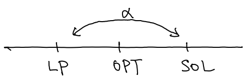
where SoL denotes the value of the solution our algorithm returns and opt denotes the optimal value.

For a maximization problem, we say an algorithm is an α-approximation algorithm ($\alpha \leq 1$) if $\frac{\text{SoL}}{\text{OPT}} \geq \alpha$.

The difficulty of designing approximation algorithms is that we can't compute opt efficiently, and so it seems difficult to bound the approximation ratio.

Instead, we use LP/SDP relaxations to compute a lower bound for minimization or an upper bound for maximization, and if we could show that our integral solution is within a factor of α to the optimal LP/SDP solutions, then our integral solution is an α-approximate solution.

minimization



How to prove such a result?

A common strategy is to start from an optimal LP solution, and design a "rounding" algorithm to turn it into an integral solution in a way that we can compare the objective values.

There are many different approaches to design rounding algorithms (see the book by Williamson and Shmoys).

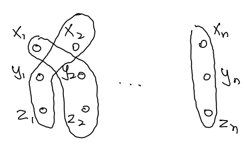
We are going to see one way by analyzing the basic optimal solutions for LP.

3-dimensional matching

Let $X = \{x_1, \dots, x_n\}$ and $Y = \{y_1, \dots, y_n\}$ and $Z = \{z_1, \dots, z_n\}$ be the 3-dimensions.

We are given m triples of the form (x_i, y_j, z_k), and the task is to find a maximum number of disjoint triples, i.e. no two chosen triples have the same coordinate.

Equivalently, this is a maximum matching problem in a 3-uniform tri-partite hypergraph, where the goal is to find a maximum set of vertex-disjoint hyperedges.



We write the following simple and natural LP relaxation of the problem.

$$\max \sum_{e \in E} x_e$$

$$x(\delta(v)) \leq 1 \quad \forall v \in V, \text{ where } \delta(v) \text{ is the set of hyperedges containing } v$$

$$x_e \geq 0 \quad \forall e \in E, \text{ where } E \text{ is the set of hyperedges.}$$

We will show that there is an LP-rounding algorithm with approximation ratio $\frac{1}{2}$.

The key is to prove that there is a hyperedge e with $x_e \geq \frac{1}{2}$ in any basic solution.

The proof is similar to the one above for bipartite matching.

<u>Lemma</u> For any basic solution x with $x_e > 0$ for all $e \in E$, there exists a hyperedge e with $x_e \geq \frac{1}{2}$.

<u>Proof</u> Recall that in a basic solution x, there are $|E|$ linearly independent tight constraints.

Suppose, by contradiction, that all edges have $x_e < \frac{1}{2}$.

Let W be the set of vertices with tight degree constraints, i.e. $x(\delta(v)) = 1 \quad \forall v \in W$.

Since $x_e < \frac{1}{2} \ \forall e$, we must have $\deg(v) \geq 3$ for $v \in W$.

So, $|W| \geq |E| = \frac{1}{3} \sum_{v \in V} \deg(v) = \frac{1}{3}\left(\sum_{v \in W} \deg(w) + \sum_{v \notin W} \deg(v)\right) \geq \frac{1}{3}\left(\sum_{v \in W} 3 + \sum_{v \notin W} 0\right) = |W|$.

Thus, inequalities hold as equalities, and in particular $\deg(v) = 0 \ \forall v \notin W$ and $|W| = |E|$.

We will show that the constraints in W are not linearly independent.

But this implies that the number of tight linearly independent constraints $\leq |W| - 1 = |E| - 1$,

contradicting x is a basic solution.

To see why the tight constraints in W are linearly independent, let $W_X = W \cap X$, $W_Y = W \cap Y$ and $W_Z = W \cap Z$.

Since $\deg(v) = 0 \ \forall v \notin W$ and every hyperedge intersects W_X, W_Y, W_Z exactly once, we have

$\sum_{w \in W_X} x_{\delta(w)} = \sum_{w \in W_Y} x_{\delta(w)} = \sum_{w \in W_Z} x_{\delta(w)} = x_E$, showing linear dependence of these tight constraints. □

<u>Rounding algorithm</u>

Compute an optimal basic solution.

① If $x_e = 0$ for some hyperedge e, delete e and recurse.

② If $x_e \geq \frac{1}{2}$ for some e, add e to the solution and delete all the intersecting hyperedges and recurse.

<u>Approximation guarantee</u>

We show that the returned solution of our algorithm is a $\frac{1}{2}$-approximate solution, by induction.

The case when $x_e = 0$ is easy, so just focus on the case when we add a hyperedge e with $x_e \geq \frac{1}{2}$.

Let this hyperedge by (v_x, v_y, v_z).

Since $x(\delta(v_x)) = 1$, the total fractional value on hyperedges intersecting e at v_x is $1 - x_e$, similarly for v_y and v_z.

So, the total fractional value we removed after adding e is at most $x_e + 3(1 - x_e) = 3 - 2x_e \leq 2$ as $x_e \geq \frac{1}{2}$.

Therefore, in the remaining hypergraph, there is an LP-solution with objective value at least opt $- 2$.

By induction, there is an integral solution M' with objective value $\geq \frac{1}{2}(\text{opt} - 2) = \frac{\text{opt}}{2} - 1$ in the remaining hypergraph.

Hence, $M = M' + e$ is an integral solution with objective value $\geq \left(\frac{opt}{2} - 1\right) + 1 = \frac{opt}{2}$, proving the approximation ratio. □

Remark: Using an additional idea called "local ratio", one can obtain a $\frac{1}{2}$-approximation algorithm for the weighted problem, and this is still the best known approximation guarantee for this problem. See Chapter 9.2 of "iterative methods in combinatorial optimization" if interested.

General assignment

In this problem, we will see the idea of relaxing a good constraint if there are no integral variables.

Problem: There are m machines $M = \{M_1, \ldots, M_m\}$ and n jobs $J = \{J_1, \ldots, J_n\}$.

There is a total available time T_i for each machine M_i.

If job j is processed on machine i, then the cost is c_{ij} and the processing time is p_{ij}.

Given the above input, the goal is to assign each job to a machine, to minimize the total cost while satisfying the time constraints, i.e. total processing on machine i is at most T_i.

LP relaxation: There is an indicator variable x_{ij} to indicate whether job j is assigned to machine i.

Let E be the set of possible pairs i,j. Initially, every pair is possible, but we will delete pairs.

$$\min_x \sum_{i \in M, j \in J} c_{ij} \cdot x_{ij}$$

$$\sum_{i \in M : ij \in E} x_{ij} = 1 \quad \forall j \in J \qquad \text{// each job is assigned to one machine}$$

$$\sum_{j \in J : ij \in E} x_{ij} \cdot p_{ij} \leq T_i \quad \forall i \in M' \subseteq M \qquad \text{// total processing time at } M_i \text{ is at most } T_i$$

$$x_{ij} \geq 0 \quad \forall ij \in E$$

It is easy to check that this is a relaxation of the problem, i.e. integral solutions are feasible.

Preprocessing: We assume $p_{ij} \leq T_i \ \forall ij \in E$, by removing those pairs i,j with $p_{ij} > T_i$. We can do this because these pairs cannot be in any feasible solution.

Theorem: Suppose there is an assignment with total cost C satisfying all the time constraints. There is a polynomial time algorithm to find an assignment with total cost C, while every time constraint is violated by at most T_i, i.e. $\sum_{j: \text{job assigned to } i} p_{ij} \leq 2T_i$.

Remark: Just to determine if there is an assignment satisfying all time constraints is NP-hard.

Iterative rounding algorithm

Let $M' = M$, where M' is the set of machines with a time constraint.

While $J \neq \emptyset$ do

① Compute a basic optimal solution x to the LP relaxation. If infeasible, return "impossible".

② Delete all pairs ij with $x_{ij}=0$ from E.

③ If $x_{ij}=1$, assign job j to machine i. Update $T_i \leftarrow T_i - p_{ij}$ and $J \leftarrow J - \{j\}$.

④ If there is a machine i with degree 1, i.e. there is only one job j with $x_{ij}>0$,

then update $M' \leftarrow M' - \{i\}$, i.e. remove the time constraint for machine i.

⑤ If there is a machine i with degree 2 and $\sum_{j: ij \in E} x_{ij} \geq 1$, then update $M' \leftarrow M' - \{i\}$.

Return the assignment.

Informally, the algorithm finds integral variables as far as possible: delete a pair if $x_{ij}=0$ and follow the LP to assign job j to machine i if $x_{ij}=1$ and then reduce the problem accordingly.

If we could always do the above, then we could solve the problem exactly.

If we couldn't find an integral variable, then we will show that either case ④ or ⑤ must happen.

Note that the machine in case ④ or ⑤ is almost settled with at most two undecided jobs.

The new idea here is to remove the time constraints on those machines, and we are going to show that the violation in these time constraints would be bounded by T_i.

The key in the analysis is to prove that without ④ or ⑤ then we can always find an integral variable in a basic solution, where the proof is again by a rank argument.

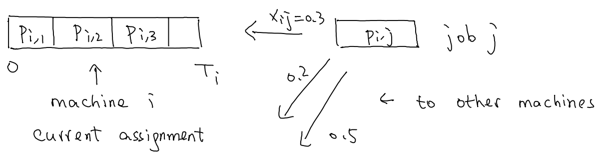
Approximation guarantee

Here we assume that the algorithm always succeeds in assigning all the jobs to machines, i.e. one of the cases applies and the algorithm won't get stuck. We will prove that this is indeed the case later.

Since we only assign a job j to a machine i if $x_{ij}=1$, a simple induction shows that the cost of our solution is no more than the objective value of the LP relaxation.

It remains to consider the violation of the time constraints.

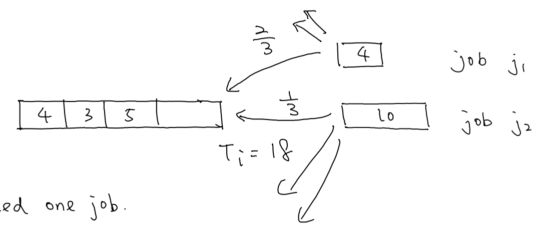
Consider case ④. The machine i was assigned some integral jobs before and now left with a fractional job.



Since machine i is only left with one job, even if we remove the time constraint, the worst case is job j will be assigned to machine i in the new LP solution.

Since $p_{ij} \leq T_i$ and the current integral assignment at machine i takes at most T_i time units, it follows that the time violation at machine i is at most T_i.

that the time violation at machine i is at most T_i.



Case ⑤ is similar but a little more involved.

Only two jobs are undecided at machine i,

but we know that $x_{ij_1} + x_{ij_2} \geq 1$, i.e. M_i is fractionally assigned one job.

We now show that the violation is at most T_i.

The LP uses time $x_{ij_1} p_{ij_1} + x_{ij_2} p_{ij_2}$, while in the worst case the integral assignment uses $p_{ij_1} + p_{ij_2}$.

So, the violation is at most

$$\begin{aligned} p_{ij_1} + p_{ij_2} - x_{ij_1} p_{ij_1} - x_{ij_2} p_{ij_2} &= (1 - x_{ij_1}) p_{ij_1} + (1 - x_{ij_2}) p_{ij_2} \\ &\leq (2 - x_{ij_1} - x_{ij_2}) T_i \qquad \text{since } p_{ij_1}, p_{ij_2} \leq T_i \\ &\leq T_i \qquad \text{since } x_{ij_1} + x_{ij_2} \geq 1. \end{aligned}$$

To summarize, assuming that the algorithm terminates, it produces an assignment with cost at most the LP objective value, while the time constraint on each machine is violated by at most T_i.

Properties of basic solutions using rank argument

To complete the proof, it remains to show that the algorithm must terminate successfully,

i.e. one of ②, ③, ④, ⑤ must apply in a basic feasible solution.

Suppose ② and ③ don't apply.

Then, there are $|E|$ linearly independent tight constraints among the job constraints and the machine constraints.

Call the set of tight job constraints J^* and the set of tight machine constraints M^*, so $|J^*| + |M^*| \geq m$.

Since $0 < x_{ij} < 1 \quad \forall ij \in E$ as ② and ③ don't apply, the degree of each job $j \in J^*$ is at least 2.

For a machine $i \in M^*$, if $\text{degree}(i) = 1$, then ④ applies.

So, suppose ④ also doesn't apply, then each machine $i \in M^*$ also has degree at least 2.

Then, $|J^*| + |M^*| \geq |E| = \frac{1}{2}\left(\sum_{j \in J} \deg(j) + \sum_{i \in M} \deg(i)\right) \geq \frac{1}{2}\left(\sum_{j \in J^*} \deg(j) + \sum_{i \in M^*} \deg(i)\right) \geq |J^*| + |M^*|$.

So, equalities hold throughout, and this implies that $\deg(j) = 0 \ \forall j \notin J^*$ and $\deg(i) = 0 \ \forall i \notin M^*$,

and furthermore the remaining graph is a disjoint union of cycles as $\deg = 2$ for remaining vertices.

Consider one such cycle C. Then $|J^* \cap C| = |M^* \cap C|$.

As each job $j \in J^* \cap C$ has fractional degree one (i.e. $\sum_{i: ij \in C} x_{ij} = 1$) by the job constraint, the total

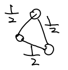
fractional value in the cycle is $|J^* \cap C| = |M^* \cap C|$, and it follows that there exists a machine

with fractional degree at least one (i.e. $\sum_{j: ij \in E} x_{ij} \geq 1$), and thus case ⑤ must apply, as desired.

To conclude, the algorithm must terminate successfully, and this completes the proof of the theorem.

Matchings in general graphs

As you may observed by now, the LP for bipartite matching is not integral for general graphs.

Of course, it has to do with an odd cycle, e.g.  , while the integral optimal is one.

Edmonds gave an exponential size LP for general matching.

$$\max_x \sum_{e \in E} w_e \cdot x_e$$

$$x(\delta(v)) \leq 1 \qquad \forall v \in V$$

$$x(E(S)) \leq \frac{|S|-1}{2} \qquad \forall S \subseteq V$$ where $|S|$ is odd and $E(S)$ is the set of edges with both endpoints in S

$$x_e \geq 0 \qquad \forall e \in E.$$

It is easy to see that it is a relaxation, as an odd set S cannot contain more than $\frac{|S|-1}{2}$ edges.

Edmonds famously proved that this LP is integral.

Also, this LP can be solved in polynomial time, using a polynomial time separation oracle.

Both results are difficult to prove and we will not do it.

See chapter 9 of "iterative methods in combinatorial optimization" if interested.

You may wonder if there is a polynomial sized LP for matchings in general graphs.

Rothvoss (2014) proved an exponential lower bound on the LP size for non-bipartite matching in a very general setting, including all the possible LPs that we could imagine.

So, perhaps surprisingly, it has to be so complicated for non-bipartite matching!

References Content is from chapter 3 and 9 of "iterative methods in combinatorial optimization".

The general assignment result is by Shmoys and Tardos.