

# 《计算机算法设计与分析》

## 第五章 动态规划

马丙鹏

2017年10月24日



# 第五章 动态规划

- 5.1 一般方法
- 5.2 多段图问题
- 5.3 每对结点之间的最短路径
- 5.4 最优二分检索树
- 5.5 0/1背包问题
- 5.6 可靠性设计
- 5.7 货郎担问题
- 5.8 流水线调度问题



# 5.5 0/1背包问题

## ● 1. 问题描述

– **KNAP(1, j, X)**

➤ **目标函数:**  $\sum_{1 \leq i \leq j} p_i x_i$

➤ **约束条件:**  $\sum_{1 \leq i \leq j} w_i x_i \leq X$

$$x_i = 0 \text{ 或 } 1, p_i > 0, w_i > 0, 1 \leq i \leq j$$

– **0/1背包问题: KNAP(1, n, M)**

– **最优性原理**对于0/1背包问题成立

– **求解策略:** 向前递推、向后递推



## 5.5 0/1背包问题

### ● 1. 问题描述

#### – 向后递推关系式

➤ 记 $f_j(X)$ 是KNAP(1, j, X)的最优解, 则 $f_n(M)$ 有

$$f_n(M) = \max\{f_{n-1}(M), f_{n-1}(M-w_n)+p_n\}$$

➤ 对于任意的 $f_i(X)$ ,  $i>0$ , 有

$$f_i(X) = \max\{f_{i-1}(X), f_{i-1}(X-w_i)+p_i\}$$



## 5.5 0/1背包问题

### ● 1. 问题描述

– 向后递推关系式

➤ 递推过程:

➤ 初始值

$$f_0 = \begin{cases} 0 & X \geq 0 \\ -\infty & X < 0 \end{cases}$$

➤ 求出所有可能的 $X$ 对应的 $f_i$ 值。

➤  $f_n(M) = \text{KNAP}(1, n, M)$



## 例1 背包问题

$n=3, (w_1, w_2, w_3)=(2, 3, 4), (p_1, p_2, p_3)=(1, 2, 5), M=6$

递推计算过程

$$f_0(X) = \begin{cases} -\infty & X < 0 \\ 0 & X \geq 0 \end{cases}$$

$$f_1(X) = \begin{cases} -\infty & X < 0 \\ \max\{0, -\infty + 1\} = 0 & 0 \leq X < 2 \\ \max\{0, 0 + 1\} = 1 & X \geq 2 \end{cases}$$

第1个物品无法放入  
第1个物品可放入

$$f_2(X) = \begin{cases} -\infty & X < 0 \\ \max\{0, -\infty + 2\} = 0 & 0 \leq X < 2 \\ \max\{1, -\infty + 2\} = 1 & 2 \leq X < 3 \\ \max\{1, 0 + 2\} = 2 & 3 \leq X < 5 \\ \max\{1, 1 + 2\} = 3 & X \geq 5 \end{cases}$$

第1个物品无法放入  
第1个物品可放入,  
第2个物品无法放入  
第1个物品或第2个

第1个物品和第2个  
物品可放入

$$f_3(M) = \max\{3, 1 + 5\} = 6$$

$$f_i(X) = \max\{f_{i-1}(X), f_{i-1}(X - w_i) + p_i\}$$

# 例1 背包问题

$n=3, (w_1, w_2, w_3)=(2, 3, 4), (p_1, p_2, p_3)=(1, 2, 5), M=6$

递推计算过程

解向量的推导(最优的决策序列)

$$f_0(X) = \begin{cases} -\infty & X < 0 \\ 0 & X \geq 0 \end{cases}$$

$$f_1(X) = \begin{cases} -\infty & X < 0 \\ \max\{0, -\infty + 1\} = 0 & 0 \leq X < 2 \\ \max\{0, 0 + 1\} = 1 & X \geq 2 \end{cases}$$

$$f_2(X) = \begin{cases} -\infty & X < 0 \\ \max\{0, -\infty + 2\} = 0 & 0 \leq X < 2 \\ \max\{1, -\infty + 2\} = 1 & 2 \leq X < 3 \\ \max\{1, 0 + 2\} = 2 & 3 \leq X < 5 \\ \max\{1, 1 + 2\} = 3 & X \geq 5 \end{cases}$$

$$f_3(M) = \max\{3, 1 + 5\} = 6$$

$$f_3(M)=6$$

$$x_3=1$$

$$f_3(M)-p_3=1$$

$$f_2(X)=1$$

$$f_1(X)=1$$

$$x_2=0$$

$$x_1=1$$

$$\text{KNAP}(1,3,6)=6$$

$$X=(1,0,1)$$



中国科学院大学

University of Chinese Academy of Sciences 7

$$f_i(X) = \max\{f_{i-1}(X), f_{i-1}(X - w_i) + p_i\}$$

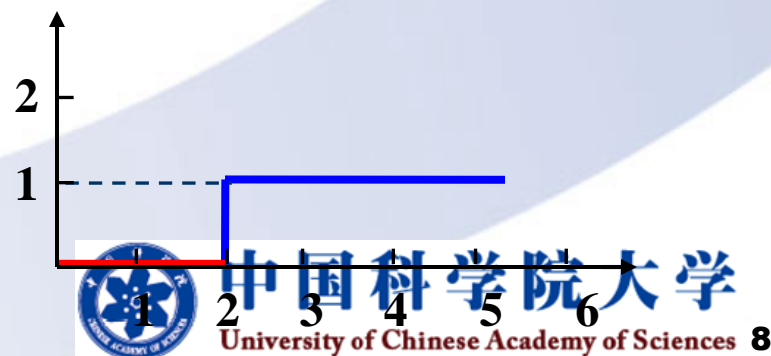
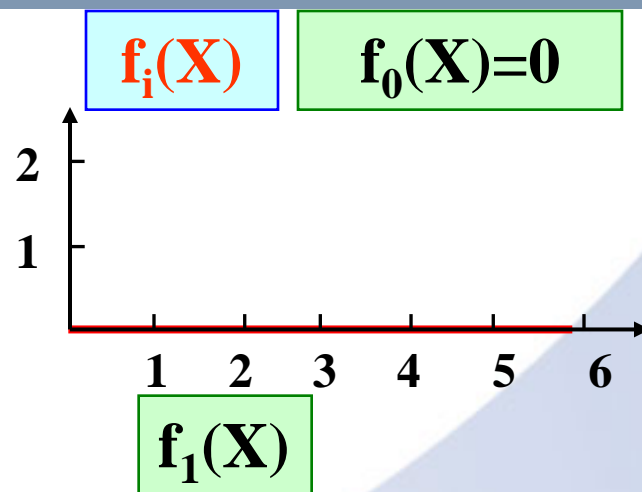
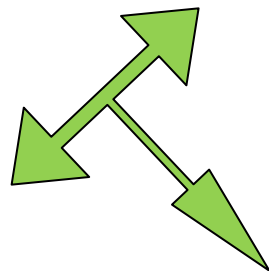
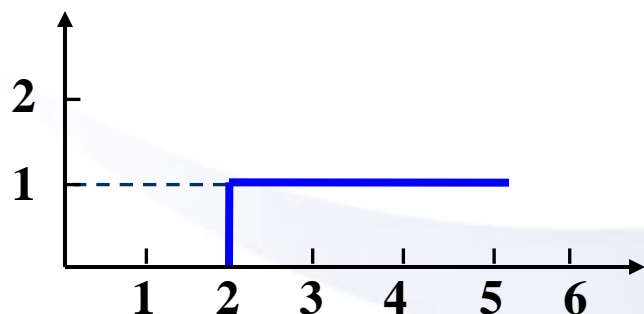
$f_1, f_2, f_3$  计算过程的图解  $(w_1, w_2, w_3)=(2, 3, 4), (p_1, p_2, p_3)=(1, 2, 5), M=6$

- $f_{i-1}(X-w_i)+p_i$  曲线的构造：将  $f_{i-1}(X)$  的曲线在  $X$  轴上右移  $w_i$  个单位，然后上移  $p_i$  个单位而得到；
- $f_i(X)$  曲线的构造：由  $f_{i-1}(X)$  和  $f_{i-1}(X-w_i)+p_i$  的曲线按  $X$  相同时  $f$  取大值的规则归并而成

$$i: f_{i-1}(X-w_i)+p_i$$

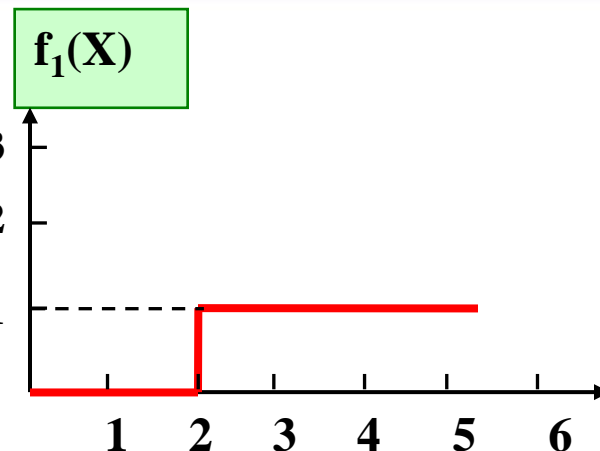
$i=0$ : 函数不存在

$$i=1: f_0(X-w_1)+p_1$$

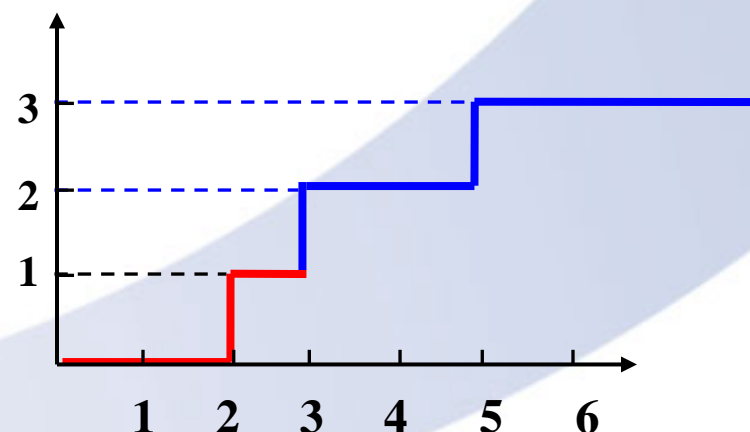




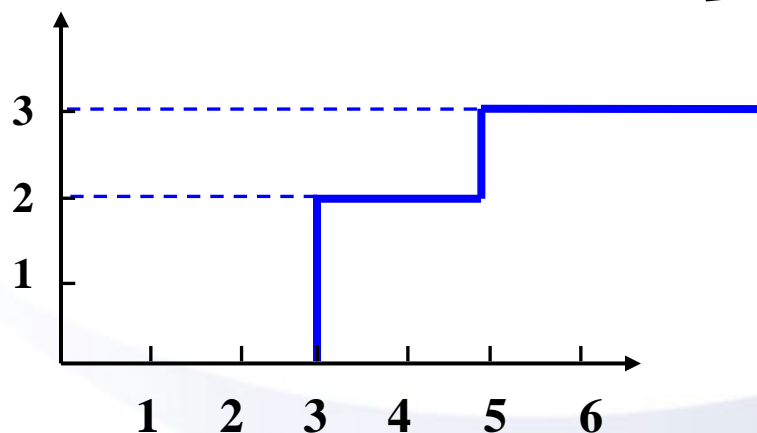
$f_1, f_2, f_3$  计算过程的图解  $(w_1, w_2, w_3)=(2, 3, 4), (p_1, p_2, p_3)=(1, 2, 5), M=6$



$f_2(X)$



$i=2: f_1(X-3)+2$

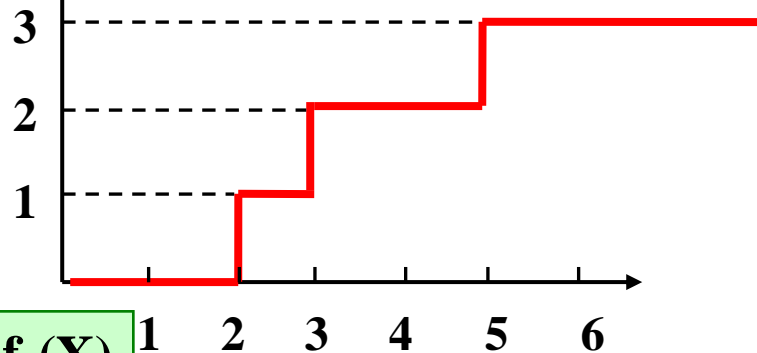


$$f_i(X) = \max\{f_{i-1}(X), f_{i-1}(X - w_i) + p_i\}$$

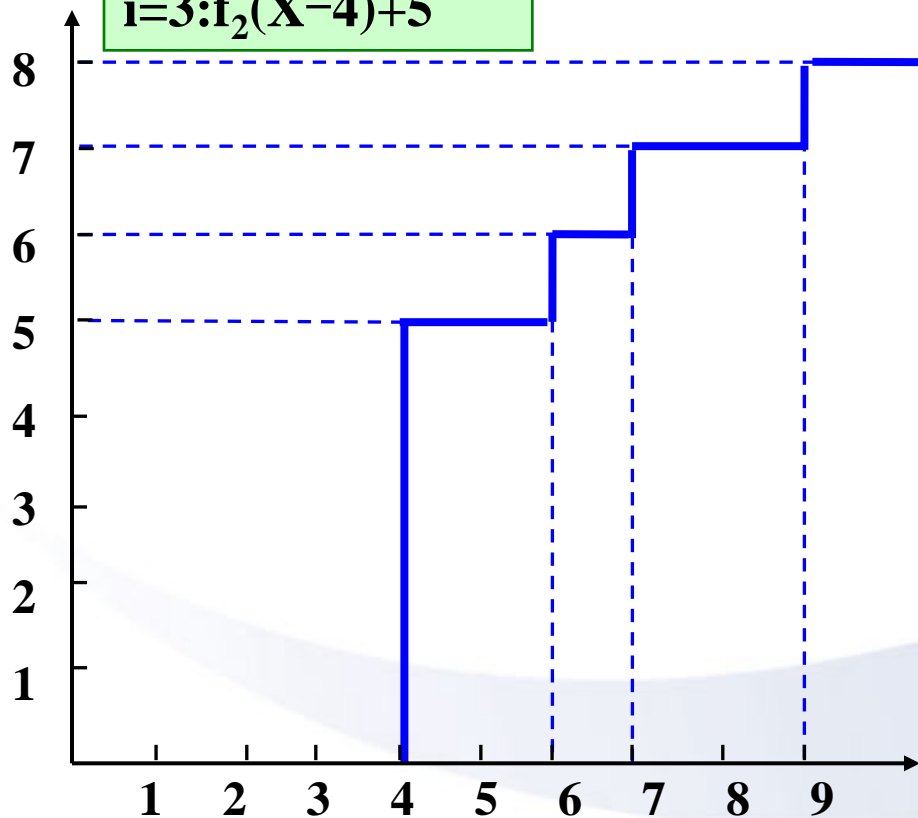
$f_1, f_2, f_3$  计算过程的图解  $(w_1, w_2, w_3)=(2, 3, 4), (p_1, p_2, p_3)=(1, 2, 5), M=6$

$$f_i(X) = \max\{f_{i-1}(X), f_{i-1}(X - w_i) + p_i\}$$

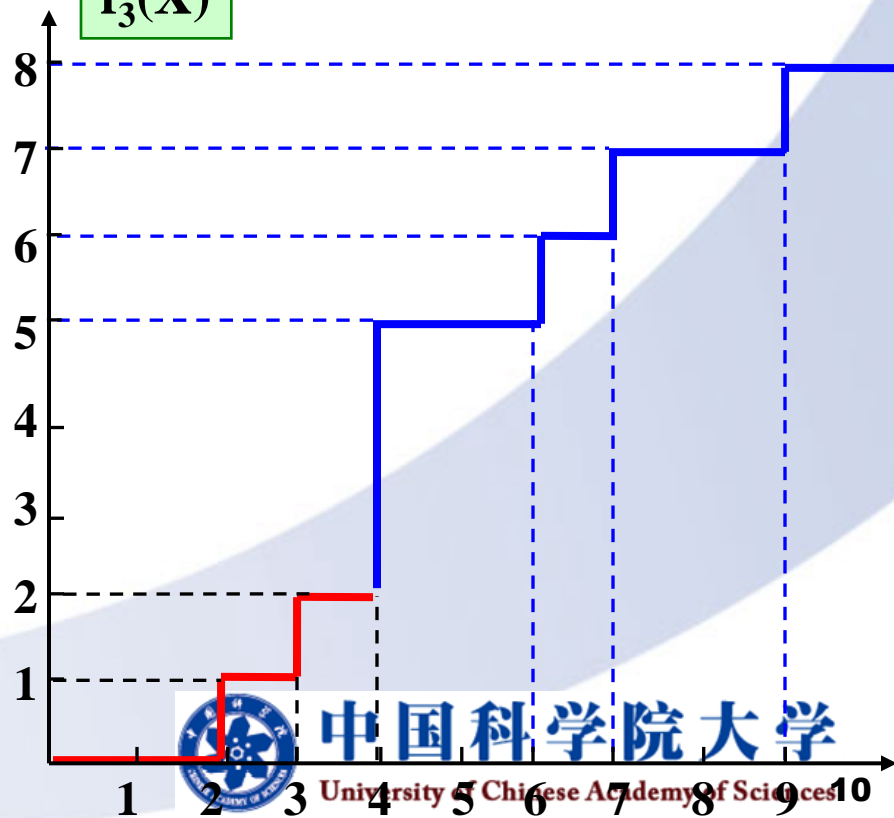
$f_2(X)$



$i=3: f_2(X-4)+5$



$f_3(X)$



中国科学院大学

University of Chinese Academy of Sciences

## 5.5 0/1背包问题

### ● 2. 序偶表示

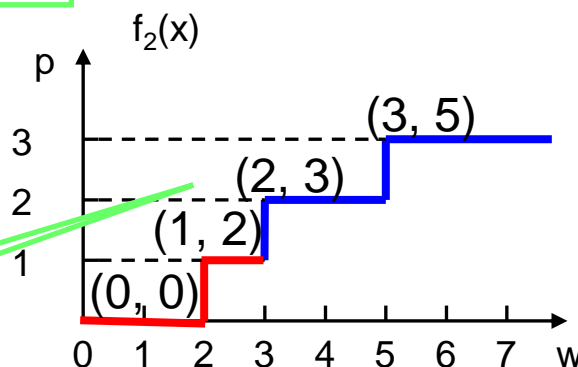
- $f_i$ 是关于 $X$ 的阶跃函数，阶跃点是 $f_i$ 的关键点。每个阶跃点用其对应坐标表示——称为一个序偶， $f_i$ 阶跃点的集合称为 $f_i$ 的序偶集合，即
- $S^i = \{(\mathbf{P}_j, \mathbf{W}_j) | \mathbf{W}_j \text{ 是 } f_i \text{ 曲线中使得 } f_i \text{ 产生一次阶跃的 } X \text{ 值, } \mathbf{P}_j = f_i(\mathbf{W}_j), 0 \leq j < r\}$ 
  - $(\mathbf{P}_0, \mathbf{W}_0) = (0, 0)$
  - 共有 $r$ 个阶跃值，分别对应 $r$ 个 $(\mathbf{P}_j, \mathbf{W}_j)$ 序偶， $1 \leq j \leq r$
  - ① 若 $\mathbf{W}_j < \mathbf{W}_{j+1}$ ，则 $\mathbf{P}_j < \mathbf{P}_{j+1}$ ， $0 \leq j < r$ ，即 $f_i$ 是关于 $X$ 的  
单调递增函数
  - ② 若 $\mathbf{W}_j \leq X < \mathbf{W}_{j+1}$ ， $f_i(X) = f_i(\mathbf{W}_j)$ ，即具有阶跃特点
  - ③ 若 $X \geq \mathbf{W}_r$ ， $f_i(X) = f_i(\mathbf{W}_r)$



## 5.5 0/1背包问题

### ● 2. 序偶表示

(P, W)



$(P_j, W_j): P_j = f_i(W_j)$

$$f_2(X) = \begin{cases} -\infty & X < 0 \\ \max\{0, -\infty + 2\} = 0 & 0 \leq X < 2 \\ \max\{1, -\infty + 2\} = 1 & 2 \leq X < 3 \\ \max\{1, 0 + 2\} = 2 & 3 \leq X < 5 \\ \max\{1, 1 + 2\} = 3 & X \geq 5 \end{cases}$$

# 5.5 0/1背包问题

## ● 2. 序偶表示

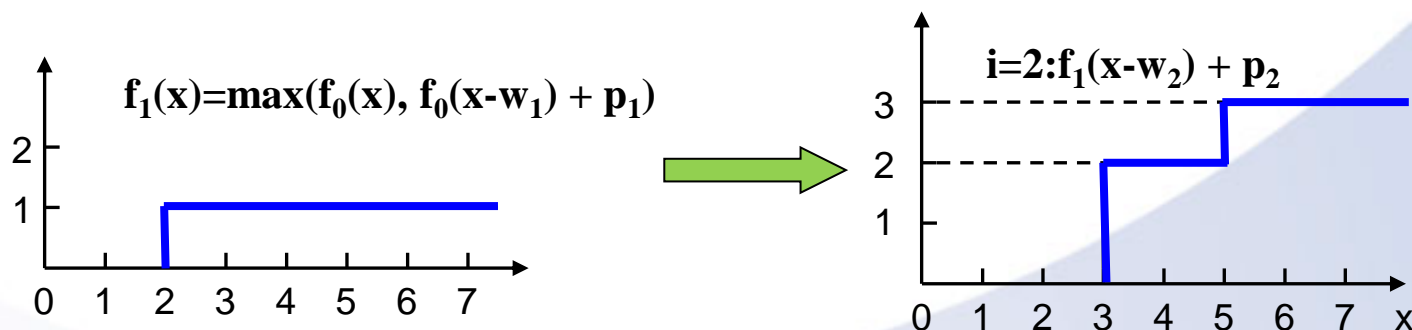
### – $S^i$ 的构造

➤ 记  $S_1^i$  是  $f_{i-1}(X-w_i)+p_i$  的所有序偶的集合，则

$$S_1^i = \{(P, W) \mid (P - p_i, W - w_i) \in S^{i-1}\}$$

➤ 其中  $S^{i-1}$  是  $f_{i-1}$  的所有序偶的集合

即：在  $S_{i-1}$  的序偶分量上增加  $p_i$ 、 $w_i$  生成



## 5.5 0/1背包问题

### ● 2. 序偶表示

#### – $S^i$ 的构造

➤ 由 $S^{i-1}$ 和 $S_1^i$ 按照支配规则合并而成。

➤ 支配规则:

✓ 如果 $S^{i-1}$ 和 $S_1^i$ 之一有序偶 $(P_j, W_j)$ ,另一有 $(P_k, W_k)$ ,  
且有  $W_j \geq W_k, P_j \leq P_k$ , 则序偶 $(P_j, W_j)$ 将被舍弃。  
(反映曲线合并过程中的取大值规则。)

✓ 注:  $S^i$ 中的所有序偶是背包问题 $\text{KNAP}(1, i, X)$   
在 $X$ 各种取值下的最优解。



## 例5.12 例5.11的序偶计算

$$(w_1, w_2, w_3) = (2, 3, 4), (p_1, p_2, p_3) = (1, 2, 5), M = 6$$

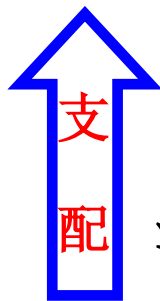
$$S^0 = \{(0, 0)\}$$

$$S_1^1 = \{(1, 2)\}$$

$$S^1 = \{(0, 0), (1, 2)\}$$

$$S_1^2 = \{(2, 3), (3, 5)\}$$

$$S^2 = \{(0, 0), (1, 2), (2, 3), \textcolor{red}{(3, 5)}\} \quad S_1^3 = \{(5, 4), (6, 6), (7, 7), (8, 9)\}$$



注：序偶(3, 5)被(5, 4)“支配”而删除

$$S^3 = \{(0, 0), (1, 2), (2, 3), (5, 4), (6, 6), \textcolor{blue}{(7, 7)}, \textcolor{blue}{(8, 9)}\}$$



## 5.5 0/1背包问题

### ● 2. 序偶表示

#### – $S^i$ 的构造

- 在 $S^i$ 中，没有两个完全一样的序偶存在，即不存在 $i$ 和 $k$ ，使得 $(P_j, W_j), (P_k, W_k) \in S^i$  且  $W_j = W_k$  且  $P_j = P_k$ ,
- 也不存在 $W_j = W_k$  或  $P_j = P_k$ 。
- 若 $W_j > W_k$  则  $P_j > P_k$ ，反之亦然，即序偶同时按照 $W_i$ 和 $P_i$ 递增有序。





## 5.5 0/1背包问题

### ● 3. 决策序列的求取

– 如何求取决策序列？

➤ 分析 $S^i$ 中序偶的来源：

✓  $S^i$ 中的序偶或者来源于 $S^{i-1}$  或者来源于  $S_1^i$  。

✓ 若来源于 $S^{i-1}$ ，则对当前的 $W$ 计算 $f_i(X)$ 时，表达式中 $f_{i-1}(X)$ 的值大些，故第 $i$ 件物品不装为好，即 $x_i = 0$ 。否则

✓ 来源于 $S_1^i$ ， $f_{i-1}(X - W_i) + P_i$ 的效益值好些，第 $i$ 件物品应该装入背包， $x_i = 1$ 。

$$f_i(X) = \max\{f_{i-1}(X), f_{i-1}(X - w_i) + p_i\}$$



## 5.5 0/1背包问题

### ● 3. 决策序列的求取

– KNAP(1, n, M)问题的解——决策序列的求取

- ① 生成序偶集 $S^i$ 。(应将 $W > M$ 的那些序偶 $(P, W)$ 去掉，因为它们不能导出满足约束条件的可行解。)
- ②  $S^n$ 是KNAP(1, n, X)在 $0 \leq X \leq M$ 各种取值下的最优解。
- ③ 通过计算 $S^n$ 可以找到KNAP(1, n, X),  $0 \leq X \leq M$ 的所有解。
- ④ KNAP(1, n, M)的最优解由 $S^n$ 的最后一对有效序偶(具有有效的最大 $W$ 值的序偶)给出。



## 5.5 0/1背包问题

### ● 3. 决策序列的求取

– KNAP(1, n, M)问题的解——决策序列的求取

⑤  $x_n$ 的计算。

- ✓ 设 $S^n$ 的最后一对有效序偶是 $(P_1, W_1)$ ,  $W_1 \leq M$ ,
- ✓ 则 $(P_1, W_1)$ 或者是 $S^{n-1}$ 的最末一对有效序偶,
- ✓ 或者是  $(P_j + p_n, W_j + w_n)$ , 其中  $(P_j, W_j) \in S^{n-1}$  且  $W_j$  是 $S^{n-1}$ 中满足 $W_j + w_n \leq M$ 的最大值。
- ✓ 若 $(P_1, W_1) \in S^{n-1}$ , 则 $x_n = 0$ ; 否则,
- ✓  $(P_1 - p_n, W_1 - w_n) \in S^{n-1}$ ,  $x_n = 1$



## 5.5 0/1背包问题

### ● 3. 决策序列的求取

– KNAP(1, n, M)问题的解——决策序列的求取

⑥  $x_{n-1}$  的计算。

✓ 若  $x_n=0$ ，则判断  $S^{n-1}$  中  $(P_1, W_1)$  的来源，以确定  $x_{n-1}$  的值

✓ 若  $x_n=1$ ，则判断  $S^{n-1}$  中  $(P_1-p_n, W_1-w_n)$  的来源，以确定  $x_{n-1}$  的值

⑦  $x_{n-2}, \dots, x_1$  将依次推导得出。



### 例5.13 (例5.12)

$$S^0 = \{(0, 0)\}$$

$$S^1 = \{(0, 0), (1, 2)\}$$

$$S^2 = \{(0, 0), (1, 2), (2, 3), (3, 5)\}$$

$$S^3 = \{(0, 0), (1, 2), (2, 3), (5, 4), (6, 6), (7, 7), (8, 9)\}$$

$M=6$ ,  $f_3(6)$ 由 $S^3$ 中的序偶 $(6, 6)$ 给出。

$$1) \because (6, 6) \notin S^2$$

$$\therefore x_3 = 1$$

$$2) \because (6-p_3, 6-w_3) = (1, 2) \in S^2 \text{ 且 } (1, 2) \in S^1$$

$$\therefore x_2 = 0$$

$$3) \because (1, 2) \notin S^0$$

$$\therefore x_1 = 1$$



## 算法5.6 非形式化的背包算法

**procedure** DKP( $p, w, n, M$ )

$S^0 \leftarrow \{(0, 0)\}$

**for**  $i \leftarrow 1$  **to**  $n-1$  **do**

$S_1^i \leftarrow \{(P_1, W_1) | (P_1 - p_i, W_1 - w_i) \in S^{i-1} \text{ and } W_1 \leq M\}$

$S^i \leftarrow \text{MERGE-PURGE}(S^{i-1}, S_1^i)$

**repeat**

$(P_X, W_X) \leftarrow S^{n-1}$ 的最末一个有效序偶

$(P_Y, W_Y) \leftarrow (P_1 + p_n, W_1 + w_n)$ , 其中,  $W_1$ 是 $S^{n-1}$ 中使得 $W + w_n \leq M$ 的所有序偶中取最大值得 $W$

//沿 $S^{n-1}, \dots, S^1$ 回溯确定 $x_n, x_{n-1}, \dots, x_1$ 的取值//

**if**  $P_X > P_Y$  **then**  $x_n \leftarrow 0$  //  $P_X$ 将是 $S^n$ 的最末序偶//

**else**  $x_n \leftarrow 1$  //  $P_Y$ 将是 $S^n$ 的最末序偶//

**endif**

回溯确定 $x_{n-1}, \dots, x_1$

**end** DKP



中国科学院大学

University of Chinese Academy of Sciences 22

## 5.5 0/1背包问题

### ● 4. DKP的实现

#### – 序偶集 $S^i$ 的存储结构

- 使用两个一维数组**P**和**W**存放所有的序偶( $P_1, W_1$ ), 其中**P**存放 $P_1$ 值, **W**存放 $W_1$ 值
- 序偶集 $S^0, S^1, \dots, S^{n-1}$ 顺序、连续存放于**P**和**W**中;
- 用指针**F(i)**表示 $S^i$ 中第一个元素在数组 (**P**, **W**)中的下标位置,  $0 \leq i \leq n$ ;
- $F(n) = S^{n-1}$ 中最末元素位置 + 1

	1	2	3	4	5	6	7	8
P	0	0	1	0	1	2	3	
W	0	0	2	0	2	3	5	

↑  
**F(0)**

↑  
**F(1)**

↑  
**F(2)**

↑  
**F(3)**



中国科学院大学

University of Chinese Academy of Sciences 23

## 5.5 0/1背包问题

### ● 4. DKP的实现

#### – 序偶的生成与合并

- $S^i$ 的序偶将按照 $P$ 和 $W$ 的递增次序生成
- $S_1^i$ 中序偶的生成将与 $S_1^i$ 和 $S^{i-1}$ 的合并同时进行
- 设 $S_1^i$ 生成的下一序偶是 $(pp, ww)$ ；对所有的 $(pp, ww)$ ，根据支配规则处理如下：

- ①  $S^{i-1}$ 中所有 $W < ww$ 的序偶 $(P, W)$ 加入 $S^i$
- ② 由支配规则看 $(pp, ww)$ 是否加入 $S^i$ 
  - a. 若 $S^{i-1}$ 中有 $W_{g+1} = ww$ ，则 $pp \leftarrow \max\{p_{g+1}, pp\}$
  - b. 若 $S^{i-1}$ 中有 $p_g > pp$ ，则舍弃 $(pp, ww)$
  - c. 若不舍弃 $(pp, ww)$ ，则 $(pp, ww) \rightarrow S^i$





## 5.5 0/1背包问题

### ● 4. DKP的实现

#### – 序偶的生成与合并

- ③ 考虑 $S^{i-1}$ 中 $W > ww$ 的序偶有无被 $(pp, ww)$ 所支配, 若 $P_{g+1} < pp$ ,  $(P_{g+1}, W_{g+1})$ 被舍弃
- ④ 对所有的 $(pp, ww)$ 重复上述处理;
- ⑤ 将最后 $S^{i-1}$ 中剩余的序偶直接计入 $S^i$ 中, (是一些 $P$ 和 $W$ 均较大的序偶);
- ⑥ 所有计入 $S^i$ 中的新序偶依次存放到由 $F(i)$ 指示的 $S^i$ 的存放位置上。

➤注: 不需要存放 $S_1^i$ 的专用空间



## 5.5 0/1背包问题

### ● 4. DKP的实现

#### – 算法中变量的含义

- $F(i)$ :  $S^i$ 中第一对序偶在数组中的位置(下标)
- $l, h$ :  $S^{i-1}$ 的第一对序偶和最后一对序偶在数组中的位置, 即 $F(i-1)$ 和 $F(i)-1$ .
- $k$ :  $S^{i-1}$ 中当前要加入 $S^i$ 的序偶的位置.
- $u$ : 在 $S^{i-1}$ 能够产生 $S^i_1$ 序偶的最后一个位置, 即对于  $u+1 \leq v \leq h$  的序偶  $(P_v, W_v)$ , 有  $W_v + w_i > M$ .
- $j$ : 当前正在生成 $S^i_1$ 的 $S^{i-1}$ 中序偶的位置.
- $next$ :  $S^i$ 中要加入序偶的位置.



## 5.5 0/1背包问题

### ● 4. DKP的实现

#### – 算法的思想

##### ➤ 1. 初始化

最初只有 $S^0$ 的信息,  $F(0), P(1), W(1), l, h, F(1), next$

##### ➤ 2. 生成 $S^i$

对 $k, u$ 赋值

(1) 依次生成 $S^i_1$ 中的序偶 $(pp, ww)$

在 $S^{i-1}$ 中, 重量比 $ww$ 小的序偶 $(P(k), W(k))$ 加入 $S^i$ 中 $(pp, ww)$ 是否被支配;

$S^{i-1}$ 中 $(pp, ww)$ 被支配的序偶

(2)  $S^i_1$ 中的序偶都生成后,  $S^{i-1}$ 中若还有序偶没有加入 $S^i$ 中, 则全部加入;

$l, h, F(i+1)$ 赋值

##### ➤ 3. 生成最末序偶, 回溯构造最优决策序列



## 算法5.7 0/1背包问题的算法描述

**procedure** DKNAP( $p, w, n, M, m$ )

**real**  $p(n), w(n), P(m), W(m), pp, ww, M$ ; **integer**  $F(0:n), l, h, u, i, j, p, next$

$F(0) \leftarrow 1; P(1) \leftarrow W(1) \leftarrow 0$  //  $S^0$  //

$l \leftarrow h \leftarrow 1$  //  $S^0$  的首端和末端;  $l$  是  $S^{i-1}$  的首端,  $h$  是  $S^{i-1}$  的末端 //

$F(1) \leftarrow next \leftarrow 2$  //  $P$  和  $W$  中第一个空位;  $next$  指示  $P/W$  中可以存放  $(P, W)$  序偶的第一个位置 //

**for**  $i \leftarrow 1$  **to**  $n-1$  **do** // 生成  $S^i$  //

$k \leftarrow l$

$u \leftarrow$  在  $l \leq r \leq h$  中使得  $W(r) + w_i \leq M$  的最大  $r$  //  $u$  指示由  $S^{i-1}$  生成  $S^i_1$  的最大有效位置 //

**for**  $j \leftarrow l$  **to**  $u$  **do** // 生成  $S^i_1$ , 同时进行归并 //

$(pp, ww) \leftarrow (P(j) + p_i, W(j) + w_i)$  // 生成  $S^i_1$  中的下一个元素 //

**while**  $k \leq h$  and  $W(k) < ww$  **do** // 从  $S^{i-1}$  中取元素并归并 //

$P(next) \leftarrow P(k); W(next) \leftarrow W(k)$  // 所有  $W(k) < ww$  的序偶直接归并 //

$next \leftarrow next + 1; k \leftarrow k + 1$

**repeat**



```

//按照支配规则考虑(pp, ww)及 $S^{i-1}$ 中的序偶//
if  $k \leq h$  and  $W(k) = ww$  then
     $pp \leftarrow \max(pp, P(k))$   $k \leftarrow k+1$ 
endif
if  $pp > P(\text{next}-1)$  then  $(P(\text{next}), W(\text{next})) \leftarrow (pp, ww)$ 
     $\text{next} \leftarrow \text{next}+1$ 
endif
//清除 $S^{i-1}$ 中的序偶//
while  $k \leq h$  and  $P(k) \leq P(\text{next}-1)$  do  $k \leftarrow k+1$  repeat
repeat
//将 $S^{i-1}$ 中剩余的元素并入 $S^i$  //
while  $k \leq h$  do
     $(P(\text{next}), W(\text{next})) \leftarrow (P(k), W(k))$ 
     $\text{next} \leftarrow \text{next}+1; k \leftarrow k+1$ 
repeat
//对 $S^{i+1}$ 置初值 //
 $l \leftarrow h+1; h \leftarrow \text{next}-1; F(i+1) \leftarrow \text{next}$ 
repeat
CALL PARTS //递推求取 $x_n, x_{n-1}, \dots, x_1$ //
END DKNAP

```



## 5.5 0/1背包问题

### ● 5. 过程DKNAP的分析

– 空间复杂度

记 $S^i$ 中的序偶数为:  $|S^i|$

则有,  $|S^i| \leq |S^{i-1}| + |S_1^i|$

又,  $|S_1^i| \leq |S^{i-1}|$

所以,  $|S^i| \leq 2|S^{i-1}|$

最坏情况下有(由 $S^{i-1}$ 生成 $S_1^i$ 和 $S^i$ 时没有序偶被支配):

$$\sum_{0 \leq i \leq n-1} |S^i| = \sum_{0 \leq i \leq n-1} 2^i = 2^n - 1$$

故, DKNAP所需的空間复杂度(P、W数组)为:  $O(2^n)$



## 5.5 0/1背包问题

### ● 5. 过程DKNAP的分析

#### – 时间复杂度

➤ 由 $S^{i-1}$ 生成 $S^i$ 的时间为： $\Theta(|S^{i-1}|)$ ， $0 \leq i \leq n-1$

➤ 故，DKNAP计算所有的 $S^i$ 所需的时间为：

$$\sum_{0 \leq i \leq n-1} |S^{i-1}| = O(2^n)$$



## 5.5 0/1背包问题

### ● 5. 过程DKNAP的分析

#### – 时间复杂度

若每件物品的重量 $w_i$ 和效益值 $p_i$ 均为**整数**，则 $S^i$ 中每个序偶 $(P, W)$ 的 $P$ 值和 $W$ 值也是整数，且有 $P \leq \sum_{0 \leq j \leq i} |p_j|$ ， $W \leq M$

又，在任一 $S^i$ 中的所有序偶具有互异 $P$ 值和 $W$ 值，故有

$$|S^i| \leq 1 + \sum_{0 \leq j \leq i} |p_j| \text{ 且 } |S^i| \leq 1 + \min\left\{\sum_{0 \leq j \leq i} |w_j|, M\right\}$$

在所有 $w_j$ 和 $p_j$ 均为**整数**的情况下，DKNAP的时间和空间复杂度将为：

$$O(\min\{2^n, n \sum_{1 \leq i \leq n} |p_i|, nM\})$$





## 5.5 0/1背包问题

### ● 6. 序偶集合的一种启发式生成策略

- 在由 $S^0$ 生成 $S^n$ 的过程中，有些序偶无论如何也不会导致问题的最优解——问题的最优解由最大有效序偶给出。
- 这些序偶最终也不会出现在任何最优决策序列中，故可以及时的舍去，以进一步降低计算量。

例：

$$S^2 = \{(\mathbf{0}, \mathbf{0}), (\mathbf{1}, \mathbf{2}), (2, 3), (3, 5)\}$$

$$p_3 = 2, w_3 = 3$$

$$(0, 0) \rightarrow (2, 3)$$

$$(1, 2) \rightarrow (3, 5)$$

怎样预知背包的可能最好效益？



中国科学院大学

University of Chinese Academy of Sciences 33

## 5.5 0/1背包问题

### ● 6. 序偶集合的一种启发式生成策略

- 设 $L$ 是最优解的估计值, 且有 $f_n(M) \geq L$
- 设 $PLEFT(i) = \sum_{i < j \leq n} p_j$ , 即 $i+1$ 至 $n$ 件物品的效益值之和
- 若正在生成的序偶 $(P, W)$ 有 $P + PLEFT(i) < L$ , 则 $(P, W)$ 将不计入 $S^i$ 中。
- $L$ 的选择:
  - ① 取 $S^i$ 的最末序偶 $(P, W)$ 的 $P$ 作为 $L$ ,  $P \leq f_n(M)$
  - ② 将某些剩余物品的 $p$ 值+ $P$ 作为 $L$

例:  $p_3=2, w_3=3 \rightarrow PLEFT(2) = 2$

取 $L=6$

$(0, 0) \rightarrow (2, 3), 0 + PLEFT(2) = 2 < 6$

$(1, 2) \rightarrow (3, 5), 1 + PLEFT(2) = 3 < 6$



### 例5.15 0/1背包问题

$n=6, (p_1, p_2, p_3, p_4, p_5, p_6) = (w_1, w_2, w_3, w_4, w_5, w_6) = (100, 50, 20, 10, 7, 3), M=165$

不使用启发方法的序偶集

$$S^0 = \{0\}$$

$$S^1 = \{0, 100\}$$

$$S^2 = \{0, 50, 100, 150\}$$

$$S^3 = \{0, 20, 50, 70, 100, 120, 150\}$$

$$S^4 = \{0, 10, 20, 30, 50, 60, 70, 80, 100, 110, 120, 130, 150, 160\}$$

$$S^5 = \{0, 7, 10, 17, 20, 27, 30, 37, 50, 57, 60, 67, 70, 77, 80, 87, 100, 107, 110, 117, 120, 127, 130, 137, 150, 157, 160\}$$

则,  $f_6(165) = 163$

注: 每对序偶(P, W)仅用单一量P(或W)表示



$n=6, (p_1, p_2, p_3, p_4, p_5, p_6) = (w_1, w_2, w_3, w_4, w_5, w_6) = (100, 50, 20, 10, 7, 3), M=165$

启发式规则求解

分析：将物品1, 2, 4, 6装入背包，将占用163的重量并产生163的效益。  
故，取期望值 $L=163$ 。

按照启发式生成规则，从 $S^i$ 中删除所有 $P+PLEFT(i) < L$ 的序偶，则有

$$PLEFT(0) = p_1 + p_2 + p_3 + p_4 + p_5 + p_6 = 190$$

$$S^0 = \{0\} \quad S_1^1 = \{100\} \quad PLEFT(1) = p_2 + p_3 + p_4 + p_5 + p_6 = 90$$

$$S^1 = \{100\} \quad S_1^2 = \{150\} \quad PLEFT(2) = p_3 + p_4 + p_5 + p_6 = 40$$

$$S^2 = \{150\} \quad S_1^3 = \emptyset \quad PLEFT(3) = p_4 + p_5 + p_6 = 20 \quad (w_3 = 20)$$

$$S^3 = \{150\} \quad S_1^4 = \{160\} \quad PLEFT(4) = p_5 + p_6 = 10$$

$$S^4 = \{160\} \quad S_1^5 = \emptyset \quad PLEFT(5) = p_6 = 3 \quad (w_5 = 7)$$

$$S^5 = \{160\} \quad PLEFT(6) = 0$$

$$f_6(165) = 160 + 3 = 163$$



中国科学院大学

University of Chinese Academy of Sciences 36

## 5.5 0/1背包问题

### ● 作业

- 用序偶的方式求0/1背包问题,  
 $n=4$ ,  
 $(w_1, w_2, w_3, w_4)=(5, 3, 4, 7)$ ,  
 $(p_1, p_2, p_3, p_4)=(3, 2, 5, 9)$ ,  
 $M=15$
- 作业提交到课程网站上
- Word文档即可



# End

