Algoritimos e Estruturas de Dados III

CIC210

Branch and Bound

Ramificar e Limitar

Haroldo Gambini Santos

Universidade Federal de Ouro Preto - UFOP

4 de dezembro de 2009

Branch and Bound - Ramificar e Limitar

Idéia Básica

O algoritmo roda sob a árvore de enumeração das soluções possíveis. No pior caso, todas as soluções serão exploradas. Na prática, frequentemente vários ramos são **podados** com o uso de limites.

Árvore de Enumeração

Branch

Branch consiste em dividir um problema em problemas menores.
Seja um problema P, divide-se em m subproblemas

$$P_1, P_2, \ldots, P_m \text{ tal que}$$

$$P_1 \cup P_2, \ldots, \cup P_m = P$$

Árvore de Enumeração

Branch

Uma maneira de se criar subproblemas é através da fixação de variáveis discretas.

Considere uma variável v, de um problema P, cujos valores possíveis sejam $1, \ldots, V$.

Pode-se dividir P em $P_1, \ldots, P_V$, onde em P_i temos a variável v fixada para o seu i-ésimo valor possível.

Árvore de Enumeração

Branch

A fixação recursiva de diferentes variáveis cria uma árvore, onde temos:

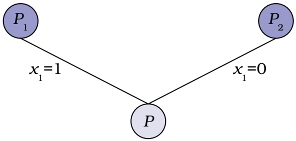
- **nós internos**: esses nós representam todas as soluções que podem ser obtidas respeitando as fixações já feitas;
- **folhas**: representam soluções completas.

Árvore de Enumeração

Ex.: Problema com 3 variáveis binárias: x_1, x_2, x_3.
Problema original P

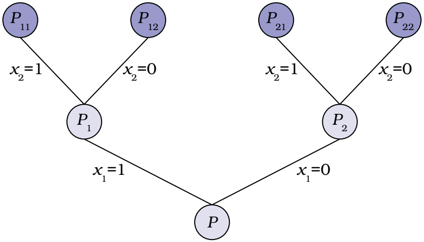
Árvore de Enumeração

Ex.: Problema com 3 variáveis binárias: x_1, x_2, x_3.
Branch na variável x_1, criam-se subproblemas P_1 e P_2



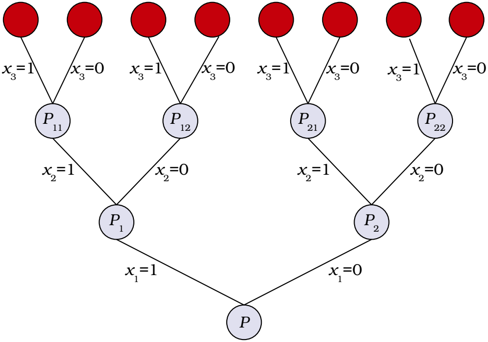
Árvore de Enumeração

Ex.: Problema com 3 variáveis binárias: x_1, x_2, x_3.
Branch na variável x_2, criam-se subproblemas P_{11},P_{12},P_{21} e P_{22}



Árvore de Enumeração

Ex.: Problema com 3 variáveis binárias: x_1, x_2, x_3.
Branch na variável x_3 leva as soluções completas



Árvore de Enumeração

Bound - Limites

- usando somente o *branch* temos um algoritmo exato que em um número finito de passos fornece a solução ótima

Árvore de Enumeração

Bound - Limites

- usando somente o *branch* temos um algoritmo exato que em um número finito de passos fornece a solução ótima
  - mas extremamente ineficiente !

Árvore de Enumeração

Bound - Limites

- usando somente o *branch* temos um algoritmo exato que em um número finito de passos fornece a solução ótima
  - mas extremamente ineficiente !
  - para n variáveis binárias temos 2^n nós a serem explorados.

Árvore de Enumeração

Bound - Limites

- usando somente o *branch* temos um algoritmo exato que em um número finito de passos fornece a solução ótima
  - mas extremamente ineficiente !
  - para n variáveis binárias temos 2^n nós a serem explorados.
- a chave para melhorar a eficiência do B & B é a poda de algumas sub-árvores através do uso de limites.

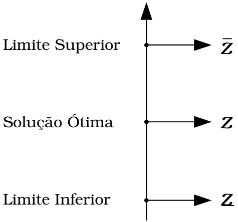
Bound - Limites

Limites

Considere um problema onde queremos achar o lucro máximo da solução ótima:

$$z = \max f(x)$$

Mesmo que z seja difícil de calcular, eventualmente podemos ter limites para z que podem ser calculados com mais facilidade.



Exemplo - Problema da Mochila 0/1

Entrada

n :	items	l_i :	lucro do item i
C :	capacidade da mochila	p_i :	peso do item i

Decisão

$$x_i = \begin{cases} 1 & \text{item } i \text{ incluso na solução} \\ 0 & \text{caso contrário} \end{cases}$$

Objetivo

$$\max \sum_{i=1,\ldots,n} l_i x_i$$

Restrição

$$\sum_{i=1,\ldots,n} p_i x_i \leq C$$

Exemplo - Problema da Mochila 0/1 (PM01)

Heurística

Uma heurística gulosa: tentar colocar os items com grande lucro e pouco peso.
Ordena-se os items por *densidade*:

$$d_i = \frac{l_i}{p_i}$$

Heurística:
$capacidadeRestante = C$
enquanto houver algum $p_i < capacidadeRestante$
    adicione o item i com maior d_i tal que $p_i < capacidadeRestante$

Problema da Mochila 0/1 (PM01)

Ex.: Problema com $n = 4$ e $C = 6$

item	lucro	peso	$\frac{l_i}{p_i}$
1	7	4	1,75
2	4	3	1,33
3	9	5	1,80
4	3	2	1,50

Problema da Mochila 0/1 (PM01)

Ex.: Problema com $n = 4$ e $C = 6$

item	lucro	peso	$\frac{l_i}{p_i}$
1	7	4	1,75
2	4	3	1,33
3	9	5	1,80
4	3	2	1,50

Solução da heurística gulosa

- Solução heurística: itens: $\{3\}$, lucro: 9

Problema da Mochila 0/1 (PM01)

Ex.: Problema com $n = 4$ e $C = 6$

item	lucro	peso	$\frac{l_i}{p_i}$
1	7	4	1,75
2	4	3	1,33
3	9	5	1,80
4	3	2	1,50

Solução da heurística gulosa

- Solução heurística: itens: $\{3\}$, lucro: 9
- A solução *ótima* com certeza é **maior ou igual** a 9, ou seja, temos um limite inferior para o custo da solução ótima

Limite Superior

O Máximo de Lucro

Para o problema da mochila 0/1, como calcular rapidamente um limite superior para o lucro que pode ser obtido ?

O Problema Fracionário da Mochila (PFM)

- trocamos $x_i \in \{0, 1\}$ por $x_i \in [0, 1]$, ou seja, agora podemos colocar "pedaços" de itens;

Limite Superior

O Máximo de Lucro

Para o problema da mochila 0/1, como calcular rapidamente um limite superior para o lucro que pode ser obtido ?

O Problema Fracionário da Mochila (PFM)

- trocamos $x_i \in \{0, 1\}$ por $x_i \in [0, 1]$, ou seja, agora podemos colocar "pedaços" de itens;

Limite Superior

O Máximo de Lucro

Para o problema da mochila 0/1, como calcular rapidamente um limite superior para o lucro que pode ser obtido ?

O Problema Fracionário da Mochila (PFM)

- trocamos $x_i \in \{0, 1\}$ por $x_i \in [0, 1]$, ou seja, agora podemos colocar "pedaços" de itens;
- a solução ótima para o PFM é fácil de calcular: simplesmente pegam-se os itens com maior densidade primeiro; ao se deparar com o primeiro item que não cabe coloca-se a maior fração possível dele;

Limite Superior

O Máximo de Lucro

Para o problema da mochila 0/1, como calcular rapidamente um limite superior para o lucro que pode ser obtido ?

O Problema Fracionário da Mochila (PFM)

- trocamos $x_i \in \{0, 1\}$ por $x_i \in [0, 1]$, ou seja, agora podemos colocar "pedaços" de itens;
- a solução ótima para o PFM é fácil de calcular: simplesmente pegam-se os itens com maior densidade primeiro; ao se deparar com o primeiro item que não cabe coloca-se a maior fração possível dele;
- o PFM é uma relaxação do PM01 visto que tem menos restrições.

Problema da Mochila 0/1 (PM01)

Ex.: Problema com $n = 4$ e $C = 6$

item	lucro	peso	$\frac{l_i}{p_i}$
1	7	4	1,75
2	4	3	1,33
3	9	5	1,80
4	3	2	1,50

Solução Ótima do PFM (Relaxação do PM01)

Problema da Mochila 0/1 (PM01)

Ex.: Problema com $n = 4$ e $C = 6$

item	lucro	peso	$\frac{l_i}{p_i}$
1	7	4	1,75
2	4	3	1,33
3	9	5	1,80
4	3	2	1,50

Solução Ótima do PFM (Relaxação do PM01)

- seleciona item 3

Problema da Mochila 0/1 (PM01)

Ex.: Problema com $n = 4$ e $C = 6$

item	lucro	peso	$\frac{l_i}{p_i}$
1	7	4	1,75
2	4	3	1,33
3	9	5	1,80
4	3	2	1,50

Solução Ótima do PFM (Relaxação do PM01)

- seleciona item 3
- seleciona $\frac{1}{4}$ do item 1

Problema da Mochila 0/1 (PM01)

Ex.: Problema com $n = 4$ e $C = 6$

item	lucro	peso	$\frac{l_i}{p_i}$
1	7	4	1,75
2	4	3	1,33
3	9	5	1,80
4	3	2	1,50

Solução Ótima do PFM (Relaxação do PM01)

- seleciona item 3
- seleciona $\frac{1}{4}$ do item 1
- solução com lucro 10,75

Problema da Mochila 0/1 (PM01)

Ex.: Problema com $n = 4$ e $C = 6$

item	lucro	peso	$\frac{l_i}{p_i}$
1	7	4	1,75
2	4	3	1,33
3	9	5	1,80
4	3	2	1,50

Limites Encontrados

Limite Superior (Relaxação PFM) lucro 10,75

$\downarrow$

? ótimo ? ?

$\uparrow$

Limite Inferior (Heurística Gulosa) lucro 9

Problema da Mochila 0/1 (PM01)

Ex.: Problema com $n = 4$ e $C = 6$

item	lucro	peso	$\frac{l_i}{p_i}$
1	7	4	1,75
2	4	3	1,33
3	9	5	1,80
4	3	2	1,50

Limites Encontrados

Limite Superior (Relaxação PFM) lucro 10,75

$\downarrow$

Solução Ótima (itens 1 e 4) lucro 10

$\uparrow$

Limite Inferior (Heurística Gulosa) lucro 9

Limites

Soluções Parciais

Tanto a heurística quanto a relaxação PFM podem ser executadas em nós internos da árvore, considerando algumas fixações de variáveis. Atualiza-se a capacidade restante e items disponíveis).

Esse procedimento oferece um limite Limite Inferior e Superior para esses nós.

Limites

Soluções Parciais

Tanto a heurística quanto a relaxação PFM podem ser executadas em nós internos da árvore, considerando algumas fixações de variáveis. Atualiza-se a capacidade restante e items disponíveis).

Esse procedimento oferece um limite Limite Inferior e Superior para esses nós.

Limites

Solução Incumbente

É a melhor solução encontrada até o momento durante a busca.

Essa solução pode aparecer durante a execução de uma heurística ou durante o percurso na árvore (ao se chegar em nós folha).

No caso de Maximização, temos um Limite Inferior.

Limites

Solução Incumbente

É a melhor solução encontrada até o momento durante a busca.

Essa solução pode aparecer durante a execução de uma heurística ou durante o percurso na árvore (ao se chegar em nós folha).

No caso de Maximização, temos um Limite Inferior.

Limite Superior

Relaxação

A solução de um problema relaxado oferece uma solução cujo lucro é melhor ou igual ao da solução ótima do problema original.

Em Maximização, temos um Limite Superior.

Poda

Razões para Podar um Nó

Limite a relaxação (Limite Superior) indica que não há possibilidade de se melhorar a solução incumbente;

Infactibilidade as fixações já feitas induzem a alguma infactibilidade (estouro da capacidade da mochila, por ex.).

Em ambos os casos o nó e todos os seus filhos são podados.

Branch and Bound - Exemplo: Problema da Mochila

Incumbente: 11

LS: 12,5

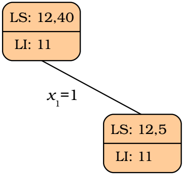
LI: 11

C=7

i	l_i	p_i	l_i/p_i	x_i
1	7	4	1,75	?
2	4	3	1,33	?

i	l_i	p_i	l_i/p_i	x_i
3	9	5	1,80	?
4	2	2	1,00	?

Branch and Bound - Exemplo: Problema da Mochila



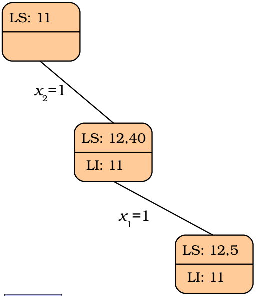
Incumbente: 11

C=7

i	l_i	p_i	l_i/p_i	x_i
1	7	4	1,75	?
2	4	3	1,33	?

i	l_i	p_i	l_i/p_i	x_i
3	9	5	1,80	?
4	2	2	1,00	?

Branch and Bound - Exemplo: Problema da Mochila



C=7

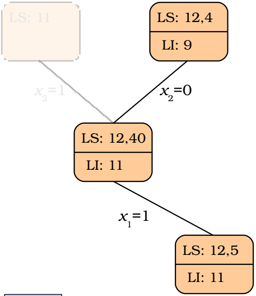
i	l_i	p_i	l_i/p_i	x_i
1	7	4	1,75	?
2	4	3	1,33	?

i	l_i	p_i	l_i/p_i	x_i
3	9	5	1,80	?
4	2	2	1,00	?

Incumbente: 11

Podado: continuar nesse nó não oferecerá nenhuma solução com custo melhor do que 11.

Branch and Bound - Exemplo: Problema da Mochila



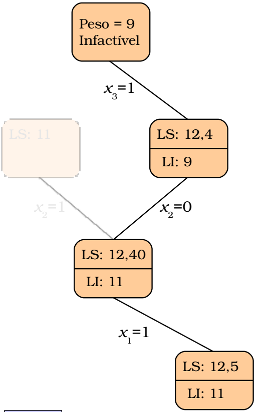
Incumbente: 11

C=7

i	l_i	p_i	l_i/p_i	x_i
1	7	4	1,75	?
2	4	3	1,33	?

i	l_i	p_i	l_i/p_i	x_i
3	9	5	1,80	?
4	2	2	1,00	?

Branch and Bound - Exemplo: Problema da Mochila



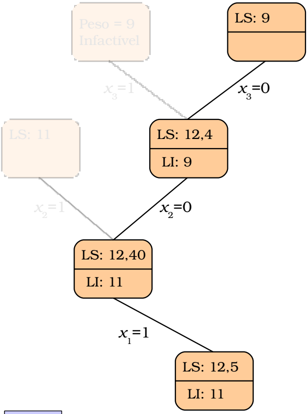
$C=7$

i	l_i	p_i	l_i/p_i	x_i
1	7	4	1,75	?
2	4	3	1,33	?

i	l_i	p_i	l_i/p_i	x_i
3	9	5	1,80	?
4	2	2	1,00	?

Incumbente: 11
Podado:
Infactibilidade.

Branch and Bound - Exemplo: Problema da Mochila



$C=7$

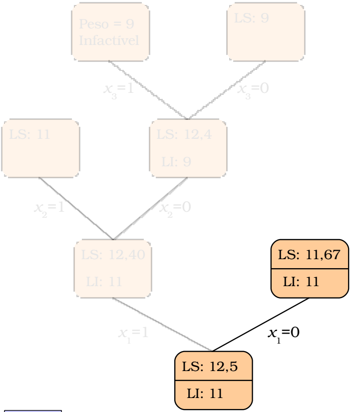
i	l_i	p_i	l_i/p_i	x_i
1	7	4	1,75	?
2	4	3	1,33	?

i	l_i	p_i	l_i/p_i	x_i
3	9	5	1,80	?
4	2	2	1,00	?

Incumbente: 11

Podado por Limite.

Branch and Bound - Exemplo: Problema da Mochila



$C=7$

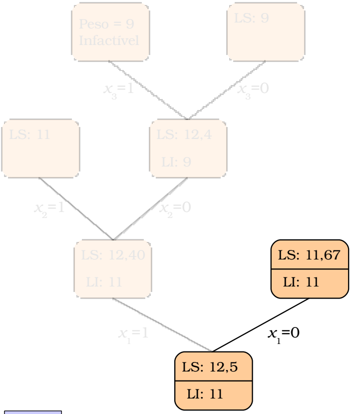
i	l_i	p_i	l_i/p_i	x_i
1	7	4	1,75	?
2	4	3	1,33	?

i	l_i	p_i	l_i/p_i	x_i
3	9	5	1,80	?
4	2	2	1,00	?

Incumbente: 11

Podado por Limite: como os lucros são inteiros, o melhor lucro possível considerando a relaxação 11,67 é 11.

Branch and Bound - Exemplo: Problema da Mochila



Incumbente: 11

Solução Provadamente Ótima

C=7

i	l_i	p_i	l_i/p_i	x_i
1	7	4	1,75	?
2	4	3	1,33	?

i	l_i	p_i	l_i/p_i	x_i
3	9	5	1,80	?
4	2	2	1,00	?