第 10 卷 第 4 期
1984 年 10 月

自 动 化 学 报
ACTA AUTOMATICA SINICA

Vol. 10, No. 4
Oct., 1984

目标规划算法的映射变换形式

傅 一 帆
(北京第三棉纺织厂)

摘 要

在微型机上实现目标规划时，突出的问题是内存容量小、运算速度慢. 本文根据正负偏差变量线性相关的特性，提出了压缩存贮的算法形式. 与其它方法比较，具有速度较快、对机器适应性强的特点.

在企事业的经营管理中，常遇到一些需要最优决策的问题. 例如在安排生产计划时，在一定资源条件下，要求利润、产值、产量等达到一定要求或最大，成本、能耗尽可能地少. 这就是多目标的优化问题. 在微型机上应用目标规划，投资少，效益高.

一、映射变换形式的基本思想

在微型机上实现目标规划，突出的问题是内存容量小. 对此有人把模型的数据放到外存，分别调入内存进行处理[1]，但速度慢，许多时间耗费在数据传输上. 例如计算京棉三厂的计划方案(40 个约束条件，24 个决策变量，6 个优先级，用八位微型机)，仅运算一次就需七个小时以上[2]，为了解决空间和时间的矛盾，笔者构造了一种算法，称为"映射变换"法，加上其它一些技巧，可使模型的规模扩大几倍到几十倍，并且和其它方法[1-3]相比，速度也有提高.

现将目标规划数学模型用矩阵形式表示，形式(一):

求一组 $\boldsymbol{x} \geqslant 0$, $\boldsymbol{d}^+ \geqslant 0$, $\boldsymbol{d}^- \geqslant 0$,

使
$$\min \boldsymbol{z} = C_a\boldsymbol{x} + C^+\boldsymbol{d}^+ + C^-\boldsymbol{d}^- \text{ (目标函数)}, \tag{1.1}$$

满足于
$$A\boldsymbol{x} - I\boldsymbol{d}^+ + I\boldsymbol{d}^- = \boldsymbol{b} \text{ (约束条件)}. \tag{1.2}$$

式中 A 为 $m \times n$ 阶约束条件矩阵； I 为 m 阶单位矩阵； $\boldsymbol{x}$ 为决策变量； $\boldsymbol{d}^+$ 和 $\boldsymbol{d}^-$ 分别为正负偏差变量，表示超过目标和未达到目标的值； $\boldsymbol{b}$ 为目标值向量，或称右边值向量； C_a, C^+, C^- 分别为 $g \times n$, $g \times m$, $g \times m$ 阶目标函数系数矩阵； $\boldsymbol{z}$ 为目标函数值向量. 形式(一)是一个有 m 个方程，n 个决策变量，g 个优先级的目标规划数学模型.

观察一下目标规划模型的约束方程可知，矩阵 A 的数据是由实际问题所决定，有相当一部分空间为矩阵 $-I$ 和 I 所占据. 单位矩阵 I 构成了初始基底，在迭代过程中所感兴趣的仅仅是基变量和它的值的对应关系，因此可以将这些变量与其值的对应关系用一定

本文于 1983 年 4 月 18 日收到.

标志“记忆”起来，从而将整个单位矩阵压缩掉．另外，在运算过程中，与这些基变量所对应的目标函数系数均为零，只要在所对应的变量进基和出基时做相应的处理，也可以将其省掉．

目标规划的一个特点是引用了正负偏差变量，而在 m 对正负偏差变量中，各 d_i^+ 与 d_i^- 在约束方程中的系数列向量是线性相关的，依据单纯形法原理，每一次迭代过程都相当于用同一个 m 阶满秩矩阵对它们左乘，经多次迭代 d_i^+ 与 d_i^- 的系数列向量仍然是线性相关的，仅相差一个负号，因此构造算法时，在每对偏差变量中最多只保留一个偏差变量的系数列向量，另一个可对它求反得到．在目标函数中，由于对应于每对偏差变量的目标函数系数列有时是无关的，所以目标函数中的系数，除了在基底中的零不保留外，其余均保留．对于目标函数的运算，如果在约束方程中偏差变量的系数列被保留下来，可用一般方法；若没保留，则需先将“代表”它的系数列取反，得出该列向量，再对目标函数做相应运算，这就是映射变换的方法，该方法可使运算量和 $2m^2 + g \times m$ 的存储量减少．

二、算 法 描 述

本节所述算法，除了映射变换形式外，还用了变量(可有正偏差变量)上、下有界的技巧．对于下界，可预先用平移坐标的方法解决，不再叙述．对于上界的处理，是在线性规划有界算法[1] 的基础上，改成目标规划的形式，为了适应微型机的特点，将一些步骤合并．

为便于描述，将模型进行变换．由 (1.1) 式

$$\min \boldsymbol{z} = C_a\boldsymbol{x} + C^+\boldsymbol{d}^+ + C^-\boldsymbol{d}^- = (C_aC^+C^-)\begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{d}^+\\ \boldsymbol{d}^-\end{bmatrix}$$

有

$$\min \boldsymbol{z} = C\boldsymbol{y}. \tag{2.1}$$

可见

$$C = (C_aC^+C^-), \quad \boldsymbol{y} = (\boldsymbol{x}\boldsymbol{d}^+\boldsymbol{d}^-)^{\mathrm{T}}.$$

由 (1.2) 式

$$A\boldsymbol{x} - I\boldsymbol{d}^+ + I\boldsymbol{d}^- = \boldsymbol{b}, \quad (A-II)\begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{d}^+\\ \boldsymbol{d}^-\end{bmatrix} = \boldsymbol{b},$$

有

$$P\boldsymbol{y} = \boldsymbol{b}. \tag{2.2}$$

可见

$$P = (A - II).$$

从而有(形式二)：

求

$$\boldsymbol{y} \geqslant \boldsymbol{0},$$

使

$$\min \boldsymbol{z} = C\boldsymbol{y},$$

1) 马仲蕃、顾基发等，运筹学讲义(上册)，清华大学经济管理工程系，1980 年，p27．

满足

$$Py = b.$$

经过对下界的预处理，该算法将变量划分为四个子集合.

定义. B_v 为基底变量下标集合；N_1 为非基底的零变量下标集合；N_2 为变量是上限极点时的下标集合；N_3 为偏差变量映射下标集合；V 为全集，即为所有变量下标集合.

$$B_V = \{J_i | y_i = b_i,\ 1 \leqslant j \leqslant n + 2m,\ 1 \leqslant i \leqslant m\},$$

J_i 是第 i 个约束条件对应的基底变量下标.

$N_1 = \{j | y_j = 0,\ j \bar{\in} B_v$，若 j 是偏差变量下标，则对应该目标的另一偏差变量下标 $j' \bar{\in} B_v,\ 1 \leqslant j \leqslant 2m + n\}$,

$N_2 = \{j | y_j = y_{uj},\ y_{uj}$ 为经过预处理后的变量上限值，$1 \leqslant j \leqslant m + n\}$,

$N_3 = \{j | (y_j = d_i^+ \wedge d_i^+ = 0 \wedge i^- \in \{N_1 \cup B_v\})$ 或 $(y_j = d_i^- \wedge d_i^- = 0 \wedge i^+ \in \{N_1 \cup N_2 \cup B_v\}),\ n < j \leqslant n + 2m,\ 1 \leqslant i \leqslant m\}$.

定义. $b_i = b_{oi} - \sum_{j \in N_2} P(i, j) y_{uj},\quad (i = 1, 2, \cdots, m).$

$$z_i = z_{oi} - \sum_{j \in N_2} C(i, j) y_{uj},\quad (i = 1, 2, \cdots, g).$$

b_{oi} 为不考虑上限时的基底变量值；b_i 为考虑上限后的基底变量值；z_{oi} 是未考虑上限时的第 i 级目标函数值；z_i 是考虑了上限后的第 i 级目标函数值. N_1, N_2, N_3 和 B_v 是 V 的划分.

迭代开始前，变量是按其下标顺序排列的，迭代开始后变量和系数列向量的位置是移动的. 因此，本文下面所谈的列下标 j, k, J_l 等均指变量及其系数列向量的序号与实际位置不一定一一对应. N_3 中的元素为偏差变量的下标，该变量的约束条件系数列不占存储单元，其目标函数系数是由 $\{\sim N_3\}$ 中与其相关的偏差变量系数经过映射变换得到的.

运算前初始状态如下：对于 $x_j, j \in N_1, j = 1, 2 \cdots n$；对于 $d_i^+, y_j = d_i^+, j \in N_3, i = 1, 2 \cdots m, j = n + 1, \cdots, n + m$；对于 $d_i^-, y_j = d_i^-, j \in B_v, i = 1, 2 \cdots m, j = m + n + 1, \cdots, 2m + n$；$N_2 = \phi$；$b_i = b_{oi}, z_i = z_{oi}$，$i$ 为 $\boldsymbol{b}$ 或 $\boldsymbol{z}$ 向量中的第 i 个分量.

计算步骤如下：

步骤 1°，在第 r 级优化时 $(1 \leqslant r \leqslant g)$，分 $r = 1$ 或 $r > 1$ 两种情况：当 $r = 1$ 时，对所有的 $j \in \{N_1 \cup N_3\}$，有 $C(r, j) \leqslant 0$；对所有的 $j \in N_2$，有 $C(r, j) \geqslant 0$ 时进入下一级优化，若这时第一级也是最后一级，则停止迭代转出. 当 $r > 1$ 时，对所有的 $j \in \{N_1 \cup N_3\}$，有 $C(r, j) \leqslant 0$，或虽有 $C(r, j) > 0$，必存在某个 $C(s, j) \neq 0$，$(s = 1, \cdots, r - 1)$；对所有的 $j \in N_2$，有 $C(r, j) \geqslant 0$，或虽有 $C(r, j) < 0$，必存在某个 $C(s, j) \neq 0$，$(s = 1, \cdots, r - 1)$. 则从步骤 1° 进入下一级优化，若已经是最后一级，则停止迭代转出.

步骤 2°，求 $k = \max \{j | C(r, j) > 0, j \in \{N_1 \cup N_3\} \wedge ((\forall s) C(s, j) = 0)$，或 $C(r, j) < 0, j \in N_2 \wedge ((\forall s) C(s, j) = 0), (s = 1, 2 \cdots r - 1)\}$.

步骤 3°，若 $k \in N_3$，则找出对应于同一目标的另一偏差变量下标 k'，使 $\boldsymbol{p}_k = -\boldsymbol{p}_{k'}$，同时交换 $\boldsymbol{c}_{k'}$ 和 $\boldsymbol{c}_k$ 列的位置，$\bar{N}_1 = (N_1 \cup \{K\}) / \{k'\}$，$\bar{N}_3 = (N_3 / \{k\}) \cup \{k'\}$.

步骤 4°，做

$$H_i = \begin{cases} P(i,k), & k \in \{\sim N_2\}, \\ -P(i,k), & k \in N_2, \end{cases} \quad (i = 1, 2 \cdots m).$$

$$\theta_l = \min\left\{\frac{b_i}{H_i}, \text{在 } H_i > 0 \text{ 时,或} \frac{b_i - y_{uji}}{H_i}, \text{在 } H_i < 0 \text{ 时} \middle| i = 1, 2, \cdots, m\right\}.$$

步骤 5°，若 θ_l 和 y_{uk} 都为无穷大，则解无界，停止迭代.

步骤 6°，若 $\theta_l < y_{uk}$，则转步骤 7°. 否则，如果 $k \in N_2$，则 $\bar{N}_1 = N_1 \cup \{k\}$. $\bar{N}_2 = N_2/\{k\}$，转步骤 9°. 否则，$\bar{N}_1 = N_1/\{k\}$，$\bar{N}_2 = N_2 \cup \{k\}$. 计算，$\bar{b}_i \Leftarrow b_{oi} - P(i,k)y_{uk}$，$(i = 1, 2, \cdots, m)$，$\bar{z}_i \Leftarrow z_{oi} - C(i,k)y_{uk}$，$(i = 1, 2, \cdots, g)$ 转步骤 1°.

步骤 7°，旋转运算

$i = l$，　$\bar{P}(l,j) \Leftarrow \dfrac{P(l,j)}{P(l,k)}$，$(j \neq J_l, J_l$ 为出基变量下标，l 为 θ_l 的行号)

$$\bar{b}_{ol} \Leftarrow \frac{b_{ol}}{P(l,k)}, \quad \bar{P}(l,J_l) \Leftarrow \frac{1}{P(l,k)};$$

$$i \neq l, \quad \bar{P}(i,j) \Leftarrow P(i,j) - \frac{P(l,j)}{P(l,k)} P(i,k), \quad (j \neq J_l)$$

$$\bar{b}_{oi} \Leftarrow b_{oi} - \frac{b_{ol}}{P(l,k)} P(i,k), \quad \bar{P}(i,J_l) \Leftarrow -\frac{P(i,k)}{P(l,k)};$$

$$\bar{C}(i,j) \Leftarrow C(i,j) - \frac{P(l,j)}{P(l,k)} C(i,k), \quad (j \neq J_l).$$

若 j 是偏差变量下标，则找出对应于同一目标的另一偏差变量下标 $j'(j' \in N_3)$，做映射变换，$\bar{C}(i,j') \Leftarrow C(i,j') + \dfrac{P(l,j)}{P(l,k)} C(i,k)$，$(j \neq J_l)$ $\bar{C}(i,J_l) \Leftarrow -\dfrac{C(i,k)}{P(l,k)}$，若 J_l 是偏差变量，则同样做映射变换，$\bar{C}(i,J_l') \Leftarrow C(i,J_l') + \dfrac{C(i,k)}{P(l,k)}$，$(J_l' \in N_3)$

$$\bar{z}_{oi} \Leftarrow z_{oi} - \frac{b_{ol}}{P(l,k)} C(i,k).$$

以上 $1 \leqslant j \leqslant 2m + n$，且 $j \bar{\in} N_3$.

步骤 8°，若 $k \in \{\sim N_2\}$，且原 $P(l,k) > 0$，则 $\bar{N}_1 = (N_1/\{k\}) \cup \{J_l\}$，$\bar{B}_v = (B_v \cup \{k\})/\{J_l\}$；

若 $k \in \{\sim N_2\}$，且原 $P(l,k) < 0$，则 $\bar{N}_1 = N_1/\{k\}$，$\bar{N}_2 = N_2 \cup \{J_l\}$，$\bar{B}_v = (B_v \cup \{k\})/\{J_l\}$；

若 $k \in N_2$，且原 $P(l,k) > 0$，则 $\bar{N}_2 = (N_2/\{k\}) \cup \{J_l\}$，$\bar{B}_v = (B_v/\{J_l\}) \cup \{k\}$；

若 $k \in N_2$，且原 $P(l,k) < 0$，则 $\bar{N}_1 = N_1 \cup \{J_l\}$，$\bar{N}_2 = N_2/\{k\}$，$\bar{B}_v = (B_v/\{J_l\}) \cup \{k\}$.

步骤 9°，计算 $\bar{b}_i \Leftarrow \bar{b}_{oi} - \sum_{j \in N_2} \bar{P}(i,j)y_{uj}$，$(i = 1, 2, \cdots, m)$

$$\bar{Z}_i \Leftarrow \bar{Z}_{oi} - \sum_{j \in N_2} \bar{C}(i,j)y_{uj}, \quad (i = 1, 2, \cdots, g).$$

转步骤 1°.

三、映射变换形式的迭代例子

纺织厂计划用 10 台细纱机生产甲乙两种纱，生产甲纱，每台每班可创产值 100 元，获利润 40 元；生产乙纱每台每班可创产值 60 元，获利润 48 元，根据设备情况及市场需求，按各项指标的优先顺序依次提出如下经营目标：

1）甲纱最多不得超过 5 台车，乙纱最多不得超过 6 台车，两种纱总共不得超过 10 台车；2）10 台车每班产值不小于 750 元；3）要求这 10 台车每班利润多于 420 元，并且越多越好。

这是确定甲乙两种纱的最佳开台数问题。由于开台数是指一个生产阶段的平均数，所以允许不是整数。现设 y_1 为甲纱开台数，y_2 为乙纱开台数，建立数学模型如下：

$$\min z_1 = y_3,$$

$$\min z_2 = y_7,$$

$$\min z_3 = -y_5 + y_8.$$

满足于

$$\begin{aligned} y_1 + y_2 - y_3 + y_6 &= 10, \\ 100y_1 + 60y_2 - y_4 + y_7 &= 750, \\ 40y_1 + 48y_2 - y_5 + y_8 &= 420. \end{aligned}$$

$$0 \leqslant y_1 \leqslant 5$$

$$0 \leqslant y_2 \leqslant 6, \qquad y_j \geqslant 0, \qquad (j = 3, 4, \cdots, 8).$$

y_3, y_4, y_5 为正偏差变量，y_6, y_7, y_8 为负偏差变量，运算前的初始形式见表 1.

表 1

	b	b_0	y_1	y_2	y_3	y_4	y_5
B_v 集合 y_6	10	10	1	1			
B_v 集合 y_7	750	750	100	60			
B_v 集合 y_8	420	420	40	48			
	z	z_0					
z_1	0	0	0	0	−1	0	0
z_2	750	750	100	60	0	−1	0
z_3	420	420	40	48	0	0	0

P 矩阵；C 矩阵；下标属于 N_1 或 N_2 集合的区域（含整个 P 矩阵）；N_3 集合

在矩阵 P 和 C 中，对应于 y_1, y_2 的变量的表格区为 N_1 或 N_2 集合，C 矩阵中对应 y_3, y_4, y_5 列的为 N_3 集合，对应 y_6—y_8 的为 B_v 集合。z_0, C 的值是相当于在一般的单纯形算法中，将所有目标函数系数取反，再从初始基底开始，以该基底变量为主元，逐个反复做步骤 7°（只含对 C 阵和 z_0 的运算，出基运算除外）而形成的。$N_1 = \{1, 2\}$, $N_2 = \phi$, $N_3 = \{3, 4, 5\}$, $B_v = \{6, 7, 8\}$, 令 $b \Leftarrow b_0$, $z \Leftarrow z_0$。

迭代开始，查 C 矩阵第一级判别数（$r=1$，即第一行），满足步骤 1° 进级条件进入第二级 $r=2$，得 $k=1, l=2, \theta_2=7.5$. 由于 $\theta_2>y_1=5$，所以做步骤 6° 后转步骤 1°. $N_1=\{2\}, N_2=\{1\}, N_3=\{3, 4, 5\}, B_v=\{6, 7, 8\}$，将新得到的 $\bar{\boldsymbol{b}}=(5, 250, 220)^T$，$\boldsymbol{z}=(0, 250, 220)^T$ 填入表 1 的 $\boldsymbol{b}$ 和 $\boldsymbol{z}$ 列中.

又返回步骤 1° 开始，$k=2, l=2, \theta_2=25/6$. 由于 $J_2=7$，y_7 出基且与 y_4 是映射对，做旋转及映射变换后，$N_1=\{7\}, N_2=\{1\}, N_3=\{3, 4, 5\}, B_v=\{6, 2, 8\}$. 见表 2.

表　2

	$\boldsymbol{b}$	$\boldsymbol{b}_0$	y_1	y_7	y_3	y_4	y_5
y_6	$\frac{5}{6}$	−2.5	$-\frac{2}{3}$	$-\frac{1}{60}$			
y_2	$\frac{25}{6}$	12.5	$\frac{5}{3}$	$\frac{1}{60}$			
y_8	20	−180	−40	−0.8			

	$\boldsymbol{z}$	$\boldsymbol{z}_0$					
	0	0	0	0	−1	0	0
	0	0	0	−1	0	0	0
	20	−180	−40	−0.8	0	0.8	0

此时第二级判别数满足进级条件，进入第三级，以后照上述形式反复迭代直至找到满意解. 这时 $N_1=\{6\}, N_2=\{2\}, N_3=\{3, 7, 8\}, B_v=\{5, 1, 4\}, \boldsymbol{b}=(28, 4, 10)^T$，得满意解为 $y_2=y_{u_2}=6$，$y_3=28, y_1=4, y_4=10$，其余为零. 代入原方程，得实际目标值为，总开台数 10 台，班产值 760 元，利润 448 元.

四、程序功能及应用简况

用 BASIC 语言编制了通用程序，采用程序覆盖技术，总共 20 多 k 字节，用户可在内存仅 20 KB 的 MCZ-1/50 微型机上，用双精度变量（占 8 字节），在充分利用上下界约束功能情况下，最大可解约束条件 230 个，决策变量 110 个，优先顺序 5 个，相当于一般方法的 235×571 规模的矩阵. 若不利用上下界约束功能，也可解 45×115 含 34 个决策变量的问题. 计算京棉三厂 1982 年三季度计划方案，约束条件 40 个，决策变量 24 个，优先级 6 个，计算时间仅 9 分钟，比一些解法[2]快 50 倍.

在编制京棉三厂 1982 年三季度计划时，对用计算机和人工编制的方案做了对比. 用计算机优选的方案，可比人工增加产值 74 万元，增加利润 24 万元，其它指标也有改善. 一般来讲，所得利润可比人工编制的计划方案增加 2—4%，条件越复杂，各种因素越活，计算机的效果就越明显.

笔者曾就有关问题请教过中国人民大学的魏权龄同志，北京钢铁学院马正午同志；此稿修改过程中得到孙一康教授的指导帮助，一一致谢. 京棉三厂何绍玉、田芳文、范鸣同志提供了三季度计划的数学模型.

参 考 文 献

[1] 陈三智，目标规划法，微计算机应用，1982 年第二期，p. 20.
[2] 陈景艳、张仲义，目标规划及其在微型计算机上的实现，铁道学报，1982 年第 3 期，p. 92.
[3] 玄光男等，目標計画プログラム，マイコソ，1981 年 3 月.

THE MAP FORM OF GOAL PROGRAMMING ALGORITHM

FU YIFAN
(Beijing No. 3 Cotton Mill)

ABSTRACT

To solve the problem of limited memory and low speed of operation of goal programming on a microcomputer, an algorithm form of compressed memory capacity is presented in accordance with the linear dependency of positive and negative deviation variables. Compared with other current algorithm forms, this method has been proved to give a higher operation speed and better adaptability.