Métodos de Apoio à Decisão

Otimização inteira: método da pesquisa em árvore

João Pedro Pedroso

2023/2024

Knapsack problem

- Um ladrão está a fazer um assalto e tem à escolha uma série de objetos
  - valor
  - peso
- Para cada um deles, pode-o roubar ou não
  - i.e., não pode *dividir* objetos
- Leva-os numa mochila, que tem capacidade limitada
  - não pode levar mais de um determinado peso
- O que deve escolher para maximizar o valor?

Problemas da mochila: formulação

- Variáveis:
  - $x_i = 1$ se escolher o objeto i, $\qquad i = 1, \ldots, n$
  - $x_i = 0$ caso contrário
- Restrição: peso total inferior ao limite b:

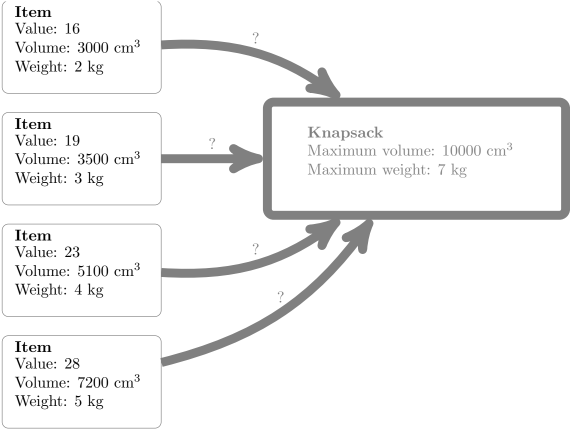
$$\sum_{i=1}^{n} a_i x_i \leq b$$

- Objetivo: maximizar valor

$$\sum_{i=1}^{n} c_i x_i$$

Examplo: Mochila com restrições múltiplas

Multi-Constrained Knapsack Problem



- volume máximo: 10000 cm^3
- peso máximo: 7 kg
- quatro objetos:
  - peso 2, 3, 4, 5
  - volume 3000, 3500, 5100, 7200
  - valor 16, 19, 23, 28
- como escolher, para levar o máximo valor?

$$
\begin{array}{lrrrrcr}
\text{maximize} & 16x_1 + & 19x_2 + & 23x_3 + & 28x_4 & & \\
\text{subject to:} & 2x_1 + & 3x_2 + & 4x_3 + & 5x_4 & \leq & 7 \\
 & 30x_1 + & 35x_2 + & 51x_3 + & 72x_4 & \leq & 100 \\
 & x_1, & x_2, & x_3, & x_4 & \in & \{0,1\}
\end{array}
$$

Como encontrar a solução?

- programação linear não pode ser usada diretamente
  - variáveis não são contínuas
- método da pesquisa em árvore (branch-and-bound)

Branch-and-bound

- Usar optimização linear para resolver problema relaxado
  - relaxação linear:
    - "ignorar" restrições de integralidade
    - problema de otimização linear
- Se a solução da relaxação linear do problema original for inteira $\rightarrow$ optima
- Caso contrário:
  - dividir sistematicamente problema em dois subproblemas
  - excluir a solução anterior de ambos

Branch-and-bound

Conceito principal: *dividir e conquistar*

- dividir o problema inicial em subproblemas cada vez menores: branching
  - partitionar o conjunto de soluções admissíveis em subconjuntos cada vez menores
- conquistar: *descartar*
  - *bounding:* verificar quão boa pode ser a melhor solução num subconjunto
  - descartar o subconjunto se esse limite (*bound*) não puder conter a solução ótima do problema original

Branch-and-bound: passos principais:

- branching
- bounding
- fathoming

Exemplo

- Modelo anterior
- Para cada variável, substituir $x_i \in \{0, 1\}$ por $0 \leq x_i \leq 1$
- Subproblema 1 (relaxação linear do problema original)

$$
\begin{array}{lrrrrll}
\text{maximize} & 16x_1 + & 19x_2 + & 23x_3 + & 28x_4 & & \\
\text{subject to:} & 2x_1 + & 3x_2 + & 4x_3 + & 5x_4 & \leq & 7 \\
 & 30x_1 + & 35x_2 + & 51x_3 + & 72x_4 & \leq & 100 \\
 & x_1, & x_2, & x_3, & x_4 & \geq 0, & \leq 1
\end{array}
$$

- Solução:

```
x [*] :=
1   1
2   1
3   0.5
4   0
;
```

$\rightarrow$ não admissível

- Valor do objetivo: $46.5 \rightarrow$ majorante do ótimo

SP1 **UB=**$46\frac{1}{2}$

$x = (1, 1, \frac{1}{2}, 0)$

Branching

- particionar conjunto de soluções admissíveis em dois subconjuntos
- $x_3 = 0.5 \Rightarrow$
  - subproblema 2: $x_3 \geq 1$ ($\rightarrow$ `x3 = 1`)
  - subproblema 3: $x_3 \leq 0$ ($\rightarrow$ `x3 = 0`)

- subproblema 2: $x_3 \geq 1$ ($\rightarrow$ `x3 = 1`)

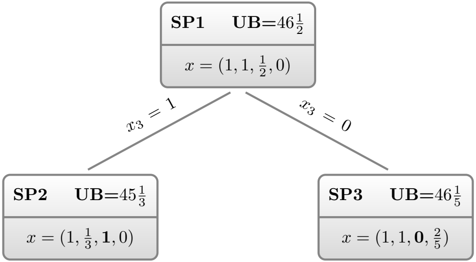
$$\begin{array}{lrrrrll}
\text{maximize} & 16x_1 + & 19x_2 + & 23x_3 + & 28x_4 & & \\
\text{subject to:} & 2x_1 + & 3x_2 + & 4x_3 + & 5x_4 & \leq & 7 \\
 & 30x_1 + & 35x_2 + & 51x_3 + & 72x_4 & \leq & 100 \\
 & & & x_3 & & = & 1 \\
 & x_1, & x_2, & & x_4 & \geq 0, & \leq 1
\end{array}$$

- subproblema 3: $x_3 \leq 0$ ($\rightarrow$ `x3 = 0`)

$$\begin{array}{lrrrrll}
\text{maximize} & 16x_1 + & 19x_2 + & 23x_3 + & 28x_4 & & \\
\text{subject to:} & 2x_1 + & 3x_2 + & 4x_3 + & 5x_4 & \leq & 7 \\
 & 30x_1 + & 35x_2 + & 51x_3 + & 72x_4 & \leq & 100 \\
 & & & x_3 & & = & 0 \\
 & x_1, & x_2, & & x_4 & \geq 0, & \leq 1
\end{array}$$

Bounding

- *Bounding:*
  - para cada subproblema, obter o *melhor valor possível* para uma solução admissível
  - maximização: majorante do ótimo
  - minimização: minorante do ótimo
- Forma "standard": resolver relaxação do problema
  - descartar as restrições que o tornam difícil
  - otimização inteira $\rightarrow$ restrições de integralidade
  - $\rightarrow$ resolver a relaxação linear do subproblema
- No exemplo: todos os coeficientes são inteiros
  - podemos arredondar para baixo valor do objetivo


SP1 UB=$46\frac{1}{2}$
$x = (1, 1, \frac{1}{2}, 0)$
$x_3 = 1$
$x_3 = 0$
SP2 UB=$45\frac{1}{3}$
$x = (1, \frac{1}{3}, \mathbf{1}, 0)$
SP3 UB=$46\frac{1}{5}$
$x = (1, 1, \mathbf{0}, \frac{2}{5})$

Escolha do subproblema para dividir

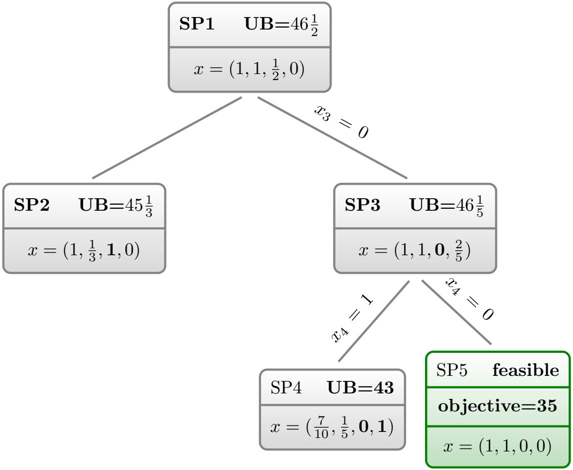
- Pesquisa não informada:
  - *breadth-first search* → fila: FIFO
  - *depth-first search* → fila: LIFO
    - Forma mais económica em memória:
    - último subproblema criado
- Forma mais habitual em otimização inteira:
  - nó da árvore com melhor *bound*
  - em maximização → majorante mais alto
  - *best-first search* → fila: lista *ordenada* de nós
- No exemplo: *SP3*
  - neste caso, só tem uma variável fracionária
  - caso haja várias, temos de escolher uma (branching rule)
    - heurística: variável *mais fracionária*
    - valor mais próximo de *.5

- subproblema 4: $x_4 \geq 1$ $\quad (\rightarrow \texttt{x4 = 1})$

$$
\begin{array}{lrrrrcr}
\text{maximize} & 16x_1 + & 19x_2 + & 23x_3 + & 28x_4 & & \\
\text{subject to:} & 2x_1 + & 3x_2 + & 4x_3 + & 5x_4 & \leq & 7 \\
 & 30x_1 + & 35x_2 + & 51x_3 + & 72x_4 & \leq & 100 \\
 & & & x_3 & & = & 0 \\
 & & & & x_4 & = & 1 \\
 & x_1, & x_2, & & & \geq 0, & \leq 1
\end{array}
$$

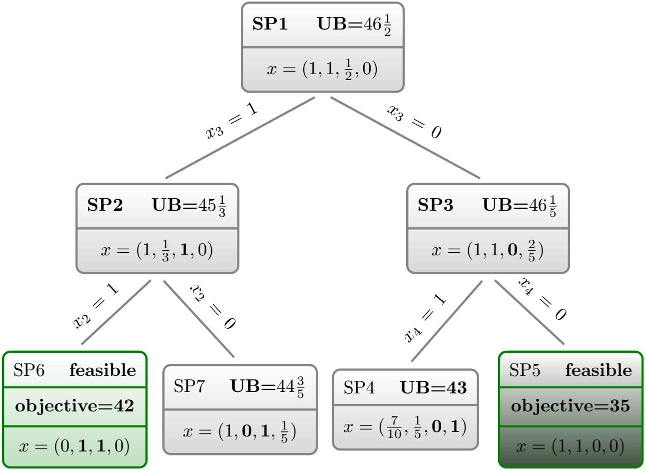
- subproblema 5: $x_4 \leq 0$ $\quad (\rightarrow \texttt{x4 = 0})$

$$
\begin{array}{lrrrrcr}
\text{maximize} & 16x_1 + & 19x_2 + & 23x_3 + & 28x_4 & & \\
\text{subject to:} & 2x_1 + & 3x_2 + & 4x_3 + & 5x_4 & \leq & 7 \\
 & 30x_1 + & 35x_2 + & 51x_3 + & 72x_4 & \leq & 100 \\
 & & & x_3 & & = & 0 \\
 & & & & x_4 & = & 0 \\
 & x_1, & x_2, & & & \geq 0, & \leq 1
\end{array}
$$

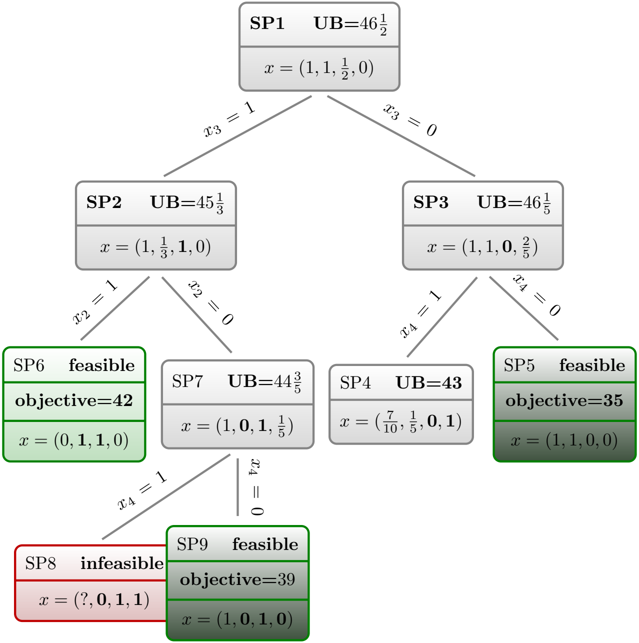

SP1 UB=$46\frac{1}{2}$
$x = (1, 1, \frac{1}{2}, 0)$
$x_3 = 0$
SP2 UB=$45\frac{1}{3}$
$x = (1, \frac{1}{3}, \mathbf{1}, 0)$
SP3 UB=$46\frac{1}{5}$
$x = (1, 1, \mathbf{0}, \frac{2}{5})$
$x_4 = 1$
$x_4 = 0$
SP4 UB=43
$x = (\frac{7}{10}, \frac{1}{5}, \mathbf{0}, \mathbf{1})$
SP5 feasible
objective=35
$x = (1, 1, 0, 0)$

Conquista

- A solução do SP5 já tem todas as variáveis inteiras:
  - SP5 não precisa de ser dividido
  - solução candidata
    - valor do objetivo: minorante para o ótimo do problema original
    - no decorrer do método: guardamos solução com o melhor dos minorantes encontrados
- Próximo SP a considerar (best-first): SP2

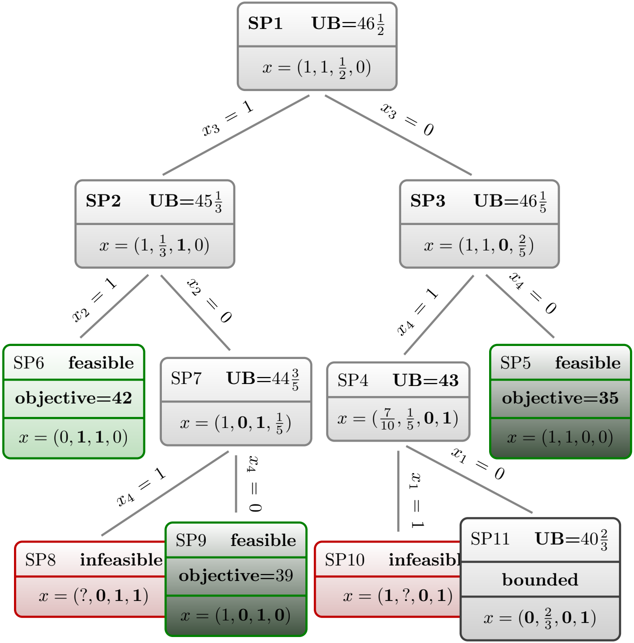

SP1 UB=$46\frac{1}{2}$
$x = (1, 1, \frac{1}{2}, 0)$
$x_3 = 1$
$x_3 = 0$
SP2 UB=$45\frac{1}{3}$
$x = (1, \frac{1}{3}, \mathbf{1}, 0)$
SP3 UB=$46\frac{1}{5}$
$x = (1, 1, \mathbf{0}, \frac{2}{5})$
$x_2 = 1$
$x_2 = 0$
$x_4 = 1$
$x_4 = 0$
SP6 feasible
objective=42
$x = (0, \mathbf{1}, \mathbf{1}, 0)$
SP7 UB=$44\frac{3}{5}$
$x = (1, \mathbf{0}, \mathbf{1}, \frac{1}{5})$
SP4 UB=43
$x = (\frac{7}{10}, \frac{1}{5}, \mathbf{0}, \mathbf{1})$
SP5 feasible
objective=35
$x = (1, 1, 0, 0)$

- SP6 já tem todas as variáveis inteiras
- Objetivo melhor que da solução candidata anterior
- $\rightarrow$ candidata passa a ser a solução do SP6
- Próximo SP a considerar (best-first): SP7


SP1 UB=$46\frac{1}{2}$
$x = (1, 1, \frac{1}{2}, 0)$
$x_3 = 1$
$x_3 = 0$
SP2 UB=$45\frac{1}{3}$
$x = (1, \frac{1}{3}, \mathbf{1}, 0)$
SP3 UB=$46\frac{1}{5}$
$x = (1, 1, \mathbf{0}, \frac{2}{5})$
$x_2 = 1$
$x_2 = 0$
$x_4 = 1$
$x_4 = 0$
SP6 feasible
objective=42
$x = (0, \mathbf{1}, \mathbf{1}, 0)$
SP7 UB=$44\frac{3}{5}$
$x = (1, \mathbf{0}, \mathbf{1}, \frac{1}{5})$
SP4 UB=43
$x = (\frac{7}{10}, \frac{1}{5}, \mathbf{0}, \mathbf{1})$
SP5 feasible
objective=35
$x = (1, 1, 0, 0)$
$x_4 = 1$
$x_4 = 0$
SP8 infeasible
$x = (?, \mathbf{0}, \mathbf{1}, \mathbf{1})$
SP9 feasible
objective=39
$x = (1, \mathbf{0}, \mathbf{1}, \mathbf{0})$

Conquista

- SP8 não tem soluções admissíveis
  - não precisa de ser dividido
- A solução do SP9 já tem todas as variáveis inteiras
  - não precisa de ser dividido
  - inferior à actual solução cadidata $\rightarrow$ descartar
- Próximo SP a considerar (best-first): SP4


SP1 UB=$46\frac{1}{2}$
$x = (1, 1, \frac{1}{2}, 0)$
$x_3 = 1$
$x_3 = 0$
SP2 UB=$45\frac{1}{3}$
$x = (1, \frac{1}{3}, \mathbf{1}, 0)$
SP3 UB=$46\frac{1}{5}$
$x = (1, 1, \mathbf{0}, \frac{2}{5})$
$x_2 = 1$
$x_2 = 0$
$x_4 = 1$
$x_4 = 0$
SP6 feasible
objective=42
$x = (0, \mathbf{1}, \mathbf{1}, 0)$
SP7 UB=$44\frac{3}{5}$
$x = (1, \mathbf{0}, \mathbf{1}, \frac{1}{5})$
SP4 UB=43
$x = (\frac{7}{10}, \frac{1}{5}, \mathbf{0}, \mathbf{1})$
SP5 feasible
objective=35
$x = (1, 1, 0, 0)$
$x_4 = 1$
$x_4 = 0$
$x_1 = 1$
$x_1 = 0$
SP8 infeasible
$x = (?, \mathbf{0}, \mathbf{1}, \mathbf{1})$
SP9 feasible
objective=39
$x = (1, \mathbf{0}, \mathbf{1}, \mathbf{0})$
SP10 infeasibl
$x = (\mathbf{1}, ?, \mathbf{0}, \mathbf{1})$
SP11 UB=$40\frac{2}{3}$
bounded
$x = (\mathbf{0}, \frac{2}{3}, \mathbf{0}, \mathbf{1})$

- todos os subproblemas foram *conquistados*
- solução candidata atual $\rightarrow$ ótima

Resumo: passos a cada iteração

- Inicialização:
  - melhor majorante $\rightarrow -\infty$
  - resolver relaxação linear $\rightarrow$ SP1
  - fila de nós em aberto: SP1
- 1 Branching (divisão)
  - Escolher um SP entre os que estão em aberto
    - *Depth-first search* $\rightarrow$ o último que foi criado
    - *Best-first search* $\rightarrow$ o que tiver melhor valor da relaxação linear
  - Escolher uma variável fracionária, criar dois SPs
- 2 Bounding
  - para cada novo SP $\rightarrow$ resolver relaxação linear
- 3 Fathoming
  - SP pode ser descartado em 3 situações
    - conhecemos solução admissível melhor que o majorante do SP
    - a relaxação linear do SP não tem soluções admissíveis
    - a solução do SP já tem todas as variáveis inteiras
      $\rightarrow$ solução candidata se melhorar o minorante atual
      $\rightarrow$ atualizar minorante
- Teste de otimalidade: parar quando não há SP em aberto

Exemplo (livro *"W. Wayne"*)

Because of excessive pollution on the Momiss River, the state of Momiss is going to build pollution control stations. Three sites (1, 2, and 3) are under consideration. Momiss is interested in controlling the pollution levels of two pollutants (1 and 2). The state legislature requires that at least 80,000 tons of pollutant 1 and at least 50,000 tons of pollutant 2 be removed from the river. The relevant data for this problem are shown below. Formulate an integer optimization problem to minimize the cost of meeting the state legislature's goals.

Site	Cost of building station ($)	Cost of treating 1 ton water ($)	Amount removed (ton) per ton of water	
			Pollutant 1	Pollutant 2
1	100000	20	0.40	0.30
2	60000	30	0.25	0.20
3	40000	40	0.20	0.25

Formulação

$$
\begin{array}{ll}
\text{minimizar} & 20x_1 + 30x_2 + 40x_3 + 100000y_1 + 60000y_2 + 40000y_3 \\
\text{sujeito a:} & 0.40x_1 + 0.25x_2 + 0.20x_3 \geq 80000 \\
 & 0.30x_1 + 0.20x_2 + 0.23x_3 \geq 50000 \\
 & x_1 \leq M_1 y_1 \\
 & x_2 \leq M_2 y_2 \\
 & x_3 \leq M_3 y_3 \\
 & x_1, x_2, x_3 \geq 0 \\
 & y_1, y_2, y_3 \in \{0, 1\}
\end{array}
$$

Formulação

$$
\begin{array}{ll}
\text{minimizar} & 20x_1 + 30x_2 + 40x_3 + 100000y_1 + 60000y_2 + 40000y_3 \\
\text{sujeito a:} & 0.40x_1 + 0.25x_2 + 0.20x_3 \geq 80000 \\
& 0.30x_1 + 0.20x_2 + 0.23x_3 \geq 50000 \\
& x_1 \leq M_1 y_1 \\
& x_2 \leq M_2 y_2 \\
& x_3 \leq M_3 y_3 \\
& x_1, x_2, x_3 \geq 0 \\
& y_1, y_2, y_3 \in \{0, 1\}
\end{array}
$$

Que valores escolher para M_i?

Big M

No problema de localização da aula passada:

$$\sum_{i \in I} x_{ij} \leq M_j y_j \quad \forall j \in J$$

- Ideia: modelar *"se não ativarmos instalação, não podemos transportar nada de lá"*
- Parametro M representa um valor suficientemente grande
  - restrição deve ser limitante se $y_j = 0$
  - não deve ser limitante no caso contrário
- No entanto:
  - na prática, valores de M muito elevados perturbam o modelo

Modeling tip

Big M deve ser calculado como o menor valor possível

- Sempre que possível, é melhor não usar *Big M*
- Se for necessário o seu uso, escolher menor valor possível
  - mas de forma a que formulação se mantenha correta
  - i.e., a restrição não seja limitante quando $y_j = 1$
- Usar números grandes, como $M = 9999999$, é impensável, exceto para instâncias muito pequenas

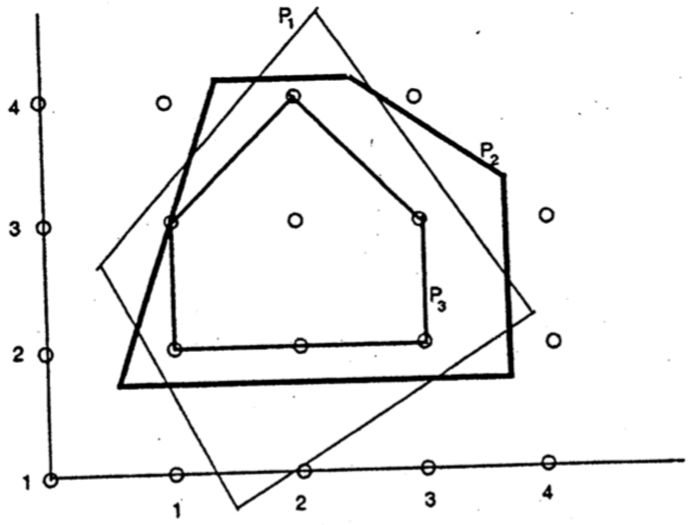
Exercício

- Ajustar o modelo anterior para usar o *menor valor possível* para M
- Usando *branch-and-bound*, resolva o problema de remoção de poluentes com
  - $M = 9999999$
  - o menor valor válido para M_i, $i = 1, 2, 3$

Observações

- Para o problema de localização não capacitado:
  - formulação justa (*tight*): definir $M =$ montante total encomendado
- No entanto, é possível melhorar a formulação:
  - adicionando $x_{ij} \leq d_i y_j$
- que formulação devemos usar?
  - a resposta depende do caso particular
  - em geral formulações mais fortes são recomendadas
    - forte/fraca $\rightarrow$ definida em termos da *relaxação linear*

Formulações diferentes



Próximas aulas:

- *branch and bound* para problemas específicos
- aplicações de otimização inteira
- otimização em grafos