A Characterization of the "Smallest Large Variable" Condition Number in Linear Programming*

Hugo J. Lara [1] *Clovis C. Gonzaga* [2]

[1]Departamento de Investigacion de Operaciones, Decanato de Ciencias
Universidad Centro Occidental Lisandro Alvarado (UCLA). Apdo 400
Barquisimeto. Venezuela
hugol@delfos.ucla.edu.ve

[2]Department of Mathematics, Federal University of Santa Catarina
Cx. Postal 5210, 88040-970 Florianopolis, SC, Brazil
clovis@mtm.ufsc.br

Abstract

Two condition numbers described by Ye, Vavasis, Stewart, Todd and others have been used to study the complexity of interior point algorithms. The first one is given by the "smallest large variable" in the optimal face of the linear programming problem, and the other by the supreme of the norms of all the oblique projection operators on the range space of A^T. The first one has the disadvantage of depending on the knowledge of the optimal partition, and the second depends only on the data but does not use the structure of the linear programming problem. We give a characterization of the first one, based on the edges of the feasible polyhedron, which does not depend on the knowledge of the optimal partition.

Keywords: Condition numbers, complexity, interior point algorithms.

1 Introduction

The complexity study of path following algorithms for linear programming traditionally uses a stopping rule which involves the "bit-size" of the problem. This model, based on the Turing machine computation model assumes that the data entries of the linear programming problem are integers. The complexity study is a strategy to measure the performance of the algorithm by counting the arithmetic operations in each iteration, and estimating bounds to the number of iterations in the "worst case"

*Research supported in part by Conicit, Venezuela.

instance. Given the continuous nature of the path following algorithms for linear programming, it seems natural to establish a real model to study the complexity. Blum, Shub and Smale [1] define a computation model called "universal machine", where the data are assumed to be real numbers, and the arithmetic operations are simple operations. Several recent works follow these ideas in the study of interior point algorithms for linear programming. For instance Renegar [5], Freund [4], Vera [13] and others use the distance to "ill posedness" as condition measures, and Vavasis and Ye use condition numbers based on the optimal set and on oblique projections. In this work we shall consider the condition number of the "smallest large variable" of the optimal set given by Ye [14]. This number has the disadvantage requiring the knowledge of the optimal partition, and so the stopping rule that it establishes has no practical sense. There is an attempt to characterize this condition measure, given by Vavasis and Ye [9], based on all the oblique projection operators on the range space of the matrix that defines the linear feasibility problem in Karmarkar's form. Tunçel [6] proved that this characterization might be poor, showing instances where the smallest large variable is big, and the characterization too small. The main idea in the complexity study is to give a stopping rule which ensures the identification of the optimal partition while following the central path. In [11] Vavasis and Ye established an active set strategy where the condition number of oblique projections is used to guess the partition. The complexity study of this algorithm is difficult. In this work we give a characterization of the smallest large variable condition number for linear programming in standard form, that does not depend on the knowledge of the optimal partition. This characterization relies on the edges of the polyhedron that defines the feasible set. In this way we establish a stopping rule based on this condition number that allows one to separate large and small variables, and identify the optimal partition. In consequence we can study the complexity of path following interior point algorithms in a real framework.

Notation: Vectors in $\mathbb{R}^n$ will be denoted with superscripts. The scalars will be denoted with subscripts. Hence, for the scalar μ, μ^i denotes μ to the power i.

Given a vector x, the corresponding uppercase symbol denotes as usual the diagonal matrix X, defined by the vector. The symbol e will represent the vector of all ones, with dimension given by the context.

Given a matrix A, its null space and the column range space are denoted respectively by $\mathcal{N}(A)$ and $\mathcal{R}(A)$. The projection matrix onto $\mathcal{N}(A)$ is P_A, and its complement $\tilde{P}_A = I - P_A$.

We shall denote component-wise operations on vectors by the usual notations for the numbers. Thus, given two vectors u, v of the same dimension, uv, u/v, etc will denote the vectors with components $u_i v_i$, u_i/v_i, etc. This notation is consistent as long as component-wise operations always have precedence in relation to matrix operations. Note that $uv \equiv Uv$ and if A is a matrix, then $Auv \equiv AUv$, but in general $Auv \neq (Au)v$.

Given an index set $\{1,\ldots,n\}$ and $J \subset \{1,\ldots,n\}$ the cardinality of J will be denoted by $|J|$; $J^c = \{1,\ldots,n\}\backslash J$ stand for the complement of J; v_J represent the restriction of the vector v indexed by J.

Given the vector $d \in I\!R^n_+$, we define $B(d) = \{i = 1,\ldots,n \mid d_i > 0\}$, and $N(d) = B(d)^c$.

The notation $g = O(r)$ means that there exists a positive constant M such that $\|g\| \leq Mr$.

Given a set $\mathcal{A} \subset I\!R^n$, and the vector $v \in I\!R^n$ the sum of sets $\{v\} + \mathcal{A}$ will be frequently written as $v + \mathcal{A}$.

We shall denote the sequences in $\mathcal{A}$ by (x^k) in $\mathcal{A}$, and subsequences by $(x^k)_K$ where $K \subset I\!N$.

2 The normalized linear programming problem

In this section we shall establish the linear programming problem, the assumptions and a "normalization"of the problem. We consider the primal and dual linear programming problems associated with the instance $\bar{A}$, $\bar{b}$, and $\bar{c}$, where $\bar{A} \in I\!R^{m\times n}, \bar{c} \in I\!R^n$ and $b \in I\!R^m$:

$$(P) \quad \begin{array}{ll} \text{minimize} & \bar{c}^T x \\ \text{subject to} & \bar{A}x = b \\ & x \geq 0 \end{array}$$

and

$$(D) \quad \begin{array}{ll} \text{maximize} & b^T y \\ \text{subject to} & \bar{A}^T y + s = \bar{c} \\ & s \geq 0 \end{array}$$

We define the primal-dual feasible set as

$$F = \{(x,s) \in I\!R^n \times I\!R^n \mid \bar{A}x = b, x \geq 0, s = \bar{c} - \bar{A}^T y, s \geq 0, y \in I\!R^m\},$$

and the interior point set as

$$F^o = \{(x,s) \in F \mid x > 0, s > 0\}.$$

We enforce the following assumptions, standard in interior point path following algorithms:

- $F^o \neq 0$
- $rank(\bar{A}) = m$.

- $\bar{c} \notin \mathcal{R}(\bar{A}^T)$.

The set of optimal solutions for the primal-dual pair of problems constitutes a face Ω of the polyhedron of feasible solutions F. By the assumptions we can prove that this face is a compact set. It is well known that this face is characterized by a partition $\{B, N\}$ of the set of indices $\{1, \ldots, n\}$ such that

$$\Omega = \{(x, s) \in F \mid x_N = 0, s_B = 0\}.$$

In the relative interior of the optimal face Ω, $x_B > 0$ and $s_N > 0$. We call x_B and s_N large variables, and x_N and s_B small variables.

We study algorithms that converge to the optimal face. Our main concern is with the behavior of the iterates when they approach the optimal face, by following the central path. For details on the central path, see Gonzaga [2].

Given $\mu > 0$, $\mu \in I\!R$, the pair $(x, s) \in F$ is the central point $(x(\mu), s(\mu))$ associated with μ if

$$xs = \mu e. \tag{1}$$

The central path is the curve in $I\!R^{2n}$ parameterized by the positive real μ, that is

$$\mathcal{CP} = \{(x, s) \in F \mid xs = \mu e, \mu > 0\}.$$

We want to study the conditioning of the pair of problems $(P), (D)$, that is how a problem defined by a given instance $\bar{A}, b, \bar{c}$ is "hard" to be solved. We now imagine an iterative procedure for finding an optimal solution of the pair $(P), (D)$. It is necessary to give the procedure an initial point, and the number of iterations clearly depends on where the initial point is. In consequence, the instance will not be well defined only with the data $\bar{A}, b, \bar{c}$. It is necessary to incorporate the initial point in the definition of the instance to complete the information. When we study the complexity of the algorithms we are interested in two questions: The first one is how many computations the computer will perform to give us a response in the "worst case" instance? and the second one is: how can we compare instances, in order to label them as "easy" or "hard"?. In order to establish a standard criterion to answer the second question, we will define a pair of primal and dual problems, equivalents to (P) and (D), but incorporating additional information to the instance, coming from the structure, giving a standard "initial point".

We denote $(\hat{x}, \hat{s}) = (x(1), s(1))$, the point on the central path associated with $\mu = 1$. We define the scaling matrix $\hat{D} = \hat{X}^{\frac{1}{2}} \hat{S}^{-\frac{1}{2}}$. We now define the scaled problem by changing the primal and dual feasibility conditions, and the objective function in the following way: We define $A = \bar{A}\hat{D}$, $\bar{x} = \hat{D}^{-1}x$, $\bar{s} = \hat{D}s$ and $c = \hat{D}\bar{c}$. The primal

and dual scaled pair of problems is

$$\begin{array}{ll} \text{Minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0 \end{array}$$

and

$$\begin{array}{ll} \text{Maximize} & b^T y \\ \text{subject to} & A^T y + s = c \\ & s \geq 0. \end{array}$$

It can be easily verified that the vector e is primal and dual feasible for this pair of scaled problems. Now we have a standard initial point for a path following algorithm.

The dual feasibility condition can be written as $s \in c + \mathcal{R}(A^T)$ where $\mathcal{R}(A^T)$ is the image space of the matrix A^T. It can be easily verified that $c + \mathcal{R}(A^T) = s + \mathcal{R}(A^T)$ for all $s \in c + \mathcal{R}(A^T)$; that is, we can replace c by some $\bar{s}$ in the data, obtaining the same problem. Since $e \in c + \mathcal{R}(A^T)$, we write the dual feasibility condition as $s \in e + \mathcal{R}(A^T)$, and the primal objective function as $e^T x$. On the other hand, since $Ae = b$, the primal feasibility condition can be written as $A(x-e) = 0$ or $x - e \in \mathcal{N}(A)$, and the dual objective function can be expressed as $e^T A^T y = e^T(e - s) = e^T e - e^T s$. Maximizing $e^T e - e^T s$ is equivalent to minimizing $e^T s$, and the optimal values differ by the constant $e^T e = n$. Now, the "normalized" primal and dual pair of problems are

$$\begin{array}{lll} & \text{minimize} & e^T x \\ (P_e) & \text{subject to} & x - e \in \mathcal{N}(A) \\ & & x \geq 0 \end{array}$$

and

$$\begin{array}{lll} & \text{minimize} & e^T s \\ (D_e) & \text{subject to} & s - e \in \mathcal{R}(A^T) \\ & & s \geq 0. \end{array}$$

Written in this way, the problems $(P_e), (D_e)$ are symmetric with respect to the orthogonal subspaces $\mathcal{N}(A)$ and $\mathcal{R}(A^T)$. It is worth noting that the definition of the pair of linear programming problems does not depends on the matrix A, but relies on the orthogonal subspaces $\mathcal{N}(A)$ and $\mathcal{R}(A^T)$.

We define the normalized feasible set as

$$F_e := \{(x, s) \in I\!R^n \times I\!R^n \mid Ax = b, x \geq 0, s = e - A^T y, s \geq 0, y \in I\!R^m\},$$

the normalized central path

$$\mathcal{CP}_e := \{(x, s) \in F_e \mid xs = \mu e, \mu > 0\}$$

and the optimal face

$$\Omega_e := \{(x, s) \in F_e \mid x_N = 0, s_B = 0\}.$$

We also define the proximity measure to the central path as

$$\mathbb{R}^n_{++} \times \mathbb{R}^n_{++} \times \mathbb{R}_+ \ni (x, s, \mu) \mapsto \delta(x, s, \mu) := \|\frac{xs}{\mu} - e\|.$$

3 The "smallest large variable"

Given the optimal face Ω_e, and the optimal partition $\{B, N\}$ of the indices $\{1, \ldots, n\}$ as above, we define the following condition number:

$$\begin{aligned} \sigma_P &= \min_{j\in B}\{\max x_j \mid (x, s) \in \Omega_e\}, \\ \sigma_D &= \min_{j\in N}\{\max s_j \mid (x, s) \in \Omega_e\}, \\ \sigma &= \min\{\sigma_P, \sigma_D\}. \end{aligned}$$

If the optimal face is non-degenerate then σ is indeed the smallest large variable of the unique optimal solution. It is intuitive that σ should act as condition measure for the linear programming problem, since the smaller this smallest large variable is, the greater the difficulty that the methods face in identifying it as large variable.

In [14], Ye studied the complexity of some path following algorithms using σ to define a stopping rule. He establishes a real model to express the complexity bounds of those algorithms, under the assumption that all the data entries are real numbers. The number σ carries more information about the difficulty of the problem than the input size L. In fact, there exist instances whose input data sizes are very large, but which define essentially easy linear programming problems. We now quote without proof Ye's complexity theorem:

Consider a sequence $(x^k, s^k, \mu^k) \in F \times \mathbb{R}_+$ verifying for $\alpha \in (0, 1)$ that

$$\delta(x^k, s^k, \mu^k) \leq \alpha \tag{2}$$

It is well known that several path following algorithms (for instance the short step path following algorithm) generates sequences verifying (2) (see for example Gonzaga [2]). We define the sets $B^k = \{j \in \{1, \ldots, n\} \mid s^k_j \leq x^k_j\}$ and $N^k = \{j \in \{1, \ldots, n\} \mid x^k_j < s^k_j\}$.

Theorem 3.1: Consider a sequence $(x^k, s^k, \mu^k) \in F^o \times \mathbb{R}_+$ with $\mu^k \to 0$ satisfying (2), and the sets B^k, B, N^k, N defined as above. Then $B^k = B$ and $N^k = N$ for all k such that

$$\mu^k < \frac{\sigma^2}{n^2}(1 - \alpha).$$

It is also well known, for several path following algorithms, that the parameter μ^k satisfies, for all $k \in \mathbb{N}$:

$$\mu^k \leq \mu^0(1 - \frac{\nu}{\sqrt{n}})^k, \tag{3}$$

where $\nu \in (0,1)$. We can estimate a polynomial bound for the number of iterations of the path following algorithms by using (3). The classical model assumes a stopping rule given by $\mu^k < 2^{-L}$ where L is the number of bits needed to store the data by assuming each entry as an integer number. The following lemma calculates such bound:

Lemma 3.2: Given a sequence (x^k, s^k, μ^k) satisfying (2) and the number L defined as above. Then:

$$k = O(\sqrt{n}L).$$

Proof.
By (3) we have

$$\frac{\mu^k}{\mu^0} \leq (1 - \frac{\nu}{\sqrt{n}})^k.$$

Taking logarithms in both sides, and using the relation $\log(1-t) \leq -t$, $\forall t \in (0,1)$ we obtain

$$\log \frac{\mu^k}{\mu^0} \leq k \log(1 - \frac{\nu}{\sqrt{n}}) \leq -\frac{k\nu}{\sqrt{n}},$$

so

$$k \leq -\frac{\sqrt{n}}{\nu} \log\left(\frac{\mu^k}{\mu^0}\right). \tag{4}$$

Now we use the stopping rule $\mu^k < 2^{-L}$ to obtain

$$k \leq \frac{\sqrt{n}\log(\frac{2}{\mu^0})}{\nu} L,$$

that is, $k = O(\sqrt{n}L)$. ∎

With an analogous argument it is possible to establish the following lemma (due to Ye [14]), which describes the complexity of the central path algorithms by using the real model given by the "smallest large variable":

Lemma 3.3: Given a sequence (x^k, s^k, μ^k) satisfying (2), and the stopping rule $\mu^k \leq \frac{\sigma^2(1-\alpha)}{n^2}$, then

$$k = O(\sqrt{n}(|\log \sigma| + \log n)).$$

Proof.
We substitute $\mu^k \leq \frac{\sigma^2}{n^2}(1-\alpha)$ in (4) to obtain

$$k \leq -\frac{\sqrt{n}}{\nu} \log(\frac{\sigma^2}{n^2\mu^0}(1-\alpha)) = \frac{2\sqrt{n}}{\nu}(|\log \sigma| + \log n) + \frac{\sqrt{n}}{\nu}(\log|1-\alpha| + \log \mu^0).$$

∎

The number σ described above defines a real model for studying the complexity of the path following algorithms. Although this model takes more structure of

the instance than the "bit-size" model, and we do not need to assume the data as integers, we cannot use the stopping rule that σ defines, because it requires the knowledge of the optimal partition $\{B, N\}$.

Another condition number, denoted by $\bar{\chi}_A$, studied by Stewart [7], Vavasis and Ye [9] and others, was used to express the complexity bounds of some path following algorithms. The number $\bar{\chi}_A$ is defined as the supreme of the norms of all the oblique projection operators into the image space of the matrix A^T. Vavasis and Ye [9] discuss ways of relating this condition measure with σ. It is easy to see that σ can be made arbitrarily large (or small) by changing b or c while keeping A fixed. Therefore, at first glance, no apparent relationship exist. However, Vavasis and Ye [10] study these condition number for feasibility problems over polyhedra expressed in Karmarkar's standard form. They proved for this problem that σ is bounded below by $1/\bar{\chi}_A$. This fact allows us to use the theorem 3.3 to study the complexity of some path following algorithms in a real framework, and without the knowledge of the optimal partition for the problem of finding a feasible point of those polyhedra in Karmarkar's form, achieving a bound of $O(\sqrt{n}(\log \bar{\chi}_A + \log n))$ for the number of iterations. However, this bound can be rough. In fact, Tunçel, [6] describes a family of instances A_ϵ which define polyhedra in standard form, with σ_ϵ of the order of $1/2$ for all ϵ sufficiently small, and $\bar{\chi}_{A_\epsilon}$ growing indefinitely when ϵ goes to zero. That is, the problem defined by A_ϵ is "well conditioned" in the sense of the smallest large variable, but we would label it as "ill conditioned" by using the characterization given by $\bar{\chi}_{A_\epsilon}$. This means that, even for polyhedra in Karmarkar's form, there is no relationship between σ and $\bar{\chi}_A$. One possible explanation for the number $\bar{\chi}_A$ failure in characterizing σ is that it uses only information of the matrix A, and not on the other data b or c. We shall define a condition number that uses more information of the structure of the problem than $\bar{\chi}_A$, but without the knowledge of the optimal partition, as σ needs. Later we relate such number with σ and establish a real framework for studying the complexity of some path following algorithms for linear programming in standard form.

4 The edge condition number

We define the sets

$$\Gamma = \{J \subset \{1, \ldots, n\} \mid \dim \mathcal{N}(A_J) = 1\},$$

$$W = \{w \in I\!R^n \mid J \in \Gamma, w_J \in \mathcal{N}(A_J), w_{J^c} = 0, e^T w \geq 0\},$$

and

$$\hat{W} = \{w \in W \mid e^T w > 0\}.$$

For the matrix Z defined in such way that its rows form a basis for $\mathcal{N}(A)$ we define

$$\Gamma' = \{J \subset \{1, \ldots, n\} \mid \dim \mathcal{N}(Z_J) = 1\},$$

$$W' = \{v \in I\!R^n \mid J \in \Gamma', v_J \in \mathcal{N}(Z_J), v_{J^c} = 0, e^T v \geq 0\},$$

and

$$\hat{W}' = \{v \in W' \mid e^T v > 0\}.$$

Vectors in W are vectors in $\mathcal{N}(A)$ parallel to the edges of the polyhedron that define the primal feasible set, and elements of W' belong to $\mathcal{R}(A^T)$ and are parallel to the edges of the dual feasible set. We define the sets W,Γ and $\hat{W}$ assuming only the knowledge of the instance data, this is we assume the optimal set to be unknown. We define a condition number, based on this sets. For a vector $t \in I\!R^n$, we define $t^+ = \max\{t_i \mid i = 1, \ldots, n\}$. We also define the edge condition number as follows:

$$\xi_P := \min\{\frac{e^T w}{w^+} \mid w \in \hat{W}\},$$

$$\xi_D := \min\{\frac{e^T v}{v^+} \mid v \in \hat{W}'\},$$

and

$$\xi := \min\{\xi_P, \xi_D\}.$$

The next theorem characterizes the dual smallest large variable σ_D:

Lemma 4.1: Consider $j \in N$ and let $\sigma_{D_j} = \max\{s_j \mid (x, s) \in \Omega_e\}$. Then there exists $w^{(j)} \in \hat{W}$ such that $\sigma_{D_j} = e^T w^{(j)} / w_j^{(j)}$.

Proof. By the definition, σ_{D_j} is the optimal value of the problem

$$\begin{array}{rll} & \text{maximize} & s_j = e_j^T s_N \\ (P_j) & \text{subject to} & A_B^T y = e_B \\ & & A_N^T y + s_N = e_N \\ & & s_N \geq 0. \end{array}$$

where e_j denotes the $j-th$ canonical vector of $I\!R^n$. The dual problem associated with (P_j) is

$$\begin{array}{rll} & \text{minimize} & e^T w \\ (D_j) & \text{subject to} & Aw = 0 \\ & & w_N \geq 0 \\ & & w_j \geq 1 \end{array}$$

The problem (P_j) has a positive optimal value because the components indexed by N are positive in the relative interior of the dual optimal face. It follows by duality that the problem (D_j) also has a positive optimal value. Let w be an optimal solution for (D_j). Then $w_j = 1$, because otherwise $w_j > 1$ and $\tilde{w} = w/w_j$ would also be feasible, with $e^T \tilde{w} = e^T w / w_j < e^T w$, contradicting the optimality of w. Hence (D_j) can be written as

$$\begin{array}{rll} & \text{minimize} & e^T w \\ (D_j) & \text{subject to} & A_B w_B + A_{N_j} w_{N_j} = -A_j \\ & & w_{N_j} \geq 0 \\ & & w_j = 1, \end{array}$$

where $N_j = N \backslash \{j\}$.

Among the optimal solutions for (D_j), let us choose w such that the number of null components is maximum. Let $J = \{i = 1, \ldots, n \mid i \neq j, w_i \neq 0\}$. Then

$$A_J w_J = -A_j.$$

We now prove the following fact: the columns of A_J are linearly independent.

Suppose by contradiction that $A_J \lambda_J = 0$ for some $\lambda_J \neq 0$. Let $\lambda_{J^c} = 0$. Then for $\gamma \in I\!R$, $A_J(w_J + \gamma\lambda_J) = -A_j$. It follows that for $|\gamma|$ sufficiently small, $w + \gamma\lambda$ is feasible for (D_j).

— If $e^T\lambda \neq 0$, we can choose $\gamma \in I\!R$ such that $\gamma e^T\lambda < 0$ and $w + \gamma\lambda$ is feasible. Then $e^T(w + \gamma\lambda) < e^T w$, contradicting the optimality of w.

— Assume that $e^T\lambda = 0$. Let β be the set of all $\gamma \in I\!R$ such that all components of $w + \gamma\lambda$ and w have the same signs. Since $\lambda_J \neq 0$, $\beta \neq I\!R$, and hence there exists $\bar{\gamma}$ such that all components of $w + \gamma\lambda$ and w have the same signs for $\gamma \in (0, \bar{\gamma})$ and $w_i + \gamma\lambda_i = 0$ for some $i \in J$. This point is obviously feasible (and hence optimal) for (D_j) and has more null components than w, which contradicts the definition of w, and completes the proof of the fact.

So, $A_J w_J = -A_j$ and the columns of A_J are linearly independent. Hence for $\bar{J} = J \cup \{j\}$, $\mathcal{N}(A_J)$ has dimension 1. We have shown that $w \in \hat{W}$ with $w_j = 1$, and $\sigma_{D_j} = e^T w$, completing the proof. ∎

Theorem 4.2: Consider the condition numbers defined above. We have

$$\sigma_D \geq \xi_P,\ \sigma_P \geq \xi_D,\ \sigma \geq \xi.$$

Proof. From the lemma above, for $j \in N$ there exists $w^{(j)} \in \hat{W}$ such that

$$\sigma_{D_j} = \frac{e^T w^{(j)}}{w_j^{(j)}} \geq \frac{e^T w^{(j)}}{w^{(j)+}}.$$

Hence

$$\sigma_D = \min_{j \in N} \frac{e^T w^{(j)}}{w_j^{(j)}} \geq \min_{w \in \hat{W}} \frac{e^T w}{w^+} = \xi_P.$$

The second inequality has an analogous proof by the symmetry of the treatment, and the third follows directly from the first two, completing the proof. ∎

This result gives us an interesting relationship between the primal feasible set and the dual optimal face. In fact, the dual "smallest large variable" is bounded below by a number that depends on the "edges" of some primal feasible set. Given σ and ξ defined as above, it follows

$$\sigma \geq \xi.$$

This result allows us to establish a bound for the number of the iteration of an interior point path following algorithm, in function of ξ, whose definition does not depend on the knowledge of optimal partition:

Corollary 4.3: Given a sequence (x^k, s^k, μ^k) generated by some interior point path following algorithm, satisfying (2); and the stopping rule

$$\mu^k \leq \frac{\xi^2}{n^2}(1-\alpha),$$

then $k = O(\sqrt{n}(|\log \xi| + \log n))$.

Proof. Immediate, by theorems 4.2 and 3.3. ∎

In our knowledge, this number ξ is the first characterization of the "smallest large variable" given by the linear programming problem in standard form. The number relies on the data A, b, c, x^o whose entries are real numbers. This model establishes a real model to study the complexity of the path following algorithms better than σ in the sense that it does not depend on the knowledge of the optimal partition. The relation of the "smallest large variable" with the "edges" of the feasible polyhedra is an interesting feature developed in this paper.

5 A simple example

In order to provide intuition on how the condition numbers measure the complexity of interior point methods, we consider the primal-dual linear programming problems defined by the following family of data instances: $A(\epsilon) = [2, 1, 2(1-\epsilon)]$, $b(\epsilon) = [5 - 2\epsilon]$ and $c(\epsilon) = e$, where $\epsilon \in (0, 1/2)$. Our aim is to calculate both the condition numbers studied in former sections. We begin by computing the smallest large variable condition number: First we determine the primal-dual optimal set $\Omega(\epsilon) = \{(x,s) = ((\frac{5-2\epsilon}{2}, 0, 0)^T, (0, \frac{1}{2}, \epsilon)^T)\}$. It is easy to verify that $\sigma_P(\epsilon) = (5-2\epsilon)/2 > 2$ and $\sigma_D(\epsilon) = \epsilon < 1/2$, in consequence $\sigma(\epsilon) = \epsilon$. While ϵ get smaller the difficulty to identify the third dual component as a large variable increases. This can be seen as a primal near degeneracy, because if $\epsilon = 0$ then the primal optimal set becomes a line segment, instead a singleton. In this case the smallest large variable is large ($\sigma(0) = 1/2$) that is the degeneration is not a problem, the hardness arises in the near-degeneracy.

Let us calculate the edges condition number: First we compute $W_P = \{w^1, w^2, w^3\} = \{(-1/2, 1, 0)^T; (0, 1, -1/2(1-\epsilon))^T; (-1+\epsilon, 0, 1)^T\}$ and $W_D = \{v^1\} = \{(1, 1/2, 1-\epsilon)^T\}$. Then $\xi_P(\epsilon) = e^T w^3 = \epsilon < 1/2$, $\xi_D(\epsilon) = e^T v^1 = (5-2\epsilon)/2 > 2$, and so $\xi(\epsilon) = \epsilon$. Note that the dual smallest large variable is calculated by using primal directions that are parallel to the edges of the primal optimal set, this is we find information about the size of dual optimal variables in the primal feasible set, and vise versa.

In this example both the numbers coincide, but unfortunately this not always happens. When the near degeneracy is far from the optimal face, ξ can be much smaller than σ.

References

[1] L. Blum, M. Shub and S. Smale. "On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions and universal machines", *AMS Bull. (New series)*, 21, pp 1-46 (1989).

[2] C. C. Gonzaga. "Path-following Methods for linear programming." *SIAM Review*. Vol. 34, No. 2, pp 167-224, June (1992).

[3] C. C. Gonzaga and H. J. Lara. A note on properties of condition numbers. *Linear Algebra and its Applications*, 261:269–273, 1997.

[4] R. Freund and J. Vera. Some characterization and properties of the "Distance to ill posedness" and the condition measure of a conic linear system. To appear in Mathematical Programming. 1995.

[5] J. Renegar. "Incorporating condition measures into the complexity theory of linear programming," *SIAM Journal on Optimization*, Vol. 5,No 3,pp 506-524, August (1995).

[6] L. Tunçel. "On the condition numbers for polyhedra in Karmarkar's form", Research Report CORR 97-24. Department of Combinatorics and Optimization. Faculty of Mathematics. University of Waterloo. Waterloo Ontario. Canada. November (1997).

[7] G. W. Stewart. On scaled projections and pseudoinverses. *Linear Algebra and its Applications*, 112:189–193, 1989.

[8] M. J. Todd. A Dantzig-Wolfe-like variant of Karmarkar's interior-point linear programming algorithm. *Operations Research*, 38:1006–1018, 1990.

[9] S. A. Vavasis. Stable numerical algorithms for equilibrium systems. *SIAM J. Matrix Analysis*, 15:1108–1131, 1994.

[10] S. A. Vavasis and Y. Ye. Condition numbers for polyhedra with real number data. Technical Report 93-1398, Dept. of Computer Sciences, Cornell University, Ithaca, NY, 1993.

[11] S. A. Vavasis and Y. Ye. Identifying an optimal basis in linear programming. Technical report, Dept. of Computer Sciences, Cornell University, Ithaca, NY, 1993.

[12] S. A. Vavasis and Y. Ye. An accelerated interior point method whose running time depends only on A. Technical Report 93-1391, Dept. of Computer Sciences, Cornell University, Ithaca, NY, 1993.

[13] J. Vera. Ill posedness and the complexity of deciding existence of solutions to linear programs. *SIAM Journal on Optimization*, 6(3): 549-569, 1996.

[14] Y. Ye. On the finite convergence of interior-point algorithms for linear programming. *Mathematical Programming*, 57(1992) 325-335.