文章编号:1007-6492(1999)03-0071-02

一类整数线性规划的算法

赵 升

(郑州工业大学工商学院,河南 郑州 450002)

摘 要:提出了一类常见的整数线性规划的新算法,该算法不是沿袭求解线性规划的传统思路,从可行域的边缘整数点上寻找最优解,而是根据各变量对目标的贡献大小确定出分配变量,经有限次分配后可获得最优解.该算法计算量较小,计算效率高,且在有限步内可获得最优解.与目前的分枝定界法、割平面法相比,具有一定的优越性.

关键词:整数线性规划;算法;分配顺序

中图分类号:O 221 **文献标识码**:A

0 引言

求解整数线性规划目前仍沿袭求解线性规划的传统方法即从可行域极点上去找最优解的做法不很凑效,计算量很大,并且在变量维数增加时,计算量以指数速度增加.本文提出的算法从约束条件入手,按照各变量对目标的贡献来确定变量的分配顺序,经过若干次分配之后,即可获得问题的最优解.

1 整数线性规划模型

本文讨论如下类型的整数线性规划(ILP):

$$\begin{cases} \max S = \sum_{i=1}^{n} c_i x_i ; \\ \sum_{j=1}^{n} a_{ij} x_j \leqslant b_i (b_i > 0), (i = 1, 2, \cdots, m) ; \\ x_i \geqslant 0, \text{且为整数} . \end{cases}$$

其中:$c_i \geqslant 0, a_{ij} \geqslant 0$,且均为有理数.

上述模型中,可将 x_i 理解为产品的数量;b_i 为资源;a_{ij} 为消耗系数即单位产品所消耗的资源数量;c_{ij} 为目标系数即单位产品所带来的目标值.

2 算法

为求解上述的 ILP 问题,首先研究在只有一个约束条件下,变量 x_i 的分配.设约束条件 l 为:

$$a_1 x_1 + a_2 x_2 + \cdots + a_n x_n \leqslant b,$$

考虑任意两个不同的变量 x_{i_1}, x_{i_2},把资源 b 分别全部分配给 x_{i_1}, x_{i_2},产生的目标值为 S_{i_1}, S_{i_2}.将 b 全部分给 x_{i_1},则有 $x_{i_1}^* = (b/a_{i_1})$,由于取整的关系,资源可能剩有不足 a_{i_1} 的数量,这部分剩余资源可在除 x_{i_1} 之外的其他变量间分配,记其产生的目标值为 $S_{\Delta i_1}$,于是

$$S_{i_1} = c_{i_1} x_{i_1}^* + S_{\Delta i_1}, S_{i_2} = c_{i_2} x_{i_2}^* + S_{\Delta i_2},$$

将两个目标值相比较:

$$S_{i_1} - S_{i_2} = c_{i_1} x_{i_1}^* - c_{i_2} x_{i_2}^* + S_{\Delta i_1} - S_{\Delta i_2}, \quad (1)$$

若 $S_{i_1} > S_{i_2}$,则资源安排给 x_{i_1} 优于给 x_{i_2};

若 $S_{i_1} < S_{i_2}$,则将资源分配给 x_{i_1} 劣于给 x_{i_2};

若 $S_{i_1} = S_{i_2}$,则分配给 x_{i_1} 或 x_{i_2},效果相同.

通过式(1)可将 n 个变量的优劣顺序排出,首位的即为应当分配的变量.由于分配优劣顺序与资源约束有关,所以一次只分一个单位,然后再确定分配变量,再分配一个单位,这样可保证每次分配都是最优分配.重复这种分配过程,直至将资源分配至不能再分配为止,最优解也到此求出.

在计算式(1)时,当 a_{i_1} 比其它变量系数大得多时,不足 a_{i_1} 的资源可能够其他多个变量分配,分配方案可有数个,此时应以最优目标值做为 $S_{\Delta i_1}, S_{\Delta i_2}$ 同理.

在分配排序中,$x_i^* = (b/a_i) = 0$ 者,可排除.

作为式(1)的一个充分条件,如果

收稿日期:1999-01-08;**修订日期**:1999-03-05

作者简介:赵 升(1964-),男,郑州工业大学讲师,硕士,主要从事管理科学方面的教学和研究.

$$c_{i_1}x_{i_1}^* \geqslant c_{i_2}x_2^*\text{，或 } x_{i_1}^*/x_{i_2}^* \geqslant c_{i_2}/c_{i_1}, \tag{2}$$

$$S_{\Delta i_1} \geqslant S_{\Delta i_2}. \tag{3}$$

则有 $S_{i_1} \geqslant S_{i_2}$，即式(2)，(3) 成立，则 x_{i_1} 先于 x_{i_2}. 若式(2)，(3)不等号反向，则 $S_{i_1} \leqslant S_{i_2}$，此时 x_{i_2} 先于 x_{i_1}；若式(2)，(3)不等式方向不一致，需用式(1)判断 x_{i_1}，x_{i_2} 的顺序. 为方便计算，设计出表1：

表1 排序计算表

x_i^*	c_i	1	2	3	…	n	S_{Δ_i}	1	2	3	…	n
x_1^*	c_1	1	x_1^*/x_2^*	x_1^*/x_3^*	…	x_1^*/x_n^*	S_{Δ_1}	0	$S_{\Delta_1}-S_{\Delta_2}$	$S_{\Delta_1}-S_{\Delta_3}$	…	$S_{\Delta_1}-S_{\Delta_n}$
x_2^*	c_2	c_2/c_1	1	x_2^*/x_3^*	…	⋮	S_{Δ_2}		0	$S_{\Delta_2}-S_{\Delta_3}$	…	⋮
⋮	⋮	c_3/c_1	c_3/c_2	1	⋮	⋮	⋮				⋱	⋮
x_n^*	c_n	c_n/c_1	…	…	…	1	S_{Δ_n}					0

表中 c_i/c_j，x_i^*/x_j^*，$S_{\Delta_i}-S_{\Delta_j}$ 行列位置与脚标的上下或左右相同. 表中的分式不要化简，以便在应用式(2)，(3)判断不出时，应用式(1)比较方便.

表的用途是确定分配变量，根据约束条件 l 可算出表中的数据. 检查 x_1^*/x_2^* 与"1"对角线对称的 c_2/c_1，再结合 $S_{\Delta_1}-S_{\Delta_2}$，应用式(2)，(3)或(1)即可确定 x_1，x_2 的顺序. 在 $S_{\Delta_i}-S_{\Delta_j}=0$ 时，用式(2)即可判断. 在计算时，x_i^* 为0的行可划去.

在一个约束条件下的求解问题解决之后，容易解决在 m 个约束条件 $l_1, l_2, \cdots, l_m$ 下的求解问题. 设 l_i 条件下分配变量为 x_{J_i}，为讨论方便，先就 l_1、l_2 来说明. 按 l_1 应分配 x_{J_1}，按 l_2 应分配 x_{J_2}，当 x_{J_1} 与 x_{J_2} 相同时，在 l_1，l_2 中同时分配 x_{J_1} 或 x_{J_2}；当 x_{J_1} 与 x_{J_2} 不相同时，由于变量要同时满足 l_1，l_2，此时应在 l_1，l_2 中同时分配 x_{J_1} 和 x_{J_2}. 类推到 $l_1, \cdots, l_m$ 条件下，应在 $l_1, \cdots, l_m$ 中同时分配 $x_{J_1}, \cdots, x_{J_m}$，其中相同的变量 x_{J_i} 不能重复出现. 重复这种分配过程，直至求出最优解，最优解就等于各变量在各次分配中分配数量之和.

3 结束语

本文提出的该类型的算法，经一些例子验证，与其他解法结果相同，这种算法在目前的国内外文献中鲜见报道. 该方法的计算量主要在变量的分配排序上，计算量随维数增加的速度与排序随个数增加的速度大致相当，为维数的平方速度. 如能考虑增加每次分配的单位数，则可以加速分配过程，进一步提高运算效率. 该法与分枝定界、割平面等法相比，具有明显的优越性，可作为该类整数线性规划的有效方法.

参考文献：

[1] 赵 升. 对分配问题求解方法的改进[J]. 郑州工业大学学报，1998，19(3)：92－96.

[2] 马湘铃，赵 升. 现代管理数学方法[M]. 西安：陕西人民出版社，1997. 285－295.

[3] 王永县. 运筹学[M]. 北京：清华大学出版社，1993. 71－88.

The Algorithm for a Kind of Integer Linear Prgramming

ZHAO Sheng

(School of Business Administration, Zhengzhou University of Technology, Zhengzhou 450002, China)

Abstract: This paper proposes an algorithm for a common kind of integer programming Unlike the traditional method of linear programming which searches for optimal solution among extreme points of feasible region, the algorithm is to find optimal solution through variable allocation based on contributions by variables to objectives. The algorithm, with less computation, is able to find solution within limited computing steps, and is obviously superior to the existing methods, such as branch and bound method, cut－plane method.

Key words: integer programming; algorithm; allocation sequence