STICHTING
MATHEMATISCH CENTRUM
2e BOERHAAVESTRAAT 49
AMSTERDAM

AFDELING MATHEMATISCHE STATISTIEK

S 416 *)

AN INPUT SYSTEM FOR LINEAR PROGRAMMING PROBLEMS

by

Jac.M. Anthonisse

October 1969

*) translation of S 414

Printed at the Mathematical Centre, 49, 2e Boerhaavestraat, Amsterdam, The Netherlands.

The Mathematical Centre, founded the 11-th of February 1946, is a non-profit institution aiming at the promotion of pure mathematics and its applications; it is sponsored by the Netherlands Government through the Netherlands Organization for the Advancement of Pure Research (Z.W.O) and the Central Organization for Applied Scientific Research in the Netherlands (T.N.O), by the Municipality of Amsterdam, by the University of Amsterdam, by the Free University at Amsterdam, and by industries.

Summary

This report contains a description of an input system for linear programming. In this system the mathematical formulation of a LP problem serves as input for a computer. The coefficients of the problem are generated from the formulas. Some examples are presented.

Introduction

Any linear programming problem can be written as

maximize $$\sum_{j=1}^{n} c_j x_j$$

subject to the constraints

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i \qquad (i = 1,\ldots,m_1) ,$$

$$\sum_{j=1}^{n} a_{ij} x_j = b_i \qquad (i = m_1+1,\ldots,m) ,$$

$$x_j \geq 0 \qquad (j = 1,\ldots,n) .$$

In this formulation n, m_1, m, c_j, a_{ij} and b_i are fixed numbers. The optimal values of x_j are to be determined.

The formulation of a practical problem as a LP problem often leads to more complex formulas. However, by re-indexing the variables and constraints the above form can be found.

Computer programs that solve LP problems require as input the coefficients c_j, a_{ij}, b_i of the problem. Usually these coefficients must be given column after column, i.e. input consists of the numerical values of

$$\begin{array}{llllll}
n & m_1 & m & & & \\
c_1 & a_{11} & . & . & . & a_{m1} \\
\vdots & & & & & \\
c_j & a_{1j} & \ldots & a_{ij} & \ldots & a_{mj} \longleftarrow \text{j-th column} \\
\vdots & & & & & \\
c_n & a_{1n} & . & . & . & a_{mn} \\
 & b_1 & \ldots & b_i & \ldots & b_m
\end{array}$$

Many programs accept a condensed form of this scheme. Instead of

$$c_j \quad a_{1j} \quad \ldots \quad a_{mj}$$

only the non-zero coefficients must be given, each preceded by a row index.

Whatever scheme of input is used, the step from the (compact) mathematical formulation of a problem to that particular scheme cannot be avoided. This step consists of mainly simple but tedious and error-prone calculations.

Such tasks are best left to a computer.

Syntactical definition of LP problems

In [1] a syntactical definition of LP problems was given, which is closely related to the usual formulation. Any LP problem can be written in accordance with this definition and then processed by a computer. Only minor modifications of an arbitrary mathematical formulation are necessary to obtain this accordance.

To illustrate this the concept 'linear formula', occurring as objective function and as left-hand-side of constraints, will be discussed now.

The primitive element for the construction of linear formulas is the <simple term>, defined as:

<simple term>::= <variable>|<real term>×<variable>

This definition means:
a <simple term> is either a <variable> or a <real term> followed by a × symbol followed by a <variable>.

The definitions of <variable> and <real term> will not be given here, a <variable> corresponds to an unknown in the LP problem, a <real term> does not contain an unknown.

Examples of <simple term> :

x

y[3]

x[i,j×n]

5 × z[k]

(a[i,j] + b[l] - 3.14) × u[i-j,l] ,

in the usual mathematical notation:

$$x$$

$$y_3$$

$$x_{i,jn}$$

$$5z_k$$

$$(a_{ij} + b_l - 3.14)\ u_{i-j,l} \quad .$$

With <real term> a <simple linear form> can be built. This is done by writing down a number of examples of <real term>, separated by + or - symbols, the first possibly preceded by a sign.
Definition:

<simple linear form> ::= <simple term>|
+ <simple term>|-<simple term>|
<simple linear form>+<simple term>|
<simple linear form>-<simple term>

Examples:

- x + 3 × y

a[i,j] × z[i,j]

- b[k] × x[k,l] - z[k]

A <simple linear form> is 'simple' because the corresponding mathematical formulation does not contain the $\sum$ symbol. This symbol is introduced with the concept <final sigma> :

<final sigma>::=

S(<index>,<lower bound>,<upper bound>,<simple linear form>).

A <final sigma> corresponds to:

$$\sum_{\text{<index>=<lower bound>}}^{\text{<upperbound>}} \text{<simple linear form>} ,$$

e.g.

S(j,1,n,a[i,j] × x[j])

means

$$\sum_{j=1}^{n} a_{ij}x_j \quad .$$

A mathematical formulation may contain multiple $\sum$'s. This is possible by the concept <sigma> :

<sigma>::= <final sigma>|
S(<index>,<lower bound>,<upper bound>,<sigma>) .

The expression

$$\sum_{i=1}^{m} \sum_{j=i}^{n} \sum_{k=i}^{j} x_{ijk}$$

now can be written as

S(i,1,m,S(j,i,n,S(k,i,j,x[i,j,k]))) .

The following definitions will cause no difficulties:

<term>::= <simple term>|<sigma>|
<real term>×<sigma>

<linear form>::= <term>|+<term>|-<term>|
<linear form>+<term>|<linear form>-<term> .

Thus a <linear form> consists of one or more examples of <term>, separated by + or - symbols, the first <term> possibly preceded by a sign.
A <term> is either a <simple term> (thus a <simple linear form> is a <linear form>), or a <sigma>, or a <sigma> preceded by a <real term> and a × symbol.

Examples of <linear form> :

$$x$$

$$y + z$$

$$- 3\times y + 4\times z$$

$$S(i,1,j,x[i]) - 3.14 \times S(k,p,q,y[k,1])$$

$$- S(j,1,3,a[i,j] \times z[i,j] - x[i]) + z$$

The expressions

$$5(x + y)$$

and

$$\sum_{i=1}^{m} (x_i + a_i \sum_{j=1}^{n} b_{ij}\, y_{ij})$$

cannot simply be copied, but should be written as

$$5x + 5y$$

and

$$\sum_{i=1}^{m} x_i + \sum_{i=1}^{m} \sum_{j=1}^{n} a_i\, b_{ij}\, y_{ij} \quad .$$

The formulas

$$5 \times x + 5 \times y$$

and

$$S(i,1,m,x[i]) + S(i,1,m,S(j,1,n,a[i] \times b[i,j] \times y[i,j]))$$

are in accordance with the definition <linear form>.

Further details can be found in [1].

Algol-60 program

Starting from the definitions in [1] a computer-program was written, in Algol-60, that accepts the mathematical formulation of a (mixed) LP problem as input.
The program re-indexes the variables and constraints of the problem in accordance with the following standard-form:

maximize $\sum_{j=1}^{N} c_j x_j$

subject to

$$\sum_{j=1}^{N} a_{ij} x_j \leq b_i \qquad (i = 1,\ldots,M_1) ,$$

$$\sum_{j=1}^{N} a_{ij} x_j = b_i \qquad (i = M_1+1,\ldots,M) ,$$

$$l_j \leq x_j \leq u_j \qquad (j = 1,\ldots,N) ,$$

$$x_j = \text{integer} \qquad (j = 1,\ldots,N_1) .$$

A list is printed, containing for each variable $(j = 1,\ldots,N)$ and each constraint $(i = 1,\ldots,M)$ its new index (j or i) and its identification in the mathematical formulation.

A tape is punched, containing

$$N_1 \quad N \quad M_1 \quad M \quad ,$$

followed by

$$b_1 \quad b_2 \quad \ldots \quad b_M$$

and, for each variable,

$$j \quad l_j \quad u_j \quad r_1 \quad a_{r_1 j} \quad \ldots \quad r_p \quad a_{r_p j} \quad M+1 \quad c_j \;,$$

where

j = new index of the variable,

$a_{ij} = 0$ if $i \notin \{r_1, \ldots, r_p\}$.

The tape can serve as input for a program solving the LP problem. In this way a number of problems was processed by the Electrologica X-8 computer of the Mathematisch Centrum.

The program can be modified to give, instead of the new indices, the original identification of each variable and constraint, on the tape.

Example 1

maximize $6x_1 + 4x_2$

subject to

$$4x_1 + 5x_2 \leq 3600$$
$$10x_1 + 4x_2 \leq 3600$$
$$x_1 , x_2 \geq 0$$

This problem can be put into the form of [1] as:

```
OPEN  ⌠simple problem⌡
continuous x1, x2;

MAXIMIZE:          6 × x1 + 4 × x2
⌠constr 1⌡         4 × x1 + 5 × x2 ≤ 3600
⌠constr 2⌡        10 × x1 + 4 × x2 ≤ 3600
⌠non neg 1⌡                     x1 ≥ 0
⌠non neg 2⌡                     x2 ≥ 0
CLOSE
```

A more flexible formulation of the same problem is:

```
OPEN ⌠simple LP problem⌡
integer  a, b, c, d, e, f;
continuous  x1, x2;

MAXIMIZE:          a × x1 + b × x2
⌠con 1⌡            c × x1 + d × x2 ≤ 3600
⌠con 2⌡            e × x1 + f × x2 ≤ 3600
⌠s1⌡                   x1          ≥ 0
⌠s2⌡                            x2 ≥ 0
```

```
INIT {values}

   a + 6          b + 4
   c + 4          d + 5
   e + 10         f + 4

CLOSE
```

The example can also be given as a general LP problem.

```
OPEN {general LP problem}
index  i,j;
integer  m,n;
real  c[j]   (1 ≤ j ≤ n) ,
       b[i]   (1 ≤ i ≤ m) ,
     a[i,j]  (1 ≤ i ≤ m, 1 ≤ j ≤ n);
continuous  x[j]   (1 ≤ j ≤ n);

MAXIMIZE:        S(j,1,n,c[j] × x[j])

{constraints}  S(j,1,n,a[i,j] × x[j]) ≤ b[i]   (1 ≤ i ≤ m)

{non neg}                x[j] ≥ 0      (1 ≤ j ≤ n)

INIT {numerical values}

   n + 2              m + 2
   a[i,j]   + 4          + 5
            + 10         + 4
   b[i]     + 3600       + 3600
   c[j]     + 6          + 4

CLOSE
```

Example 2

The problem

$$\text{minimize} \quad \sum_{j=1}^{t} p_j\, x_j + c_1 \sum_{j=1}^{t} \left[\sum_{i=1}^{j} x_i - \left(\sum_{i=1}^{j-1} d_i + \tfrac{1}{2}\, d_j \right) \right] .$$

subject to

$$\sum_{i=1}^{j} x_i - \sum_{i=1}^{j-1} d_i \le k \qquad (j = 1,\ldots,t) ,$$

$$\sum_{i=1}^{j} x_i \ge \sum_{i=1}^{j} d_i \qquad (j = 1,\ldots,t) ,$$

$$x_j \ge 0 \qquad (j = 1,\ldots,t) ,$$

leads to

```
OPEN {example}
index  i,j;
integer t, c1, k,
        p[j]  (1 ≤ j ≤ t),
        d[i]  (1 ≤ i ≤ t);
continuous  x[j]  (1 ≤ j ≤ t);

MINIMIZE:   S(j,1,t,p[j] × x[j])
     + c1 × S(j,1,t,S(i,1,j,x[i]))
{stock}
 S(i,1,j,x[i]) ≤ k + SUM(i,1,j-1,d[i])   (1 ≤ j ≤ t)
{demand}
 S(i,1,j,x[i]) ≥ SUM(i,1,j,d[i])         (1 ≤ j ≤ t)
{non neg}
                  x[j] ≥ 0               (1 ≤ j ≤ t)
```

```
INIT {specification}

  t + 3
  c1 + 1
  k + 100
  p[j] + 20 + 30 + 35
  d[j] + 60 + 70 + 50

CLOSE
```

Example 3

d_{ik} = demand, in tons, for product i (i = 1,...,5) in period k (k = 1,2,3).

t_{ij} = time, in hours, necessary to produce 1 ton of i on machine j (j = 1,...,4).

In each period each machine is available for 168 hours.

x_{ijk} = number of hours that machine j is used to produce i in period k.

The x_{ijk} are to be determined in such a way that the demand in each period is satisfied, the available hours are not exceeded and the total time is minimal.

minimize $$\sum_{j=1}^{4} \sum_{k=1}^{3} \sum_{i=1}^{5} t_{ij} \, x_{ijk}$$

subject to

$$\sum_{j=1}^{4} \sum_{k=1}^{l} x_{ijk} \geq \sum_{k=1}^{l} d_{ik} \qquad (i = 1,\dots,5;\ l = 1,2,3),$$

$$\sum_{i=1}^{5} t_{ij} \, x_{ijk} \leq 168 \qquad (j = 1,\dots,4;\ k = 1,2,3),$$

$$x_{ijk} \geq 0$$

```
OPEN {planning problem}
index  i,j,k,l;
integer  d[i,k]  (1 ≤ i ≤ 5, 1 ≤ k ≤ 3);
real  t[i,j]  (1 ≤  i ≤ 5, 1 ≤ j ≤ 4);
continuous  x[i,j,k]  (1 ≤ i ≤ 5, 1 ≤ j ≤ 4, 1 ≤ k ≤ 3);

MINIMIZE:

 S(j,1,4,S(k,1,3,S(i,1,5,t[i,j] × x[i,j,k])))

{demand}

 S(j,1,4,S(k,1,l,x[i,j,k])) ≥ SUM(k,1,l,d[i,k])
                                              (1 ≤ i ≤ 5, 1 ≤ l ≤ 3)

{time}

 S(i,1,5,t[i,j] × x[i,j,k]) ≤ 168
                                              (1 ≤ j ≤ 4, 1 ≤ k ≤ 3)

{non neg}

     x[i,j,k] ≥ 0             (1 ≤ i ≤ 5, 1 ≤ j ≤ 4, 1 ≤ k ≤ 3)

INIT {coefficients}

d[i,l]    + 25    + 20    + 30
          + 44    + 40    + 46
          +  6    +  7    +  6
          + 22    + 11    + 32
          + 28    + 29    + 23

t[1,j]  +    6.28  +    3.06  + 100      +  6.07
t[2,j]  +    4.24  + 100      +    4.97  +  5.05
t[3,j]  +    5.27  + 100      + 100      +  5.27
t[4,j]  + 100      +    3.31  + 100      +  6.33
t[5,j]  + 100      + 100      +    3.29  +  4.96

CLOSE
```

Example 4

A graph consists of n vertices, labeled 1,...,n, and m edges, labeled 1,...,m. Edge j connects the vertices s_j and t_j.

A maximal number of vertices is required, such that no pair of vertices is connected by an edge.

maximize $\sum_{i=1}^{n} x_i$

subject to

$$x_{s_j} + x_{t_j} \leq 1 \qquad (j = 1,\dots,m)$$

$$x_i = 0 \text{ or } 1 \ .$$

```
OPEN {internal stability}
index  i,j;
integer  m,n,
         s[j]  (1 ≤ j ≤ m),  t[j]  (1 ≤ j ≤ m);
discrete x[i]  (1 ≤ i ≤ n);

MAXIMIZE:   S(i,1,n,x[i])

{stable}    x[s[j]] + x[t[j]] ≤ 1       (1 ≤ j ≤ m)
{upper}     x[j] ≤ 1                    (1 ≤ j ≤ n)
{lower}     x[j] ≥ 0                    (1 ≤ j ≤ n)

INIT {example}

  n + 5          m + 3
  s[j]  + 1   + 3   + 4
  t[j]  + 2   + 1   + 5

CLOSE
```

Literature

1. Jac. M. Anthonisse

An input system for linear programming problems, part 1: formal description of L.P. problems.

Report S 371, Mathematisch Centrum