Consiglio Nazionale delle Ricerche

ISTITUTO DI ELABORAZIONE DELLA INFORMAZIONE

PISA

NUMERICAL STUDY ON SOME FEASIBLE DIRECTION METHODS IN MATHEMATICAL PROGRAMMING

M. Arioli, A. Laratta, O. Menchi

Nota interna B80-30
Dicembre 1980

NUMERICAL STUDY ON SOME FEASIBLE DIRECTION METHODS IN MATHEMATICAL PROGRAMMING.

+Arioli M., +Laratta A., ++Menchi C.

Research supported by Progetto Finalizzato Informatica del C.N.R. (Sottoprogetto P1, SOFMAT).
+ I.E.I. CNR, via S.Maria 46, Pisa
++I.S.I. Universita' degli studi and CNUCE, Pisa

Introduction

Any method of the feasible directions is an iterative procedure based on three main points:

a) computation of an initial point;

b) computation of a feasible direction;

c) computation of an optimum point along a direction.

In this paper the Zoutendijk method in Euclidean norm, the Rosen method and some slight modifications are studied. In [16] some efficient algorithms for computing the initial point and the optimum point along a direction are discussed. Here efficient procedures are described for the computation of the direction. Only linearly constrained problems are taken into account; besides, it is known that non linearly constrained problems can be efficiently solved by solving a sequence of linearly constrained ones. Convergence properties can be found in [9], [13], [15].

The direction given by the Zoutendijk method, as well known, is the solution of the least-squares problem:

$$(1) \qquad \min_{s} \{ \, ||p-s|| \; : \; Cs \leq 0 \, \}$$

where p is the gradient of the objective function at a point, C is q X n matrix ($q \leq n$), whose rows are the coefficients of the active constraints at that point and $||.||$ denotes the Euclidean norm.

In [1] numerical procedures for the computation of the solution of (1) are described which are based on either Gauss-Jordan like or orthogonal matrix transformations. Here the results of [1] are employed for an efficient implementation of some methods of feasible directions.

The results of the evaluation are presented in paragraph 5.

1 - Formulation of the problem

Let $f(x) \in C^1$ be a convex function of $\mathbb{R}^n \to \mathbb{R}$, let A be a real m X n matrix, a_j^T, j=1,...,m being the rows of A, and let b be a real vector of order m.

We consider the convex programming problem:

(2) $\min\{ f(x) : Ax \le b \}$.

Let

$$K(x) = \{ k : a_k^T x = b_k \}$$

be the set of indices of active constraints in x, and let

$$S(x) = \{ s : s \in R^n , a_j^T s \le 0 , j \in K(x) \}$$

be the cone of the feasible directions in x.

Having chosen an initial feasible point $x^{(0)}$, the Zoutendijk method determines a sequence $\{ x^{(i)} \}$ of feasible points by the formulae

(3) $$x^{(i+1)} = x^{(i)} + \lambda_i s^{(i)}$$

where $s^{(i)}$ is the solution of the problem

(4) $$\max \{ p^T s : s \in S(x^{(i)}) , \|s\| \le 1 \}$$

$\|.\|$ denoting the Euclidean norm and p being the vector $-\nabla f(x^{(i)})$, and λ_i is chosen so that

$$f(x^{(i+1)}) = \min_\lambda \{ f(x^{(i)} + \lambda s^{(i)}) : A(x^{(i)} + \lambda s^{(i)}) \le b \}$$

We focus the attention on the solution of the problem (4). For the computation of $x^{(0)}$ and λ_i we refer to known results. In the following we denote by C the q X n matrix of the active constraints at $x^{(i)}$; we suppose the rows of C linearly independent, whence $q \le n$.

In [16] it is proved that the solution of (4) is the same, apart from a proportionality factor, as the projection of p onto the cone $S(x^{(i)})$, which is the solution of the problem

(5) $\min \{ \|p-s\| : Cs \leq 0 \}$.

In other words, if $p \in S(x^{(i)})$ then p is solution of (5) and $p/\|p\|$ is solution of (4); if $p \notin S(x^{(i)})$, then the solution of (4) is the normalized solution of (5). Obviously, if p belongs to the cone

$$\bar{S}(x^{(i)}) = \{ s : s = C^T u , u \geq 0 \}$$

then $x^{(i)}$ is a solution of (2).

If $p \notin S(x^{(i)})$ then the projection of p onto $S(x^{(i)})$ belongs to the boundary of $S(x^{(i)})$, therefore to a linear manifold obtained by intersecting some hyperplanes $a_j^T x = b_j$, $j \in K(x^{(i)})$.

The method we are going to describe finds such a manifold and computes the solution of (5) on it, according to the following scheme:

a) at each step it computes the projection p_C of p onto a set of equality constraints $Cs=0$;

b) if p_C is not a solution of (5), it adds a constraint to the above set, or deletes one.

For both points a) and b) techniques of Gauss-Jordan like and orthogonal matrix transformations are used [1], [3], [4], [5], [11], [12], [14].

2 - Analytical preliminaries

2.1 - Denote by Q_C the manifold generated by the matrix C:

$$Q_C = \{ s \in \mathbb{R}^n : Cs = 0 \}$$

and by N_C the orthogonal complement of Q_C in $\mathbb{R}^n$. Then N_C is a q-dimensional vector space, and the rows of C form a basis of N_C (in what follows we suppose that the rows of C are normalized, $\|c_j\|=1$, $j=1,\dots,q$).

Then the vector p (also normalized $\|p\|=1$) can be written as:

(6) $p = p_C + \bar{p}_C$, $p_C \in Q_C$, $\bar{p}_C \in N_C$

where p_C is the projection of p onto the manifold $\mathcal{C}_C$.

On the other hand there exists one and only one vector $u^C \in \mathbb{R}^q$ such that

$$\bar{p}_C = C^T u^C ,$$

hence

$$(7) \qquad p = p_C + C^T u^C .$$

Multiplying by matrix C, we obtain the linear system with the unknowns u^C :

$$(8) \qquad CC^T u^C = Cp .$$

Having assumed that C has maximal rank, the matrix CC^T is positive definite and the system (8) has one and only one solution u^C.

Let us remark that:

a) if $Cp \le 0$ then p is solution of (5);

b) if $Cp \not\le 0$ then, after computing u^C, from (7) we obtain p_C:

if $u^C \ge 0$ then p_C is the solution of (5);

if $u^C \not\ge 0$ then the solution of (5) is the projection of p onto a subset of the constraints specified by C.

Let be $Cp \not\le 0$ and $u^C \not\ge 0$; we choose r rows of C (assumed to be the first r ones, otherwise it is sufficient to reorder appropriately the equations of the system $Cs=0$).

Subdividing C :

$$(9) \qquad C = \begin{bmatrix} C_1 \\ C_2 \end{bmatrix}$$

where C_1 e C_2 are respectively $r \times n$ and $(q-r) \times n$ matrices, by applying the block Gauss-Jordan method, the system (8) is transformed into

$$(10)\quad \begin{cases} u^{C}_{1} + (C_1 C_1^T)^{-1} C_1 C_2^T u^{C}_{C_2} = u^{C_1} \\ -C_2 P_{C_1} C_2^T u^{C}_{C_2} = -C_2 p_1 \end{cases}$$

where P_{C_1} is the projection matrix onto the manifold Q :

$$P_{C_1} = I - C_1^T (C_1 C_1^T)^{-1} C_1$$

and p_{C_1} is the projection of p onto Q_{C_1} , u^{C_1} is the vector of the Lagrangian multipliers of p related to Q_{C_1}.

These formulae make the Lagrangian multipliers u^C of p , related to C s=0, correspond to the Lagrangian multipliers u^{C_1} related to $C_1 s=0$.

From them it follows that:

a) if the right-hand side is positive, then p_{C_1} is the solution of (5).

b) if $C_2 p_{C_1}$ is non-positive, then $p_{C_1} = p - C_1^T u^{C_1}$ is feasible.

For brevity sake, we shall call optimal matrix the submatrix C_1 of C if a) holds, that is the solution of (5) coincides with the solution of the problem

$$\min \{ \| p - s \| \ : \ C_1 s = 0 \}$$

From (10) the following consequences can be easily obtained [1]:

Theorem 1. - Let C_1 be an optimal matrix and subdivide C_1 as follows:

$$C_1 = \begin{bmatrix} C'_1 \\ \\ C''_1 \end{bmatrix} \quad ; \text{ then}$$

$$u^{C_1}_{C''_1} \neq 0 \implies C''_1 \, p_{C'_1} \not\leq 0 \quad , \qquad u^{C_1}_{C'_1} \neq 0 \implies C'_1 \, p_{C''_1} \not\leq 0 \, .$$

Corollary 1. - Under the hypotheses of theorem 1, it follows that at least one constraint specified by a row of C''_1 is among those which are not verified by $p_{C'_1}$.

Corollary 2. - The solution of (5) verifies at least one of the constraints not verified by p.

Theorem 2. - Let C_1 be an optimal matrix and $C_2 p_{C_1} \neq 0$, then

$$u^{C}_{C_2} \ngeq 0 .$$

Theorem 3. - Denoting by d^T any row of C_2 and by D the matrix

$$D = \begin{bmatrix} C_1 \\ \\ d^T \end{bmatrix}$$

it follows:

a) $d^T p_{C_1} \gtreqless 0 \quad \Longleftrightarrow \quad u^D_d \gtreqless 0$

b) $u^D_d = 0 \ (d^T p_{C_1} = 0) \quad \Longrightarrow \quad p - p_D = p - p_{C_1}$

c) $u^D_d \neq 0 \ (d^T p_{C_1} \neq 0) \quad \Longrightarrow \quad \|p - p_{C_1}\| > \|p - p_D\|$.

3 - Methods for computing the direction.

We now describe briefly three methods for the computation of the search direction. The first method solves the problem (5), the second and third ones compute feasible directions as suggested by Zoutendijk [16] and Rosen [9].

The first method constructs a sequence $\{s^{(i)}\}$ such that the sequence $\{\|p-s^{(i)}\|\}$ is not increasing. At the i-th step it operates as follows:

a) it computes the projection p_{C_1} of p onto k constraints $C_1 s=0$ active in $s^{(i)}$. Let be $y = s^{(i)} + p$.

b) if $C\, p_{C_1} \leq 0$ and $u^{C_1} \geq 0$, then p_{C_1} is the solution of the problem (5);

c) if $C\,p_{C_1} \nleq 0$ it computes:

$$\beta = \min_{j \in J} \frac{-a_j^T s^{(i)}}{a_j^T (p_{C_1} - s^{(i)})} \qquad J = \{j : a_j^T p_{C_1} > 0\}$$

and

$$s^{(i+1)} = s^{(i)} + \beta\,(p_{C_1} - s^{(i)})$$

d) if $C\,p_{C_1} \leq 0$ and $u^{C_1} \ngeq 0$, then it deletes the constraints with the least negative Lagrangian multiplier and puts $s^{(i+1)} = y$.

At the first step, referring to Corollary 2, p is projected onto a constraint not verified by p.

On the basis of the preceding theorems, it can be proved that this method reaches the solution of (5) in a finite number of steps [1].

The second method works like the first one except for step d) which is modified as follows:

d) if $C\,p_{C_1} \leq 0$ and $u^{C_1} \ngeq 0$ and $p_{C_1} \neq 0$ then p_{C_1} is the looked for feasible direction; if $p_{C_1} = 0$ then it deletes the constraint with the least negative Lagrangian multiplier and puts

$$s^{(i+1)} = y$$

Obviously this method computes the feasible direction in a finite number of steps.

The third method computes the direction in the following way:

a) it projects p onto the whole set of active constraints in $x^{(i)}$

b) if $p_{C_1} \neq 0$ then p_{C_1} is the looked for direction

c) if $p_{C_1}=0$ and $u^{C_1}\geq 0$, then $x^{(i)}$ is the solution of (2)

d) if $p_{C_1}=0$ and $u^{C_1}\ngeq 0$, then it deletes the constraint with the least negative Lagrangian multiplier and the looked for direction is the projection onto the remaining constraints.

In all the methods, for the computation of the search direction at a point $x^{(i)}$, the results at the previous step could be employed, as in Rosen [9], so that computation time be minimized. For the sake of stability we chose not to follow this indication: it may happen that at an intermediate step one has to project onto a constraint which does not belong to the optimal set of active constraints, but generates numerical instability.

These three methods compute at each step the projection of p onto a polyhedral cone. In [1] two numerical methods for the computation of the projection are described, the former based on matrix transformations of system (8) (by employing (10)), the latter based on Householder orthogonal transformations of problem (5).

The procedure which implement the three methods are described in the next paragraph.

4 - Procedures

We now present a set of procedures for the solution of the problem (2), which generate a sequence of feasible points $\{x^{(i)}\}$, where the function f(x) is not increasing.

The main procedure FD calls a procedure DIR which computes the search direction. DIR is a dummy and can be any of the three following:

Z1, which implements the first method

Z2, which implements the second method

R , which implements the third method

of the preceding paragraph.

All the methods project the vector $p=-\nabla f(x)$ onto a manifold generated by a submatrix of the matrix A.

The addition and deletion of a constraint in the procedure Z1, Z2, R is performed by calling a procedure PRO; PRO is a dummy and can be either one of:

GJ , which employs Gauss-Jordan like transformations

OR , which employs Householder orthogonal tranformations.

Let L be a vector of integers of order ℓ ; we denote by A_L the submatrix formed by the rows of A whose row indices are in L.

The procedure PRO computes the vector u of Lagrangian multipliers of p related to the manifold generated by A_L .

$$||A_L^T u - p|| = \min_{v \in R^{\ell}} ||A_L^T v - p||$$

4.1 - Procedure FD

comment: this procedure implements a feasible directions method [15]; if the output vector s of DIR is zero, then the output vector u is non negative and the procedure terminates because $x^{(r)}$ is the optimal point.

step 0. Let m be the number of constraints, r=0

step 1. let $x=x^{(r)}$ and consider the vector I of order m_I formed by the indices of active constraints in x

let $p=-\nabla f(x)$

step 2. if $A_I p \leq 0$ then s=p , else call procedure DIR

step 3. if s=0 then stop. Otherwise

step 4. compute $\bar{\alpha}$ such that

$$f(x+\bar{\alpha} s) = \min_{\alpha} f(x+\alpha s)$$

and put $y=x+\bar{\alpha} s$

step 5. if $a_j^T y \le b_j$ for any $j \le m$, then $\beta = \bar{\alpha}$

$$\text{else } \beta = \min\left\{ \frac{b_j - a_j^T x}{a_j^T s} \ : \ a_j^T(x + \bar{\alpha} s) > b_j \ , \ j \le m \right\}$$

step 6. put $x^{(r+1)} = x + \beta s$, $r = r+1$ and go to step 1.

4.2 - Procedure Z1

comment: this procedure implements the first method of paragraph 3. I is the vector of order m_I constructed in the procedure PD which contains the indices of active constraints.

step 0. put s=0 , h=0 , k=1

let $j \in I$ such that $a_j^T p > 0$

comment: in the following step the procedure PRO is called to compute the Lagrangian multipliers of p related to a subset of the active constraints. The input values of h and k for the procedure PRO indicate the number of constraints involved respectively at the previous and at the present step; if h<k the j-th constraint will be added, if h>k the j-th constraint will be deleted. The output values of k and h coincide. In the procedure PRO the constraints are so rearranged that the first k constraints are those onto which it is projected; the components of the vector I are exchaged accordingly.

step 1. call procedure PRO,

let K be the subvector of the first k elements of I; compute

$$p_1 = p - A_K^T u$$

step 2. if $a_j^T p_1 \le 0$, for any $j \le m_I$, then go to step 4. Otherwise

step 3. compute j such that

$$\beta = \frac{-a_j^T s}{a_j^T (p_1 - s)} = \min \left\{ \frac{-a_i^T s}{a_i^T (p_1 - s)} \; : \; a_i^T p > 0 \; , \; i \le m_I \right\}$$

$s = s + \beta (p_1 - s)$, $k = k+1$,

go to step 1 (where an adding step is performed because $h<k$).

step 4. let be j, $u_j = \min_{i \in K} u_i$

if $u_j \ge 0$ then $s = p_1$, return

else $k=k-1$ and go to step 1 (where a deleting step is performed because $h>k$).

4.3 - Procedure Z2

comment: this procedure implements the second method of paragraph 3. It coincides with the procedure Z1 save for the step 4 which is replaced by the steps:

step 4. if $p_1 \ne 0$ then $s = p_1$, return. Otherwise

step 5. let be j, $u_j = \min_{i \in K} u_i$

if $u_j \ge 0$ then $s = 0$, return

else $k=k-1$ and go to step 1 (where a deleting step is performed because $h>k$).

4.4 - Procedure R

comment: this procedure implements the third method of paragraph 3.

step 0. put $s=0$, $h=0$, $k=1$

comment: in the following step the procedure PRC is called to compute Lagrangian multipliers of p related to the whole set of active constraints. As for h and k , see comment preceding step 1 of the procedure Z1.

step 1. for $j=1$ to m_I

call procedure PRO, put $k=k+1$

end j

$K=I$

step 2. put $p_1 = p - A_K^T u$

step 3. if $p_1 \neq 0$ then $s = p_1$, return. Otherwise

step 4. let be j, $u_j = \min_{i \in K} u_i$

if $u_j \geq 0$ then $s = 0$, return. Otherwise

comment: in the following step the procedure PRO is called to delete the j-th constraint.

step 5. put $k=k-1$,

call procedure PRO,

let K be the subvector of the first k elements of I and go to step 2.

4.5 - Procedure GJ

comment: this procedure uses elementary Gauss-Jordan matrices for adding and deleting the j-th costraint; as for h and k see the comment preceding step 1 of procedure Z1. The first k columns of the product of the elementary matrices are stored in the first k columns of the output matrix M.

step 0. if h=0 then $M = [A_I A_I^T \mid A_I p]$

step 1. let q be such that $I_q = j$

if h<k then h=k

step 2. if q≠h then exchange rows and columns of indices q and h of the matrix M and the elements of indices q and h in vector I.

step 3. put

$$m_{rs} = \begin{cases} \dfrac{1}{m_{hh}} & \text{if } r=s=h \\ -\dfrac{m_{rh}}{m_{hh}} & \text{if } r\neq h\ ,\ s=h \\ \dfrac{m_{hs}}{m_{hh}} & \text{if } r=h\ ,\ s\neq h \\ m_{rs} - \dfrac{m_{rh}}{m_{hh}} m_{hs} & \text{if } r\neq h\ ,\ s\neq h \end{cases}$$

u is the vector formed by the first h elements of the last column of M

step 4. h=k , return

4.6 - Procedure OR

comment: this procedure uses Householder orthogonal matrices both for adding and deleting the j-th constraint; as for h and k see the comment preceding step 1 of the procedure Z1. When deleting a constraint the construction of the orthogonal matrix is simplified because the first k columns of M are in Hessemberg form.

step 0. if $h=0$ then $M = [\, A_I \mid p \,]$

step 1. let q be such that $I_q = j$

if $k<h$ then go to step 3.

step 2. $h=k$

if $q \neq h$ then exchange the rows of indices q and h of M and the elements of indices q and h of the vector I;

compute $\sigma = \sqrt{\sum_{r=h}^{n} m_{rh}^2}$, $\beta = \dfrac{1}{\sigma\,(\sigma + |m_{hh}|)}$

$$u_r = \begin{cases} 0 & \text{if } r<h \\ \operatorname{sgn}(m_{hh})(\sigma + |m_{hh}|) & \text{if } r=h \\ m_{rh} & \text{if } r>h \end{cases}$$

$M = M - u\,(\beta\, u^T M)$

and go to step 6

step 3. $h=k$

step 4. for $r=q$ to k exchange the columns of indices r and r+1 in M and the elements of indices r and r+1 in I

step 5. for $r=q$ to k do

$$\sigma = \sqrt{m_{rr}^2 + m_{r+1,r}^2} \quad , \quad \beta = \frac{1}{(\sigma + |m_{rr}|)\,\sigma}$$

$$u_s = \begin{cases} \mathrm{sgn}(m_{rr})(\sigma + |m_{rr}|) & \text{if } s=r \\ m_{r+1,r} & \text{if } s=r+1 \\ 0 & \text{otherwise} \end{cases}$$

$$M = M - u(\beta u^T M)$$

end r

step 6. compute the solution of the system Tu=t, where T is the triangular submatrix formed by the first h columns of M and t is the vector formed by the first h elements of the last column of M

step 7. return

5 - Numerical results

The parameters, from which the efficiency of the methods depends, are in a large number and can not be easily checked: for this reason, to find some significant indications, we limit the numerical experiments to dimensionally small (n=10) quadratic problems with structured matrices and linear constraints, whose objective function depends on one parameter; when its value decreases, the condition number of the matrix of the quadratic form increases. Also the constraints matrix depends on one parameter, by which the active constraints at the optimal point can be made almost linearly dependent.

We do not consider non-quadratic convex problems, because they would require an additional testing of unidimensional minimization techniques.

In the experiments we consider two parameters: the average time required by a method to execute a step and a suitable factor which measures the rate of convergence.

5.1 - The test problems we consider are of the form

$$\min \{ x^T H x + h^T x : Ax \leq b \}$$

where $x \in \mathbb{R}^n$, $n=10$, H is a positive definite matrix of order n, A is a square matrix of order $m=10$:

$$H_{ij} = \begin{cases} y & \text{for } i=j \\ -1 & \text{for } |i-j|=1 \\ 0 & \text{otherwise} \end{cases} \qquad ,i,j=1,\ldots,n$$

$$A_{ij} = \begin{cases} w+1 & \text{for } i=j \\ 1 & \text{otherwise} \end{cases} \qquad ,i,j=1,\ldots,m$$

where y and w are suitable real numbers.

The set K of indices of the active constraints at the optimal point is fixed and a point $x^* \neq 0$ is randomly generated; the vectors b and h are determined, as described in [10], so that x^* be the optimal point and the point $x=0$ be feasible.

Taking the quotient

$$\text{cond}(H) = \frac{\mu_{max}}{\mu_{min}}$$

where μ_{max} and μ_{min} are the maximum and the minimum eigenvalue of H, as the condition number of H, we have

$$\text{cond}(H) = \frac{y + 2\cos(\pi/11)}{y - 2\cos(\pi/11)} \quad .$$

We then define the constraint degeneration parameter as the maximum minor of maximal rank of the submatrix A_K

$$\deg(A_K) = w^{k-1}(w + k)$$

where k is the number of active constraints at x^*.

We take into account the following values for the parameters

$k = 2 , 5 , 7 , 8 , 9 , 10$

$y = 2.5 , 2.0 , 1.96 , 1.92$

$w = (1/2)^i , i=0,...,5$

Thus we obtain problems whose condition number varies from 7.6 to 3490, while the degeneration parameter varies from 11 to 2.85 10^{-13}.

The procedure FD and the subprocedures Z1,Z2,R,GJ,CH are implemented in double precision on the IBM 370/168.

5.2 - For any problem and method, taking as the initial point the point $x^{(0)}=0$, and putting a bound to the maximum iteration number, we measure the execution time not including the input/output operations. Hence we compute the iteration time by dividing the execution time by the number of performed iterations and, for each method and each k value, letting y and w vary, we compute the average iteration time. We do not consider the time length for those problems where the projection of the gradient can not be computed by the procedure GJ, being too low the value of the degeneration parameter of the constraints at the optimum [1].

k	Z1		Z2		R	
	OR	GJ	OR	GJ	OR	GJ
2	.51	.46	.50	.46	.50	.46
5	.81	.73	.81	.73	.74	.65
7	1.00	.94	1.00	.93	.85	.78
8	1.03	.96	1.01	.96	.94	.88
9	1.03	1.00	1.03	.99	.94	.89
10	1.10	1.05	1.10	1.05	.91	.86

Table 1 Average iteration times in hundredths of a sec.

Although the dimension of the problems is small, it can be noted from the table that the average iteration times given by the procedure GJ are lower than those given by the procedure OR, independently from the procedures Z1, Z2, R.

The procedure Z1 requires generally more time for an iteration, and this can be explained by the fact that the number of intermediate cycles in the procedure Z1 can be much greater than in the other procedures. The number of controls required by the procedure Z2 seems to have a negative influence on its efficiency with respect to the procedure R.

5.3 - Let x^* be the optimal point of a test problem and let the convergent to x^* sequence, $\{x^{(i)}\}$, be obtained by the iterative method (3) starting from a feasible initial point $x^{(0)}$ (we assume that $x^{(0)} \neq x^*$). As a convergence factor of the sequence $\{x^{(i)}\}$, on the basis of the results of [2] and [8], we consider as particularly significant the root-convergence factor, or $R_1\{x^{(i)}\}$-factor, [6]:

$$R_1\{x^{(i)}\} = \limsup_{i \to \infty} \|x^{(i)} - x^*\|^{1/i}$$

At any step of each method the i-th root

$$r_i = \left(\|x^{(i)} - x^*\| / \|x^{(0)} - x^*\|\right)^{1/i}$$

is computed. The sequences $\{r_i\}$ so obtained have $R_1\{x^{(i)}\}$ as upper limit.

5.4 - From the results of the experiments it comes out that, except when the degeneration parameter is too low, the optimal set of constraints is reached easily enough. Afterwards the method behaves like a gradient method on the manifold generated by the active constraints. Hence if $k=n$, after finding the optimal set, the optimal point is immediately reached; if $k=n-1$, after finding the optimal set, the optimal point is reached by only one step of unidimensional minimization (in both cases the $R_1\{x^{(i)}\}$ is zero); if $k \leq n-2$ the sequence $\{r_i\}$ has $0 < R_1\{x^{(i)}\} < 1$. Figures from 1 to 9 show some of the most significant graphs for the values $k=5,8,9$, which illustrate the variations of the sequence $\{r_i\}$ with the condition number and the degeneration number, with the aim of specifying the parameters on which the evaluation is based.

The sequence $\{r_i\}$ plotted in figures from 1 to 9 were obtained by procedure Z1 with OR (the sequences obtained by the other procedures have a similar behaviour) for the values of the condition number cond(H)=7.6 and cond(H)=3490.

The figures 1 and 2, obtained for k=5, show the typical behaviour of the sequence $\{r_i\}$ when k<n-2: in this case, after an initial stage when the method finds the optimal set of constraints, the sequences settle down and show a behaviour analogous to that of a steepest descent method on the manifold generated by the constraints on which it is projected, with a linear convergence order. From the graphs we find that the degeneration parameter has an influence on the sequence $\{r_i\}$ only for very low values.

The figures 7 and 8, obtained for k=9, show the typical behaviour of the sequence $\{r_i\}$ when k=n-1: in this case, after the initial stage when the method finds the optimal set of constraints, only one iteration is required to reach the optimal point. From the graphs we find that the degeneration parameter has an influence on the number of iterations of the initial stage.

In figures 4 and 5, obtained for k=8, we find plotted the sequences $\{r_i\}$ when k=n-2: in this case the method shows a hybrid behaviour: the initial stage is similar to that for k=n-1, but once found the optimal set of constraints, the sequence $\{r_i\}$ has a non-zero upper limit and shows an oscillatory behaviour, caused by the accumulation of the rounding errors.

The figures 3, 6 and 9 show the sequences $\{r_i\}$ for k=5, k=8 and k=9 when the condition number varies. In our opinion the condition number actually influencing the convergence factor is that, not of the matrix H, but of the matrix of the quadratic form obtained by considering the problem on the manifold generated by the intersection of the optimal constraints set.

5.5 - Comparing the procedures OR and GJ from the point of view of the sequences $\{r_i\}$, we find that there can be considerable differences only for low values of the degeneration parameter, in accordance with what reported in [1], [7].

As for the procedures Z1, Z2 and R, we find that no considerable differences can be detected when k=2 and k=5. For k≥7, the procedure R finds the optimal constraints set faster: we think that this is due to the fact that it implements an implicit antizigzagging procedure. This explains the fact that for k=9 and k=10 the sequences $\{r_i\}$ relative to the procedure R converge in a smaller number of iterations (see table 2, where the numbers of iterations required by the method Z1, Z2 and R are compared for different values of w and y; see also figure 10 where the sequences $\{r_i\}$ of the three methods are plotted in the case: k=9, y=1.96).

On the other hand the procedure R, bound to project on the whole set of active constraints at a point, for low values of the degeneration parameter, is influenced by the rounding errors more than the procedures Z1 and Z2, which, by projecting on a subset of active constraints, can recognize as verified the remaining ones: this fact is particularly evident when k<n-1 (see figure 11).

W	y	Z1	Z2	R
1	2.5	10	10	10
	1.92	11	11	11
1/2	2.5	11	11	11
	1.92	13	12	13
1/4	2.5	14	14	12
	1.92	17	16	13
1/8	2.5	13	13	13
	1.92	15	15	15
1/16	2.5	14	14	12
	1.92	20	21	21
1/32	2.5	15	15	15
	1.92	24	23	20

Table 2: number of iterations required by the methods Z1, Z2 and R for some values of the condition number and the degeneration parameter.

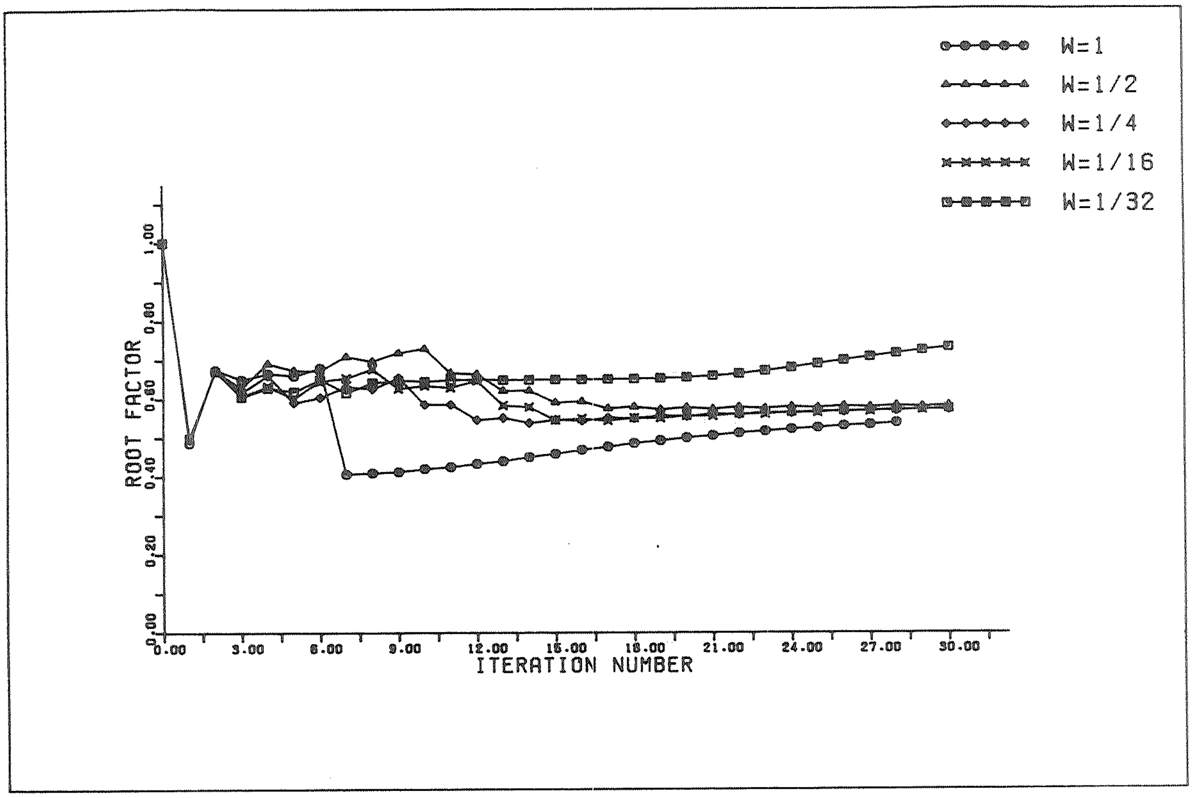

Fig. 1: sequences $\{r_i\}$ for k=5, y=2.5, w=1,1/2,1/4,1/16,1/32

(in abscissa the iteration index i)

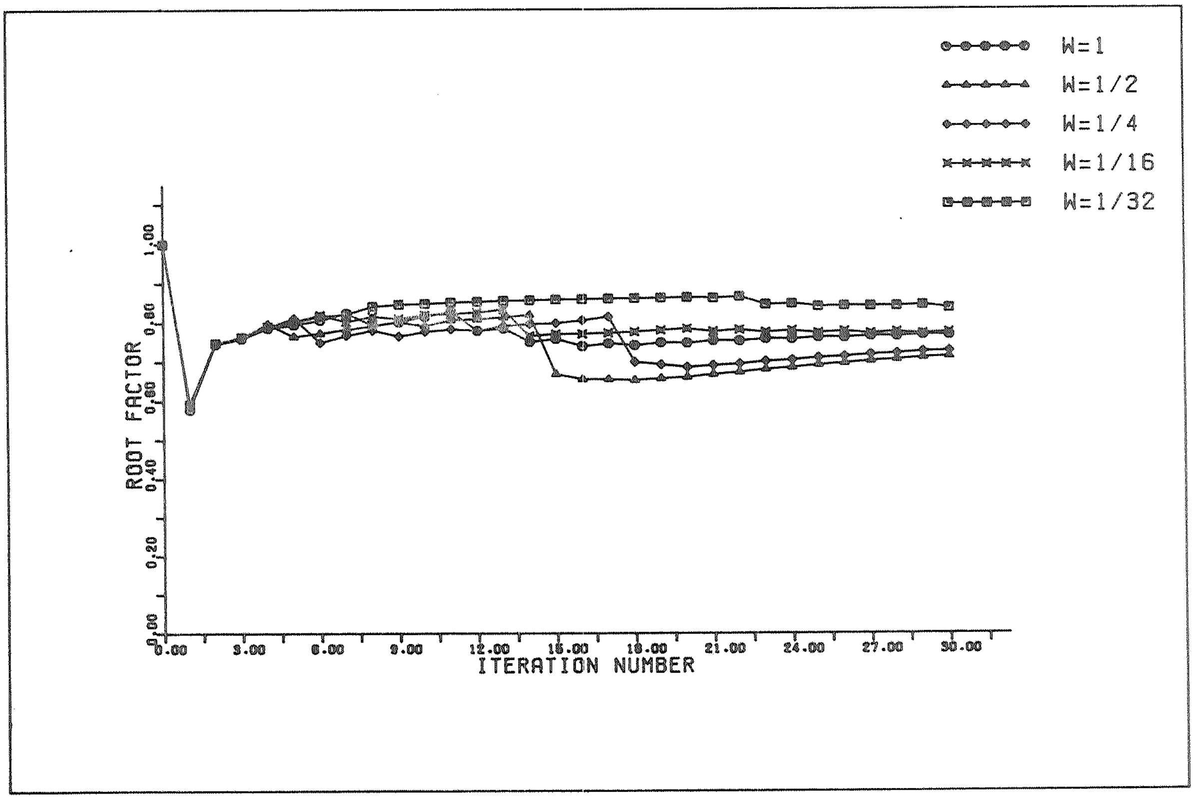

Fig. 2: sequences $\{r_i\}$ for k=5, y=1.92, w=1,1/2,1/4,1/16,1/32

(in abscissa the iteration index i)

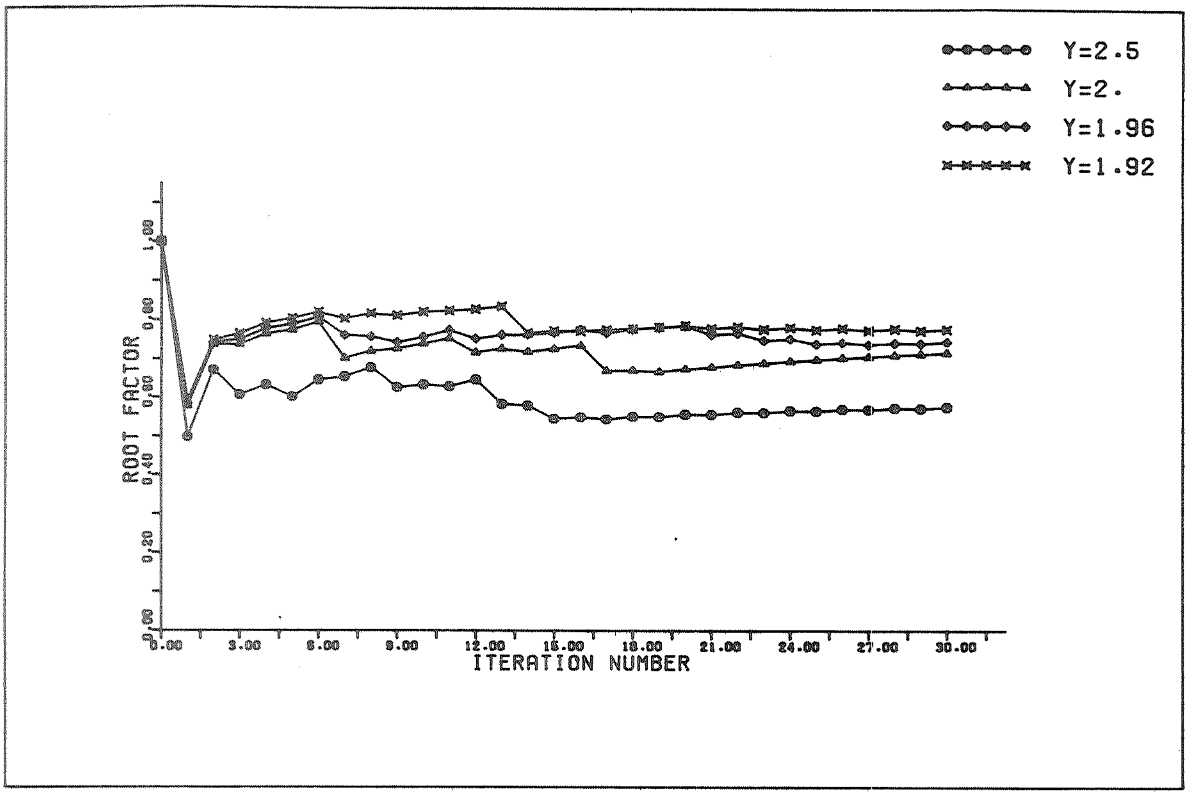

Fig. 3: sequences $\{r_i\}$ for k=5, w=1/16, y=2.5,2.,1.96,1.92

(in abscissa the iteration index i)

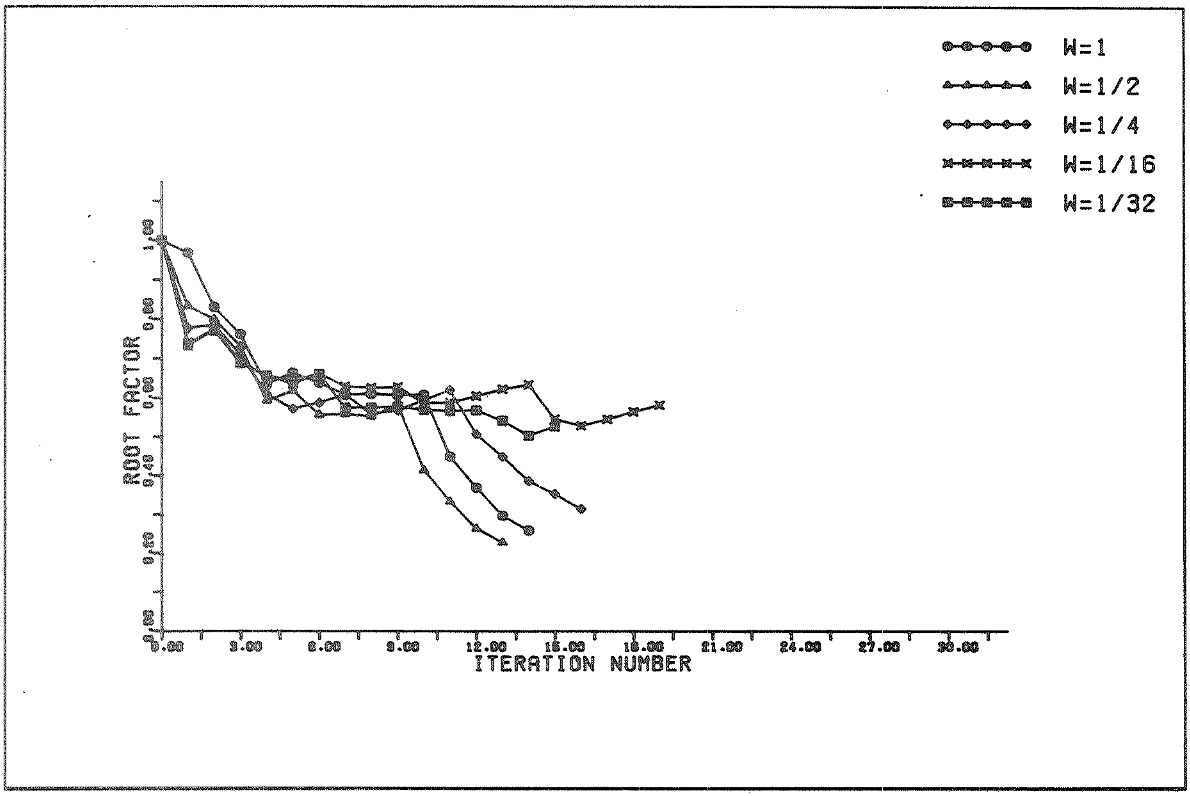

Fig. 4: sequences $\{r_i\}$ for k=8, y=2.5, w=1,1/2,1/4,1/16,1/32

(in abscissa the iteration index i)

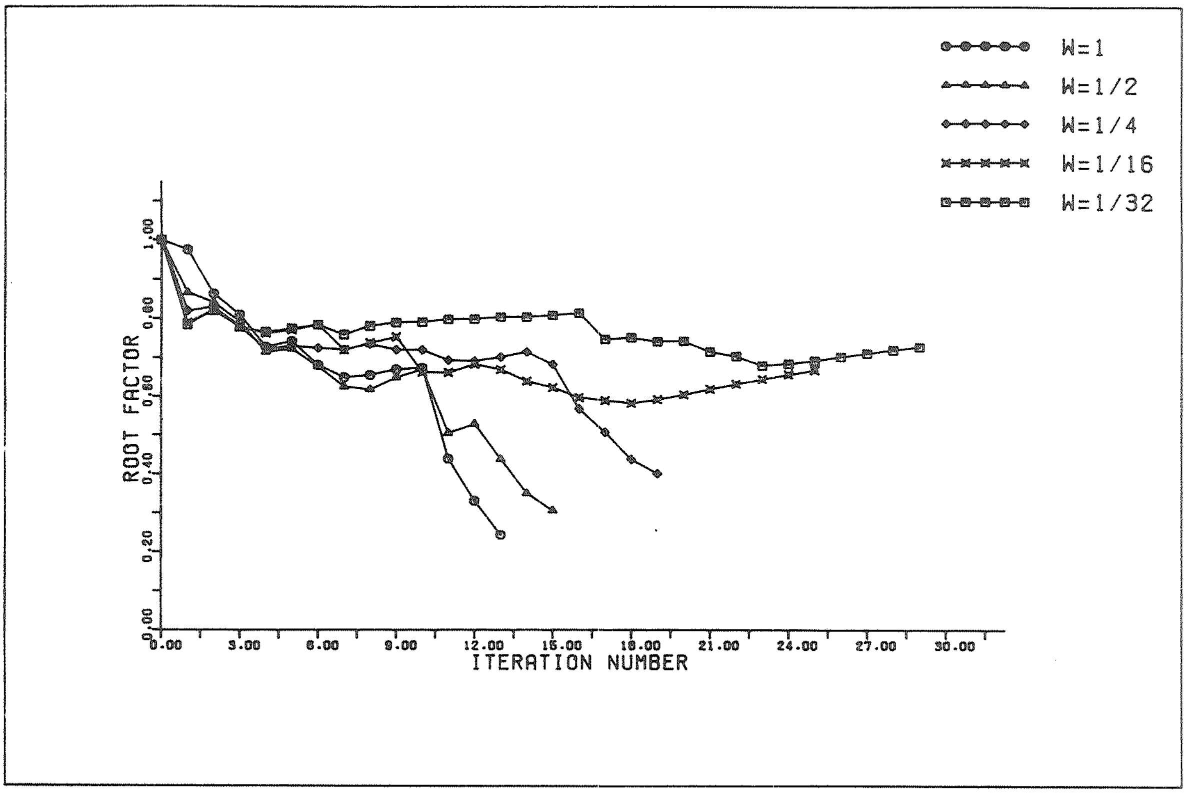

Fig. 5: sequences $\{r_i\}$ for k=8, y=1.92, w=1,1/2,1/4,1/16,1/32

(in abscissa the iteration index i)

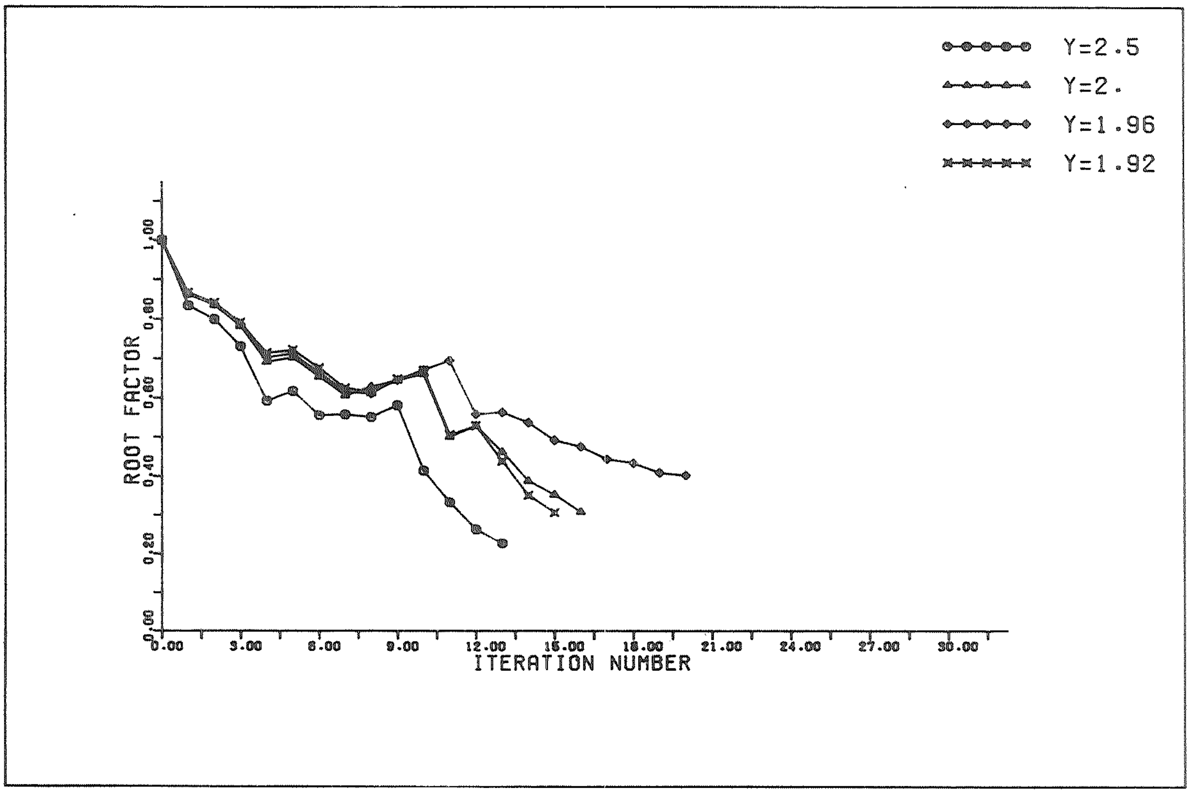

Fig. 6: sequences $\{r_i\}$ for k=8, w=1/2, y=2.5,2.,1.96,1.92

(in abscissa the iteration index i)

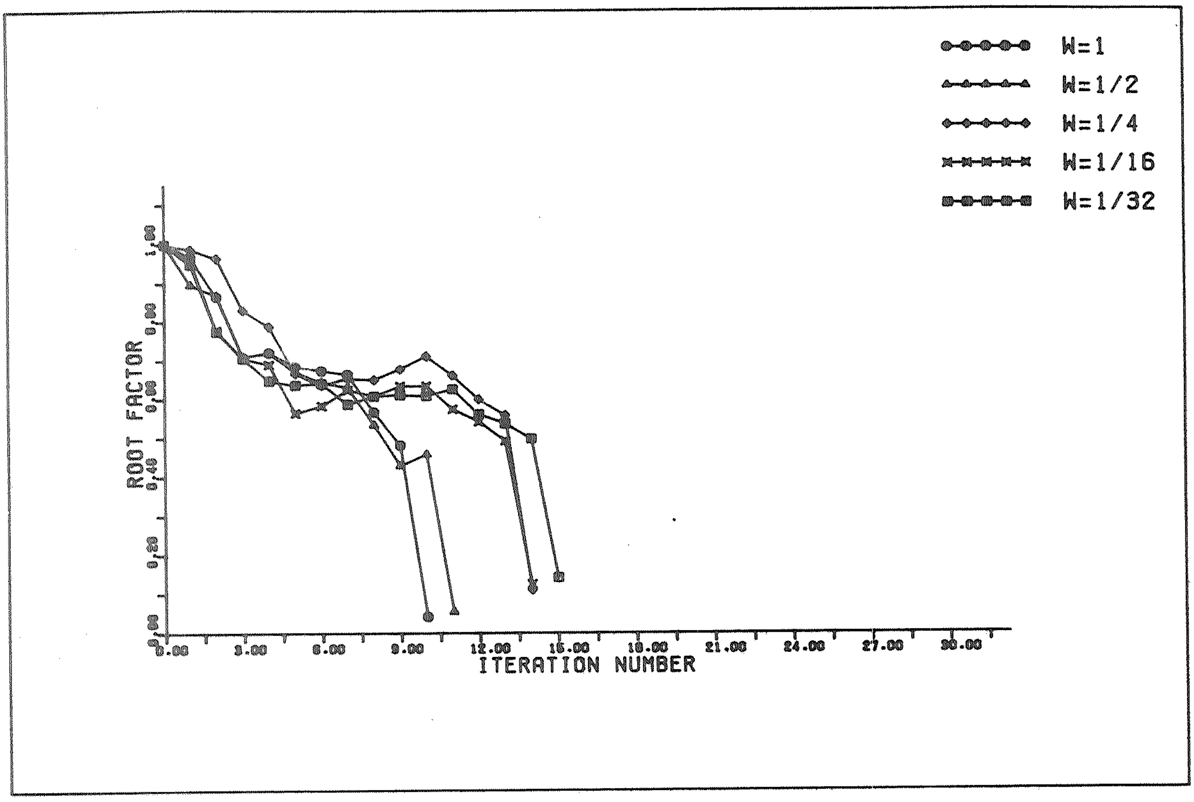

Fig. 7: sequences {r_i} for k=9, y=2.5, w=1,1/2,1/4,1/16,1/32

(in abscissa the iteration index i)

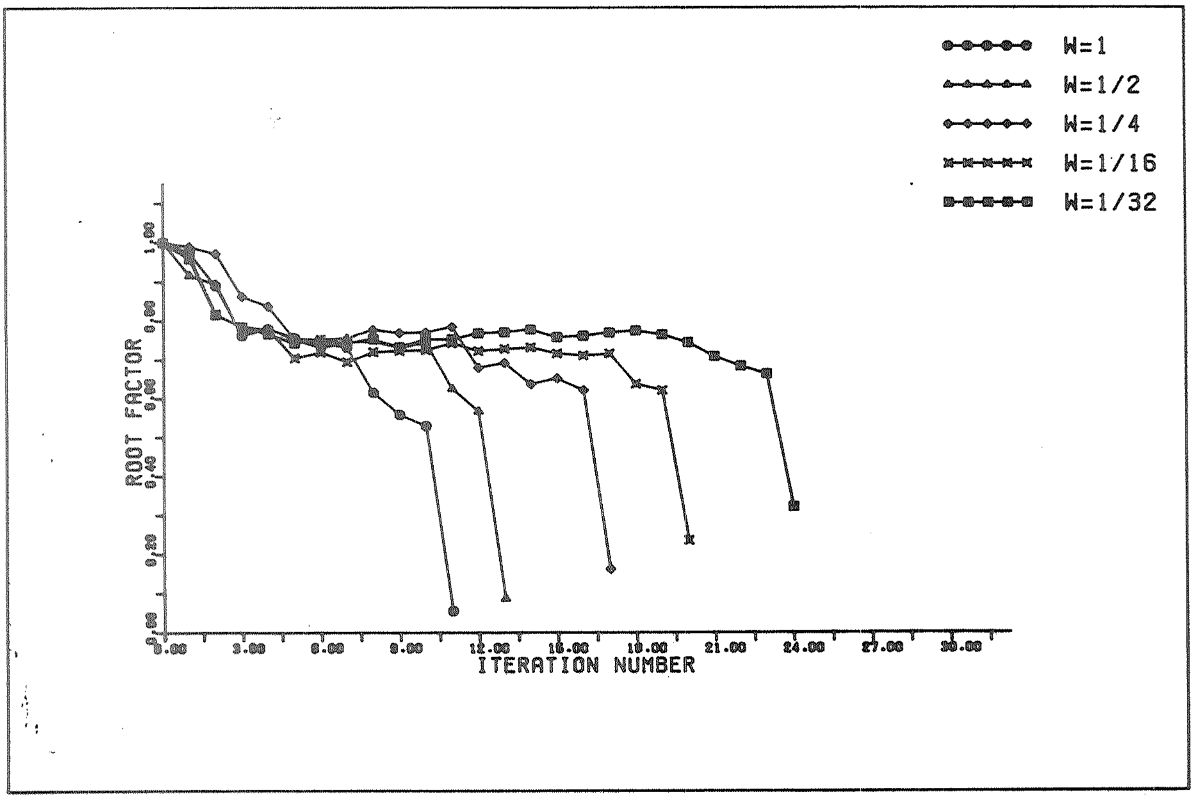

Fig. 8: sequences {r_i} for k=9, y=1.92, w=1,1/2,1/4,1/16,1/32

(in abscissa the iteration index i)

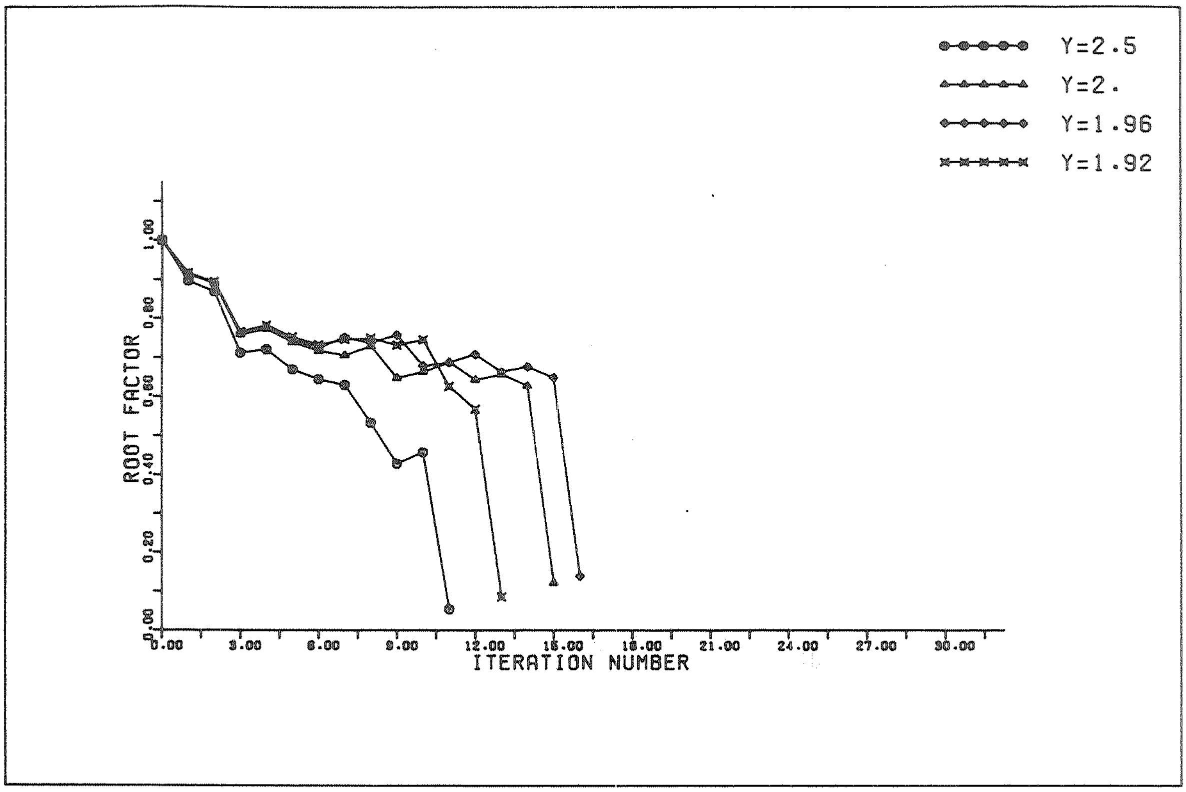

Fig. 9: sequences $\{r_i\}$ for $k=9$, $w=1/2$, $y=2.5, 2., 1.96, 1.92$

(in abscissa the iteration index i)

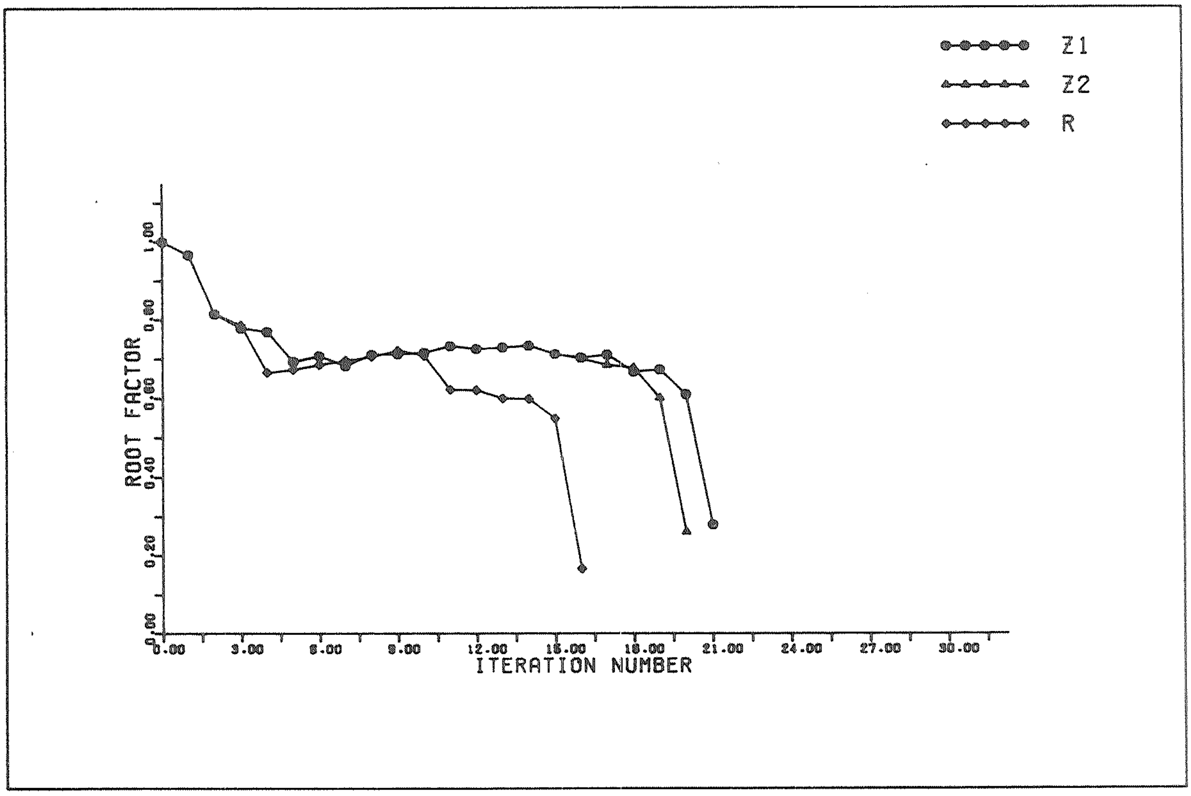

Fig. 10: sequences $\{r_i\}$ for $k=9$, $y=1.96$, $w=1/16$ by Z1, Z2, R

(in abscissa the iteration index i)

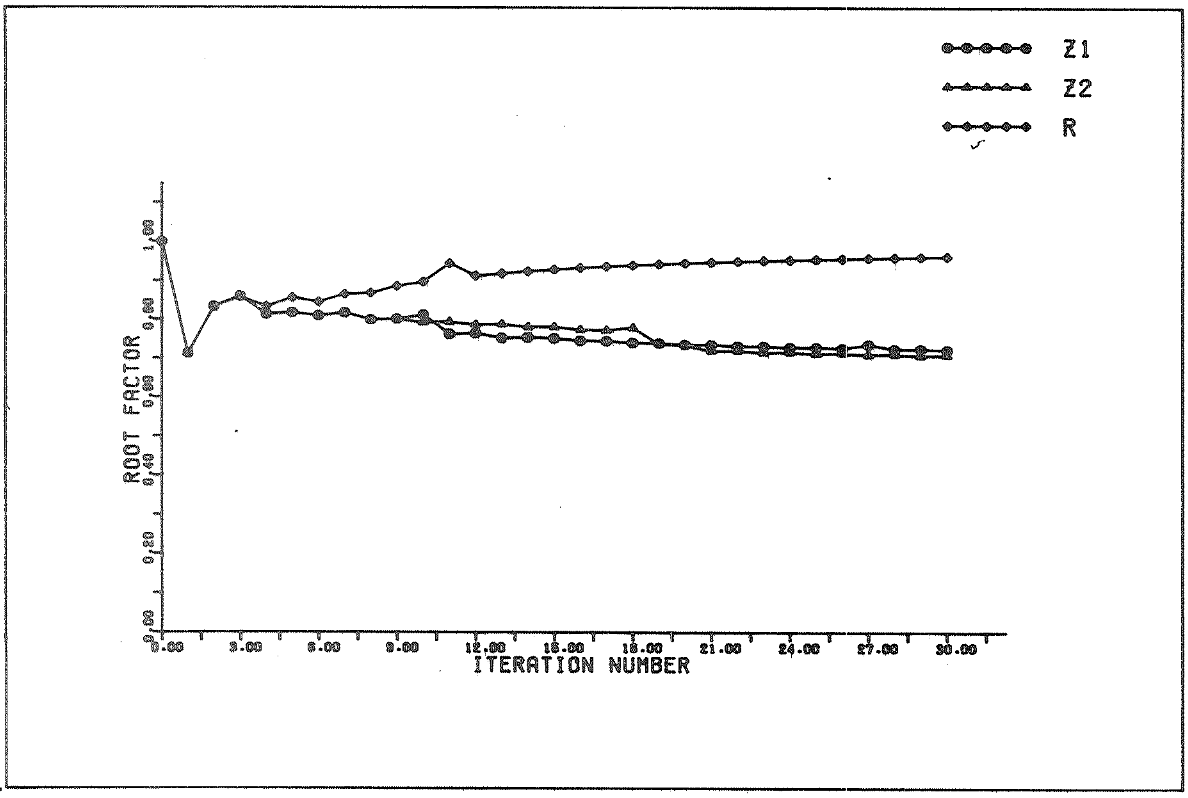

Fig. 11: sequences $\{r_i\}$ for $k=7$, $y=1.92$, $w=1$ by Z1, Z2, R

(in abscissa the iteration index i)

REFERENCES

[1] Arioli M., Laratta A., Menchi O. - ON THE COMPUTATION OF THE LEAST DISTANCE FROM A POLYHEDRAL CONVEX SET Nota interna B80-29 I.E.I. Pisa (1980)

[2] Chaney K.W. - ON THE RATE OF CONVERGENCE OF SOME FEASIBLE DIRECTION ALGORITHMS - JOTA vol.20 n.3 (1976).

[3] Golub G.H. - NUMERICAL METHODS FOR SOLVING LINEAR LEAST SQUARES PROBLEMS - Numer. Math. vol.7 (1965).

[4] Lawson C.L., Hanson R.J. - SOLVING LEAST SQUARES PROBLEMS - Prentice-Hall Publishing (1974).

[5] Mifflin R. - A STABLE METHOD FOR SOLVING CERTAIN CONSTRAINED LEAST SQUARES PROBLEMS - Math. Prog. vol.16 (1979)

[6] Ortega J. M., Rheinboldt W. C. - ITERATIVE SOLUTION OF NON LINEAR EQUATIONS IN SEVERAL VARIABLES - Academic Press (1970)

[7] Peters G., Wilkinson J.H. - ON THE STABILITY OF GAUSS-JORDAN ELIMINATION WITH PIVOTING - Communications of ACM vol.13 n.3 (1975).

[8] Pironneau O., Polak E. - RATE OF CONVERGENCE OF METHODS OF FEASIBLE DIRECTIONS - S.I.A.M. J. Num. Anal. vol.10 n.1 (1973).

[9] Rosen J.B. - THE GRADIENT PROJECTION METHODS FOR NON-LINEAR PROGRAMMING. Part 1 LINEAR CONSTRAINTS - S.I.A.M. J. Num. Anal. vol.8 n.1 (1960).

[10] Rosen J.B.,Suzuki S. - CONSTRUCTION OF NONLINEAR PROGRAMMING TEST PROBLEMS - Communications of ACM vol.8 n.2 (1965).

[11] Schittkowski K., Stoer J. - A FACTORIZATION METHOD FOR THE SOLUTION OF CONSTRAINED LINEAR LEAST SQUARES PROBLEMS ALLOWING SUBSEQUENT DATA CHANGES - Num. Math. vol.31 (1979).

[12] Stoer J. - ON THE NUMERICAL SOLUTION OF CONSTRAINED LEAST SQUARES PROBLEMS - S.I.A.M. J. Num. Anal. vol.8 n.2 (1971).

[13] Topkis D.M., Veinott A.F. - ON THE CONVERGENCE OF SOME FEASIBLE DIRECTION ALGORITHMS FOR NONLINEAR PROGRAMMING - S.I.A.M. J. Control vol.5 n.2 (1967).

[14] Wilkinson J.M. - THE ALGEBRAIC EIGENVALUE PROBLEM - Oxford University Press, London (1965).

[15] Zoutendijk G. - METHODS OF FEASIBLE DIRECTIONS - Elsevier, Amsterdam (1960).

[16] Zoutendijk G. - MATHEMATICAL PROGRAMMING METHODS - North-Holland Publishing (1976).