

Redundancy for Solving Degeneracy Systems **الفائضية في حل الانظمة الخطية المضمحلة**

Alauldin N. Ahmed* & Iraq Tariq Abbas**

(*)Al-Nahrain University/ Science College/ Math. & Comptr. Appl. Dept.

(**)Baghdad University/ Science College/ Math. Dept.

Abstract

A new approach for solving degenerate linear system, is developed, by constructing new rules, making use of the philosophy of redundancy constraints, whether the selective pivot degenerate constraint is active or not, A good results have been obtain compared with the lowest-index rules for solving such problems.

Key Words: Linear Programming, Simplex Method, Degeneracy, Redundancy.

الخلاصة :-

في هذا البحث تم تقديم أسلوب جديد لحل نظام خطي مضمحل (degenerate linear system) في تحويل طريقة السمبلكس (Simplex Method) من خلال بناء اساس جديدة, وبلاستفادة من فلسفة مفهوم الفائضية لضوابط المسألة عند اختيارها فيما اذا كانت الضوابط المضمحلة (degenerate constraint) فعالة ام لا. وقد تم الحصول على نتائج جيدة مقارنة بالاساليب المعتمدة (lowest-index rules) لحل هكذا مسائل.

1. Introduction:

The degeneracy is the worst-case complexity of the randomized simplex algorithm. It is well know that every linear programming (LP) problem can be perturbed into a non-degenerate (ND) problem [6,9,10]. The original proof that the simplex algorithm would converge to an optimal solution based on the non-degenerate assumption (NDA). For such a problem, if a problem does not satisfy the NDA, then there is the possibility that the simplex algorithm would not converge to the optimal solution, that is, it would be cycle. Problem that did not satisfy the NDA were easy to construct, but to find one that did not converge took some effort. The first instance of a linear programming problem that was shown to cycle is the one constructed by Hoffman [6].

All commercial LP software that we are aware of apply rules for handling degeneracy, breaking ties, perturbation techniques, and composite primal and dual computations that enable the computer-based simplex algorithm to converge to an optimal solution even if the given problem exhibits classical cycling. The following linear programming problem has been considered :

Minimize $C^T X$,

Subject to:

$$AX \geq B,$$

$$X \geq 0$$

Where $X = (x_1, x_2, \dots, x_n)^T$, $C^T = (c_1, c_2, \dots, c_n)$, $B = (b_1, b_2, \dots, b_n)^T$, with $b_i = 0$ (for some i) and $A = m * n$ matrix.

A degenerate system could cause difficulties during performing the simplex method. Degeneracy may become evident in the simplex method, when leaving variable is being selected in the iterative process, under the pivot column which determines the leaving variable if there is a tie (if the minimum ratio is the same for two or more rows), arbitrary selection of one of these variables may result in one or more variables becoming zero in the next iteration and the problem become degenerate, and in this case, it is usual that one or more of the subsequent pivots will be degenerate, and return to a case that has appeared before, in which case the simplex enters an infinite loop and never attains to the optimal solution, and this behavior is called “Cycling”. Therefore, if the simplex method cycles, then all the pivots within the cycle must be degenerate, since the objective function value never changes. Hence, it follows that all the pivots within the cycle must have the same objective function value, i.e., all of these pivots must be degenerate. In practice, degeneracy is very common, but cycling is rare. In fact, it is so rare that most efficient implementations do not take precautions against it.

2. Previous Approach:

The first method, that deals with the degeneracy is the perturbation for each constraint, and smaller on each succeeding constraint, in which it turns out that the method produces a variant of the simplex method that never cycle [12]. In [5] and [7], a modified the simplex method that do not cycle, by making a new pivoting rules for which the simplex method will definitely either reach an optimal solution or prove that no such solution exists, are presented. One of the modifications rules is in the selecting choices, when there are ties in selecting the variable x_k to enter the basic solution and ties occur in selecting the variable x_j to be removed from the basic solution. In selecting x_k , we can choose any variable that will improve the value of the objective function, while the choice of x_j must correspond to a basis change that preserves feasibility, in which it does require a different and more computationally involved process for determining x_l . In contrast, a very-easy-to-use anti cycling procedure, due to **Bland** in [3], requires the selection of both x_k and x_l to be made under modified, but simple, decision rules. The modified rules are the following:

1. Among all candidates **to enter** the basic solution, select the variable x_j having the lowest index.
2. Among all candidates **to leave** the basic solution, select variable x_i having the lowest index.

One possible use of the lowest-index rules is as an anti-degeneracy procedure; that is apply the rules to a problem after a number of iterations have been completed without a change in the value of the objective function. To demonstrate the performance, of the standard simplex, and the lowest –index rules, we will consider the following problem [4]:

$$\text{Minimize } Z = -3/4x_1 + 150x_2 - 1/50x_3 + 6x_4$$

Subject to:

$$\begin{aligned} 1/4x_1 - 60x_2 - 1/25x_3 + 9x_4 + x_5 &= 0 \\ 1/2x_1 - 90x_2 - 1/50x_3 + 3x_4 + x_6 &= 0 \\ x_3 + x_7 &= 1 \\ x_j \geq 0, j = 1,2,3,4,\dots,7 \end{aligned}$$

The iterations of the cycle are shown in [4], using the standard rule for selecting a vector to enter the basis, and, seven iterations the solution is identical the first one and never reached to optimal solution. But whenever ties occur in which vector is to leave by applying the degeneracy procedure, that choosing one with the lowest index, a different sequence of solutions is obtained and determine the minimum solution.

3. Proposed Approach:

Although the lowest-index rules are easy to apply, but still inefficient from the point of the number of simplex iterations in solving the problem, in which, tests had been done, where the standard rules and the lowest-index rules were compared showed that the standard rules required less number of iterations on their test problems, see [8].

Before we propose our approach, first, we present some essential definitions that are required in our approach, see [1], [2] and [5]. We consider the feasible region Ω define as following

$\Omega = \{x \in R^n : A_i^T \leq b_i, i \in I\}$, where $A_i^T \leq b_i$ is refer to the i -th constraint. The region represented by all but the i -th constraint is given by

$\Omega_j = \{x \in R^n : A_j^T \leq b_j, i \in I \setminus \{j\}\}$, where $I \setminus \{j\}$ is the set I with the element j removed.

Definition(3.1):The i -th constr. $A_i^T \leq b_i$ is said to be inactive in the description of Ω if $\Omega = \Omega_j$, and otherwise is said to be active.

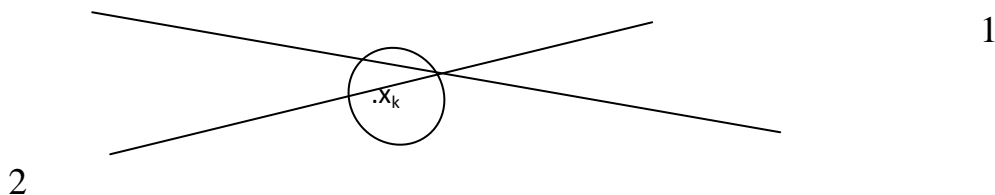
Definition (3.2): The i -th constr. $A_k^T \leq b_k, \forall k$ with $b_k=0$ is an active constraint, if $a_{kj} \geq 0$ for all index j correspond to nonbasic slack variables.

In this paper, we are implemented the philosophy of active constraints in a degenerate linear system, in order to overcome this problem, by investigating two rules, based on, whether the i -th constraints $A_i^T x=0$, can be identified are active or not, in order to be selected as a pivot constraint or not, in performing the simplex method. The correct identification of active constraints is important from both a theoretical and a practical point of view. Theoretically, the identification of the active constraints is not difficult. However, as far as we are aware of, to date no technique can successfully identify all active constraints, see [1], [2], [5], and [11]. To do this, we are presenting the following definition.

Definition (3.3): The projection $P_j(x_k)$ of the point x_k onto the hyper plane $H_j = \{x \in R^n : A_j^T = b_j\}$, is defined by $P_j(x_k) = x_k + A_j(b_j - A_j^T x_k)$.

Consequently, we have $\| P_j(x_k) - x_k \| = A_j(b_j - A_j^T x_k) = \text{dis}(x_k, H_j)$.

In [2], a definition of local inactive nonlinear constraint is presented, which is of no use, in identifying whether the constraint is active or not, (since local redundant constraint may be non-redundant in another local feasible region), as illustrated in the following figure:



(An illustration of a locally active constraint. Constraint 1 is locally active at x_k)

Therefore, we prefer to define a local active constraint, which its existence is necessary to keep the whole feasible region of the problem unchanged. In doing so, we set $\delta > 0$, and define

$U_\delta(x) = \{y \in R^n : \|y - x\| \leq \delta\}$. We use the usual definition of the distance function $dis(.,.)$ between the point and set of linear equations. Suppose that " ϵ " the infimum distance between the point x and the set of the constraints Ω at a local region, we can present the following definition.

Definition (3.4): The constraint $A_r(x) \leq 0$ is *locally active* at x if there exist an open set $\delta(\epsilon)$, such that $A_r \in \Omega \cap \delta(\epsilon)$ & $\Omega_r \cap \delta(\epsilon) = \emptyset$, where $\delta(\epsilon)$ is the closure of δ .

We can state the above definition into another way:

Definition (3.5): The constraint $A_r(x) \leq 0$ is *locally active* at x , if for some (ϵ) , $\Omega \cap \delta(\epsilon) \neq \Omega_r \cap \delta(\epsilon)$.

Otherwise, it is in active. If we denote the following:

x_j^B = Basic Real variables ($j=1, \dots, n$),

s_i^B = Basic Slack variables ($i=1, \dots, m$),

$int(x_k)$ = Interior feasible point at the current iteration k ,

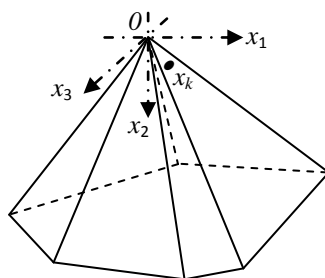
$dist(int(x_k), A_i)$ = The distance between the $int(x_k)$ and the i -th constraint A_i .

Our procedure, start at any iteration with degenerate solution, by constructing the feasible interior point $int(x_k)$ near the current extreme solution, as in the case of degenerate active, when the basis at the iteration k does not have full rank, and the active constraints are linearly dependent, which can be illustrated by the following example, see [2].

Suppose that, the feasible region, is defined by the following linear system of inequalities:

$$\begin{aligned} x_1 - 2x_2 - 2x_3 &\leq 0 \\ -2x_1 + x_2 - 2x_3 &\leq 0 \\ -2x_1 - 2x_2 + x_3 &\leq 0 \\ x_1 &\geq 0 \\ x_2 &\geq 0 \\ x_3 &\geq 0 \end{aligned}$$

If the $int(x_k) \in R^3$ is near the origin, all six constraints are active, linearly dependent, and illustrated in the following figure:



Now, we can state the following tests:

Test (1):

At any degenerate linear system, if basic real variable $x_j^B=0$, with the positive coefficient in the objective function, and its corresponding current pivot element is positive, and then it is an inefficient variable.

Test (2):

At any degenerate constraint $A_r x \leq b_r$, with all coefficient are non-negative, a variable x_k correspond positive coefficient inefficient.

Test (3):

At any iteration, a variable x_k , with negative reduce cost is inefficient, if its all colum coefficient are non-positive.

Test (4):

At any iteration, a variable x_k , with negative reduce cost is inefficient, its all colum coefficient corresponding to each degenerate constraint.

Test (5):

At any degenerate solution, if the constraint A_r corresponding to $\text{Min}b_i\{\text{dis}(\text{int}(x^B_j),A_j)\}$, then A_r is an active constraint.

As a result of the above tests, the following theorem can be stated:

Theorem:

The modified simplex method for degeneracy problem, always terminated in less number of iteration, provided that the leaving variable is selected according to test (5).

Proof:

The constriction of an interior feasible solution and Applying test(5) to identify the pivot constraint, is presented the simplex pivot iteration from cycling, and the iteration will terminated. while the other remain tests, is presented any inefficient variable from entering the basis, since such variables are enter the basis and leaving the basis later, therefore the iteration of our modification method is converge either the optimal or unbounded solution , and terminated after a finite less number of iterations.

Basis		b	x_1	x_2	x_3	x_4	x_5	x_6	x_7
	C		-3/4	150	-1/50	6	0	0	0
x_5	0	0	1/4	-60	-1/25	9	1	0	0
x_6	0	0	1/2	-90	-1/30	3	0	1	0
x_7	0	1	0	0	1	0	0	0	1
z_j-c_j			3/4	-150	1/50	-6	0	0	0

According to test (3) x_2 is an inefficient variable, and it may be deleted from the tableau, and according to test(5) constraint 2 is more active than constraint 1, and become the pivot constraint.

II

Basis		B	x_1	x_2	x_3	x_4	x_5	x_6	x_7
	C		-3/4	150	-1/50	6	0	0	0
x_5	0	0	0	-15	-7/300	15/2	1	-1/2	0
x_1	-3/4	0	1	-180	-1/15	6	0	2	0
x_7	0	1	0	0	1	0	0	0	1
z_j-c_j		0	0	-15	7/100	-21/2	0	-3/2	0

III

Basis		B	x_1	x_2	x_3	x_4	x_5	x_6	x_7
	C		-3/4	150	-1/50	6	0	0	0
x_5	0	7/300	0	-15	0	15/2	1	-1/2	7/300
x_1	-3/4	1/15	1	-180	0	6	0	2	1/15
x_3	-1/50	1	0	0	1	0	0	0	1
$z_j - c_j$		-7/100	0	-15	0	-21/2	0	-3/2	-7/100

Prblem1[7]

Minimize $-2.2361x_4+2x_5+4x_7+3.6180x_8+3.236x_9+3.6180x_{10}+0.764x_{11}$

Subject to:

$x_1=1$

$x_2+0.3090x_4-0.6180x_5-0.8090x_6-0.3820x_7+0.8090x_8+0.3820x_9+0.3090x_{10}+0.6180x_{11}=0$

$x_3+1.4635x_4+0.3090x_5+1.4635x_6-0.8090x_7-0.9045x_8-0.8090x_9+0.3090x_{10}+0.6180x_{11}=0$

$x_j \geq 0$

Solution: $x_1=1;x_j=0(j=2,\dots,11)$;Minimum=0;Cycle=10"MathLab Version 6";no cycle "Proposed Method".

Prblem2[9]

Maximize $x_3-x_4+x_5-x_6$

Subject to:

$x_1+2x_3-3x_4-5x_5+6x_6=0$

$x_2+6x_3-5x_4+2x_6=0$

$3x_3+x_4+2x_5+4x_6+x_7=1$

$x_j \geq 0$

Solution: $x_1=2.5;x_2=1.5,x_5=0.5$;Maximize=0.5;Cycle=6"MathLab Version 6";no cycle "Proposed Method".

Prblem3[9]

Maximize $x_3-x_4+x_5-x_6$

Subject to:

$x_1+x_3-3x_4-5x_5+6x_6=0$

$x_2+6x_3-5x_4+2x_6=0$

$3x_3+x_4+2x_5+4x_6+x_7=1$

$x_j \geq 0$

Solution: $x_1=3;x_2=2,x_5=1$;Minimum=1;Cycle=6"MathLab Version 6";no cycle "Proposed Method".

Prblem4[6]

Maximize $2x_1+4x_4+4x_6$

Subject to:

$x_1-3x_2-x_3-x_4-x_5+6x_6=0$

$2x_2+x_3-35x_4-x_5+2x_6=0$

$x_j \geq 0$

Solution: All variable=0; Minimum=0; Cycle=6"MathLab Version 6";no cycle "Proposed Method".

Prblem5[9]

Minimize $-2x_3+8x_5+2x_6$

Subject to:

$$x_1-7x_3-3x_4+7x_5+2x_6=0$$

$$x_2+2x_3+x_4-3x_5-x_6=0$$

$$x_j \geq 0$$

Solution: All variable=0; Minimum=0; Cycle=6"MathLab Version 6";no cycle "Proposed Method".

Prblem6[13]

Maximize $3x_1-80x_2+2x_3-24x_4$

Subject to:

$$x_1-32x_2-4x_3+36x_4+x_5=0$$

$$x_1-24x_2-x_3+6x_4+x_6=0$$

$$x_j \geq 0$$

Solution: Unbounded; Cycle=6"MathLab Version 6"; no cycle "Proposed Method".

Discussion and Conclusion:

One can see, applying lowest-index rules, the solution path is different that the solution path in standard simplex method, and more number of iterations are needed to be performed to reach an optimal solution. While as we have seen, applying our suggestion rules in selecting the pivot constraint and ignoring any inefficient variable to enter the basis, will change the solution path in an optimal direction to reach an optimal solution, in two iteration only, comparing with six iteration by applying the lowest-index rules. We believe that in the degeneracy linear system, the performing of the lowest-index rules, does not considering the selected pivot constraint, whether, active or not, among several degenerate constraints, would cause a different path solution to reach optimal solution. Indeed, such result, required more studies to be done, on different structure of degeneracy problems to extend and verify a theory of our approach.

References:

- [10] **Audet C., and Dennis Jr., (2003)**; Analysis of generalized pattern searches. SIAM J. Optima., 13(3), 889-903.
- [11] **Abramson M.A., et. al. (2007)**; Pattern search in the presence of degeneracy. Available online at: (mark.abrason@afit.edu)
<http://www.afit.edu/en/enc/Faculty/MABramson/abramson.html>).
- [7] **Bland, R.G. (1977)**;"New finite pivoting rules". Mathematics of Operation Research 2,103-107.
- [8]**Beale, E.M.L. (1955)**; "Cycling in the Dual Simplex Algorithm", Naval Research Logistics, Quarterly, Vol.2, No.4.
- [5] **Caron R.J., et. al. (1989)**; A degenerate extreme point strategy for the classification of linear constraints as redundant or necessary. J. Optim. Theory Appl., 62(2), 225-237.
- [1] **Hoffman AJ. (1953)**;"Cycling in the simplex algorithm". Washington, DC: National Bureau of Standards.
- [6] **Gass S. I. (1985)**; Linear Programming, Methods and Applications. 5-th edition, McGraw-Hill Book Company.
- [9] **Gill P., Murray W. & Wright M. (1991)**; Numerical Algebra and Optimization. Addison-Wesley.
- [2] **Marshal K.T. &Suurballe J.W. (1969)**;"A note on cycling in the simplex method", Naval Research Logistics Quarterly; 16(1):121-37.
- [3] **Megiddo N. (1986)**;"A note on degeneracy in linear programming".Math.prog. 35,365-367.
- [12] **Price C.J. & Coop I.D. (2003)**;"Frames and Grids in Unconstraint and Linearly Constraint Optimization"; A no smooth approach.SIAMJ.Optim. , 14(2), 415-438.
- [4] **Robert J.V.(2008)**;"Linear programming, Foundation and extensions 3-rd edition". Springer Company.
- [13] **Solow D. (1984)**;"Linear Programming: An Introduction to finite improvement algorithms".Amestrdam: North-Holland.