求解整数线性规划的一种高效隐数搜寻

高培旺
GAO Pei-wang

广西财经学院 数学与统计系,南宁 530003
Department of Mathematics and Statistics,Guangxi University of Finance and Economics,Nanning 530003,China
E-mail:pwgao@yahoo.cn

GAO Pei-wang.Efficient implicit enumerative search for solution to integer linear programs.Computer Engineering and Applications,2009,45(26):24-26.

Abstract: This paper presents a new implicit enumerative algorithm for integer linear programs.First of all,based on the optimality of the linear programming relaxation problem,a linear transformation of the optimal nonbasic variables into new variables is introduced so that the new variables have fewer intervals.Then,an efficient implicit enumerative search is done on the nonbasic variables and new ones on the objective function hyperplane.The algorithm is of interests in theory.

Key words: linear programming;integer programming;linear transformation;implicit enumerative algorithm

摘 要:提出了一种求解整数线性规划的新的隐数算法。首先,该算法引入了一组线性变换,将线性松弛问题的最优非基变量变换到一组新变量,使新变量有更小的取值范围。然后,在目标函数超平面上对非基变量和新变量进行隐数计算,从而大大提高了隐数搜寻的效率。

关键词:线性规划;整数规划;线性变换;隐数算法

DOI:10.3778/j.issn.1002-8331.2009.26.007 **文章编号:**1002-8331(2009)26-0024-03 **文献标识码:**A **中图分类号:**O221.4

1 引言

整数线性规划在计算机、控制、经济管理等许多领域有着重要应用。目前,求解整数线性规划最常用的算法是各种切割和分支方法[1-4]。然而,这些算法需要求解一系列由切割或分支所产生的子问题。

如果整数线性规划的解接近相应的线性规划松弛问题的解,在整数值目标函数超平面上搜寻整数规划的解也许更加容易。基于这种思路,Joseph 等[5]开发了一种启发式算法,Gao[6]提出了一种隐数算法。然而,这些方法在计算凸多面体的顶点时也许需要复杂的枢轴计算。

提出了一种在目标函数超平面上求解整数线性规划问题的新的隐数算法。显然,隐数搜寻的效率主要取决于变量的取值范围。为此,该算法引入了一组线性变换,将线性松弛问题的最优非基变量变换到一组新变量,使新变量有更小的取值范围。然后,在目标函数超平面上对非基变量和新变量进行隐数计算,在隐数搜寻过程中,该算法通过一种迭代计算可以确定自由变量更好的上、下界,从而大大提高了隐数搜寻的效率。

2 一组线性变换的构造

考虑如下形式的纯整数线性规划问题:

$$(\text{ILP})\max\ f=\boldsymbol{c}^{\mathrm{T}}x$$
$$\text{s.t.}\ \boldsymbol{A}x\leqslant\boldsymbol{b}$$
$$x\geqslant 0,\text{整数}$$

这里,$\boldsymbol{A}=(a_{ij})\in Z^{m\times n}$,$\boldsymbol{b}\in Z^m$,$\boldsymbol{c}\in Z^n$。与 ILP 相应的线性规划松弛问题,用 RILP 表示。

假设通过应用单纯形算法[7],获得了 RILP 的一个基本最优解:$x_{B^*}=\boldsymbol{b}^*$,$x_{N^*}=0$,相应的最优值为 f^*,这里 x_{B^*} 和 x_{N^*} 分别是最优基变量和非基变量。令 $\bar{\boldsymbol{c}}_{N^*}$ 表示与非基变量对应的缩减费用(the reduced costs),则目标函数和约束条件可表示为

$$f=f^*-\bar{\boldsymbol{c}}_{N^*}^{\mathrm{T}}x_{N^*} \tag{1}$$

$$x_{B^*}=\boldsymbol{b}^*-\boldsymbol{B}^{*-1}\boldsymbol{N}^*x_{N^*} \tag{2}$$

$$x_{B^*}\geqslant 0,x_{N^*}\geqslant 0 \tag{3}$$

其中,$\boldsymbol{B}^*$ 和 $\boldsymbol{N}^*$ 分别是最优基矩阵和非基矩阵。假设 $\bar{\boldsymbol{c}}_{N^*}>0$。

如果 RILP 的基本最优解是整数向量,则它也是 ILP 的最优解。否则,根据假设 $\bar{\boldsymbol{c}}_{N^*}>0$,ILP 的最优值肯定小于 f^*。为此,令目标函数 f 作为一个参数从最优值 f^* 处向下变化来生成一系列目标函数超平面,用 S_f 表示。显然,如果 ILP 有可行解的话,其可行解常位于整数值目标函数超平面上。该算法就是令参数 f 从大到小取满足 $f<[f^*]$(这里,[·]表示小于或等于“·”的最大整

基金项目:广西自然科学基金(the Natural Science Foundation of Guangxi of China under Grant No.桂科自 0728260)。
作者简介:高培旺(1964-),男,博士,教授,主要研究方向:最优化理论及其应用。
收稿日期:2009-06-03 **修回日期:**2009-07-20

数）的离散整数值，然后依次在相应的目标函数超平面上搜寻求解。按照这种方式，一旦在某个目标函数超平面 S_f 中找到 ILP 的一个可行解，它也是 ILP 的一个最优解，算法终止。

在下一节，将应用隐数算法在整数值目标函数超平面 S_f 搜寻 ILP 的一个可行解。不失一般性，假设在式（1）中，$\bar{c}_{N^*1}\geqslant\bar{c}_{N^*2}\geqslant\cdots\geqslant\bar{c}_{N^*n}>0$，构造一组线性变换如下：

$$y_i=\sum_{j=i}^{n}d_jx_{N^*j},i=1,2,\cdots,n \tag{4}$$

这里，$d_j=[\frac{\bar{c}_{N^*j}}{\bar{c}_{N^*n}}](j=1,2,\cdots,n)$。显然，所有系数 $d_j(j=1,2,\cdots,n)$ 都是不小于 1 的整数且 $d_n=1$。因此，通过变换式（4），与 ILP 的可行解相应的变量 $y_j(j=1,2,\cdots,n)$ 的取值也是非负整数，其取值范围由定理 1 描述。

定理 1 令 $\alpha_j=\frac{\bar{c}_{N^*j}}{d_j}(j=1,2,\cdots,n)$，$\alpha_i^U=\max\limits_{i\leqslant j\leqslant n-m}\{\alpha_j\}(i=1,2,\cdots,n)$。假设对 f 满足 $f<f^*$ 的一个固定整数值，变量 $x_{N^*j}(j=1,2,\cdots,i-1)$ 在搜寻过程中被指定了一组整数值，则与 ILP 的可行解相应的变量 y_i 的取值满足

$$\frac{f^*-f}{\alpha_1^U}\leqslant y_1\leqslant\frac{f^*-f}{\alpha_n} \tag{5}$$

$$\frac{f^*-f-\sum\limits_{j=1}^{i-1}\bar{c}_{N^*j}x_{N^*j}}{\alpha_i^U}\leqslant y_i\leqslant\frac{f^*-f-\sum\limits_{j=1}^{i-1}\bar{c}_{N^*j}x_{N^*j}}{\alpha_n},i=2,\cdots,n \tag{6}$$

证明 注意到对 f 满足 $f<f^*$ 的一个固定整数值和在相应的目标函数超平面 S_f 上的可行解，式（1）成立，即有

$$\sum_{j=1}^{n}\alpha_j(d_jx_{N^*j})=f^*-f$$

于是，当变量 $x_{N^*j}(j=1,2,\cdots,i-1)$ 在搜寻过程中被指定了一组整数值时，自由变量 $x_{N^*j}(j=i,\cdots,n)$ 的取值满足

$$\sum_{j=i}^{n}\alpha_j(d_jx_{N^*j})=f^*-f-\sum_{j=1}^{i-1}\bar{c}_{N^*j}x_{N^*j}$$

由于 $\alpha_i^U=\max\limits_{i\leqslant j\leqslant n}\{\alpha_j\}$ 且 $\alpha_n=\max\limits_{i\leqslant j\leqslant n}\{\alpha_j\}=1$，故有

$$\alpha_n=\sum_{j=i}^{n}d_jx_{N^*j}\leqslant f^*-f-\sum_{j=1}^{i-1}\bar{c}_{N^*j}x_{N^*j}\leqslant\alpha_i^U\sum_{j=i}^{n}d_jx_{N^*j}$$

再结合线性变换式（4），即可由上式导出式（5）、式（6）。

根据定理 1，很容易得到下述推论。

推论 1 假设对 f 满足 $f<f^*$ 的一个固定整数值和变量 $x_{N^*j}(j=1,2,\cdots,i-1)$ 的一组指定整数值，在变量 y_i 的取值范围式（5）、式（6）内不存在整数，则自由变量 $x_{N^*j}(j=i,\cdots,n)$ 的任意整数取值都不能产生 ILP 的可行解。

3　基于线性变换的隐数搜寻过程

根据隐数算法的原理，当一个变量在隐数搜寻过程中被赋予了一个确定整数值时，称之为指定变量；否则，就是自由变量。接下来，S 表示指定变量的下标集合，T 表示自由变量的下标集合。

注意到在式（2）中基变量的取值由非基变量唯一确定。因此，只需对 RILP 的最优非基变量进行隐数指定。首先，将缩减费用 $\bar{\boldsymbol{c}}_{N^*}$ 按单调不增的顺序排序产生：

$$\bar{c}_{N^*j_1}\geqslant\bar{c}_{N^*j_2}\geqslant\cdots\geqslant\bar{c}_{N^*j_n}$$

这里，$j_1,j_2,\cdots,j_n$ 是 $1,2,\cdots,n$ 的一个置换，该算法将按 $x_{N^*j_1},x_{N^*j_2},\cdots,x_{N^*j_n}$ 的顺序对变量进行隐数指定。

给定 f 满足 $f<f^*$ 的一个固定整数值，通过式（1）容易获得一个自由变量的初始上界，这由下述定理描述。

定理 2 假设对 f 满足 $f<f^*$ 的一个固定整数值，变量 $x_{N^*j_s}(s=1,2,\cdots,i-1)$ 在隐数搜寻过程中被指定了一组整数值（这里，$i=1$ 意味着没有任何变量被指定）。则 ILP 的可行解中任意一个自由变量 $x_{N^*t}(t\in T)$ 的取值满足

$$x_{N^*t}\leqslant\frac{f^*-f-\sum\limits_{s=1}^{i-1}\bar{c}_{N^*j_s}x_{N^*j_s}}{\bar{c}_{N^*t}} \tag{7}$$

根据式（7），令 $x_{N^*t}^{IU}(f)=[\frac{f^*-f-\sum\limits_{s=1}^{i-1}\bar{c}_{N^*j_s}x_{N^*j_s}}{\bar{c}_{N^*t}}]$，则 $x_{N^*t}^{IU}(f)$ 就是自由变量 x_{N^*t} 的一个整数上界。显然，自由变量 $x_{N^*t}(t\in T)$ 有一个初始下界 0，用 $x_{N^*t}^{IL}(f)$ 表示。

根据推论 1，如果在变量 y_i 的取值范围式（5）、式（6）内不存在整数，隐数搜寻进程应该改变；否则，隐数搜寻继续向前进行。在这种情况下，令 $y_i^{IU}(f)$、$y_i^{IL}(f)$ 分别是变量 y_i 满足式（5）、式（6）的整数上、下界，注意到变量 y_i 的取值区间一般比变量 $x_{N^*j_n}$ 的取值区间更窄，由式（4）将 $x_{N^*j_n}$ 用 y_i 表示为：$x_{N^*j_n}=y_i-\sum\limits_{s=1}^{i-1}d_{j_s}x_{N^*j_s}$，再代入到式（1）中，得：

$$\sum_{s=i}^{i-1}\hat{c}_{N^*j_s}x_{N^*j_s}+\hat{c}_{N^*j_n}y_i=f^*-f-\sum_{s=1}^{i-1}\bar{c}_{N^*j_s}x_{N^*j_s} \tag{8}$$

这里，$\hat{c}_{N^*j_s}=\bar{c}_{N^*j_s}-\bar{c}_{N^*j_n}d_{j_s}$，$(s=1,2,\cdots,n-1)$，$\hat{c}_{N^*j_n}=\bar{c}_{N^*j_n}$。

为方便起见，令 $z_{N^*j_s}=x_{N^*j_s}$，$(s=1,2,\cdots,n-1)$，$z_{N^*j_n}=y_i$，通过下述定理，还可获得自由变量新的取值范围。

定理 3 假设对 f 满足 $f<f^*$ 的一个固定整数值，变量 $x_{N^*j_s}(s=1,2,\cdots,i-1)$ 在隐数搜寻过程中被指定了一组整数值（这里，$i=1$ 意味着没有任何变量被指定）。令 $z_{N^*j}^{IU}(f)$、$z_{N^*j}^{IL}(f)$ 分别是自由变量 z_{N^*j} 的上、下界，则对任意 $t\in T$，当 $\hat{c}_{N^*t}>0$ 时，变量 z_{N^*t} 的取值满足：

$$\frac{1}{\hat{c}_{N^*t}}(f^*-f-\sum_{s=1}^{i-1}\bar{c}_{N^*j_s}x_{N^*j_s}-\sum_{s=i,s\neq t}^{n}\hat{c}_{N^*j_s}z_{N^*j_s}^{IU}(f))\leqslant z_{N^*t}\leqslant$$
$$\frac{1}{\hat{c}_{N^*t}}(f^*-f-\sum_{s=1}^{i-1}\bar{c}_{N^*j_s}x_{N^*j_s}-\sum_{s=i,s\neq t}^{n}\hat{c}_{N^*j_s}z_{N^*j_s}^{IL}(f)) \tag{9}$$

证明 注意到对 f 满足 $f<f^*$ 的一个固定整数值，在 ILP 的可行解中所有自由变量的取值必须满足：

$$z_{N^*j}^{IL}(f)\leqslant z_{N^*j}\leqslant z_{N^*j}^{IU}(f),j\in T$$

于是对任意 $t\in T$,由式(8),有:

$$\hat{c}_{N^*_t} z_{N^*_t} + \sum_{s=i,s\neq t}^{n} \hat{c}_{N^*_{j_s}} z^{IU}_{N^*_{j_s}}(f) \geqslant f^* - f - \sum_{s=1}^{i-1} \bar{c}_{N^*_{j_s}} x_{N^*_{j_s}}$$

且

$$\hat{c}_{N^*_t} z_{N^*_t} + \sum_{s=i,s\neq t}^{n} \hat{c}_{N^*_{j_s}} z^{IL}_{N^*_{j_s}}(f) \leqslant f^* - f - \sum_{s=1}^{i-1} \bar{c}_{N^*_{j_s}} x_{N^*_{j_s}}$$

当 $\hat{c}_{N^*_t}>0$ 时,上述两个不等式即产生不等式(9)。

如果对任意 $t\in T$,变量 $z_{N^*_t}$ 的取值区间式(9)不含有任何整数,则隐数搜寻进程应该终止和改变;否则,隐数搜寻继续向前进行。在这种情况下,令 $z^{NIU}_{N^*_t}(f)$、$z^{NIL}_{N^*_t}(f)$ 分别是变量 $z_{N^*_t}$ 满足式(9)的整数上、下界,如果 $z^{NIU}_{N^*_t}(f)<z^{IU}_{N^*_t}(f)$,则 $z^{NIU}_{N^*_t}(f)$ 是变量 $z_{N^*_t}$ 的一个更好的上界;否则,置 $z^{NIU}_{N^*_t}(f)=z^{IU}_{N^*_t}(f)$。类似地,如果 $z^{NIL}_{N^*_t}(f)>z^{IL}_{N^*_t}(f)$,则 $z^{NIL}_{N^*_t}(f)$ 是变量 $z_{N^*_t}$ 的一个更好的下界;否则,置 $z^{NIL}_{N^*_t}(f)=z^{IL}_{N^*_t}(f)$。显然,一旦自由变量的界通过式(9)的迭代计算得到了改进,该算法的计算效率将大大提高。另外,注意到 $y_n=x_{N^*_{j_n}}$,当变量 y_n 在它的取值区间内被指定时,最后一个自由变量 $x_{N^*_{j_n}}$ 也随之被指定了。

根据上述算法理论,给出详细的算法步骤如下:

步骤 1 求解 RILP 得到一个基本最优解 $x_{B^*}=\boldsymbol{b}^*$,$x_{N^*}=0$ 和相应的最优目标值 f^* 后,转下一步。

步骤 2 如果 $x_{B^*}=\boldsymbol{b}^*$ 是整数向量,算法输出 ILP 的一个最优解后终止;否则,将缩减费用 $\bar{\boldsymbol{c}}_{N^*}$ 按单调不增的顺序排序产生:$\bar{c}_{N^*_{j_1}}\geqslant\bar{c}_{N^*_{j_2}}\geqslant\cdots\geqslant\bar{c}_{N^*_{j_n}}$,这里,$j_1,j_2,\cdots,j_n$ 是 $1,2,\cdots,n$ 的一个置换,引入线性变换式(4),然后转下一步。

步骤 3 令 f^{IU} 是小于 f^* 的最大整数,M 是一个给定的适当小的整数(满足 $M<[f^*]$),置 $f=f^{IU}$,然后转下一步。

步骤 4 检查 $f<M$ 是否成立。若成立,算法以 ILP 没有任何可行解而终止;否则,转下一步。

步骤 5 置 $i=0$,$s=(0,\cdots,0)\in R^n$,$S=\Phi$,$T=\{j_1,j_2,\cdots,j_n\}$,给定变量 x_{N^*} 的初始下界 $x^{IL}_{N^*}(f)=0$,然后转下一步。

步骤 6 如果 $i=n$,则将非基变量 x_{N^*} 的值代入式(2)中计算基变量 x_{B^*} 的值,然后转下一步;否则,转步骤 8。

步骤 7 如果所有变量的值都是非负整数,算法以输出 ILP 的一个最优解而终止;否则,转步骤 12。

步骤 8 对所有 $t\in T$,通过式(7)计算变量 $x_{N^*_t}$ 的初始上界 $x^{IU}_{N^*_t}(f)$,然后再通过式(5)、式(6)计算变量 y_{i+1} 的取值范围。如果变量 y_{i+1} 的取值范围内没有整数存在,则转步骤 11;否则,取变量 y_{i+1} 的整数上、下界 $y^{IU}_{i+1}(f)$ 和 $y^{IL}_{i+1}(f)$,转下一步。

步骤 9 通过式(9)的迭代计算求自由变量 $z_{N^*_t}(t\in T)$ 的新的上、下界 $z^{NIU}_{N^*_t}(f)$、$z^{NIL}_{N^*_t}(f)$。如果对所有 $t\in T$,$z^{NIL}_{N^*_t}(f)\leqslant z^{NIU}_{N^*_t}(f)$ 成立,则转下一步;否则,转步骤 11。

步骤 10 置 $i=i+1$,然后指定变量 $x_{N^*_{j_i}}$ 取它的下界 $x^{NIL}_{N^*_{j_i}}(f)$,令 $S=S\cup\{j_i\}$,$T=T\backslash\{j_i\}$,返回步骤 6。

步骤 11 检查是否有 $S=\Phi$。如果是,则置 $f=f-1$,然后返回步骤 4;否则,转下一步。

步骤 12 检查是否有 $x_{N^*_{j_i}}<x^{NIU}_{N^*_{j_i}}(f)$。如果是,则置 $x_{N^*_{j_i}}=x_{N^*_{j_i}}+1$,然后返回步骤 8;否则,令 $S=S\backslash\{j_i\}$,$T=T\cup\{j_i\}$,$i=i-1$,然后返回步骤 11。

通过执行上述算法步骤,或者获得 ILP 的一个最优解,或者当目标值下降到 M 时得到一个没有可行解的事实。

4 数值计算

为了比较该算法与经典的分支定界算法的计算效率,使用 MATLAB V6.5 对两种算法进行了编程并在 HASEE S262C 计算机上进行了求解计算。

对文献[8]在第 7 章提供的 9 个典型算例进行了计算,用于比较的主要指标有所使用的隐数步数,分支定界树的结点数,所需要的迭代(枢轴)数和所花费的计算时间,计算结果如表 1 所示。

表 1 该算法与经典的分支定界算法的比较

算例	该文的隐数算法			经典的分支定界算法		
	隐数步数	迭代数	时间/s	结点数	迭代数	时间/s
1	2	0	0.063	35	120	0.359
2	3	0	0.078	37	159	0.468
3	21	0	0.624	113	613	1.875
4	42	0	1.251	47	463	1.375
5	12	0	1.387	201	600	2.485
6	15	0	1.415	203	707	2.797
7	18	0	1.649	203	707	2.797
8	9	0	0.391	275	3 369	12.516
9	1 637	0	53.172	*	*	*

注:在算例 9 的计算中,使用经典的分支定界算法在求解了2 000 个子(根)问题后,没有获得问题的解。

从表 1 看到,该算法比经典的分支定界算法使用了更少的计算时间,且不需要任何迭代(枢轴)计算。其计算效率和稳定性明显优于经典的分支定界算法。

5 结论

该算法的特征就是通过线性变换式(4)引入了一组新变量 $y_1,y_2,\cdots,y_n$,且 y_i 的取值区间随着指标 i 的增加而变得越来越窄。这样,可以在隐数搜寻过程中根据变量 y_i 的取值情况和式(9)的迭代计算来排除对大量不可行解的搜寻,从而大大改进隐数搜寻的效率。这在上一章已经得到了一些经典算例的部分验证。

该算法还可与各种切割和分支算法结合起来使用,进一步提高求解整数线性规划的效率和性能。

参考文献:

[1] Boyd E A.Fenchel cutting planes for integer programs[J].Operations Research,1994,42:53-64.

[2] Eckstein J,Nediak M.Depth-optimized convexity cuts[J].Annals of Operations Research,2005,139:95-129.

[3] Letchford A N.Binary clutter inequalities for integer programs[J].Mathematical Programming:Ser B,2003,98:201-221.

(下转 52 页)