УДК 512.25/26 + 519.3:330.II5

О ДВУХ ПРЯМЫХ АЛГОРИТМАХ ЛИНЕЙНОГО ЦЕЛОЧИСЛЕННОГО ПРОГРАММИРОВАНИЯ

А.А.Колоколов, В.И.Хохлюк

Данная работа посвящена рассмотрению двух прямых алгоритмов отсечения для решения полностью целочисленной задачи линейного программирования. Для упрощенного прямого алгоритма доказана теорема о конечности при более простом правиле выбора строки, порождающей отсечения Гомори, чем в [I-3]. Работа алгоритмов иллюстрируется двумя численными примерами. В заключение обсуждается вопрос о машинной реализации рассматриваемых алгоритмов.

I. Описание алгоритмов

Рассмотрим полностью целочисленную задачу линейного программирования.

Найти максимум функции

$$x_0 = a_{00} + \sum_{j=1}^{n} a_{0j}(-x_j)$$

при условиях:

$$\sum_{j=1}^{n} a_{ij} x_j \leq a_{i0} \quad (i=1,\dots,m),$$

$$x_j \geq 0 \quad (j \in \mathcal{J}),$$

x_j - целые $(j \in \mathcal{J})$,

где все a_{ij} - заданные целые числа ($\mathcal{J} = \{1,\dots,n\}$) .

Вводя тривиальные равенства $x_j = (-1)(-x_j)$ $(j \in \mathcal{J})$

слабые переменные x_{n+i} $(i=1,\dots,m)$, будем рассматривать эквивалентную задачу, которая состоит в максимизации x_0 при условиях $X=A^\circ T^\circ$, где переменные x_i $(i\in J)$ принимают неотрицательные целые значения $(J=\{1,\dots,n+m\})$. Здесь

$$X=(x_0,x_1,\dots,x_{n+m})^T,\quad T^\circ=(1,-t_1^\circ,\dots,-t_n^\circ)^T=$$
$$=(1,-x_1,\dots,-x_n)^T,$$

$$A^\circ=\|a_{ij}^\circ\|=\left\|\begin{matrix} a_{00} & a_{01} & a_{02} & \dots & a_{0n} \\ 0 & -1 & 0 & \dots & 0 \\ 0 & 0 & -1 & \dots & 0 \\ \cdot & \cdot & \cdot & \dots & \cdot \\ 0 & 0 & 0 & \dots & -1 \\ a_{10} & a_{11} & a_{12} & \dots & a_{1n} \\ \cdot & \cdot & \cdot & \dots & \cdot \\ a_{m0} & a_{m1} & a_{m2} & \dots & a_{mn} \end{matrix}\right\|$$

Через A_j° будем обозначать j-ый столбец матрицы $(j=0,1,\dots,n)$. Такая форма записи целочисленной задачи облегчает ее анализ.

Будем считать, что матрица A° прямо допустима $(a_{i0}^\circ\geq 0,$ $i\in J)$.

Пусть A есть матрица порядка $(n+m+1)\cdot(n+1)$, а $(\lambda_0,\dots,\lambda_n)$ – заданная центральная строка. Тогда матрица A', полученная из A после одного преобразования с центральным элементом $\lambda_q\neq 0$ $(q\neq 0)$, связана с A соотношениями

$$A'_q=-(1/\lambda_q)A_q,\quad A'_j=A_j-(\lambda_j/\lambda_q)A_q\qquad (j\neq q)$$

Матрица A' представима в виде: $A'=A\cdot Q$, где Q – матрица, получающаяся из единичной матрицы порядка $n+1$ заменой q-ой единичной строки на строку $(\dots-\lambda_{q-1}/\lambda_q,-1/\lambda_q,-\lambda_{q+1}/\lambda_q,\dots)$.

В симплексном методе в качестве центральной строки выступает выбираемая по определенным правилам строка матрицы A, а в алгоритмах отсечения – строка, соответствующая отсечению Гомори.

В симплекс-алгоритмах и алгоритмах отсечения строится последовательность матриц A^k $(k=0,1,\dots)$, причем матрица A^{k+1} получается из A^k после одного преобразования с центральным элементом. При дальнейшем изложении, ради простоты, индекс k иногда будет опускаться, а также не будет подчеркиваться зависимость номера q центрального столбца от k. Для пары матриц A^k и A^{k+1}, а также векторов T^k и T^{k+1} часто будут использоваться соответственно обозначения A и A', T и T'.

Элементарный прямой алгоритм

Пусть исходная матрица A° является целочисленной и прямо допустимой.

I) Выбор центрального столбца.

В качестве центрального выберем произвольный столбец $A_q\,(q\in\mathcal{J})$, для которого $a_{0q}<0$. Если $a_{0j}\geq 0\ (j\in\mathcal{J})$, то матрица A оптимальна и процесс заканчивается, при этом $X=A_c$ - оптимальное решение целочисленной задачи.

2) Выбор производящей строки.*)

Построим множество индексов строк

$$P(q)=\{i\in\mathcal{J}/a_{iq}>0,\ 0\leq[a_{i0}/a_{iq}]\leq\theta_q\},$$

где $\mathcal{J}=\{1,\ldots,n+m\}$, $\theta_q=\min\limits_{a_{iq}>0}(a_{i0}/a_{iq})$, $[a]$ - целая часть от действительного числа a. Если $P(q)=\phi$, то x. не ограничена сверху и процесс заканчивается. Если же $P(q)\neq\phi$, то в качестве производящей выберем первую сверху строку p, индекс которой принадлежит $P(q)$.

3) Преобразование симплексной таблицы.

В качестве центральной выберем строку, соответствующую отсечению Гомори

$$s=[a_{p0}/\lambda]+\sum_{j=1}^{n}[a_{pj}/\lambda](-t_j), \qquad \text{где} \quad \lambda=a_{pq}.$$

Выполним преобразование с центральным элементом $[a_{pq}/a_{pq}]=1$ и перейдем к I), заменив A на A'.

До сих пор нет доказательства конечности элементарного прямого алгоритма. Ниже более подробно будет рассмотрен вариант прямого алгоритма (упрощенный прямой алгоритм), для которого доказана теорема о конечности при простом конкретном правиле выбора производящей строки.

Присоединим к матрице A° снизу строку $n+m+1$, обозначаемую в дальнейшем для краткости индексом $r\ (r=n+m+1)$. Элементы $a_{rj}\ (j=0,1,\ldots,n)$ этой строки, которые будем считать также целыми, должны удовлетворять следующим условиям:

а) если $A_j^{\circ}\prec 0\quad(j\in\mathcal{J})$, то $a_{rj}>0$;

б) если $a_{rj}<0\ (j\in\mathcal{J})$, то $(1/a_{rj})A_j^{\circ}\prec(1/a_{rq})A_q^{\circ}$,

где $(1/a_{rq})A_q^{\circ}=\min\limits_{\substack{A_j^{\circ}\prec 0\\ j\neq 0}}(1/a_{rj})A_j^{\circ}$ (в лексикографическом смысле).

---

*) Строка, из которой строится отсечение.

в) любое допустимое решение целочисленной задачи удовлетворяет ограничению $\sum_{j=1}^{n} a_{rj} x_j \prec a_{r0}$ („$\prec$" – знак лексикографического сравнения).

Если исходное многогранное множество ограничено, то в качестве такой строки можно взять, например, строку, соответствующую ограничению $\sum_{j=1}^{n} x_j \leq a_{r0}$, где a_{r0} – верхняя граница для всех значений $\sum_{j=1}^{n} x_j$, определяемых допустимыми решениями целочисленной задачи. В частности, величину a_{r0} можно найти, решив задачу линейного программирования, которая состоит в максимизации $\sum_{j=1}^{n} x_j$, при условиях линейной задачи, соответствующей рассматриваемой целочисленной задаче.

Строка r преобразуется по тем же правилам, что и любая другая строка матрицы A. Поэтому в дальнейшем под $A^k (k=0,1,\ldots)$ будем понимать матрицу, содержащую строку r.

Покажем, что из свойств строки r следует ограниченность сверху целевой функции x_0 на многогранном множестве, определяемом матрицей A^0. Действительно, выбирая в качестве центральной строку r и выполняя преобразование с центральным элементом a_{rq}, получим матрицу, двойственно допустимую в лексикографическом смысле, и верхнюю границу $a_{00}-(a_{r0}/a_{rq})a_{0q}$ для значений переменной x_0.

Упрощенный прямой алгоритм

Пусть исходная матрица A^0 является целочисленной, прямо допустимой и содержит строку r.

I) Выбор центрального столбца.

В качестве центрального выберем столбец A_q, для которого

$$(1/a_{rq})A_q = \min_{\substack{a_{0j}<0 \\ j\neq 0}} (1/a_{rj})A_j$$

(сравнение вектор-столбцов производится в лексикографическом смысле). Если $a_{0j} \geq 0 \ (j \in \mathcal{J})$, то матрица A оптимальна и процесс заканчивается, при этом $X=A_0$ – оптимальное решение целочисленной задачи.

2) Выбор производящей строки.

Построим множество индексов строк

$$P(q) = \{i \in J \mid a_{iq} > 0,\ 0 \leq [a_{i0}/a_{iq}] \leq \theta_q\},$$

где $J=\{1,\dots,n+m+1\}$, $\theta_q = \min\limits_{a_{iq}>0} (a_{i0}/a_{iq})$.

Если $r \in P(q)$, то выберем строку $p=r$. Если $r \notin P(q)$, то выберем первую сверху строку p, индекс которой принадлежит $P(q)$. Выбрав производящую строку p, перейдем к 3).

3) Преобразование симплексной таблицы.

В качестве центральной выберем строку, соответствующую отсечению Гомори

$$s=[a_{p0}/\lambda]+\sum_{j=1}^{n}[a_{pj}/\lambda](-t_j), \qquad \text{где } \lambda=a_{pq}.$$

Выполним преобразование с центральным элементом $[a_{pq}/a_{pq}]=1$ и перейдем к 1), заменив A на A'.

Если матрица A является прямо допустимой и целочисленной, то из правил преобразования для рассматриваемых алгоритмов следует, что матрица A' будет такой же.

Для упрощенного прямого алгоритма выбор центрального столбца определен однозначно. Это следует из линейной независимости вектор-столбцов A_j $(j\in J)$ и того факта, что преобразование с центральным элементом является невырожденным преобразованием. Выбор же центрального столбца в элементарном прямом алгоритме может оказаться неоднозначным.

Таким образом, прямой алгоритм представляет собой итерационный процесс, k-ая итерация которого состоит в том, что, исходя из матрицы A^{k-1}, строится матрица A^k $(k\geq 1)$.

2. Вспомогательные утверждения

В лемме 1 указаны свойства, которыми обладают элементы производящей строки после одного преобразования. В леммах 2 и 3 устанавливается лексикографическое возрастание вектор-столбца $(1/a^k_{rq})A^k_q$ от итерации к итерации, в лемме 4 доказывается, что монотонно неубывающая последовательность специального вида через конечное число членов достигает своего предела.

Положим $\tilde{A}_j=(1/a_{rj})A_j$, $\tilde{a}_{ij}=a_{ij}/a_{rj}$ $(j\in J$, $a_{rj}\neq 0$, $i=0,1,\dots,n+m+1)$.

ЛЕММА 1. Пусть $a_{pq}>0$ и в качестве центральной строки выбирается строка, соответствующая отсечению Гомори

$$s=[a_{p0}/\lambda]+\sum_{j=1}^{n}[a_{pj}/\lambda](-t),$$

где $\lambda = a_{pq}$ $(p \neq 0,\ q \neq 0)$.

Тогда $a'_{pq} = -a_{pq}$, $a_{pq} > a'_{pj} \geq 0$ $(j = 0, 1, \ldots, n,\ j \neq q)$

Доказательство. Так как $[a_{pq}/a_{pq}] = 1$, то $a'_{pq} = -a_{pq}$. Для $j \neq q$ $a'_{pj} = a_{pj} - [a_{pj}/a_{pq}]\, a_{pq} = a_{pq}(a_{pj}/a_{pq} - [a_{pj}/a_{pq}])$.

Утверждение леммы непосредственно следует из соотношения $0 < a - [a] < 1$, справедливого для любого действительного числа a.

ЛЕММА 2. Пусть $A'_j = A_j - \lambda A_q$ $(j \in \mathcal{J},\ j \neq q)$ для некоторого действительного числа λ. Тогда для индекса r соотношение

$$a_{rj} A_q \prec a_{rq} A_j \qquad (1)$$

имеет место тогда и только тогда, когда

$$a'_{rj} A_q \prec a_{rq} A'_j \qquad (2)$$

(аналогичное утверждение справедливо и для знака " $\succ$ ").

Доказательство. Для любого индекса i

$$a'_{rj} a_{iq} = (a_{rj} - \lambda a_{rq}) a_{iq} = a_{rj} a_{iq} - \lambda a_{rq} a_{iq},$$

$$a_{rq} a'_{ij} = a_{rq}(a_{ij} - \lambda a_{iq}) = a_{rq} a_{ij} - \lambda a_{rq} a_{iq},$$

откуда $a_{rj} a_{iq} - a_{rq} a_{ij} = a'_{rj} a_{iq} - a_{rq} a'_{ij}$ $(i = 0, 1, \ldots, n+m+1)$.

Из последних равенств следует, что $a_{rj} A_q - a_{rq} A_j \prec 0$ тогда и только тогда, когда $a'_{rj} A_q - a_{rq} A'_j \prec 0$.

Знак равенства в соотношениях (1) и (2) леммы не имеет места, так как рассматриваемые вектор-столбцы линейно независимы, а $a_{rq} > 0$.

Лемма доказана.

Аналогичная лемма может быть сформулирована не только для специально выбранных r и q, но и для любой пары индексов.

Пусть теперь центральный столбец A_q выбирается по правилу, сформулированному для упрощенного прямого алгоритма.

ЛЕММА 3. Пусть неоптимальная матрица A удовлетворяет следующим условиям:

а) если $A_j \prec 0$ $(j \in \mathcal{J})$, то $a_{rj} > 0$;

б) если $a_{rj} < 0$ $(j \in \mathcal{J})$, то $\tilde{A}_j \prec \tilde{A}_q$.

Тогда этими же свойствами будет обладать неоптимальная матрица A', полученная из A после одного преобразования с центральным элементом 1. Кроме того, имеет место соотношение $\tilde{A}_q \prec \tilde{A}'_{q'}$, где q' - номер центрального столбца в A'.

Доказательство. Покажем, что выполняется условие (1) леммы 2 для матрицы A. Из определения $\tilde{A}_q$ непосредственно следует выполнение условия (I) для всех $j \neq q$ таких, что $a_{rj} > 0$. Из условия б) вытекает, что соотношение (I) справедливо и для j таких, что $a_{rj} < 0$. Если же $a_{rj} = 0$, то соотношение (I) также имеет место, так как в этом случае $A_j > 0$.

Таким образом, условие (I) леммы 2 справедливо для всех $j \in J$ $(j \neq q)$ и, по лемме 2, справедливо (2) для тех же индексов.

Если $A'_j < 0$, то из (2) следует, что $a'_{rj} > 0$. Следовательно, условие а) леммы 3 для матрицы A' выполняется.

Разделим обе части соотношения (2) на $a'_{rj} a_{rq} \neq 0$. Если $a'_{rj} > 0$, то $\tilde{A}_q < \tilde{A}'_j$. В частности, так как $a'_{rq'} > 0$, то $\tilde{A}_q < \tilde{A}'_{q'}$. Если же $a'_{rj} < 0$, то $\tilde{A}_q > \tilde{A}'_j$ $(j \neq q)$. Из соотношений $\tilde{A}_q < \tilde{A}'_{q'}$ и $\tilde{A}_q > \tilde{A}'_j$ следует, что $\tilde{A}'_{q'} > \tilde{A}'_j$ для j таких, что $a'_{rj} < 0$ $(j \neq q)$. Для $j = q$, $a'_{rq} = -a_{rq} < 0$ и $\tilde{A}'_q = \tilde{A}_q$, поэтому опять $\tilde{A}'_{q'} > \tilde{A}'_q$.

Таким образом, для матрицы A' выполняется и б). Лемма доказана.

Поскольку для исходной матрицы A^*, содержащей строку r, по построению выполняются условия а)–б), то эти же условия будут иметь место и для всех последующих неоптимальных матриц A^K $(K = 1, 2, \ldots)$, а также будет иметь место лексикографическое возрастание вектор-столбца $\tilde{A}^K_q$ от итерации к итерации.

ЛЕММА 4. Пусть дана бесконечная неубывающая числовая последовательность $\{\alpha_i / \beta_i\}$ $(i = 1, 2, \ldots)$, члены которой удовлетворяют условиям:

а) α_i, β_i – целые;

б) $\alpha_i / \beta_i < d$ $(i = 1, 2, \ldots)$;

в) существует бесконечная подпоследовательность $\{\alpha_{i_K} / \beta_{i_K}\}$ $(K = 1, 2, \ldots)$, для которой $1 \leq \beta_{i_K} \leq M$ (здесь d и M – некоторые положительные числа).

Тогда через конечное число членов от начала данной последовательности будет иметь место соотношение $\alpha_i / \beta_i = const$ для всех оставшихся i.

Доказательство. Пусть первый член из указанной подпоследовательности равен u. Тогда для всех остальных ее членов справедливо неравенство $\alpha_{i_K} / \beta_{i_K} \geq u$, откуда $\alpha_{i_K} \geq u \beta_{i_K}$. Если $u \leq 0$, то $\alpha_{i_K} \geq uM$. Если же $u > 0$, то $\alpha_{i_K} \geq u$.

Таким образом, для чисел α_{i_K} $(K = 1, 2, \ldots)$ имеют место соотножения

$u \cdot M \leqslant \alpha_{i_K} < d \cdot M$ при $u \leqslant 0$ и $u \leqslant \alpha_{i_K} < d M$, если $u > 0$.

Так как числа α_{i_K} и β_{i_K} – целые, то в рассматриваемой подпоследовательности встретится лишь конечное число различных членов.

Ввиду же неубывания этой подпоследовательности, через конечное число членов от ее начала всегда будет $\alpha_{i_K}/\beta_{i_K} = a$, следовательно, и $\alpha_i/\beta_i = a$, где a – некоторая константа.

Лемма доказана.

3. Доказательство конечности упрощенного прямого алгоритма

Теорема. Пусть исходная матрица A° является целочисленной, прямо допустимой и содержит строку r. Тогда упрощенный прямой алгоритм через конечное число итераций приводит к оптимальному решению целочисленной задачи.

Доказательство. Предположим, что теорема неверна. Тогда процесс порождает бесконечную последовательность S неоптимальных матриц A^K $(K = 0, 1, \dots)$, удовлетворяющих условиям а) и б) леммы 3. Рассмотрим связанную с этой последовательностью последовательность столбцов $A^K_\circ$ $(K = 0, 1, \dots)$.

Покажем, что через конечное число итераций от начала процесса столбец $A^K_\circ$ зафиксируется и в дальнейшем изменяться не будет. Действительно, выберем произвольную матрицу A из S и рассмотрим вместе с ней $A' \in S$. В A' $a'_{\circ\circ} = a_{\circ\circ} - [a_{p\circ}/a_{pq}]\, a_{\circ q}$. По условию $a_{p\circ} \geqslant 0$, $a_{pq} > 0$, $a_{\circ q} < 0$, поэтому $a_{\circ\circ} \leqslant a'_{\circ\circ}$. Строгое возрастание на целую величину возможно лишь в том случае, когда $[a_{p\circ}/a_{pq}] \geqslant 1$, т.е. $\theta_q \geqslant 1$.

Ввиду того, что целевая функция $x_\circ$ ограничена сверху на многогранном множестве, определяемом матрицей A°, в последовательности S может встретиться лишь конечное число матриц, для которых $[a_{p\circ}/a_{pq}] \geqslant 1$. Это означает, что через конечное число членов от начала S $[a^K_{p\circ}/a^K_{pq}] = 0$ для всех оставшихся K, а следовательно, для этих же K и столбец $A^K_\circ$ изменяться не будет. Бесконечную последовательность матриц, для которых это имеет место, обозначим через S_1.

Ввиду лексикографического возрастания вектор-столбца $\tilde{A}^K_q$ от итерации к итерации, найдется наименьший индекс l $(0 \leqslant l \leqslant n)$

и такая матрица $A \in S_1$, начиная с которой компонента $\tilde{a}^{\kappa}_{lq}$ не убывает, строго возрастая через конечное число итераций, а компоненты $\hat{a}^{\kappa}_{iq}$ $(i < l)$ не изменяются. Бесконечную последовательность матриц, для которых это имеет место, обозначим через S_2.

Из предположения о бесконечности процесса следует, что существует непустое множество J_∞ индексов строк i $(i=1,\dots,n+m)$, которые выбираются в качестве производящих бесконечное число раз через конечное число итераций. Через конечное число членов от начала S_2 в качестве производящих будут выступать только строки из J_∞ и, возможно, строка r. Бесконечную последовательность матриц, для которых это имеет место, обозначим через S_3. Пусть μ есть максимальный индекс из J_∞ $(\mu \geq 1)$.

Покажем, что для всех матриц из S_3 $1 \leq \mu \leq l$. Пусть $l = 0$. Согласно правилу выбора производящей строки, в S_3 найдется бесконечная подпоследовательность S_3', в которой $a^{\kappa}_{0q} < 0$ и $1 \leq a^{\kappa}_{rq} \leq a^{\kappa}_{r0}$ (например, подпоследовательность S_3' может быть составлена из матриц, в которых строка μ выбирается в качестве производящей). Тогда по лемме 4 числовая последовательность $\{a^{\kappa}_{0q}/a^{\kappa}_{rq}\}$ через конечное число членов достигла бы своего предела, что несовместно с выбором l. Поэтому $l \geq 1$. Если бы было $\mu > l$, то рассматривая бесконечную подпоследовательность, для которой $a^{\kappa}_{lq} \leq a^{\kappa}_{l0}$, $1 \leq a^{\kappa}_{rq} \leq a^{\kappa}_{r0}$, получим то же самое противоречие.

Пусть строка μ выбирается в качестве производящей в некоторой матрице $A \in S_3$. Тогда в матрице A' не может выступать в качестве производящей строка, отличная от μ.

В матрице A $a_{\mu q} > a_{\mu 0}$, $a_{r0} \geq a_{rq}$, $a_{i0} \geq a_{iq}$ $(1 \leq i < \mu)$. Согласно леммам I и 3 в матрице A' $0 \leq a'_{\mu q'} < a_{\mu q}$ и $a_{\mu q}/a_{rq} \leq a'_{\mu q'}/a'_{rq'}$, что дает $a'_{rq'} < a_{rq}$. Из последнего неравенства для индексов i $(1 \leq i < \mu)$ следуют соотношения: $a'_{iq'} < a_{iq} \leq a_{i0} = a'_{i0}$, если $a_{iq} > 0$ и $a'_{iq'} \leq 0 \leq a'_{i0}$, если $a_{iq} \leq 0$. Учитывая последние соотношения и тот факт, что в качестве производящих могут выступать только строки из J_∞ и, возможно, r, заключаем, что в матрице A' не может быть производящей строки, отличной от μ.

В силу леммы I, строка μ может выбираться в качестве производящей конечное число раз подряд. Пусть в матрице A^{κ} строка μ выступает последний раз в качестве производящей, тогда в матрице $A^{\kappa+1}$ ни строка μ, ни строка, отличная от μ, не может быть производящей, а таковая должна существовать.

Установленное противоречие указывает на то, что исходное предположение о бесконечности процесса неверно, т.е. алгоритм конечен.

Теорема доказана.

Можно сформулировать и другие правила выбора производящей строки, совместные с конечностью алгоритма (см., например, [I-3]).

В качестве примера может служить следующее правило.

а) Выберем из $P(q)$ в качестве производящей строку ρ, которая выбиралась на предыдущей итерации. Если такой строки нет в $P(q)$, перейдем к б).

б) Если $z \in P(q)$, то выберем строку $\rho = z$. Если же $z \bar{\in} P(q)$, то выберем первую сверху строку ρ, индекс которой принадлежит $P(q)$.

4. Численные примеры

С целью выяснения вычислительной эффективности прямой алгоритм был записан на α-варианте языка АЛГОЛ-60, при этом была использована общая вычислительная схема для алгоритмов отсечения [4].

Отметим ряд особенностей общей вычислительной схемы:

а) стандартная форма задания исходного числового материала, которая используется для алгоритмов отсечения, а также предусмотрена возможность алгоритмического формирования исходной матрицы, исходя из более экономной записи данных;

б) блочность, т.е. отдельные блоки схемы могут быть использованы в подобных алгоритмах;

в) компактная запись, т.е. хранятся и обрабатываются только ненулевые элементы матрицы;

г) введены параметры наблюдения за счетом, характеризующие работу алгоритма (число итераций, время счета, число ненулевых элементов матрицы и т.п.).

При помощи рабочей программы, полученной по α-программе, для прямого алгоритма был решен ряд задач. Параметры наблюдения за счетом для некоторых из этих задач приведены в таблице.

В примерах I и II, иллюстрирующих работу алгоритмов, производящая строка и центральный столбец указаны стрелочкой. Для удобства выписана центральная строка, центральный элемент отмечен звездочкой.

Пример I. Элементарный прямой алгоритм.

A^0	1	$-x_1$	x_2
x_0	-20	-II	4
x_1	0	- I	0
x_2	0	0	-I
x_3	5	4	-I
x_4	I3	II	-3
x_5	8	I0	-3
S_1	0	I*	-I

A^1	1	$-S_1$	$-x_2$
x_0	-20	II	-7
x_1	0	I	-I
x_2	0	0	-I
x_3	5	-4	3
x_4	I3	-II	8
x_5	8	-I0	7
S_2	I	- 2	I*

A^2	1	$-S_1$	$-S_2$
x_0	-I3	-3	7
x_1	I	-I	I
x_2	I	-2	I
x_3	2	2	-3
x_4	5	5	-8
x_5	I	4	-7
S_3	0	I*	-2

A^3	1	$-S_3$	$-S_2$
x_0	-I3	3	I
x_1	I	I	-I
x_2	I	2	-3
x_3	2	-2	I
x_4	5	-5	2
x_5	I	-4	I

Оптимальное решение:

x_0 = -I3
x_1 = I
x_2 = I
x_3 = 2
x_4 = 5
x_5 = I

Пример П. Упрощенный прямой алгоритм

A^0

	1	$-x_1$	$-x_2$	$-x_3$
x_0	0	-I	-I	-I
x_1	0	-I	0	0
x_2	0	0	-I	0
x_3	0	0	0	-I
x_4	4	-4	5	2
x_5	5	-2	5	0
x_6	6	3	-2	2
x_7	I	2	-5	0
x_z	II	I	3	2
s_1	0	I*	-3	0

$P(1)=\{7\}$

A^1

	1	$-s_1$	$-x_2$	$-x_3$
x_0	0	I	-4	-I
x_1	0	I	-3	0
x_2	0	0	-I	0
x_3	0	0	0	-I
x_4	4	4	-7	2
x_5	5	2	-I	0
x_6	6	-3	7	2
x_7	I	-2	I	0
x_z	II	-I	6	2
s_2	0	-I	I*	0

$P(2)=\{6\}$

A^2

	1	$-s_1$	$-s_2$	$-x_3$
x_0	0	-3	4	-I
x_1	0	-2	3	0
x_2	0	-I	I	0
x_3	0	0	0	-I
x_4	4	-3	7	2
x_5	5	I	I	0
x_6	6	4	-7	2
x_7	I	-I	-I	0
x_z	II	5	-6	2
s_3	I	I*	-2	0

$P(1)=\{6\}$

A^3

	1	$-s_3$	$-s_2$	$-x_3$
x_0	3	3	-2	-I
x_1	2	2	-I	0
x_2	I	I	-I	0
x_3	0	0	0	-I
x_4	7	3	I	2
x_5	4	-I	3	0
x_6	2	-4	I	2
x_7	2	I	-3	0
x_z	6	-5	4	2
s_4	I	-2	I*	0

$P(2)=\{5, z\}$

A^4

	1	$-s_3$	$-s_4$	$-x_3$
x_0	5	-I	2	-I
x_1	3	0	I	0
x_2	2	-I	I	0
x_3	0	0	0	-I
x_4	6	5	-I	2
x_5	I	5	-3	0
x_6	I	-2	-I	2
x_7	5	-5	3	0
x_z	2	3	4	2
s_5	0	-I	-I	I*

$P(3)=\{6\}$

A^5

	1	$-s_3$	$-s_4$	$-s_5$
x_0	5	-2	I	I
x_1	3	0	I	0
x_2	2	-I	I	0
x_3	0	-I	-I	I
x_4	6	7	I	-2
x_5	I	5	-3	0
x_6	I	0	I	-2
x_7	5	-5	3	0
x_z	2	5	-2	-2
s_6	0	I*	-I	-I

$P(1)=\{4, 5, z\}$

A^6

	1	$-s_6$	$-s_4$	$-s_5$
x_0	5	2	-I	-I
x_1	3	0	I	0
x_2	2	I	0	-I
x_3	0	I	-2	0
x_4	6	-7	8	5
x_5	I	-5	2	5
x_6	I	0	I	-2
x_7	5	5	-2	-5
x_z	2	-5	3	3
s_7	0	-2	I	I*

$P(3)=\{5, z\}$

A^7

	1	$-s_6$	$-s_4$	$-s_7$
x_0	5	0	0	I
x_1	3	0	I	0
x_2	2	-I	I	I
x_3	0	I	-2	0
x_4	6	3	3	-5
x_5	I	5	-3	-5
x_6	I	-4	3	2
x_7	5	-5	3	5
x_z	2	I	0	-3

A^7 - оптимальна

Оптимальное решение:

$x_0 = 5$
$x_1 = 3$
$x_2 = 2$
$x_3 = 0$
$x_4 = 6$
$x_5 = I$
$x_6 = I$
$x_7 = 5$
$x_z = 2$

В работе [3] пример П был решен за I8 итераций.

Параметры наблюдения за счетом (элементарный прямой алгоритм)

N	m	n	$c[1]$	$c[2]$	$c[3]$	$c[4]$	$c[5]$	$c[6]$	$c[7]$	$c[8]$
I	7	6	3	0	3	33/33	I/I	I/3	I	0
2	7	8	7	0	3	41/49	I/I	I/3	I	0
3	7	9	4	0	3	45/49	I/I	I/3	I	0
4	7	9	9	0	3	45/52	I/I	I/3	I	0
5	7	9	4	0	3	45/48	I/I	I/3	I	0
6	3	3	3	0	9	I8/20	I/I	I3/I9	I	0
7	2	5	II	0	I06	23/31	I/I	47/I06	I	0

Обозначения:

N – номер задачи;

m – число ограничений;

n – число переменных;

$c[1]$ – число итераций;

$c[2]$ и $c[3]$ – связаны со значением целевой функции ($x_0 = \Omega c[2] + c[3]$, где Ω – достаточно большое положительное число);

$c[4]$ – число ненулевых элементов матрицы, максимальное по всем предыдущим итерациям;

$c[5]$ и $c[6]$ – абсолютная величина элемента матрицы, соответственно минимальная и максимальная по всем предыдущим итерациям;

$c[7]$ – абсолютная величина произведения центральных элементов, максимальная по всем предыдущим итерациям;

$c[8]$ – число итераций для построения прямо допустимой матрицы.

Первые значения параметров $c[4]$ – $c[6]$ приведены для исходной матрицы, а значения параметров $c[1]$ – $c[8]$ (включая вторые значения $c[4]$ - $c[6]$) - после окончания счета.

Задачи I-5 представляют собой целочисленные задачи линейного программирования, эквивалентные некоторым задачам "о максимальном паросочетании в графе", а задачи 6,7 - общего вида.

Литература

I. Glover F. A new Foundation for a Simplified Primal Integer programming Algorithm."Operations Research",I968,I6, N6,727-740.

2. Young R.D. A Primal (All-Integer) Programming Algorithm. J.Res.National Bureau of Standarts",I965,69B,N3,2I3-250.

3. Young R.D. Simplified Primal (All-Integer) Integer Programming Algorithm."Operations Research",I968,I6,N6, 750-782.

4. Анцыз С.М., Хохлюк В.И. Алгоритм I Гомори и модификация Мартина для этого алгоритма (две процедуры для численного решения линейной полностью целочисленной задачи). Оптимальное планирование,вып.I3,Новосибирск,I969,стр. 39-56.

Поступила в редакцию
2.II. 1970 г.