分支限界
及其应用

组合优化问题

- 组合优化问题的相关概念

  目标函数（极大化或极小化）

  约束条件（解满足的条件）

  可行解: 搜索空间满足约束条件的解

  最优解: 使得目标函数达到极大 (或极小)的可行解

- 背包问题

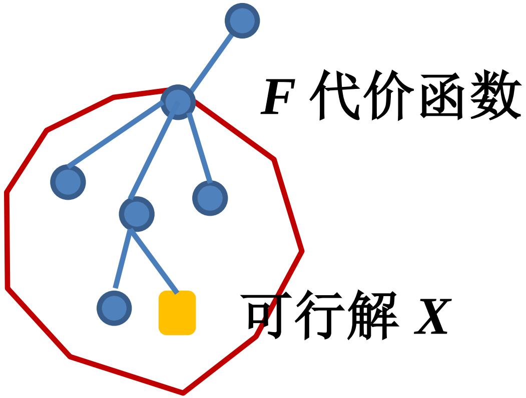
$$\max \quad x_1 + 3x_2 + 5x_3 + 9x_4$$

$$2x_1 + 3x_2 + 4x_3 + 7x_4 \le 10$$

$$x_i \in \mathrm{N}, \quad i = 1,2,3,4$$

代价函数

- 计算位置：搜索树的结点
- 值：极大化问题是以该点为根的子树所有可行解的值的上界（极小化问题为下界）
- 性质：对极大化问题父结点代价不小于子结点的代价 (极小化问题相反)

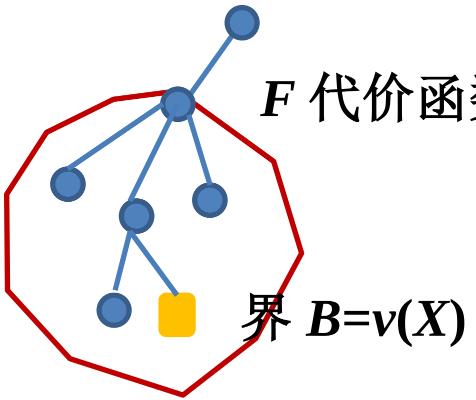


$F \geq v(X)$

极大化

界

- 含义：当前得到可行解的目标函数的最大值(极小化问题相反）
- 初值：极大化问题初值为 0 (极小化问题为最大值)
- 更新：得到更好的可行解时



分支限界

停止分支回溯父结点的依据：

1. 不满足约束条件

2. 对于极大化问题，代价函数值小于当前界（对于极小化问题是大于界）

界的更新

对极大化问题，如果一个新的可行解的优化函数值大于 (极小化问题为小于) 当前的界，则把界更新为该可行解的值

实例

背包问题：

4 种物品，重量 w_i 与价值 v_i 分别为

$v_1=1, v_2=3, v_3=5, v_4=9$

$w_1=2, w_2=3, w_3=4, w_4=7$

背包重量限制为10

$$\max x_1 + 3x_2 + 5x_3 + 9x_4$$

$$2x_1 + 3x_2 + 4x_3 + 7x_4 \le 10$$

$$x_i \in \mathrm{N}, i = 1,2,3,4$$

代价函数的设定

- 对结点$< x_1, x_2, \ldots, x_k >$，估计以该结点为根的子树中可行解的上界.
- 按 v_i/w_i 从大到小排序，$i = 1,2, \ldots, n$
- 代价函数 = 已装入价值+ Δ

  Δ：还可继续装入最大价值的上界

  Δ = 背包剩余重量 $\times\, v_{k+1}/w_{k+1}$ (可装)

  $\Delta = 0$ (不可装)

实例：背包问题

$$\max x_1 + 3x_2 + 5x_3 + 9x_4$$
$$2x_1 + 3x_2 + 4x_3 + 7x_4 \le 10$$
$$x_i \in \mathrm{N}, i = 1,2,3,4$$

对变元重新排序使得 $\frac{v_i}{w_i} \ge \frac{v_{i+1}}{w_{i+1}}$

排序后

$$\max 9x_1 + 5x_2 + 3x_3 + x_4$$
$$7x_1 + 4x_2 + 3x_3 + 2x_4 \le 10$$
$$x_i \in \mathrm{N}, i = 1,2,3,4$$

代价函数与分支策略

结点<$x_1, x_2, \ldots, x_k$> 的代价函数

$$\sum_{i=1}^{k} v_i x_i + \left(b - \sum_{i=1}^{k} w_i x_i\right) \cdot v_{k+1} / w_{k+1}$$

若对某个 $j > k$ □ $b - \sum_{i=1}^{k} w_i x_i \geq w_j$

$$\sum_{i=1}^{k} v_i x_i$$

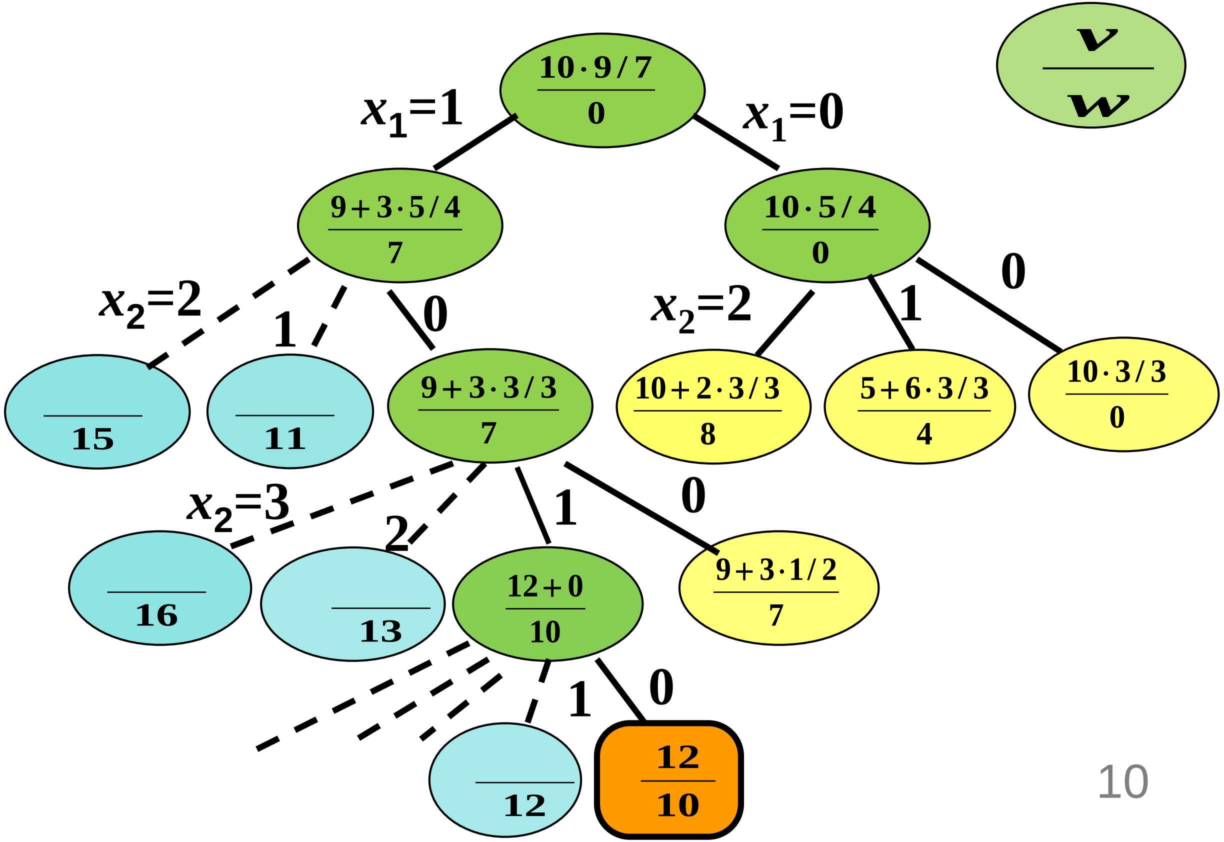
否则

分支策略----深度优先

实例

$$\max 9x_1 + 5x_2 + 3x_3 + x_4$$

$$7x_1 + 4x_2 + 3x_3 + 2x_4 \le 10, \quad x_i \in N, i = 1,2,3,4$$



小结

- 分支限界适用于组合优化问题
- 对结点$<x_1, \ldots, x_k>$定义代价函数

  当前结点为根子树的可行解的上界或下界

  极大化问题与极小化问题的区别
- 定义界的初值

  得到新的更好的可行解就更新界