动态规划

Dynamic Programming

主讲人： 薛 毅 教授

北京工业大学应用数理学院

运筹学学科部主任

xueyi@bjut.edu.cn

第四章
动态规划

第一节 最短路问题与动态规划的基本思想

第二节 资源分配问题

第三节 背包问题

第四节 多阶段生产安排问题

第五节 用LINGO软件求解动态规划问题

1 最短路问题与动态规划的基本思想

动态规划是运筹学的一个重要分支，它是解决多阶段决策过程最优化的一种数学手段，它为确定最优决策提供了一个系统的过程。

动态规划是由Richard Bellman (理查德·贝尔曼)提出并加以解决的，他提出了解决问题的“最优化原理”。一般认为，从他的专著“Dynamic Programming, Princeton University Press, N. J. 1957”作为动态规划的开始。

与线性规划相比较，动态规划不存在所谓求解动态规划问题的标准的数学公式，而是提供了适应于某一类问题的求解方法、特定的方程以及分析问题的基本思想。所以当我们求解动态规划问题时，更应注重从解题过程中来培养对问题求解的洞察力与创造力。

1.1. 最短路问题

例4.1.1 在图4.1.1中，用圈表示城市，现有A, B_1, B_2, C_1, C_2, C_3和D共7 个城市。圈间的连线表示城市间有道路相连。连线旁的数字表示道路的长度。现要从城市A到D之间寻找一条的最短路线。

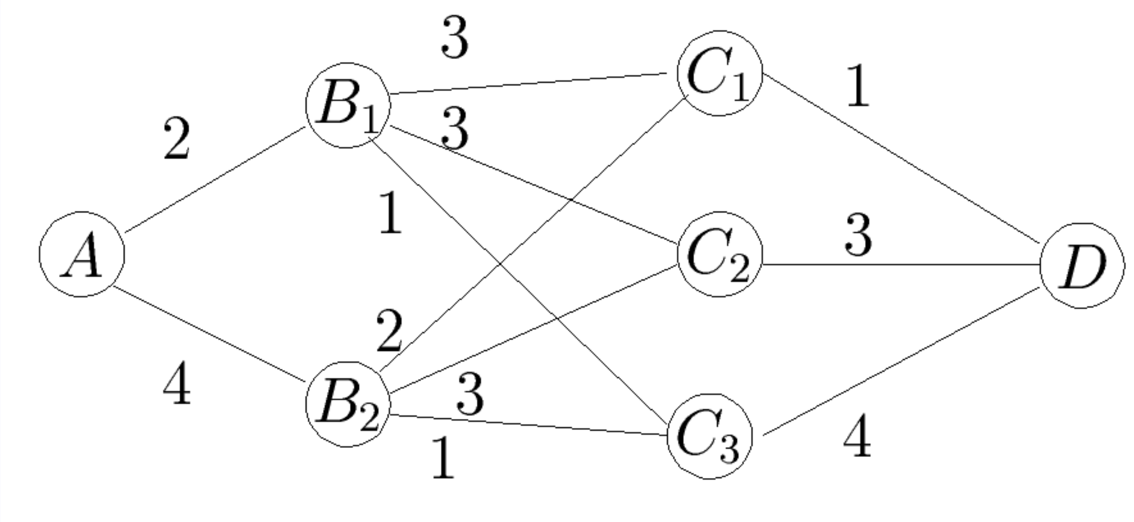

图4.1.1 最短路问题

最短路问题有很强的应用背景，如要建设一条从A城市到D城市的输油管道，如何找出铺设的最短路线；如A 表示商品的出口国，D表示商品的进口国，而B, C的各点表示有可能发货和收货的海关口岸，如何找一条最短进出口渠道等。

还可以换个轻松点的话题。有一些学生，暑假要从A市到D市去旅游，中间想在B和C的某个城市作短暂停留，顺便看看那里的风景，但由于经济能力有限，他们又不想花更多的钱，请你帮他们设计出一条最短的旅游路线。

还有许许多多的问题可以归结为最短路问题。最短路问题也称为驿站问题，问题起源于十九世纪中期，住在美国密苏里(Missouri)的人们要去美国西部的加利福尼亚(California)淘金，尽管他们出发的起点和到达目的地的终点是固定的，但中间途经的一些州和要选择的一些驿站是不确定的。在当时的情况下，旅途是不安全的，因为可能会受到各种各样的抢劫者的攻击，因此需要设计一条安全到达目的地的路线。有一位小心谨慎旅行者要认真考虑他的安全问题。在经过一段时间的思考后，他提出了一个相当聪明的确定安全路线的方法。过往驿站的旅客可以拿到人寿保险单，而保险单的费用是基于驿站的旅客安全的一个细致的估计，因此，最安全的路线就是那一条使得总保险单费用最低的路线。

1.2. 问题的求解

从图4.1.1来看，该问题的解决似乎并不难，直观的想法就是枚举法，全部列出从城市A到D的全部路线，即从$2 \times 3 \times 1 = 6$个路线中，挑出一种最好方案就可以了。但是，当城市的个数增加或情况变的复杂后，这种方法就不可取了。因为枚举法的计算量太大，即便是目前的高速度计算机，在城市的个数稍大一点后，计算量也是不可接受的。

如何解决这个问题呢？我们将问题分解一下，我们先并不急于求出城市A到城市D的最短路，我们先考虑最后一个城市到D市的最短路，也就是从C的各城市到D市的最短路。这个问题好解决，因为从C的各点到D点之间只有一条路线，即$C_i \to D$。

这个问题解决后，我们再考虑从B的各点到D点的最短路。这个问题也不难，因为从B的各点到D点之间，只经过C的各点，而C的各点到D点之间的最短路我们已经计算过了。

当我们得到B的各点到D点的最短路后，我们就可以考虑从A点到D点的最短路了，因为有了前面的计算，这个问题已经简单多了。

这就给我们求解问题提供了一种思想，就是将问题分成多个阶段，一个一个阶段地考虑问题，将一个复杂的问题简单化，这就是解动态规划问题的基本思想。图4.1.2给出了我们对求解问题的思考图。

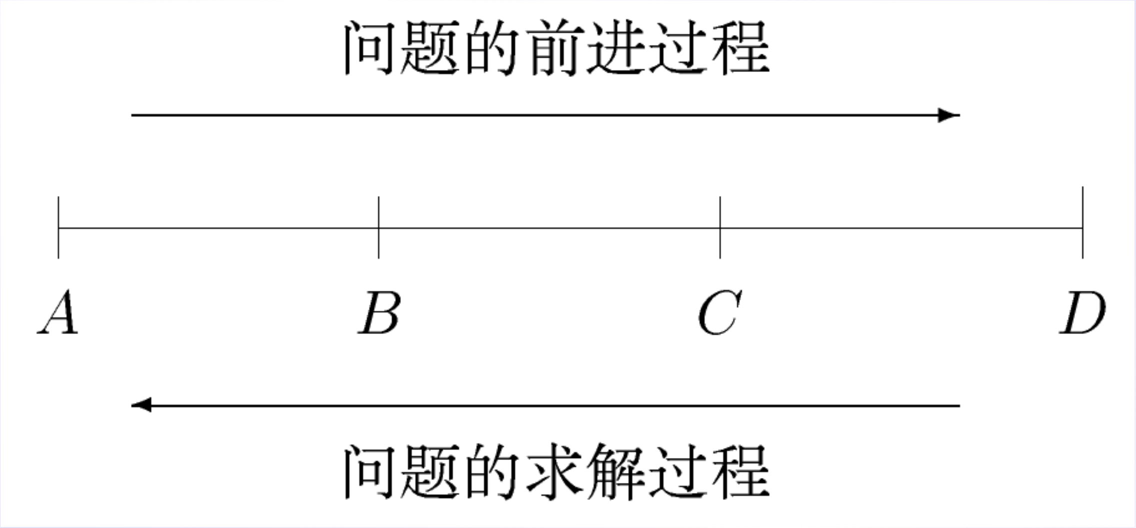

图4.1.2　问题的求解过程的示意图

在介绍动态规划的方法以前，我们结合最短路实例对几个术语进行解释，这些术语是动态规划中常常提到的。

(1) 阶段。在采用动态规划求解时，首先要把整个过程分为几个阶段。

对于上述最短路线问题，可分为3 个阶段来进行讨论。第一阶段：A到$\{B_1, B_2\}$，第二阶段：$\{B_1, B_2\}$到$\{C_1, C_2, C_3\}$，第三阶段：$\{C_1, C_2, C_3\}$到D。

(2) 状态。状态表示某个阶段的起点，也是前一阶段的终点。

描述状态的变量称为状态变量。

例如，在最短路问题中，第一阶段起点只有1个A，因而有一个状态。但第二阶段有2个状态$\{B_1, B_2\}$，第三个状态有3 个状态$\{C_1, C_2, C_3\}$，状态的选取具有无后效性。

所谓无后效性，就是指状态有这样的性质：在某阶段状态给定后，过程今后的发展仅与当前状态有关，而与过去的历史，即它是怎样到达当前状态无关。

对于最短路线来讲，假如当前处于C_1状态，则从C_1到D的最短路线仅与当前状态C_1有关，而与前面两阶段怎样到达C_1无关。

(3) 决策。决策就是作出决定。

例如，在第一阶段过程处于状态A, 此时，从A到$\{B_1, B_2\}$有二种可能性，即由A到B_1, 或由A到B_2. 此时，称$\{B_1, B_2\}$为第一阶段的允许决策集合，记为$D(A) = \{B_1, B_2\}$. 当做出决定从A到B_1后，记为$u_1(A) = B_1$, 称为第一阶段决策。

反之，若从A到B_2, 则此阶段决策写成$u_1(A) = B_2$.

在第二阶段中，同样，$D(B_1) = \{C_1, C_2, C_3\}$, $D(B_2) = \{C_1, C_2, C_3\}$. 若选择由$B_1$到$C_1$, 则第二阶段决策为$u_2(B_1) = C_1$.

由此类推，可以得出各个阶段的决策。

(4) 策略。策略是一个按顺序排列的决策组成的集合。

例如，在最短路的例子中，选择一条路线

$$A \to B_1 \to C_1 \to D$$

就是一个策略。

在实际问题中，可供选择的策略有一定的范围，这些范围称为允许策略集合，如在最短路中，允许的策略集合为

$$\begin{array}{ll} A \to B_1 \to C_1 \to D & A \to B_2 \to C_1 \to D \\ A \to B_1 \to C_2 \to D & A \to B_2 \to C_2 \to D \\ A \to B_1 \to C_3 \to D & A \to B_2 \to C_3 \to D \end{array}$$

从允许策略集合中找出最优效果的策略称为最优策略。

(5) 状态转移方程。状态转移方程是确定前一阶段状态转变到后一状态的过程。

如果给定了第k阶段的状态变量u_k和决策变量$x_k(u_k)$, 则第$k+1$阶段的状态变量u_{k+1}也随之确定了。记为

$$u_{k+1} = T_k(u_k, x_k(u_k)).$$

(6) 指标函数和最优值函数。用来衡量过过程优劣的数量指标，称为指标函数。

指标函数与该阶段的状态变量和该阶段后各阶段的决策变量有关，记为$V_{k,n}$, 即

$$V_{k,n} = V_{k,n}(u_k, x_k, x_{k+1}, \cdots, x_n).$$

称指标函数的最优值为最优值函数，记为$f_k(u_k)$, 即

$$f_k(u_k) = \text{opt}\ V_{k,n},$$

其中opt表示求最优，有时求最大(max), 有时则求最小(min).

下面来求解最短路问题。按照前面讲的思想，从最后一段开始计算，由后向前逐步移至A.

第三阶段。由C_1到终点D只有一条路，因此，$f_3(C_1)=1$.

同理得到，$f_3(C_2)=3$, $f_3(C_3)=4$.

第二阶段。有二个出发点B_1, B_2. 若从B_1出发，则有三种选择，分别是至C_1, C_2和C_3到D, 因此，

$$f_2(B_1) = \min\left\{\begin{array}{l}\underline{d(B_1, C_1) + f_3(C_1)} \\ d(B_1, C_2) + f_3(C_2) \\ d(B_1, C_3) + f_3(C_3)\end{array}\right\} = \min\left\{\begin{array}{l}\underline{3+1} \\ 3+3 \\ 1+4\end{array}\right\} = 4.$$

相应的决策为$u_2(B_1) = C_1$. 这说明，由B_1至终点D的最短距离为4，其最短路线是

$$B_1 \to C_1 \to D.$$

同理，从B_2出发有

$$f_2(B_2) = \min\left\{\begin{array}{c} \underline{d(B_2, C_1) + f_3(C_1)} \\ d(B_2, C_2) + f_3(C_2) \\ d(B_2, C_3) + f_3(C_3) \end{array}\right\} = \min\left\{\begin{array}{c} \underline{2+1} \\ 3+3 \\ 1+4 \end{array}\right\} = 3.$$

相应的决策为$u_2(B_2) = C_1$. 这说明，由B_2至终点D的最短距离为3，其最短路线是

$$B_2 \to C_1 \to D.$$

第一阶段。只有一个出发点A，则有

$$f_1(A) = \min \left\{ \begin{matrix} d(A, B_1) + f_2(B_1) \\ d(A, B_2) + f_2(B_2) \end{matrix} \right\} = \min \left\{ \begin{matrix} 2+4 \\ 4+3 \end{matrix} \right\} = 6.$$

相应的决策为$u_1(A) = B_1$. 于是得到从起点A至终点D的最短距离为6。

为了找出最短路，再按计算的顺序反过来推，由$u_1(A) = B_1$，$u_2(B_1) = C_1, u_3(C_1) = D$组成一个最优策略，从而找出相应的最短路线：

$$A \to B_1 \to C_1 \to D.$$

事实上，该方法不但找到了从起点A到终点D的最短路，而且找到了从各点到终点的最短路。

从上述计算过程可以看出，在求解的各阶段中，利用了k阶段与$k+1$阶段之间的递推关系：

$$
\begin{aligned}
f_k(x_k) &= \min_{u_k \in D_k(x_k)} \{d(x_k, u_k(x_k)) + f_{k+1}(u_k(x_k))\}, \quad (1.1)\\
& \qquad k = 3, 2, 1,\\
f_4(x_4) &= 0. \quad (1.2)
\end{aligned}
$$

称(1.1)–(1.2)为动态规划的基本方程。

上述最短路线问题的计算过程也可借助于图形直观简明的表示出来，如图4.1.3 所示。

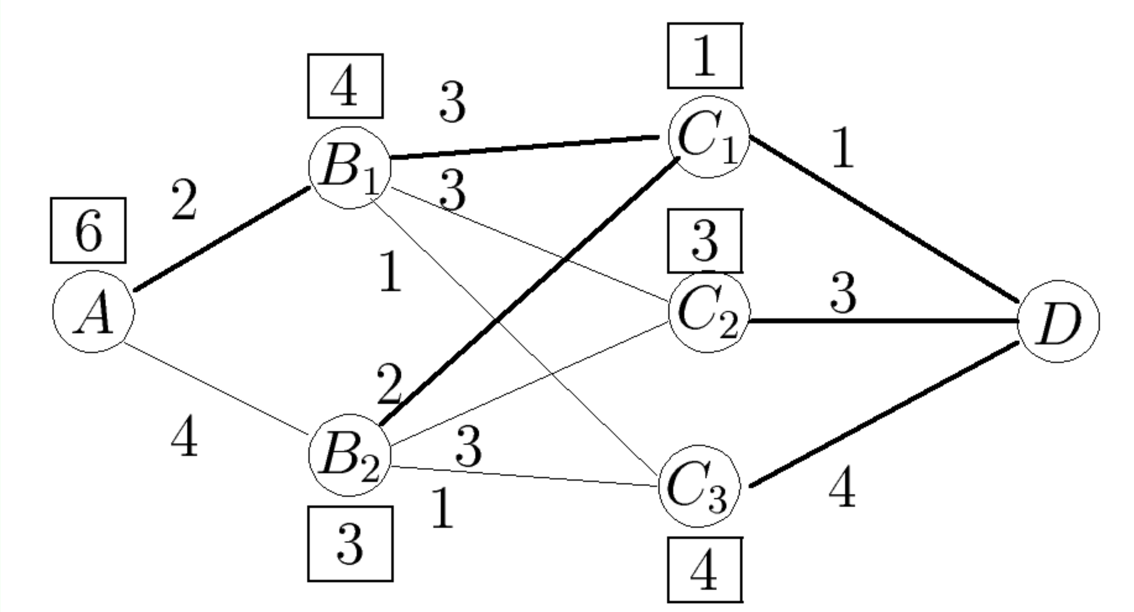

图4.1.3 最短路的计算结果

求解动态规划问题的基本思想需要用到最优化原理，它是R. Bellman在1957年提出的，因此也称为Bellman最优化原理。下面给出最优化原理的陈述。

1.3. 最优化原理(Bellman 最优化原理)

一个过程的最优策略具有这样的性质，即无论其初始状态及其初始决策如何，其后诸决策对以第一个决策所形成的状态作为初始状态而言，必须构成最优策略。

为了更好地体会动态规划的基本思想，再看一个更复杂的例子。

例4.1.2 求如图4.1.4所示的A到G的最短路。

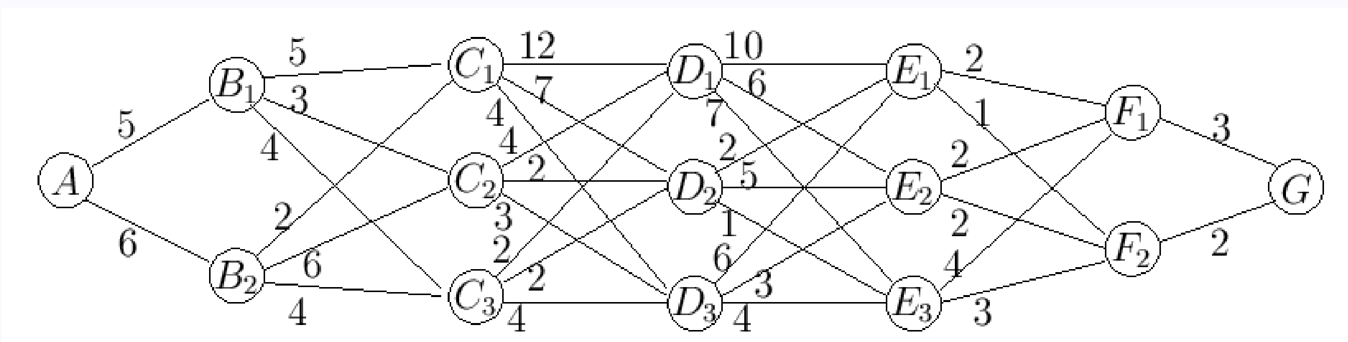

图4.1.4 最短路问题

按照上述方法，在图上直接计算得到：

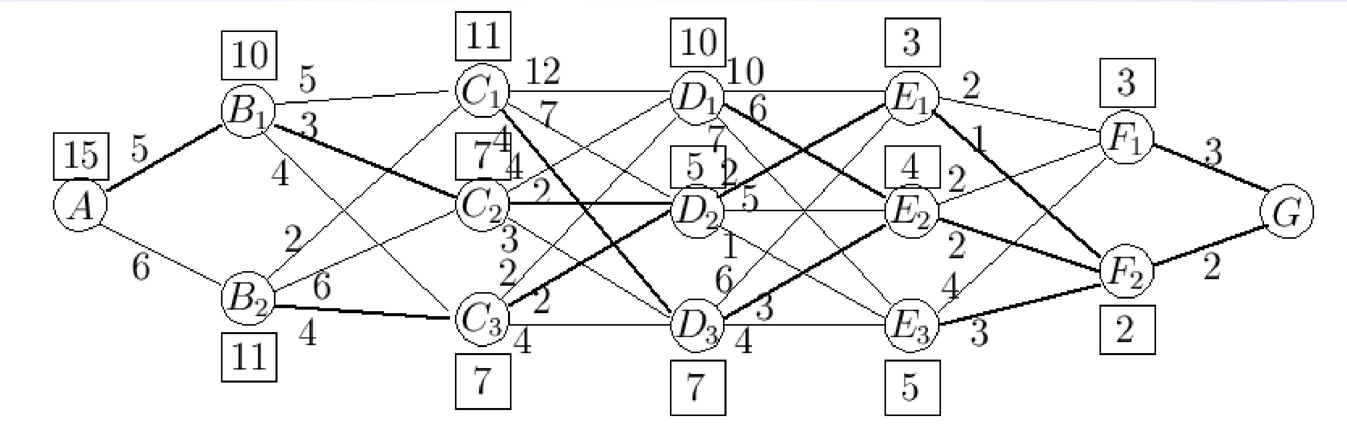

图4.1.5 最短路问题的计算结果

在本节的最后，讲一个海盗分宝的问题。这个问题大家可能见过，在这里请用动态规划的基本思想来求解。

从前有五个海盗，获得100颗大钻石。他们不想平均分配，就想出了这样一种办法：先作五个签分别是1至5，然后进行抽签。由抽到第一签的那个海盗先说出一种分配方案，如果半数（不含半数）以上人同意，就按他的方案进行分配；否则，他就会被仍进海里。再由抽到第二签的人说出下一种分配方案，依次类推。问抽到第一个签的海盗如何分配才能使自己的利益最大化？

在思考问题时应注意以下几点：(1) 半数同意也要被推下海。(2) 如果海盗获得的钻石不变，那么他认为死的人越多越好。(3) 每个海盗都是聪明绝顶。

2 资源分配问题

资源分配问题就是将有限的资源分配给若干个使用者，使得总的效益最大。

设某公司现有一笔资金a(万元)，打算分配给它下属的n个子公司，用于扩大再生产。假设x_i(万元)为第i个子公司分配到的资金数，$g(x_i)$为第i个子公司在得到x_i(万元)资金后所产生的利润(万元)。那么总公司将如何投资，才能使得总利润达到最大？

这种投资问题的数学模型为

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n} g_i(x_i), \\
\text{s.t.} \quad & \sum_{i=1}^{n} x_i = a, \\
& x_i \geq 0, \quad i = 1, 2, \cdots, n.
\end{aligned}
$$

用动态规划方法求解。将问题分成多个阶段。设

$f_k(y)$ — 以数量为y的资金分配给前k个子公司产生的最大利润，

则问题转化为求$f_n(a)$.

分析：我们考虑第一步，将数量为y的资金全部分给第一个子公司所产生的最大利润，即

$$f_1(y) = \max_{0 \le x_1 \le y} g_1(x_1), \tag{2.1}$$

那么，将数量y的资金分给前k个子公司产生的最大利润是多少呢？我们导出相应的状态转移方程。如果将数量为x_k的资金分给第k个子公司，那么剩下的$y - x_k$的资金就分给前$k-1$个子公司，按照我们的假设，前$k-1$个子公司产生的最大利润是$f_{k-1}(y - x_k)$, 所以状态转移方程为

$$f_k(y) = \max_{0 \le x_k \le y} \{g_k(x_k) + f_{k-1}(y - x_k)\}. \tag{2.2}$$

由上述递推公式，就可以计算出$f_n(a)$。

我们看一个例子。假设总公司打算拿出60万元，分配给下属的4个子公司，每个子公司产生的利润$g_i(x_i)$由下表给出

表 4.2.1 四个子公司的利润表

投资 x	0	10	20	30	40	50	60
$g_1(x)$	0	20	50	65	80	85	85
$g_2(x)$	0	20	40	50	55	60	65
$g_3(x)$	0	25	60	85	100	110	115
$g_4(x)$	0	25	40	50	60	65	70

解：第一阶段：求 $f_1(y)$. 由(2.1)式，即

$$f_1(y) = \max_{0 \le x_1 \le y} g_1(x_1) = g_1(y),$$

所以

表 4.2.2　只分配给第一个子公司的最优策略表

投资 y	0	10	20	30	40	50	60
$f_1(y)$	0	20	50	65	80	85	85
最优策略	0	10	20	30	40	50	60

第二阶段：求$f_2(y)$. 由状态转移方程(2.2), 即

$$f_2(y)=\max_{0\le x_2\le y}\{g_2(x_2)+f_1(y-x_2)\}$$

计算$f_2(y)$. 在这里只取离散值，即取$y=0,10,20,30,40,50,60$值的情况。显然有$f_2(0)=0$.

当$y=10$时，有

$$\begin{aligned} f_2(10) &= \max_{x_2=0,10}\{g_2(x_2)+f_1(10-x_2)\} \\ &= \max\{\underline{g_2(0)+f_1(10)}, \underline{g_2(10)+f_1(0)}\} \\ &= \max\{\underline{0+20}, \underline{20+0}\} \\ &= 20, \end{aligned}$$

其最优策略是$(10,0)$或$(0,10)$。

当$y=20$时，有

$$
\begin{aligned}
f_2(20) &= \max_{x_2=0,10,20}\{g_2(x_2)+f_1(20-x_2)\} \\
&= \max\{\underline{g_2(0)+f_1(20)}, g_2(10)+f_1(10), g_2(20)+f_1(0)\} \\
&= \max\{\underline{0+50}, 20+20, 40+0\} \\
&= 50,
\end{aligned}
$$

其最优策略是$(20,0)$.

当$y=30$时，有

$$\begin{aligned}f_2(30) &= \max_{x_2=0,10,20,30}\{g_2(x_2)+f_1(30-x_2)\}\\ &= \max\{g_2(0)+f_1(30),\underline{g_2(10)+f_1(20)},g_2(20)+f_1(10),\\ &\qquad g_2(30)+f_1(0)\}\\ &= \max\{0+65,\underline{20+50},40+20,50+0\}\\ &= 70,\end{aligned}$$

其最优策略是$(20,10)$.

当$y=40$时，有

$$
\begin{aligned}
f_2(40) &= \max_{x_2=0,10,\cdots,40}\{g_2(x_2)+f_1(40-x_2)\} \\
&= \max\{g_2(0)+f_1(40), g_2(10)+f_1(30), \underline{g_2(20)+f_1(20)}, \\
&\qquad g_2(30)+f_1(10), g_2(40)+f_1(0)\} \\
&= \max\{0+80, 20+65, \underline{40+50}, 50+20, 55+0\} \\
&= 90,
\end{aligned}
$$

其最优策略是$(20,20)$.

当$y=50$时，有

$$
\begin{aligned}
f_2(50) &= \max_{x_2=0,10,\cdots,50} \{g_2(x_2)+f_1(50-x_2)\} \\
&= \max\{g_2(0)+f_1(50), g_2(10)+f_1(40), \underline{g_2(20)+f_1(30)}, \\
&\qquad g_2(30)+f_1(20), g_2(40)+f_1(10), g_2(50)+f_1(0)\} \\
&= \max\{0+85, 20+80, \underline{40+65}, 50+50, 55+20, 60+0\} \\
&= 105,
\end{aligned}
$$

其最优策略是$(30,20)$.

当$y=60$时，有

$$\begin{aligned} f_2(60) &= \max_{x_2=0,10,\cdots,60}\{g_2(x_2)+f_1(60-x_2)\} \\ &= \max\{g_2(0)+f_1(60), g_2(10)+f_1(50), \underline{g_2(20)+f_1(40)}, \\ &\qquad g_2(30)+f_1(30), g_2(40)+f_1(20), \\ &\qquad g_2(50)+f_1(10), g_2(60)+f_1(0)\} \\ &= \max\{0+85, 20+85, \underline{40+80}, 50+65, \\ &\qquad 55+50, 60+20, 65+0\} \\ &= 120, \end{aligned}$$

其最优策略是$(40,20)$.

由上述计算，得到前两个子公司的最优方案，如表4.2.3所示

表 4.2.3　只分配给前两个子公司的最优策略表

投资 y	0	10	20	30	40	50	60
$f_2(y)$	0	20	50	70	90	105	120
最优策略	$(0,0)$	$(10,0)$ $(0,10)$	$(20,0)$	$(20,10)$	$(20,20)$	$(30,20)$	$(40,20)$

第三阶段：经过与上述类似的计算，得到前三个子公司的最优方案，如表4.2.4所示.

表 4.2.4　只分配给前三个子公司的最优策略表

投资 y	0	10	20	30	40	50	60
$f_3(y)$	0	25	60	85	110	135	155
最优策略	$(0,0,0)$	$(0,0,10)$	$(0,0,20)$	$(0,0,30)$	$(20,0,20)$	$(20,0,30)$	$(20,10,30)$

最后，第四阶段，再利用(2.2)式。这里只需计算$f_4(60)$

$$\begin{aligned}
f_4(60) &= \max_{x_2=0,10,\cdots,60}\{g_4(x_2)+f_3(60-x_2)\} \\
&= \max\{g_4(0)+f_3(60), \underline{g_4(10)+f_3(50)}, \\
&\qquad g_4(20)+f_3(40), g_4(30)+f_3(30), g_4(40)+f_3(20), \\
&\qquad g_4(50)+f_3(10), g_4(60)+f_3(0)\} \\
&= \max\{0+155, \underline{25+135}, 40+110, 50+85, \\
&\qquad 60+60, 65+25, 70+0\} \\
&= 160,
\end{aligned}$$

现在考虑最优策略。由上式可知，得到最大利润的策略是将50万元的资金分配给前3个子公司，第4个子公司分配10万元。

再由表4.2.4知，前3个子公司得到50万元的最优策略是$(20, 0, 30)$, 这样最终的最优策略为$(20, 0, 30, 10)$.

在上述例子中，$g_i(x_i)$是离散函数。对于一般情况，$g_i(x_i)$是连续函数，求解可能会复杂一些。但它给我们提供了一种求解此类优化问题

$$\max \quad g_1(x_1)+g_2(x_2)+\cdots+g_n(x_n), \tag{2.3}$$

$$\text{s.t.} \quad a_1x_1+a_2x_2+\cdots+a_nx_n=b, \tag{2.4}$$

$$x_1,x_2,\cdots,x_n \geq 0 \tag{2.5}$$

的方法，这里$a_1,a_2,\cdots,a_n$均为正数。

可按如下方法构造出问题(2.3)–(2.5)的状态转移方程。令

$$f_1(y) = \max_{a_1x_1\leq y} g_1(x_1), \tag{2.6}$$

$$f_k(y) = \max_{a_kx_k\leq y} \{g_k(x_k)+f_{k-1}(y-a_kx_k)\}. \tag{2.7}$$

因此，问题的解为$f_n(b)$.

最短路问题与动态规划...
资源分配问题
背包问题
多阶段生产安排问题
用LINGO软件求解动...
访问主页
标题页
第 42 页 共 70 页
返 回
全屏显示
关 闭
退 出

例4.2.1 求解非线性规划问题

$$\begin{aligned} \max \quad & 4x_1^2 - x_2^2 + 2x_3^2 + 12, \\ s.t. \quad & 3x_1 + 2x_2 + x_3 = 9, \\ & x_1, x_2, x_3 \geq 0. \end{aligned}$$

解：按照定义，可得到$g_1(x_1) = 4x_1^2$, $g_2(x_2) = -x_2^2$, $g_3(x_3) = 2x_3^2 + 12$. 由(2.6)式计算出$f_1(y)$.

$$f_1(y) = \max_{0 \leq 3x_1 \leq y} 4x_1^2 = \frac{4}{9}y^2, \quad \left(x_1 = \frac{y}{3}\right)$$

此时，$x_1 = \dfrac{y}{3}$取到最大值。

再由(2.7)式计算出$f_2(y)$,

$$\begin{aligned} f_2(y) &= \max_{0\le 2x_2\le y}\left\{-x_2^2+\frac{4}{9}(y-2x_2)^2\right\} \\ &= \max_{0\le x_2\le \frac{y}{2}}\left\{\frac{1}{9}(7x_2^2-16yx_2+4y^2)\right\} \\ &= \frac{4}{9}y^2, \quad (x_2=0). \end{aligned}$$

最后再利用(2.7)式计算$f_3(9)$,

$$\begin{aligned} f_3(9) &= \max_{0\le x_3\le 9}\left\{2x_3^2+12+\frac{4}{9}(9-x_3)^2\right\} \\ &= \max_{0\le x_3\le 9}\left\{\frac{1}{9}(22x_3^2-72x_3+432)\right\} \\ &= 172, \quad (x_3=9). \end{aligned}$$

现在来分析一下，最优策略是什么。注意到在第3阶段，$x_3=9$, 和第2阶段，$x_2=0$, 那么第1阶段的x_1的取值应为

$$x_1=9-9-0=0,$$

所以最优点是$x^*=(0,0,9)^T$.

还有一类连乘积的问题

$$\begin{aligned} \max \quad & g_1(x_1)\cdot g_2(x_2)\cdot\cdots\cdot g_n(x_n), \\ s.t. \quad & a_1x_1+a_2x_2+\cdots+a_nx_n=b, \\ & x_1,x_2,\cdots,x_n\geq 0. \end{aligned}$$

也可用此方法计算。其状态转移方程为

$$\begin{aligned} f_1(y) &= \max_{a_1x_1\leq y} g_1(x_1), \\ f_k(y) &= \max_{a_kx_k\leq y} \{g_k(x_k)\cdot f_{k-1}(y-a_kx_k)\}. \end{aligned}$$

我们可以用这个方法求解下面的问题。

例4.2.2 求解非线性规划问题

$$\max \quad x_1x_2\cdots x_n, \tag{2.8}$$

$$s.t. \quad x_1+x_2+\cdots+x_n=c, \tag{2.9}$$

$$x_1,x_2,\cdots,x_n\geq 0. \tag{2.10}$$

问题(2.8)–(2.10)是一个典型的几何规划问题。让你证明：当n个数的和为一常数，其乘积在什么时候达到最大。

（结论是：n个数均相等时乘积达到最大。）请你用我们讲的方法求出该问题的解，在这里就不给出详细的推导过程了。

3 背包问题

背包问题是一类整数规划问题，它的叙述如下：设有n件物品，且第i件物品的重量为a_i, 其价值为c_i, 而背包能承受的总重量是b，问应如何选择这些物品，使背包中所装物品的价值最大？

设第i件物品装x_i件，则得到如下数学表达式：

$$\begin{aligned} \max \quad & c_1x_1 + c_2x_2 + \cdots + c_nx_n, \\ s.t. \quad & a_1x_1 + a_2x_2 + \cdots + a_nx_n \le b, \\ & x_1, x_2, \cdots, x_n \ge 0, \ \text{且}\ x_i\ \text{取整数}. \end{aligned}$$

这是一个整数规划问题，当然可以用整数规划的方法求解。我们还是建立状态转移方程，设$f_k(y)$表示包中只能装k件物品，且总重量不能超过y的最大价值。这样我们得到

$$f_1(y) = \max_{\substack{0\le a_1x_1\le y\\ x_1\text{取整数}}} c_1x_1 = \max_{0\le x_1\le\lceil\frac{y}{a_1}\rceil\text{且取整数}} c_1x_1 = c_1\left\lceil\frac{y}{a_1}\right\rceil, \quad (3.1)$$

$$f_k(y) = \max_{0\le x_k\le\lceil\frac{y}{a_k}\rceil\text{且取整数}} \{c_kx_k + f_{k-1}(y-a_kx_k)\}, \quad (3.2)$$

其中$\lceil\cdot\rceil$表示不超过它的最大整数。

现在我们来看一个具体的例子。设有背包问题如下

i	1	2	3
a_i	3	2	5
c_i	8	5	12

且背包的总重量为5。

上述背包问题的成数学表达式为

$$\begin{aligned} \max \quad & (8x_1 + 5x_2 + 12x_3), \\ s.t. \quad & 3x_1 + 2x_2 + 5x_3 \leq 5, \\ & x_1, x_2, x_3 \geq 0 \text{ 且取整数}. \end{aligned}$$

现在的具体问题是有3件物品可以装，背包允许装的总重量为5，也就是说我们要求的是$f_3(5)$.

我们利用状态转移方程(3.2)来计算$f_3(5)$，因此有

$$\begin{aligned} f_3(5) &= \max_{0\le x_3\le\lceil\frac{5}{5}\rceil\text{且取整数}} \{12x_3 + f_2(5-5x_3)\} \\ &= \max_{x_3=0\text{或 }1} \{12x_3 + f_2(5-5x_3)\} \\ &= \max\{0+f_2(5),\ \ 12+f_2(0)\}, \end{aligned} \tag{3.3}$$

这里$f_2(5)$和$f_2(0)$是未知的。下面我们就来计算它们。

仍然利用状态转移方程(3.2), 得到

$$\begin{aligned} f_2(5) &= \max_{0\le x_2\le\lceil\frac{5}{2}\rceil\text{且取整数}} \{5x_2 + f_1(5-2x_2)\} \\ &= \max_{x_2=0,\ 1,\ 2} \{5x_2 + f_1(5-2x_2)\} \\ &= \max\{0+f_1(5),\ \ 5+f_1(3),\ \ 10+f_1(1)\}, \end{aligned} \tag{3.4}$$

这里$f_1(5)$, $f_1(3)$ 和$f_1(1)$仍是未知的。再计算$f_2(0)$, 由(3.2)式得到

$$\begin{aligned} f_2(0) &= \max_{0\le x_2\le\lceil\frac{0}{2}\rceil\text{且取整数}} \{5x_2 + f_1(0-2x_2)\} \\ &= \max_{x_2=0}\{5x_2 + f_1(0-2x_2)\} \\ &= 0 + f_1(0). \end{aligned} \tag{3.5}$$

最后，我们计算$f_1(5), f_1(3), f_1(1)$和$f_1(0)$，由(3.1)式，得到

$$f_1(5) = 8\left\lceil\frac{5}{3}\right\rceil = 8,\ f_1(3) = 8\left\lceil\frac{3}{3}\right\rceil = 8, \tag{3.6}$$

$$f_1(1) = 8\left\lceil\frac{1}{3}\right\rceil = 0,\ f_1(0) = 0. \tag{3.7}$$

将式(3.6)和式(3.7)代回(3.5)和(3.4)式，得到

$$f_2(0) = 0 + f_1(0) = 0, \tag{3.8}$$

$$\begin{aligned} f_2(5) &= \max\{0 + f_1(5),\ \underline{5 + f_1(3)},\ 10 + f_1(1)\} \\ &= \max\{0 + 8,\ \underline{5 + 8},\ 10 + 0\} = 13, \end{aligned} \tag{3.9}$$

再将(3.8)和(3.9)式代入(3.3)式，得到

$$\begin{aligned} f_3(5) &= \max\{\underline{0+f_2(5)},\ \ 12+f_2(0)\} \\ &= \max\{\underline{0+13},\ \ 12+0\} = 13. \end{aligned} \tag{3.10}$$

下面找出问题的解。由(3.10)式知，当$x_3=0$时，$f_3(\cdot)$达到最大值。在这种情况下，由(3.9)式知，当$x_2=1$时，$f_2(\cdot)$达到最大值。再由(3.6)式知，$f_1(3)$是$x_1=1$, 所以最优解为$x^*=(1,1,0)$，即第一件物品装1件，第二件物品装1件，而第三件物品不装，此时总重量为5，最优的目标值为13.

4 多阶段生产安排问题

某种材料可用于两种方式生产，用后除产生效益外，还有一部分回收，表4.4.1表示的是生产方式、效益及回收之间的关系

表 4.4.1 生产、效益与回收

生产方式	I	II
效益函数	$g_1(x)$	$g_2(x)$
回收函数	a_1x	a_2x

其中$a_1 \in (0,1), a_2 \in (0,1)$.

问题：若有材料$\bar{x}$个单位，计划进行n个阶段的生产，如何投入材料，使总效益达到最大？

设$f_k(x)$表示投入x个单位的材料，还剩k个阶段的最大效益，因此得到如下的递推方程

$$f_1(x) = \max_{0\le y\le x}\{g_1(y)+g_2(x-y)\}, \quad (4.1)$$

$$f_k(x) = \max_{0\le y\le x}\{g_1(y)+g_2(x-y)+f_{k-1}(a_1y+a_2(x-y))\} \quad (4.2)$$

这样问题简化为求$f_n(\bar{x})$.

我们看一个具体的问题。假设现有材料$\bar{x}=100$个单位，计划进行三个阶段（$n=3$）的生产，其效益函数分别是$g_1(x)=0.6x$, $g_2(x)=0.5x$, 回收率分别为$a_1=0.1$, $a_2=0.4$, 问：应如何安排生产方式?

由(4.1)式和已知条件知，

$$
\begin{aligned}
f_1(x) &= \max_{0\le y\le x}\{g_1(y)+g_2(x-y)\}\\
&= \max_{0\le y\le x}\{0.6y+0.5(x-y)\}\\
&= \max_{0\le y\le x}\{0.5x+0.1y\}=0.6x \quad (y=x),\\
f_2(x) &= \max_{0\le y\le x}\{g_1(y)+g_2(x-y)+f_1(a_1y+a_2(x-y))\}\\
&= \max_{0\le y\le x}\{0.6y+0.5(x-y)+0.6(0.1y+0.4(x-y))\}\\
&= \max_{0\le y\le x}\{0.74x-0.08y\}=0.74x \quad (y=0),\\
f_3(x) &= \max_{0\le y\le x}\{g_1(y)+g_2(x-y)+f_2(a_1y+a_2(x-y))\}\\
&= \max_{0\le y\le x}\{0.6y+0.5(x-y)+0.74(0.1y+0.4(x-y))\}\\
&= \max_{0\le y\le x}\{0.796x-0.122y\}=0.796x \quad (y=0).
\end{aligned}
$$

当$\bar{x}=100$时，有$f_3(100)=79.6$.

下面我们来看一下生产方式。

- 第一年（$k=3$，即还剩三年）全部用方式II生产（$y=0$）；
- 第二年（$k=2$，即还剩二年）全部用方式II生产（$y=0$）；
- 第三年（$k=1$，即还剩一年）全部用方式I生产（$y=x$）。

5 用LINGO软件求解动态规划问题

在本节介绍用LINGO软件求解动态规划问题。

5.1. 最短路问题

最短路问题是典型的动态规划问题，在§4.1 中介绍了最短路的求解方法，这里介绍用LINGO软件求解最短路问题，下面的LINGO程序是以例4.1.1为例编写的。

```
MODEL:
SETS:
! Dynamic programming illustration.
  We have a network of 7 cities.
  We want to find the shortest route
  from city A to city D. (see example 4.1.1);
! Here is our primitive set of 7 cities,
  where F( i) is the shortest path distance
  from city i to the last city;
     CITIES /A, B1, B2, C1, C2, C3, D/: F;
```

最短路问题与动态规划...
资源分配问题
背包问题
多阶段生产安排问题
用LINGO软件求解动...
访问主页
标题页
第 60 页 共 70 页
返回
全屏显示
关闭
退出

```
! The derived set ROADS lists the roads that
  exist between the cities (note: not all
  city pairs are directly linked by a road,
  and roads are assumed to be one way.);
    ROADS( CITIES, CITIES)/
     A,B1   A,B2  B1,C1  B1,C2 B1,C3  B2,C1
    B2,C2  B2,C3   C1,D  C2,D  C3,D/: W;
! W(i,j) is the cost to travel from city i to j;
ENDSETS

```

```
DATA:
 ! Here are the distances that correspond
   to the above links;
     W =  2    4    3    3     1     2
          3    1    1    3     4;
ENDDATA

 ! If you are already in City D, then the
   cost to travel to City D is 0;
      F( @SIZE( CITIES)) = 0;

```

```
! The following is the classic Dynamic
  Programming recursion. In words, the
  shortest distance from City i to City D is
  the minimum over all cities j reachable
  from i of the sum of the distance from
  i to j plus the minimal distance from
  j to City D;
     @FOR( CITIES( i)| i #LT# @SIZE( CITIES):
       F(i)=@MIN(ROADS(i,j): W(i,j)+F(j)));
END
```

Lingo软件给出了各点到D点的最短路，

Variable	Value
F(A)	6.000000
F(B1)	4.000000
F(B2)	3.000000
F(C1)	1.000000
F(C2)	3.000000
F(C3)	4.000000
F(D)	0.000000

其缺点是没有直接给出相应的最优策略。

5.2. 投资分配问题

所谓投资分配问题就是一个数学规划问题

$$\max \quad \sum_{i=1}^{n} g_i(x_i), \tag{5.1}$$

$$s.t. \quad \sum_{i=1}^{n} x_i = a, \tag{5.2}$$

$$x_i \geq 0, \quad i = 1, 2, \cdots, n. \tag{5.3}$$

可以直接利用LINGO软件求解。如对于例4.2.1，其LINGO语言如下，

```
MODEL:
  SETS:
    variables/1..3/: x;
  ENDSETS

  max=4*x(1)^2-x(2)^2+2*x(3)^2+12;
  3*x(1)+2*x(2)+x(3)=9;
END
```

其计算结果为

```
Local optimal solution found at step:        3
Objective value:                      174.0000

Variable           Value        Reduced Cost
   X( 1)       0.0000000           -108.0000
   X( 2)       0.0000000           -72.00002
   X( 3)        9.000000           0.0000000
```

5.3. 背包问题

背包问题实际上就是一个整数规划问题

$$
\begin{aligned}
\max \quad & c_1x_1 + c_2x_2 + \cdots + c_nx_n, \\
s.t. \quad & a_1x_1 + a_2x_2 + \cdots + a_nx_n \le b, \\
& x_1, x_2, \cdots, x_n \ge 0, \ \text{且 } x_i \text{ 取整数}.
\end{aligned}
$$

对应于§4.3的背包问题的LINGO语言如下，

```
MODEL:
   SETS:
    variables/1..3/: a, c, x;
   ENDSETS

   DATA:
     a=3, 2, 5;
     b=5;
     c=8, 5, 12;
   ENDDATA

   MAX=@SUM(variables: c*x);
   @SUM(variables: a*x)<=b;
   @FOR(variables: @GIN(x));
END
```

相应的计算结果为

```
Global optimal solution found at step: 0
Objective value:                13.00000
Branch count:                          0

Variable        Value       Reduced Cost
   X( 1)     1.000000          -8.000000
   X( 2)     1.000000          -5.000000
   X( 3)    0.0000000          -12.00000
```