

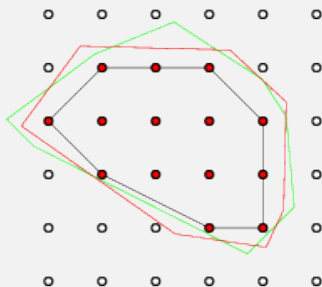
Branch and Cut

Prof. Eduardo Gontijo Carrano

Departamento de Engenharia Elétrica – DEE
Universidade Federal de Minas Gerais – UFMG

Introdução

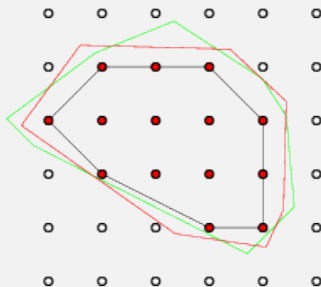
- Um mesmo problema pode ser representado por formulações distintas.
- Existem formulações mais fortes que outras:
 - Formulações mais fortes geram melhores limites.
 - Resultados mais próximos da solução ótima inteira.



- Em cada nó do algoritmo de Branch and Bound é resolvida a relaxação linear do modelo.
 - As restrições de integralidade são ignoradas.
- O valor obtido, chamado de limitante dual, é utilizado para realizar podas na árvore.
 - Quanto melhor forem os limitantes duais, mais podas são realizadas:
 - Menos nós a serem explorados.

Introdução

- Se a envoltória convexa das soluções inteiras for encontrada, é possível resolver o problema de programação linear inteira como um problema de programação linear, sem as restrições de integralidade.
- Logo, é interessante encontrar desigualdades válidas que aproximem a região factível do problema relaxado da envoltória convexa das soluções inteiras.



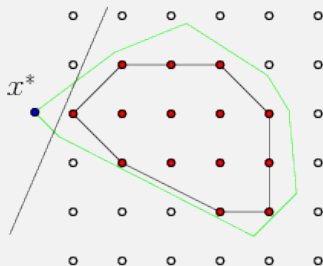
Introdução

- Para problemas *NP*-difíceis, não é possível encontrar todas as desigualdades que definem a envoltória convexa em tempo polinomial.
- O número de desigualdades pode ser muito grande, o que torna a resolução da relaxação linear lenta.
- Nem toda desigualdade válida leva a melhores limitantes duais.

Proposta

Adicionar as desigualdades iterativamente, de forma a produzir modelos cada vez mais fortes que gerem melhores limitantes duais.

Planos de Corte



- O ponto ótimo (azul) da relaxação linear (polígono verde) não possui coordenadas inteiras, não sendo portanto uma solução válida para o problema.
- Devem ser definidos hiperplanos que excluam esse ponto da região factível do problema, mas sem excluir pontos que são factíveis no problema original.
 - Tais hiperplanos, chamados de planos de corte, são definidos por meio de desigualdades que devem ser satisfeitas por todos os pontos viáveis (vermelhos).

Exemplos de Planos de Corte

Mochila 0-1:

- Seja um conjunto de N itens, cada um com sua utilidade c e seu peso w . O problema da mochila 0-1 consiste em identificar o subconjunto de itens que representa a maior utilidade, sem exceder a capacidade da mochila W .

$$\begin{aligned} & \max \sum_{i=1}^N c_i \cdot x_i \\ \text{sujeito a: } & \begin{cases} \sum_{i=1}^N w_i \cdot x_i \leq W \\ x_i \in \{0, 1\} \quad \forall i \in \{1, \dots, N\} \end{cases} \end{aligned}$$

Exemplos de Planos de Corte

Mochila 0-1:

- Considere 4 itens tais que a restrição de peso seja:

$$x_1 - 2x_2 + 5x_3 + 2x_4 \leq 4$$

- Possíveis planos de corte:
 - Se $x_3 = 1$, então obrigatoriamente $x_2 = 1$. Logo:
 - Possível plano de corte: $x_3 \leq x_2$.
 - Se $x_3 = x_4 = 1$, então a restrição é necessariamente infactível. Logo:
 - Possível plano de corte: $x_3 + x_4 \leq 1$.

Exemplos de Planos de Corte

Mochila 0-1:

- Considere 6 itens tais que a restrição de peso seja:

$$18x_1 + 22x_2 + 13x_3 + 10x_4 + 12x_5 + 7x_6 \leq 28$$

- Possíveis planos de corte:
 - Conjuntos de cobertura mínima. Exemplos:
 - $x_1 + x_2 \leq 1$.
 - $x_1 + x_3 \leq 1$.
 - $x_3 + x_4 + x_5 \leq 2$.
 - $x_4 + x_5 + x_6 \leq 2$.
 - etc...

- Em alguns casos identificar se uma desigualdade representa um plano de corte é, por si só, um problema *NP*-difícil.
- Em geral, são utilizadas famílias de desigualdades válidas a priori:
 - Chvátal-Gomory
 - Clique
 - Cover
 - Flow cover
 - Flow path
 - etc...

Planos de Corte

Algoritmo

Dados um problema de programação linear inteira

$\min \{ \sum_i c_i x_i : x \in P \cap \mathbb{Z} \}$ e uma família de desigualdades válidas \mathcal{F} :

Algoritmo de Planos de Corte

início

$k \leftarrow 0$;

$P_0 \leftarrow P$;

repita

 Resolva $\bar{w}_k = \min \{ \sum_i c_i x_i : x \in P_k \}$ obtendo \bar{x}_k ;

$D_k \leftarrow \text{resolverSeparacao}(\bar{x}_k, \mathcal{F})$;

se $D_k \neq \emptyset$ **então**

$P_{k+1} \leftarrow P_k \cap D_k$;

$k \leftarrow k + 1$;

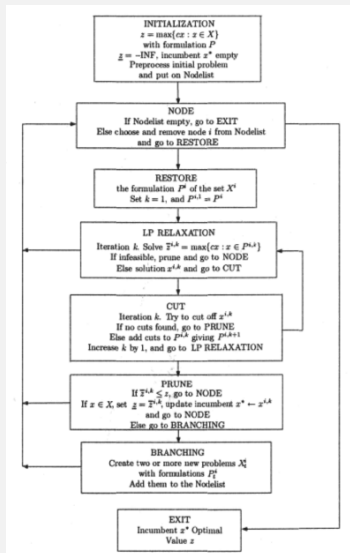
até $D_{k-1} = \emptyset$;

Se o algoritmo convergir antes que uma solução inteira tenha sido obtida, o resultado é uma formulação mais forte.

Branch and Cut

- O algoritmo de Branch and Cut é a combinação do algoritmo de Branch and Bound (B&B) com Planos de Corte.
- Em cada nó são gerados cortes, fazendo com que os subproblemas gerem melhores limites duais.
- Tipos de corte
 - Globais: válidos para todos os nós na árvore B&B.
 - Locais: válidos para o nó da árvore B&B em que foi gerado.

Branch and Cut Algorithm



Branch and Cut

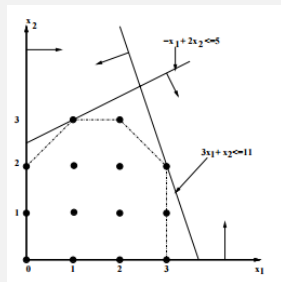
Exemplo

$$\min z = -6x_1 - 5x_2$$

$$3x_1 + x_2 \leq 11$$

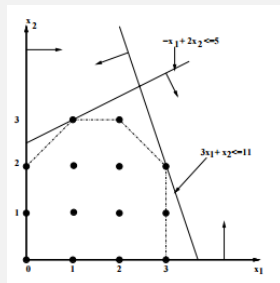
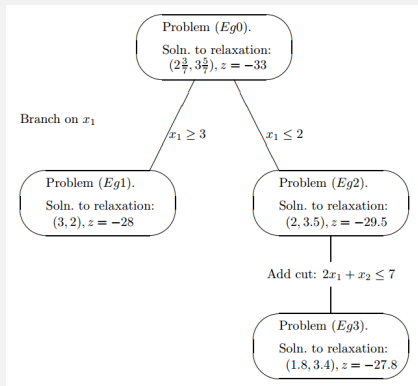
$$-x_1 + 2x_2 \leq 5$$

$$x_1, x_2 \geq 0, \text{ integer}$$



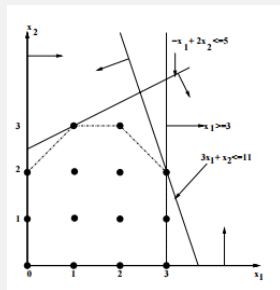
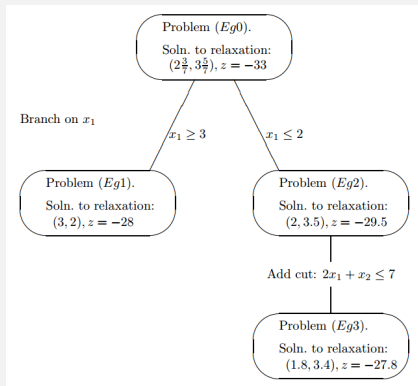
Branch and Cut

Exemplo



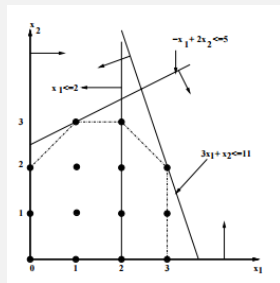
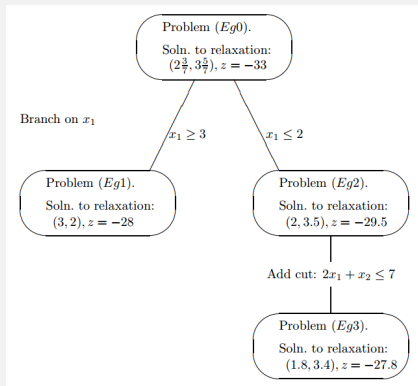
Branch and Cut

Exemplo



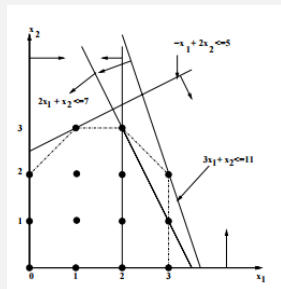
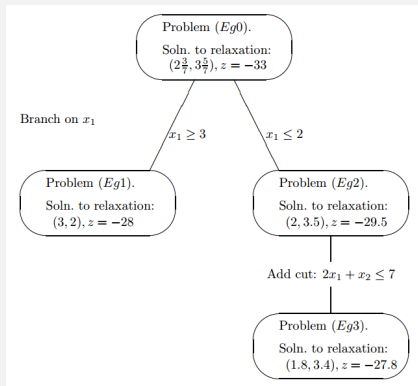
Branch and Cut

Exemplo



Branch and Cut

Exemplo



Branch and Cut

Gurobi

- O Gurobi é um solver comercial que já implementa o algoritmo de Branch and Cut.
- Além disso, o Gurobi também já tem implementado:
 - Pré-processamento.
 - Diversas famílias de desigualdades válidas.
 - Heurísticas.
 - Processamento paralelo.

Agradecimentos

Ao aluno de doutorado André L. Maravilha que construiu o material que serviu de base para esta aula.

- Arenales et al, 2007: M. Arenales; V. Armentano; R. Morabito; H. Yanasse. Pesquisa Operacional para Cursos de Engenharia, Editora Campus / Elsevier, 2007.
- Goldbarg et al, 2005: M. C. Goldbarg; H. P. Luna. Otimização Combinatória e Programação Linear - Modelos e Algoritmos, 2a ed., Editora Campus / Elsevier, 2005.
- Hillier et al, 2013: F. S. Hillier; G. J. Lieberman. Introdução à Pesquisa Operacional, 9a ed., Editora Mc Graw Hill, 2013.