Integer Programming
ISE 418

Lecture 17

Dr. Ted Ralphs

Reading for This Lecture

- Nemhauser and Wolsey Sections II.2.2
- Wolsey Chapter 9
- CCZ Chapter 7

Valid Inequalities from Relaxations

- In the last lecture, we saw examples of inequalities specific to a given class of integer program.
- These inequalities depended on the overall structure of the problem.
- We can also generate inequalities based on analysis of a well-known relaxation.
- Commonly arising substructures can give rise to classes valid for a wide variety of integer programs.

A General Node Packing Relaxation

- Although the clique inequalities were introduced as valid inequalities for the node packing problems, they can be applied more generally.
- Consider a general MILP with at least some binary variables.
- The node packing relaxation of an MILP is defined with respect to the so-called *conflict graph*.
- The conflict graph is a graph $G = (V, E)$, where
  - V is the set of all binary variables and their complements.
  - $\{i, j\} \in E$ if we cannot assign the variables (or complements of variables) associated with nodes i and j value 1 simultaneously.
  - The conflict graph is constructed during preprocessing (we will cover this later) and may be updated during the solution process.
- The node packing problem on this graph is a relaxation of the original MILP.
- Therefore, clique and odd hole inequalities generated with respect to this graph are valid inequalities for the MILP.

Valid Inequalities for the Knapsack Problem

- Consider the set $\mathcal{S} = \{x \in \mathbb{B}^n \mid \sum_{j=1}^{n} a_j x_k \leq b\}$ where $a \in \mathbb{Z}_+^n$ and $b \in \mathbb{Z}_+$ are positive integers.
- This is the feasible set for a 0-1 knapsack problem.
- Let $C \subset N$ be such that $\sum_{j \in C} a_j > b$ (called a *dependent set*).
- Then the inequality
$$\sum_{j \in C} x_j \leq |C| - 1$$
is a valid inequality for $\mathcal{S}$.
- These inequalities are known as *cover inequalities*.

Minimal Dependent Sets and Extended Cover Inequalities

- Consider again a knapsack set $\mathcal{S}$.
- A dependent set is *minimal* if all of its subsets are independent.
- The *extension* of a minimal dependent set C is

$$E(C) = C \cup \{k \in N \setminus C \mid a_k \geq a_j \text{ for all } j \in C\}.$$

Proposition 1. *If C is a minimal dependent set, then*

$$\sum_{j \in E(C)} x_j \leq |C| - 1$$

is a valid inequality for $\mathcal{S}$.

- Under certain conditions, the extended cover inequalities are facet-defining for $\text{conv}(S)$.

Facet-defining Cover Inequalities

Proposition 2. *If C is a minimal dependent set for $\mathcal{S}$ and (C_1, C_2) is any partition of C with $C_1 \neq \emptyset$, then $\sum_{j \in C_1} x_j \leq |C_1| - 1$ is facet-defining for $\operatorname{conv}(S(C_1, C_2))$, where*

$$S(C_1, C_2) = S \cap \{x \in \mathbb{B}^n \mid x_j = 0 \textit{ for } j \in N \setminus C, x_j = 1 \textit{ for } j \in C_2\}$$

Hence, beginning with any minimal dependent set, we can use lifting to derive a variety of facet-defining inequalities.

Inequalities from Common Relaxations

- Standard methods for generating cuts
  - Gomory, GMI, MIR, and other tableau-based disjunctive cuts.
  - Cuts from the node packing relaxation (clique, odd hole)
  - Knapsack cover cuts from knapsack relaxation.
  - Flow cover cuts from single node flow relaxation.
  - Simple cuts from pre-processing (probing, etc).
- We will discuss how to choose which ones to apply in each node in Lecture 23.
- We must in general decide on the level of effort we want to put into cut generation.

Structured Inequalities for Specific Problem Classes

- Up until the late 1990s, it was thought that solving large-scale MILPs was not possible without exploiting structure.
- Much research effort was focused on determining classes of inequalities for specific (combinational) problems that could be effectively generated.
- The situation has now changed dramatically, but it's instructive to look at one particular case in which much was discovered during this time.

Case Study: The Traveling Salesman Problem

- Consider a complete graph $G = (V, E)$.
- A *tour* in this graph is a cycle containing all nodes, i.e., a set of edges inducing a connected subgraph where the degree of every node is 2.
- Let $\mathcal{S}$ be the set of all incidence vectors of tours.
- The set $\mathcal{S}$ can be defined as follows.

$$\sum_{j:\{i,j\}\in E} x_{ij} = 2 \text{ for } i \in V, \quad (1)$$

$$\sum_{\{i,j\}\in E: i\in U, j\in N\setminus U} x_{ij} \geq 2 \text{ for } U \subset N \text{ with } 2 \leq |U| \leq |V| - 2, \quad (2)$$

$$0 \leq x_{ij} \leq 1 \text{ for } \{i,j\} \in E, \text{ and} \quad (3)$$

$$x \in \mathbb{Z}^E \quad (4)$$

where the binary variables x_{ij} represent whether i to j are adjacent in the final tour for each $(i, j) \in E$.

A Relaxation

- Let $T \supset S$ be defined by

$$T = \{x \in \mathbb{B}^n \mid x \leq x' \text{ for some } x' \in S\}$$

- We are interested in T because $\text{conv}(T)$ is full-dimensional and therefore easier to analyze.
- The dimension of $\text{conv}(S)$, on the other hand, is $|E| - |V|$ (proving this is nontrivial).
- All inequalities valid for T are also valid for $\mathcal{S}$.

Trivial Inequalities of the TSP Polytope

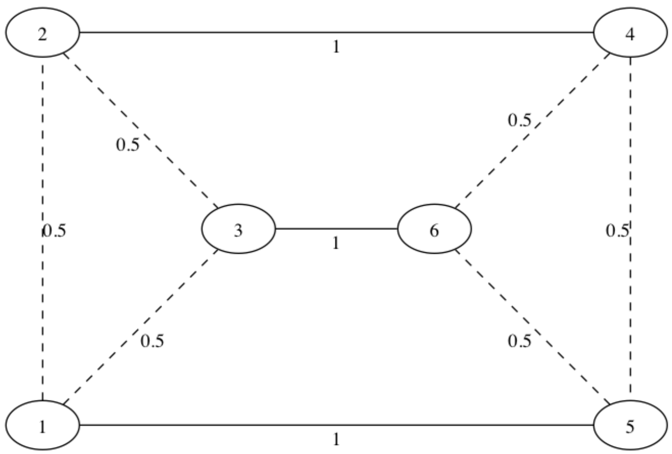
- It is easy to show that the upper and lower bound constraints are facets of $\mathrm{conv}(T)$.
- In fact, they are also facets of $\mathrm{conv}(S)$ for all graphs with $|V| \geq 5$.
- The degree constraints $\sum_{e \in \delta(\{v\})} x_e = 2$ are valid for $\mathrm{conv}(S)$.
- The inequalities $\sum_{e \in \delta(\{v\})} x_e \leq 2$ are facets of $\mathrm{conv}(T)$.
- How do we separate these inequalities?

The Subtour Elimination Constraints

- The constraints (2) are called the *subtour elimination constraints*.
- These constraints eliminate integer solutions with cycles that do not include all of the nodes.
- The subtour elimination constraints are facet-defining for $\mathrm{conv}(S)$ if $m \geq 4$ for all W with $2 \leq |W| \leq \lfloor m/2 \rfloor$.
- How do we separate these?

Further C-G Inequalities

- Even for small examples, the set of inequalities we have discussed so far do not describe the convex hull of integer solutions.
- For instance, consider the following fractional solution:



- This fractional solution satisfies all of the inequalities we've considered so far, but is not a tour.
- It is a fractional solution to the relaxation we get by considering all but the subtour elimination constraints.
- This relaxation is known as the *perfect 2-matching* problem.

The 2-Matching Inequalities

- To cut off the point from the previous slide, we consider a rank 1 C-G inequality.
- Let H be any subset of the nodes with $3 \leq |H| \leq |V| - 1$.
- Let $\hat{E} \subseteq H \times V \setminus H$ be an odd set of disjoint edges crossing the cut defined by H.
- By combining the degree constraints for the nodes in H and the non-negativity constraints for the edges in $\hat{E}$, we get the *2-matching inequalities*.

$$\sum_{e \in E(H)} x_e + \sum_{e \in \hat{E}} x_e \leq |H| + \left\lfloor \frac{|\hat{E}|}{2} \right\rfloor .$$

- These are similar to the odd set inequalities for the perfect matching problem.
- Combining these inequalities with the degree constraints yields a complete description of the convex hull of incidence vectors of perfect binary 2-matchings.

Generalizing the 2-matching Inequalities

- The 2-matching inequalities can be restated as

$$\sum_{e \in E(H)} x_e + \sum_{i=1}^{k} \sum_{e \in E(W_i)} x_e \leq |H| + \sum_{i=1}^{k} (|W_i| - 1) - \frac{k+1}{2}.$$

- To get a 2-matching inequality, we can simply take the sets W_i to be the endpoints of the edges in $\hat{E}$.
- This inequality remains valid even if the sets W_i contain more than two points.
- Each set must contain at least one node in H and one node not in H and the sets must all be disjoint.
- These inequalities are called the *comb inequalities* and are also rank 1 C-G inequalities.
- The sets W_i are called the *teeth* and the set H is called the *handle*.

Higher Rank C-G Inequalities

- We can further generalize the comb inequalities by constructing combs whose teeth are themselves combs.
- These *generalized comb inequalities* are obtained by combining the degree constraints, nonnegativity constraints, subtour elimination constraints, and comb inequalities.
- In fact, the generalized comb inequalities turn out to be facet-defining for $conv(S)$.
- By allowing the vertices of the comb to be cliques, we get the facet-defining *clique-tree inequalities*.
- Additional known classes of facet-defining inequalities.
  - Path Inequalities
  - Wheelbarrows
  - Bicycles
  - Ladders
  - Crowns

More Inequalities

- The inequalities we have discussed so far are still not enough to define the convex hull of solutions.
- There are small graphs for which these inequalities are not enough.
- Because the TSP is $\mathcal{NP}$-hard, it is unlikely that the TSP polytope has bounded rank, so it is likely that many more facets exist.
- Computationally, knowledge of just this set of inequalities has been enough to solve very large examples, however.
- The largest TSP solved to date is 24978 cities.
- This is an integer program with on the order of half a billion variables.
- Of course, it took *85 years* (yes, years!) of CPU time to solve ;).

Separation Procedures

- An *exact separation procedure* for a class of inequalities is an algorithm that is guaranteed to return an inequality of that class violated by a given point if one exists.
- A *heuristic separation procedure* is a procedure that may or may not return a violated inequality of a given class.
- The subtour elimination constraints and the 2-matching inequalities are the only classes for which we have polynomial time exact separation procedures.
- The separation problem for all other known classes of facet-defining inequalities is NP-complete.
- However, powerful heuristics are known for many classes.
- These heuristics can take a long time to run.