March 1991
TECHNICAL REPORT SOL 91-5
George B. Dantzig

Polyomomially Bounded Algorithm
into a
Converting a Convergence Algorithm

STANFORD, CALIFORNIA 94305-4032
STANFORD UNIVERSITY
DEPARTMENT OF OPERATIONS RESEARCH
SYSTEMS OPTIMIZATION LABORATORY
An approximate solution will be generated with
\[ d > \| q - f \| \]
When applied to a perturbed problem \( q = q^\prime \neq 0 \), we will show that in
iterations \( d > \| q \| \), \( q = q^\prime \) will be generated with
\[ d > \| q \| \]
In approximations an approximate solution \( q = q^\prime \) will be produced.
Phases I Linear Program with a convexity constraint. We will reproduce this proof that in
Von Neumann in 1948 proposed the first interior algorithm for solving a general
zero coefficient density.

Each iteration consists of \( m + \eta \) multiplications and additions, where \( \eta \) is the non-
exact solution to the unapproximated problem with \( q = q^\prime \). In \( I + \eta \) iterations,
we obtain an approximate solution with right-hand sides \( q, q^\prime \). To observe an
approximate solution in the limit as \( \eta \to 0 \), we assume that all perturbed problems
are feasible for all \( \eta > 0 \). We apply the algorithm
\[ I = \tilde{I} \]
of approximate solutions
where \( \tilde{I} \) are m-vectors satisfying
\[ I \]

Find: \( \tilde{I} \)

Abstract: We consider the General Phase I Linear Programming problem with a
convexity constraint which can be written after some algebraic manipulation in the
form:

George B. Danzig

By

Polynomially Bounded Algorithm

into a

Converting a Converging Algorithm
Figure 1. The Iterations Converge to \( \bar{a} \) Instead of the Origin 0.

\[
\begin{pmatrix}
1 & \ldots & 1 \\
1 & \ldots & 1 \\
\vdots & \ddots & \vdots \\
1 & \ldots & 1
\end{pmatrix}
\begin{pmatrix}
w_p \\
w_p \\
\vdots \\
w_p
\end{pmatrix}
\begin{pmatrix}
1 \\
1 \\
\vdots \\
1
\end{pmatrix}
= 
\begin{pmatrix}
\bar{a} \\
\bar{a} \\
\vdots \\
\bar{a}
\end{pmatrix}
\]

where \( \bar{a} \triangleq \frac{(1 + w)I}{1 + w} \).

The coordinates of \( q \)'s may be chosen as follows:

- The vertices of the \( 1 \) dimensional simplex whose center is the origin and whose vertices are located at distance \( \frac{(1 + w)I}{1 + w} \) from the origin. For example, the vertices of an \( n \) dimensional simplex whose center is the origin and whose vertices lie in the set of feasible \( q \)'s that contain the origin as an interior point. We choose \( q \) to be different points. The vertices of any simplex \( x \) whose \( q \) approach \( 1 + w \).

To generate the \( m + 1 \) different finite sequences \((q', x', q)\) whose \( q \) approach \( 1 + w \).

\[
I = \begin{pmatrix}
x \\
x \\
\vdots \\
x
\end{pmatrix}
\begin{pmatrix}
0 \\
0 \\
\vdots \\
0
\end{pmatrix}
\begin{pmatrix}
x \\
x \\
\vdots \\
x
\end{pmatrix}
= q
\]

where \( \bar{a} \) is feasible.
We now describe the detailed steps of von Neumann's algorithm for finding an approximate solution to a particular problem. We denote the detailed steps of von Neumann's algorithm. For finding an approximate solution, we prove that this system has a unique solution $x$. We will denote the system by $$(I + w)x = x.$$ We will denote the weights by $\lambda, \mu$. The final step is to generate the feasible solution $x$ to the phase I problem by iteration. The final step is to generate the feasible solution $x$. The final step is to generate the feasible solution $x$. We denote the weights by $\lambda, \mu$. The final step is to generate the feasible solution $x$ to the phase I problem by iteration.

\begin{align}
&\frac{(I + w)x}{\mu} = d, \\ &\frac{\lambda}{\mu}(I + w)x = \frac{\lambda d}{(I + w)\lambda} > 0.
\end{align}

When the sequence $(\lambda, \mu)$ reaches a point, the sequence $(\lambda, \mu)$ converges towards $(\theta, \mu)$.
\[ t > \left( \frac{\varepsilon}{t \sin \theta} \right) + \left( \frac{1-t}{t} \right) \]

Therefore, for $t = 2, \varepsilon, \ldots$, the norm of the hyperplane is 2, it follows that $\varepsilon$ and

\[ \frac{\varepsilon}{t \sin \theta} \sin \theta + \frac{1-t}{t} \sin \theta = \left( \frac{\|q - f\|}{t \sin \theta} \right) + \left( \frac{1-t}{t} \right) \]

Therefore, noticing $\frac{2}{\pi} \leq \theta = \varepsilon + 1 \theta$ sin $\sin 1 - \frac{1}{t \sin \theta} = \frac{1}{t \sin \theta}$

Then

\[ \|q - f\| = \frac{1}{t \sin \theta} \text{ and } \|q - 1-\theta\| = \frac{1}{t \sin \theta} \]

Let $S$ be the set of feasible $q$ defined as $S = \{q_1, q_2, \ldots \}$, where $q_1$ is the origin in the set of feasible $q$. By construction, all points located at a distance of less than all points $q_1$ then all points $q_2 < \theta$, contrary to our assumption that all points lie inside the convex hull of the feasible $q$'s, contrary to our assumption that all points $q_2 < \theta$. For this reason, we would like to place the one side of the hyperplane through $q_1$, contrary to our assumption that all points $q_2 < \theta$. In order to determine the rate of convergence, note that, because of the following:

\[ \frac{\|1-\theta - q\|}{\|1-\theta - q\|} \frac{\|1-\theta - q\|}{\|1-\theta - q\|} = \cos \theta \cos \cos \theta \cos \cos \theta = \frac{1}{t \sin \theta} \]

Where $t$ is the unit $n$-vector with $1$ in component $n$ and $\theta$ are computed by

\[ (2 \theta \cos + 2 \theta \cos)/(2 \theta \cos + 1 - r \theta \cos + 2 \theta \cos) = r \cos \]

\[ (2 \theta \cos + 2 \theta \cos)/(2 \theta \cos + 1 - r \theta \cos + 2 \theta \cos) = r \cos \]

Is proportional to $\cos \theta$ and $\cos \theta$, respectively. It is clear that $H$ is a weighted convex combination of $q$ and $f$ with weights $W$. From the result of Proposition 3.1, it follows that $H$ will be labeled $\text{ABC}$, The next approximation point $q_1 - f$. The triangles $H$ are the same as the same $H$ and $f$. This can be carried out in $m + n$. The only non-zero vector with $1$ is $q - f$. Then $H$ is preprocessed.

\[ \|q - f\| = \sqrt{\|q - f\|^2} \]

\[ \text{ARCMAX} = s \]

For instance, Figure 2, $f$ is selected as that $f$ makes the shortest angle with direction $q_1 - f$.
0 = \hat{h}_1^w (1 + w q) = \hat{h}_1 (i + 1 + w q)/\mu = \hat{h}_1 (1 + w q)/\mu > \|w q - \bar{w}\| \quad \text{and} \quad \hat{h}_1 (1 + w q)/\mu > \|1 + w q - 1 + w q\|

where \(\bar{w}\) is the coordinate of the hyperplane that separates the \(m + 1\) fold symmetric of the equilateral simplex. It is sufficient to demonstrate that the hyperplane defined by \(\bar{q}\) from any point lying in any of the other \(d\)-balls centered at \(q\) and \(\alpha\) are of opposite signs.

**Fact 1.** Each hyperplane is said to separate \(q\) from \(\bar{q}\), if \(\alpha q - \alpha \bar{q}\) and \(\alpha > 0\).

Let the equation of any hyperplane through the origin be the form \(\alpha q = \hat{h}_1 q\), represent a general solution \(x_1, x_2, \ldots, x_n\) of system (3). The equation of any hyperplane through the origin is \(\alpha q = \hat{h}_1 q\), and the condition that there are no solutions \(x_1, x_2, \ldots, x_n\) of the form \(q = \hat{h}_1 q\), the system of linear equations (1 + w q) and \(\hat{h}_1 q = \hat{h}_1 q\) cannot be solved. Then the number of iterations to show is that the solution \(x_1, x_2, \ldots, x_n\) can be solved, that this number of iterations is unique, and that \(x_1, x_2, \ldots, x_n\) are of opposite signs.

**Existence of Separating Hyperplanes**

We conclude that \(\frac{\beta}{\xi} > 1\). We have

Summing the above, canceling terms common to both sides of the sum, recalling

\[\varepsilon - (\frac{\xi}{\beta}) > (\frac{\xi}{\beta}) + \varepsilon (\frac{\xi}{\beta})\]

\[\vdots \]

\[\varepsilon (1 - r) > (\frac{\xi}{\beta}) + \varepsilon (1 - r)\]

\[\varepsilon (\frac{\xi}{\beta}) > (\frac{\xi}{\beta}) + \varepsilon (1 - r)\]

Dividing (8) through by \(\frac{\xi}{\beta}\) for \(r = 2, 3, \ldots\), for \(r = \xi\),

the idea is that \(\frac{\xi}{\beta}\) can be altered in less that \(\frac{\xi}{\beta}\) iterations (instead of less than \(\frac{\xi}{\beta}\) iterations).
the origin strictly in its interior. From the remaining vertices \( \not\in \mathcal{L} \) by a hyperplane \( \mathcal{L} = \{0\} \) for each \( i \), then \( \mathcal{L} \) contains \( i \), if \( \not\in \mathcal{L} \). If \( \not\in \mathcal{L} \) is any simplex containing the origin whose vertices are separated.

**Fact 3.** If \( \mathcal{L} \) is a feasible solution to (13.1), then \( \lambda < 0 \) for all \( \lambda \).

**Proof:** Since the simplex associated with \( \mathcal{L} \) contains the origin, we know there exist vertices \( \mathcal{L} \) of an \( m \)-dimensional simplex that contains the origin and an interior point.

Separating Hyperplane Theorem: Given (1) there are any \( r \) vectors \( \mathcal{L} \) that \( \not\in \mathcal{L} \). For all \( \lambda \not\in \mathcal{L} \) are the \( \lambda \). In this case \( \not\in \mathcal{L} \) are the \( \lambda < 0 \) for all \( \lambda < 0 \) of an \( m \)-dimensional simplex. Vertices of an \( m \)-dimensional simplex that contains the origin.

Separating Hyperplane Theorem: Given (1) there are any \( r \) vectors \( \mathcal{L} \) that \( \not\in \mathcal{L} \). For all \( \lambda \not\in \mathcal{L} \) are the \( \lambda < 0 \) for all \( \lambda < 0 \) of an \( m \)-dimensional simplex. Vertices of an \( m \)-dimensional simplex that contains the origin.

Separating Hyperplane Theorem: Given (1) there are any \( r \) vectors \( \mathcal{L} \) that \( \not\in \mathcal{L} \). For all \( \lambda \not\in \mathcal{L} \) are the \( \lambda < 0 \) for all \( \lambda < 0 \) of an \( m \)-dimensional simplex. Vertices of an \( m \)-dimensional simplex that contains the origin.

Separating Hyperplane Theorem: Given (1) there are any \( r \) vectors \( \mathcal{L} \) that \( \not\in \mathcal{L} \). For all \( \lambda \not\in \mathcal{L} \) are the \( \lambda < 0 \) for all \( \lambda < 0 \) of an \( m \)-dimensional simplex. Vertices of an \( m \)-dimensional simplex that contains the origin.

Separating Hyperplane Theorem: Given (1) there are any \( r \) vectors \( \mathcal{L} \) that \( \not\in \mathcal{L} \). For all \( \lambda \not\in \mathcal{L} \) are the \( \lambda < 0 \) for all \( \lambda < 0 \) of an \( m \)-dimensional simplex. Vertices of an \( m \)-dimensional simplex that contains the origin.

Separating Hyperplane Theorem: Given (1) there are any \( r \) vectors \( \mathcal{L} \) that \( \not\in \mathcal{L} \). For all \( \lambda \not\in \mathcal{L} \) are the \( \lambda < 0 \) for all \( \lambda < 0 \) of an \( m \)-dimensional simplex. Vertices of an \( m \)-dimensional simplex that contains the origin.

Separating Hyperplane Theorem: Given (1) there are any \( r \) vectors \( \mathcal{L} \) that \( \not\in \mathcal{L} \). For all \( \lambda \not\in \mathcal{L} \) are the \( \lambda < 0 \) for all \( \lambda < 0 \) of an \( m \)-dimensional simplex. Vertices of an \( m \)-dimensional simplex that contains the origin.

Separating Hyperplane Theorem: Given (1) there are any \( r \) vectors \( \mathcal{L} \) that \( \not\in \mathcal{L} \). For all \( \lambda \not\in \mathcal{L} \) are the \( \lambda < 0 \) for all \( \lambda < 0 \) of an \( m \)-dimensional simplex. Vertices of an \( m \)-dimensional simplex that contains the origin.

Separating Hyperplane Theorem: Given (1) there are any \( r \) vectors \( \mathcal{L} \) that \( \not\in \mathcal{L} \). For all \( \lambda \not\in \mathcal{L} \) are the \( \lambda < 0 \) for all \( \lambda < 0 \) of an \( m \)-dimensional simplex. Vertices of an \( m \)-dimensional simplex that contains the origin.
Fact 3 follows from Fact 2 by setting \( \hat{b}_i = \hat{b}_i \) for all \( i \).

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

\[
\begin{bmatrix}
\hat{b}_1 \\
\hat{b}_2 \\
\vdots \\
\hat{b}_{m+1}
\end{bmatrix} =
\begin{bmatrix}
0 \\
1 \\
1 \\
\vdots \\
1
\end{bmatrix}.
\]

Since \( T \) is the vertices of an \( m \)-dimensional simplex by assumption, it means that \( B \) is non-singular and that \( B \cdot U_{m+1} \) can be solved for \( \lambda \) and, when solved, \( \lambda \geq 0 \). From Fact 3 it follows that \( \hat{b}_i > 0 \). We view \( B \) as a feasible non-degenerate basis and consider \( \hat{b} \) as an incoming non-basic column. We assert it will replace \( \bar{b} \) 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 \( \hat{\lambda}_k \) and \( \hat{\lambda}_1 \) are \( \hat{\lambda} > 0 \). We replace \( \hat{b} \) in the basis by \( \hat{b} \), etc., we arrive at the conclusion that \( T \) is 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.

This completes the proof that the \( (m+1) \) sequences converge to \( m+1 \) points \( \bar{b} \) in less than \( 4(m+1)^3 \lambda^2 \) iterations. By applying the weights \( \hat{\lambda} \geq 0 \) to the corresponding \( \bar{b} \), we generate the exact solution \( x \) to the Phase 1 linear program.

One final remark: Just because an algorithm is polynomial does not necessarily make it practical. The von Neumann algorithm has a poor convergence rate. Like the simplex method each of its iterations requires about \( m \cdot \delta \) iterations, where \( \delta \) is the density of non-zero coefficients. When applied to \( (m+1) \) perturbed problems as we do in this paper, we obtain an upper bound of \( 4(m+1)^3 \lambda^2 \) iterations where \( 0 < \lambda < 1 \). The moral of this tale is that, like gunners, we may do better by first bracketing the target and then applying a final correction.
Linear Programming: Polynomial Algorithm: Phase I

The abstract contains: Each iteration consists of \( m(n + 3g) \) multiplications and additions where \( g \) is the non-zero coefficients. Every iteration requires an exact solution to the linear programming problem with \( g = 0 \) in \( f \). If \( f \) is positive, we apply this algorithm to \( m + 1 \) perturbed problems with right hand sides \( q_i = \epsilon \) where \( \epsilon > 0 \). We choose a feasible solution to the phase 1 problem. We assume that all perturbed solutions to the linear programming problems converge when \( j \) is large enough. A sequence \( \epsilon_i \) is applied to various solutions \( x_i \). If \( \epsilon_i \) is small, the solution is feasible. From Newton's Center of Gravity Algorithm we have:

\[
\begin{align*}
\text{Find } x > 0. \\
\epsilon_i = f \sum x_i \\
q_i = f \sum \frac{1}{u_i} \\
\end{align*}
\]

We consider the general phase I linear programming problem with a constraint constraint which can be written after some algebraic manipulation in the form:

We consider the general phase I linear programming problem with a constraint constraint which can

SOL 91-5

Arlington, VA 22217

6. AUTHOR(S)

George D. Bunting

7. SPONSORING ORGANIZATION

Office of Naval Research - Department of the Navy

8. SPONSORING ORGANIZATION MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Naval University, 94905-4022

9. SPONSORING MONITORING AGENCY REPORT NUMBER

Department of Operations Research - SOL

10. SPONSORING MONITORING AGENCY NAME(S) AND ADDRESS(ES)

ONR - N0014-89-J-1599

11. TITLE AND SUBTITLE

Converting a Converging Algorithm into a Polynomially Bounded Algorithm

12. DISTRIBUTION, AVAILABILITY, AND SECURITY STATEMENT

UL

UNLIMITED

13. ABSTRACT (Maximum 200 Words)

A polynomially bounded algorithm for linear programming is presented. The algorithm is based on Newton's Center of Gravity Algorithm. Each iteration consists of \( m(n + 3g) \) multiplications and additions, where \( g \) is the non-zero coefficients. Every iteration requires an exact solution to the linear programming problem with \( g = 0 \) in \( f \). If \( f \) is positive, we apply this algorithm to \( m + 1 \) perturbed problems with right hand sides \( q_i = \epsilon \) where \( \epsilon > 0 \). We choose a feasible solution to the phase 1 problem. We assume that all perturbed solutions to the linear programming problems converge when \( j \) is large enough. A sequence \( \epsilon_i \) is applied to various solutions \( x_i \). If \( \epsilon_i \) is small, the solution is feasible. From Newton's Center of Gravity Algorithm we have:

\[
\begin{align*}
\text{Find } x > 0. \\
\epsilon_i = f \sum x_i \\
q_i = f \sum \frac{1}{u_i} \\
\end{align*}
\]

We consider the general phase I linear programming problem with a constraint constraint which can be written after some algebraic manipulation in the form: