COURSE # 11 **CUTTING PLANES**

Thursday, January 11, 2024 5:55 PM

TOMORROW 16:20 - 17:50

EXAM COURSE PART 19 - 24 JANUARY

ILP: - branch an bound
- use the LP relaxation at each node as a branching heuristic (pruning)

- need a good LP formulation

formulation minimizes int gap $\rightarrow \rho = \dfrac{OPT(IP)}{OPT(LP)} \geq 1$

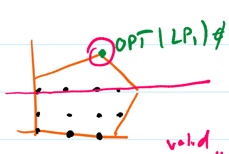
(minimization)

$\rho = 1$ $OPT(IP) = OPT(LP)$

A way to improve formulation before branching.

$$\begin{array}{ccccc} IP_1 & \xrightarrow{+I_1} & IP_2 & \rightarrow & IP_3 \\ \downarrow & & \downarrow & & \downarrow \\ LP_1 & & LP_2 & & LP_3 \end{array}$$

$OPT(LP_1) \notin \mathbb{Z}^n$



inequality
↓
valid for all solutions
integral

invalid for $OPT(LP_1)$

Cutting plane - inequality valid for all integral solns
- fails for the current OPTIMAL LP solution

cuts the OPTIMAL LP soln from the soln polyhedron

$OPT(LP_2) < OPT(LP_1)$

$OPT(IP_2) = OPT(IP_1)$

$\rho_2 < \rho_1$

Hopefully $OPT(LP_n) = OPT(IP_n) \longrightarrow$ simplex gives me the optimal integral soln.

Plan - stay at the root node of the branching tree as much as possible

Plan – stay at the root node of the branching tree as much as possible

(avoid branching + cutting planes)

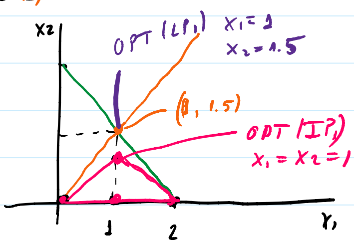
- when adding cutting planes stops significantly improving soln → BRANCH

Exp

$$(IP_1)\begin{cases} \max(x_2) \\ 3x_1 + 2x_2 \le 6 \quad (1) \\ -3x_1 + 2x_2 \le 0 \quad (2) \\ x_1, x_2 \ge 0 \\ x_1, x_2 \text{ integral} \end{cases}$$

→ LP_1

drop integrality constraint



$OPT(LP_1) > OPT(IP_1)$

(1.5) (1)

integral solns

(0,0) (1,1)

(1,0) (2,0)

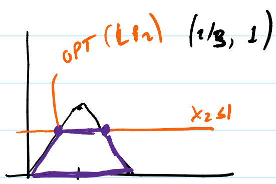
frac soln

(1, 1.5)

$x_2 \le 1$

$x_2 \le x_1$

$x_1 + x_2 \le 2$



$OPT(LP_2) = 1$

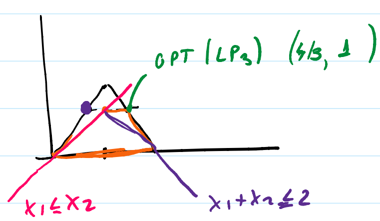
$< OPT(LP_1) = 1.5$

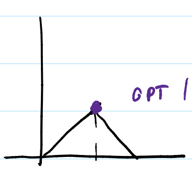
$-3x_1 + 2x_2 = 0$

$x_2 = 1$

$3x_1 = 2$

$x_1 = 2/3$





OPT (LP4) (1, 1)

integral!

⇓

$OPT(IP_1) \to (1,1)$

$$\boxed{\text{Solved IP without branching!}}$$

Branch & Cut

start with initial LP relaxation of IP

while (OPT (LP) not integral & can generate CP)

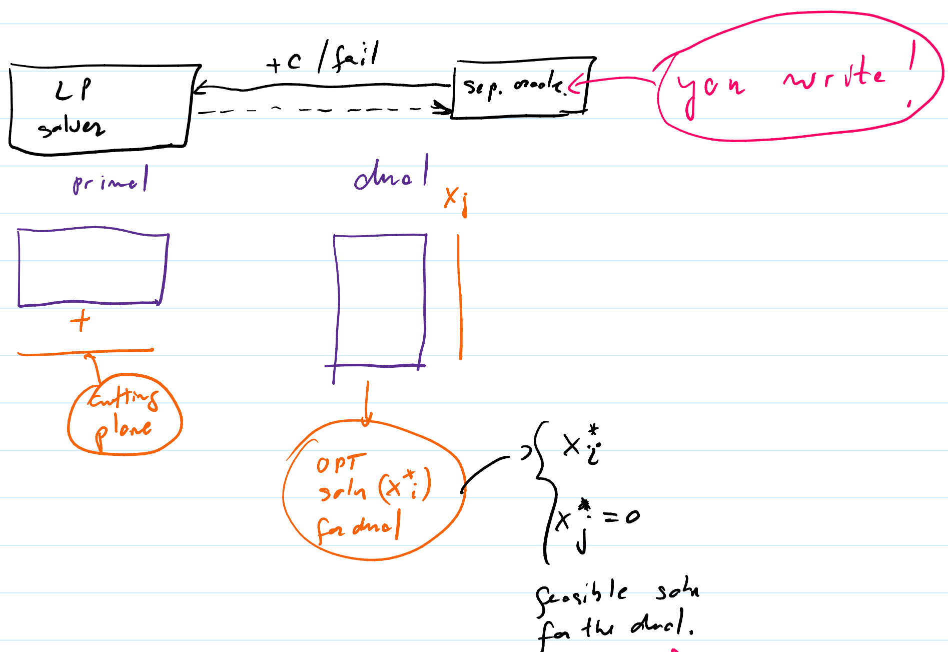
add cutting plane

resolve.

branch. //otherwise.

IN PRACTICE Might want to branch when last cutting plane failed to significantly improve soln. for some time.

Procedure that generates cutting planes SEPARATION ORACLE



hot start the recomputation of OPT for the modified LP

CONCLUSION Don't need to resolve from scratch the extended LP.

(1) GOMORY CUTTING PLANES

(RALPH GOMORY) '50s

INVENTED IN THE '50s

↓

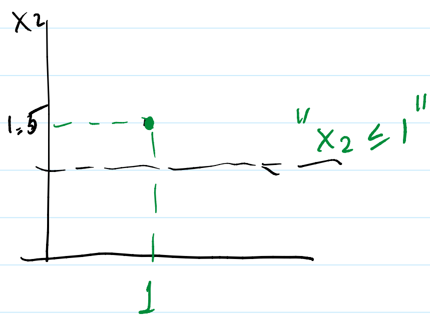
THEORETICAL CURIOSITY

mid 90's GERARD CORNUÉJOLS

(C.M.U., in the us)

showed how to use G.C.? in branch & cut

STANDARD METHOD (automatic)



$\left. \begin{array}{l} x_2 \le 1.5 \\ x_2 \text{ integral} \end{array} \right\} \Rightarrow \underbrace{x_2 \le \lfloor 1.5 \rfloor}_{= 1}$

opt soln for LP (bfs) $\quad x_j = 0 \quad j \in N$

$x \notin \mathbb{Z}^n \quad \exists\, i \in B \quad x_i \notin \mathbb{Z}$

TABLEAU

$$\boxed{x_i = \underbrace{\bar{b}_i}_{\notin \mathbb{Z}} - \sum_{j \in N} \bar{a}_{ij} x_j}$$

$x_i = \bar{b}_i$

$$\boxed{x_i + \sum_{j \in N} \bar{a}_{ij} x_j = \bar{b}_i} \quad (1)$$

Take $(x_1, \ldots x_n)$ integral soln Want to modify (1)

to derive an inequality that holds for <u>every</u> integral soln.

$$x_i + \sum_{j \in N} \lfloor \bar{a}_{ij} \rfloor x_j \le x_i + \sum \bar{a}_{ij} x_j = \bar{b}_i$$

integral !

$$\left.\begin{array}{l} x \le a \\ x \text{ integer} \end{array}\right\} \Rightarrow x \le \lfloor a \rfloor$$

Gomory cutting plane

$$\boxed{x_i + \sum_{j \in N} \lfloor \bar{a}_{ij} \rfloor x_j \le \lfloor \bar{b}_i \rfloor}$$

not true for OPT LP

$x^*_i = \bar{b}_i \notin \mathbb{Z}$

$> \lfloor \bar{b}_i \rfloor$

true for all integral solns.

How do we generate it? → from the optimal simplex tableau !

Practical conclusion

Gomory c.p generator → callback into solver

↓

Julia & JuMP

PRACTICAL (equivalent) version

$$(1) \quad x_i + \sum_{j \in N} \lfloor \bar{a}_{ij} \rfloor x_j \le \lfloor \bar{b}_i \rfloor \qquad \text{(theoretical)}$$

$$(2) \quad x_i + \sum_{j \in N} \bar{a}_{ij} x_j = \bar{b}_i \qquad \text{(optimal LP)}$$

$$\bar{b}_i - \lfloor \bar{b}_i \rfloor \ge \sum_{j \in N} \left(\bar{a}_{ij} - \lfloor \bar{a}_{ij} \rfloor \right) x_j \qquad (2) - (1)$$

or $\qquad \{x\} = x - \lfloor x \rfloor$

$$\boxed{\{\bar{b}_i\} \ge \sum_{j \in N} \{\bar{a}_{ij}\} x_j} \qquad \text{practical version.}$$

$\Downarrow$

comes from simplex tableau for OPT(LP)

$\Downarrow$

can generate it automatically.

IN PRACTICE

Add to the simplex tableau equality

$$\boxed{\sum_{j \in N} \{\bar{a}_{ij}\} x_j + y = \{\bar{b}_i\}} \qquad \text{DUAL } (\alpha)$$

$$y \ge 0 \quad \text{(new variable)}$$

Dual soln feasible soln for the extended pb.

$LP_1 \qquad (D_1)$

$+ CP_1$

$LP_2 \qquad (D_1, \alpha = 0) \rightarrow$ initial BFS

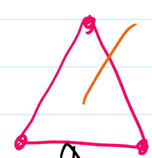
LP_2 $\quad (D_1, \alpha = 0) \rightarrow$ initial BFS for dual of LP_2.

Practical problem: If $\{\bar{a}_{ij}\}, \{\bar{b}_i\}$ very small

$\Downarrow$

numerical stability issues

(2) Polyhedral cuts



convex hull of the set of integral solns.

facet of this polyhedron

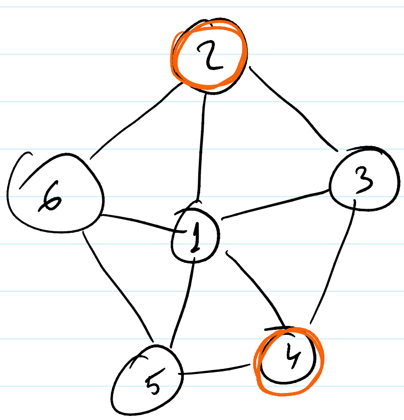
$\downarrow$

polyhedral cut

$x_1 \le 1$ not polyhedral cut

$$\left.\begin{array}{l} x_1 \not\geq x_2 \\ x_1 + x_2 \le 2 \end{array}\right\} \rightarrow \text{polyhedral cuts}$$

EXAMPLE MAXIMUM INDEPENDENT SET



$S \subseteq V$ is called indep. set

$\forall x, y \in S \quad x \not\sim y$

MIS want S indep. set $|S|$ maximal

Exp $\{2, 4\}$ Indep. set maximal

MIS as IP

$x_1, \dots x_6 \in \{0,1\}$

$$x_i = \begin{cases} 1 & i \in S \\ 0 & i \notin S \end{cases}$$

$$(IP) \begin{cases} \max (x_1 + \dots + x_6) \\ x_1 + x_2 \leq 1 & (1,2) \in E \\ x_2 + x_3 \leq 1 & (2,3) \in E \\ \vdots \\ x_i + x_j \leq 1 & (i,j) \in E \\ \vdots \\ x_5 + x_6 \leq 1 & (5,6) \in E \\ x_1, \dots x_6 \in \{0,1\} \end{cases}$$