Chapter 10
Conjugate Gradient Methods and Generalisations

Abstract The conjugate gradient method is the best-known semi-iteration. Consuming only a small computational overhead, it is able to accelerate the underlying iteration. However, its use is restricted to positive definite matrices and positive definite iterations. There are several generalisations to the Hermitian and to the general case. In Section 10.1 we introduce the general concept of the required orthogonality conditions and the possible connection to minimisation principles. The standard conjugate gradient method is discussed in Section 10.2. The method of conjugate residuals introduced in Section 10.3 applies to Hermitian but possibly indefinite matrices. The method of orthogonal directions described in Section 10.4 also applies to general Hermitian matrices. General nonsymmetric problems are treated in Section 10.5. The generalised minimal residual method (GMRES; cf. §10.5.1), the full orthogonalisation method (cf. §10.5.2), and the biconjugate gradient method and its variants (cf. §10.5.3) are discussed.

10.1 Preparatory Considerations

In the following $x^* := A^{-1}b$ denotes the exact solution, while $x \in \mathbb{K}^I$ may be used as a variable. The iterate x^m is associated with the error $e^m = x^m - x^*$ and the residual $r^m = b - Ax^m = -Ae^m$.

10.1.1 Characterisation by Orthogonality

As seen in Conclusion 8.13a, the semi-iterates (8.3) belong to the affine space $x^0 + N\mathcal{K}_m(AN, r^0) = x^0 + \mathcal{K}_m(NA, Nr^0)$. In the following, we replace the Krylov space $\mathcal{K}_m(NA, Nr^0)$ by a general subspace

$$\mathcal{U}_m \subset \mathbb{K}^I \qquad \text{with} \quad \dim(\mathcal{U}_m) = m. \tag{10.1a}$$

W. Hackbusch, *Iterative Solution of Large Sparse Systems of Equations*,
Applied Mathematical Sciences 95, DOI 10.1007/978-3-319-28483-5_10

The reason for using $\mathcal{U}_m$ is that the following arguments are independent of the special nature of the Krylov space. We are looking for candidates

$$x^m \in x^0 + \mathcal{U}_m. \tag{10.1b}$$

The second space

$$\mathcal{V}_m \subset \mathbb{K}^I \qquad \text{with} \quad \dim(\mathcal{V}_m) = m$$

may coincide with $\mathcal{U}_m$.

For the practical implementation, we use bases

$$\mathcal{U}_m = \operatorname{span}\{u^1, \ldots, u^m\}, \qquad \mathcal{V}_m = \operatorname{span}\{v^1, \ldots, v^m\}. \tag{10.1c}$$

Remark 10.1. (a) The spaces are *nested* if

$$\mathcal{U}_1 \subset \ \ldots \ \subset \mathcal{U}_m \subset \mathcal{U}_{m+1} \subset \ \ldots \ . \tag{10.2}$$

In this case, it is advantageous if the basis vectors $u^1, \ldots, u^m$ of $\mathcal{U}_m$ coincide with the first m basis vectors of $\mathcal{U}_{m+1}$.

(b) For stability reasons, orthonormal bases are a good choice. In the case of $\mathcal{U}_m \subset \mathcal{U}_{m+1}$, $u^{m+1} \in \mathcal{U}_{m+1}$ is the normalised vector with $u^{m+1} \perp \mathcal{U}_m$.

The following methods are directly or indirectly characterised by the condition that the m-th iterate x^m fulfils an orthogonality condition:

$$x^m \text{ satisfies (10.1b) and } r^m := b - Ax^m \perp \mathcal{V}_m. \tag{10.3}$$

The questions that arise are:

1. Is (10.3) uniquely solvable?
2. Can we derive estimates for the error e^m in some norm?
3. How costly is the solution of (10.3)?

The first question will be answered in §10.1.2 and the second in §10.1.5. The cost is discussed later for the concrete choice of spaces.

Remark 10.2. Condition (10.3) is equivalent to

$$\left\langle r^m, v^i \right\rangle = 0 \qquad \text{for all } \ 1 \le i \le m \tag{10.4}$$

(cf. (10.1c)). A generalisation of (10.3) could be $\left\langle r^m, v^i \right\rangle_X = 0$ using another scalar product $\langle u, v \rangle_X := \langle Xu, v \rangle$ for some $X > 0$ (cf. Remark C.10). However, this approach is identical to (10.3) with $\mathcal{V}_m$ replaced with $X\mathcal{V}_m$.

10.1.2 Solvability

As required in (2.2), we always assume that the underlying matrix A of the system $Ax = b$ is regular.

The basis (10.1c) of $\mathcal{U}_m$ allows us to make the ansatz $x^m = x^0 + \sum_{j=1}^m a_j u^j$. This implies that $r^m = r^0 - \sum_{j=1}^m a_j A u^j$. The conditions in (10.4) yield the system

$$Za = z \qquad \text{with} \quad Z_{ij} := \left\langle Au^j, v^i \right\rangle, \quad z_i := \left\langle r^0, v^i \right\rangle. \tag{10.5}$$

In general, there is no guarantee that Z is regular. If $m \le \#I/2$, the orthogonal situation $A\mathcal{U}_m \perp \mathcal{V}_m$ is possible and yields the extreme case of $Z = 0$.

Remark 10.3. The regularity of Z is equivalent to either of the conditions

$$(A\mathcal{U}_m)^\perp \cap \mathcal{V}_m = \mathcal{U}_m^\perp \cap A^{\mathsf{H}}\mathcal{V}_m = A\mathcal{U}_m \cap \mathcal{V}_m^\perp = \mathcal{U}_m \cap A^{\mathsf{H}}\mathcal{V}_m^\perp = \{0\}.$$

It remains to formulate sufficient conditions ensuring the regularity of Z.

Criterion 10.4. *(a) Let $\mathcal{U}_m = \mathcal{V}_m$ and assume $A + A^{\mathsf{H}} > 0$. Then Z is regular.*
(b) If $\mathbb{K} = \mathbb{C}$, the previous condition may be replaced with $\mathrm{i}\,(A^{\mathsf{H}} - A) > 0$.
(c) $\mathcal{U}_m = \mathcal{V}_m$ and $A > 0$ are sufficient.
(d) For $N > 0$ and a general regular matrix A, the choice of $\mathcal{V}_m = NA\mathcal{U}_m$ ensures regularity of Z.

Proof. (a) If Z is singular, there is some $0 \neq a \in \mathbb{K}^m$ with $Za = 0$, i.e., $Au \perp \mathcal{V}_m$ for $u := \sum_{j=1}^m a_j u^j \neq 0$. This is a contradiction to $0 < \left\langle (A + A^{\mathsf{H}})u, u \right\rangle = 2\,\mathfrak{Re}\langle Au, u\rangle$, since $u \in \mathcal{V}_m$.

(b) $\frac{1}{\mathrm{i}}(A - A^{\mathsf{H}}) > 0$ implies that $0 < 2\,\mathfrak{Im}\langle Au, u\rangle$. Part (c) is trivial.

(d) Without loss of generality, the basis of $\mathcal{V}_m$ can be defined by $v^i = NAu^i$. Then $Z_{ij} = \left\langle Au^j, v^i \right\rangle = \left\langle A^{\mathsf{H}}NAu^j, v^i \right\rangle$ corresponds to case (c) with A replaced by $A^{\mathsf{H}}NA > 0$. □

10.1.3 Galerkin and Petrov–Galerkin Methods

Appendix E describes the discretisation of boundary value problems by the Galerkin method. This method can also be applied to finite-dimensional problems. The system $Ax = b$ $(x, b \in \mathbb{K}^I)$ can be rewritten as the variation problem

$$\langle Ax, v\rangle = \langle b, v\rangle \qquad \text{for all } v \in \mathbb{K}^I.$$

Using the initial value x^0, we write $x = x^0 + u$ so that

$$\langle Au, v\rangle = \left\langle r^0, v\right\rangle \qquad \text{for all } v \in \mathbb{K}^I \text{ with } r^0 = b - Ax^0.$$

The Galerkin method replaces this problem by a system of lower dimension m. Let $\mathcal{U}_m$ be the subspace in (10.1a). Then the Galerkin solution $x^m \in x^0 + \mathcal{U}_m$ is defined by $x^m = x^0 + u$ with

$$u \in \mathcal{U}_m \text{ satisfying} \qquad \langle Au, v\rangle = \langle r^0, v\rangle \quad \text{for all } v \in \mathcal{U}_m.$$

Obviously this problem is equivalent to $r^m \perp \mathcal{U}_m$ and therefore to the condition (10.3) with $\mathcal{V}_m = \mathcal{U}_m$.

The coercivity formulated in (E.3) requires $A + A^{\mathsf{H}} \geq \frac{1}{C} I$ for some $C > 0$. In the finite-dimensional case, this is equivalent to $A + A^{\mathsf{H}} > 0$ as in Criterion 10.4a.

The more general Petrov–Galerkin method in Definition E.7 yields the problem

$$\text{find } x^m \in x^0 + u,\ u \in \mathcal{U}_m \text{ with} \qquad \langle Au, v\rangle = \langle r^0, v\rangle \quad \text{for all } v \in \mathcal{V}_m,$$

where now $\mathcal{V}_m$ may be different from $\mathcal{U}_m$. This yields the general condition (10.3).

10.1.4 Minimisation

The orthogonality condition (10.3) may be a consequence of another formulation. If $A > 0$, the Galerkin formulation is the first variation of the minimisation problem (9.2): $F(x) = \frac{1}{2}\langle Ax, x\rangle - \langle b, x\rangle = \min$.

The most general quadratic form whose minimum is the solution of $Ax = b$, is described in Lemma 9.3: $F(x) = \frac{1}{2}\|H^{1/2} r\|_2^2 + c$ with $r = b - Ax$ and $H > 0$. Its first variation leads us to the case (d) in Criterion 10.4 with $N = H$.

10.1.5 Error Statements

Even if the auxiliary system in (10.5) is solvable, there is no guarantee that the quality of x^m improves with increasing m. Nevertheless, if the method makes sense for all $m \leq \#I$, we reach the exact solution, provided that the arithmetic is exact.

Remark 10.5. Let $n := \#I$. (a) The iterate x^n is the exact solution: $x^n = A^{-1}b$. (b) If $r^m \in \mathcal{V}_m$ for some $m \leq n$, then x^m is the exact solution.

Proof. By definition, $r^m \perp \mathcal{V}_m$ holds. Combining this statement with $r^m \in \mathcal{V}_m$ yields $r^m = 0$, i.e., $x^m = A^{-1}b$. This proves part (b). Since $\mathcal{V}_n = \mathbb{K}^I$ because of $\dim(\mathcal{V}_n) = n$, part (b) applies. □

Error estimates can be based on an underlying minimisation problem, provided that condition (10.3) is the result of an optimisation problem. The formulation of the optimisation problem also defines the norm for measuring the error.

10.1.5.1 Energy Norm

Assume a positive definite matrix $A > 0$ so that the energy norm $\|\cdot\|_A$ can be defined (cf. (C.5a)). The Galerkin formulation in a subspace $\mathcal{U}_m = \mathcal{V}_m$ determines the minimiser of $F(x) = \frac{1}{2}\langle Ax, x\rangle - \langle b, x\rangle$ in $x^0 + \mathcal{U}_m$, i.e.,

$$\|x^m - x^*\|_A = \min\left\{\|x - x^*\|_A : x \in x^0 + \mathcal{U}_m\right\}. \tag{10.6}$$

If the spaces are nested (cf. (10.2)), the norm $\|x^m - x^*\|_A$ decreases weakly with increasing m. This statement also holds for minimisation later on.

In the case of $\Phi(\cdot,\cdot,A) \in \mathcal{L}_{>0}$, i.e., $NA > 0$, the minimisation in (10.6) uses the norm $\|\cdot\|_{NA}$ instead of $\|\cdot\|_A$.

The classical CG method in §10.2 will lead us to (10.6) with $\mathcal{U}_m = \mathcal{K}_m(A, r^0)$.

10.1.5.2 Residual Norm

The norm $\|\|x^m - x^*\|\|_A$ coincides with $\|A(x^m - x^*)\|_2 = \|r^m\|_2$. The minimisation of the residual

$$\|r^m\|_2 = \min\left\{\|A(x - x^*)\|_2 : x \in x^0 + \mathcal{U}_m\right\} \tag{10.7}$$

implies the orthogonality

$$r^m \perp A\mathcal{U}_m =: \mathcal{V}_m. \tag{10.8}$$

The latter statement can also be written as $A^{\mathsf{H}} r^m \perp \mathcal{U}_m$. For general matrices, the use of Krylov subspaces leads us to the GMRES method in §10.5.1.

The minimisation of $F(x) = \frac{1}{2}\|N^{1/2}r\|_2^2 + c$ for some $N > 0$ (cf. §10.1.4) generalises (10.7) to

$$\left\|N^{1/2}r^m\right\|_2 = \min\left\{\left\|N^{1/2}A(x - x^*)\right\|_2 : x \in x^0 + \mathcal{U}_m\right\}.$$

In the case of Hermitian matrices A, the realisation with Krylov spaces is given in §10.3.

One must be aware of the fact that a small residual $\|r^m\|_2$ does not necessarily imply that the error $\|x^m - x^*\|_2$ is small (cf. Remark 2.35).

10.1.5.3 Euclidean Norm

The first idea may be to approximate the solution x of $Ax = b$ by the best approximation x^* in $x^0 + \mathcal{U}_m$:

$$\|x^m - x^*\|_2 = \min\left\{\|x - x^*\|_2 : x \in x^0 + \mathcal{U}_m\right\}.$$

The first variation yields the orthogonality condition $e^m \perp \mathcal{U}_m$. In terms of the condition (10.3), this can be written as

$$r^m \perp A^{-\mathsf{H}} \mathcal{U}_m \,. \tag{10.9}$$

However, in general, this problem is not feasible. The cost for computing x^m is at least as high as solving the system $Ax = b$. For instance, $x^1 = x^0 + \alpha u$ (u normalised vector with $\mathcal{U}_1 = \operatorname{span}\{u\}$) is the minimiser if $\alpha = -\langle e^0, u\rangle$. However, $e^0 = x^0 - x^*$ is not available unless the exact solution x^* is known. Therefore, evaluating the scalar product $\langle e^0, u\rangle$ causes a problem.

Nevertheless, the problem becomes solvable if the subspace $\mathcal{U}_m$ can be written as $\mathcal{U}_m = A^{\mathsf{H}}\mathcal{V}_m$ and a basis $\{v^1, \ldots, v^m\}$ of $\mathcal{V}_m$ is known. Then the basis of $\mathcal{U}_m$ can be chosen as $\{u^1, \ldots, u^m\}$ with $u^j := A^{\mathsf{H}} v^j$. In this case, condition (10.9) becomes $r^m \perp \mathcal{V}_m$.

For $A = A^{\mathsf{H}}$ and Krylov spaces $\mathcal{V}_m$, this approach is realised by the method of orthogonal directions in §10.4.

10.2 Conjugate Gradient Method

Concerning books on Krylov methods we refer, e.g., to Greenbaum [167, Part I], Liesen–Strakos [265], Meurant [283], Saad [328], Stoer [355], and van der Vorst [373, §§5–12]. The history is described by Golub–O'Leary [155].

10.2.1 First Formulation

In the following, the gradient method and the conjugate directions in §§9.2–9.3 will be combined. In order not to lose optimality with respect to the previous search directions, we only permit conjugate directions. The residuals (negative gradients) are used to determine the search direction p^m in (9.24). As for the gradient method we assume

$$A > 0 \quad \text{and} \quad F(x) = \frac{1}{2}\langle Ax, x\rangle - \langle b, x\rangle \,.$$

After constructing (linearly independent) $p^0, p^1, \ldots, p^{m-1}$, we can orthogonalise r^m with respect to the energy scalar product $\langle\cdot,\cdot\rangle_A$ (cf. Remark A.26a):

$$p^m := r^m - \sum_{\ell=0}^{m-1} \frac{\langle Ar^m, p^\ell\rangle}{\langle Ap^\ell, p^\ell\rangle}\, p^\ell, \tag{10.10a}$$

$$p^0 := r^0. \tag{10.10b}$$

Note that for $m = 0$ the empty sum in (10.10a) implies (10.10b).

Remark 10.6. (a) p^m in (10.10a) is conjugate to all p^ℓ with $0 \le \ell \le m-1$.
(b) The directions p^ℓ span the Krylov subspace

$$\mathcal{K}_m(A, r^0) = \operatorname{span}\{p^0, \dots, p^{m-1}\} = \operatorname{span}\{r^0, \dots, r^{m-1}\}. \tag{10.11a}$$

(c) Having constructed x^m and its residual r^m by the method of conjugate directions, the vectors r^m and p^m can only vanish simultaneously. This means that either $x^m = x^*$ is the exact solution or $p^m \neq 0$ holds.
(d) The residual is orthogonal to the preceding subspaces:

$$r^m \perp \mathcal{K}_\ell(A, r^0) \qquad \text{for all } \ell \le m. \tag{10.11b}$$

Proof. (a) By construction (10.10a), $\langle Ap^m, p^j \rangle = 0$ holds for $j < m$.

(b) Equation (10.11a) holds for $m = 1$. Let (10.11a) be valid for m. Definition (10.10a) implies the identity $\operatorname{span}\{\mathcal{K}_m(A, r^0), p^m\} = \operatorname{span}\{\mathcal{K}_m(A, r^0), r^m\}$ because of Exercise 8.8a and yields assertion (10.11a) for $m+1$.

(d) Repeat (9.25b) stated in Theorem 9.27.

(c) By (10.10a), $p^m = 0$ follows from $r^m = 0$. Assume the case of $p^m = 0$. (10.10a) shows that $r^m \in \mathcal{K}_m(A, r^0)$. On the other hand, $r^m \perp \mathcal{K}_m(A, r^0)$ holds (cf. (10.11b)). Both statements together imply that $r^m = 0$. □

A first provisional representation of the conjugate gradient method reads as follows:

start: x^0 arbitrary; $r^0 := b - Ax^0$;	(10.12a)
Loop over $m = 0, 1, \dots, n-1$: $(n := \#I)$	
stop if $r^m = 0$, otherwise compute p^m from r^m according to (10.10a,b)	(10.12b)
$x^{m+1} := x^m + \lambda_{\text{opt}}(r^m, p^m, A)\, p^m$ with λ_{opt} in (9.24c);	(10.12c)
$r^{m+1} := r^m - \lambda_{\text{opt}}(r^m, p^m, A)\, Ap^m$;	(10.12d)

The properties of this method are summarised below.

Theorem 10.7. *(a) Let m_0 be the value when the loop (10.12b–d) terminates with $r^{m_0} = 0$ and $x^{m_0} = x^*$. Assuming exact arithmetic, $m_0 = \deg_A(e^0) = \deg_A(r^0)$ holds. Since $m_0 \le n := \#I$, the loop terminates latest after n steps.*
(b) The iterates x^m $(0 \le m \le m_0)$ can be characterised by each of the following minimisation problems:

$$F(x^m) = \min\left\{ F\Big(x^0 + \sum_{\ell=0}^{m-1} \lambda_\ell\, p^\ell\Big) : \lambda_0, \dots, \lambda_{m-1} \in \mathbb{K} \right\}, \tag{10.13a}$$

$$F(x^m) = \min\Big\{F\Big(x^0 + \sum_{\ell=0}^{m-1} \mu_\ell\, r^\ell\Big) : \mu_0, \dots, \mu_{m-1} \in \mathbb{K}\Big\}, \tag{10.13b}$$

$$F(x^m) = \min\Big\{F\Big(x^0 + p_{m-1}(A)\, r^0\Big) : p_{m-1} \in \mathcal{P}_{m-1}\Big\}. \tag{10.13c}$$

(c) The minima (10.13a–c) can also be expressed by the energy norm $\|\cdot\|_A$:

$$\begin{aligned}\|e^m\|_A &= \min_{\lambda_0,\dots,\lambda_{m-1}\in\mathbb{K}} \Big\|e^0 + \sum_{\ell=0}^{m-1} \lambda_\ell\, p^\ell\Big\|_A = \min_{\mu_0,\dots,\mu_{m-1}\in\mathbb{K}} \Big\|e^0 + \sum_{\ell=0}^{m-1} \mu_\ell\, r^\ell\Big\|_A \\ &= \min_{p_{m-1}\in\mathcal{P}_{m-1}} \|e^0 + p_{m-1}(A)\, r^0\|_A = \min_{\xi\in\mathcal{K}_m(A,r^0)} \|e^0 + \xi\|_A. \end{aligned} \tag{10.13d}$$

Proof. (b) Because of (10.11a), all minimisation problems (10.13a–c) are of the form

$$F(x^m) = \min\{F(x^0 + \xi) : \xi \in \mathcal{K}_m(A, r^0)\}.$$

This statement coincides with (10.11b) in Remark 10.6d.

(c) The equivalence of the statements in the parts (b) and (c) follows from (9.3): $F(x) = \|x - A^{-1}b\|_A^2 + \text{const}$.

(a) For $m^* = \deg_A(e^0)$, there is a polynomial $p = p_{m^*}$ of degree m^* with $p(A)e^0 = 0$. The scaling can be chosen so that $p(0) = 1$ (cf. Lemma 8.12). Define $q \in \mathcal{P}_{m^*-1}$ by $p(\xi) = 1 - q(\xi)\xi$. Since $0 = p(A)e^0 = e^0 - q(A)Ae^0 = e^0 + q(A)r^0$, the minimum in (10.13d) yields $e^{m^*} = 0$. This proves that the first $m = m_0$ with $e^{m_0} = 0$ satisfies $m_0 \le m^*$. On the other hand, $e^{m_0} = 0$ and (10.13d) prove that there is some polynomial $p(\xi) = 1 - q(\xi)\xi$ of degree m_0 with $p(A)e^0 = 0$. Hence $m_0 \ge m^* = \deg_A(e^0)$. □

The proposed algorithm (10.12a–d) can significantly be simplified in step (10.12b). Computing most of the scalar products $\langle Ar^m, p^\ell\rangle$ in (10.10a) can be avoided.

Lemma 10.8. *$\langle Ar^m, p^\ell\rangle = 0$ holds for all $0 \le \ell \le m-2$, $m \le m_0$.*

Proof. We have $\langle Ar^m, p^\ell\rangle = \langle r^m, Ap^\ell\rangle$. Equation (10.11a) and inclusion (8.9) show that $Ap^\ell \in A\mathcal{K}_{\ell+1}(A, r^0) \subset \mathcal{K}_{\ell+2}(A, r^0) \subset \mathcal{K}_m(A, r^0)$. Therefore, the assertion follows from (10.11b): $r^m \perp \mathcal{K}_m(A, r^0)$. □

Only the term for $\ell = m-1$ does remain in the sum (10.10a):

$$p^m := r^m - \frac{\langle Ar^m, p^{m-1}\rangle}{\langle Ap^{m-1}, p^{m-1}\rangle}\, p^{m-1} = r^m - \frac{\langle r^m, Ap^{m-1}\rangle}{\langle Ap^{m-1}, p^{m-1}\rangle}\, p^{m-1}. \tag{10.14}$$

The second representation in (10.14) has the advantage that only the product Ap^{m-1} is needed which already appears in the denominator, in λ_{opt}, and in (10.12d).

10.2.2 CG Method (Applied to Richardson's Iteration)

Using (10.14), we present the CG method (10.12a–d) in the following form ('CG' abbreviates 'conjugate gradient').

$\Upsilon_{\mathrm{CG}}[\Phi_1^{\mathrm{Rich}}]$	CG method (applied to Richardson's iteration)	(10.15)
start:	x^0 arbitrary; $r^0 := b - Ax^0$; $p^0 := r^0$;	(10.15a)
Loop over	$m = 0, 1, \ldots, n-1$: stop if $r^m = 0$, otherwise:	
	$x^{m+1} := x^m + \lambda_{\mathrm{opt}}\, p^m$ with	(10.15b)
	$\lambda_{\mathrm{opt}} := \lambda_{\mathrm{opt}}(r^m, p^m, A) = \langle r^m, p^m\rangle \,/\, \langle Ap^m, p^m\rangle$;	(10.15c)
	$r^{m+1} := r^m - \lambda_{\mathrm{opt}}\, Ap^m$;	(10.15d)
	$p^{m+1} := r^{m+1} - \frac{\langle r^{m+1}, Ap^m\rangle}{\langle Ap^m, p^m\rangle}\, p^m$;	(10.15e)

Exercise 10.9. The following alternatives are equivalent to (10.15c,e):

$$\lambda_{\mathrm{opt}}(r^m, p^m, A) = \|r^m\|_2^2 \,/\, \langle Ap^m, p^m\rangle\,, \tag{10.15c$'$}$$

$$p^{m+1} = r^{m+1} + \frac{\|r^{m+1}\|_2^2}{\|r^m\|_2^2}\, p^m. \tag{10.15e$'$}$$

Remark 10.10. One CG step $x^m \mapsto x^{m+1}$ requires one multiplication Ap^m and, in addition, only simple vector operations and scalar products. On the other hand, the storage requirement is higher. Besides x^m, also r^m and p^m are needed.

The CG method was first presented in 1952 by Stiefel [353] in a paper still worth reading. Independently, the method was described in the same year by Hestenes (cf. Hestenes [218], Hestenes–Stiefel [219]).

The CG method can be interpreted in two completely different ways:

- as a direct method,
- as an iterative method.

Formally, the CG algorithm is a direct method because it produces the exact solution x^* after finitely many operations (see m_0 in Theorem 10.7a). For the practical performance, this is not true. Since the later and smaller residuals r^m arise from linear combinations of larger quantities, cancellation leads to an error amplification, so that the vectors $\{p^0, \ldots, p^{n-1}\}$ no longer form a conjugate system. After losing the orthogonality, two cases may appear:

- stagnation: the errors e^m fluctuate about the reached level of accuracy,
- instability: the errors start to grow again.

The first case is harmless, provided that the reached error level is sufficient. The second case will happen for many Krylov methods discussed later. This is the reason that there are many equivalent algorithms, i.e., algorithms producing identical results under exact arithmetic, but behaving differently under floating-point perturbation. In the best case, there are 'stabilised' versions which do not become unstable. There is a further, still more severe problem. Division by $\langle Ap^m, p^m\rangle$ already appears in (10.15c). A division by zero leads to a breakdown of the algorithm. A *lucky breakdown* happens if a vanishing divisor only appears if x^m is already the exact solution (so that the algorithm need not to be continued). In the 'unlucky' cases, the 'stabilised' versions should overcome this difficulty (cf. §10.3.3). In any case, one can state that the Krylov methods cannot be used as a practical method for the *direct* solution of large linear systems.

It was Reid [319] how emphasised the use of the CG method as an *iterative* method. Although the limit process $m \to \infty$ does not make sense,[1] the decrease of the error e^m in a range $0 \le m \le m_0$ with $m_0 \ll n = \#I$ is all we need for practical applications, provided that at least e^{m_0} is small enough.

The problem caused by floating-point perturbations suggests a modification towards an infinite CG iteration.[2] Assume that perturbations get out of control after (more than) k steps. Then, after every k steps (i.e., for $m = 0, k, 2k, \ldots$), we start again with the last descent direction $p^k := r^k$, etc. This method is usually called the *restarted CG method*:

$$\begin{array}{ll} \text{start:} & x^0 \text{ arbitrary starting iterate,} \\ \text{iteration } m = 1, 2, \ldots & x^m\text{: as in (10.15b,d),} \\ & r^m, p^m\text{: as in (10.15c–e), if } m \text{ is not a multiple of } k; \\ & r^m := p^m := b - Ax^m, \quad \text{if } m = 0, k, 2k, \ldots \end{array} \tag{10.16}$$

10.2.3 Convergence Analysis

The convergence analysis is based on the following observation corresponding to Remark 9.9 in the case of the gradient method. Property (10.17d) stated below coincides with the characterisation in §10.1.5.1.

Proposition 10.11. *Let $x^0, \ldots, x^{m_0}$ be the sequence of the CG iterates.*
(a) The CG results can be regarded as the results of the semi-iterative Richardson iteration Φ_1^{Rich}. The related polynomials $p_k \in \mathcal{P}_k$ in (8.6c) with $p_k(1) = 1$ yield the error representation

$$e^k = x^k - x^* = p_k(M_1^{\mathrm{Rich}})e^0 = p_k(I - A)e^0 \qquad (M_1^{\mathrm{Rich}} = I - A). \tag{10.17a}$$

[1] The terms 'convergence' and 'asymptotic convergence rate' lose their meaning since no limit can be formed.

[2] This is still a nonlinear iteration. If $x^m = A^{-1}b$, the *lucky breakdown* stops the iteration.

(b) p_k and $q_{k-1}(\xi) := [p_k(1-\xi)-1]/\xi$ are the optimal polynomials solving the respective minimisation problems

$$\|e^k\|_A = \|p_k(M_1^{\mathrm{Rich}})e^0\|_A \le \|\tilde{p}_k(M_1^{\mathrm{Rich}})e^0\|_A \tag{10.17b}$$
$$\text{for all polynomials } \tilde{p}_k \in \mathcal{P}_k \text{ with } \tilde{p}_k(1) = 1,$$
$$\|e^k\|_A = \|e^0 + q_{k-1}(A)r^0\|_A \le \|e^0 + \tilde{q}_{k-1}(A)r^0\|_A \tag{10.17c}$$
$$\text{for all polynomials } \tilde{q}_{k-1} \in \mathcal{P}_{k-1},$$
$$\|e^k\|_A = \min\left\{\|x - x^*\|_A : x \in x^0 + \mathcal{K}_k(A)r^0\right\}. \tag{10.17d}$$

Proof. (a) (10.15b) shows that $x^k = x^0 + \sum_{\nu=0}^{k-1} \beta_\nu p^\nu$ with $\beta_\nu := \lambda_{\mathrm{opt}}(r^\nu, p^\nu, A)$, i.e.,

$$x^k - x^0 = e^k - e^0 \in \mathrm{span}\{p^0, \ldots, p^{k-1}\} = \mathcal{K}_k(A, r^0) \qquad \text{(cf. (10.11a))}.$$

Hence, there is a polynomial $q_{k-1} \in \mathcal{P}_{k-1}$ with $e^k = e^0 - q_{k-1}(A)r^0$. Since $r^0 = -Ae^0$, $e^k = \hat{p}_k(A)e^0$ holds for the polynomial $\hat{p}_k(\xi) := 1 + \xi q_{k-1}(\xi) \in \mathcal{P}_k$. The related polynomial $p_k(\xi) := \hat{p}_k(1-\xi)$ satisfies the consistency condition $p_k(1) = \hat{p}_k(0) = 1$. The identity $p_k(M_1^{\mathrm{Rich}})e^0 = p_k(I-A)e^0 = \hat{p}_k(A)e^0 = e^k$ proves that $p_k \in \mathcal{P}_k$ is the polynomial in (10.17a).

(b) Since the CG results satisfy (10.13d), the polynomial q_{k-1} is the minimiser in (10.17c). Problem (10.17b) is equivalent to (10.17c) and (10.17d). □

Remark 10.12. The CG iterates x^m are not the solutions of the minimisation problem posed in §8.3.1 because there the minimisation is required with respect to the Euclidean norm $\|\cdot\|_2$. However, if $\|\cdot\|_2$ is replaced by $\|\cdot\|_A$, the CG method offers the possibility of solving the modified minimisation problem $\|p_m(M)e^0\|_A = \min$ without knowledge of the initial error e^0 and the spectrum of $M_1^{\mathrm{Rich}} = I - A$ (equivalently, of the spectrum of A).

Remark 10.13. For any polynomial $P_m \in \mathcal{P}_m$ satisfying $P_m(1) = 1$, the errors $e^m = x^m - x^*$ of the CG iterates satisfy the error estimate

$$\|e^m\|_A \le \max\{|P_m(1-\lambda)| : \lambda \in \sigma(A)\}\, \|e^0\|_A. \tag{10.18}$$

Proof. (10.17b) shows that $\|e^m\|_A \le \|P_m(I-A)\|_A \|e^0\|_A$. The matrix norm $\|\cdot\|_A$ has the representation $\|X\|_A = \|A^{1/2} X A^{-1/2}\|_2$ (cf. (C.5d)). $A^{1/2}$ commutes with polynomials in A: $A^{1/2}P_m(I-A)A^{-1/2} = P_m(I-A)$. The assertion (10.18) follows from $\|P_m(I-A)\|_2 = \max\{|P_m(1-\lambda)| : \lambda \in \sigma(A)\}$. □

The following theorem shows that—as in the case of the Chebyshev method—an order improvement can be achieved.

Theorem 10.14. *Let A be positive definite with $\lambda := \lambda_{\min}(A)$, $\Lambda := \lambda_{\max}(A)$ and abbreviate the spectral condition number by $\kappa = \kappa(A) = \Lambda/\lambda$. The errors e^m of the CG iterates x^m satisfy the estimate*

$$\|e^m\|_A \le \frac{2\left(1-\frac{1}{\kappa}\right)^m}{\left(1+\frac{1}{\sqrt{\kappa}}\right)^{2m}+\left(1-\frac{1}{\sqrt{\kappa}}\right)^{2m}}\,\|e^0\|_A = \frac{2c^m}{1+c^{2m}}\|e^0\|_A \qquad (10.19)$$

with $c := \dfrac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1} = \dfrac{\sqrt{\Lambda}-\sqrt{\lambda}}{\sqrt{\Lambda}+\sqrt{\lambda}}$.

Proof. Let P_m be the transformed Chebyshev polynomial (8.27a) belonging to $\sigma_M := [a,b] \supset \sigma(M^{\mathrm{Rich}}) = \sigma(I-A)$ with $a = 1-\Lambda$ and $b = 1-\lambda$. (10.18) and (8.27b) yield $\|e^m\|_A \le \|e^0\|_A / C_m$. (8.28c) proves (10.19). □

The error estimate (10.19) uses an upper bound that may be too pessimistic. It is based on the Chebyshev polynomial P_m which is the optimal choice for minimising $\max\{|P_m(\xi)| : \xi \in \sigma_M = [a,b]\}$, but not necessarily for minimising $\max\{|P_m(\xi)| : \xi \in \sigma(M^{\mathrm{Rich}}) = \sigma(I-A)\} = \max\{|P_m(1-\lambda)| : \lambda \in \sigma(A)\}$. This leads to the following statement.

Remark 10.15. Although the asymptotic convergence rate of the gradient method depends exclusively on the spectral condition number $\kappa(A)$ and therefore the extreme eigenvalues, the convergence of the CG method is influenced by the whole spectrum.

The following simple example will illustrate this fact. Assume that the inclusion $\sigma(M^{\mathrm{Rich}}) \subset [a,b]$ with $a = 1-\Lambda$, $b = 1-\lambda$ can be strengthened to $\sigma(M^{\mathrm{Rich}}) \subset \sigma_M := [a,a'] \cup [b',b]$ with $a \le a' < b' \le b$. Then one may find a polynomial P_m for which $\max\{|P_m(1-\lambda)| : \lambda \in \sigma(A)\}$ is smaller than for the Chebyshev polynomial (cf. §8.3.6). Hence, P_m yields a better estimate than (10.19). Generally speaking, if the eigenvalues of A are not distributed uniformly over $[\lambda,\Lambda]$ (e.g., if they accumulate in smaller subintervals), the CG method converges better than estimated by (10.19).

Exercise 10.16. If the spectrum $\sigma(M^{\mathrm{Rich}}) = \{\lambda,\Lambda\}$ contains only the extreme eigenvalues λ and Λ, the cg method yields $x^{m_0} = x^*$ for $m_0 \le 2$.

Even if the eigenvalue distribution permits no better polynomial than the Chebyshev polynomial, the ratios $\|e^{m+1}\|_A / \|e^m\|_A$ improve with increasing iteration number m and become smaller than $c \approx 1 - 2/\sqrt{\kappa}$ in (10.19). The reason is as follows. In the case of the gradient method (9.11a–c), the error e^m converges to the subspace $V := \mathrm{span}\{v_1, v_2\}$ spanned by the eigenvectors belonging to $\lambda := \lambda_{\min}(A)$ and $\Lambda := \lambda_{\max}(A)$ (see the proof of Corollary 9.11). For the CG case, this behaviour cannot occur. If the CG error e^m lies exactly in the subspace V, $2 = \dim V$ steps of the CG methods would be sufficient to obtain $e^{m+2} = 0$. It can be proved that the CG error moves towards $V^\perp$. Restricting the matrix A to $V^\perp$, we obtain the spectrum $\sigma(A)\backslash\{\lambda,\Lambda\}$ and the condition is Λ_2/λ_2, where λ_2 is the second smallest and Λ_2 the second largest eigenvalue. Hence, after a certain number of steps, the error ratios behave more like $c \approx 1 - 2/\sqrt{\Lambda_2/\lambda_2} < c$. A precise analysis of this superconvergence phenomenon is given by van der Sluis–van der Vorst [370]. See also Strakos [357].

10.2.4 CG Method Applied to Positive Definite Iterations

10.2.4.1 Standard Version

As the gradient method, the method of conjugate gradients can be applied to other positive definite iterations than the Richardson method. This yields the so-called *preconditioned CG method* (but notice that the gradients are preconditioned not the CG method). Assume $\Phi \in \mathcal{L}_{\mathrm{pos}}$. Hence, the standard assumption $A > 0$ implies $N > 0$ for the matrix $N = N[\Phi]$ in

$$x^{m+1} = x^m - N(Ax^m - b) \qquad \text{with } A, N \text{ positive definite.} \tag{10.20a}$$

As in (9.15b), we introduce $\check{A} := N^{\frac{1}{2}} A N^{\frac{1}{2}}$ and $\check{b} := N^{\frac{1}{2}} b$. Algorithm (10.20a) is equivalent to the Richardson iteration (10.20b) for solving $\check{A}\check{x} = \check{b}$:

$$\check{x}^{m+1} = \check{x}^m - (\check{A}\check{x}^m - \check{b}). \tag{10.20b}$$

Applying the CG algorithm (10.15a–e) to $\check{A}\check{x} = \check{b}$, we obtain:

$$\begin{aligned}
&\text{start: } \check{x}^0 := N^{-1/2}x^0; \quad \check{r}^0 := \check{b} - \check{A}\check{x}^0; \quad \check{p}^0 := \check{r}^0; && (10.21a)\\
&\text{for } m = 0, 1, 2, \ldots \text{ (while } \check{r}^m \neq 0)\text{:}\\
&\quad \check{x}^{m+1} := \check{x}^m + \lambda_{\mathrm{opt}}\, \check{p}^m \text{ with} && (10.21b)\\
&\quad \lambda_{\mathrm{opt}} := \lambda_{\mathrm{opt}}(\check{r}^m, \check{p}^m, A) = \langle \check{r}^m, \check{p}^m \rangle \,/\, \langle \check{A}\check{p}^m, \check{p}^m \rangle; && (10.21c)\\
&\quad \check{r}^{m+1} := \check{r}^m - \lambda_{\mathrm{opt}}\, \check{A}\, \check{p}^m \quad (= \check{b} - \check{A}\check{x}^{m+1}); && (10.21d)\\
&\quad \check{p}^{m+1} := \check{r}^{m+1} - \langle \check{r}^{m+1}, \check{A}\check{p}^m \rangle \,/\, \langle \check{A}\check{p}^m, \check{p}^m \rangle\, \check{p}^m; && (10.21e)
\end{aligned}$$

Insert $\check{A} = N^{1/2} A N^{1/2}$ and $\check{b} = N^{1/2} b$, define x^m and p^m by

$$\check{x}^m = N^{-1/2} x^m, \qquad \check{p}^m = N^{-1/2} p^m \tag{10.21f}$$

and use $N^{1/2} r^m = N^{1/2}(b - Ax^m) = \check{b} - \check{A}\check{x}^m = \check{r}^m$. (10.21a–e) becomes

$$\begin{aligned}
&\text{start:} \quad x^0 \text{ arbitrary}; \quad r^0 := b - Ax^0; \quad p^0 := Nr^0; && (10.22a)\\
&\text{iteration: for } m = 0, 1, 2, \ldots \text{ (while } r^m \neq 0)\text{:}\\
&\quad x^{m+1} := x^m + \lambda_{\mathrm{opt}}\, p^m \text{ with} && (10.22b)\\
&\quad \lambda_{\mathrm{opt}} := \lambda_{\mathrm{opt}}(r^m, p^m, A) = \langle r^m, p^m \rangle \,/\, \langle Ap^m, p^m \rangle; && (10.22c)\\
&\quad r^{m+1} := r^m - \lambda_{\mathrm{opt}}\, A\, p^m; && (10.22d)\\
&\quad p^{m+1} := Nr^{m+1} - \frac{\langle Nr^{m+1}, Ap^m \rangle}{\langle Ap^m, p^m \rangle}\, p^m; && (10.22e)
\end{aligned}$$

The expression (10.22c) coincides with the original definition (9.6a) of λ_{opt}. (10.22e) shows that the search directions p^m are produced from the 'preconditioned'

gradient Nr^m by an A-orthogonalisation. Exploiting the equivalent formulations (10.15c′,e′), we end up with

$$\lambda_{\rm opt} := \langle Nr^m, r^m\rangle \,/\, \langle Ap^m, p^m\rangle\,;$$
$$p^{m+1} := Nr^{m+1} + \frac{\langle Nr^{m+1}, r^{m+1}\rangle}{\langle Nr^m, r^m\rangle}\, p^m.$$

If one carries along the variables x^m, p^m, r^m, and $\rho_m := \langle Nr^m, r^m\rangle$ during the iteration, the CG algorithm $\Upsilon_{\rm CG}[\Phi]$ takes the form (10.23a–f):

start: x^0 arbitrary;	(10.23)
$r^0 := b - Ax^0$; $\quad p^0 := Nr^0$; $\quad \rho_0 := \langle p^0, r^0\rangle$;	(10.23a)
iteration: for $m = 0, 1, 2, \ldots$ (while $m < n := \#I$ and $r^m \neq 0$):	
$a^m := Ap^m$; $\quad \lambda_{\rm opt} := \rho_m / \langle a^m, p^m\rangle$;	(10.23b)
$x^{m+1} := x^m + \lambda_{\rm opt}\, p^m$;	(10.23c)
$r^{m+1} := r^m - \lambda_{\rm opt}\, a^m$;	(10.23d)
$q^{m+1} := Nr^{m+1}$; $\quad \rho_{m+1} := \langle q^{m+1}, r^{m+1}\rangle$;	(10.23e)
$p^{m+1} := q^{m+1} - \frac{\rho_{m+1}}{\rho_m}\, p^m$;	(10.23f)

The error estimate for $e^m = x^m - x^*$ follows as in §9.2.4, since the inequality (10.19) for $\check{e}^m = \check{x}^m - \check{x}^* = N^{-1/2}e^m$ can be transferred to e^m: $\|\check{e}^m\|_{\check{A}} = \|e^m\|_A$. Notice that $\kappa = \kappa(\check{A}) = \kappa(N^{1/2}AN^{1/2}) = \kappa(NA) = \Gamma/\gamma$ with Γ and γ in (10.24a).

Theorem 10.17 (error estimate). *Assume $\Phi \in \mathcal{L}_{\rm pos}$ and $A > 0$. The matrix $W = N^{-1}$ of the third normal form is assumed to satisfy*

$$\gamma W \le A \le \Gamma W \qquad (\gamma > 0,\ \textit{cf. (9.18a)}). \tag{10.24a}$$

Then the iterates x^m of the CG method $\Upsilon_{\rm CG}[\Phi]$ in (10.23a–f) are the minimisers of $\min\{\|x - x^*\|_A : x = x^0 + \mathcal{K}_m(NA)Nr^0\}$ and fulfil the energy norm estimate

$$\|e^m\|_A \le \frac{2c^m}{1 + c^{2m}}\|e^0\|_A \quad \textit{with } c = \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} = \frac{\sqrt{\Gamma} - \sqrt{\gamma}}{\sqrt{\Gamma} + \sqrt{\gamma}},\ \kappa = \frac{\Gamma}{\gamma}. \tag{10.24b}$$

Lemma 10.18. *(a) $m_0 = \deg_{\check{A}}(\check{e}^0) = \deg_{NA}(e^0) = \deg_{AN}(r^0) \le n = \#I$ is the first index m_0 with $r^{m_0} = 0$ and $x^{m_0} = x^*$.*

(b) The search directions generated by (10.23a–f) are conjugate with respect to the original matrix A:

$$\langle p^k, p^\ell\rangle_A = 0 \qquad \textit{for } k \neq \ell. \tag{10.25}$$

(c) The statements in (10.11a,b) become

$$r^m \perp \mathcal{K}_m(NA, Nr^0) = \operatorname{span}\{p^0, \ldots, p^{m-1}\}$$
$$= \operatorname{span}\{Nr^0, \ldots, Nr^{m-1}\}.$$

(d) The iterate x^m is the minimiser of the expressions

$$F(x^m) = \min_{\lambda_0,\dots,\lambda_{m-1}\in\mathbb{K}} F\Big(x^0+\sum_{\ell=0}^{m-1}\lambda_\ell\, p^\ell\Big) = \min_{\mu_0,\dots,\mu_{m-1}\in\mathbb{K}} F\Big(x^0+N\sum_{\ell=0}^{m-1}\mu_\ell\, r^\ell\Big)$$
$$= \min_{p_{m-1}\in\mathcal{P}_{m-1}} F\big(x^0+p_{m-1}(NA)Nr^0\big) = \min_{\xi\in\mathcal{K}_m(NA,Nr^0)} F\big(x^0+\xi\big).$$

Proof. Part (a) is identical to Theorem 10.7a. Part (b) follows from (10.21f),

$$\big\langle \check p^k,\check p^\ell\big\rangle_{\check A} = \big\langle \check A\check p^k,\check p^\ell\big\rangle = \Big\langle N^{1/2}AN^{1/2}N^{-1/2}p^k, N^{-1/2}p^\ell\Big\rangle = \big\langle Ap^k,p^\ell\big\rangle$$
$$= \big\langle p^k,p^\ell\big\rangle_A$$

and the $\check A$-orthogonality of the search directions $\check p^k$. Parts (c) and (d) are consequences of (10.13a–c) applied to the $^\vee$-quantities in (10.21f). □

The alternative reformulation $\bar x^{m+1} := \bar x^m - (\bar A\bar x^m - \bar b)$ used in §9.2.4.2 will be discussed in §10.3.

10.2.4.2 Directly Positive Definite Case

Assume $\Phi\in\mathcal{L}_{>0}$, i.e., the iteration $\Phi(\cdot,\cdot,A)$ is directly positive definite: $N[A]A>0$ (cf. Definition 5.14). Now we substitute A, x, b by $\hat A := NA$, $\hat x = x$, $\hat b = Nb$ in the CG algorithm (10.21a–e). Reformulation the algorithm in terms of the quantities A, x, b yields

start: x^0 arbitrary; $r^0 := b - Ax^0$; $p^0 := Nr^0$; $m := 0$;
iteration: $x^{m+1} := x^m + \lambda_{\rm opt}\, p^m$ with
$\lambda_{\rm opt} := \lambda_{\rm opt}(Nr^m, p^m, NA) = \langle Nr^m, p^m\rangle \,/\, \langle NAp^m, p^m\rangle$;
$r^{m+1} := r^m - \lambda_{\rm opt}\, A\, p^m$;
$p^{m+1} := Nr^{m+1} - \frac{\langle Nr^{m+1}, NAp^m\rangle}{\langle NAp^m, p^m\rangle}\, p^m$;

Proposition 10.19. *(a) The final iteration number is $m_0 = \deg_{NA}(e^0)$.*
(b) The directions p^m are NA-orthogonal.
(c) The transformed residuals are orthogonal:

$$Nr^m \perp \mathcal{K}_m(NA, Nr^0) = \operatorname{span}\{p^0,\dots,p^{m-1}\} = \operatorname{span}\{Nr^0,\dots,Nr^{m-1}\}.$$

(d) $\|e^m\|_{NA} \le \dfrac{2c^m}{1+c^{2m}}\|e^0\|_{NA}$ holds with $c = \dfrac{\sqrt{\Gamma}-\sqrt{\gamma}}{\sqrt{\Gamma}+\sqrt{\gamma}}$, where γ and Γ are the minimal and maximal eigenvalues of NA.

10.2.5 Numerical Examples

We choose the Poisson model problem for $h = 1/32$. Applying the CG method to the Richardson iteration (i.e., algorithm $\Upsilon_{\mathrm{CG}}[\Phi_1^{\mathrm{Rich}}]$ in (10.15a–e)) yields the results given in Table 10.1. Due to inequality (10.19), the convergence factors $\|e^m\|_A/\|e^{m-1}\|_A$ measured with respect to the energy norm $\|\cdot\|_A$ should become smaller than $c = (\sqrt{\Lambda} - \sqrt{\lambda})/(\sqrt{\Lambda} + \sqrt{\lambda})$. Inserting the eigenvalues λ and Λ in (3.1b,c) for $h = 1/32$, we obtain $c = 0.9063471$. In fact, the convergence factor decreases from 0.9 to 0.66 when $m = 30$ increases to $m = 90$. This 'superlinear' convergence behaviour illustrates the improvement of the effective condition during the iteration as discussed in the last paragraph of §10.2.3.

m	value in the middle	$\frac{\|e^m\|_A}{\|e^{m-1}\|_A}$
1	-0.00186560978	0.670874
2	-0.00460087980	0.791286
3	-0.00739241614	0.860663
4	-0.01111605755	0.865691
10	-0.04408187826	0.917138
20	-0.11796241337	0.939358
30	0.40673579950	0.918423
40	0.49137792828	0.843496
50	0.50013929834	0.832459
60	0.50010381735	0.738779
70	0.50001053720	0.761377
80	0.50000013936	0.708295
90	0.50000000342	0.661969
100	0.50000000001	

Table 10.1 Results of $\Upsilon_{\mathrm{CG}}[\Phi_1^{\mathrm{Rich}}]$ applied to the Poisson model problem with $h = 1/32$.

Table 10.2 reports the CG results for $h = \frac{1}{32}$ with the SSOR and ILU iteration as basic iterations. The optimal SSOR parameter is the same as for Table 9.2. The ILU iteration is the modified five-point version ILU_5 with $\omega = -1$ and enlargement of the diagonal by 5 (cf. §7.3.10). The condition of the SSOR method determined in §9.2.5 is $\kappa \approx 7.66$. This yields the value $c \approx 0.47$ for c in (10.24b). In the SSOR case, the averaged convergence factors $(\|e^m\|_A/\|e^0\|_A)^{\frac{1}{m}}$

	5-point ILU with $\omega = -1$		SSOR with $\omega = 1.8212691200$	
m	$u_{16,16}$	$\frac{\|e^m\|_A}{\|e^{m-1}\|_A}$	$u_{16,16}$	$\frac{\|e^m\|_A}{\|e^{m-1}\|_A}$
1	0.2262513522	0.156365	0.0285107511	0.457624
2	0.5320480495	0.446360	0.1146321025	0.307093
3	0.4582969109	0.465620	0.2093879771	0.599140
4	0.4818928890	0.459572	0.3500438579	0.530214
5	0.4827955876	0.490598	0.4301535841	0.491911
10	0.4999129317	0.380570	0.4992951874	0.464830
11	0.5000044282	0.358332	0.4998541213	0.465082
12	0.4999850353	0.429905	0.4999456258	0.394760
20	0.5000000033	0.342381	0.5000000087	0.320139
21	0.5000000026	0.388711	0.5000000020	0.487606
22	0.5000000008	0.405064	0.5000000055	0.405755
23	0.5000000002	0.313452	0.5000000041	0.408013
24	0.5000000000	0.355741	0.5000000000	0.332715
25	0.5000000000	0.451311	0.5000000005	0.432772
26	0.5000000000	0.557156	0.5000000001	0.334264
27	0.5000000000	0.517255	0.5000000001	0.366209
28	0.5000000000	0.802069	0.5000000000	0.365471
29	0.5000000000	0.969482	0.5000000000	0.487797
30	0.5000000000	1.00102	0.5000000000	0.776690

Table 10.2 The CG method (10.21a–e) applied to the ILU and SSOR iterations.

are around 0.47 until $m = 11$. Afterwards they decrease to 0.42 for $m \approx 30$. The values $u_{16,16}$ ('value in the middle') given in Table 10.2 show that for $m \geq 27$ the rounding errors acquire the upper hand. Nevertheless, the CG algorithm is stable.

The superlinear convergence behaviour mentioned in connection with Table 10.1 should not be overrated. Its advantage can be exploited only if m becomes sufficiently large. In the case of Table 10.1, 'sufficiently large' means $m \geq 30$; in the SSOR case of Table 10.2, it is $m \geq 17$. Inspecting the values in these tables illustrates the following dilemma:

- Either the iteration is fast (as in Table 10.2). Then one would like to stop the iteration before reaching the critical value of m indicating the appearance of superconvergence.
- Or the iteration is slow (as in Table 10.1). Then one would prefer to replace the iteration Φ with a better one.

10.2.6 Amount of Work of the CG Method

One iteration step (10.23b–f) requires one evaluation of $p \mapsto Ap$ and $r \mapsto Nr$, three vector additions, three multiplications of a vector by a scalar number, and two scalar products. This adds up to

$$\text{CG-Work}(\Phi) = C(A) + C(N) + 8n$$

arithmetic operations for $\Upsilon_{\text{CG}}[\Phi]$, where

$$C(A)\text{: work for } p \mapsto Ap, \qquad C(N)\text{: work for } r \mapsto Nr.$$

Performing the Φ-iteration step in the form $\Phi(x, b) = x - N(Ax - b)$, we need $C(A) + C(N) + 2n$ operations, so that

$$\text{CG-Work}(\Phi) = \text{Work}(\Phi) + 6n.$$

Hence, as in the semi-iterative case (cf. §8.3.9), the cost factor is equal to

$$C_{\Phi,\text{cg}} = C_\Phi + 6/C_A.$$

According to the analysis of convergence behaviour discussed above, we choose $c = (\sqrt{\Lambda} - \sqrt{\lambda})/(\sqrt{\Lambda} + \sqrt{\lambda})$ in (10.24b) as the asymptotic rate on which we base the effective amount of work:

$$\text{Eff}_{\text{cg}}(\Phi) = -\left(C_\Phi + \frac{6}{C_A}\right) \log\left(\frac{\sqrt{\Lambda} - \sqrt{\lambda}}{\sqrt{\Lambda} + \sqrt{\lambda}}\right).$$

Remark 10.20. Even if these numbers coincide exactly with those obtained in §8.3.9 for the Chebyshev method, one has to emphasise one important advantage of the CG method: The eigenvalue bounds γ and Γ may be unknown to the user. Vice versa, the efficacy of the Chebyshev method deteriorates if too pessimistic bounds $\gamma,\ \Gamma$ are inserted.

10.2.7 Suitability for Secondary Iterations

Section 5.5 describes composed iterations arising from $x \mapsto x - B^{-1}(Ax - b)$ by replacing the exact solution of $B\delta = d$ with the approximation by a secondary iteration. Now we can start with $\delta^0 = 0$ and perform m steps of the CG algorithm. Positive and negative comments concerning this approach are given in the next lemma.

Lemma 10.21. *Let A and B be positive definite matrices.*

$$\Phi_A(x, b) = x - B^{-1}(Ax - b)$$

is the primary iteration. For solving $B\delta = d$, the CG method $\Upsilon_{\mathrm{CG}}[\Phi_B]$ based on the iteration

$$\Phi_B(\delta, d) = \delta - C^{-1}(B\delta - d)$$

with a starting iterate $\delta^0 = 0$ is inserted as a secondary solver. The number k of CG steps is chosen such that $2c^k \le \varepsilon$ holds with

$$c = (\sqrt{\Lambda} - \sqrt{\lambda})/(\sqrt{\Lambda} + \sqrt{\lambda}), \quad 0 < \delta C \le B \le \Delta C.$$

The composed iteration Φ_k is no longer linear, but it still can be written in the form

$$\Phi_k(x, b) = M_k(Ax - b)x + N_k(Ax - b)b \tag{10.26a}$$

with matrices $M_k(d)$, $N_k(d)$ depending on the defect $d = Ax - b$. They have the contraction number (10.26b) with respect to the energy norm:

$$\|M_k(Ax - b)\|_A \le \|M_A\|_A + \varepsilon \|A^{\frac{1}{2}} B^{-1} A^{\frac{1}{2}}\|_2 \quad (M_A = I - B^{-1}A). \tag{10.26b}$$

Before proving the lemma, we comment on (10.26b). If, as in §5.5.1, B is a preconditioner with $\kappa(B^{-1}A) = \|A^{1/2}B^{-1}A^{1/2}\|_2 = \mathcal{O}(1)$, the right-hand side in (10.26b) is bounded by $\|M_A\|_A + C\varepsilon$. For example, one should choose ε such that

$$\|M_A\|_A + C\varepsilon \le \frac{1}{2}(1 + \|M_A\|_A) < 1.$$

Proof of Lemma 10.21. The right-hand side d in $B\delta = d$ is the defect $d = Ax^m - b$ (cf. (5.18a)). Because of $\delta^0 = 0$, the error estimate (10.24b) yields the B-energy norm $\|\delta^k - \delta\|_B \le \varepsilon \|\delta^0 - \delta\|_B = \varepsilon \|\delta\|_B,\ \delta := B^{-1}d$. From

$$\|\delta\|_B = \|B^{1/2}\delta\|_2 = \|B^{-1/2}A(x^m - x^*)\|_2 \le \|B^{-1/2}AB^{-1/2}\|_2 \|x^m - x^*\|_B,$$

we deduce

$$\begin{aligned}\|x^{m+1} - x^*\|_B &= \|x^m - \delta^k - x^*\|_B \le \|x^m - \delta - x^*\|_B + \|\delta^k - \delta\|_B \\ &= \|\Phi_A(x^m, b) - x^*\|_B + \|\delta^k - \delta\|_B \\ &\le \|M_A\|_B \, \|x^m - x^*\|_B + \varepsilon\|\delta\|_B \\ &\le \left[\|M_A\|_B + \varepsilon\|B^{-1/2}AB^{-1/2}\|_2\right] \|x^m - x^*\|_B.\end{aligned}$$

The identity $\|B^{-1/2}AB^{-1/2}\|_2 = \|A^{1/2}B^{-1/2}\|_2^2 = \|A^{1/2}B^{-1}A_2^{1/2}\|$ (cf. (B.21a)) proves the contraction number (10.26b). The definition of $M_k(Ax - b)$ and $N_k(Ax - b)$ in (10.26a) is obvious. Since the CG method is nonlinear (analogous to Remark 9.8a), Φ_k is also. □

Remark 10.22. The composed iteration Φ_k defined in Lemma 10.21 is not well suited to be the basic iteration for the Chebyshev or CG method because the matrix $W_k(\delta) = A(I - M_k(\delta))$, $\delta = Ax - b$, of the third normal form of Φ_k depends on the value of the iterates x^m. Concerning this problem, see Golub–Overton [156] and Axelsson–Vassilevski [17].

10.2.8 Three-Term Recursion for p^m

Finally, we describe another formulation of the CG method. The three-term formulation is less important for the CG method itself, but is required as a stabilisation of, e.g., the CR algorithm in §10.3.3.

Inserting definition (10.23b,d): $r^{m+1} := r^m - \lambda A p^m$ into (10.23f), one obtains $p^{m+1} := Nr^m - \lambda NAp^m + \mathrm{const}\cdot p^m$. Since the scaling of the search direction is irrelevant, we may replace p^{m+1} by $-p^{m+1}/\lambda$. Because $Nr^m \in \mathcal{K}_{m+1}(NA, Nr^0)$ and $p^m \in \mathcal{K}_{m+1}(NA, Nr^0)$, the following ansatz is justified:

$$p^{m+1} := NA\,p^m - \sum_{\mu=0}^{m} \alpha_{\mu,m+1}\, p^{m-\mu}. \tag{10.27}$$

Condition (10.25) states that $\langle Ap^{m+1},\, p^m\rangle = 0$ and determines the coefficients

$$\alpha_{0,m+1} = \langle ANA\,p^m, p^m\rangle \,/\, \langle Ap^m, p^m\rangle\,,$$

since $\langle Ap^{m-\mu}, p^m\rangle = 0$ for $\mu > 0$. Similarly we obtain

$$a_{1,m+1} = \langle ANA\,p^m, p^{m-1}\rangle \,/\, \langle Ap^{m-1}, p^{m-1}\rangle\,.$$

Lemma 10.23. *Assume $(AN)^{\mathsf{H}} = NA$. Then the coefficients in (10.27) satisfy $\alpha_{\mu,m+1} = 0$ for $\mu \ge 2$.*

Proof. The condition $\langle Ap^{m+1}, p^{m-\mu}\rangle = 0$ yields the equation

$$\langle ANA\,p^m, p^{m-\mu}\rangle = \alpha_{\mu,m}\,\langle Ap^{m-\mu}, p^{m-\mu}\rangle\,.$$

The assertion follows from

$$\begin{aligned}\langle ANAp^m, p^{m-\mu}\rangle &= \langle Ap^m, NAp^{m-\mu}\rangle \\ &\underset{(10.27)}{=} \left\langle Ap^m,\, p^{m+1-\mu} + \sum_{\nu=0}^{m-\mu} \alpha_{\mu,m+1-\mu}\, p^{m-\mu-\nu}\right\rangle = 0\,. \qquad \square\end{aligned}$$

Thanks to Lemma 10.23, p^{m+1} can be calculated from the three-term recursion

$$\begin{aligned} p^{m+1} &= NA\,p^m - \alpha_0\,p^m - \alpha_1\,p^{m-1} \\ \text{with } \alpha_0 &= \frac{\langle ANAp^m, p^m\rangle}{\langle Ap^m, p^m\rangle},\quad \alpha_1 = \frac{\langle ANAp^m, p^{m-1}\rangle}{\langle Ap^{m-1}, p^{m-1}\rangle},\end{aligned}$$

where the last term is absent for $m = 0$ (formally, we may set $\alpha_1 = 0$, $p^{-1} = 0$). The CG algorithm (10.23a–f) is equivalent to (10.28a–e):

start: x^0 arbitrary; $r^0 := b - Ax^0$; $p^{-1} := 0$; $p^0 := Nr^0$; (10.28a)

iteration: for $m = 0, 1, 2, \ldots$ while $\langle Ap^m, p^m\rangle \neq 0$:

$x^{m+1} := x^m + \lambda_{\text{opt}} p^m$; $\quad r^{m+1} := r^m - \lambda_{\text{opt}} Ap^m$ with (10.28b)

$\lambda_{\text{opt}} := \langle r^m, p^m\rangle \,/\, \langle Ap^m, p^m\rangle$; (10.28c)

$p^{m+1} := NA\,p^m - \alpha_0\,p^m - \alpha_1\,p^{m-1}$ with (10.28d)

$\alpha_0 := \frac{\langle ANAp^m, p^m\rangle}{\langle Ap^m, p^m\rangle}$; $\quad \alpha_1 = \frac{\langle ANAp^m, p^{m-1}\rangle}{\langle Ap^{m-1}, p^{m-1}\rangle}$; (10.28e)

where again $\alpha_1 := 0$ is chosen for $m = 0$.

The next theorem is based on the very weak assumption $(AN)^{\mathsf{H}} = NA$ which follows from $A > 0$, $N > 0$ or from $A = A^{\mathsf{H}}$, $N = N^{\mathsf{H}}$.

Theorem 10.24. *Assume $(AN)^{\mathsf{H}} = NA$. Let m_0 be the maximal index such that the directions generated in (10.28d) satisfy $\langle Ap^m, p^m\rangle \neq 0$ for all $0 \le m \le m_0$.*
(a) The quantities x^m, r^m, p^m $(0 \le m \le m_0)$ in (10.28a–e) satisfy $r^m = b - Ax^m$ and

$$\begin{aligned}&\langle Ap^m, p^\ell\rangle = \langle r^m, Nr^\ell\rangle = \langle r^m, p^\ell\rangle = 0 \qquad \textit{for } 0 \le \ell < m, \\ &\operatorname{span}\{p^0, \ldots, p^m\} = \mathcal{K}_{m+1}(NA, Nr^0) \supset \operatorname{span}\{Nr^0, \ldots, Nr^m\}\end{aligned}$$

for $0 \le m \le m_0$. More precisely, we have

$$Nr^m \in \operatorname{span}\{p^m, p^{m-1}\} \qquad \textit{for } 0 \le m \le m_0\,. \tag{10.29}$$

(b) As long as algorithm (10.22a–e) does not terminate, (10.22a–e) and (10.28a–e) produce the same iterates x^m, whereas the search directions p^m may differ by a nonvanishing factor.

(c) Assume, in addition, that $N + N^{\mathsf{H}} > 0$. If the iteration (10.28a–e) terminates because of $p^m = 0$, the iterate x^m is already the exact solution.

Proof. The assertion is proved by induction. The start $m = 0$ is trivial. Let the statements hold for $0, 1, \ldots, m-1$. We abbreviate $\mathcal{K}_m(NA, Nr^0)$ by $\mathcal{K}_m$.

(i) For the proof of $\langle Ap^m, p^\ell \rangle = 0$, we use (10.28d):

$$Ap^m = ANA\, p^{m-1} - \alpha_0\, Ap^{m-1} - \alpha_1\, Ap^{m-2}.$$

For $\ell \in \{m-2, m-1\}$, the definitions of α_0 and α_1 prove $\langle Ap^m, p^\ell \rangle = 0$. Let $\ell \le m-3$. The assumption $(AN)^{\mathsf{H}} = NA$ yields $\langle ANA\, p^{m-1}, p^\ell \rangle = \langle A\, p^{m-1}, NA\, p^\ell \rangle$. From $p^\ell \in \operatorname{span}\{p^0, \ldots, p^\ell\} = \mathcal{K}_{\ell+1}$, we conclude that

$$NA\, p^\ell \in \mathcal{K}_{\ell+2} \subset \mathcal{K}_{m-1} = \operatorname{span}\{p^0, \ldots, p^{m-2}\} \perp A\, p^{m-1}.$$

Since Ap^{m-1} and $A\, p^{m-2}$ are also perpendicular to p^ℓ, $\langle Ap^m, p^\ell \rangle = 0$ follows.

(ii) By induction $\mathcal{K}_m = \operatorname{span}\{p^0, \ldots, p^{m-1}\}$ holds. We use again (10.28d): $p^m = NA\, p^{m-1} - \alpha_0\, p^{m-1} - \alpha_1\, p^{m-2} \in NA\mathcal{K}_m + \operatorname{span}\{p^0, \ldots, p^{m-1}\} \subset \mathcal{K}_{m+1}$. This proves $\operatorname{span}\{p^0, \ldots, p^m\} \subset \mathcal{K}_{m+1}$. On the other hand, we have

$$\mathcal{K}_{m+1} \subset \mathcal{K}_m + NA\mathcal{K}_m = \operatorname{span}\{p^0, \ldots, p^{m-1}\} + NA \operatorname{span}\{p^0, \ldots, p^{m-1}\} \ni p^m$$

because of (10.28d). This proves the reverse inclusion $\mathcal{K}_{m+1} \subset \operatorname{span}\{p^0, \ldots, p^m\}$.

(iii) $0 = \langle r^m, p^\ell \rangle = \langle r^{m-1}, p^\ell \rangle - \lambda_{\mathrm{opt}} \langle Ap^m, p^\ell \rangle = 0$ holds for $\ell < m-1$ by induction and follows for $\ell = m-1$ by definition of λ_{opt}. This proves $r^m \perp \mathcal{K}_m$.

(iv) Now we prove (10.29). The definition of r^m in (10.28b) shows that $Nr^m = Nr^{m-1} - \lambda NAp^{m-1}$. By induction $Nr^{m-1} \in \operatorname{span}\{p^{m-2}, p^{m-1}\}$ holds, while (10.28d) yields $NAp^{m-1} = p^m + \alpha_0\, p^{m-1} + \alpha_1\, p^{m-2} \in \operatorname{span}\{p^{m-2}, p^{m-1}, p^m\}$. Hence ANr^m has the representation

$$ANr^m = b_0 Ap^m + b_1 Ap^{m-1} + b_2 Ap^{m-2}.$$

The scalar product with p^{m-2} yields the value $b_2 = \frac{\langle ANr^m, p^{m-2} \rangle}{\langle Ap^{m-2}, p^{m-2} \rangle}$. By assumption $(AN)^{\mathsf{H}} = NA$, $\langle ANr^m, p^{m-2} \rangle = \langle r^m, NAp^{m-2} \rangle$ holds. Since $NAp^{m-2} \in NA\mathcal{K}_{m-1} \subset \mathcal{K}_m$, part (iii) proves $b_2 = 0$ and $Nr^m \in \operatorname{span}\{p^{m-1}, p^m\}$ follows.

(v) $\langle r^m, Nr^\ell \rangle = 0$ for $\ell < m$ is a consequence of (10.29) and $r^m \perp \mathcal{K}_m$.

(vi) Part (b) holds, since another scaling of p^m does not change x^m.

(vii) If $p^m = 0$, (10.29) implies $Nr^m \in \operatorname{span}\{p^0, \ldots, p^{m-1}\}$. Since $\langle r^m, p^\ell \rangle = 0$ for $\ell < m$, we conclude that $\langle r^m, Nr^m \rangle = 0$ and the assumption $N + N^{\mathsf{H}} > 0$ implies that $r^m = 0$. □

10.3 Method of Conjugate Residuals (CR)

10.3.1 Algorithm

In the case of the gradient method, a residual oriented transformation is discussed in §9.2.4.2. Under the assumption $A > 0$, the iteration $\Phi \in \mathcal{L}_{\rm pos}$ with $N > 0$ is transformed to $\bar{x}^{m+1} := \bar{x}^m - (\bar{A}\bar{x}^m - \bar{b})$ with $\bar{A} := A^{1/2}NA^{1/2} > 0$, $\bar{b} := A^{1/2}Nb$, $\bar{x}^m := A^{1/2}x^m$, $\bar{p}^m = A^{1/2}p^m$, $\bar{r}^m = A^{1/2}Nr^m$ (cf. (9.19)). As in (10.21a–e), we can formulate the CG algorithm (10.21a–e) with A, x, b, p, r replaced by $\bar{A}, \bar{x}, \bar{b}, \bar{p}, \bar{r}$. Then we substitute these quantities by the original ones and obtain the following algorithm $\Upsilon_{\rm CR}[\Phi]$:

$$\begin{aligned}
&\text{start:} && x^0 \text{ arbitrary}; \quad r^0 := b - Ax^0; \quad p^0 := Nr^0; \quad m = 0;\\
&\text{iteration:} && x^{m+1} := x^m + \lambda_{\rm opt}\, p^m \text{ with}\\
&&& \lambda_{\rm opt} := \lambda_{\rm opt}(r^m, NAp^m, N) = \tfrac{\langle Nr^m, Ap^m\rangle}{\langle NAp^m, Ap^m\rangle};\\
&&& r^{m+1} := r^m - \lambda_{\rm opt}\, A\, p^m;\\
&&& p^{m+1} := Nr^{m+1} - \tfrac{\langle ANr^{m+1}, NAp^m\rangle}{\langle NAp^m, Ap^m\rangle}\, p^m;
\end{aligned} \tag{10.30}$$

For $N = I$, this method is equivalent to the *method of the conjugate residuals* (CR) of Stiefel [354].

The following statements follow from the properties of the CG method applied to $\bar{A}$, $\bar{x}$, $\bar{b}$, $\bar{p}$, $\bar{r}$ after a reformulation by A, x, b, p, r.

Proposition 10.25. *(a) The number* $m_0 = \deg_{\bar{A}}(\bar{e}^0) = \deg_{NA}(e^0) = \deg_{AN}(r^0)$ *is the same as in Lemma 10.18a.*

(b) The directions p^m *are* ANA*-orthogonal.*

(c) The statements in (10.11a,b) become

$$ANr^m \perp \mathcal{K}_m(NA, Nr^0) = \operatorname{span}\{p^0, \ldots, p^{m-1}\} = \operatorname{span}\{Nr^0, \ldots, Nr^{m-1}\}.$$

(d) The convergence rate c *is the same* c *as in Theorem 10.17. Note that the involved norms are different. Here the residuals are the minimisers of*

$$\min\left\{\|N^{1/2}A(x - x^*)\|_2 : x = x^0 + \mathcal{K}_m(NA, Nr^0)\right\}$$

and are bounded by

$$\|N^{1/2}r^m\|_2 \le \frac{2c^m}{1 + c^{2m}}\|N^{1/2}r^0\|_2.$$

In the case of $N = I$, the CR method corresponds to the formulation in §10.1.5.2 with the Krylov spaces $\mathcal{U}_m = \mathcal{K}_m(A, r^0)$ and $\mathcal{V}_m = A\mathcal{K}_m(A, r^0)$ (note that $A = A^{\sf H}$).

10.3.2 Application to Hermitian Matrices

In the following we assume

$$A = A^{\mathsf{H}} \text{ regular} \qquad \text{and} \qquad N > 0.$$

Since $A^{\mathsf{H}}NA > 0$, the denominator $\langle NAp^m, Ap^m\rangle$ in (10.30) vanishes if and only if $p^m = 0$. Hence, the algorithm (10.30) is applicable as long as $p^m \neq 0$. In the indefinite case, however, there is a severe difference to the conjugate gradient method. The CG method for $A > 0$ terminates with $r^m = 0$, i.e., $x^m = A^{-1}b$ ('lucky breakdown'), whereas for an indefinite matrix A an unlucky breakdown may occur.

Remark 10.26. Assume that $A = A^{\mathsf{H}}$ has positive and negative eigenvalues. Then there are initial values $x^0 \neq A^{-1}b$ so that $\lambda_{\mathrm{opt}}(r^0, NAp^0, N) = 0$. Then $p^1 = 0$ leads to a breakdown, while $x^1 = x^0$ is still different from the true solution.

Proof. $\lambda_{\mathrm{opt}}(r^0, NAp^0, N) = 0$ follows from $\langle Ap^0, p^0\rangle = \langle NANp^0, p^0\rangle = 0$ which holds for certain $p^0 \neq 0$. Since $p^1 \perp_{NAN} p^0$ and $p^1 \in \mathrm{span}\{p^0\}$ because of $\lambda_{\mathrm{opt}} = 0$, $p^1 = 0$ follows. □

Lemma 10.27. *Let $A = A^{\mathsf{H}}$ and $N > 0$. Assume that the algorithm (10.30) for a fixed x^0 is applicable for all $0 \leq m \leq m_0$. Then, as in Proposition 10.25b–c, the search directions p^m are ANA-orthogonal and*

$$ANr^m \perp \mathcal{K}_m(NA, Nr^0) = \mathrm{span}\{p^0, \ldots, p^{m-1}\} = \mathrm{span}\{Nr^0, \ldots, Nr^{m-1}\}$$

holds. The iterate x^m in (10.30) minimises the norm

$$\|N^{1/2}r^m\|_2 = \min\left\{\|N^{1/2}A(x - x^*)\|_2 : x \in x^0 + \mathcal{K}_m(NA, Nr^0)\right\}. \quad (10.31)$$

Proof. (i) Concerning the first two statements, the previous proof by induction can be repeated without change.

(ii) Note that $x^m - x^0 \in \mathcal{K}_m := \mathrm{span}\{p^0, \ldots, p^{m-1}\} = \mathcal{K}_m(NA, Nr^0)\}$. Because of $ANA > 0$, $\{\langle A(x - x^*), NA(x - x^*)\rangle : x - x^0 \in \mathcal{K}_m\}$ attains its minimum at $x = x^m$ if and only if the gradient $ANA(x^m - x^*) = -ANr^m$ is orthogonal to $\mathcal{K}_m$. This, however, is the second statement of the lemma. □

The reason for the breakdown mentioned in Remark 10.26 is that the spaces $\mathrm{span}\{Nr^0, Nr^1\} = \mathrm{span}\{Nr^0\}$ and $\mathrm{span}\{Nr^0, NANr^0\}$ differ. This fact suggests that the subspace $\mathcal{K}_m(NA, Nr^0) = \mathrm{span}\{Nr^0, \ldots, (NA)^{m-1}Nr^0\}$ is better suited than $\mathrm{span}\{Nr^0, \ldots, Nr^{m-1}\}$.

Even if $\langle ANr^m, Nr^m\rangle = 0$ does not occur during the calculations, it may happen that Nr^m is 'almost' contained in $\mathcal{K}_m(NA, Nr^0)$, leading to a numerical instability of the algorithm. One remedy is constructing the search directions p^m by the three-term recursion explained in §10.2.8.

10.3.3 Stabilised Method of Conjugate Residuals

Using the three-term recursion in algorithm (10.30), we obtain the following algorithm $\Upsilon_{\mathrm{CR}}^{\mathrm{stab}}[\Phi]$:

$\Upsilon_{\mathrm{CR}}^{\mathrm{stab}}[\Phi]$	stabilised method of the conjugate residuals	(10.32)
start:	x^0 arbitrary; $r^0 := b - Ax^0$; $p^{-1} := 0$; $p^0 := Nr^0$;	(10.32a)
iteration:	for $m = 0, 1, 2, \ldots$ while $\langle Ap^m, NAp^m \rangle \neq 0$:	
	$x^{m+1} := x^m + \lambda p^m$; $\quad r^{m+1} := r^m - \lambda Ap^m \quad$ with	(10.32b)
	$\lambda := \langle r^m, NAp^m \rangle \,/\, \langle Ap^m, NAp^m \rangle$;	(10.32c)
	$p^{m+1} := NAp^m - \alpha_0 p^m - \alpha_1 p^{m-1} \quad$ with	(10.32d)
	$\alpha_0 := \frac{\langle ANAp^m, NAp^m \rangle}{\langle Ap^m, NAp^m \rangle}$; $\alpha_1 = \frac{\langle ANAp^m, NAp^{m-1} \rangle}{\langle Ap^{m-1}, NAp^{m-1} \rangle}$;	(10.32e)

Exercise 10.28. By $ANAp^m$ appearing in (10.32e), algorithm (10.32a–e) seems to cost two multiplications by the matrix A per iteration step. Rewrite algorithm (10.32a–e) with an additional recursion for $a^m := Ap^m$ so that only one multiplication by A is needed.

Theorem 10.29. *Assume $(AN)^{\mathsf{H}} = NA$. Let m_0 be the maximal index such that the directions generated in (10.28d) satisfy $\langle Ap^m, NAp^m \rangle \neq 0$ for all $0 \leq m \leq m_0$.*
(a) The quantities x^m, r^m, p^m $(0 \leq m \leq m_0)$ in (10.28a–e) satisfy $r^m = b - Ax^m$ and

$$\langle Ap^m, NAp^\ell \rangle = \langle r^m, NANr^\ell \rangle = \langle r^m, NAp^\ell \rangle = 0 \qquad \text{for } 0 \leq \ell < m,$$
$$\operatorname{span}\{p^0, \ldots, p^m\} = \mathcal{K}_{m+1}(NA, Nr^0) \supset \operatorname{span}\{Nr^0, \ldots, Nr^m\}$$

for $0 \leq m \leq m_0$. More precisely, we have

$$Nr^m \in \operatorname{span}\{p^m, p^{m-1}\} \qquad \text{for } 0 \leq m \leq m_0. \tag{10.33}$$

(b) As long as algorithm (10.30) does not terminate, (10.30) and (10.32a–e) produce the same iterates x^m, whereas the search directions may differ by a nonvanishing factor.
(c) Assume in addition that $N + N^{\mathsf{H}} > 0$. If the iteration (10.32a–e) terminates because of $p^m = 0$, the iterate x^m is already the exact solution.

Proof. The assertion is proved by induction. The start $m = 0$ is trivial. Let the statements hold for $0, 1, \ldots, m-1$. We abbreviate $\mathcal{K}_m(NA, Nr^0)$ by $\mathcal{K}_m$.

(i) For the proof of $\langle Ap^m, NAp^\ell \rangle = 0$, we use (10.32d):

$$Ap^m = ANA\, p^{m-1} - \alpha_0\, Ap^{m-1} - \alpha_1\, Ap^{m-2}.$$

For $\ell \in \{m-2, m-1\}$, the definitions of α_0 and α_1 prove $\langle Ap^m, NAp^\ell \rangle = 0$. Let $\ell \leq m-3$. The assumption $(AN)^{\mathsf{H}} = NA$ yields $\langle ANA\, p^{m-1}, NAp^\ell \rangle = \langle A\, p^{m-1}, (NA)^2 p^\ell \rangle$. From $p^\ell \in \operatorname{span}\{p^0, \ldots, p^\ell\} = \mathcal{K}_{\ell+1}$, we conclude that $NA\, p^\ell \in \mathcal{K}_{\ell+2} \subset \mathcal{K}_{m-1} = \operatorname{span}\{p^0, \ldots, p^{m-2}\}$ and $NA\mathcal{K}_{m-1} \perp A\, p^{m-1}$. Since Ap^{m-1} and Ap^{m-2} are also perpendicular to NAp^ℓ, $\langle Ap^m, NAp^\ell \rangle = 0$ follows.

(ii) By induction, $\mathcal{K}_m = \operatorname{span}\{p^0, \ldots, p^{m-1}\}$ holds. We again use (10.32d): $p^m = NA\, p^{m-1} - \alpha_0\, p^{m-1} - \alpha_1\, p^{m-2} \in NA\mathcal{K}_m + \operatorname{span}\{p^{m-2}, p^{m-1}\} \subset \mathcal{K}_{m+1}$. This proves $\operatorname{span}\{p^0, \ldots, p^m\} \subset \mathcal{K}_{m+1}$. On the other hand, the inclusion

$$\mathcal{K}_{m+1} \subset \mathcal{K}_m + NA\mathcal{K}_m = \operatorname{span}\{p^0, \ldots, p^{m-1}\} + NA \operatorname{span}\{p^0, \ldots, p^{m-1}\} \ni p^m$$

follows from (10.32d) proving the reverse inclusion $\mathcal{K}_{m+1} \subset \operatorname{span}\{p^0, \ldots, p^m\}$.

(iii) $0 = \langle r^m, NAp^\ell \rangle = \langle r^{m-1}, NAp^\ell \rangle - \lambda_{\text{opt}} \langle Ap^m, NAp^\ell \rangle = 0$ holds for $\ell < m-1$ by induction and follows for $\ell = m-1$ by definition of λ_{opt}. This proves $r^m \perp NA\,\mathcal{K}_m$.

(iv) For the proof of (10.33), use the definition of r^m in (10.32b): $Nr^m = Nr^{m-1} - \lambda NAp^{m-1}$. By induction $Nr^{m-1} \in \operatorname{span}\{p^{m-2}, p^{m-1}\}$ holds, while (10.32d) yields $NAp^{m-1} = p^m + \alpha_0\, p^{m-1} + \alpha_1\, p^{m-2} \in \operatorname{span}\{p^{m-2}, p^{m-1}, p^m\}$. Hence Nr^m has the representation

$$Nr^m = b_0\, p^m + b_1\, p^{m-1} + b_2\, p^{m-2}.$$

Using part (i), we obtain $b_2 = \frac{\langle ANr^m, NAp^{m-2} \rangle}{\langle Ap^{m-2}, NAp^{m-2} \rangle}$ by taking the scalar product of ANr^m with $NA\, p^{m-2}$. $(AN)^{\mathsf{H}} = NA$ implies that $\langle ANr^m, NAp^{m-2} \rangle = \langle r^m, (NA)^2 p^{m-2} \rangle$ holds. Since $NA\, p^{m-2} \in NA\,\mathcal{K}_{m-1} \subset \mathcal{K}_m$, part (iii) proves $b_2 = 0$, and $Nr^m \in \operatorname{span}\{p^{m-1}, p^m\}$ follows.

(v) $\langle r^m, NANr^\ell \rangle = 0$ for $\ell < m$ is a consequence of (10.29) and $r^m \perp NA\mathcal{K}_m$.

(vi) Statement (b) follows as in Theorem 10.24. For Part (c), use that $p^m = 0$ implies $Nr^m = cp^{m-1}$ for some $c \in \mathbb{K}$. Obviously, $c = 0$ and $r^m = 0$ follow from $0 = \langle ANr^m, NAp^{m-1} \rangle = \langle r^m, (NA)^2 p^{m-1} \rangle$. This equation holds since $p^m = 0$ implies $\mathcal{K}_m = \mathcal{K}_{m+1}$ and therefore $\langle r^m, (NA)^2 p^{m-1} \rangle = 0$ because of $r^m \perp NA\mathcal{K}_m = NA\,\mathcal{K}_{m+1} = (NA)^2 \mathcal{K}_m$. □

10.3.4 Convergence Results for Indefinite Matrices

Lemma 10.27 carries over to algorithm (10.32) since it produces the same iterates x^m. The error estimate in Proposition 10.25d cannot be transferred directly to indefinite matrices because the spectrum of NA no longer lies in the positive part. In the general case, the resulting convergence speed is definitely slower than in the positive definite case. Note that the quantity c below is defined in terms of κ, whereas c in Proposition 10.25d is derived from $\sqrt{\kappa}$. Hence, in general, the typical acceleration by the conjugate gradient technique does not take place, but notice Theorem 10.31.

Theorem 10.30. *Assume $N > 0$, $A = A^{\mathsf{H}}$ regular, and $\kappa = \kappa(NA)$. Then the iterates x^m of algorithm (10.32) satisfy the error estimate*

$$\|N^{1/2}A(x^m - x^*)\|_2 \le \frac{2c^\mu}{1+c^{2\mu}} \|N^{1/2}A(x^0 - x^*)\|_2 \tag{10.34}$$

with $c := (\kappa-1)/(\kappa+1)$ and $\frac{m}{2} - 1 < \mu \le \frac{m}{2}$, where $\mu \in \mathbb{N}_0$. Hence, the asymptotic convergence rate amounts to $\sqrt{c} = 1 - 1/\kappa + \mathcal{O}(\kappa^{-2})$.

Proof. For odd m, we exploit the monotone convergence $\|N^{1/2}A\,e^{m+1}\|_2 \le \|N^{1/2}A\,e^m\|_2$ following from (10.31). Therefore, consider an even $m = 2\mu$. Analogously to Remark 10.13,

$$\|N^{1/2}Ae^m\|_2 \le \Big(\max_{\lambda\in\sigma(NA)} |P_m(1-\lambda)| \Big) \|N^{1/2}Ae^0\|_2 \tag{10.35}$$

holds for any polynomial $P_m \in \mathcal{P}_m$ with $P_m(1) = 1$. Let p_μ be a polynomial of degree $\le \mu = \frac{m}{2}$ with $p_\mu(1) = 1$. $P_m(\xi) := p_\mu(\xi(2-\xi))$ is of degree $\le m$ and satisfies $P_m(1) = 1$. Evidently, $P_m(1-\lambda) = p_\mu(1-\lambda^2)$ holds, from which

$$\|N^{1/2}A\,e^m\|_2 \le \max\big\{\big|p_\mu(1-\lambda^2)\big| : \lambda \in \sigma(NA)\big\}\ \|N^{1/2}A\,e^0\|_2.$$

follows. If $\lambda \in \sigma(NA)$, we have $|\lambda| \in [\gamma, \Gamma]$ and $\lambda^2 \in [\gamma^2, \Gamma^2]$, where

$$\gamma := 1/\rho(A^{-1}N^{-1}) = \min\{|\lambda| : \lambda \in \sigma(NA)\}, \quad \Gamma := \rho(NA).$$

Since $[\gamma^2, \Gamma^2]$ lies in the positive half-axis, the Chebyshev polynomial (8.27a) yields the following estimate with $c = (\Gamma-\gamma)/(\Gamma+\gamma) = (\kappa-1)/(\kappa+1)$:

$$\max\big\{\big|p_\mu(1-\lambda^2)\big| : \lambda \in \sigma(NA)\big\} \le \max_{\gamma^2\le\xi\le\Gamma^2} |p_\mu(1-\xi)| \le \frac{2c^\mu}{1+c^{2\mu}}. \qquad \square$$

Estimate (10.34) may be too pessimistic. Often a milder form of indefiniteness occurs. If, for instance, the Helmholtz equation $-\Delta u - cu = f$ with $c > 0$ is discretised, A has eigenvalues λ_μ^h $(1 \le \mu \le n = n_h)$, where

$$\lambda_\mu^h = \lambda_{\mu,0}^h - c, \ \ 0 < \lambda_{\mu,0}^h : \text{eigenvalues of the Poisson model case (3.1a).}$$

For $h \to 0$, the discrete eigenvalues λ_μ^h tend to the Laplace eigenvalues λ_μ which cannot accumulate (cf. [193, §11]). Therefore the following properties are satisfied:

$$\text{The number } k \text{ of negative eigenvalues is bounded for } h \to 0\,. \tag{10.36a}$$

$$\text{For all } h > 0, \text{ the nonpositive eigenvalues belong to } [-c_1, -c_0] \text{ with } 0 < c_0 \le c_1\,. \tag{10.36b}$$

$$\text{The positive eigenvalues are in } [\gamma, \Gamma] \text{ with } 0 < \gamma \le \Gamma\,. \tag{10.36c}$$

Let $k = k_h$ be the number of negative eigenvalues λ_μ^h, $1 \le \mu \le k$. Define

$$\pi_h(1-\xi) = \prod_{\mu=1}^{k}(1-\xi/\lambda_\mu^h).$$

Let p_μ be the Chebyshev polynomial (8.27a) of degree $\mu := m-k$ for $a = 1-\Gamma$ and $b = 1-\gamma$. The product $P_m(\xi) := \pi_h(\xi)p_\mu(\xi)$ is of degree m with $P_m(1) = 1$. Since $P_m(1-\lambda) = 0$ holds for the negative eigenvalues $\lambda \in \sigma(NA)$, the factor on the right-hand side in (10.35) reduces to

$$\max\{\,|P_m(1-\lambda)| : \lambda \in [\gamma,\Gamma]\,\} \le \max\{\,|\pi_h(1-\lambda)| : \lambda \in [\gamma,\Gamma]\,\}\,\frac{2c^\mu}{1+c^{2\mu}}$$

with $c := \frac{\sqrt{\Gamma}-\sqrt{\gamma}}{\sqrt{\Gamma}+\sqrt{\gamma}}$ (cf. (10.36c)). $|\pi_h(1-\lambda)|$ can be estimated by $(1+\Gamma/c_0)^k$ (cf. (10.36b)). The m-th root of the bound $(1+\Gamma/c_0)^k\frac{2c^\mu}{1+c^{2\mu}}$ tends to c. Hence, the asymptotic convergence rate is not influenced by the negative eigenvalues. This proves the next theorem.

Theorem 10.31. *Assume $A = A^{\mathsf{H}}$, $N > 0$, and let the eigenvalues of NA satisfy (10.36a–c). Replace the spectral condition number $\kappa(NA)$ by the possibly smaller number $\kappa := \Gamma/\gamma$ (γ, Γ in (10.36c)). Then the error estimate for the algorithm (10.32) of the conjugate residuals reads*

$$\|N^{1/2}A(x^m - x^*)\|_2 \le 2\left(\frac{1+\Gamma/c_0}{c}\right)^k \|N^{1/2}A(x^0 - x^*)\|_2$$

with the asymptotic convergence rate $c := \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1} = \frac{\sqrt{\Gamma}-\sqrt{\gamma}}{\sqrt{\Gamma}+\sqrt{\gamma}}$ and c_0 in (10.36a).

An alternative to the method (10.32) of conjugate residuals is the application of the standard CG method to the Kaczmarz iteration (cf. §5.6.3). Then the convergence speed is as slow as in Theorem 10.30. In the situation of (10.36a–c), the convergence rate would not improve.

m	value in the middle	$\frac{\|e^m\|_A}{\|e^{m-1}\|_A}$
1	0.2222124445	0.157356
2	0.4269164370	0.537790
3	0.4510237348	0.439627
4	0.4759275765	0.438732
10	0.4998558015	0.384330
20	0.5000000047	0.338399
21	0.5000000029	0.389211
22	0.5000000012	0.407384
23	0.5000000003	0.317164
24	0.5000000002	0.442606
25	0.5000000000	0.691876
26	0.5000000000	0.768926
27	0.5000000000	0.955932
28	0.5000000000	1.02596
29	0.5000000000	1.03161
30	0.5000000000	1.03076

Table 10.3 $\Upsilon_{\mathrm{CR}}^{\mathrm{stab}}[\Phi_{\mathrm{ILU}}]$: CR method for the Poisson model problem applied to the 5-point-ILU iteration ($\omega = -1,\ h = 1/32$).

10.3.5 Numerical Examples

For reasons of comparison, we first test the positive definite Poisson model problem with $h = \frac{1}{32}$. We apply the CR method to the ILU iteration (five-point pattern) with the same parameters as in Table 10.2. The results given in Table 10.3 are similar to those of the standard CG method in Table 10.2.

Next, we choose the discrete Helmholtz equation $-\Delta u - 50u = f$ as an indefinite example. Here the matrix A is the Poisson model matrix minus $50\,I$. It has three negative eigenvalues $\lambda_1 = -30.277$,

$\lambda_2 = \lambda_3 = -0.7866$, while $\lambda_4 = 28.7$ is the smallest positive eigenvalue. For the modified ILU decomposition, the diagonal must be enlarged by 55 (cf. Remark 7.44). The results of Table 10.4 show that the reduction factor moves toward the asymptotic convergence rate and is of a size similar to the positive definite case of Table 10.3. The stagnation for $m \geq 33$ is due to rounding. In both examples the algorithm behaves stable.

m	value in the middle	$\frac{\|e^m\|_A}{\|e^{m-1}\|_A}$	m	value in the middle	$\frac{\|e^m\|_A}{\|e^{m-1}\|_A}$
1	-1.129805206	1.36998	15	0.5017304154	0.97466
2	0.5616735534	0.41788	16	0.5019212154	0.86920
3	0.9170148791	0.77945	17	0.5018558645	0.56086
4	0.7375934000	0.78685	18	0.5017568935	0.32511
5	0.6675855715	0.88467	19	0.5008067252	0.27287
6	0.5834957931	0.95835	20	0.5003741869	0.19130
7	0.5440078825	0.99228	21	0.5003841894	0.35875
8	0.5222771713	1.00338	30	0.4999998664	0.30740
9	0.5099064832	1.00768	31	0.4999999994	0.40910
10	0.5053055088	1.00956	32	0.4999999838	0.69469
11	0.5029466483	1.01213	33	0.4999999962	0.90769
12	0.5020970259	1.01621	34	0.4999999984	0.97862
13	0.5015223028	1.01159	35	0.4999999986	0.98121
14	0.5015388760	1.00335	36	0.4999999994	0.99385

Table 10.4 CR method $\Upsilon_{\mathrm{CR}}^{\mathrm{stab}}[\Phi_{\mathrm{ILU}}]$ for an indefinite problem based on the 5-point-ILU iteration.

10.4 Method of Orthogonal Directions

The CG method (10.15a–e) minimises the error $\|e^m\|_A = \|A^{1/2}e^m\|_2$ with respect to the energy norm over the Krylov space $\mathcal{K}_m(A, r^0)$. The method of conjugate residuals (with $N = I$) minimises the residual $\|r^m\|_2 = \|Ae^m\|_2$ over the same space. A more natural norm would be $\|e^m\|_2$. Then the search directions p^m should be orthogonal in the usual sense. This can be achieved by replacing the Krylov space $\mathcal{K}_m(A, r^0)$ by $A\mathcal{K}_m(A, r^0) = \mathcal{K}_m(A, Ar^0)$. The corresponding algorithm (10.37) is described by Fridman [140] (1963) and called the *method of orthogonal directions* (OD) since the search directions form an orthogonal system if $N = I$. The application of OD to an iteration Φ with the matrix $N[\Phi] > 0$ takes the form

$\Upsilon_{\mathrm{OD}}[\Phi]$	*method of orthogonal directions*	(10.37)
start:	x^0 arbitrary; $r^0 := b - Ax^0$; $q^{-1} := r^0$; $q^0 := ANq^{-1}$;	(10.37a)
iteration:	for $m = 0, 1, 2, \ldots$ while $q^m \neq 0$:	
	$x^{m+1} := x^m + \lambda p^m$; $\quad r^{m+1} := r^m - \lambda Ap^m$ with	(10.37b)
	$p^m := Nq^m$; $\rho_m := \langle q^m, p^m \rangle$; $\lambda := \frac{\langle r^m, p^{m-1} \rangle}{\rho_m}$;	(10.37c)
	$q^{m+1} := Ap^m - \alpha_0 q^m - \alpha_1 q^{m-1}$ with	(10.37d)
	$\alpha_0 := \langle Ap^m, p^m \rangle / \rho_m$; $\quad \alpha_1 := \langle Ap^m, p^{m-1} \rangle / \rho_m$;	(10.37e)

where $\alpha_1 := 0$ is set for $m = 0$. The method (10.37) is unstable as we observe from the results in Tables 10.5 and 10.6. A stabilisation is given by Stoer [355, (3.16)]. On the other hand, we can do without it if only a few iteration steps are required.

m	value in the middle	$\|e^m\|_2$	$\frac{\|e^m\|_2}{\|e^{m-1}\|_2}$	$\sqrt[m]{\frac{\|e^m\|_2}{\|e^0\|_2}}$
1	-4.57492859$_{10}$-2	2.93956$_{10}$-1	3.92836$_{10}$-1	3.92836$_{10}$-1
10	4.989708480$_{10}$-1	5.16807$_{10}$-4	3.74216$_{10}$-1	4.82976$_{10}$-1
11	5.002475524$_{10}$-1	1.91138$_{10}$-4	3.69844$_{10}$-1	4.71399$_{10}$-1
15	5.000015863$_{10}$-1	5.80274$_{10}$-6	4.67856$_{10}$-1	4.56356$_{10}$-1
16	5.000085958$_{10}$-1	2.76800$_{10}$-6	4.77016$_{10}$-1	4.57621$_{10}$-1
17	4.999867395$_{10}$-1	4.98160$_{10}$-6	1.79971$_{10}$+0	4.96007$_{10}$-1
18	4.999923552$_{10}$-1	1.35781$_{10}$-5	2.72567$_{10}$+0	5.45253$_{10}$-1
19	4.999645661$_{10}$-1	3.77863$_{10}$-5	2.78287$_{10}$+0	5.94095$_{10}$-1
20	4.998667835$_{10}$-1	1.05920$_{10}$-4	2.80315$_{10}$+0	6.42016$_{10}$-1
27	5.290373141$_{10}$-1	7.10159$_{10}$-2	3.20465$_{10}$+0	9.16477$_{10}$-1
30	2.379020942$_{10}$+0	1.33836$_{10}$+0	1.77625$_{10}$+0	1.01957$_{10}$+0

Table 10.5 OD method $\Upsilon_{\mathrm{OD}}[\Phi_{\mathrm{ILU}}]$ applied to the same problem as in Table 10.3.

m	value in the middle	$\|e^m\|_2$	$\frac{\|e^m\|_2}{\|e^{m-1}\|_2}$	$\sqrt[m]{\frac{\|e^m\|_2}{\|e^0\|_2}}$
1	1.288025563$_{10}$+0	4.58268$_{10}$-1	6.12419$_{10}$-1	6.12419$_{10}$-1
10	5.107964511$_{10}$-1	2.08681$_{10}$-1	9.93821$_{10}$-1	8.80119$_{10}$-1
20	5.084149051$_{10}$-1	1.00523$_{10}$-2	4.81485$_{10}$-1	8.06139$_{10}$-1
30	5.000072697$_{10}$-1	5.10416$_{10}$-6	4.28537$_{10}$-1	6.72659$_{10}$-1
35	4.999124543$_{10}$-1	2.09700$_{10}$-4	2.28682$_{10}$+0	7.91591$_{10}$-1
40	4.178915511$_{10}$-1	5.64009$_{10}$-2	6.53540$_{10}$+0	9.37412$_{10}$-1

Table 10.6 OD method $\Upsilon_{\mathrm{OD}}[\Phi_{\mathrm{ILU}}]$ for the indefinite problem in Table 10.4.

The proof of the following theorem is left to the reader.

Theorem 10.32. *Assume that $N > 0$ and $A = A^{\mathsf{H}}$. Let m_0 be the largest index with $q^m \neq 0$. $q^{m_0+1} = 0$ implies $x^{m_0+1} = x^*$. For all $0 \le m \le m_0$, (10.38a–c) hold:*

$$\begin{aligned} &\langle Nq^k, q^\ell \rangle = 0 \quad \textit{for } 0 \le k \neq \ell \le m_0, \\ &\langle Nq^k, q^k \rangle \neq 0 \quad \textit{for } 0 \le k \le m_0, \end{aligned} \tag{10.38a}$$

$$r^m \perp N\mathcal{K}_m(AN, r^0), \tag{10.38b}$$

$$\operatorname{span}\{q^0, \ldots, q^{m-1}\} = AN\mathcal{K}_m(AN, r^0) = \mathcal{K}_m(AN, ANr^0). \tag{10.38c}$$

x^m is the minimiser $\min\{\|N^{-1/2}(x - x^*)\| : x \in x^0 + N\mathcal{K}_m(AN, r^0)\}$.

This case corresponds to the choice of the spaces in §10.1.5.3. The connection with the Lanczos method is described by Paige–Saunders [307]. The method SYMMLQ defined there is a further stabilisation of the method (10.37).

A review of the algorithms discussed above and of additional variants is given by Stoer [355].

10.5 Solution of Nonsymmetric Systems

Some of the methods described above do not require the assumption (9.1) of positive definiteness of A and are also applicable to indefinite but still symmetric matrices. The nonsymmetric situation is more difficult.

10.5.1 Generalised Minimal Residual Method (GMRES)

The following method generalises the minimal residual iteration described in §9.4 and corresponds to the approach in §10.1.5.2.

10.5.1.1 General Setting and Convergence

The 'generalised minimal residual method' described by Saad–Schultz [329] (see also Walker [388]) determines the vector in the affine space $x^0 + \mathcal{K}_m(A, r^0)$ minimising the residual:

$$x^m = \operatorname{argmin} \left\{ \|b - Ax\|_2 : x \in x^0 + \mathcal{K}_m(A, r^0) \right\}. \tag{10.39}$$

We recall that the control of the residual might be questionable (cf. Remark 2.35).

As the CG method, GMRES (with exact arithmetic) yields the true solution after at least $\#I$ steps.

Proposition 10.33. *For regular A and $m_0 := \deg_A(e^0) = \deg_A(r^0) \le \#I$, the iterate x^{m_0} is the exact solution x^*.*

Proof. For regular A, the statements $p_{m_0}(A)e^0 = 0$ and $p_{m_0}(A)r^0 = -Ap_{m_0}(A)e^0 = 0$ are equivalent. Let $p_{m_0} \in \mathcal{P}_{m_0}$ be the polynomial with $p_{m_0}(A)e^0 = 0$. Note that $p_{m_0}(0) \neq 0$ by Lemma 8.12. After a suitable scaling, $p_{m_0}(0) = 1$ holds so that $p_{m_0}(\xi) = 1 - \xi q_{m_0-1}(\xi)$. The correction $q_{m_0-1}(A)r^0 \in \mathcal{K}_{m_0}(A, r^0)$ yields

$$x^{m_0} - A^{-1}b = e^0 + q_{m_0-1}(A)r^0 = (I - Aq_{m_0-1}(A))e^0 = p_{m_0}(A)e^0 = 0,$$

i.e., x^{m_0} is the exact solution. □

In the case of a general matrix A, one cannot expect other convergence statements than $x^{m_0} = A^{-1}b$, as the following example shows.

Example 10.34. Define $A \in \mathbb{R}^{n\times n}$ by the entries $A_{ij} = \left\{ \begin{matrix} 1 & j - i = 1 \mod n \\ 0 & \text{otherwise} \end{matrix} \right\}$ (e.g., $A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$ for $n = 3$). Then there are initial values x^0 so that the equality $\|r^m\|_2 = \|r^0\|_2$ holds for all residuals $r^m = b - Ax^m$ with $m < n$.

Proof. Choose x^0 such that $r^0 = [1\ 0\ \dots\ 0]^\mathsf{T}$ is the first unit vector. The solution x^m of (10.39) is of the form $x^m = x^0 + q_{m-1}(A)r^0$. The corresponding residual is $r^m = r^0 - q_{m-1}(A)Ar^0 = p_m(A)r^0$ with the polynomial $p_m(\xi) := 1 - \xi q_{m-1}(\xi)$, i.e., $p_m(\xi) = \sum_{\nu=0}^{m} a_\nu \xi^\nu$ with $a_0 = 1$. Note that the optimal polynomial p_m minimises $\|r^m\|_2$. One checks that the product $A^\nu r^0$ is the μ-th unit vector with $\mu = n + 1 - \nu \bmod n$. Hence $r^m = p_m(A)r^0 = [a_0\ a_{n-1}\ \dots\ a_2\ a_1]^\mathsf{T}$ yields the squared norm $\|r^m\|_2^2 = \sum_{\nu=0}^{m} |a_\nu|^2$. The minimum is achieved for $a_1 = a_2 = \ldots = a_{n-1} = 0$ resulting in $\|r^m\|_2 = 1 = \|r^0\|_2$. □

Better results can be obtained if A is Hermitian: $A = A^\mathsf{H}$. However, in this case, the cheaper method of conjugate residuals can be applied, which yields the same iterates (set $N = I$ in Lemma 10.27).

In the case of $A + A^\mathsf{H} > 0$, the convergence can be derived from the convergence of the minimal residual iteration (cf. §9.4).

Proposition 10.35. *Assume $A + A^\mathsf{H} > 0$. Then the residuals of GMRES satisfy $\|r^m\|_2 \le c^m \|r^0\|_2$ with c in (9.26).*

Proof. By construction, $r^m = p_m(A)\, r^0$ holds with a polynomial $p_m \in \mathcal{P}_m$ with $p_m(0) = 1$. The minimal residual iteration yields the sequence $(\hat{x}^k)$ with residuals $\hat{r}^k$. Assume $\hat{r}^0 = r^0$. There are polynomials $q_k \in \mathcal{P}_1$ with $q_k(0) = 1$ and $\hat{r}^k = q_k(A)\hat{r}^{k-1}$. The product $\hat{p}_m(\xi) := \prod_{k=1}^{m} q_k(\xi)$ satisfies $\hat{r}^m = \hat{p}_m(A)r^0$, $\hat{p}_m \in \mathcal{P}_m$, and $\hat{p}_m(0) = 1$. The optimality of the GMRES algorithm yields $\|r^m\|_2 = \min\{\|\rho_m(A)r^0\|_2 : \rho_m \in \mathcal{P}_m, \rho_m(0) = 1\} \le \|\hat{r}^m\|_2 \le c^m \|r^0\|_2$. □

10.5.1.2 Arnoldi Basis

Let $\{v^1, \dots, v^m\}$ be any basis of $\mathcal{K}_m(A, r^0)$ (this is possible if and only if $m \le \deg_A(r^0)$). According to (10.8), the minimiser x^m and its residual r^m are characterised by $r^m \perp A\mathcal{K}_m(A, r^0)$. The ansatz $x^m = x^0 + \sum_{\nu=1}^{m} \alpha_\nu v^\nu$ yields

$$r^m = b - Ax^m = r^0 - \sum_{\nu=1}^{m} \alpha_\nu A v^\nu,$$

and $r^m \perp A\mathcal{K}_m(A, r^0)$ produces the m equations

$$0 = \langle r^m, Av^\mu \rangle = \langle r^0, Av^\mu \rangle - \sum_{\nu=1}^{m} \alpha_\nu \langle Av^\nu, Av^\mu \rangle \qquad (1 \le \mu \le m) \quad (10.40)$$

for the m unknown factors α_ν.

Lemma 10.36. *For regular $A \in \mathbb{K}^{I \times I}$, the matrix $G_m := (\langle Av^\nu, Av^\mu \rangle)_{1 \le \nu, \mu \le m}$ is regular for all $m \le \deg_A(r^0)$ so that the system (10.40) is uniquely solvable.*

Proof. Since A is regular, $\{Av^1, \dots, Av^m\}$ is also a basis of $A\mathcal{K}_m(A, r^0)$. Hence, the Gram matrix G_m is regular. □

For the actual computation, the basis should be suitably chosen. One strategy is to arrange the vectors v^k such that $\mathcal{K}_m(A, r^0) = \operatorname{span}\{v^1, \ldots, v^m\}$ for all $m \le \deg_A(r^0)$; i.e., $\mathcal{K}_{m+1}(A, r^0) = \operatorname{span}\{\mathcal{K}_m(A, r^0), v^m\}$. For the purpose of stability, the basis should be orthonormal. Finally, the basis should be such that the involved computational work is as small as possible.

Instead of the orthonormalisation procedure in Remark A.26a, we use the *Arnoldi algorithm*:

$w^0 := r^0;\ h_{0,-1} := \|r^0\|_2;\ m := 0;$
while $h_{m,m-1} \neq 0$ **do**
begin $v^m := w^m / h_{m,m-1};$
  for $i := 1$ **to** m **do** $h_{im} := \langle Av^m, v^i \rangle;$
  $w^{m+1} := Av^m - \sum_{i=1}^{m} h_{im} v^i;\ h_{m+1,m} := \|w^m\|_2;$
  $m := m + 1$
end;

One easily checks that $\langle v^m, v^i \rangle = \delta_{mi}$; i.e., $(v^i)_{1 \le i \le m}$ is an orthonormal basis of $\mathcal{K}_m(A, r^0)$. The construction implies the property

$$Av^m = \sum_{i=1}^{m+1} h_{im} v^i. \tag{10.41}$$

Therefore, $\langle Av^k, v^i \rangle = h_{ik}$ holds, where we define $h_{ik} := 0$ for $i > k + 1$. We form the matrices

$$V_m = [v^1\ v^2\ \ldots\ v^m] \in \mathbb{K}^{I \times m}, \quad H_m = (h_{ik})_{1 \le i,k \le m} \in \mathbb{K}^{m \times m},$$
$$\hat{H}_{m+1} = (h_{ik})_{1 \le i \le m+1, 1 \le k \le m} \in \mathbb{K}^{(m+1) \times m}.$$

Note that H_m and $\hat{H}_{m+1}$ are Hessenberg matrices, i.e., $h_{ik} = 0$ for $i > k + 1$. From (10.41), we derive

$$V_m^{\mathsf{H}} A V_m = H_m, \qquad V_{m+1}^{\mathsf{H}} A V_m = \hat{H}_{m+1}.$$

The ansatz $x^m \in x^0 + \mathcal{K}_m(A, r^0)$ becomes $x^m = x^0 + V_m z^m$ for a vector $z^m \in \mathbb{K}^m$ to be determined. The residual is $r^m = r^0 - AV_m z^m$. Note that $r^0 = \|r^0\|_2\, v^1 = \|r^0\|_2\, V_{m+1}\, \mathbf{e}^1$ ($\mathbf{e}^1$ is the first unit vector). Since V_{m+1} is an orthogonal matrix, $V_{m+1} V_{m+1}^{\mathsf{H}}$ is the orthogonal projection onto $\mathcal{K}_{m+1}(A, r^0)$. Therefore, $\operatorname{range}(AV_m) \subset \mathcal{K}_{m+1}(A, r^0)$ implies that

$$AV_m = (V_{m+1} V_{m+1}^{\mathsf{H}})(AV_m) = V_{m+1} \hat{H}_{m+1}.$$

Together we obtain

$$r^m = V_{m+1} \left[\|r^0\|_2\, \mathbf{e}^1 + \hat{H}_{m+1}\, z^m \right].$$

Exploiting again the orthogonality of V_{m+1}, we conclude that

$$\|r^m\|_2 = \left\| \left[\|r^0\|_2 \, \mathbf{e}^1 + \hat{H}_{m+1} z^m \right] \right\|_2$$

has to be minimised over all $z^m \in \mathbb{K}^m$ (cf. Exercise B.22). This is a least-squares problem as considered in Remark B.23: apply the QR decomposition: $\hat{H}_{m+1} = QR$ and solve $Rz^m = -\|r^0\|_2 Q^{\mathsf{H}} \mathbf{e}^1$. Because of the Hessenberg form of $\hat{H}_{m+1}$, the QR decomposition is rather cheap (m Givens rotations have to be applied).

Remark 10.37 (cost). The cost of the m-th GMRES step is $\mathcal{O}(m \, \#I)$, so that m steps yield a total amount of $\mathcal{O}(m^2 \, \#I)$ operations. The storage cost is $\mathcal{O}(m \, \#I)$.

The reason is that the involved matrix $\hat{H}_{m+1}$ has Hessenberg structure instead of a tridiagonal one. The existence of short recursions as in the classical CG method is connected with the B-normality of A as discussed in Liesen–Saylor [264].

In the case of a Hermitian matrix $A = A^{\mathsf{H}}$, the Hessenberg structure becomes a tridiagonal one and short recursions can be applied. The resulting method is called MINRES (cf. van der Vorst [373, §6.4]).

10.5.1.3 GMRES(m)

The increasing cost mentioned above is the reason for introducing a restart after a fixed number of m steps. After reaching the GMRES iterate x^m, this value is used as the new starting value for the next m GMRES steps. The size of m may be determined by the maximal available storage $S_{\max}$: $\mathcal{O}(m \, \#I) \le S_{\max}$.

Since already for GMRES no convergence statement for $m < \deg_A(e^0)$ could be given in the general case, the situation is even worse for GMRES(m). In this case, not even $x^m = A^{-1}b$ for $m = n$ can be expected. An alternative approach is to restrict the orthogonalisation to the last m directions.

10.5.2 Full Orthogonalisation Method (FOM)

The full orthogonalisation method or Arnoldi method tries to determine $x^m \in x^0 + \mathcal{K}_m(A, r^0)$ such that

$$r^m \perp \mathcal{K}_m(A, r^0)$$

(we recall that $r^m \perp A\mathcal{K}_m(A, r^0)$ holds for GMRES).

In the general case, the method can break down without obtaining the exact solution. For instance, $r^0 \neq 0$ with $\langle Ar^0, r^0 \rangle = 0$ yields a breakdown since $x^1 = x^0 + \alpha r^0$ leads to a zero division in $\alpha = \|r^0\|_2^2 \,/\, \langle Ar^0, r^0 \rangle$.

If the method can be performed successfully, $x^{m_0} = A^{-1}b$ holds for the index $m_0 = \deg_A(r^0)$. For a proof, use that $r^{m_0} \in \mathcal{K}_{m_0+1}(A, r^0) = \mathcal{K}_{m_0}(A, r^0)$ can be perpendicular to $\mathcal{K}_{m_0}(A, r^0)$ only if $r^{m_0} = 0$.

10.5.3 Biconjugate Gradient Method and Variants

The biconjugate gradient method (abbreviated as BCG or BiCG) uses two different Krylov subspaces $\mathcal{K}_m(A, r^0)$ and $\mathcal{K}_m(A^{\mathsf{H}}, r_*^0)$. Here r_*^0 is any vector with $\langle r^0, r_*^0 \rangle \neq 0$. As the original conjugate gradient method, it uses a short recursion for the search directions $p^m \in \mathcal{K}_{m+1}(A, r^0)$ and $p_*^m \in \mathcal{K}_{m+1}(A^{\mathsf{H}}, r_*^0)$. As a result the residuals are *biconjugate*: $\langle r^i, r_*^j \rangle = 0$ for $i \neq j$, while $\langle Ap^i, p_*^j \rangle = 0$ for $i \neq j$. The formulation of the method goes back to Lanczos [255] and Fletcher [136]. This method does not aim at the minimisation of the error in some norm.

The use of A^{H} in the algorithm may lead to problems since sometimes only a subroutine for $x \mapsto Ax$ is available. On the other hand, all vectors $v \in \mathcal{K}_m(A^{\mathsf{H}}, r_*^0)$ have the representation $p_m(A^{\mathsf{H}})\, r_*^0$ with some polynomial $p_m \in \mathcal{P}_m$. The arising scalar products $\langle v, x \rangle$ with $x = q_m(A) r^0 \in \mathcal{K}_m(A, r^0)$ can be rewritten as $\langle p_m(A^{\mathsf{H}}) r_*^0, x \rangle = \langle r_*^0, p_m(A) q_m(A) r^0 \rangle$. Fortunately, the products $p_m q_m$ are of the form $p_m^2(\xi)$ or $\xi p_m^2(\xi)$. This gives rise to the conjugate gradient squared method CGS by Sonneveld [344] (see also Sonneveld–Wesseling–de Zeeuw [345]).

A stabilised version of CGS called Bi-CGSTAB is developed by van der Vorst [372]. For details, see the original papers or van der Vorst [373, §7], Kanzow [233, §7], Saad [328, §§7.3–7.4], Gutknecht [173, 174, 175], and Bank–Chan [26].

10.5.4 Further Remarks

Since matrices that are not positive definite require more or less involved CG variants, another remedy is worth being considered. As in §5.5, an indefinite or non-symmetric problem can be preconditioned by a positive definite matrix B, so that for solving $B\delta = d$ the standard CG method can be applied as a secondary iteration.

Concus–Golub [98] and Widlund [396] describe an interesting method for general matrices A that are split into their symmetric and skew-symmetric parts: $A = A_0 + A_1,\ A_0 = \frac{1}{2}(A + A^{\mathsf{H}})$. For many applications, A_0 proves to be positive definite. A two-sided transformation by $A^{-1/2}$ yields the matrix $A' := I - S$ with the skew-symmetric term $S := A^{-1/2} A_1 A^{-1/2}$. The eigenvalues of A' lie in a complex interval instead of a real one (cf. Hageman–Young [212, p. 336]. For the respective CG version, one finds an error estimate with respect to the A_0-energy norm, depending on $\Lambda := \|A_0^{-1} A_1\|_2$ and leading to the asymptotic convergence rate $1 - \mathcal{O}(1/\Lambda)$. In the cases of systems arising from partial differential equations, Λ is usually h-independent, leading to a convergence rate independent of the discretisation parameter h. For each step of the algorithm, one system $A_0\delta = d$ must be solved. This fact limits practicability. Under similar assumptions, the multigrid iteration of the second kind even achieves a convergence rate $\mathcal{O}(h^\tau)$ with positive (!) exponent τ (cf. §11.9.1).

Young calls NA *symmetrisable* if there is a similarity transformation such that $WNAW^{-1} > 0$. Then there exist a matrix Z with $ZNA > 0$. The methods called ORTHODIR, ORTHOMIN, and ORTHORES are based on this assumption (cf. Hageman–Young [212, pp. 340–346]).