SYSTEMS OPTIMIZATION LABORATORY
DEPARTMENT OF OPERATIONS RESEARCH
STANFORD UNIVERSITY
STANFORD, CALIFORNIA 94305-4022

Bracketing to Speed Convergence Illustrated on the von Newmann Algorithm for Finding a Feasible Solution to a Linear Program with a Convexity Constraint

by
George B. Dantzig

TECHNICAL REPORT SOL 92-6

October 1992

Research and reproduction of this report were partially supported by the National Science Foundation Grants DMS-8913089, ECS-8906260; the Department of Energy Grant DE-FG03-92ER5116 and the Electric Power Research Institute Grant RP8010-09.

Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do NOT necessarily reflect the views of the above sponsors.

Also issued as Operations Research Department Technical Report 92-15.

Bracketing to Speed Convergence Illustrated on the von Neumann Algorithm for Finding a Feasible Solution to a Linear Program with a Convexity Constraint

by George B. Dantzig
Stanford University

Abstract

Analogous to gunners firing trial shots to bracket a target in order to adjust direction and distance, we demonstate that it is sometimes faster not to apply an algorithm directly, but to roughly approximately solve several perturbations of the problem and then combine these rough approximations to get an exact solution. To find a feasible solution to an m-equation linear program with a convexity constraint, the von Neumann Algorithm generates a sequence of approximate solutions which converge very slowly to the right hand side b^o. However, it can be redirected so that in the first few iterations it is guaranteed to move rapidly towards the neighborhood of one of $m+1$ perturbed right hand sides $\hat{b}^i$, then redirected in turn to the next $\hat{b}^i$. Once within the neighborhood of each $\hat{b}^i$, a weighted sum of the approximate solutions $\bar{x}^i$ yields the exact solution of the unperturbed problem where the weights are found by solving a system of $m+1$ equations in $m+1$ unknowns. It is *assumed* an $r > 0$ is given for which the problem is feasible for all right hand sides b whose distance $\|b - b^o\|_2 \leq r$. The feasible solution is found in less than $4(m+1)^3/r^2$ iterations. The work per iteration is $\delta mn + 2m + n + 9$ multiplications plus $\delta mn + m + n + 9$ additions or comparisons where δ is the density of nonzero coeffients in the matrix.

Introduction:

To illustrate bracketing, we consider the problem of finding a feasible solution to the linear program with a convexity constraint, which (as explained in [1]) can be written after some algebraic manipulation in the form:

$$(1) \qquad \text{Find } x_j \geq 0, \quad \sum_1^n x_j = 1, \quad \sum_1^n P_j x_j = 0,$$

where P_j are m-vectors satisfying $||P_j|| = 1$ (the symbol $||\cdot||$ denotes $||\cdot||_2$). If feasible, von Neumann's "Center of Gravity" Algorithm generates a sequence of approximate solutions $x_j^t \geq 0, \quad \sum_1^n x_j^t = 1, \quad \sum_1^n P_j x_j^t = b^t$ with the property that $||b^t|| < \epsilon$ can be achieved in $t < 1/\epsilon^2$ iterations or given $t, ||b^t|| < 1/\sqrt{t}$, implying $b^t \to 0$ as $t \to \infty$.

A description of the von Neumann Algorithm and a proof by the author of its convergence properties can be found in a companion paper in this volume, [1]. It improves an approximation b^t by finding a P_s that minimizes $P_j^T b^t$; if $P_s^T b^t > 0$, the problem is infeasible and the algorithm terminates or else it generates b^{t+1} as the convex combination of b^t and P_s that is closest to the origin.

We *assume* that all perturbed problems of the form:

$$(2) \qquad x_j \geq 0, \quad \sum_1^n x_j = 1, \quad \sum_1^n P_j x_j = b,$$

are feasible for all $||b|| < r$ where $r > 0$ is given. The von Neumann Algorithm, when applied to a problem with $b \neq 0$, generates a sequence $t = 1, 2, \ldots$ of approximate solutions $x_j^t > 0, \quad \sum_1^n x_j^t = 1, \quad \sum_1^n P_j x_j^t = b^t$ with the property that $||b^t - b|| \leq \epsilon$ can be achieved in $\hat{t}$ iterations where

$$(3) \qquad \hat{t} - 1 \leq (1 + ||b||^2)/\epsilon^2$$

We apply this algorithm to $m + 1$ perturbed problems with right hand sides $b = \hat{b}^i$ that form the vertices of an equilateral simplex with the origin as center and $||\hat{b}^i|| = rm/(m+1)$. We define a *neighborhood* of $\hat{b}^i$ as a ball B_i of radius $1/(m+1)$ centered at $\hat{b}_i$; that is, all points b such that $||b - \hat{b}^i|| \leq 1/(m+1)$. We terminate the moves towards each $\hat{b}^i$ as soon as a

$b^t = \bar{b}^i$ has reached the neighborhood B_i of $\hat{b}^i$. In other words, when

$$\|\bar{b}^i - \hat{b}^i\| \leq r/(m+1) \text{ for some } b^t = \bar{b}^i \tag{4}$$

Noting $\|\hat{b}^i\| = rm/(m+1)$, each of these neighborhoods of $\hat{b}^i$ can be reached using the von Neumann algorithm in less than

$$[1 + rm/(m+1)]^2/\quad [r/(m+1)]^2 \text{ iterations .} \tag{5}$$

Noting $r \leq 1$ the total number of iterations required to reach all $m+1$ neighborhoods is less than

$$[1 + rm/(m+1)]^2(m+1)^3/r^2 < 4(m+1)^3/r^2. \tag{6}$$

The final step is to generate the feasible solution $\bar{x} \geq 0$, $\sum_1^n \bar{x}_j = 1$, $\sum_1^n P_j \bar{x} = 0$ by finding weights $\bar{\lambda}_i > 0$ such that $\bar{x} = \sum \lambda_i \bar{x}^i$. These weights $\bar{\lambda} = (\bar{\lambda}_1, \bar{\lambda}_2, \ldots, \bar{\lambda}_{m+1})$ are found by solving the $(m+1) \times (m+1)$ system

$$\sum \bar{b}^i \bar{\lambda}_i \quad = 0 \tag{7}$$

$$\sum \bar{\lambda}_i \quad = 1. \tag{8}$$

We will now prove that this system has a unique solution $\bar{\lambda} > 0$.

The geometry depicted in the figure is the m-space of the columns. Since $\|P_j\| = 1$, all points P_j lie on the surface of the m-dimensional hyperspace S_1 of unit radius with center at the origin. We are given r the radius of a concentric hypersphere $S_r \subseteq S_1$ such that S_r lies in the convex hull of the points P_j. Thus r is a measure of how deeply the origin is embedded in the set of b such that $b = \sum P_j x_j$, $x_j \geq 0$, $\sum x_j = 1$ is feasible.

To generate the $m+1$ different *finite* sequences (x^t, b^t) whose b^t approach $m+1$ different balls B_i, their centers $\hat{b}^i$ are prechosen. These can be the vertices of any simplex lying in the set of feasible b that contains the origin as an interior point. We chose $\hat{b}^i$ to be the vertices of an $(m+1)$ *equilateral simplex* whose center is the origin and whose vertices are located at distances $rm/(m+1)$ from the origin; the coordinates of $\hat{b}^i$ might be chosen as follows:

$$
(9)\qquad
\begin{array}{lcccccc}
\hat{b}^{m+1} & = & [\ 0 & 0 & \dots & 0 & ma_m]^T \\
\hat{b}^{m} & = & [\ 0 & 0 & \dots & (m-1)a_{m-1} & -a_m]^T \\
\vdots & \vdots & & \vdots & \vdots & \vdots & \vdots \\
\hat{b}^{3} & = & [\ 0 & +2a_2 & \dots & -a_{m-1} & -a_m]^T \\
\hat{b}^{2} & = & [\ a_1 & -a_2 & \dots & -a_{m-1} & -a_m]^T \\
\hat{b}^{1} & = & [\ -a_1 & -a_2 & \dots & -a_{m-1} & -a_m]^T
\end{array}
$$

where $a_i = r\sqrt{\frac{m}{m+1}} \cdot \sqrt{\frac{1}{i(i+1)}}$.

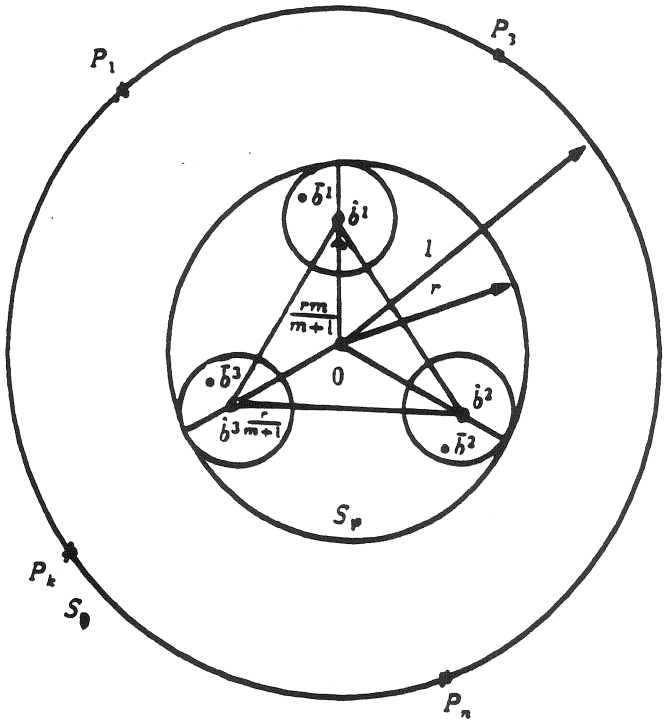

Figure: The Iterations Converge to $\hat{b}^i$ Instead of the Origin 0.

Existence of Separating Hyperplanes:

Let $y = (y_1, y_2, \ldots, y_m)$represent a general point in R^m. The equation of any hyperplane through the origin has the form $a^T y = 0$. This hyperplane is said to *separate* y^1 from y^2 if $a^T y^1$ and $a^T y^2$ are of opposite signs.

Fact 1. Each hyperplane $(\hat{b}^i)^T y = 0$ for $i = 1, 2, \ldots, m$ separates any point in the ball B_i centered at $\hat{b}^i$ from any point lying in any of the other balls B_j centered at $\hat{b}^j$.

Proof: Because of the $m+1$-fold symmetry of the equilateral simplex it is sufficient to demonstrate that the hyperplane $(\hat{b}^{m+1})^T y = 0$ separates $\bar{b}^{m+1}$ from $\bar{b}^m$ where $||\bar{b}^{m+1} - \hat{b}^{m+1}|| < r/(m+1)$ and $||\bar{b}^m - \hat{b}^m|| < r/(m+1)$. Substituting $\hat{b}^{m+1} = (0, \ldots, rm/(m+1))^T$ from (9) into $(\hat{b}^{m+1})^T y = 0$, the hyperplane reduces to $U_m^T y = 0$ where $U_m^T = (0, \ldots, 1)$. Letting $\bar{b}^{m+1} = \hat{b}^{m+1} + u$, where $||u|| < r/(m+1)$, we have

$$U_m^T \bar{b}^{m+1} = \bar{b}_m^{m+1} = \hat{b}_m^{m+1} + u > rm/(m+1) - r/(m+1) > 0$$

Letting $\bar{b}^m = \hat{b}^m + v$, where $\hat{b}^m = (0, 0, \ldots, r\sqrt{m-1}/\sqrt{m+1}, -r/(m+1))^T$ and $||v|| < r/(m+1)$, we have

$$U_m^T \bar{b}_m = \bar{b}_m^m = \hat{b}_m^m + v_m < -r/(m+1) + r/(m+1) = 0$$

Thus $U_m^T \bar{b}^{m+1}$ and $U_m^T \bar{b}^m$ have opposite signs and so the hyperplane $U_m^T y = 0$ separates $\bar{b}^{m+1}$ from $\bar{b}^m$. ■

The Separating Hyperplanes Theorem below states conditions which imply that the points $\bar{b}^1, \bar{b}^2, \ldots, \bar{b}^{m+1}$ are the vertices of a simplex containing the origin in its interior. That these conditions are satisfied follows from Fact 1.

Separating Hyperplanes Theorem: *Given that $\hat{b}^1, \hat{b}^2, \ldots, \hat{b}^{m+1}$ are the vertices of an m-dimensional Simplex $\hat{T}$ containing the origin in its interior; given that $a^i y = 0$ for $i = 1, 2, \ldots, m+1$ are the equations of $m+1$ hyperplanes separating $\hat{b}^i$ from $\hat{b}^j$ for all $j \neq i$; and given any $m+1$ points $\bar{b}^1, \bar{b}^2, \ldots, \bar{b}^{m+1}$ such that each hyperplane $a^i y = 0$ separates $\bar{b}^i$ (on the same side as $\hat{b}^i$) from $\bar{b}^j$ for all $j \neq i$; then $\bar{b}^1, \bar{b}^2, \ldots, \bar{b}^{m+1}$ are the vertices $\bar{T}$ of an m-dimensional simplex that contains the origin as an interior point.*

Proof: Since the simplex associated with $\hat{T}$ contains the origin, we know there exist $\hat{\lambda}_i \geq 0, \bar{\lambda}_i \geq 0$ such that

$$\sum \hat{b}^i \hat{\lambda}_j \quad + \sum \bar{b}^i \bar{\lambda}_i \quad = 0 \tag{10}$$

$$\sum \hat{\lambda}_i \quad + \sum \bar{\lambda}_i \quad = 1. \tag{11}$$

Before continuing with the proof, we show two more facts:

Fact 2. If $(\hat{\lambda}, \bar{\lambda}) \geq 0$ is a feasible solution to (10), (11), then $\hat{\lambda}_i + \bar{\lambda}_i > 0$ for all i.

Suppose, on the contrary, $\hat{\lambda}_k = 0, \bar{\lambda}_k = 0$ for some k. Multiply (10) on the left by a^k; recall, by assumption, $a^k \hat{b}^i < 0$ and $a^k \bar{b}^i < 0$ for all $i \neq k$. We have, after dropping the $i = k$ terms which are zero-valued:

$$\sum_{i \neq k} (a^k \hat{b}^i) \hat{\lambda}_i \quad + \sum_{i \neq k} (a^k \bar{b}^i) \bar{\lambda}_i \quad = 0, \quad \hat{\lambda}_i \geq 0, \bar{\lambda}_i \geq 0, \tag{12}$$

$$\sum_{i \neq k} \hat{\lambda}_i \quad + \sum_{i \neq k} \bar{\lambda}_i \quad = 1, \tag{13}$$

implying, that (12) is the sum of non-positive terms (not all zero by (13)), a contradiction. ■

Fact 3. If T is any simplex containing the origin whose vertices i are separated from the remaining vertices $j \neq i$ by a hyperplane $a^i y = 0$ for each i, then T contains the origin strictly in its interior.

Fact 3 follows from Fact 2 by setting $\hat{b}^i = \bar{b}^i$ for all i. ■

Continuing with the proof of the separating hyperplanes theorem, define B and U_{m+1} by

$$B = \begin{pmatrix} \hat{b}^1 & \hat{b}^2 & \dots & \hat{b}^{m+1} \\ 1 & 1 & & 1 \end{pmatrix}, \quad U_{m+1} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

Since $\hat{T}$ are the vertices of an m-dimensional simplex, B is non-singular, $B\hat{\lambda} = U_{m+1}$ can be solved for $\hat{\lambda}$ and, when solved, $\hat{\lambda} > 0$ by hypothesis.

We view B as a feasible non-degenerate basis and consider $\begin{pmatrix} \bar{b}^1 \\ 1 \end{pmatrix}$ as an incoming non-basic column. We assert it will replace $\begin{pmatrix} \hat{b}^1 \\ 1 \end{pmatrix}$ in the basis because, on the contrary, if it replaced some column $k \neq 1$ in the basis, it would imply after the replacement that both $\bar{\lambda}_k$ and $\hat{\lambda}_k$ are 0 in a feasible solution, contrary to Fact 2. By replacing in turn basis columns $\begin{pmatrix} \bar{b}^2 \\ 1 \end{pmatrix}$ by $\begin{pmatrix} \hat{b}^2 \\ 1 \end{pmatrix}$, $\begin{pmatrix} \hat{b}^3 \\ 1 \end{pmatrix}$ by $\begin{pmatrix} \bar{b}^3 \\ 1 \end{pmatrix}$, etc., we arrive at the conclusion that $\bar{T}$ are the vertices of a simplex containing the origin. It then follows from Fact 3 that this simplex contains the origin as a strictly interior point. ■

Conclusion

We sometimes can do better by first firing trial shots which bracket the target and then apply a final correction. The von Neumann Algorithm is an infinite one with poor convergence for finding a feasible solution to an m-equation linear program with a convexity constraint. Given that the problem is feasible for all perturbations of the right hand side within a radius r, we showed how to convert the method into a finite one by applying it to finding very rough approximate solutions x^i to $m+1$ perturbations $\hat{b}^i$ of the right hand side. We do this by generating $m+1$ sequences that stop when they reach the neighborhood of each of the $m+1$ perturbations $\hat{b}^i$. We then solve $m+1$ equations in $m+1$ unknowns λ_i and apply these weights λ_i to the $m+1$ approximate solutions x^i to obtain an exact feasible solution to the unperturbed problem.

The feasiblity problem is solved by this procedure in less than $f^2(m+1)^3/r^2$ iterations where $f = 1 + rm/(m+1) < 2$; the work per iteration is $\delta mn + 2m + n + 9$ multiplications and $\delta mn + m + n + 9$ additions or comparisons; δ is the density of nonzeros.

Is the algorithm practical? The purpose of this paper was not to find a method competitive to the simplex method, [2], or to an interior method,

[3]; it was to illustrate the concept of bracketing by showing how it can be used to obtain a low-order polynomial bound on the number of arithmetic operations if we are given a tiny bit of extra information (such as the radius of feasibility r). Nevertheless, I was curious to see how efficient it would be in solving a suite of small test problems.

The algorithm was programmed by Elaine Chew in partial fulfillment of her undergraduate honors thesis at Stanford University. In practice, of course, we don't know the value of r, but quite often practical problems are not tight, but less often a problem could be infeasible or just barely feasible. What Elaine Chew did was to adaptively choose r, starting optimistically with r close to 1. If it turned out that one of the $\hat{b}^i$ neighborhoods was not feasible, the value of r was replaced by $r/2$ and the process repeated. She also tried out a number of heuristics. For problems not very tight, her procedure worked very well. If the problem was infeasible or just barely feasible, the method failed. If infeasible the method will not detect it but would be able to if an extra $\hat{b}^i = 0$ were included with a neighborhood radius $r/(m+1)$. The material that follows was authored by Elaine Chew.

Test Results of Elaine Chew

To test the practicality of the algorithm, a program was written in Pascal, run on a SPARCstation, and tested on 47 files of data in the form of matrices ranging from size 3x4 to 28x40. Right hand side values were chosen to guarantee the existence of a feasible solution.

The first guess overstated the radius of feasibility; r was taken to be $(m+1)/m$, where m is the number of rows in the given matrix, to allow for the largest possible balls, B_i to iterate toward. The first approximation to b^1 was taken to be the P_j closest to $\hat{b}^1$. After getting the approximations to the $m+1$ vertices in this manner, regardless of whether or not these approximations fell within the balls B_i, we solved (7) and (8) for the origin whenever possible. These approximations often were sufficiently well located within the unit hypersphere to cover the origin, and thus a convex combination was immediately found. When no convex combination of these approximations covered the origin, we then replaced r by $rm/(m+1)$, and iteratively tried again.

The following table is a summary of the test results. The definitions for the column headings are:

m	=	number of rows in the given matrix
n	=	number of columns in the given matrix
min $\|\|\bar{b}^i\|\|$	=	closest $\bar{b}^i$ to the origin
q	=	number of multiplications/divisions done
fn	=	number of function calls (sqrt, arctan)
lin_sys	=	number of linear systems solved
VN_itn	=	number of von Neumann iterations
Soln	=	Yes, if a feasible solution found, else No[1].

Problem	**m**	**n**	**min$\|\|\bar{b}^i\|\|$**	**q**	**fn**	**lin_sys**	**VN_itn**	**Soln**
p 1.dat	3	4	0.7217	326	36	1	5	Yes
p 2.dat	3	5	0.5280	447	44	1	7	Yes
p 3.dat	3	5	0.4899	447	44	1	7	Yes
p 4.dat	3	5	0.3316	462	44	1	8	Yes
p 5.dat	3	5	0.8150	377	44	1	5	Yes
p 6.dat	3	5	0.2074	841	74	2	13	Yes
p 7.dat	3	5	0.6107	497	44	1	9	Yes
p 8.dat	3	6	0.6602	467	52	1	6	Yes
p 9.dat	3	6	0.6557	431	52	1	4	Yes
p10.dat	3	6	0.7365	431	52	1	4	Yes
p11.dat	3	6	0.5701	563	52	1	9	Yes
p12.dat	3	8	0.4800	553	68	1	5	Yes
p13.dat	3	10	1.0000	432	84	1	0	Yes
p14.dat	4	6	0.3312	1832	126	2	24	Yes
p15.dat	4	7	0.1196	3250	202	4	40	Yes
p16.dat	4	7	0.4172	829	76	1	8	Yes

[1]When Soln=No, it means that the algorithm was terminated when r became < 0.05. Hence the problem was feasible but had a radius of feasibility less than 0.05.

Problem	m	n	$\min\|\|\bar{b}^i\|\|$	q	fn	lin_sys	VN_itn	Soln
p18.dat	4	8	0.1949	1780	134	3	13	Yes
p19.dat	4	11	0.8033	1152	116	1	7	Yes
p20.dat	5	9	0.0694	12455	530	18	76	No
p21.dat	6	11	0.0092	30728	1264	9	211	Yes
p22.dat	6	13	0.4947	2854	192	1	14	Yes
p23.dat	7	9	0.0267	55900	1362	10	426	Yes
p24.dat	7	11	0.0016	77610	1750	24	452	No
p25.dat	7	11	0.2901	9415	540	3	47	Yes
p26.dat	7	12	0.5221	3385	204	1	15	Yes
p27.dat	7	12	0.7276	3135	204	1	15	Yes
p28.dat	12	19	0.6376	30481	1010	2	63	Yes
p29.dat	6	10	0.3995	4898	290	2	31	Yes
p30.dat	7	11	0.3403	6873	364	2	35	Yes
p31.dat	8	12	0.3670	9126	446	2	38	Yes
p32.dat	9	13	0.2231	39131	1316	5	157	Yes
p33.dat	10	14	0.1716	50547	1558	5	175	Yes
p34.dat	11	15	0.2096	61738	1820	5	180	Yes
p35.dat	12	16	0.3147	72086	2102	5	181	Yes
p36.dat	13	17	0.3351	87418	2404	5	192	Yes
p37.dat	14	18	0.5934	53701	1646	3	97	Yes
p38.dat	15	19	0.3689	152351	3676	6	271	Yes
p39.dat	16	20	0.1194	314973	6110	9	538	Yes
p40.dat	24	32	0.6459	322765	6446	4	199	Yes
p41.dat	14	30	0.1373	321894	7826	10	435	Yes
p42.dat	28	40	0.1104	4108412	51094	25	2233	Yes
p43.dat	24	36	0.0012	382965219	95446	75	363401	No
p44.dat	9	21	0.1835	74014	2788	7	202	Yes
p45.dat	12	30	0.1167	277960	6382	9	472	Yes
p46.dat	5	11	0.5935	1785	140	1	11	Yes
p47.dat	14	22	0.3773	117247	3326	5	191	Yes
p48.dat	21	33	0.6758	238514	5848	4	160	Yes

In conclusion, we have found on small test problems that the method works well for most problems, yielding a solution with relatively few von Neumann steps. However, when the radius of feasibility is very small and the m dimension high,. as in the case of problem p43.dat, the iterations converged very slowly, because the algorithm became practically the same as the original von Neumann method. Note that the algorithm worked best for problems with $n >> m$. Apparently, for these test problems, the points P_j on the hypersphere S_1 were (in most cases) more evenly distributed, and therefore had a large radius of feasibility, and thus required very few von Neumann iterations.

References

[1] Dantzig, George B., "An ϵ-precise feasible solution to a linear program with a convexity constraint in $1/\epsilon^2$ iterations independent of problem size", 1992 Mathematical Programming, Series B in honor of Philip Wolfe.

[2] Dantzig, George B., in chapters on the Simplex Method, *Linear Programming and Extentions*, Princeton University Press (1963).

[3] Karmarkar, N., "A new polynomial algorithm for linear programming", *Combinatorica* 4 (1984) 373-395.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	October 1992	Technical Report

4. TITLE AND SUBTITLE
Bracketing to Speed Convergence Illustrated on the von Newmann Algorithm for Finding a Feasible Solution to a Linear Program with a Convexity Constraint

5. FUNDING NUMBERS
DE-FG03-92ER5116

6. AUTHOR(S)
GEORGE B. DANTZIG

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Department of Operations Research - SOL
Stanford University
Stanford, CA 94305-4022

8. PERFORMING ORGANIZATION REPORT NUMBER
1111MA

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Office of Energy Research
U.S, Department of Energy
Washington, DC 20585

10. SPONSORING / MONITORING AGENCY REPORT NUMBER
SOL 92-6

12a. DISTRIBUTION / AVAILABILITY STATEMENT
UNLIMITED

12b. DISTRIBUTION CODE
UL

13. ABSTRACT *(Maximum 200 words)*

Please see other side for ABSTRACT.

14. SUBJECT TERMS
Linear Program; Speeding Convergence; von Newmann Algorithm

15. NUMBER OF PAGES
11 pp.

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED			SAR

Abstract

Analogous to gunners firing trial shots to bracket a target in order to adjust direction and distance, we demonstate that it is sometimes faster not to apply an algorithm directly, but to roughly approximately solve several perturbations of the problem and then combine these rough approximations to get an exact solution. To find a feasible solution to an m-equation linear program with a convexity constraint, the von Neumann Algorithm generates a sequence of approximate solutions which converge very slowly to the right hand side b^o. However, it can be redirected so that in the first few iterations it is guaranteed to move rapidly towards the neighborhood of one of $m+1$ perturbed right hand sides $\hat{b}^i$, then redirected in turn to the next $\hat{b}^i$. Once within the neighborhood of each $\hat{b}^i$, a weighted sum of the approximate solutions $\bar{x}^i$ yields the exact solution of the unperturbed problem where the weights are found by solving a system of $m+1$ equations in $m+1$ unknowns. It is *assumed* an $r > 0$ is given for which the problem is feasible for all right hand sides b whose distance $\|b - b^o\|_2 \leq r$. The feasible solution is found in less than $4(m+1)^3/r^2$ iterations. The work per iteration is $\delta mn + 2m + n + 9$ multiplications plus $\delta mn + m + n + 9$ additions or comparisons where δ is the density of nonzero coeffients in the matrix.