Cent. Eur. J. Math. • 8(4) • 2010 • 795-806
DOI: 10.2478/s11533-010-0049-9

VERSITA

Central European Journal of **Mathematics**

Mathematical programming via the least-squares method

Research Article

Evald Übi[1]*

1 *School of Economics and Business Administration, Tallinn University of Technology, Tallinn, Estonia*

Received 16 December 2009; accepted 16 June 2010

Abstract: The least-squares method is used to obtain a stable algorithm for a system of linear inequalities as well as linear and nonlinear programming. For these problems the solution with minimal norm for a system of linear inequalities is found by solving the non-negative least-squares (NNLS) problem. Approximate and exact solutions of these problems are discussed. Attention is mainly paid to finding the initial solution to an LP problem. For this purpose an NNLS problem is formulated, enabling finding the initial solution to the primal or dual problem, which may turn out to be optimal. The presented methods are primarily suitable for ill-conditioned and degenerate problems, as well as for LP problems for which the initial solution is not known. The algorithms are illustrated using some test problems.

MSC: 90C05, 65K05

Keywords: Non-negative least-squares solution • System of linear inequalities • Initial solution for linear programming problem • Householder transformation

1. Introduction

It is widely believed that mathematical programming should start with linear programming, where the objective function is linear and constraints are linear inequalities in the unknowns. But how do we solve the inequalities? D.Gale [4] wrote in 1969, "There is one group in the mathematical community who do know how to solve inequalities; these are the people who work in linear programming. The situation here is again curious. Linear programming involves maximizing or minimizing a linear function using variables which are required to satisfy a system of linear inequalities. Thus, in order to solve a linear program one must in the process find a solution of these inequalities. It turns out, on the other

* *E-mail: Evald.Ubi@tseba.ttu.ee*

hand, that the problem of solving inequalities can itself be thought of as linear programming problem in which one is minimizing an artificial objective function... Logically one would first learn to solve the inequalities and then worry about minimizing or maximizing over the set of solutions." For solving inequalities D.Gale recommended using the lexicographic variant of the simplex method of Dantzig, Orden and Wolfe [3].

However, linear inequalities are more complicated than linear equations. Equations as constraints can be transformed into linear inequalities by replacing each equation with the opposing pair of inequalities, but linear inequalities cannot be transformed into equations.

This paper focuses on finding solutions to systems of linear inequalities and other mathematical programming problems by applying the least-squares method (see Section 3). This highly developed method is much older than the simplex method, which was introduced by G.Dantzig only in the the mid-20th century. In the degenerate case the simplex algorithm may stall, performing a number of iterations at a degenerate vertex before producing any improvement in the objective value. With help of the least-squares method such a situation can be avoided.

The least-squares method is not only used in mathematics but also in statistics, physics etc., where mainly nonlinear problems are solved by composing a certain number of similar linear least-squares problems, differing by a variable or constraint. This paper proves that such an idea can also be used for solving linear and quadratic programming problems. The least-squares technique and its applications to mathematical programming are described thoroughly in [6]. The basic problem (Section 2) used is minimizing $\|Eu - f\|^2$, s.t. $x \geq 0$, which is the NNLS (non-negative least-squares) problem. It is equivalent to the phase I algorithm for simplex method, discussed by Leichner, Dantzig and Davis in [7]. Their algorithm solves least-squares subproblems and guarantees strict improvement at each step for degenerate problems. They consider the advantages of using the least-squares method in calculating the initial solution to degenerate problems, but their initial solution does not depend on the objective function. The main contribution of this paper is that this dependency exists, and moreover that the initial solution may even turn out to be optimal. The calculation of the initial basis is of great importance since it determines, to a large extent, the amount of computation that will be required to solve the LP problem, see [2].

The primal-dual simplex method solves a sequence of small phase I linear programming problems to improve the dual solution. There is a description in [1] of a version of the primal-dual method, the least-squares primal-dual algorithm (LSPD), that uses the non-negative least-squares sub-problem and is impervious to degeneracy. Unlike the classical primal-dual simplex method, at each step of the LSPD algorithm a least-squares problem is solved. This algorithm takes less than half the number of iterations required by the primal and dual simplex methods. The assumption set forth in [1], though, is that the initial solution to the dual problem is already known. If not, it is suggested to add an extra constraint and an extra variable to the primal problem, in line with the classical primal-dual simplex method. In this case the dual problem has a trivial feasible solution, which is usually very far from the optimum though. In Sections 4 and 5 we consider finding the initial solution for the LSPD algorithm and for the primal-dual simplex method.

In line with [1] and [7], this paper uses orthogonal transformations of the least-squares method, which have a natural place in linear programming computations since they leave the Euclidean length of a vector invariant. The use of Householder transformations, for achieving greater accuracy, is thus warranted, although they are twice as time consuming as Gaussian eliminations. The simplex method, which has been perfected for the last 60 years, is more suitable for solving stable and non-degenerate problems.

The exact methods for solving linear and quadratic programming problems can for the most part be categorized as either modified-simplex-type methods or projection methods. The former perform simplex-type pivots. The latter are based upon projections onto active sets of constraints. Projection methods are usually more efficient for quadratic programming problems and require less storage than methods of the modified simplex type. In this paper we present a projection-type dual algorithm for solving systems of linear inequalities as well as linear and quadratic programming problems.

In Section 6, exact and approximate quadratic programming algorithms are described. In Section 7, the moving direction for the nonlinear programming problem is found as the solution with minimal norm to the system of linear inequalities.

The main results of this paper are calculating the initial solution to the linear programming problem by applying the least-squares method (Section 5), developing the theoretical foundation for the initial solution in Section 4, and determining the moving direction for the non-linear programming problem (solving a system of linear inequalities using the least-squares method, instead of computing an inverse matrix).

2. The LS1 Algorithm for the non-negative least-squares (NNLS) problem

Consider the non-negative least-squares (NNLS) problem, which is an over or underdetermined system of linear equations

$$Du = f, \quad u \geq 0, \tag{1}$$

or

$$\min\left\{\varphi(u) = 0.5\|Du - f\|^2\right\}, \quad u \geq 0, \tag{2}$$

where D is an $m \times n$ matrix, $u \in R^n$, and $f \in R^m$, see [6].

Assumption 2.1.

The columns of the matrix D and vector f are unit vectors,

$$\|D_j\| = 1, \; j = 1, ..., n, \quad \|f\| = 1.$$

To solve the least-squares problem (1), three finite orthogonal methods are given in [6]. This paper proposes an algorithm that corresponds to the first version of the second method in [6], in which the matrix of the system is not transformed in the computation process. Instead, the QR-transformation is found, where Q is an orthogonal and R is an upper-triangular matrix. Then the least-squares solution is found using the upper-triangular system.

LS1 Algorithm:

Step 0. Initialize $u = 0$, the set of active columns $E = \emptyset$.

Step 1. Find maximum of inner products $F_j = (D_j, f) = F_{j0}$.

Step 2. If $F_{j0} \leq 0$, then u is the solution of problem (1), Stop;

Else $E = [E, D_{j0}]$.

Step 3. Solve the NNLS problem $\|f - Eu_E\|^2 = min\|f - Eu\|^2, u \geq 0$.

Step 4. Find the maximum of inner products $F_j = (D_j, f - Eu_E) = F_{j0}$, and go to Step 2.

Remark 2.1.

The optimality and convergence of the LS1 algorithm are shown in Theorem 4.1 in [14].

Remark 2.2.

When a new variable is activated, we have to first find the corresponding column of the triangular matrix R, then the corresponding column of the inverse matrix R^{-1} and finally using vector u, the new vector u (see [14]).

Remark 2.3.

In the LS1 algorithm rectangular systems of linear equations $(1 \times m, 2 \times m, \dots)$ are solved. LU-factorization for the simplex method does not work for such a system. The columns corresponding to zero components u_j in the solution are removed even though they are linearly independent of the other columns. The solution improves strictly and there is no degeneracy.

Remark 2.4.

The LS1 algorithm is similar to the revised simplex method. The matrix D of the system is not transformed, only the triangular matrix R is used for computing.

Remark 2.5.

The Householder transformations used in the LS1 algorithm are memorized as products, and the algorithm's stability analysis is presented in [6]. The Householder triangularization needs $mn^2 - n^3/3$ operations for solving the least-squares problem (2).

Remark 2.6.

Similar algorithms for solving the NNLS problem, which are also based on the least-squares method, are described in [6, 7, 9]. In these papers, determination of the variable becoming active or inactive is more complicated. As the aforementioned papers do not present the results of solving NNLS explicitly, a comparison of different algorithms under similar conditions is the focus of a forthcoming paper.

Remark 2.7.

The vectors $[E, D_{j0}]$ in Step 2 are linearly independent, see Corollary 2.19 in [7] or [10]. For linearly dependent columns D_j the inner product $F_j = 0$. The criterion for activating variables (Step 4) ensures "maximal" linearly independent columns of the matrix D.

Remark 2.8.

In Step 0, the set of active columns E is empty. If we could easily pick out a starting set of linearly independent columns, then we could start with them already in E, see [7].

3. Theorem of alternatives and the solution with minimal norm to the system of linear inequalities

This section deals with finding the solution of a system of linear inequalities by applying the least-squares method. We first show how to solve inequalities and then transform other mathematical programming problems to the system of inequalities. This approach is suitable for degenerate and ill-conditioned problems.

Let us consider the basic problem

$$\min\{z = \|x\|^2/2\}, \quad Ax \le b, \tag{3}$$

where A and b are given matrices with dimensions $m \times n$ and $m \times 1$, and $\hat{x}(n \times 1)$ is the solution with minimal norm.

Assumption 3.1.

The rows a_i of the matrix A are non-zero vectors, $\|a_i\| \ne 0$, $i = 1, \ldots, m$.

As shown in [6], problem (3) is equivalent to the least-squares problem

$$(b, u) = -1, \quad A^T u = 0, \quad u \ge 0 \tag{4}$$

or

$$\min\left\{\varphi(u) = (1 + (b, u))^2 + \|A^T u\|^2\right\}, \quad u \ge 0, \tag{5}$$

where $u \in R^m$.

Farkas' Lemma.

The system $Ax \le b$ has no solutions if and only if there is a vector u such that

$$(b, u) = -1, \quad A^T u = 0, \quad u \ge 0. \tag{6}$$

Let us compose the dual problem for (3):

$$\min\left\{w = -0.5\|x\|^2 + (y, b - Ax)\right\}, \quad -x - A^T y = 0, \\ y_i(b_i - (a_i, x)) = 0, \quad i = 1, \dots, m, \quad y \geq 0. \tag{7}$$

Theorem 3.1 (Cline).

The relationships

$$\hat{x} = \frac{-A^T \hat{u}}{1 + (b, \hat{u})}, \tag{8}$$

$$\hat{y} = \frac{\hat{u}}{1 + (b, \hat{u})}, \tag{9}$$

hold for the optimal solutions to problems (3), (4) and (7).

Equation (8) was proven by Cline [6], and equation (9) follows from the constraints of the dual problem (7) and equation (8). Vector $\hat{x}$ is the gradient of the objective function (3); it is expressed as a linear combination of the constraint gradients with non-positive coefficients [8].

Theorem 3.2 (of alternatives).

Exactly one of the following alternatives is true:

1) $\exists x \in R^n$ *such that* $Ax \leq b$*, whereby the solution with minimal norm is*

$$\hat{x} = \frac{-A^T \hat{u}}{1 + (b, \hat{u})},$$

and $\hat{u}$ *is the solution of the least-squares problem (4),*

2) *the set* $Q = [x : Ax \leq b]$ *is empty and the least-squares problem (5) has an exact non-negative solution* $\hat{u}$*,* $\varphi(\hat{u}) = 0$.

Remark 3.1.

The solution with minimal norm to the system of linear inequalities (3) may be found using the LS1 algorithm (see Section 2). A detailed description of this process is presented in [14] together with the estimates characterizing the performance of the algorithm.

4. Duality theorems and the least-squares solution to the LP problem

We consider the linear programming problem

$$\min\{z = (c, x)\} \tag{10}$$

subject to

$$Ax = b, \quad x \geq 0,$$

and its dual

$$\max\{w = (b, y)\}, \quad yA \leq c, \tag{11}$$

where A, b, c are given matrices with dimensions $m \times n$, $m \times 1$ and $1 \times n$ respectively, and x and y are $n \times 1$ and $1 \times m$ vectors of variables.

Assumption 4.1.

The columns A_j of the matrix A are non-zero vectors, $\|A_j\| \neq 0$, $j = 1, ..., n$, $\|b\| \neq 0$.

We now introduce into linear programming the notion of a b-basis which is a generalization of the basis.

Definition 4.1.

A set of linearly independent columns of a matrix A, $[A_{i1}, A_{i2}, \dots, A_{ik}] = B$, is said to be a *b-basis* if there exist positive components $x_{i1} > 0, \dots\dots, x_{ik} > 0$ such that

$$\sum_{i1}^{ik} A_{ik} x_{ik} = b, \quad k \leq m.$$

Note that this notion of a b-basis differs from that of the simplex method, since our basis B may contain less than m columns and that these columns may not be sufficient in themselves to span the column space of A. For example, if there exists column A_j, such that $A_j x_j = b$, $x_j > 0$, then the b-basis is $B = [A_j]$ and the corresponding b-basic feasible solution is $x = (0, 0, , , x_j, ...0, 0)$.

The least-squares method is effective in the case of a degenerate basis, with strict improvement attained at each iteration, see [7]. Unlike the simplex method, the least-squares method does not make use of variables whose values equal zero, see Example 4.2.

Let us write for the dual problem of a system of linear inequalities

$$-(y, b) \leq -z_0, \quad yA \leq c, \tag{12}$$

which has unique solution with minimal norm $\hat{y}$, when the parameter $z_0 = z_{\min} = w_{\max}$. In the general case this is one of the optimal solutions to the dual problem. Let us define the NNLS problem (4) corresponding to this system of inequalities:

$$-u_0 z_0 + (c, u) = -1, \quad -u_0 b + Au = 0, \quad u_0 \geq 0, \quad u = (u_1, ..., u_n)^T \geq 0. \tag{13}$$

When constructing the dual problem (11) we rotated the initial problem by 90 degrees. In the corresponding least-squares problem (13) we made another 90 degree rotation and obtained a form similar to the initial problem.

Next we shall study the relationships between problems (10)–(13). We will show that using problem (13) at each value of the parameter z_0, an initial solution can be found to the primal or dual problem, which may turn out to be an optimal one or differ from it only slightly. The number k of positive variables $u_0, u_1, ...u_m$ in the least-squares solution always satisfies the inequalities

$$1 \leq k \leq m + 1,$$

since the number k cannot be larger than the number of linearly independent columns in that problem.

Theorem 4.1.

*If primal problem (10) **has an unbounded** optimal solution, then for every z_0, the least-squares problem (13) has an exact solution $\hat{u}$, $\varphi(\hat{u}) = 0$.*

If the primal problem has an unbounded optimal solution, then the dual problem is infeasible and according to Theorem 3.2 there exists a vector $\widehat{u}$ such that $\varphi(\widehat{u}) = 0$.

Example 4.1.

$$\min\{z = -2x_1\}, \quad -x_1 + x_2 = 1, \quad x \geq 0.$$

Let us compose the least-squares problem (13):

$$-u_0 z_0 - 2u_1 = -1, \quad -u_0 - u_1 + u_2 = 0, \quad u \geq 0.$$

If $z_0 > 0$, then the exact solution is $\widehat{u}_0 = 1/z_0$, $\widehat{u}_1 = 0$ and $\widehat{u}_2 = 1/z_0$. If $z_0 \leq 0$, then $\widehat{u}_0 = 0, \widehat{u}_1 = 1/2$ and $\widehat{u}_2 = 1/2$.

Remark 4.1.

If the LP problem (10) is **infeasible** and the dual problem (11) has an unbounded solution, then for every z_0 the system of linear inequalities (12) holds. We solve problem (13) and by using equation (8) obtain a feasible solution to the dual problem (11). If both problems, primal and dual, are infeasible, then the least-squares problem (13) has an exact solution $\hat{u}$, $\varphi(\hat{u}) = 0$ for every z_0.

Case I. For the chosen parameter z_0 the inequality

$$z_0 > z_{\min} \tag{14}$$

holds.

Theorem 4.2.

In case of inequality (14), problem (13) has an exact solution $\widehat{u}_0, \widehat{u}_i$, with corresponding b-basic solution $\widehat{x}$, where

$$\widehat{x} = \frac{\widehat{u}}{\widehat{u}_0}. \tag{15}$$

Proof. In Case I, system (12) does not hold. According to Farkas' Lemma, problem (13) has an exact solution, while the complementary slackness theorem implies $u_0 > 0$. Dividing the constraints of problem (13) by this variable, we get the b-basic solution with the following value for the objective function:

$$(c, \widehat{x}) = -\frac{1}{\widehat{u}_0} + z_0. \tag{16}$$

□

Case II

$$z_0 \leq z_{\min}. \tag{17}$$

After finding the least-squares solution to problem (13), with the help of equation (8), we can calculate a feasible solution to the dual problem (11) with the value of the objective function not less than z_0. Let us denote by $\widehat{y}$ the element with minimal norm of the set of feasible solutions of the dual problem $\widehat{w} = (\widehat{y}, b)$. If $z_0 < \widehat{w}$, we can calculate the element with minimal norm of the dual problem using the least-squares solution of problem (13) and Theorem 3.2, with $u_0 = 0$. If $z_0 > \widehat{w}$, we can find the feasible solution of the dual problem using equation (8), with the value of the objective function $w = z_0$ (see Example 4.2).

Example 4.2.

Let us consider the LP problem

$$A = \begin{pmatrix} 4 & 2 & 0 & 75 & -3 & 1 & 0 & 5 \\ 5 & 3 & 1 & 90 & -2 & 1 & 0 & 20 \\ 12 & 10 & 0 & 84 & 15 & 0 & -1 & 84 \end{pmatrix}, \quad b = \begin{pmatrix} 75 \\ 90 \\ 84 \end{pmatrix}, \quad c = \begin{pmatrix} -3, & 0, & 0, & -171, & 15, & -2, & -1, & 1 \end{pmatrix}.$$

The exact solution is $x = (15, 0, 0, 0, 0, 15, 96, 0)^T$ or $x = (0, 0, 0, 1, 0, 0, 0, 0)^T, z_{min} = -171$, and $y = (5, -7, 1)$.

The results obtained by solving the least-squares problem (13) in Case I and the corresponding b-basic solutions are given in Table 1. Table 2 presents the feasible solutions to the dual problem found in Case II.

Table 1. Computing results for $z_0 > z_{min}$

z_0	0	1	2	3	4	5	6	7	8
-170	x	15	0	0	0	0	15	96	0
u	1	15	0	0	0	0	15	96	0
-160	x	0	0	0	1	0	0	0	0
u	0.091	0	0	0	0.091	0	0	0	0
-150	x	15	0	0	0	0	15	96	0
u	0.048	0.714	0	0	0	0	0.714	4.571	0
150	x	0	0	0	0	0	70	0	1
u	0.003	0	0	0	0	0	0.242	0	0.003
1000	x	7	0	8	0	0	47	0	0
u	0.0009	0.006	0	0.007	0	0	0.042	0	0

Table 2. Computing results for $z_0 \leq z_{min}$

z_0	0	1	2	3	4	5	6	7	8
-171	y	5	-7	1					
u	0.046	0.855	0	0	0	0	0	6.371	0
-172	y	4.667	-6.733	1					
u	0.048	0.892	0	0	0	0	0	6.648	0
-173	y	4.333	-6.467	1					
u	0.051	0.932	0	0	0	0	0	6.946	0
-180	y	4.286	-6.429	1					
u	0	0.064	0	0	0	0.109	0	2.428	0
-1000	y	4.286	-6.429	1					
u	0	0.064	0	0	0	0.109	0	2.428	0

5. Solving linear programming problems

Two possible ways of using the least-squares method for solving a linear programming problem are described in [1, 7]. In this section we consider finding the initial solution and a method for an approximate solution, both of which are based on the least-squares method.

5.1. Initial solution

Theorem 4.2 is used for finding the initial solution. It is necessary to give the minimum of the objective function z_0 for the initial problem (10) and to solve the least-squares problem (13). When solving the problem the value of the parameter z_0 can be estimated more easily if problem (10) contains the constraints

$$l_j \leq x_j \leq u_j, \quad j = 1, \dots, n.$$

If no such constraints are given, l_j, u_j can be estimated for each particular problem.

As seen in Table 1, if the values of z_0 are within the interval

$$z_{\min} < z_0 \leq z_{\min} + \Delta, \tag{18}$$

for a particular $\Delta > 0$, then we can immediately find the optimal solution to the initial problem (10) using the solution of the least squares problem (13) and equation (15). If the chosen z_0 is too large, $z_{\min} + \Delta < z_0$, we can calculate the feasible solution of the initial problem and continue with the simplex method. According to equation (16), when $z_0 > z_{\min}$, and the difference $z_0 - z_{\min}$ is small, the calculation process may become unstable since the optimal value of u_0 is large.

Table 2 presents the calculation results for Case II. Once $z_{\min}$ is known, we can find the solution with minimal norm to the dual problem by solving problem (13). Then we continue using the primal-dual simplex-method or the least-squares primal-dual method.

If the least-squares problem (13) has an exact solution $\hat{u}$, $\varphi(\hat{u}) = 0$, then for $\hat{u}_0 > 0$ we obtain a b-basic solution to problem (10). If $\varphi(\hat{u}) = 0$ and $\hat{u}_0 = 0$, then both the primal and dual problems are infeasible. We solved the test problems of low and medium dimensions given in the paper [8], and our initial experiments are very encouraging. To sum up, if the LP is degenerate or ill-conditioned, or if we have no initial solution, then it is expedient to start the solution process with the least-squares problem (13) and subsequently implement the respective simplex method.

The classical simplex method is based on computing the inverse matrix. In (12), the basis matrix in triangular form is used for ill-conditioned problems instead of the inverse matrix. The primal and dual variable values are found from the triangular system, by transforming Householder reflections. At every step of the revised simplex method the following two systems are to be solved

$$Bx = b, \tag{19}$$

$$yB = \bar{c}, \tag{20}$$

where B is the basis matrix, $\bar{c}$ is the part of vector c corresponding to basic variables, and $y = (y_1, \dots, y_m)$ is the vector of dual variables.

Let Q be an orthogonal $m \times m$ matrix such that $QB = R$ is an upper triangular matrix. There is no need to compute Q; it will be presented as a product of $m - 1$ Householder transformations [6]. The necessary information is stored under the main diagonal of the triangular matrix R (instead of zeros). The dual variables are sought in the form $y = tQ$, where t is an m-vector, found from the triangular system $tR = \bar{c}$. Finding the inverse matrix Q^{-1} is not necessary either. In order to find the dual variables $y = (Q^{-1}t^T)^T$ one has to apply the $m - 1$ Householder transformations to the vector t^T in reverse order, see [13]. After finding the initial solution by solving problem (13) we can leave the basis matrix in the triangular form, both in the first and the second case, and continue with the respective implementation of the simplex method.

5.2. Approximate method

Let $x*$ be the optimal solution to problem (10) and $x(\varepsilon)$ the least-squares solution to the NNLS problem

$$Ax = b, \quad \varepsilon x = -c^T, \quad x \geq 0, \tag{21}$$

where T denotes the transpose and $\varepsilon > 0$. In [12] it is shown that $x(\varepsilon) \to x*$, as $\varepsilon \to 0$, and the finite VL algorithm for solving the NNLS problem is presented. For large and sparse matrices A, the well-known least-squares technique should be used. Compute the sum of squares

$$\varphi_\varepsilon(x)/\varepsilon = (Ax - b, Ax - b)/2\varepsilon + (c, c)/2\varepsilon + (c, x) + \varepsilon(x, x)/2.$$

Thus, the least-squares solution of problem (21) is equivalent to applying penalty functions and the regularization method to the LP problem. The term $\varepsilon(x, x)/2$ guarantees the stability of the method and enables us to solve unstable problems, see [12]. For example, a test problem with the Hilbert matrix is solved up to the 220th order, while many implementations deal with 4th- to 10th-order matrices.

6. Solving quadratic programming problems

6.1. Exact method

Consider the quadratic programming problem

$$\min\{0.5(y, By) + (d, y)\}, \quad Gy \leq h, \tag{22}$$

where B is a positive-definite $n \times n$ matrix, G is an $m \times n$ matrix, $y \in R^n, d \in R^n$, and $h \in R^m$. Using the Choleski decomposition with $B = D^T D$ and shifting $y = D^{-1}x - B^{-1}d$, problem (22) is transformed to the basic problem (3): find the solution with minimal norm to the system of linear inequalities

$$\min\{z = \|x\|^2/2\}, \quad Ax \leq b, \tag{23}$$

where $A = GD^{-1}$ and $b = h + GB^{-1}d$. The computational results are presented in [15]. Compared to the modified-simplex-type methods, the presented dual algorithm requires less storage and solves ill-conditioned problems more precisely.

6.2. Approximate method

We shall consider the problem of finding an n-vector $x*$ satisfying

$$\min\{z = \|f - Ex\|^2\}, \quad Ax = b, \quad x \geq 0, \tag{24}$$

where E is a $p \times n$ matrix, A is $m \times n$ matrix, $f \in R^p$, and $b \in R^m$. To solve this, consider the NNLS problem

$$Ax = b, \quad \varepsilon Ex = \varepsilon f, \quad x \geq 0, \tag{25}$$

whose least-squares solution is denoted by $x(\varepsilon)$, $\varepsilon > 0$. In [11] it is shown that $x(\varepsilon) \to x*$, as $\varepsilon \to 0$. Usually the quadratic programming problem is formulated in the form of (22). If we transform the objective function z to this form, then matrix E is squared, and its condition number must also be squared (depending on the definition of the condition number). However, [11] recommends the opposite – to transform the objective function of problem (22) using the Cholesky (square root) method to the form of (24).

7. Moving direction for linearly constrained problems

Let us consider the nonlinear programming problem

$$\max\{z = f(x_1, x_2, \ldots, x_n)\}, \quad Ax \leq b, \tag{26}$$

where A is an $m \times n$ matrix, and $b \in R^m$. Below we give a brief overview of the active set method, concentrating first on determining the moving direction p^k from the current point x^k, where

$$x^{k+1} = x^k + s_k p^k, \quad k = 0, 1, 2, \ldots \tag{27}$$

A detailed description of the straightforward part of the algorithm is provided in [5, 16], for when step s_k has positive length. In this case the vector $x^k + s_k grad\, f(x^k)$ is feasible. We examine the case where we have to find the projection of the gradient. We choose the moving direction according to Zoutendijk [16], using normalization N1. In this case, a least-squares problem that changes for one constraint must be solved at every step. To find the projection p of $grad\, f$, we solve the least-squares problem

$$\min\left\{zp = 0.5(p_1^2 + \ldots + p_n^2)\right\}, \quad -(grad\, f(x^k), p) \leq -1, \quad \widehat{A}p \leq 0, \tag{28}$$

where matrix $\widehat{A}$ is the part of matrix A corresponding to the working set $\{i : (a_i, x) = b_i\}$. This problem is solved using the LS1 algorithm from Section 2.

Remark 7.1.

If problem (28) is infeasible, then the condition for convergence is satisfied, and vector x^k is the local optimal solution.

Example 7.1.

$$\max\{z = x_1 + 2x_2 + 2x_3 - 0.1x_1^2 - 0.1x_2^2\}, \quad x_2 + 2x_3 \leq 12, \quad 2x_1 + x_2 + x_3 \leq 16, \quad x_3 \leq 8, \quad x \geq 0.$$

Step 0. Take $x^0 = (2, 4, 4)^T$ as a starting point. The moving direction is found using the least-squares problem

$$\min\left\{zp = 0.5(p_1^2 + p_2^2 + p_3^2)\right\}, \quad 6/10p_1 - 12/10p_2 - 2p_3 \leq -1, \quad p_2 + 2p_3 \leq 0.$$

The solution with minimal norm is $p = p^0 = 1/49(75, 20, -10)^T$, and the step length is $s = min\{196/160, 196/10\} = 196/160 = s_0$.

Step 1. $x^1 = x^0 + s_0 p^0 = 1/40(155, 180, 150)^T$.

$$\min\{zp = 0.5(p_1^2 + p_2^2 + p_3^2)\},$$
$$-9/40p_1 - 44/40p_2 - 2p_3 \leq -1, \quad p_2 + 2p_3 \leq 0, \quad 2p_1 + p_2 + p_3 \leq 0, \quad p^1 = 1/7(-40, 160, -80)^T.$$

Step 2. $x^2 = 1/17(65, 80, 62)^T$.

$$\min\{zp = 0.5(p_1^2 + p_2^2 + p_3^2)\}, \quad -20/85p_1 - 90/85p_2 - 170/85p_3 \leq -1, \quad p_2 + 2p_3 \leq 0, \quad 2p_1 + p_2 + p_3 \leq 0,$$

This system of inequalities has no solution, since the least-squares problem (4)

$$-u_1 = -1, \quad -20/85u_1 + 2u_3 = 0, \quad -90/85u_1 + u_2 + u_3 = 0, \quad -170/85u_1 + 2u_2 + u_3 = 0, \quad u \geq 0$$

has exact solution $\hat{u} = (1, 80/85, 10/85)^T$. The vector x^2 is an optimal solution.

8. Conclusion and future work

The simplex method and various interior point methods are still dominant algorithms used for solving LPs. However, for degenerate and ill-conditioned problems these methods yield poor results. This paper introduced the LS1 algorithm for solving the NNLS problem. It is used to obtain a stable solution to a system of linear inequalities as well as linear and nonlinear programming problems. When the value of the objective function (z_0) is successfully estimated, then the initial least-squares solution to the LP problem is optimal and the presented method may be less labor-intensive than the simplex method. In its current form, z_0 is a parameter of the method, and it determines the amount of computation required to solve the LP problem. In comparison, the simplex method does not have that obstacle.

Future work includes developing a better rule for determining the value z_0 for the objective function.

References

[1] Barnes E., Chen V., Gopalakrishnan B., Johnson E.L., A least-squares primal-dual algorithm for solving linear programming problems, Oper. Res. Lett., 2002, 30(5), 289–294

[2] Bixby R.E., Implementing the simplex method: The initial basis, ORSA J. Comput., 1992, 4(3), 267–284

[3] Dantzig G.B., Orden A., Wolfe P., The generalized simplex method for minimizing a linear form under linear inequality restraints, Pacific J. Math., 1955, 5, 183–195

[4] Gale D., How to solve linear inequalities, Amer. Math. Monthly, 1969, 76(6), 589–599

[5] Gill P.E., Murray W., Wright M.H., Practical Optimization, Academic Press, London, 1981

[6] Lawson C.L., Hanson R.J., Solving Least Squares Problems, Classics in Applied Mathematics, 15, SIAM, Philadelphia, 1995

[7] Leichner S.A., Dantzig G. B., Davis J.W., A strictly improving linear programming Phase I algorithm, Ann. Oper. Res., 1993, 46-47(2), 409–430

[8] Netlib LP Test Problem Set, available at www.numerical.rl.ac.uk/cute/netlib.html

[9] Kong S., Linear Programming Algorithms Using Least-Squares Method, Ph.D. thesis, Georgia Institute of Technology, Atlanta, USA, 2007

[10] Übi E., Least squares method in mathematical programming, Proc. Estonian Acad. Sci. Phys. Math., 1989, 38(4), 423–432, (in Russian)

[11] Übi E., An approximate solution to linear and quadratic programming problems by the method of least squares, Proc. Estonian Acad. Sci. Phys. Math., 1998, 47(4), 19-28

[12] Übi E., On computing a stable least squares solution to the linear programming problem, Proc. Estonian Acad. Sci. Phys. Math., 1998, 47(4), 251–259

[13] Übi E., Application of orthogonal transformations in the revised simplex method, Proc. Estonian Acad. Sci. Phys. Math., 2001, 50(1), 34–41

[14] Übi E., On stable least squares solution to the system of linear inequalities, Cent. Eur. J. Math., 2007, 5(2), 373–385

[15] Übi E., A numerically stable least squares solution to the quadratic programming problem, Cent. Eur. J. Math., 2008, 6(1), 171–178

[16] Zoutendijk G., Mathematical Programming Methods, North Holland, Amsterdam–New York–Oxford, 1976