AN EFFICIENT MODIFICATION OF THE PRIMAL - DUAL TWO PATHS SIMPLEX ALGORITHM

K. DOSIOS, K. PAPARRIZOS, N. SAMARAS*AND A. SIFALERAS
Dept. of Applied Informatics, University of Macedonia,
156 Egnatia Street, Zip code 540 06, Thessaloniki, Greece,
E-mail: dosios@uom.gr, paparriz@uom.gr, samaras@uom.gr, sifalera@uom.gr

Linear programming deals with the problem of minimizing or maximizing a linear function in the presence of linear constraints. The popularity of linear programming can be attributed to many factors such as the ability to model large problems, and the ability to solve large problems in a reasonable amount of time. Lots of real world problems can be formulated as linear programs. The explosion in computational power of hardware has made it possible to solve large-scale linear problems in personal computers.

Since the development of the simplex algorithm by Dantzig [1], many researchers have contributed to the growth of linear programming. In 1979, Khachiyan [2] proposed the ellipsoid method to solve linear problems in polynomial time. Then, in 1984 Karmarkar [3] developed another polynomial, $O(n^{3\ 5})$, algorithm based on projective transformation. Recently, a new algorithm for linear problems, which generates two paths to optimal solution, has been constructed by Paparrizos [4]. This new algorithm is reported in the bibliography as exterior point simplex algorithm (EPSA). EPSA has two major computational disadvantages. These are

(1) It is difficult to construct good moving directions. The two paths generated by the algorithm depend on the initial feasible direction that is closely related to the initial feasible vertex.
(2) There is no known way of moving into the interior of the feasible region. This movement will provide more flexibility in the searching of computationally good directions.

A well-established way of avoiding the previous computational disadvantages is the transformation of exterior path of EPSA into a dual feasible

*corresponding author

simplex path. The algorithm that results from this transformation is called primal-dual two paths simplex algorithm (PDTPSA).

The aim of this paper is to present an improved modification of the Primal-Dual Two Paths Simplex Algorithm (PDTPSA), developed by Paparrizos et al. Let the linear problem be written in the standard form as

$$\begin{array}{llcl} \min & c^T \mathrm{x} & & \\ \text{s.t.} & \mathrm{Ax} & = & \mathrm{b} \\ & \mathrm{x} & \geqslant & 0 \end{array} \qquad \text{(PLP)}$$

where A is an m×n constraint matrix, c is an n×1 column vector, b is an m×1 column vector, T indicates the transpose and 0 represents the n×1 null column vector. The dual problem associated with (PLP) is

$$\begin{array}{llcl} \max & b^T \mathrm{w} & & \\ \text{s.t.} & A^T \mathrm{w+s} & = & \mathrm{c} \\ & \mathrm{s} & \geqslant & 0 \end{array} \qquad \text{(DLP)}$$

where w is an m×1 column vector and s is an n×1 column vector of slack variables. PDTPSA requires an initial dual feasible basic solution and a feasible point for the (PLP) problem. This means that at every iteration the relation s = c -A^Tw $\geqq$ 0 holds. y is a boundary point of the feasible region of (PLP). Then the direction $d^1 = x^1 - y^0$, is calculated where y^0 is an initial, not basic, point of problem (PLP). If point x^1 is feasible to problem (DLP) and point y^1 is feasible to problem (PLP) then the direction d^1 is ascent for the objective function c^Tx.

The direction d^1 creates a ray $R^1 = x^1 + td^1 : t \geqslant 0$, which enters into the feasible region from the boundary point y^2. The hyperplane from which ray R^1 enters the feasible region corresponds to the basic variable x_k. Then a dual pivot operation on which basic variable x_k exits the basis, is performed.

PDTPSA is better than EPSA, because it faces with success the two previous computational disadvantages. The fact that the points y^t, t = 1, 2, ... are boundary leads to a serious computational disadvantage. The disadvantage derives from the point y belongs to more than one hyperplanes. This means that ties exist in the choice of leaving variable which in turn can lead to stalling and/or cycling. In order to avoid stalling and cycling it is preferable the point y to be interior of the feasible region and no boundary as it is in PDTPSA.

Specifically, the new algorithm that was developed and is presented in this paper traverses across the interior of the feasible region in an attempt to avoid combinatorial complexities of vertex-following algorithms. This new algorithm is called Primal-Dual Interior Point Simplex Algorithm (PDIPSA). Also, moving into the interior of the feasible region is translated into an important reduction of number of iterations and CPU time, particularly in linear problems, which are degenerate.

A means of comparison of different algorithms is the execution of extended experimental computational studies. The computational studies constitute a useful tool in the hands of operational researchers for the classification and hierarchy of different algorithms. A computational study on randomly generated sparse dual feasible linear problems is presented to establish the practical value of the PDIPSA. The results are very encouraging for the modified algorithm.

In the computational study with dual feasible randomly generated linear problems an extended comparison performed among Dual Simplex (DSA), Primal-Dual Two Paths Simplex Algorithm (PDTPSA) and Primal-Dual Interior Point Simplex Algorithm (PDIPSA). The revised form of the Simplex Algorithm was used in these algorithm's implementation .

In the experimental computational study three different densities for the randomly generated linear problems were used. The densities are: 1.25%, 2.5% and 5%. For each density three different categories of linear problems were solved; square n×n and rectangles (n/2)×n and n×(n/2). Each one of them includes 6 different classes of problems. Every class includes 10 randomly generated linear problems. The ranges of values of the linear problem coefficients used for the computational study are $c \in$ [0 250], $b \in$ [10 100] and $A \in$ [-200 500]. The feasibility tolerance used is 10^{-8} and the tolerance on a pivot row and column is 10^{-10}. Totally 540 linear problems were solved for all different densities and dimensions.

In particular for linear problems of size 1000×1000 and for densities 5%, 2.5% and 1.25% PDTPSA is 4.09, 5.24 and 5.17 times faster than DSA concerning the number of iterations, while PDIPSA is 4.49, 5.72 and 5.99 times faster. In terms of CPU time PDTPSA is 7.61, 9.01 and 8.31 times faster than DSA whereas PDIPSA is 8.33, 10.05, 10.59 times. The conclusion which results is that for the linear problems as the dimension increases and the density decreases, so much more rapid becomes the PDITPSA over the DSA and PDTPSA.

References

1. B.G. Dantzig, Programming in a linear structure, Comptroller, US Air Force, Washington, D.C., 1948.
2. L.G. Khachiyan, A polynomial algorithm in linear programming, *Soviet Mathematics Doklady* **20** pp. 191-194(1979).
3. K.N. Karmarkar, A new polynomial time algorithm for linear pro-gramming, *Combinatorica* **4** pp. 373-395(1984).
4. K. Paparrizos, An exterior point simplex algorithm for (general) linear programming problems, *Annals of Operations Research* **47** pp. 497-508(1993).