ISSN 0257-4306

INVESTIGACIÓN OPERACIONAL

DEPARTAMENTO DE MATEMÁTICA APLICADA
FACULTAD DE MATEMÁTICA Y COMPUTACIÓN
UNIVERSIDAD DE LA HABANA

Vol. 19, No. 2-3, 1998

INDICE

Linear Programming with Linear Restricted Parameters

Leo Schubert, FH-Konstanz, Germany

Abstract

The parameters of linear programming models are often not (in a deterministic or stochastic sense) fix, nor totally free selectable. Classic linear optimisation as an example proceeds on the assumption that prices for different products to be produced are predetermined by the market and on this basis defines quantities for production that maximize the profit. Only in perfect and polypolistic structured markets, prices are fixed by supply and demand. In parametric linear programming models this assumption is modified in a way that the parameters of the model (e. g. the prices) depend on an additional parameter $t \in IR$. Stochastic linear programming replaces determined parameters by stochastic ones. None of the upper models offer the possibility to select parameters out of a constrained area (e. g. prices are constrained by costs and also by the prices of the competition) to maximize the objective function. Hence the following optimisation problem with

the objective function $\quad g(x,p) = p^T x \rightarrow \max$

and the restrictions $\quad A_x x \le b_x, \quad x \in IR_{0+}^n, A_x \in IR^{m,n}, \ b_x \in IR^m,$

$\quad A_p p \le b_p, \quad p \in IR_{0+}^n, A_p \in IR^{m',n}, b_p \in IR^{m'}$

will be regarded.

Two procedures, based on the Simplex-Algorithm will be presented as the solution.

Introduction

Markets are not perfekt and polypolistic structured. Therefore the output-prices are not always predetermined only by the market. Companies attempt to achieve sales advantages by strategies which make their products different in comparison with the competitors. Hence the comparison of products is only reduced possible the companies get their own range for setting the prices.

In the case of continuous batch production the products are manufactured out of the same types of resources. This resources can not or should not be enlarged in short date.

Therefore in batch type costing and planning the use of Linear Programming is helpful. The objective function value is the contribution margin, the profit or the sales volume which should be maximized. For constructing the objective function, it is necessary to know the selling prices of the products. But as mentioned earlier companies have their own range for setting the selling prices. Sometimes cost accountant try to find a better result by alternating the selling prices. Usually the setting of prices is limited by competition, the direct costs, the total production costs and so on.

In the following model, the assumption of fixed prices will be replaced by the assumption of linear restricted prices.

Model

In Linear Programming the physical units of output x are the variables. If the prices for the products are also variables then a nonlinear programming problem results. Because of the tools, which will be used for finding the solution of the problem, it is also part of the linear programming.

The optimization problem consits of the objective function

$$g(x,p) = p^T x \to \max, \tag{1a}$$

the physical units of output constraints

$$A_x\, x \le b_x, \qquad x \in IR_{0+}{}^n,\ A_x \in IR^{m,n},\ b_x \in IR^m \tag{1b}$$

und the price constraints

$$A_p\, p \le b_p, \qquad p \in IR_{0+}{}^n,\ A_p \in IR^{m',n},\ b_p \in IR^{m'} \tag{1c}$$

By joining slacks in (1b) and (1c) equation systems result representing basic feasible solutions[1].

The Fundamental theorem[2] of Linear Programming states besides: "if there is an optimal feasible solution, there is an optimal basic feasible solution". The equivalence of extreme points of the feasible region and basic feasible solutions is stated in the Equivalenztheorem[3] .

If the feasible region for the physical units of output is defined according to (1b) by

$$Z_x = \{x \,/\, A_x\, x \le b_x, \quad x \in IR_{0+}{}^n\} \tag{ 2}$$

and the feasible region for the prices according to (1c) by

$$Z_p = \{p \,/\, A_p\, p \le b_p, \quad p \in IR_{0+}{}^n\} \tag{ 3}$$

then the complete feasible region is

$$Z = Z_x \cup Z_p. \tag{ 4}$$

By fixed p results the optimal region

$$Z_x^* = \{x^* \,/\, p^T x^* = \max_{x \in Z_x} p^T x \} \tag{ 5}$$

and by fixed x

$$Z_p^* = \{p^* \,/\, p^{*T} x = \max_{p \in Z_p} p^T x \}. \tag{ 6}$$

If either p nor x are fixed, the optimal region is called

$$Z^* = \{(x^*,p^*) \,/\, p^{*T} x^* = \max_{x \in Z_x,\ p \in Z_p} p^T x \}. \tag{ 7}$$

---

[1] If a basic feasible solution of the linear equation system exists, it can be found by the "Big-M" method or the "Two -Phase-Simplex" method. (see e.g. Winston (1994), pp.164 and pp. 170)

[2] see Luenenberger (1984), p. 19

[3] see Luenenberger (1984), p. 21

The linear programs with fixed p and the program with fixed x corresponding with (5) and (6) are

$$g(x) = p^T x \rightarrow \max \qquad \text{with } x \in Z_x \tag{8}$$

and

$$g(p) = p^T x \rightarrow \max \qquad \text{with } p \in Z_p. \tag{9}$$

Naturally the Fundamental theorem and the Equivalenz theorem are valid for both linear programs. Therefore,

if $Z_p \neq \varnothing$ and $Z_x \neq \varnothing$, then

$\forall p \in Z_p$: Z_x^* consists at least one optimal feasible solution x^*

(10)

and

$\forall x \in Z_x$: Z_p^* consists at least one optimal feasible solution p^* . (11)

The result of (10) and (11) implies:

Z^* consists at least one combination of a basic feasible solution of the system (1b) with one of (1c). (12)

After a comment on the price constraints a solution for the problem (1) founded on (12) will be presented.

Price constraints

As price constraints are only a special set of constraints regarded. Normally it is not to difficult to estimate the coefficients of that set of constraints.

(The coefficients b_p are in the following displayed without the Index p.)

Kinds of price constraints:

Ia)	$\sum_{j=1}^{n} p_j \leq b^{up}$		*price level - upper bound*
Ib)	$\sum_{j=1}^{n} p_j \geq b^{lo}$	mit $b^{up} \geq b^{lo} \geq 0$	*price level - lower bound*
II)	$p_j \geq p_{j'}$	für $j \neq j'$ und $j, j' \in \{1, ..., n\}$	*value rank order*
IIIa)	$p_j - p_{j'} \leq b^u_{jj'}$		*price difference - upper bound*
IIIb)	$p_j - p_{j'} \geq b^l_{jj'}$	mit $b^u_{jj'} \geq b^l_{jj'} \geq 0$	*price difference - lower bound*
IVa)	$p_j \leq b^u_j$		*price - upper bound*
IVb)	$p_j \geq b^l_j$	mit $b^u_j \geq b^l_j \geq 0$	*price - lower bound*

The average of the prices (see Ia) and Ib)) are in some branches used for the positioning of the assortment as high-priced or low-priced. The decision for a certain price level is an important

part of the price policy. Like in Ia) it is possible to use the sum of prices, instead of the average, if every product has to be supplied.
By the viewpoint of the customer the value of all or some products can be described by a rank order. The type II) of constraints allows to take such ranking order into account.
Sometimes, equal prices for product j and j´ in the rank order should be avoided. This can be reached by the type III) of constraints.
The upper bound b^u_j for the product j (see IVa)) is determined by the market, the customer (or better the price elasticity of demand) or by the prices of the competitors. The lower bound b^l_j in IVb) depends of the direct cost or the total production cost of product j.
Additionally some constrains can be joined, which represent different combinations of prices and demands. In this case, the unequation system (1b) must be adequate constructed.

If the feasible region Z_p is not limited then an optimum may result by $p_j = +\infty$. To prevent such an optimum, the type Ia) can be used. An other possibility for limitation of Z_p is the value rank order type II) with a price upper bound (see IVa)) for at least the product with the highest value rank order.

> If the system of linear unequations of (1c) is build up by constraints of the type Ia) up to IVb) and if $Z_p \neq \emptyset$, then <u>inefficient feasible basic solutions</u> may exist. If Z_p is restricted by type Ia) all feasible basic solutions p^d with
>
> $$\sum_{j=1}^{n} p_j^d < b' \qquad \text{with } b' = \max \{b \,/\, b = \sum_{j=1}^{n} p_j ,\ p \in Z_p\}$$
>
> are inefficient. (13)

For to prove this, the remaining price constraints Ib) up to IVb) must be considered. That constraints are responsible for edges of the convex polyhedron. This edges must be used to reach a vertex p^d with $\sum p_j^d < b'$ coming from a neighbouring vertex p^e with $\sum p_j^e = b'$ (see Ia)). The statement "vektor p^e dominates p^d", is equivalent with $p^d \leq p^e$ and $p_j^d < p_j^e$ for at least one $j \in \{1, ..., n\}$.
The consideration of the constraints begin with constraint Ib). Since the plane of the polyhedron corresponding with the constraint Ib) is parallel to that of constraint Ia), the vertex of constraint Ib) can only be reached about the edges of the other constraints. It is sufficient to regard the rest of the constraints. The edges of the constraints hold the following conditions:

$p_j = p_{j'}$ *(see II))*
$p_j = b^u_{jj'} + p_{j'}$ *(see IIIa))*
$p_j = b^l_{jj'} + p_{j'}$ *(see IIIb))*
$p_j = b^u_j$ *(see VIa))*
$p_j = b^l_j$ *(see VIb)).*

None of the conditions allows the increase of p_j while $p_{j'}$ decreases. So, if a neighbouring vertex p^d should be reached as stated using one of that edges p_j and $p_{j'}$ must decrease. Since the polyhedron is convex, there can´t exist other vertex p^d which are not dominated besides the dominated in the neighborhood of p^e.

To find the efficient feasible basic solutions satisfying (1c), the vector x = 1 in the objective function (9) can be used.

As stated alternatively the value rank order II) with a price bound for at least the product j^1 with the highest value rank order can also be used for the limitation of Z_p. In that case the efficient feasible basic solutions can be found by an appropriate vektor x in the objective function (9).

Solution

For the solution of problem (1) two algorithms will be presented. The first serves a global optima and is useful if n or m´or m are small. The second algorithm is useful in the case of many variables and many constraints. The second algorithm cann´t promise a global optima.

Algorithm 1:

The optimal region Z^* contains at least one combination of the basic solution of (1b) and (1c) as stated in (12).

For solving the problem (1) a modification of the Simplex-Algorithm will be applied. The modification considers that there are m´´ different efficient basic solutions p^* of (1c). That means m´´ different rows instead of one row with the objective function coefficients (Tab. 1).

<table>
<tr><td>x_1</td><td>x_2</td><td>...</td><td>x_{n+1}</td><td>...</td><td>$x_{n'}$</td><td></td></tr>
<tr><td colspan="6">p_{11}^* ... $p_{1n'}^*$
(rows with the objective function coefficients)
(eff. feasible basic solutions of $A_p\, p \le b_p$)
$p_{m''1}^*$ ... $p_{m''n'}^*$</td><td>values of the objective functions g(x,p)</td></tr>
<tr><td colspan="6">A_x and E
(physical units of output constraints)</td><td>b_x</td></tr>
</table>

Tableau 1: Simplex-tableau with m´´ objective functions

The values in the rows with the objective function coefficients indicate like in the simplex algorithm the additional contribution of variable x_j (j=1, ..., n´) to the objective function value if variable x_j becomes a basic variable. For getting an equation system, some slacks x_{n+1}, ..., $x_{n´}$
and the identity-matrix E were joined in (1b).
Each of the m´´ rows with the objective function coefficients must once have non positiv values to indicate that their optimal basic solution x^* had been found. By changing the basic variables, at least m´´ relativ optimal basic solutions x^* will be found. Some of that relativ basic solutions may be identical. At the end the optimal solution (p^*, x^*) of (1) must be selected out of the m´´ relativ optimal basic solution by the value of the objective function.

For the selection of the objective function row i´ the sum-criterion

$$\sum_{p_{i´j}>0} p_{i´j} = \max_i \sum_{p_{ij}>0} p_{ij} \qquad (14)$$

or the number-criterion

$$|\{p_{i´j} \,/\, p_{i´j} > 0, j\in\{1, ..., n´\}\}| = \max_i |\{p_{ij} \,/\, p_{ij} > 0, j\in\{1, ..., n´\}\}| \qquad (15)$$

can be used .
To take the sum-criterion means, to start with that row, which seems to get a high objective function value. But the sum-maximum may be originated in only one big $p_{ij} > 0$. On the other side, the number-criterion starts with a objective function row, which has the most variables offering a contribution to increase the objective function.
The objectiv function column (or pivot column) can also be selected by a sum- and a number-criterion. After the selection of the objective function row i´, the number-criterion

$$|\{p_{ij´} \,/\, p_{ij´} > 0, i\in\{1, ..., m´´\}\}| = \max_{\substack{j \\ p_{i´j}> 0}} |\{p_{ij} \,/\, p_{ij} > 0, i\in\{1, ..., m´´\}\}| \qquad (16)$$

can be used for the selection of the pivot column j´.
The selection of the pivot column by (16) may offer the advantage to discover other optimas while searching the optima of the objective funktion row i´. Hence a combination of the selection rule (15) and (16) seems to be useful.
The selection of the pivot row will be done like the simplex algorithm do.
The described modification of the Simplex-Algorithm may serve to solve the problem (1) under certain conditions. Because of the combinatorial increase of the number m´´ of rows with objective function coefficients, the number of variables n or the number m´ of constraints (1c) or the number m of physical units of output constraints (1b) must be small. If there are only certain types of constraints in use, the number m´´ of basic solutions of (1c) will be small (e.g. if only the constraints of type Ia) and II) are used, then m´´ = n).
If the number of constraints of (1b) is small it is possible to use the algorithm. Then the parts of (1b) and (1c) must be exchanged in the algorithm.

Algorithm 2:

If none of the conditions for using the discribed modification of the Simplex-Algorithm, which guarantees a global optima, are satisfied, another algorithm, based on the Simplex-Algorithm too will solve the problem (1).

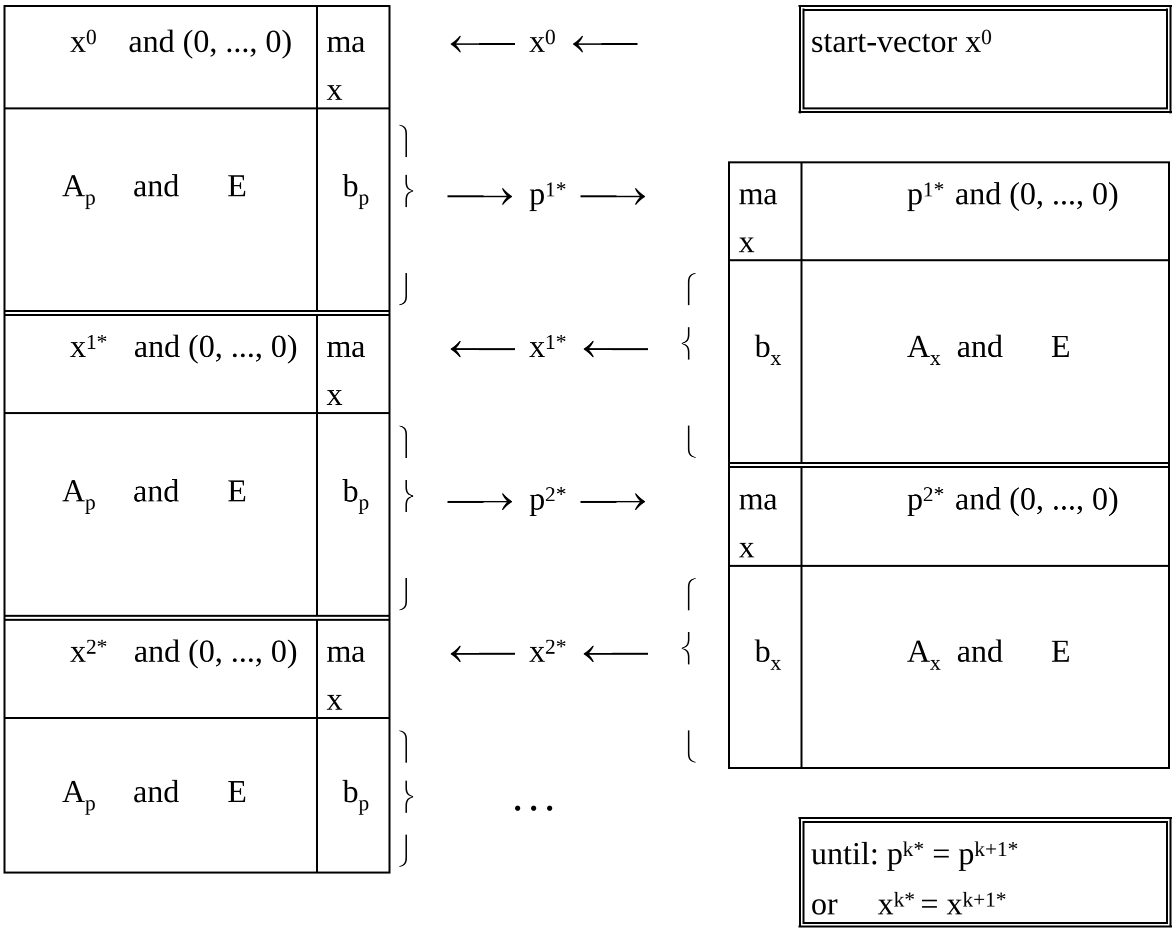

Tableau 2: Alternating use of the Simplex-Algorithm

The principle of Algorithm 2 is shown in tableau 2. First a start-vector x^0 has to be determined. This could be a random vector. Then using this vector x^0 as fixed values in the objective function g, the system (1c) can be solved by the Simplex-Algorithm. Let p^{1*} be the optimal solution of that step. Next the vector p^{1*} is taken as fixed values in the system (1b). Let x^{1*} be the resulting optimal solution of that step. And so on. In every double-step k = 1, 2, ... p^{k*} and x^{k*} are determined. The alternating determination of the solution of (1c) and (1b) using the solution of its previous step will be finished, if $p^{k*} = p^{k+1*}$ or $x^{k*} = x^{k+1*}$.

The alternating determination of the solution of (1c) and (1b) causes an ascending sequence

$$g(p^0, x^{1*}) \leq g(p^{1*}, x^{1*}) \leq g(p^{1*}, x^{2*}) \leq g(p^{2*}, x^{2*}) \leq \ldots$$

of objective function values of (1).

Since the number of feasible basic solutions of (1b) and (1c) is finite (see also (12)) the algorithm converges in a finite number of steps.

Conclusion

Like the Simplex-Algorithm the algorithms 1 and 2 for the solution of the problem (1) are not polynominal time algorithms. It isn´t rare, that in continuous batch production the conditions for using Algorithm 1 (small n or m´ or m) are satisfied. It serves a global optima. The alternating procedure of Algorithm 2 has it´s origin in the principle of ALS (alternating least squares). In data analysis ALS is useful for handling qualitative data[5)] . ALS only guarantees suboptimal fixed points. In opposition to ALS the Algorithm 2 is not a least squares procedure and guarantees local optima. Using the price constraint Ia), it seems, that a start-vector $x^0 = 1$ serves better to find the global optima than random vectors. Since Algorithm 2 uses in every step the Simplex-Algorithm a polynominal time algorithm[4)] could be integrated instead.

References

Hansohm J.: On Convergence of the Normalized Alternating Least Square (ALS) Algorithm, Arbeitspapiere zur Mathematischen Wirtschaftsforschung 68, Universität Augsburg, 1983.

Karmarkar N.: A New Polynominal Time Algorithm for Linear Programming, Combinatorica 4 (1984), pp. 373-395.

Luenenberger D.: Linear and Nonlinear Programming, 2d ed.Reading, Mass.: Addison-Wesley, 1984.

Takane, Y.: Young, F. W., De Leeuw J.: Nonmetric Individual Differences in Multidimensional Scaling: An Alternating Least Square Method with Optimal Scaling Features, Psychometrika, 1977, Vol. 42, pp. 7-67.

Winston W. L.: Operations Research, 3d. ed., Duxbury-Press-Wadsworth, Belmont (California), 1994.

[5)] see Hansohm J. (1983) and Takane Y, Young F. W., De Leeuw J. (1977)
[4)] see Karmarkar (1984)