УДК 512.25+519.8

В.И. МУРАВЬЕВ

АЛГОРИТМ СИМПЛЕКС-МЕТОДА ДЛЯ РЕШЕНИЯ ЗАДАЧ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ С ДВУХСТОРОННИМИ ОГРАНИЧЕНИЯМИ

Рассмотрим задачу, отличающуюся от основной задачи линейного программирования наличием двухсторонних ограничений на переменные, а именно задачу вида:

I

$$\sum_{j=1}^{n} t_{ij} x_j \leq F_i \ , \quad i=1,2,\dots,m, \qquad \text{I (1)}$$

$$0 \leq x_j \leq N_j \ , \quad j=1,2,\dots,n, \qquad \text{I (2)}$$

$$\min \sum_{j=1}^{n} c_j x_j \ . \qquad \text{I (3)}$$

Ограничения I (1) могут быть заданы в виде равенств, т.е. задача может иметь вид:

II

$$\sum_{j=1}^{n} t_{ij} x_j = F_i \ , \quad i=1,2,\dots,m, \qquad \text{II (1)}$$

$$0 \leq x_j \leq N_j \ , \quad j=1,2,\dots,n, \qquad \text{II (2)}$$

$$\min \sum_{j=1}^{n} c_j x_j \ . \qquad \text{II (3)}$$

Задачу I будем рассматривать уже с добавленными в каждое из m уравнений I (1) переменными x_{n+i} $(i=\overline{1,m})$:

$$\sum_{j=1}^{n+m} t_{ij} x_j = F_i \ , \quad i=1,2,\dots,m \ , \qquad \text{I (1')}$$

$$0 \leq x_j \leq N_j \ , \quad j=1,2,\dots,n+m \ ,$$

$$N_j = F_i \ \text{для} \ j=n+1,\dots,n+m \ . \qquad \text{I (2')}$$

В линейную форму дополнительные переменные входят с нулевыми коэффициентами. Исходное базисное решение задачи I очевидно:

$$X = (\underbrace{0, 0, \ldots, 0}_{n}, F_1, F_2, \ldots, F_m).$$

Для решения задачи (П) вводятся искусственные переменные x_{n+i} $(i = \overline{1,m})$ и соответствующие им единичные векторы условий, и задача П решается методом искусственного базиса.

В настоящей статье приводится алгоритм и составленная согласно этому алгоритму алгол-программа решения задач I, П, причем в зависимости от вида решаемой задачи в вводимой исходной симплекс-таблице последняя строка (строка коэффициентов минимизируемой линейной формы) имеет различный вид; по виду этой строки и определяется путь, по которому пойдет решение задачи.

Каждая итерация (переход от одного базисного решения к другому) разбита на два этапа.

I-ый этап (соответствует процедуре new в общей схеме симплекс-метода). Базисное решение проверяется на оптимальность.

Если не все Δ_j, $j \in N$, положительны или не все Δ_j, $j \in H$, отрицательны, то оно неоптимально, и мы переходим ко второму этапу -элементарному преобразованию, связанному с введением в базис вектора $A_{\cdot s}$ $(s \bar{\in} B)$.

Здесь B - множество номеров базисных переменных;

N - множество номеров внебазисных переменных, имеющих граничное значение „N_j";

H - множество номеров внебазисных переменных, имеющих граничное значение " 0 ".

Если же все Δ_j, $j \in N$, положительны и все Δ_j, $j \in H$, отрицательны, то базисное решение оптимально, и решение задачи закончено.

П-й этап.

1) Определяется вектор, подлежащий включению в базис.

В базис вводится вектор $A_{\cdot s}$, имеющий наибольшую по модулю оценку Δ_s:

$$\Delta_s = max(|\Delta_I|, |\Delta_{II}|),$$

где

$$\Delta_I = \min_{\substack{\Delta_j < 0 \\ j \in N}} \Delta_j, \quad \Delta_{II} = \max_{\substack{\Delta_j > 0 \\ j \in H}} \Delta_j$$

(в программе Δ_j обозначены через $t\,[m+1, j]$).

2) Определяется вектор, подлежащий исключению из базиса.

Исключается вектор $A_{\cdot s_r}$, на котором достигается наименьшее из трёх чисел $\theta_0^{(i)}$ вычисляемых по формулам :

$$\theta_0^{(1)} = \min_{t_{is}>0} t_{i0}/t_{is}\ , \quad i=1,2,\ldots,m\ , \tag{4}$$

$$\theta_0^{(2)} = \min_{t_{is}<0} (t_{i0}-t_{0j})/t_{is}\ , \quad i=1,2,\ldots,m\ , \tag{5}$$

где $t_{0j} = N_j \quad (j \in B)$, t_{i0} - значения базисных переменных $x_j\ (j \in B)$;

$$\theta_0^{(3)} = t_{0s} \quad ,\text{где} \quad t_{0s} = N_s\ , \tag{6}$$

если переменная x_s , соответствующая вектору $A_{\cdot s}$, имеет граничное значение "ноль" или исключается тот $A_{\cdot s_r}$, на котором достигается наибольшее из трех чисел $\theta_0^{(\ell)}$ и по формулам:

$$\theta_0^{(1)} = \max_{t_{is}>0} (t_{i0}-t_{0j})/t_{is}\ , \quad i=1,2,\ldots,m\ , \tag{7}$$

$$\theta_0^{(2)} = \max_{t_{is}<0} t_{i0}/t_{is}\ , \quad i=1,2,\ldots,m\ , \tag{8}$$

$$\theta_0^{(3)} = -t_{0s}\ , \tag{9}$$

если переменная, соответствующая вводимому в базис вектору $A_{\cdot s}$,имеет граничное значение "N_s" .Пункт2) П-го этапа соответствует в общей схеме процедуре place .

3) Если вводимому вектору $A_{\cdot s}$ соответствует компонента $x_s = 0$ и наименьшим из трех чисел $\theta_0^{(\ell)}$, вычисленных по формулам (4)- (6),оказалось $\theta_0^{*} = t_{0s}$, то

а) пересчитываются базисные переменные и значение линейной формы, иными словами, пересчитывается весь нулевой столбец по формуле:

$$t'_{i0} = t_{i0} - t_{0s} \times t_{is}\ , \quad i=1,2,\ldots,m+1\ ;$$

где $\quad t_{i0} = x_{s_i} \quad (s_i \in B,\ i=1,2,\ldots,m)$.

$$t_{os} = \begin{cases} -N_s, & \text{когда вводимому } A_{\cdot s} \text{ соответствует переменная } x_s \text{, имеющая граничное значение } "N_s"; \\ N_s, & \text{когда вводимому } A_{\cdot s} \text{ соответствует переменная } x_s \text{, имеющая граничное значение "ноль"} \end{cases}$$

(Значения N_j, соответствующие переменным, достигшим в процессе вычисления оптимального решения своей верхней границы, помечаются знаком "минус").

б) меняет знак элемент t_{os} нулевой строки (строки ограничений N_j).

Возвращаемся к пункту 2) II-го этапа

4) Если не выполнены условия пункта 3), то

а) пересчитываются базисные переменные и значение линейной формы по формулам:

$$t'_{io} = \begin{cases} \text{если } i = r\text{, то} & \begin{cases} \theta^*_o, & \text{если } s \in \Theta; \\ \theta^*_o + |t_{os}|, & \text{если } s \in N; \end{cases} \\ \text{если } i \neq r\text{, то} & t_{io} - t_{is} \times \theta^*_o, \end{cases}$$
$$i = 1, 2, \ldots, m+1.$$

где $t_{os} = \begin{cases} -N_s \\ N_s \end{cases}$ (см. п.п. 3, а);

б) пересчитывается главная часть и $m+1$ -ая строка (строка оценок Δ_j) симплекс-таблицы по формулам:

$$t'_{ij} = \begin{cases} t_{rj} / t_{rs} & \text{, если } i = r; \\ t_{ij} - \dfrac{t_{rj}}{t_{rs}} \cdot t_{is}, & \text{если } i \neq r, \end{cases}$$
$$i = 1, 2, \ldots, m+1; \quad j = 1, 2, \ldots, n+m;$$

в) преобразуется t_{os_r} - элемент нулевой строки:

$$t'_{os_r} = \begin{cases} |t_{os_r}| & \text{, если } j \in \Theta,\ t_{rs} < 0 \text{ или } j \in N,\ t_{rs} > 0; \\ -|t_{os_r}| & \text{, если } j \in \Theta,\ t_{rs} > 0 \text{ или } j \in N,\ t_{rs} < 0, \end{cases}$$

где t_{os_r} либо "$+N_{s_r}$", либо "$-N_{s_r}$".

г) в столбце номеров базисных переменных номер выводимой из базиса переменной заменяется номером вводимой. (Пункты 3) и 4) соответствуют процедуре work общей схемы).
Возвращаемся к первому этапу алгоритма (соответствует переходу goto 11 в общей схеме).

Ниже изложена алгол-программа.

```
begin integer k,l;  ввод (k,l);
  begin real array t[0:l+I,0:k+l+I], B[I:I000];
  comment k - количество неизвестных задачи,
          l - количество ограничений задачи,
          t - исходная симплекс-таблица ;
  drum B  ; ввод ( t );
  begin procedure SWTPR (n,m);value
        n,m; integer n,m; comment
основная часть программы состоит из процедуры
      "SWTPR"-"Simplex With Two Ply Restrictions"-
"Симплекс с двухсторонними ограничениями " и обращения к ней;
  begin integer i,j,it,r,s; integer array index[I:m];
        real M,MI,ε; boolean z;
        comment i - номер строки,      j - номер столбца, it
 - количество итераций симплекс-метода,  r  -номер генеральной
строки,  s  - номер генерального столбца, index  - массив номеров базисных переменных,  ε  -точность счёта,  M,MI  -рабочие переменные ,  z  служит признаком оптимальности базисного решения;   it:=0;    ввод ( ε );
     СП  0I77 (I037,t[m+I,I], t[m+I,m+n],0,B[I]);

     comment запись на барабан строки коэффициентов линейной
формы;
   for i:=I step I until m do index [i]:=n+i;
   for j:=n+I step l until n+m do
   if t[m+I,j]≠I then go to N;
   comment определение вида решаемой задачи и выбор пути
решения, метка N обозначает начало отдельной итерации симплекс-метода;
   t[0,0]:=II;   comment t00   - условное число,показывающее,какого вида решается задача: если оно равно II, то
решается задача минимизации  Σ(i=1..m) x(n+i) ,
   for j:=I step I until n do
   t[m+I,j]:=0; go to Is;
R:  СП 0I77(I032,t[m+I,I], t[m+I,
   m+n],0,B[I]);  n:=n-m;
```

t[0,0]:=7 ;comment списывание с барабана коэффициентов линей - ной формы, отбрасывание столбцов векторов условий, соответствующих искусственным переменным, в "R" попадаем лишь тогда, когда получено исходное базисное решение задачи П ;

```
Is:       for i:=I step I until m do
          t[i,m+n+I]:=t[m+I,index[i]];
          for j:=0 step I until n+m do
             begin M:=0; for i:=I step I
                until m do M:=M+
                t[i,j]×t[i,m+n+I];t[m+I,j]:=M-t[m+I,j]
   end подсчета оценки Δj ;
N:        it:=it+I; j:=I;  M:=0;
AO:       if (t[0,j]>0 ∧ t[m+I,j]>ε) then
              begin z:=true; go to A
              end      else
          if (t[0,j]>0 ∧ t[m+I,j]<-ε) then
              begin  z:=false;
A:                   M:=abs(t[m+I,j]);
                     s:=j
              end;
B:        j:=j+I; if j>n+m then go to C
          else if abs(t[m+I,j])>M then
          go to  AO  else go to B;
          comment  I-ый этап алгоритма и пункт I)П-го этапа;

C:        if  M=0  then go to E;
          for i:=I step I until m do
          if (z ∧ t[i,s]>ε) ∨ (¬z ∧
          t[i,s]<-ε) then t[i,m+n+I]:=
          t[i,0]/t[i,s]  else
          if (z ∧ t[i,s]<-ε)∨(¬z ∧
          t[i,s]>ε) then
             begin j:=index[i];
                   t[i,m+n+I]:=(t[i,0]-
                   abs (t[0,j]))/t[i,s]
             end
          else t[i,m+n+I]:= if z then
           10^18 else -10^18;   i:=I;
AI:       MI:=t[i,m+n+I];  r:=i;
BI:       i:=i+I; if i>m then go to CI
```

```
        else if  (z ∧ t[i,m+n+I]< MI)
             ∨ (¬z ∧ t[i,m+n+1]> MI)
        then go to AI else go to  Bi;
        comment нахождение генеральной строки (пункт 2) П-го
этапа);
  CI:   if abs (MI) < abs(t[0,s]) then
        begin  j:=index[r]; t[0,j]:=
               if (z ∧ t[r,s]>0) ∨
                  (¬z ∧ t[r,s]<0) then
               abs (t[0,j]) else-abs(t[0,j]);
               t[r,0]:=if z then MI else
               (MI+abs (t[0,s]));

        comment  проверка выполнения пункта 4) П-го этапа, опе-
рации, связанные с  подпунктами в) и а) пункта 4) ;
               for i:=I step I until m+I do
               begin if i≠r then t[i,0]:=t[i,s] MI;
                     if t[i,0]<0 then
                     t[i,0]:=0; comment
                        оператор,вызванный особенностью транс-
лятора;
               t[i,m+n+I]:=t[i,s]
               end;
               for j:=I step I until m+n do
               begin t[r,j]:=t[r,j]/
                t[r,m+n+I];
                 for i:=I step I until m do
                 if i≠r then t[i,j]:=
                 t[i,j]-t[r,j] t[i,m+n+i]
               end; comment    см. п/п   б) п.4);
            index[r]:=s; comment
            см.  п/п. г) пункта 4);
      end
      else
      begin for i:=I step I until m+I do
            t[i,0]:=t[i,0]-t[0,s]×t[i,s];
      comment   см. п/п  а) пункта  3) ;
            t[0,s]:=-t[0,s]
      end; comment   см. п/п. б) пункт 3);
```

go **to** M; **comment** возвращаемся к I-му этапу;

if(abs(t[m+I,0])<ε ∧ t[0,0] > I0)

then **go** **to** R; **comment** если решалась задача П и удалось минимизировать $\sum_{i=1} x_{n+i}$, т.е. значение линейной формы равно нулю с требуемой точностью, то переходим к " R "- подготовке симплекс-таблицы к минимизации исходной линейной формы задачи П;

```
else
begin for j:=I step I until n+m do
      if t[0,j]>0 then t[0,j]:=0
      else t[0,j]:=abs(t[0,j]);
      for i:=I step I until m do
      t[0,index[i]] :=t[i,0];
```

comment формирование на месте нулевой строки (строки верхних ограничений) оптимального базисного решения;

```
if t[0,0]=? then
СП 0177 (I032,t[m+I,I],t[m+I,m+n+m],0,B[I])
else
СП 0177 (I032,t[m+I,I],
t[m+I,m+n],0,B[I]);
```

comment списывание коэффициентов линейной формы с барабана для подсчёта (уточнения) значения линейной формы на полученном оптимальном решении: в первом случае для задачи П, во втором - для задачи I ;

```
t[0,m+n+I]:=0; for j:=I step
I until m+n do t[0,m+n+I] :=
t[0,m+n+I]+t[0,j]×t[m+I,j];
```

comment подсчёт значения линейной формы;

```
СП 0I76 (II53,t[0,0],
t[0,m+n+I],0,0);
```

comment вывод части массива t (нулевой строки) -оптимального решения; вывод (index); вывод (it)

```
      end
    end;
    SWTPR (k,l)
    end
  end;
  stop end
```

Поступило в редакцию
I5.XI.I967 г.