NUMERICAL ANALYSIS AND MATHEMATICAL MODELLING
BANACH CENTER PUBLICATIONS, VOLUME 29
INSTITUTE OF MATHEMATICS
POLISH ACADEMY OF SCIENCES
WARSZAWA 1994

ORTHOGONAL POLYNOMIALS AND THE LANCZOS METHOD

C. BREZINSKI, H. SADOK

Laboratoire d'Analyse Numérique et d'Optimisation, UFR IEEA–M3
Université des Sciences et Technologies de Lille
F-59655 Villeneuve d'Ascq Cedex, France
E-mail: BREZINSK@FRCITL81.BITNET

M. REDIVO ZAGLIA

Dipartimento di Elettronica e Informatica, Università degli Studi di Padova
Via Gradenigo 6/a, I-35131 Padova, Italy
E-mail: ELEN@ELETT1.DEI.UNIPD.IT

Abstract. Lanczos method for solving a system of linear equations is well known. It is derived from a generalization of the method of moments and one of its main interests is that it provides the exact answer in at most n steps where n is the dimension of the system. Lanczos method can be implemented via several recursive algorithms known as *Orthodir*, *Orthomin*, *Orthores*, *Biconjugate gradient*, ... In this paper, we show that all these procedures can be explained within the framework of formal orthogonal polynomials. This theory also provides a natural basis for curing breakdown and near-breakdown in these algorithms. The case of the conjugate gradient squared method can be treated similarly.

1. Lanczos method.

Let us consider in $\mathbb{C}^n$ the system of linear equations

$$Ax = b\,.$$

The Lanczos method [14] for solving this system consists in constructing the sequence of vectors (x_k) as follows:

• choose two arbitrary nonzero vectors x_0 and y,

1991 *Mathematics Subject Classification*: 65F10, 65F25.
Key words and phrases: Lanczos method, biconjugate gradient, projection, orthogonal polynomials.
The paper is in final form and no version of it will be published elsewhere.

- set $r_0 = b - Ax_0$,
- determine x_k such that

$$x_k - x_0 \in E_k = \text{span}(r_0, Ar_0, \ldots, A^{k-1}r_0)\,,$$
$$r_k = b - Ax_k \perp F_k = \text{span}(y, A^*y, \ldots, {A^*}^{k-1}y)$$

where A^* is the conjugate transpose of A.

These two conditions determine x_k if it exists. Indeed, $x_k - x_0$ can be written in the form

$$x_k - x_0 = -\alpha_1 r_0 - \ldots - \alpha_k A^{k-1} r_0$$

and the orthogonality conditions give

$$({A^*}^i y, r_k) = 0 \quad \text{for } i = 0, \ldots, k-1\,,$$

which is a system of k linear equations in the k unknowns $\alpha_1, \ldots, \alpha_k$. This system is singular if r_0, $Ar_0, \ldots, A^{k-1}r_0$ or y, $A^*y, \ldots, {A^*}^{k-1}y$ are linearly dependent.

If we set

$$P_k(\xi) = 1 + \alpha_1 \xi + \ldots + \alpha_k \xi^k$$

then we have

$$r_k = P_k(A) r_0\,.$$

Moreover, if we set

$$c_i = (y, A^i r_0), \quad i = 0, 1, \ldots$$

and if we define the linear functional c on the space of polynomials by

$$c(\xi^i) = c_i, \quad i = 0, 1, \ldots$$

then the preceding orthogonality conditions can be written as

$$c(\xi^i P_k) = 0 \quad \text{for } i = 0, \ldots, k-1\,.$$

These relations show that P_k is a polynomial of degree at most k belonging to the family of formal orthogonal polynomials with respect to c [1]. This polynomial is defined apart from a multiplying factor which is chosen, in our case, such that $P_k(0) = 1$. Due to this normalization, P_k exists and is unique if and only if the Hankel determinant

$$H_k^{(1)} = \begin{vmatrix} c_1 & c_2 & \ldots & c_k \\ c_2 & c_3 & \ldots & c_{k+1} \\ \vdots & \vdots & \ldots & \vdots \\ c_k & c_{k+1} & \ldots & c_{2k-1} \end{vmatrix}$$

is different from zero.

The polynomials P_k can be recursively computed in different ways which lead to the various Lanczos type algorithms known as *Orthores*, *Orthodir*, *Orthomin*, *Biores*, *Biodir*, *biconjugate gradient*, and so on. A unified presentation and derivation of all these methods can be based on the theory of formal orthogonal polynomials [8].

The main property of the Lanczos method is its finite convergence, namely that $\exists k \leq n$ such that $r_k = 0$ and $x_k = x = A^{-1}b$.

The theory of the Lanczos method has its basis in generalizations of the method of moments and of Galerkin's method, as will now be explained.

2. Theory of the Lanczos method. Let E be a vector space and E^* its algebraic dual. Let $E_k = \text{span}(u_0, \ldots, u_{k-1})$ where $u_0, \ldots, u_{k-1} \in E$ are assumed to be linearly independent. Let $L_0, L_1, \ldots, L_{k-1}$ be k linearly independent linear functionals in E^*.

Let $f \in E$ and let $R_k \in E_k$ be defined by

$$\langle L_i, R_k \rangle = \langle L_i, f \rangle \quad \text{for } i = 0, \ldots, k-1\,.$$

Let H_k be the linear mapping on E defined by

$$H_k f = R_k\,.$$

Then $H_k R_k = R_k$ and thus $H_k^2 = H_k$, which shows that H_k is a projection on E_k.

The method of moments, introduced by Vorobyev [16], is based on orthogonal projections in a Hilbert space. We shall now give its generalization as presented in [3].

It consists in constructing a linear mapping A_k on E_k such that

$$\begin{aligned} u_1 &= A_k u_0\,, \\ u_2 &= A_k u_1 = A_k^2 u_0\,, \\ &\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ u_{k-1} &= A_k u_{k-2} = A_k^{k-1} u_0\,, \\ H_k u_k &= A_k u_{k-1} = A_k^k u_0 \end{aligned}$$

with H_k as defined above. These relations completely determine the mapping A_k. Since $H_k u_k \in E_k$, we can write

$$H_k u_k = -\beta_0 u_0 - \ldots - \beta_{k-1} u_{k-1}\,,$$

that is,

$$\beta_0 u_0 + \ldots + \beta_{k-1} u_{k-1} + H_k u_k = (\beta_0 + \beta_1 A_k + \ldots + \beta_{k-1} A_k^{k-1} + A_k^k) u_0 = 0\,.$$

But we have

$$\langle L_i, u_k - H_k u_k \rangle = 0 \quad \text{for } i = 0, \ldots, k-1\,,$$

that is,

$$\beta_0 \langle L_i, u_0 \rangle + \ldots + \beta_{k-1} \langle L_i, u_{k-1} \rangle + \langle L_i, u_k \rangle = 0 \quad \text{for } i = 0, \ldots, k-1\,.$$

Thus, if we set

$$\widetilde{P}_k(\xi) = \beta_0 + \ldots + \beta_{k-1} \xi^{k-1} + \xi^k$$

then it can be proved that

$$\widetilde{P}_k(A_k) u_0 = 0\,.$$

Let us assume that A_k is invertible in E_k and solve the equation

$$A_k v_k = u_0$$

where $v_k \in E_k$.

Let P_k and V_{k-1} be two arbitrary polynomials of degree k and $k-1$ respectively, related by

$$1 - P_k(\xi) = \xi V_{k-1}(\xi)\,.$$

Thus P_k must satisfy $P_k(0) = 1$ and we have

$$u_0 - P_k(A_k)u_0 = A_k V_{k-1}(A_k)u_0\,.$$

If we take

$$P_k(\xi) = \widetilde{P}_k(\xi)/\widetilde{P}_k(0)$$

then

$$P_k(0) = 1 \quad \text{and} \quad P_k(A_k)u_0 = 0\,.$$

It follows that

$$A_k V_{k-1}(A_k)u_0 = u_0\,,$$

which shows that

$$v_k = V_{k-1}(A_k)u_0\,.$$

Let us now make the particular choice $E_k = \mathbb{C}^n$ and

$$u_i = A^i r_0, \qquad i = 0, \ldots, k\,,$$

where A is an $n \times n$ regular matrix.

Let A_k be the matrix obtained by the method of moments and let x_k be the vector defined by

$$A_k(x_k - x_0) = r_0\,.$$

The linear functional L_i is uniquely represented by a vector y_i and

$$\langle L_i, u_j\rangle = (y_i, u_j)$$

where $(\cdot,\cdot)$ denotes the usual scalar product in $\mathbb{C}^n$. H_k is the oblique projection on E_k along $F_k^{\perp}$ where $F_k = \operatorname{span}(y_0, \ldots, y_{k-1})$.

The polynomial P_k defined above is the same as that of the preceding section and

$$x_k - x_0 = V_{k-1}(A_k)r_0\,.$$

But, since V_{k-1} has degree $k-1$ and $A^i r_0 = A_k^i r_0$ for $i = 0, \ldots, k-1$, we have

$$x_k - x_0 = V_{k-1}(A)r_0\,.$$

That is,

$$Ax_k - b - (Ax_0 - b) = AV_{k-1}(A)r_0$$

or

$$r_k = r_0 - AV_{k-1}(A)r_0 = P_k(A)r_0\,.$$

Moreover, from the preceding results, we have

$$(y_i, r_k) = 0 \quad \text{for } i = 0, \ldots, k-1\,,$$

which is exactly the Lanczos method if $y_i = A^{*^i} y$ for $i = 0, \ldots, k-1$. For more details, see [2].

Let us now look at the various possibilities for computing recursively the orthogonal polynomials P_k.

3. Orthogonal polynomials. We shall first consider the case where $\forall k$, $H_k^{(1)} \neq 0$.

Let us define the linear functional $c^{(1)}$ by

$$c^{(1)}(\xi^i) = c(\xi^{i+1}) = c_{i+1}, \qquad i = 0, 1, \ldots$$

Let $\{P_k^{(1)}\}$ be the family of monic orthogonal polynomials with respect to the linear functional $c^{(1)}$. $P_k^{(1)}$ exists under the condition that $H_k^{(1)} \neq 0$. As previously seen, this condition also ensures the existence of P_k. $\{P_k\}$ and $\{P_k^{(1)}\}$ are said to be *adjacent families of formal orthogonal polynomials*.

Since the polynomials of both families are uniquely determined, it is easy to see that

$$P_{k+1}(\xi) = P_k(\xi) - \lambda_k \xi P_k^{(1)}(\xi) \tag{1}$$

with $P_0(\xi) = P_0^{(1)}(\xi) = 1$.

We now show how to obtain an expression for λ_k. Let $\{U_i\}$ be an arbitrary family of polynomials such that $\forall i$, U_i has exact degree i. Multiplying both sides of (1) by U_k and applying the functional c, we obtain

$$\lambda_k = c(U_k P_k)/c(\xi U_k P_k^{(1)})\,.$$

Moreover, it is well known that a family of orthogonal polynomials satisfies a three-term recurrence relationship. Thus we have

$$P_k^{(1)}(\xi) = (\xi - a_k) P_{k-1}^{(1)}(\xi) - b_k P_{k-2}^{(1)}(\xi) \tag{2}$$

with $P_0^{(1)}(\xi) = 1$ and $P_{-1}^{(1)}(\xi) = 0$.

As proved in [4], using again the auxiliary family $\{U_i\}$ we have, by a similar argument,

$$b_k = c(\xi^2 U_{k-2} P_{k-1}^{(1)})/c(\xi U_{k-2} P_{k-2}^{(1)})\,,$$
$$a_k = [c(\xi^2 U_{k-1} P_{k-1}^{(1)}) - b_k c(\xi U_{k-1} P_{k-2}^{(1)})]/c(\xi U_{k-1} P_{k-1}^{(1)})\,.$$

For example, if we choose $U_k = P_k^{(1)}$, we recover the usual formula

$$a_k = c(\xi^2 P_{k-1}^{(1)^2})/c(\xi P_{k-1}^{(1)^2})\,.$$

The family $\{P_k\}$ also satisfies a three-term recurrence relationship which can be written as

$$P_{k+1}(\xi) = -D_k[(\xi - B_k)P_k(\xi) - C_kP_{k-1}(\xi)] \tag{3}$$

with $P_0(\xi) = 1$ and $P_{-1}(\xi) = 0$. Since $P_k(0) = 1$, we must have

$$D_k = 1/(B_k + C_k)\,.$$

Using again the auxiliary family $\{U_i\}$, as above, we have

$$c(U_{k-1}P_{k+1}) = 0 = -D_k[c(\xi U_{k-1}P_k) - C_kc(U_{k-1}P_{k-1})]$$

and thus

$$C_k = c(\xi U_{k-1}P_k)/c(U_{k-1}P_{k-1})\,.$$

Similarly,

$$c(U_kP_{k+1}) = 0 = -D_k[c(\xi U_kP_k) - B_kc(U_kP_k) - C_kc(U_kP_{k-1})]$$

and thus we obtain

$$B_k = [c(\xi U_kP_k) - C_kc(U_kP_{k-1})]/c(U_kP_k)\,.$$

Again, for the particular choice $U_k = P_k$, we recover the usual formulae.

Assume that $H_k^{(0)} \neq 0$ and set

$$Q_k(\xi) = (-1)^k H_k^{(0)} P_k^{(1)}(\xi)/H_k^{(1)}\,.$$

Since P_k and $P_k^{(1)}$ both have degree k exactly, the coefficients of ξ^k in P_k and Q_k are the same and Q_k is proportional to $P_k^{(1)}$. Moreover,

$$c(\xi^{i+1}Q_k) = 0 \quad \text{for } i = 0, \ldots, k-1\,.$$

By using the same uniqueness argument as above we can write

$$Q_k(\xi) = P_k(\xi) + \alpha_kQ_{k-1}(\xi)\,, \tag{4}$$

$$P_{k+1}(\xi) = P_k(\xi) - \beta_k\xi Q_k(\xi) \tag{5}$$

with

$$\alpha_k = -c(\xi U_{k-1}P_k)/c(\xi U_{k-1}Q_{k-1}) \quad \text{and} \quad \beta_k = c(U_kP_k)/c(\xi U_kQ_k)\,.$$

4. Lanczos type algorithms. We shall now examine the various possibilities for computing recursively the vectors r_k defined by $r_k = P_k(A)r_0$. They give rise to the different methods which are known and some new ones can also be obtained as we shall see below.

4.1. *Lanczos/Orthodir*. Let us set

$$z_k = P_k^{(1)}(A)r_0\,.$$

From (1) we immediately obtain

$$r_{k+1} = r_k - \lambda_kAz_k\,.$$

Using $r_k = b - Ax_k$, this gives

$$x_{k+1} = x_k + \lambda_k z_k \,.$$

From (2) we have

$$z_k = Az_{k-1} - a_k z_{k-1} - b_k z_{k-2} \,.$$

This method is called Lanczos/Orthodir.

Let us now see how to compute the coefficients λ_k, a_k and b_k appearing in these formulae or, in other words, how to choose the auxiliary polynomials $\{U_i\}$.

The simplest choice consists in taking

$$U_k(\xi) = \xi^k \,.$$

We have

$$\lambda_k = c(\xi^k P_k)/c(\xi^{k+1} P_k^{(1)}) \,.$$

But

$$c(\xi^k P_k) = (A^{*^k} y, P_k(A) r_0) \quad \text{and} \quad c(\xi^{k+1} P_k^{(1)}) = (A^{*^{k+1}} y, P_k^{(1)}(A) r_0)$$

and we finally obtain

$$\lambda_k = (A^{*^k} y, r_k)/(A^{*^{k+1}} y, z_k) \,.$$

Similarly, we have

$$b_k = (A^{*^k} y, z_{k-1})/(A^{*^{k-1}} y, z_{k-2})$$

and

$$a_k = [(A^{*^k} y, Az_{k-1}) - b_k (A^{*^{k-1}} y, Az_{k-2})]/(A^{*^k} y, z_{k-1}) \,.$$

Now let us make the choice

$$U_k(\xi) = P_k^{(1)}(\xi) \,.$$

We have

$$\lambda_k = c(P_k^{(1)} P_k)/c(\xi P_k^{(1)^2}) \,.$$

Thus if we set

$$\widetilde{z}_k = \overline{P}_k^{(1)}(A^*) y = P_k^{(1)}(A)^* y$$

we obtain

$$\lambda_k = (\widetilde{z}_k, r_k)/(\widetilde{z}_k, Az_k) \,.$$

We also have

$$a_k = \frac{c(\xi^2 P_{k-1}^{(1)^2})}{c(\xi P_{k-1}^{(1)^2})} = \frac{(A^* \widetilde{z}_{k-1}, Az_{k-1})}{(\widetilde{z}_{k-1}, Az_{k-1})}$$

and then

$$b_k = \frac{c(\xi^2 P_{k-2}^{(1)} P_{k-1}^{(1)})}{c(\xi P_{k-2}^{(1)^2})} = \frac{(A^* \widetilde{z}_{k-2}, Az_{k-1})}{(\widetilde{z}_{k-2}, Az_{k-2})} \,.$$

But $c(\xi^2 P_{k-2}^{(1)} P_{k-1}^{(1)}) = c(\xi {P_{k-1}^{(1)}}^2)$ by the orthogonality of $P_{k-1}^{(1)}$ to any polynomial of degree strictly less than $k-1$, and thus

$$b_k = (\widetilde{z}_{k-1}, Az_{k-1})/(\widetilde{z}_{k-2}, Az_{k-2})\,.$$

Thus, we obtain

$$\widetilde{z}_k = A^*\widetilde{z}_{k-1} - \overline{a}_k\widetilde{z}_{k-1} - \overline{b}_k\widetilde{z}_{k-2}\,.$$

If we set

$$\widetilde{r}_k = P_k(A)^* r_0$$

then we have

$$\widetilde{r}_{k+1} = \widetilde{r}_k - \overline{\lambda}_k A^*\widetilde{z}_k\,.$$

If we gather all these formulae together, we obtain the algorithm known under the name of BIODIR [12].

R e m a r k. This algorithm is defined (that is, x_{k+1} exists) only if $P_k^{(1)}$ has exact degree k, or, in other words, if $\forall k$, $H_k^{(1)} \neq 0$. If this condition is not satisfied, it is possible to jump over the non-existing polynomials $P_k^{(1)}$ and to use only those which exist. Thus we obtain a generalization of BIODIR without breakdown. This algorithm, called the MRZ (Method of Recursive Zoom), was given in [6]. Related algorithms can also be found in [12]. They will be briefly described in Section 5.

4.2. *Lanczos/Orthores.* Let us now assume that P_k has exact degree k. Thus the three-term recurrence relationship (3) holds and we immediately obtain

$$r_{k+1} = -D_k(Ar_k - B_k r_k - C_k r_{k-1})\,.$$

Since $r_k = b - Ax_k$, we have

$$x_{k+1} = D_k(r_k + B_k x_k + C_k x_{k-1})\,.$$

These formulae define the method known under the name of Lanczos/Orthores (see [17]).

Let us now see how to compute the coefficients D_k, B_k and C_k appearing in these formulae. There are several possibilities according to the choice of the auxiliary polynomials $\{U_i\}$.

Let us start again by taking

$$U_k(\xi) = \xi^k\,.$$

We have

$$C_k = c(\xi^k P_k)/c(\xi^{k-1} P_{k-1})$$

and hence

$$C_k = (A^{*^k} y, r_k)/(A^{*^{k-1}} y, r_{k-1})\,.$$

Similarly, we have

$$B_k = [c(\xi^{k+1} P_k) - C_k c(\xi^k P_{k-1})]/c(\xi^k P_k)\,,$$

that is,

$$B_k = [(A^{*^k} y, Ar_k) - C_k(A^{*^{k-1}} y, Ar_{k-1})]/(A^{*^k} y, r_k)\,.$$

Since P_k has exact degree k, we can also make the choice

$$U_k(\xi) = P_k(\xi)\,.$$

In that case

$$C_k = c(\xi P_{k-1} P_k)/c(P_{k-1}^2)\,.$$

But, multiplying both sides of (3) by P_{k+1} and applying c, we obtain

$$c(P_{k+1}^2) = -D_k c(\xi P_k P_{k+1})$$

and thus

$$C_k = -\frac{1}{D_{k-1}} \frac{c(P_k^2)}{c(P_{k-1}^2)}\,.$$

If we set

$$\widetilde{r}_k = \overline{P}_k(A^*) y$$

then

$$C_k = -\frac{1}{D_{k-1}} \frac{(\widetilde{r}_k, r_k)}{(\widetilde{r}_{k-1}, r_{k-1})}\,.$$

We also have, since $c(P_{k-1}P_k) = 0$,

$$B_k = c(\xi P_k^2)/c(P_k^2)\,,$$

that is,

$$B_k = (\widetilde{r}_k, Ar_k)/(\widetilde{r}_k, r_k).$$

Finally, (3) gives us

$$\widetilde{r}_{k+1} = -\overline{D}_k(A^* \widetilde{r}_k - \overline{B}_k \widetilde{r}_k - \overline{C}_k \widetilde{r}_{k-1})\,.$$

These formulae define the algorithm known under the name of BIORES.

R e m a r k. The two algorithms described in this subsection need the supplementary assumption that P_k has exact degree k. Thus, for all k, $H_k^{(0)}$ and $H_k^{(1)}$ both have to be different from zero. The supplementary assumption $H_k^{(0)} \neq 0$ is needed by the procedure used in the recursive computation but it is, in fact, an unnecessary assumption in the theory of the Lanczos method and it is a supplementary reason for a breakdown. A remedy will be indicated in Section 5.

4.3. *Lanczos/Orthomin.* Instead of using the polynomials $P_k^{(1)}$ as in the method Lanczos/Orthores, we shall make use of the polynomials Q_k defined in Section 3. Thus P_k is again assumed to have exact degree k. We set

$$p_k = Q_k(A) r_0\,.$$

From (4) and (5) we have

$$p_k = r_k + \alpha_k p_{k-1} \quad \text{and} \quad r_{k+1} = r_k - \beta_k A p_k\,.$$

By using the definition of r_k, it follows that

$$x_{k+1} = x_k + \beta_k p_k \,.$$

These formulae define the method known under the name of Lanczos/ Orthomin (see [11]).

Let us now see how to compute the coefficients α_k and β_k appearing in these formulae. We begin by the choice

$$U_k(\xi) = \xi^k \,.$$

We have

$$\alpha_k = -c(\xi^k P_k)/c(\xi^k Q_{k-1}), \qquad \beta_k = c(\xi^k P_k)/c(\xi^{k+1} Q_k)$$

and it follows that

$$\alpha_k = -(A^{*^k} y, r_k)/(A^{*^k} y, p_{k-1}), \qquad \beta_k = (A^{*^k} y, r_k)/(A^{*^{k+1}} y, p_k) \,.$$

Let us now choose

$$U_k(\xi) = P_k(\xi) \,.$$

In that case

$$\beta_k = c(P_k^2)/c(\xi P_k Q_k) \,.$$

But, since the coefficients of ξ^k in P_k and in Q_k are the same, we have

$$P_k(\xi) = Q_k(\xi) + p(\xi)$$

where p is a polynomial of degree $k-1$ at most. Thus

$$c(\xi P_k Q_k) = c(\xi Q_k^2) + c(\xi p Q_k) \,.$$

But, as Q_k is proportional to $P_k^{(1)}$, we have, by the orthogonality property of $P_k^{(1)}$, $c(\xi p Q_k) = 0$ and thus

$$\beta_k = c(P_k^2)/c(\xi Q_k^2) \,.$$

Setting

$$\widetilde{r}_k = P_k(A)^* y$$

we obtain

$$\beta_k = (\widetilde{r}_k, r_k)/(\widetilde{p}_k, A p_k) \,.$$

We have

$$\alpha_k = -c(\xi P_{k-1} P_k)/c(\xi P_{k-1} Q_{k-1}) \,.$$

But $\beta_{k-1} = c(P_{k-1}^2)/c(\xi P_{k-1} Q_{k-1})$ and thus

$$\alpha_k = -\beta_{k-1} c(\xi P_{k-1} P_k)/c(P_{k-1}^2) \,.$$

On the other hand, writing (5) for the index k, multiplying it by P_k and applying c gives

$$c(P_k^2) = -\beta_{k-1} c(\xi P_k Q_{k-1}) \,.$$

Moreover, from (4)

$$c(\xi P_k Q_{k-1}) = c(\xi P_k P_{k-1}) + \alpha_k c(\xi P_k Q_{k-2}) \,.$$

Since P_k is orthogonal to ξQ_{k-2} which is a polynomial of degree $k-1$, it follows that

$$c(P_k^2) = -\beta_{k-1} c(\xi P_k P_{k-1})\,.$$

Thus, we finally obtain

$$\alpha_k = c(P_k^2)/c(P_{k-1}^2)\,,$$

that is,

$$\alpha_k = (\widetilde{r}_k, r_k)/(\widetilde{r}_{k-1}, r_{k-1})\,.$$

If we set

$$\widetilde{p}_k = Q_k(A)^* y$$

we obtain from (5)

$$\widetilde{r}_{k+1} = \widetilde{r}_k - \overline{\beta}_k A^* \widetilde{p}_k$$

and from (4)

$$\widetilde{p}_k = \widetilde{r}_k + \overline{\alpha}_k \widetilde{p}_{k-1}\,.$$

These formulae define the so-called *biconjugate gradient method* (BCG) due to Lanczos [14], but popularized by Fletcher [11]. When the matrix A is Hermitian and when $y = r_0$ this method reduces to the conjugate gradient method of Hestenes and Stiefel [13].

4.4. *Other methods.* All the methods are characterized by the choice of one or two recurrence relationships for computing the orthogonal polynomials. There are many other possible choices, thus leading to other (new) methods. Some of them are described in [8].

5. Breakdown and near-breakdown. Let us now assume that some of the Hankel determinants $H_k^{(1)}$ are equal to zero. In that case some of the polynomials P_k, and thus the corresponding polynomials $P_k^{(1)}$, do not exist. Then, due to a division by zero, a *breakdown* will occur in the Lanczos type algorithms described in Section 4. It is possible to avoid such a breakdown by jumping over the non-existing polynomials and considering only the existing ones which are usually called regular. We shall now change a little bit our notations and call P_k and P_{k+1} two successive regular orthogonal polynomials with respect to c, of respective degrees n_k and $n_{k+1} = n_k + m_k$ at most. Similarly, $P_k^{(1)}$ and $P_{k+1}^{(1)}$ will denote two successive regular monic orthogonal polynomials with respect to $c^{(1)}$, of respective degrees n_k and n_{k+1}.

It was proved by Draux [10] that $P_k^{(1)}$ satisfies the conditions

$$c^{(1)}(\xi^i P_k^{(1)}) = c(\xi^{i+1} P_k^{(1)}) = 0 \quad \text{for } i = 0, \ldots, n_k + m_k - 2\,,$$
$$c^{(1)}(\xi^{n_k+m_k-1} P_k^{(1)}) = c(\xi^{n_k+m_k} P_k^{(1)}) \neq 0\,.$$

These conditions determine the length m_k of the *jump*. Moreover, the polynomials $P_k^{(1)}$ satisfy the recurrence relationship

$$P_{k+1}^{(1)}(\xi) = q_k(\xi)P_k^{(1)}(\xi) - C_{k+1}P_{k-1}^{(1)}(\xi), \qquad k = 0, 1, \ldots \tag{6}$$

with $P_{-1}^{(1)}(\xi) = 0$, $P_0^{(1)}(\xi) = 1$ and q_k a monic polynomial of degree m_k.

It can also be proved that, in such a case,

$$P_{k+1}(\xi) = P_k(\xi) - \xi w_k(\xi)P_k^{(1)}(\xi) \tag{7}$$

with $P_0(\xi) = 1$ and w_k a polynomial of degree $m_k - 1$ at most.

The coefficients of the polynomials q_k and w_k and the constants C_{k+1} are obtained by imposing the orthogonality conditions on the respective polynomials.

Gathering all these formulae together we have the following algorithm called the MRZ (Method of Recursive Zoom) [6]:

- Choose x_0 and y. Set
$$r_0 = z_0 = b - Ax_0, \quad z_{-1} = 0, \quad n_0 = 0\,.$$
- For $k = 0, 1, \ldots$ compute m_k and then if $n_k + m_k \leq n$ set
$$x_{k+1} = x_k + w_k(A)z_k, \qquad r_{k+1} = r_k - Aw_k(A)z_k\,.$$
- If $r_{k+1} \neq 0$, then compute
$$n_{k+1} = n_k + m_k, \qquad z_{k+1} = q_k(A)z_k - C_{k+1}z_{k-1}\,.$$

Clearly this algorithm is a generalization of the Lanczos/Orthodir and BIODIR algorithms. It cannot suffer from breakdown except the incurable hard one which occurs if $c(\xi^n P_k^{(1)}) = 0$. The corresponding subroutine is given in [7].

If we make the supplementary assumption that $c(\xi^{n_k} P_k) \neq 0$ then it is possible to generalize some of the other algorithms given in Section 4.

As explained above, breakdown is due to the non-existence of some polynomials P_k and the remedy is to jump over these non-existing polynomials. In the methods Lanczos/Orthores and Lanczos/Orthomin, we made the supplementary assumption that P_k has exact degree k. In fact this assumption is totally unnecessary in the theory of the Lanczos method but it was required by the form of the recurrence relation used, thus being a supplementary (and unnecessary) cause for breakdown.

Now if $|c^{(1)}(\xi^{n_k+m_k-1}P_k^{(1)})|$ is different from zero but small (and possibly badly computed) the coefficients of the recurrence relations given in Section 4 could be large and badly computed and rounding errors could affect the algorithm. The same is true if the quantities $|c^{(1)}(\xi^i P_k^{(1)})|$ are not zero for $i = n_k, \ldots, n_k + m_k - 2$ but small; in that case no breakdown occurs in the method but numerical instability will be present, a situation called *near-breakdown*.

It is possible to avoid such a near-breakdown by jumping over these polynomials which could be badly computed and to compute directly the first regular polynomial following them.

Thus, let $\varepsilon \geq 0$ be given. We define $m_k \geq 1$ such that

$$|c^{(1)}(\xi^i P_k^{(1)})| \leq \varepsilon \quad \text{for } i = n_k, \ldots, n_k + m_k - 2\,,$$
$$|c^{(1)}(\xi^i P_k^{(1)})| > \varepsilon \quad \text{for } i = n_k + m_k - 1\,.$$

Let $n_{k+1} = n_k + m_k$. We shall denote by $P_{k+1}^{(1)}$ the regular orthogonal polynomial of degree n_{k+1} with respect to $c^{(1)}$. As explained in the sequel if such a polynomial does not exist (and we are able to detect such a case) the value of m_k has to be increased until a regular polynomial has been obtained. We shall denote by P_{k+1} the corresponding orthogonal polynomial of degree n_{k+1} at most with respect to c normalized by the condition $P_{k+1}(0) = 1$.

Let us first compute P_{k+1}. We shall write it in the form

$$P_{k+1}(\xi) = P_k(\xi) - \xi w_k(\xi) P_k^{(1)}(\xi) - \xi v_k(\xi) P_k(\xi) \tag{8}$$

where w_k is a polynomial of degree at most $m_k - 1$ and v_k a polynomial of degree at most $m_k - 2$ if $n_k - m_k + 1 \geq 0$ and of degree at most $n_k - 1$ if $n_k - m_k + 1 < 0$.

Let us recall that, in the case of a breakdown, v_k is identically zero.

Let us now try to compute recursively $P_{k+1}^{(1)}$. We shall look for $P_{k+1}^{(1)}$ in the form

$$P_{k+1}^{(1)}(\xi) = q_k(\xi) P_k^{(1)}(\xi) + t_k(\xi) P_k(\xi) \tag{9}$$

where q_k is a monic polynomial of degree m_k and t_k a polynomial of degree at most $m_k - 1$ if $n_k - m_k \geq 0$ and of degree at most $n_k - 1$ otherwise.

If $c(\xi^{n_k} P_k) = 0$ this system is singular. However, this case will seldom occur and most of the time $|c(\xi^{n_k} P_k)|$ will not be equal to zero but small. Such a case can also be avoided.

If the systems giving the coefficients of the above polynomials are singular, it means that $P_{k+1}^{(1)}$ does not exist and the value of m_k has to be increased until a regular polynomial $P_{k+1}^{(1)}$ has been obtained.

Setting $r_k = P_k(A) r_0$ and $z_k = P_k^{(1)}(A) r_0$, we obtain an algorithm for avoiding near-breakdown in the Lanczos method; it was called the BSMRZ, [7].

6. The CGS. A variant of the Lanczos method was proposed by Sonneveld [15]. It is the so-called *conjugate gradient squared method* (CGS for short) which consists in taking

$$r_k = P_k^2(A) r_0\,.$$

This method obviously has the same finite convergence property as the Lanczos method. From the above formulae it is obviously possible to avoid also breakdown and near-breakdown in the CGS algorithm by squaring the above formulae. This is done in [9] for the breakdown and in [5] for the near-breakdown. As an example let us show how to treat the case of near-breakdown. We first take the relation

(8) of the BSMRZ and square it. We obtain

$$P_{k+1}^2 = (1-\xi v_k)^2 P_k^2 - 2(1-\xi v_k)\xi w_k P_k P_k^{(1)} + \xi^2 w_k^2 P_k^{(1)^2} .$$

For using this relation, it is necessary to compute recursively the polynomials $P_{k+1}^{(1)^2}$ and $P_{k+1}P_{k+1}^{(1)^2}$. Thus, let us square the relation (9) of the BSMRZ. We obtain

$$P_{k+1}^{(1)^2} = q_k^2 P_k^{(1)^2} + 2q_k t_k P_k P_k^{(1)} + t_k^2 P_k^2 .$$

Finally, multiplying (8) by (9) leads to

$$P_{k+1}P_{k+1}^{(1)} = (q_k - \xi q_k v_k - \xi t_k w_k) P_k P_k^{(1)} - \xi q_k w_k P_k^{(1)^2} + t_k(1-\xi v_k) P_k^2 .$$

Thus, if we set

$$r_k = P_k^2(A) r_0, \qquad z_k = P_k^{(1)^2}(A) r_0, \qquad s_k = P_k(A) P_k^{(1)}(A) r_0,$$

then we obtain the following algorithm called, for obvious reasons, the BSMRZS:

$$\begin{aligned}
r_{k+1} &= (I - Av_k(A))^2 r_k - 2(I - Av_k(A)) A w_k(A) s_k + A^2 w_k^2(A) z_k ,\\
x_{k+1} &= x_k - (Av_k(A) - 2I) v_k(A) r_k + 2(I - Av_k(A)) w_k(A) s_k - A w_k^2(A) z_k ,\\
z_{k+1} &= q_k^2(A) z_k + 2q_k(A) t_k(A) s_k + t_k^2(A) r_k ,\\
s_{k+1} &= (q_k(A) - Aq_k(A)v_k(A) - At_k(A)w_k(A)) s_k - Aq_k(A) w_k(A) z_k\\
&\quad + t_k(A)(I - Av_k(A)) r_k .
\end{aligned}$$

References

[1] C. Brezinski, *Padé-type Approximation and General Orthogonal Polynomials*, Internat. Ser. Number. Math. 50, Birkhäuser, Basel 1980.

[2] —, *The methods of Vorobyev and Lanczos*, to appear.

[3] —, *Biorthogonality and its Applications to Numerical Analysis*, Marcel Dekker, New York 1992.

[4] C. Brezinski and M. Redivo Zaglia, *A new presentation of orthogonal polynomials with application to their computation*, Numer. Algorithms 1 (1991), 207–221.

[5] —, —, *Treatment of near-breakdown in the CGS algorithm*, to appear.

[6] C. Brezinski, M. Redivo Zaglia and H. Sadok, *A breakdown-free Lanczos type algorithm for solving linear systems*, Numer. Math. 63 (1992), 29–38.

[7] —, —, —, *Avoiding breakdown and near-breakdown in Lanczos type algorithms*, Numer. Algorithms 1 (1991), 261–284.

[8] C. Brezinski and H. Sadok, *Lanczos type methods for solving systems of linear equations*, Appl. Numer. Math., to appear.

[9] —, —, *Avoiding breakdown in the CGS algorithm*, Numer. Algorithms 1 (1991), 199–206.

[10] A. Draux, *Polynômes Orthogonaux Formels — Applications*, Lecture Notes in Math. 974, Springer, Berlin 1983.

[11] R. Fletcher, *Conjugate gradient methods for indefinite systems*, in: Numerical Analysis, G. A. Watson (ed.), Lecture Notes in Math. 506, Springer, Berlin 1976, 73–89.

[12] M. H. Gutknecht, *A completed theory of the unsymmetric Lanczos process and related algorithms. Part I*, SIAM J. Matrix Anal. Appl. 13 (1992), 594–639.

[13] M. R. Hestenes and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, J. Res. Nat. Bur. Standards 49 (1952), 409–439.
[14] C. Lanczos, *Solution of systems of linear equations by minimized iterations*, ibid., 33–53.
[15] P. Sonneveld, *CGS, a fast Lanczos-type solver for nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput. 10 (1989), 36–52.
[16] Yu. V. Vorobyev, *Method of Moments in Applied Mathematics*, Gordon and Breach, New York 1965.
[17] D. M. Young and K. C. Jea, *Generalized conjugate-gradient acceleration of nonsymmetrizable iterative methods*, Linear Algebra Appl. 34 (1980), 159–194.