Integer Optimization

November 27, 2017

Contents

Chapter 1

Class 1

An integer program is a problem of the form

$$\begin{aligned} \text{Maximize } & c^T x \\ \text{subject to } & Ax \leq b \\ & x \in \mathbb{Z}^n \qquad \text{(Integrality constraint)} \end{aligned} \tag{IP}$$

for some $A \in \mathbb{R}^{n\times m}, b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$.

The LP relaxation of (IP) is the LP obtained from (IP) by dropping the integrality constraint. Since the feasible region of (IP) is contained in the feasible region of its LP relaxation, the answer for that problem is an upper bound to the IP.

Definition. An optimization problem is an oerdered pair (X, f) where X is a set and $f : X \to [-\infty, +\infty]$ is a (extended real-valued) function. It is customarily written as

$$\begin{aligned} \text{Minimize } & f(x) \\ \text{subject to } & x \in X \end{aligned} \tag{OPT}$$

Elements of X are called feasible points or feasible solutions, everything else is infeasible. The optimization problem is feasible if $X \neq \emptyset$, other it's infeasible. The objective value of $x \in X$ is $f(x)$. The optimal value of (OPT) is

$$\inf_{x\in X} f(x) \quad \in [-\infty, +\infty]$$

.

A feasible solution $\bar{x}$ is optimal if $f(\bar{x})$ is the optimal value of the problem, i.e., if $f(\bar{x}) = f(x)\ \forall x \in X$. If the optimal value is $-\infty$, the problem is unbounded.

When we write

$$\begin{aligned} \text{Maximize } & f(x) \\ \text{s.t. } & x \in X \end{aligned} \tag{OPT}$$

we are referring to the optimization problem $(X, -f)$, with the obvious changes.

Definition. A mixed integer program (MIP) is an optimization problem of the form.

$$\begin{aligned} \text{Min } & c^T x + h^T r \\ \text{s.t. } & Ax + Gr \leq b \\ & x \in \mathbb{Z}^V, \ r \in \mathbb{R}^W \end{aligned} \tag{MIP}$$

for some finite sets U, V, W, matrices $A \in \mathbb{R}^{U \times V}, G \in \mathbb{R}^{U \times W}$, and vectors $b \in \mathbb{R}^U, c \in \mathbb{R}^V, h \in \mathbb{R}^W$.

The system $Ax + Gy \leq b$ is an MIP formulation for (MIP). If $W \neq \emptyset$, the MIP is an integer program (IP). A subset P of $\mathbb{R}^V$ is a polyhedron if $P = \{x \in \mathbb{R}^V : Ax \leq b\}$ for some finite set U, matrix $A \in \mathbb{R}^{U \times V}$ and vector $b \in \mathbb{R}^V$.

A linear program (LP) is an optimization problem such that the feasible region is a polyhedron in $\mathbb{R}^V$ and whose objective function is linear from $\mathbb{R}^V$ to $\mathbb{R}$.

The LP relaxation of an MIP formulation $Ax + Gr \leq b$ with objective function $(x, r) \mapsto c^T x + h^T r$ is the LP

$$\begin{aligned} \text{Min } & c^T x + h^T r \\ \text{s.t } & Ax + Gr \leq b \\ & x \in \mathbb{R}^V, x \in \mathbb{R}^W \end{aligned} \tag{LPr}$$

If $A_1 x + G_1 r \leq b_1$ and $A_2 x + G_2 r \leq b_2$ are MIP formulations with the same feasible region, the first formulation is stronger than the second one if the LP relaxation of the first is contained in the second.

Chapter 2

Class 2

Formulation Examples

Satisfiability (CNF)

Given $n \in \mathbb{N}$ and a collection $\mathcal{C}$ of subsets of $[n]$[1] $\cup -[n]$[2], decide whether there exists $u : [n] \to \{\text{true}, \text{false}\}$ such that

$$\bigwedge_{C \in \mathcal{C}} \left(\bigvee_{\substack{i \in C \\ i > 0}} u_i \vee \bigvee_{\substack{i \in C \\ i < 0}} \neg u_{-i} \right) \text{ is true.}$$

The latter is true if the IP

$$\begin{aligned} &\text{Max (any function)} \\ &\text{s.t. } \sum_{\substack{i \in C \\ i > 0}} x_i + \sum_{\substack{i \in C \\ i < 0}} (1 - x_{-i}) \geq 1 \qquad \forall C \in \mathcal{C} \\ &\quad x \in \{0,1\}^n \text{[3]} \end{aligned} \tag{SAT}$$

is feasible.

Sudoku

Given $C \subseteq [9]^3$, decide whether it is possible to complete a sudoku board s.t. the entry i, j equals k for each $(i, j, k) \in C$.

Formulation trick: Send all of the complexity of the problem into the indices of the variables.

The variables will be

$$x_{i,j,k} := [\text{entry } i, j \text{ is filled with } k]\text{[5]} \quad \forall (i, j, k) \in [9]^3$$

[1] $[n] := \{1, \ldots, n\}$
[2] $-X := \{-x : x \in X\}$ (Minkowski notation)
[3] short for $0 \leq x \leq \mathbb{1}$[4]
[4] $\mathbb{1}$ is the vector of all ones
[5] Iverson notation: for a predicate P, $[P]$ is 1 if P is true, and 0 otherwise.

The restrictions will be:

$$\sum_{j=1}^{9} x_{i,j,k} = 1 \qquad \forall (i,j) \in [9]^2 \qquad \text{(each row has each number exactly once)}$$
$$\sum_{i=1}^{9} x_{i,j,k} = 1 \qquad \forall (j,) \in [9]^2 \qquad \text{(each column has each number exactly once)}$$
$$\sum_{a=0}^{2}\sum_{b=0}^{2} x_{i+a,j+b,k} = 1 \qquad \forall (i,j,k) \in \{1,4,7\} \times [9]^2 \qquad \text{(each 3x3 box has each number exactly once)}$$
$$\sum_{k=1}^{9} x_{i,j,k} = 1 \qquad \forall (i,j) \in [9]^2 \qquad \text{(Each entry } i,j \text{ has exactly one number)}$$
$$x_{i,j,k} = 1 \qquad \forall (i,j,k) \in C \qquad \text{(can't change the original number)}$$
$$x \in \{0,1\}^{[9]^3}$$

Expanded form of the formulation trick:

- Make heavy use of binary values.
- Avoid the use of nonbinary integer variables, unless dealing with quantities.

A Scheduling Problem

—

The Knapsack Problem

Given a set V of item types, and vectors $a : V \to \mathbb{R}_{++}$[6] and $c : V \to \mathbb{R}_{++}$ that encode the weight and value of each type, respectively, what is the maximum total value of items that can fit in a knapsack that can carry a maximum weight of $\beta \in \mathbb{R}_{++}$?

Formulation:

$$\begin{aligned} \text{Max } & c^T x \\ \text{s.t. } & a^T x \le \beta \\ & x \in \mathbb{Z}_+{}^V \end{aligned} \tag{KNPSK}$$

If items of type $U \subseteq V$ have a finite supply of $u : U \to \mathbb{Z}_+$, we may add the constraints

$$x_i \le u_i \quad \forall i \in U$$

[6] $\mathbb{R}_{++} := \{x \in \mathbb{R} : x > 0\}$

Max Sat

Given $n \in \mathbb{N}$ and a collection $\mathcal{C}$ of subsets of $[n] \cup -[n]$, find $u : [n] \to \{\text{true}, \text{false}\}$ that maximizes

$$\left|\left\{C \in \mathcal{C} : \bigvee_{\substack{i \in C \\ i > 0}} u_i \vee \bigvee_{\substack{i \in C \\ i < 0}} \neg u_{-i} \text{ is true}\right\}\right|$$

To do that, add new variables y_C, for $C \in \mathcal{C}$, which will represent whether C is satisfied.

$$\begin{aligned} &\text{Max } \mathbb{1}^T y \\ &\text{s.t. } \sum_{\substack{i \in C \\ i > 0}} x_i + \sum_{\substack{i \in C \\ i < 0}} (1 - x_{-i}) \geq y_C \quad \forall C \in \mathcal{C} \\ &\qquad x \in \{0,1\}^n \\ &\qquad y \in \{0,1\}^{\mathcal{C}} \end{aligned} \tag{MAXSAT}$$

Formulation trick: (for optimization in general)

- A variable need only have its desired meaning at optimal solutions, not at every feasible solution.

Chapter 3

Class 3

The assignment problem

Each worker in $[n]$ must be assigned to perform exactly one job in $[n]$, and when worker $i \in [n]$ performs job $j \in [n]$, he gets paid $c_{i,j}$ dollars. What is the minimum cost to get all n jobs done?

This is the minimum cost perfect matching problem. The formulation using IP is as follows:

$$
\begin{aligned}
\text{Min } & c^T x \\
\text{s.t. } & x \in \{0,1\}^{[n]\times[n]} \\
& \sum_{j\in[n]} x_{i,j} = 1 \qquad \forall i \in [n] \\
& \sum_{i\in[n]} x_{i,j} = 1 \qquad \forall j \in [n]
\end{aligned}
\tag{ASGN}
$$

The assymetric travelling salesman problem (ATSP)

Let $D = (V, A)$ be a digraph, and let $c : A \to \mathbb{R}_+$. A hamiltonian tour in D is a circuit in D that visits each vertex in V exactly once. Find a hamiltonian tour $(v_0, v_1, \ldots, v_n)$ that minimizes $\sum_{i=1}^{n} c(v_{i-1}v_i)$.

Notation: Let $S \subseteq V$. Define

$$
\begin{aligned}
&\delta^{out}(S) := \{a = uv \in A : u \in S, v \notin S\} \text{ and} \\
&\delta^{in}(S) := \delta^{out}(V \setminus S).
\end{aligned}
$$

Let $u \in V$. Define

$$
\begin{aligned}
&\delta^{out}(u) := \delta^{out}(\{u\}) \text{ and} \\
&\delta^{in}(u) := \delta^{in}(\{u\}).
\end{aligned}
$$

The IP formulation is the following, originally by Dantzig-Fulkerson-Johnson (1954).

$$
\begin{aligned}
\text{Min } & c^T x \\
\text{s.t. } & \mathbb{1}^T_{\delta^{out}(S)}{}^{1} x \geq 1 && \forall \emptyset \neq S \subsetneq V \\
& \mathbb{1}^T_{\delta^{out}(u)} x = 1 && \forall u \in V \\
& \mathbb{1}^T_{\delta^{in}(u)} x = 1 && \forall u \in V \\
& x \in \{0,1\}^A
\end{aligned} \tag{ATSP}
$$

The x vector will be the incidence vector of the chosen edges. The second and third restrictions are called degree constraints, and they guarantee that each vertex is visited exactly once and is in exactly one cycle. It does not guarantee that there is exactly one cycle. The first restriction is called subtour elimination constraint and guarantees this.

For any $x \in [0,1]^A$, we can find if any of the subtour elimination constraints are unsatisfied using minimum cut, and therefore solve the LP-relaxation of ATSP in polynomial time.

Cutting Stock

We have a supply of p large rolls of paper, each of width W. Given $w : [m] \to \mathbb{R}_+$ and $b : [m] \to \mathbb{Z}_+$, we need to meet a demand for b_i small rools of width w_i for each $i \in [m]$ to be cut out of the large rolls. Minimize the total number of large rolls used to meet this demand.

The variables will have the following meanings:

- $y_j := 1$ iff the roll j was cut
- $x_{i,j} :=$ the number of rolls of width w_i to be cut from large roll j.

The IP formulation will be as follows

$$
\begin{aligned}
\text{Min } & \mathbb{1}^T y \\
\text{s.t } & \sum_{i=1}^{m} w_i x_{i,j} \leq W y_j && \forall j \in [p] \\
& \sum_{j=1}^{p} x_{i,j} = b_i && \forall i \in [m] \\
& x \in \mathbb{Z_+}^{[m]\times[p]} \\
& y \in \{0,1\}^p
\end{aligned} \tag{CUT}
$$

The first restriction guarantees that each roll is not cut more than possible, and the second guarantees that all the desired b_i pieces are obtained for each client. The " $\leq W y_j$" restriction is a IP trick. If y_j is 0, no piece can be cut from the roll, otherwise we can cut until its size. Notice that W is a constant, so this is indeed an IP.

The usual format of this trick is using $x \geq 0$ and $x \leq U y_i$ where y_i is a 0,1 variable and U is a trivial upper bound for x. That means if y_i is 0 then so is x, and otherwise there is no actual restriction on x. This trick usually yields poor LP relaxations.

[1] If $A \subseteq B$, then $\mathbb{1}_A := x \in B \mapsto [x \in A]$ is the incidence vector of A in B.

Facility Location

A company needs to meet the demands of n customers. Customer $j \in [n]$ requires d_j units of the product (e.g. orange juice). The company may set up a subset of certain m supply facilities. Setting up facility i costs f_i, and, if set up, it can supply u_i units of the product.

Sending one unit of the product from facility i to customer j costs $c_{i,j}$. Minimize the total operation cost.

The variables will have the following meanings:

- $x_i :=$ [facility i is set up]
- $y_{i,j} :=$ how much facility i supplies j.

The IP will be as follows:

$$\begin{aligned} \text{Min } & f^T x + c^T y \\ \text{s.t. } & x \in \{0,1\}^m \\ & y \in \mathbb{R}_+^{[n]\times[m]} \\ & \sum_{j=1}^{n} y_{i,j} \le u_i x_i \quad \forall i \in [m] \\ & \sum_{i=1}^{m} y_{i,j} = d_j \qquad \forall j \in [n] \end{aligned} \tag{FL}$$

The first restriction guarantees facility i doesn't produce more product than u_i, and the second guarantees every customer gets what he asked for.

Formulation Trick: Fixed charges

To mode the "cost" of a variable x given by $x \in [0, M] \mapsto cx + [x > 0]f \in \mathbb{R}_{++}$ use the constraints

$$\begin{aligned} & x \ge 0 \\ & x \le My \\ & y \in \{0,1\} \end{aligned}$$

and formulate the cost as $cx + fy$ (for minimization). This is sometimes called big-M formulation.

Modeling Disjunctions

Let $A_i \in \mathbb{R}^{[m_i]\times[n]}$ and $b_i \in \mathbb{R}^{m_i}$ for $i \in [2]$. Suppose there exist $M_i \in \mathbb{R}$ such that $x \in \mathbb{R}^n$ and $A_i x \le b_i \Rightarrow 0 \le x \le M_i \mathbb{1}$ for $i \in [2]$.

Then the constraint that at least one of $A_1 x \le b_1$ or $A_2 x \le b_2$ can be modeled as

$$
\begin{aligned}
A_i u_i \leq y_i b_i \qquad & \forall i \in [2] \\
u_1 + u_2 = x \\
u_1,\, u_2 \in \mathbb{R}^n \\
0 \leq u_i \leq M_i y_i \mathbb{1} \quad & \forall i \in [2] \\
y \in \{0,1\}^2 \\
y_1 + y_2 = 1
\end{aligned}
\tag{DSJ}
$$

Given $\bar{x}$ such that $A_1\bar{x} \leq b_1$, then $u_1 = \bar{x}$, $u_2 = 0$, $y_1 = 1$, $y_2 = 0$ works as a solution to (DSJ). Given a solution to (DSJ). Assume $y_1 = 1$, then $y_2 = 0$, $u_1 = x$, $u_2 = 0$ and $A_1 x \leq b1$ is satisfied.

Exercise 1. *Formulate and prove a claim that states that the formulation is correct.*

Chapter 4

Class 4

Hypergraphs

Let V be a ground set. Let $\mathcal{F} \subseteq \mathcal{P}(V)$[1]. A set $U \subseteq V$ is a packing of $\mathcal{F}$ if $|U \cap F| \leq 1$ for each $F \in \mathcal{F}$. A set $U \subseteq V$ is a covering of $\mathcal{F}$ if $|U \cap F| \geq 1$ for all $F \in \mathcal{F}$.

The ordered pair $\mathcal{H} := (V, \mathcal{F})$ is called a hypergraph. The elements of $\mathcal{F}$ are the edges of $\mathcal{H}$. The $V \times \mathcal{F}$ incidence matrix of $\mathcal{H}$ is the matrix $B_{\mathcal{H}} \in \{0,1\}^{V \times \mathcal{F}}$ such that $[B_{\mathcal{H}}]_{v,F} := [v \in F]$ for all $(v, F) \in V \times \mathcal{F}$. The $\mathcal{F} \times V$ incidence matrix is $B_{\mathcal{H}}^T$.

The set packing problem for $\mathcal{H}$ is: given $w \in \mathbb{R}_+^V$, maximize $w^T \mathbb{1}_S$ such that S is a packing of $\mathcal{H}$ (of $\mathcal{F}$). It may be formulated as an IP as follows

$$\begin{aligned} \text{Max } & w^T x \\ \text{s.t. } & B_{\mathcal{H}}^T x \leq \mathbb{1} \text{ (that is, } \sum_{v \in F} x_v \leq 1 \ \forall F \in \mathcal{F}) \\ & x \in \{0,1\}^V. \end{aligned} \tag{PCK}$$

The set covering problem for $\mathcal{H}$ is: given $w \in \mathbb{R}_+^V$, minimize $w^T \mathbb{1}_S$ such that S is a covering of $\mathcal{H}$ (of $\mathcal{F}$). It may be formulated as an IP as follows

$$\begin{aligned} \text{Min } & w^T x \\ \text{s.t. } & B_{\mathcal{H}}^T x \geq \mathbb{1} \\ & x \in \{0,1\}^V. \end{aligned} \tag{PCK}$$

These are called standard IP formulations of the set packing and set covering problems.

Formulations

Let $G = (V, E)$ be a graph.

The set packing problem for G is the maximum weight stable (independent) set problem.

[1] $\mathcal{P}(V) := \{S : S \subseteq V\}$ is the power set of V.

($S \subseteq V$ is stable if $G[S]$ has no edges)

The set covering problem for G is the minimum weight vertex cover problem. ($C \subseteq V$ is a vertex cover if $G[V \setminus C]$ is stable)

The set packing problem for $(E, \{\delta(u) : u \in V\})$ is the maximum weightmatching problem ($F \subseteq E$ is a matching if $deg_{(V,F)} \leq \mathbb{1}$)

The set covering problem for $(E, \{\delta(u) : u \in V\})$ is the minimum weightedge cover problem ($F \subseteq E$ is an edge cover if $deg_{(V,F)} \geq \mathbb{1}$)

The set packing problem for $(V, \{K \subseteq V : K \text{ is a clique in } G\})$ is the maximum weight stable ser problem.

The minimum weight cut problem is the set covering problem for $(E, \{F \subseteq E : (V, F) \text{ is a tree}\})$. ($F \subseteq E$ is a cut of G if $F = \delta(S)$ for some $\emptyset \neq S \subsetneq V$)

The set covering problem for $(E, \{E \setminus \delta(S) : \emptyset \neq S \subsetneq V\})$ is the minimum weight odd cycle problem.

The minimum weight dominating set problem is the set covering problem for $(V, \{\{u\} \cup N(u) : u \in V\})$, where $N(u) := \{v : uv \in E\}$. ($S \subseteq V$ is dominating if every vertex in $V \setminus S$ has a neighbor in S)

Fix distinct $s, t \in V$. The set covering problem for $(E, \{E(P) : P \text{ is an } st\text{-path}\})$ is the minimum st-cut problem. ($F \subseteq E$ is an st-cut of G if $F = \delta(S)$ for some $S \subseteq V$ such that $s \in S$, and $t \notin S$).

Chapter 5

Class 5

Recall. A hypergraph is an ordered pair $(V, \mathcal{E}) =: \mathcal{H}$ where V is a (finite) set and $\mathcal{E} \subseteq \mathcal{P}(V)$. Let $V(\mathcal{H}) := V$ and $\mathcal{E}(\mathcal{H}) := \mathcal{E}$.

Let $U \subseteq V$. We say that

- U is a packing of $\mathcal{H}$ if $|U \cap F| \leq 1 \quad \forall F \in \mathcal{E}$.
- U is a covering (or vertex cover) of $\mathcal{H}$ if $|U \cap F| \geq 1 \quad \forall F \in \mathcal{E}$.

We have seen that many classical combinatorial optimization problems on graphs may be formulated as packing or covering problems.

Exercise 2. *Let $G = (V, E)$ be a graph with non-negative weights at the edges, and let $S \subseteq V$. A Steiner tree of G with respect to S is a subgraph of G that is a tree and whose vertex set contains S. (Thus, the Steiner trees of G w.r.t V are exactly the spanning trees of G)*

Formulate the problem of finding a minimum cost Steiner tree of G w.r.t. S as a covering problem. Prove it.

Exercise 3. *Let $G = (V, E)$ be a graph and let $T \subseteq V$. We say that $F \subseteq E$ is a T-join of G if $|F \cap \delta(v)| \equiv [v \in T] \mod 2$ for any $v \in V$. (Thus, every perfect matching of G is a V-join of G)*

Formulate and prove the provlem of finding a minimum cost T-join of G as a set-covering problem. All edge costs are non-negative.

Let $\mathcal{H} = (V, \mathcal{E})$ be a hypergraph. Define

$$H^{\min} := (V, \{F \in \mathcal{E} : \nexists E \in \mathcal{E} \text{ s.t. } E \subsetneq F\}),$$

the inclusion-wise minimal elements of $\mathcal{E}$, and

$$H^{\uparrow} := (V, \{F \subseteq V : \exists E \in \mathcal{E} \text{ s.t. } E \subseteq F\}).$$

Exercise 4. *If $\mathcal{H}, \mathcal{H}_1$ and $\mathcal{H}_2$ are hypergraphs*

(i) $\mathcal{H}^{\uparrow\uparrow} = \mathcal{H}^{\uparrow}$ *and* $\mathcal{H}^{\min\min} = \mathcal{H}^{\min}$.

(ii) $\mathcal{H}^{\uparrow\min} = \mathcal{H}^{\min}$ *and* $\mathcal{H}^{\min\uparrow} = \mathcal{H}^{\uparrow}$.

(iii) $\mathcal{H}_1 \subseteq \mathcal{H}_2 \Rightarrow \mathcal{H}_1^{\uparrow} \subseteq \mathcal{H}_2^{\uparrow}$.

(iv) $\mathcal{H}_1 \subseteq \mathcal{H}_2^{\uparrow}$ *and* $\mathcal{H}_2 \subseteq \mathcal{H}_1^{\uparrow} \Rightarrow \mathcal{H}_1^{\min} = \mathcal{H}_2^{\min}$.

(v) $\mathcal{H}_1 \subseteq \mathcal{H}_2^{\uparrow} \Rightarrow \mathcal{H}_1^{\uparrow} \subseteq \mathcal{H}_2^{\uparrow}$.

Let $\mathcal{H}_1$ and $\mathcal{H}_2$ be hypergraphs. $\mathcal{H}_1 \subseteq \mathcal{H}_2$ if $V(\mathcal{H}_1) \subseteq V(\mathcal{H}_2)$ and $\mathcal{E}(\mathcal{H}_1) \subseteq \mathcal{E}(\mathcal{H}_2)$. This partially orders hypergraphs.

A hypergraph $\mathcal{H}$ is a clutter if, whenever $A, B \in \mathcal{E}$ are distinct, we have $A \subsetneq B$ and $B \subsetneq A$. For instance, $\mathcal{H}^{\min}$ is a clutter. In fact, a hypergraph $\mathcal{H}$ is a clutter iff $\mathcal{H} = \mathcal{H}^{\min}$.

The blocker of a hypergraph $\mathcal{H}$ is the clutter

$$b(\mathcal{H}) := (V, \{C \subseteq V : C \cap F \neq \emptyset \quad \forall F \in \mathcal{E}\})^{\min}.$$

Note: the edges of $b(\mathcal{H})^{\uparrow}$ are exactly the vertex covers of $\mathcal{H}$. Also, $b(\mathcal{H}) = b(\mathcal{H}^{\min}) = b(\mathcal{H}^{\uparrow})$. (exercise: prove it)

Exercise 5. *If* $\mathcal{H}_1$ *and* $\mathcal{H}_2$ *are hypergraphs on the same vertex set, then*

$$\mathcal{H}_1 \subseteq \mathcal{H}_2 \Rightarrow b(\mathcal{H}_2)^{\uparrow} \subseteq b(\mathcal{H}_1)^{\uparrow}.$$

Set covering problems are the problems of finding a minimum weight/cost edge of the blocker of some hypergraph.

Theorem 6. *(Lawler '66, Edmonds and Fulkerson '70)*
Let $\mathcal{H}$ *be a hypergraph. Then* $b(b(\mathcal{H})) = \mathcal{H}^{\min}$. *In particular, if* $\mathcal{H}$ *is a clutter, then* $b(b(\mathcal{H})) = \mathcal{H}$.

Proof. Let's show that $b(b(\mathcal{H}))^{\uparrow} = H^{\uparrow}$ (A).

Let $F \in \mathcal{E}(\mathcal{H}^{\uparrow})$. Let $E \in \mathcal{E}(b(\mathcal{H}))$, then $F \cap E \neq \emptyset$. So F is a covering of $b(\mathcal{H})$, that is, $F \in \mathcal{E}(b(b(\mathcal{H}))^{\uparrow})$. So we proved $\mathcal{E}(H^{\uparrow}) \subseteq \mathcal{E}(b(b(\mathcal{H}))^{\uparrow})$.

We want to prove $\mathcal{E}(b(b(\mathcal{H}))^{\uparrow}) \subseteq \mathcal{E}(H^{\uparrow}) \iff \mathcal{P}(V) \setminus \mathcal{E}(b(b(\mathcal{H}))^{\uparrow}) \supseteq \mathcal{P}(V) \setminus \mathcal{E}(\mathcal{H}^{\uparrow})$.

Let $U \in \mathcal{P}(V) \setminus \mathcal{E}(\mathcal{H}^{\uparrow})$, then

$$\begin{aligned} U \notin \mathcal{E}(\mathcal{H}^{\uparrow}) &\iff \nexists F \in \mathcal{E}(\mathcal{H}) \text{ s.t. } F \subseteq U \\ &\iff \forall F \in \mathcal{E}(\mathcal{H}) \quad F \subsetneq U, \text{ that is, } (V \setminus U) \cap F \neq \emptyset \\ &\iff V \setminus U \text{ is a covering of } \mathcal{H} \\ &\iff V \setminus U \in \mathcal{E}(b(\mathcal{H})^{\uparrow}) \end{aligned}$$

That implies U is a not a covering of $b(\mathcal{H}^{\uparrow})$, since $U \cap (V \setminus U) = \emptyset$ and $V \setminus U \in \mathcal{E}(b(\mathcal{H})^{\uparrow})$. That means $U \notin \mathcal{E}(b(b(\mathcal{H})^{\uparrow})^{\uparrow}) = \mathcal{E}(b(b(\mathcal{E}))^{\uparrow})$, that is, $U \in \mathcal{P}(V) \setminus \mathcal{E}(b(b(\mathcal{H}))^{\uparrow})$, so (A) is proved.

If we apply min to both sides of (A), we get $b(b(\mathcal{H})) = H^{\min}$. □

A clutter and a blocker are said to form a blocking pair.

Class 23/08

Corollary 7. *Let $\mathcal{H}$ and $\mathcal{B}$ be a hypergraphs on the same vertex set. If $\mathcal{B} \subseteq b(\mathcal{H})^{\uparrow}$ and $b(\mathcal{B})^{\uparrow} \subseteq \mathcal{H}^{\uparrow}$, then $b(\mathcal{H}) = \mathcal{B}^{\min}$, so $\mathcal{H}^{\min}$ and $\mathcal{B}^{\min}$ for a blocking pair.*

Proof. We will prove $b(\mathcal{H})^{\uparrow} = \mathcal{B}^{\uparrow}$.

Let's prove '$\subseteq$'. We have that $b(\mathcal{B})^{\uparrow} \subseteq \mathcal{H}^{\uparrow} \overset{\text{Ex5}}{\Longrightarrow} b(\mathcal{H}^{\uparrow})^{\uparrow} \subseteq b(b(\mathcal{B})^{\uparrow})^{\uparrow} = b(b(\mathcal{B}))^{\uparrow} \overset{\text{Thm6}}{=} \mathcal{B}^{\uparrow}$.

Let's prove '$\supseteq$'. By hypothesis and Exercise 4, $\mathcal{B} \subseteq b(\mathcal{H})^{\uparrow} \implies \mathcal{B}^{\uparrow} \subseteq (b(\mathcal{H})^{\uparrow})^{\uparrow} = b(\mathcal{H})^{\uparrow}$.

Now, $b(\mathcal{H})^{\uparrow} = \mathcal{B}^{\uparrow} \implies b(\mathcal{H}) = b(\mathcal{H})^{\min} \subseteq \mathcal{B}^{\min}$. □

Exercise 8. *For the hypergraphs $\mathcal{H}$ and $\mathcal{B}$ below, prova that $\mathcal{H}^{\min}$ and $\mathcal{B}^{\min}$ form a blocking pair.*

(a) If $G = (V, E)$ is a graph, take

$$\mathcal{H} := (E, \{E(T) \mid T \textit{ is a spanning tree of } G\})$$

and

$$\mathcal{B} := (E, \{\delta(S) \mid \emptyset \neq S \subsetneq V\}).$$

(b) If $G = (V, E)$ is a graph and $s, t \in V$ are distinct, take

$$\mathcal{H} := (E, \{E(P) \mid P \textit{ is an st-path in } G\})$$

and

$$\mathcal{B} := (E, \{\delta(S) \mid s \in S \subseteq V \setminus \{t\}\}).$$

(c) If $G = (V, E)$ is a graph, take

$$\mathcal{H} := (E, \{E(C) \mid C \textit{ is an odd cycle in } G\})$$

and

$$\mathcal{B} := (E, \{E \setminus \delta(S) \mid \emptyset \neq S \subsetneq V\}).$$

A clutter $\mathcal{H}$ is binary if $|F \cap B|$ is odd for all $(F, B) \in \mathcal{E}(\mathcal{H}) \times \mathcal{E}(b(\mathcal{H}))$.

Exercise 9. *Determine (and prove) which of the clutters in exercise 8 are binary.*

Exercise 10. *Let $H = (V, \mathcal{E})$ be a hypergraph with $V = [n]$. Suppose $\mathcal{H}$ is "given" by an oravle that answers queries of the form: "Given $i, j \in V$, is there an edge $F \in \mathcal{E}(\mathcal{H})$ such that $i, j \in F$?". Provide a polynomial-time algorithm (with polynomial number of queries to the oracle) that builds a graph $G = (V, E)$ such that the packing problem for $\mathcal{H}$ is* <u>*equal*</u> *to the maximum weighted stable* *set problem for G. (Equality between optimization problems means equal sets and optimization functions)*

The Branch-and-Bound Method

Description of the Branch-and-Bound algorithm:

input. An MIP formulation $Ax + Gy \leq b$ and $x \in \mathbb{Q}^n, h \in \mathbb{Q}^r$.

output. If the algorithm terminates, it returns an optimal solution for the MIP or answers that no feasible solution exists.

The algorithm keeps

(i) a finite list L of nodes N_i, with $i \in \mathbb{N}$.

(ii) each node N_i in L has an associate LP, denoted by LP_i with optimal value z_i.

(iii) a subset $\overline{X}$ of at most one feasible solution $(\bar{x}, \bar{y})$ for the MIP with objective value $\bar{z}$. If $\overline{X} = \emptyset$, we have $\bar{z} = -\infty$.

Description:

▷ 0. Initialize
$L = \{N_0\}$, LP_0 is the LP relaxation of $Ax + Gy \leq b$ with objective function $c^T x + h^T y$.
$\overline{X} = \emptyset$, $\bar{z} = -\infty$.
▷ 1. Terminate?
if $L = \emptyset$ **then**
  if $\overline{X} = \emptyset$ **then**
    answer "infeasible".
  else
    return the unique element of $\overline{X}$.
▷ 2. Select Node
Choose a node N_i from L.
$L = L \setminus N_i$.
▷ 3. Bound
Solve LP_i (assuming it is bounded)
if LP_i is infeasible **then** ▷ Pruned by infeasibility
  Go to "1. Terminate?"
elseLet $(x^\star, y^\star)$ be an optimal solution for LP_i with object value z_i.
▷ 4. Prune?
if $z_i \leq \bar{z}$ **then** ▷ Pruned by Bound
  Go to "1. Terminate?"
if $x^\star \in \mathbb{Z}^n$ **then**
  $\bar{z} = z_i$.
  $\overline{X} = \{(x^\star, y^\star)\}$.

Go to "1. Terminate?" ▷ Pruned by integrality

▷ 5. Branch

Let $j \in [n]$ such that $x_j^\star \notin \mathbb{Z}$.

Form two new LPs from LP_i by adding the constraints $x_j \leq \lfloor x_j^\star \rfloor$ to one and $x_j \geq \lceil x_j^\star \rceil$ to the other and add the two corresponding nodes to L.

Go to "1. Terminate?"

An example run on

$$\begin{aligned} \text{Maximize } & 55x_1 + 21x_2 \\ \text{Subject to } & -x_1 + x_2 \leq 2 \\ & 8x_1 + 2x_2 \leq 17 \\ & x \in {\mathbb{Z}_+}^2 \end{aligned} \qquad \text{(MIP)}$$