# 3

# Conjugate Gradients for Unconstrained Minimization

We shall begin our development of scalable algorithms by description of the *conjugate gradient method* for the solution of

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}), \tag{3.1}$$

where $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathsf{A}\mathbf{x} - \mathbf{x}^T\mathbf{b}$, $\mathbf{b}$ is a given column $n$-vector, and $\mathsf{A}$ is an $n \times n$ symmetric positive definite or positive semidefinite matrix. We are interested especially in problems with $n$ large and $\mathsf{A}$ sparse and reasonably conditioned. We have already seen in Sect. 2.2.2 that (3.1) is equivalent to the solution of a system of linear equations $\mathsf{A}\mathbf{x} = \mathbf{b}$, but our main goal here is not to solve large systems of linear equations, but rather to describe our basic tool for dealing with the auxiliary linear systems that are generated by algorithms for the solution of constrained quadratic programming problems.

We shall use the conjugate gradient (CG) method as an *iterative method* which generates improving approximations to the solution at each step. The cost of one step of the CG method is typically dominated by the cost of the multiplication of a vector by the matrix $\mathsf{A}$, which is proportional to the number of nonzero entries of $\mathsf{A}$. The memory requirements are also proportional to the number of nonzero entries of $\mathsf{A}$.

To develop optimal algorithms for more general quadratic programming problems, it is important that the rate of convergence of the conjugate gradient method depends on the distribution of the spectrum of $\mathsf{A}$. In particular, given a positive interval $[a_{\min}, a_{\max}]$ with the spectrum of $\mathsf{A}$, it is possible to give a bound in terms of $a_{\max}/a_{\min}$ on a number of the conjugate gradient iterations that are necessary to solve problem (3.1) to a given relative precision. It is also important that the number of steps that are necessary to obtain an approximate solution of a given problem is typically proportional to the logarithm of prescribed precision, so that the algorithm can return a low-precision solution at a reduced time.

Zdeněk Dostál, *Optimal Quadratic Programming Algorithms*,
Springer Optimization and Its Applications, DOI 10.1007/978-0-387-84806-8_3,

*Overview of Algorithms*

The first algorithm of this chapter is the *method of conjugate directions* defined by the two simple formulae (3.6). The algorithm assumes that we are given an $\mathsf{A}$-orthogonal basis of $\mathbb{R}^n$, leaving open the problem how to get it.

The *conjugate gradient algorithm*, Algorithm 3.1, the main hero of this chapter, combines the conjugate gradient direction method with a clever construction of conjugate directions. It is the best method as it exploits effectively all the information gathered during the solution in order to maximize the decrease of the cost function. The CG method can be considered both as a direct method and an iterative method.

A step of the *restarted conjugate gradient method* described in Sect. 3.4 comprises a fixed number of the conjugate gradient steps. Such algorithm is more robust, but usually less efficient. If the chain of the CG iterations reduces to just one iteration, we get the *gradient method*, known also as the *method of the steepest descent*. It is the most robust and most simple variant of the restarted CG method. See Algorithm 3.2 for a more formal description.

If we are able to find an easily invertible approximation of the Hessian, we can use it to improve the performance of the CG method in the *preconditioned conjugate gradient method* described in Sect. 3.6 as Algorithm 3.3. The construction of preconditioners is problem dependent. The *preconditioning by a conjugate projector* described in Sect. 3.7 as Algorithm 3.4 is useful in the minimization problems arising from the discretization of elliptic partial differential equations and variational inequalities.

## 3.1 Conjugate Directions and Minimization

The conjugate gradient method, an ingenious and powerful engine of our algorithms, is based on simple observations. In this section we examine the first one, namely, that it is possible to reduce the solution of (3.1) to the solution of a sequence of one-dimensional problems.

Let $\mathsf{A} \in \mathbb{R}^{n\times n}$ be an SPD matrix and let us assume that there are nonzero $n$-vectors $\mathbf{p}^1, \dots, \mathbf{p}^n$ such that

$$(\mathbf{p}^i, \mathbf{p}^j)_{\mathsf{A}} = (\mathbf{p}^i)^T \mathsf{A} \mathbf{p}^j = 0 \quad \text{for} \quad i \neq j.$$

We call such vectors $\mathsf{A}$-*conjugate* or briefly *conjugate*. Specializing the arguments of Sect. 1.7, we get that $\mathbf{p}^1, \dots, \mathbf{p}^n$ are independent. Thus $\mathbf{p}^1, \dots, \mathbf{p}^n$ form the basis of $\mathbb{R}^n$ and any $\mathbf{x} \in \mathbb{R}^n$ can be written in the form

$$\mathbf{x} = \xi_1 \mathbf{p}^1 + \cdots + \xi_n \mathbf{p}^n.$$

Substituting into $f$ and using the conjugacy results in

$$\begin{aligned} f(\mathbf{x}) &= \left(\frac{1}{2}\xi_1^2 (\mathbf{p}^1)^T \mathsf{A} \mathbf{p}^1 - \xi_1 \mathbf{b}^T \mathbf{p}^1\right) + \cdots + \left(\frac{1}{2}\xi_n^2 (\mathbf{p}^n)^T \mathsf{A} \mathbf{p}^n - \xi_n \mathbf{b}^T \mathbf{p}^n\right) \\ &= f(\xi_1 \mathbf{p}^1) + \cdots + f(\xi_n \mathbf{p}^n). \end{aligned}$$

Thus

$$f(\widehat{\mathbf{x}}) = \min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}) = \min_{\xi_1\in\mathbb{R}} f(\xi_1\mathbf{p}^1) + \cdots + \min_{\xi_n\in\mathbb{R}} f(\xi_n\mathbf{p}^n).$$

We have thus managed to decompose the original problem (3.1) into $n$ one-dimensional problems. Since

$$\left.\frac{\mathrm{d}f\left(\xi\mathbf{p}^i\right)}{\mathrm{d}\xi}\right|_{\xi_i} = \xi_i(\mathbf{p}^i)^T\mathsf{A}\mathbf{p}^i - \mathbf{b}^T\mathbf{p}^i = 0,$$

the solution $\widehat{\mathbf{x}}$ of (3.1) is given by

$$\widehat{\mathbf{x}} = \xi_1\mathbf{p}^1 + \cdots + \xi_n\mathbf{p}^n, \quad \xi_i = \mathbf{b}^T\mathbf{p}^i/(\mathbf{p}^i)^T\mathsf{A}\mathbf{p}^i,\ i = 1,\ldots,n. \tag{3.2}$$

If the dimension of problem (3.1) is large, the task to evaluate $\widehat{\mathbf{x}}$ may be too ambitious. In this case it may be useful to modify the procedure that we have just described so that it can be used to find an approximation $\widetilde{\mathbf{x}}$ to the solution $\widehat{\mathbf{x}}$ for (3.1) by means of some initial guess $\mathbf{x}^0$ and a few vectors $\mathbf{p}^1,\ldots,\mathbf{p}^k,\ \ k \ll n$. A natural choice for the approximation $\widetilde{\mathbf{x}}$ is the minimizer $\mathbf{x}^k$ of $f$ in $\mathcal{S}^k = \mathbf{x}^0 + \mathrm{Span}\{\mathbf{p}^1,\ldots,\mathbf{p}^k\}$. To find it, notice that any $\mathbf{x}\in\mathcal{S}^k$ can be written in the form

$$\mathbf{x} = \mathbf{x}^0 + \xi_1\mathbf{p}^1 + \cdots + \xi_k\mathbf{p}^k,$$

so, after substituting into $f$ and using that $\mathbf{p}^1,\ldots,\mathbf{p}^k$ are conjugate, we get

$$\begin{aligned} f(\mathbf{x}) = f(\mathbf{x}^0) &+ \left(\frac{1}{2}\xi_1^2(\mathbf{p}^1)^T\mathsf{A}\mathbf{p}^1 + \xi_1\left(\mathsf{A}\mathbf{x}^0 - \mathbf{b}\right)^T\mathbf{p}^1\right) + \ldots \\ &+ \left(\frac{1}{2}\xi_k^2(\mathbf{p}^k)^T\mathsf{A}\mathbf{p}^k + \xi_k\left(\mathsf{A}\mathbf{x}^0 - \mathbf{b}\right)^T\mathbf{p}^k\right). \end{aligned}$$

Denoting $\mathbf{g}^0 = \mathbf{g}(\mathbf{x}^0) = \nabla f(\mathbf{x}^0) = \mathsf{A}\mathbf{x}^0 - \mathbf{b}$ and

$$f_0(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathsf{A}\mathbf{x} + \mathbf{x}^T\mathbf{g}^0,$$

we have

$$f(\mathbf{x}) = f(\mathbf{x}^0) + f_0(\xi_1\mathbf{p}^1) + \cdots + f_0(\xi_k\mathbf{p}^k)$$

and

$$f(\mathbf{x}^k) = \min_{\mathbf{x}\in\mathcal{S}^k} f(\mathbf{x}) = f(\mathbf{x}^0) + \min_{\xi_1\in\mathbb{R}} f_0(\xi_1\mathbf{p}^1) + \cdots + \min_{\xi_k\in\mathbb{R}} f_0(\xi_k\mathbf{p}^k). \tag{3.3}$$

We have thus again reduced our problem to the solution of a sequence of simple one-dimensional problems. The approximation $\mathbf{x}^k$ is given by

$$\mathbf{x}^k = \mathbf{x}^0 + \xi_1\mathbf{p}^1 + \cdots + \xi_k\mathbf{p}^k, \quad \xi_i = -(\mathbf{g}^0)^T\mathbf{p}^i/(\mathbf{p}^i)^T\mathsf{A}\mathbf{p}^i,\ i = 1,\ldots,k, \tag{3.4}$$

as

$$\left.\frac{\mathrm{d}f\left(\xi\mathbf{p}^i\right)}{\mathrm{d}\xi}\right|_{\xi_i} = \xi_i(\mathbf{p}^i)^T\mathsf{A}\mathbf{p}^i + (\mathbf{g}^0)^T\mathbf{p}^i = 0.$$

Since by (3.3) for $k \geq 1$

$$f(\mathbf{x}^k) = \min_{\mathbf{x}\in\mathcal{S}^k} f(\mathbf{x}) = f(\mathbf{x}^{k-1}) + \min_{\xi\in\mathbb{R}} f_0(\xi\mathbf{p}^k), \tag{3.5}$$

we can generate the approximations $\mathbf{x}^k$ iteratively. The *conjugate direction method* starts from an arbitrary initial guess $\mathbf{x}^0$. If $\mathbf{x}^{k-1}$ is given, then $\mathbf{x}^k$ is generated by the formula

$$\mathbf{x}^k = \mathbf{x}^{k-1} - \alpha_k\mathbf{p}^k, \quad \alpha_k = (\mathbf{g}^0)^T\mathbf{p}^i/(\mathbf{p}^i)^T\mathsf{A}\mathbf{p}^i. \tag{3.6}$$

Thus $f(\mathbf{x}^{k-1} + \xi\mathbf{p}^k)$ achieves its minimum at $\xi = -\alpha_k$ and the procedure guarantees that the *successive iterates* $\mathbf{x}^k$ *minimize* $f$ *over a progressively expanding manifold* $\mathcal{S}^k$ *that eventually includes the global minimum of* $f$.

The coefficients $\alpha_k$ can be evaluated by alternative formulae. For example, using Corollary 2.9 and the definition of $\mathcal{S}^k$, we get

$$(\mathbf{g}^k)^T\mathbf{p}^i = 0, \quad i = 1, \ldots, k. \tag{3.7}$$

Since for $i \geq 1$

$$\begin{aligned}\mathbf{g}^i &= \mathsf{A}\mathbf{x}^i - \mathbf{b} = \mathsf{A}\left(\mathbf{x}^{i-1} - \alpha_i\mathbf{p}^i\right) - \mathbf{b} = \left(\mathsf{A}\mathbf{x}^{i-1} - \mathbf{b}\right) - \alpha_i\mathsf{A}\mathbf{p}^i \\ &= \mathbf{g}^{i-1} - \alpha_i\mathsf{A}\mathbf{p}^i,\end{aligned}$$

we get for $k \geq 1$ and $i = 1, \ldots, k-1$, by using the conjugacy, that

$$(\mathbf{g}^i)^T\mathbf{p}^k = (\mathbf{g}^{i-1})^T\mathbf{p}^k - \alpha_i(\mathbf{p}^i)^T\mathsf{A}\mathbf{p}^k = (\mathbf{g}^{i-1})^T\mathbf{p}^k.$$

Thus

$$(\mathbf{g}^0)^T\mathbf{p}^k = (\mathbf{g}^1)^T\mathbf{p}^k = \cdots = (\mathbf{g}^{k-1})^T\mathbf{p}^k$$

and

$$\alpha_k = \frac{(\mathbf{g}^0)^T\mathbf{p}^k}{(\mathbf{p}^k)^T\mathsf{A}\mathbf{p}^k} = \cdots = \frac{(\mathbf{g}^{k-1})^T\mathbf{p}^k}{(\mathbf{p}^k)^T\mathsf{A}\mathbf{p}^k}. \tag{3.8}$$

Combining the latter formula with the Taylor expansion, we get

$$f\left(\mathbf{x}^k\right) = f\left(\mathbf{x}^{k-1}\right) - \frac{1}{2}\frac{\left(\left(\mathbf{g}^{k-1}\right)^T\mathbf{p}^k\right)^2}{\left(\mathbf{p}^k\right)^T\mathsf{A}\mathbf{p}^k}. \tag{3.9}$$

So far, we have not discussed how to get the vectors $\mathbf{p}^1, \ldots, \mathbf{p}^n$. Are we able to generate them efficiently? Positive answer in the next section is a key to the success of the conjugate gradient method.

## 3.2 Generating Conjugate Directions and Krylov Spaces

Let us now recall how to generate conjugate directions with the *Gramm–Schmidt procedure*. Let us first suppose that $\mathbf{p}^1, \dots, \mathbf{p}^k$ are nonzero conjugate directions, $1 \le k < n$, and let us examine how to use $\mathbf{h}^k \notin \mathrm{Span}\{\mathbf{p}^1, \dots, \mathbf{p}^k\}$ to generate a new member $\mathbf{p}^{k+1}$ in the form

$$\mathbf{p}^{k+1} = \mathbf{h}^k + \beta_{k1}\mathbf{p}^1 + \cdots + \beta_{kk}\mathbf{p}^k. \tag{3.10}$$

Since $\mathbf{p}^{k+1}$ should be conjugate to $\mathbf{p}^1, \dots, \mathbf{p}^k$, we get

$$\begin{aligned} 0 = (\mathbf{p}^i)^T \mathsf{A}\mathbf{p}^{k+1} &= (\mathbf{p}^i)^T \mathsf{A}\mathbf{h}^k + \beta_{k1}(\mathbf{p}^i)^T \mathsf{A}\mathbf{p}^1 + \cdots + \beta_{kk}(\mathbf{p}^i)^T \mathsf{A}\mathbf{p}^k \\ &= (\mathbf{p}^i)^T \mathsf{A}\mathbf{h}^k + \beta_{ki}(\mathbf{p}^i)^T \mathsf{A}\mathbf{p}^i, \quad i = 1, \dots, k. \end{aligned}$$

Thus

$$\beta_{ki} = -\frac{(\mathbf{p}^i)^T \mathsf{A}\mathbf{h}^k}{(\mathbf{p}^i)^T \mathsf{A}\mathbf{p}^i}, \ i = 1, \dots, k. \tag{3.11}$$

Obviously

$$\mathrm{Span}\{\mathbf{p}^1, \dots, \mathbf{p}^{k+1}\} = \mathrm{Span}\{\mathbf{p}^1, \dots, \mathbf{p}^k, \mathbf{h}^k\}.$$

Therefore, given any independent vectors $\mathbf{h}^0, \dots, \mathbf{h}^{k-1}$, we can start from $\mathbf{p}^1 = \mathbf{h}^0$ and use (3.10) and (3.11) to construct a set of mutually $\mathsf{A}$-conjugate directions $\mathbf{p}^1, \dots, \mathbf{p}^k$ such that

$$\mathrm{Span}\{\mathbf{h}^0, \dots, \mathbf{h}^{i-1}\} = \mathrm{Span}\{\mathbf{p}^1, \dots, \mathbf{p}^i\}, \ i = 1, \dots, k.$$

For $\mathbf{h}^0, \dots, \mathbf{h}^{k-1}$ arbitrary, the construction is increasingly expensive as it requires both the storage for the vectors $\mathbf{p}^1, \dots, \mathbf{p}^k$ and heavy calculations including evaluation of $k(k+1)/2$ scalar products. However, it turns out that we can adapt the procedure so that it generates very efficiently the conjugate basis of the *Krylov spaces*

$$\mathcal{K}^k = \mathcal{K}^k(\mathsf{A}, \mathbf{g}^0) = \mathrm{Span}\{\mathbf{g}^0, \mathsf{A}\mathbf{g}^0, \dots, \mathsf{A}^{k-1}\mathbf{g}^0\}, \quad k = 1, \dots, n,$$

with $\mathbf{g}^0 = \mathsf{A}\mathbf{x}^0 - \mathbf{b}$ defined by a suitable initial vector $\mathbf{x}^0$ and $\mathcal{K}^0 = \{\mathbf{o}\}$. The powerful method is again based on a few simple observations.

First assume that $\mathbf{p}^1, \dots, \mathbf{p}^i$ form a conjugate basis of $\mathcal{K}^i$, $i = 1, \dots, k$, and observe that if $\mathbf{x}^k$ denotes the minimizer of $f$ on $\mathbf{x}^0 + \mathcal{K}^k$, then by Corollary 2.9 the gradient $\mathbf{g}^k = \nabla f(\mathbf{x}^k)$ is orthogonal to the Krylov space $\mathcal{K}^k$, that is,

$$(\mathbf{g}^k)^T \mathbf{x} = 0 \ \text{ for any } \ \mathbf{x} \in \mathcal{K}^k.$$

In particular, if $\mathbf{g}^k \ne \mathbf{o}$, then

$$\mathbf{g}^k \notin \mathcal{K}^k.$$

Since $\mathbf{g}^k \in \mathcal{K}^{k+1}$, we can use (3.10) with $\mathbf{h}^k = \mathbf{g}^k$ to expand any conjugate basis of $\mathcal{K}^k$ to the conjugate basis of $\mathcal{K}^{k+1}$. Obviously

$$\mathcal{K}^k(\mathsf{A}, \mathbf{g}^0) = \mathrm{Span}\{\mathbf{g}^0, \dots, \mathbf{g}^{k-1}\}.$$

Next observe that for any $\mathbf{x} \in \mathcal{K}^{k-1}$ and $k \geq 1$

$$\mathsf{A}\mathbf{x} \in \mathcal{K}^k,$$

or briefly $\mathsf{A}\mathcal{K}^{k-1} \subseteq \mathcal{K}^k$. Since $\mathbf{p}^i \in \mathcal{K}^i \subseteq \mathcal{K}^{k-1},\ i = 1, \dots, k-1$, we have

$$(\mathsf{A}\mathbf{p}^i)^T \mathbf{g}^k = (\mathbf{p}^i)^T \mathsf{A}\mathbf{g}^k = 0,\ i = 1, \dots, k-1.$$

It follows that

$$\beta_{ki} = -\frac{(\mathbf{p}^i)^T \mathsf{A}\mathbf{g}^k}{(\mathbf{p}^i)^T \mathsf{A}\mathbf{p}^i} = 0,\ i = 1, \dots, k-1.$$

Summing up, if we have a set of such conjugate vectors $\mathbf{p}^1, \dots, \mathbf{p}^k$ that

$$\mathrm{Span}\{\mathbf{p}^1, \dots, \mathbf{p}^i\} = \mathcal{K}^i,\ i = 1, \dots k,$$

then the formula (3.10) applied to $\mathbf{p}^1, \dots, \mathbf{p}^k$ and $\mathbf{h}^k = \mathbf{g}^k$ simplifies to

$$\mathbf{p}^{k+1} = \mathbf{g}^k + \beta_k \mathbf{p}^k \tag{3.12}$$

with

$$\beta_k = \beta_{kk} = -\frac{(\mathbf{p}^k)^T \mathsf{A}\mathbf{g}^k}{(\mathbf{p}^k)^T \mathsf{A}\mathbf{p}^k}. \tag{3.13}$$

Finally, observe that the orthogonality of $\mathbf{g}^k$ to $\mathrm{Span}\{\mathbf{p}^1, \dots, \mathbf{p}^k\}$ and (3.12) imply that

$$\|\mathbf{p}^{k+1}\| \geq \|\mathbf{g}^k\|. \tag{3.14}$$

In particular, if $\mathbf{g}^{k-1} \neq \mathbf{o}$, then $\mathbf{p}^k \neq \mathbf{o}$, so the formula (3.13) is well defined provided $\mathbf{g}^{k-1} \neq \mathbf{o}$.

## 3.3 Conjugate Gradient Method

In the previous two sections, we have found that the conjugate directions can be used to reduce the minimization of any convex quadratic function to the solution of a sequence of one-dimensional problems, and that the conjugate directions can be generated very efficiently. The famous *conjugate gradient (CG) method* just puts these two observations together.

The algorithm starts from an initial guess $\mathbf{x}^0$, $\mathbf{g}^0 = \mathsf{A}\mathbf{x}^0 - \mathbf{b}$, and $\mathbf{p}^1 = \mathbf{g}^0$. If $\mathbf{x}^{k-1}$ and $\mathbf{g}^{k-1}$ are given, $k \geq 1$, it first checks if $\mathbf{x}^{k-1}$ is the solution. If not, then the algorithm generates

$$\mathbf{x}^k = \mathbf{x}^{k-1} - \alpha_k \mathbf{p}^k \ \text{ with } \ \alpha_k = (\mathbf{g}^{k-1})^T \mathbf{p}^k / (\mathbf{p}^k)^T \mathsf{A}\mathbf{p}^k$$

and

$$\begin{aligned}\mathbf{g}^k = \mathsf{A}\mathbf{x}^k - \mathbf{b} &= \mathsf{A}\left(\mathbf{x}^{k-1} - \alpha_k \mathbf{p}^k\right) - \mathbf{b} = \left(\mathsf{A}\mathbf{x}^{k-1} - \mathbf{b}\right) - \alpha_k \mathsf{A}\mathbf{p}^k \\ &= \mathbf{g}^{k-1} - \alpha_k \mathsf{A}\mathbf{p}^k.\end{aligned} \tag{3.15}$$

Finally the new conjugate direction $\mathbf{p}^{k+1}$ is generated by (3.12) and (3.13).

The decision if $\mathbf{x}^{k-1}$ is an acceptable solution is typically based on the value of $\|\mathbf{g}^{k-1}\|$, so the norm of the gradient must be evaluated at each step. It turns out that the norm can also be used to replace the scalar products involving the gradient in the definition of $\alpha_k$ and $\beta_k$. To find the formulae, let us replace $k$ in (3.12) by $k-1$ and multiply the resulting identity by $(\mathbf{g}^{k-1})^T$. Using the orthogonality, we get

$$(\mathbf{g}^{k-1})^T \mathbf{p}^k = \|\mathbf{g}^{k-1}\|^2 + \beta_{k-1}(\mathbf{g}^{k-1})^T \mathbf{p}^{k-1} = \|\mathbf{g}^{k-1}\|^2, \tag{3.16}$$

so by (3.8)

$$\alpha_k = \frac{\|\mathbf{g}^{k-1}\|^2}{(\mathbf{p}^k)^T \mathsf{A}\mathbf{p}^k}. \tag{3.17}$$

To find an alternative formula for $\beta_k$, notice that $\alpha_k > 0$ for $\mathbf{g}^{k-1} \neq \mathbf{o}$ and that by (3.15)

$$\mathsf{A}\mathbf{p}^k = \frac{1}{\alpha_k}(\mathbf{g}^{k-1} - \mathbf{g}^k),$$

so that

$$\alpha_k (\mathbf{g}^k)^T \mathsf{A}\mathbf{p}^k = (\mathbf{g}^k)^T(\mathbf{g}^{k-1} - \mathbf{g}^k) = -\|\mathbf{g}^k\|^2$$

and

$$\beta_k = -\frac{(\mathbf{p}^k)^T \mathsf{A}\mathbf{g}^k}{(\mathbf{p}^k)^T \mathsf{A}\mathbf{p}^k} = \frac{\|\mathbf{g}^k\|^2}{\alpha_k (\mathbf{p}^k)^T \mathsf{A}\mathbf{p}^k} = \frac{\|\mathbf{g}^k\|^2}{\|\mathbf{g}^{k-1}\|^2}. \tag{3.18}$$

The complete CG method is presented as Algorithm 3.1.

**Algorithm 3.1. Conjugate gradient method (CG).**

```
Given a symmetric positive definite matrix $\mathsf{A} \in \mathbb{R}^{n\times n}$ and $\mathbf{b} \in \mathbb{R}^n$.
Step 0. {Initialization.}
        Choose $\mathbf{x}^0 \in \mathbb{R}^n$, set $\mathbf{g}^0 = \mathsf{A}\mathbf{x}^0 - \mathbf{b}$, $\mathbf{p}^1 = \mathbf{g}^0$, $k = 1$
Step 1. {Conjugate gradient loop. }
        while $\|\mathbf{g}^{k-1}\| > 0$
          $\alpha_k = \|\mathbf{g}^{k-1}\|^2/(\mathbf{p}^k)^T \mathsf{A}\mathbf{p}^k$
          $\mathbf{x}^k = \mathbf{x}^{k-1} - \alpha_k \mathbf{p}^k$
          $\mathbf{g}^k = \mathbf{g}^{k-1} - \alpha_k \mathsf{A}\mathbf{p}^k$
          $\beta_k = \|\mathbf{g}^k\|^2/\|\mathbf{g}^{k-1}\|^2 = -(\mathsf{A}\mathbf{p}^k)^T \mathbf{g}^k/\left((\mathbf{p}^k)^T \mathsf{A}\mathbf{p}^k\right)$
          $\mathbf{p}^{k+1} = \mathbf{g}^k + \beta_k \mathbf{p}^k$
          $k = k + 1$
        end while
Step 2. {Return the solution.}
          $\widehat{\mathbf{x}} = \mathbf{x}^k$
```

Each step of the CG method can be implemented with just one matrix–vector multiplication. This multiplication by the Hessian matrix $\mathsf{A}$ typically dominates the cost of the step. Only one generation of vectors $\mathbf{x}^k, \mathbf{p}^k$, and $\mathbf{g}^k$ is typically stored, so the memory requirements are modest.

Let us recall that the algorithm finds at each step the minimizer $\mathbf{x}^k$ of $f$ on $\mathbf{x}^0 + \mathcal{K}^k = \mathbf{x}^0 + \mathcal{K}^k(\mathsf{A}, \mathbf{g}^0)$ and expands the conjugate basis of $\mathcal{K}^k$ to that of $\mathcal{K}^{k+1}$ provided $\mathbf{g}^k \neq \mathbf{o}$. Since the dimension of $\mathcal{K}^k$ is less than or equal to $k$, it follows that for some $k \leq n$

$$\mathcal{K}^k = \mathcal{K}^{k+1}.$$

Since $\mathbf{g}^k \in \mathcal{K}^{k+1}$ and $\mathbf{g}^k$ is orthogonal to $\mathcal{K}^k$, Algorithm 3.1 implemented in the exact arithmetics finds the solution $\widehat{\mathbf{x}}$ of (3.1) in at most $n$ steps. We can sum up the most important properties of Algorithm 3.1 into the following theorem.

**Theorem 3.1.** *Let $\{\mathbf{x}^k\}$ be generated by Algorithm 3.1 to find the solution $\widehat{\mathbf{x}}$ of (3.1) starting from $\mathbf{x}^0 \in \mathbb{R}^n$. Then the algorithm is well defined and there is $k \leq n$ such that $\mathbf{x}^k = \widehat{\mathbf{x}}$. Moreover, the following statements hold for $i = 1, \dots, k$:*

*(i)* $f(\mathbf{x}^i) = \min\{f(\mathbf{x}) : \ \mathbf{x} \in \mathbf{x}^0 + \mathcal{K}^i(\mathsf{A}, \mathbf{g}^0)\}$.
*(ii)* $\|\mathbf{p}^{i+1}\| \geq \|\mathbf{g}^i\|$.
*(iii)* $(\mathbf{g}^i)^T\mathbf{g}^j = 0$ *for* $i \neq j$.
*(iv)* $(\mathbf{p}^i)^T\mathsf{A}\mathbf{p}^j = 0$ *for* $i \neq j$.
*(v)* $\mathcal{K}^i(\mathsf{A}, \mathbf{g}^0) = \mathrm{Span}\{\mathbf{g}^0, \dots, \mathbf{g}^{i-1}\} = \mathrm{Span}\{\mathbf{p}^1, \dots, \mathbf{p}^i\}$.

It is usually sufficient to find $\mathbf{x}^k$ such that $\|\mathbf{g}^k\|$ is small. For example, given a small $\varepsilon > 0$, we can consider $\mathbf{g}^k$ small if

$$\|\mathbf{g}^k\| \leq \varepsilon\|\mathbf{b}\|.$$

Then $\widetilde{\mathbf{x}} = \mathbf{x}^k$ is an approximate solution which satisfies

$$\|\mathsf{A}(\widetilde{\mathbf{x}} - \widehat{\mathbf{x}})\| \leq \varepsilon\|\mathbf{b}\|, \quad \|\widetilde{\mathbf{x}} - \widehat{\mathbf{x}}\| \leq \varepsilon\lambda_{\min}(\mathsf{A})^{-1},$$

where $\lambda_{\min}(\mathsf{A})$ denotes the least eigenvalue of $\mathsf{A}$. It is easy to check that the approximate solution $\widetilde{\mathbf{x}}$ solves the perturbed problem

$$\min_{\mathbf{x}\in\mathbb{R}^n} \widetilde{f}(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathsf{A}\mathbf{x} - \widetilde{\mathbf{b}}^T\mathbf{x}, \quad \widetilde{\mathbf{b}} = \mathbf{b} + \mathbf{g}^k.$$

What is "small" depends on the problem solved. To keep our exposition general, we shall often not specify the test in what follows. Of course $\mathbf{g}^k = \mathbf{o}$ is always considered small.

## 3.4 Restarted CG and the Gradient Method

Given an approximation $\mathbf{x}^0$ of the solution $\widehat{\mathbf{x}}$, we can use $k$ conjugate gradient iterations to find an improved approximation $\mathbf{x}^k$. Repeating the procedure with $\mathbf{x}^0 = \mathbf{x}^k$, we get the *restarted conjugate gradient method.*

A special case with $k = 1$ and $\mathbf{p}^1 = \nabla f(\mathbf{x}^0)$ is of independent interest. Given $\mathbf{x}^k$, the *gradient method* (also called the *steepest descent method*) generates $\mathbf{x}^{k+1}$ by

$$\mathbf{x}^{k+1} = \arg\min_{\alpha \in \mathbb{R}} f(\mathbf{x}^k - \alpha \mathbf{g}^k), \quad \mathbf{g}^k = \nabla f(\mathbf{x}^k).$$

The name "steepest descent" is derived from observation that the linear model of $f$ at $\mathbf{x}$ achieves its minimum on the set of all unit vectors

$$\mathcal{U} = \{\mathbf{d} \in \mathbb{R}^n, \|\mathbf{d}\| = 1\}$$

at $\widehat{\mathbf{d}} = -\|\nabla f(\mathbf{x})\|^{-1} \nabla f(\mathbf{x})$. Indeed, for any $\mathbf{d} \in \mathcal{U}$

$$\nabla f(\mathbf{x})^T \mathbf{d} \geq -\|\nabla f(\mathbf{x})\| \|\mathbf{d}\| = -\|\nabla f(\mathbf{x})\| = \nabla f(\mathbf{x})^T \widehat{\mathbf{d}}.$$

The complete steepest descent method reads as follows:

**Algorithm 3.2. Gradient (steepest descent) method.**

```
Given a symmetric positive definite matrix $\mathsf{A} \in \mathbb{R}^{n \times n}$ and $\mathbf{b} \in \mathbb{R}^n$.
Step 0. {Initialization.}
            Choose $\mathbf{x}^0 \in \mathbb{R}^n$, set $\mathbf{g}^0 = \mathsf{A}\mathbf{x}^0 - \mathbf{b}$, $k = 0$
Step 1. {Steepest descent loop. }
            while $\|\mathbf{g}^k\|$ is not small
              $\alpha_k = \|\mathbf{g}^k\|^2 / (\mathbf{g}^k)^T \mathsf{A}\mathbf{g}^k$
              $\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k \mathbf{g}^k$
              $\mathbf{g}^{k+1} = \mathbf{g}^k - \alpha_k \mathsf{A}\mathbf{g}^k$
              $k = k + 1$
            end while
Step 2. {Return a (possibly approximate) solution.}
              $\widetilde{\mathbf{x}} = \mathbf{x}^k$
```

The gradient method is known to converge, but its convergence is for ill-conditioned problems considerably slower than that of the conjugate gradient method, as we shall see in the next section. The slow convergence is illustrated in Fig. 3.1.

In spite of its slow convergence, the gradient method is useful as it is easy to implement and uses a robust decrease direction. It is illustrated in Fig. 3.2 that even if $\partial \mathbf{g}$ is a relatively large perturbation of the gradient $\mathbf{g}$, the vector $-\mathbf{g} - \partial \mathbf{g}$ is still a decrease direction, while a small perturbation $\partial \mathbf{p}$ of the CG direction $\mathbf{p}$ can cause that $-\mathbf{p} - \partial \mathbf{p}$ is not a decrease direction.

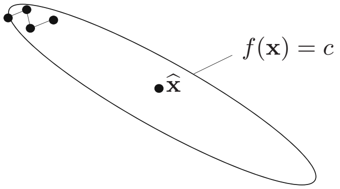

**Fig. 3.1.** Slow convergence of the steepest descent method

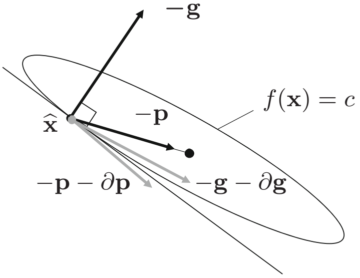
$-\mathbf{g}$
$f(\mathbf{x}) = c$
$-\mathbf{p}$
$\widehat{\mathbf{x}}$
$-\mathbf{p} - \partial\mathbf{p}$
$-\mathbf{g} - \partial\mathbf{g}$

**Fig. 3.2.** Robustness of the gradient and CG decrease directions $\mathbf{g}$ and $\mathbf{p}$

## 3.5 Rate of Convergence and Optimality

Although the conjugate gradient method finds by Theorem 3.1 the exact solution $\widehat{\mathbf{x}}$ of (3.1) in a number of steps which does not exceed the dimension of the problem, it turns out that it can often produce a sufficiently accurate approximation $\widetilde{\mathbf{x}}$ of $\widehat{\mathbf{x}}$ in a much smaller number of steps than required for exact termination. This observation suggests that the conjugate gradient method may also be considered as an iterative method. In this section we present the results which substantiate this claim and help us to identify the favorable cases.

### 3.5.1 Min-max Estimate

Let us denote the *solution error* as

$$\mathbf{e} = \mathbf{e}(\mathbf{x}) = \mathbf{x} - \widehat{\mathbf{x}}$$

and observe that

$$\mathbf{g}(\widehat{\mathbf{x}}) = \mathsf{A}\widehat{\mathbf{x}} - \mathbf{b} = \mathbf{o}.$$

It follows that

$$\mathbf{g}^k = \mathsf{A}\mathbf{x}^k - \mathbf{b} = \mathsf{A}\mathbf{x}^k - \mathsf{A}\widehat{\mathbf{x}} = \mathsf{A}(\mathbf{x}^k - \widehat{\mathbf{x}}) = \mathsf{A}\mathbf{e}^k,$$

so in particular

$$\mathcal{K}^k(\mathsf{A}, \mathbf{g}^0) = \mathrm{Span}\{\mathbf{g}^0, \mathsf{A}\mathbf{g}^0, \ldots, \mathsf{A}^{k-1}\mathbf{g}^0\} = \mathrm{Span}\{\mathsf{A}\mathbf{e}^0, \ldots, \mathsf{A}^k\mathbf{e}^0\}.$$

We start our analysis of the solution error by using the Taylor expansion (2.5) to obtain the identity

$$\begin{aligned} f(\mathbf{x}) - f(\widehat{\mathbf{x}}) &= f(\widehat{\mathbf{x}} + (\mathbf{x} - \widehat{\mathbf{x}})) - f(\widehat{\mathbf{x}}) \\ &= f(\widehat{\mathbf{x}}) + \mathbf{g}(\widehat{\mathbf{x}})^T(\mathbf{x} - \widehat{\mathbf{x}}) + \frac{1}{2}\|\mathbf{x} - \widehat{\mathbf{x}}\|_{\mathsf{A}}^2 - f(\widehat{\mathbf{x}}) \\ &= \frac{1}{2}\|\mathbf{x} - \widehat{\mathbf{x}}\|_{\mathsf{A}}^2 = \frac{1}{2}\|\mathbf{e}\|_{\mathsf{A}}^2. \end{aligned}$$

Combining the latter identity with Theorem 3.1, we get

$$\begin{aligned} \|\mathbf{e}^k\|_A^2 = 2\left(f(\mathbf{x}^k) - f(\widehat{\mathbf{x}})\right) &= \min_{\mathbf{x} \in \mathbf{x}^0 + \mathcal{K}^k(\mathsf{A}, \mathbf{g}^0)} 2\left(f(\mathbf{x}) - f(\widehat{\mathbf{x}})\right) \\ &= \min_{\mathbf{x} \in \mathbf{x}^0 + \mathcal{K}^k(\mathsf{A}, \mathbf{g}^0)} \|\mathbf{x} - \widehat{\mathbf{x}}\|_{\mathsf{A}}^2 = \min_{\mathbf{x} \in \mathbf{x}^0 + \mathcal{K}^k(\mathsf{A}, \mathbf{g}^0)} \|\mathbf{e}(\mathbf{x})\|_{\mathsf{A}}^2. \end{aligned}$$

Since any $\mathbf{x} \in \mathbf{x}^0 + \mathcal{K}^k(\mathsf{A}, \mathbf{g}^0)$ may be written in the form

$$\mathbf{x} = \mathbf{x}^0 + \xi_1\mathbf{g}^0 + \xi_2\mathsf{A}\mathbf{g}^0 + \cdots + \xi_k\mathsf{A}^{k-1}\mathbf{g}^0 = \mathbf{x}^0 + \xi_1\mathsf{A}\mathbf{e}^0 + \cdots + \xi_k\mathsf{A}^k\mathbf{e}^0,$$

it follows that

$$\mathbf{x} - \widehat{\mathbf{x}} = \mathbf{e}^0 + \xi_1\mathsf{A}\mathbf{e}^0 + \cdots + \xi_k\mathsf{A}^k\mathbf{e}^0 = p(\mathsf{A})\mathbf{e}^0,$$

where $p$ denotes the polynomial defined for any $x \in \mathbb{R}$ by

$$p(x) = 1 + \xi_1 x + \xi_2 x^2 + \cdots + \xi_k x^k.$$

Thus denoting by $\mathcal{P}^k$ the set of all $k$th degree polynomials $p$ which satisfy $p(0) = 1$, we have

$$\|\mathbf{e}^k\|_{\mathsf{A}}^2 = \min_{\mathbf{x} \in \mathbf{x}^0 + \mathcal{K}^k(\mathsf{A}, \mathbf{g}^0)} \|\mathbf{e}(\mathbf{x})\|_{\mathsf{A}}^2 = \min_{p \in \mathcal{P}^k} \|p(\mathsf{A})\mathbf{e}^0\|_{\mathsf{A}}^2. \tag{3.19}$$

We shall now derive a bound on the expression on the right-hand side of (3.19) that depends on the spectrum of $\mathsf{A}$, but is independent of the direction of the initial error $\mathbf{e}^0$. Let a spectral decomposition of $\mathsf{A}$ be written as $\mathsf{A} = \mathsf{U}\mathsf{D}\mathsf{U}^T$, where $\mathsf{U}$ is an orthogonal matrix and $\mathsf{D}$ =diag$(\lambda_1, \dots, \lambda_n)$ is a diagonal matrix defined by the eigenvalues of $\mathsf{A}$. Since $\mathsf{A}$ is assumed to be positive definite, the square root of $\mathsf{A}$ is well defined by

$$\mathsf{A}^{\frac{1}{2}} = \mathsf{U}\mathsf{D}^{\frac{1}{2}}\mathsf{U}^T.$$

Using $p(\mathsf{A}) = \mathsf{U}p(\mathsf{D})\mathsf{U}^T$, it is also easy to check that

$$\mathsf{A}^{\frac{1}{2}}p(\mathsf{A}) = p(\mathsf{A})\mathsf{A}^{\frac{1}{2}}.$$

Moreover, for any vector $\mathbf{v} \in \mathbb{R}^n$

$$\|\mathbf{v}\|_{\mathsf{A}}^2 = \mathbf{v}^T\mathsf{A}\mathbf{v} = \mathbf{v}^T\mathsf{A}^{\frac{1}{2}}\mathsf{A}^{\frac{1}{2}}\mathbf{v} = (\mathsf{A}^{\frac{1}{2}}\mathbf{v})^T\mathsf{A}^{\frac{1}{2}}\mathbf{v} = \|\mathsf{A}^{\frac{1}{2}}\mathbf{v}\|^2.$$

Using the latter identities, (3.19), and the properties of norms, we get

$$\begin{aligned}\|\mathbf{e}^k\|_A^2 &= \min_{p\in\mathcal{P}^k} \|p(\mathsf{A})\mathbf{e}^0\|_{\mathsf{A}}^2 = \min_{p\in\mathcal{P}^k} \|\mathsf{A}^{\frac{1}{2}}p(\mathsf{A})\mathbf{e}^0\|^2 = \min_{p\in\mathcal{P}^k} \|p(\mathsf{A})\mathsf{A}^{\frac{1}{2}}\mathbf{e}^0\|^2\\ &\le \min_{p\in\mathcal{P}^k} \|p(\mathsf{A})\|^2\|\mathsf{A}^{\frac{1}{2}}\mathbf{e}^0\|^2 = \min_{p\in\mathcal{P}^k} \|p(\mathsf{D})\|^2\|\mathbf{e}^0\|_{\mathsf{A}}^2.\end{aligned}$$

Since

$$\|p(\mathsf{D})\| = \max_{i\in\{1,\dots,n\}} |p(\lambda_i)|,$$

we can write

$$\|\mathbf{e}^k\|_{\mathsf{A}} \le \min_{p\in\mathcal{P}^k} \max_{i\in\{1,\dots,n\}} |p(\lambda_i)|\ \|\mathbf{e}^0\|_{\mathsf{A}}. \tag{3.20}$$

### 3.5.2 Estimate in the Condition Number

The estimate (3.20) reduces the analysis of convergence of the CG method to the analysis of approximation of the zero function on the spectrum of $\mathsf{A}$ by a $k$th degree polynomial with the value one at origin. This result helps us to identify the favorable cases when the conjugate gradient method is effective. For example, if the spectrum of $\mathsf{A}$ is clustered around a single point $\xi$, then the minimization by the CG should be very effective because $|(1 - x/\xi)^k|$ is small near $\xi$. We shall use (3.20) to get a "global" estimate of the rate of convergence of the CG method in terms of the condition number of $\mathsf{A}$.

**Theorem 3.2.** *Let $\{\mathbf{x}^k\}$ be generated by Algorithm 3.1 to find the solution $\widehat{\mathbf{x}}$ of (3.1) starting from $\mathbf{x}^0 \in \mathbb{R}^n$. Then the error*

$$\mathbf{e}^k = \mathbf{x}^k - \widehat{\mathbf{x}}$$

*satisfies*

$$\|\mathbf{e}^k\|_{\mathsf{A}} \le 2\left(\frac{\sqrt{\kappa(\mathsf{A})}-1}{\sqrt{\kappa(\mathsf{A})}+1}\right)^k \|\mathbf{e}^0\|_{\mathsf{A}}, \tag{3.21}$$

*where $\kappa(\mathsf{A})$ denotes the spectral condition number of $\mathsf{A}$.*

*Proof.* First notice that if $\mathcal{P}^k$ is the set of all $k$th degree polynomials $p$ such that $p(0) = 1$, then for any $t \in \mathcal{P}^k$

$$\min_{p\in\mathcal{P}^k} \max_{\lambda\in[\lambda_{\min},\lambda_{\max}]} |p(\lambda)| \le \max_{\lambda\in[\lambda_{\min},\lambda_{\max}]} |t(\lambda)|. \tag{3.22}$$

A natural choice for $t$ is the $k$th (weighted and shifted) Chebyshev polynomial on the interval $[\lambda_{\min}, \lambda_{\max}]$

$$t_k(\lambda) = T_k\left(\frac{2\lambda - \lambda_{\max} - \lambda_{\min}}{\lambda_{\max} - \lambda_{\min}}\right) / T_k\left(-\frac{\lambda_{\max} + \lambda_{\min}}{\lambda_{\max} - \lambda_{\min}}\right),$$

where $T_k(x)$ is the Chebyshev polynomial of the first kind on the interval $[-1, 1]$ given by

$$T_k(x) = \frac{1}{2}\left(x + \sqrt{x^2 - 1}\right)^k + \frac{1}{2}\left(x - \sqrt{x^2 - 1}\right)^k.$$

This $t_k$ is known to minimize the right-hand side of (3.22) (see, e.g., [172]). Obviously $t_k \in \mathcal{P}^k$, so that we can use its well-known properties to get

$$\max_{\lambda \in [\lambda_{\min}, \lambda_{\max}]} |t_k(\lambda)| = 1/T_k\left(\frac{\lambda_{\max} + \lambda_{\min}}{\lambda_{\max} - \lambda_{\min}}\right).$$

Simple manipulations then show that

$$T_k\left(\frac{\lambda_{\max} + \lambda_{\min}}{\lambda_{\max} - \lambda_{\min}}\right) = \frac{1}{2}\left(\frac{\sqrt{\kappa(\mathsf{A})} + 1}{\sqrt{\kappa(\mathsf{A})} - 1}\right)^k + \frac{1}{2}\left(\frac{\sqrt{\kappa(\mathsf{A})} - 1}{\sqrt{\kappa(\mathsf{A})} + 1}\right)^k.$$

Thus for any $\lambda \in [\lambda_{\min}, \lambda_{\max}]$

$$|p_k(\lambda)| \leq 2\left(\frac{\sqrt{\kappa(\mathsf{A})} - 1}{\sqrt{\kappa(\mathsf{A})} + 1}\right)^k.$$

Substituting this bound into (3.20) then gives the required result. □

The estimate (3.21) can be improved for some special distributions of the eigenvalues. For example, if the spectrum of $\mathsf{A}$ is in a positive interval $[a_{\min}, a_{\max}]$ except for $m$ isolated eigenvalues $\lambda_1, \ldots, \lambda_m$, then we can use special polynomials $p \in \mathcal{P}^{k+m}$ of the form

$$p(\lambda) = \left(1 - \frac{\lambda}{\lambda_1}\right) \ldots \left(1 - \frac{\lambda}{\lambda_m}\right) q(\lambda), \;\; q \in \mathcal{P}^k$$

to get the estimate

$$\|\mathbf{e}^{k+m}\|_{\mathsf{A}} \leq 2\left(\frac{\sqrt{\widetilde{\kappa}} - 1}{\sqrt{\widetilde{\kappa}} + 1}\right)^k \|\mathbf{e}^0\|_{\mathsf{A}}, \tag{3.23}$$

where $\widetilde{\kappa} = a_{\max}/a_{\min}$.

If the spectrum of $\mathsf{A}$ is distributed in two positive intervals $[a_{\min}, a_{\max}]$ and $[a_{\min} + d, a_{\max} + d]$, $d > 0$, then

$$\|\mathbf{e}^k\|_{\mathsf{A}} \leq 2\left(\frac{\sqrt{\overline{\kappa}} - 1}{\sqrt{\overline{\kappa}} + 1}\right)^k \|\mathbf{e}^0\|_{\mathsf{A}}, \tag{3.24}$$

where $\overline{\kappa} = 4a_{\max}/a_{\min}$ approximates the *effective condition number* of a matrix $\mathsf{A}$ with the spectrum in $[a_{\min}, a_{\max}] \cup [a_{\min} + d, a_{\max} + d]$. An interesting feature of the estimates (3.23) and (3.24) is that the *upper bound is independent of the values of some eigenvalues or d.* The proofs of the above and some other interesting estimates can be found in papers by Axelsson [3] and Axelsson and Lindskøg [5].

### 3.5.3 Convergence Rate of the Gradient Method

Observing that the step of the gradient method defined by Algorithm 3.2 is just the first step of the CG algorithm, we can use the results of Sect. 3.5.1 to find the rate of convergence of the gradient method. The estimate is formulated in the following proposition.

**Proposition 3.3.** *Let* $\{\mathbf{x}^k\}$ *be generated by Algorithm 3.2 to find the solution* $\widehat{\mathbf{x}}$ *of (3.1) starting from* $\mathbf{x}^0 \in \mathbb{R}^n$. *Then the error*

$$\mathbf{e}^k = \mathbf{x}^k - \widehat{\mathbf{x}}$$

*satisfies*

$$\|\mathbf{e}^k\|_{\mathsf{A}} \le \left( \frac{\kappa(\mathsf{A}) - 1}{\kappa(\mathsf{A}) + 1} \right)^k \|\mathbf{e}^0\|_{\mathsf{A}}, \tag{3.25}$$

*where* $\kappa(\mathsf{A})$ *denotes the spectral condition number of* $\mathsf{A}$.

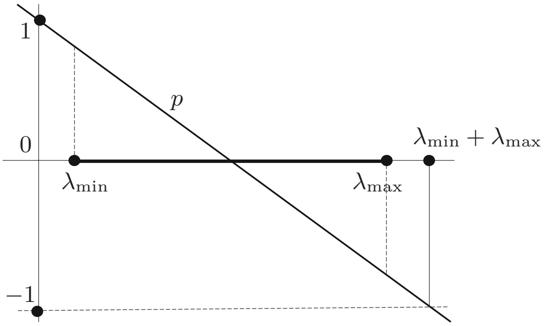

**Fig. 3.3.** The best approximation of zero on $\sigma(\mathsf{A})$ by linear polynomial with $p(0) = 1$

*Proof.* Let $\mathbf{x}^{k+1}$ be generated by the gradient method from $\mathbf{x}^k \in \mathbb{R}^n$ and let $\mathcal{P}^1$ denote the set of all linear polynomials $p$ such that $p(0) = 1$. Then the energy norm $\|\mathbf{e}^k\|_{\mathsf{A}}$ of the error

$$\mathbf{e}^k = \mathbf{x}^k - \widehat{\mathbf{x}}$$

is by (3.20) reduced by a factor which can be estimated from

$$\|\mathbf{e}^{k+1}\|_{\mathsf{A}} \le \min_{p \in \mathcal{P}^1} \max_{i \in \{1,\dots,n\}} |p(\lambda_i)| \; \|\mathbf{e}^k\|_{\mathsf{A}} = \min_{\xi_1 \in \mathbb{R}} \max_{i = \{1,\dots,n\}} |\xi_1 \lambda_i + 1| \; \|\mathbf{e}^k\|_{\mathsf{A}}.$$

Using elementary properties of linear functions or Fig. 3.3, we get that the minimizer $\overline{\xi}_1$ satisfies

$$\overline{\xi}_1\lambda_{\min} + 1 = -(\overline{\xi}_1\lambda_{\max} + 1).$$

It follows that

$$\overline{\xi}_1 = -2/(\lambda_{\min} + \lambda_{\max})$$

and

$$\|\mathbf{e}^{k+1}\|_{\mathsf{A}} \leq \left(\frac{-2\lambda_{\min}}{\lambda_{\min} + \lambda_{\max}} + 1\right)\|\mathbf{e}^k\|_{\mathsf{A}} = \frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}}\|\mathbf{e}^k\|_{\mathsf{A}}. \tag{3.26}$$

The estimate (3.25) can be obtained from (3.26) by simple manipulations. □

Notice that the estimate (3.21) for the first step of the conjugate gradient method may give worse bound than the estimate (3.25) for the gradient method, but for large $k$, the estimate (3.21) for the $k$th step of the conjugate gradient method is much better than the estimate (3.25) for the $k$th step of the gradient method. The reason is that (3.21) captures the global performance of the CG method, in particular its capability to exploit the information from the previous steps, while (3.25) is based on analysis of just one step, in agreement with the one-step information used by the gradient method.

### 3.5.4 Optimality

Theorem 3.2 implies an easy optimality result concerning the number of iterations of the CG algorithm. To formulate it, let $\mathcal{T}$ denote any set of indices and assume that for any $t \in \mathcal{T}$ there is defined the problem

$$\text{minimize} \quad f_t(\mathbf{x})$$

with $f_t(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathsf{A}_t\mathbf{x} - \mathbf{b}_t^T\mathbf{x}$, $\mathsf{A}_t \in \mathbb{R}^{n_t \times n_t}$ symmetric positive definite, and $\mathbf{b}_t, \mathbf{x} \in \mathbb{R}^{n_t}$. Moreover, assume that the eigenvalues of any $\mathsf{A}_t$ are in the interval $[a_{\min}, a_{\max}]$, $0 < a_{\min} \leq a_{\max}$. Then the number of the CG iterations that are necessary to reduce the error by a given factor $\varepsilon$ is uniformly bounded. It easily follows that *the CG algorithm starting from* $\mathbf{x}_t^0 = \mathbf{o}$ *finds* $\mathbf{x}_t^k$ *such that*

$$\|\mathsf{A}_t\mathbf{x}_t^k - \mathbf{b}_t\| \leq \epsilon\|\mathbf{b}_t\|$$

*at* $O(1)$ *iterations.* It follows that if the matrices $\mathsf{A}_t$ have $O(n_t)$ elements, then we can get approximate solutions at the optimal $O(n_t)$ arithmetic operations.

## 3.6 Preconditioned Conjugate Gradients

The analysis of the previous section shows that the rate of convergence of the conjugate gradient algorithm depends on the distribution of the eigenvalues of the Hessian $\mathsf{A}$ of $f$. In particular, we argued that CG converges very rapidly if the eigenvalues of $\mathsf{A}$ are clustered around one point, i.e., if the condition number $\kappa(\mathsf{A})$ is close to one. We shall now show that we can reduce our minimization problem to this favorable case if we have a symmetric positive definite matrix $\mathsf{M}$ such that $\mathsf{M}^{-1}\mathbf{x}$ can be easily evaluated for any $\mathbf{x}$ and $\mathsf{M}$ approximates $\mathsf{A}$ in the sense that $\mathsf{M}^{-1}\mathsf{A}$ is close to the identity.

First assume that $\mathsf{M}$ is available in the form

$$\mathsf{M} = \widetilde{\mathsf{L}}\widetilde{\mathsf{L}}^T,$$

so that $\mathsf{M}^{-1}\mathsf{A}$ is similar to $\widetilde{\mathsf{L}}^{-1}\mathsf{A}\widetilde{\mathsf{L}}^{-T}$ and the latter matrix is close to the identity. Then

$$f(\mathbf{x}) = \frac{1}{2}(\widetilde{\mathsf{L}}^T\mathbf{x})^T(\widetilde{\mathsf{L}}^{-1}\mathsf{A}\widetilde{\mathsf{L}}^{-T})(\widetilde{\mathsf{L}}^T\mathbf{x}) - (\widetilde{\mathsf{L}}^{-1}\mathbf{b})^T(\widetilde{\mathsf{L}}^T\mathbf{x})$$

and we can replace our original problem (3.1) by the *preconditioned problem* to find

$$\min_{\mathbf{y}\in\mathbb{R}^n} \bar{f}(\mathbf{y}), \tag{3.27}$$

where we substituted $\mathbf{y} = \widetilde{\mathsf{L}}^T\mathbf{x}$ and set

$$\bar{f}(\mathbf{y}) = \frac{1}{2}\mathbf{y}^T(\widetilde{\mathsf{L}}^{-1}\mathsf{A}\widetilde{\mathsf{L}}^{-T})\mathbf{y} - (\widetilde{\mathsf{L}}^{-1}\mathbf{b})^T\mathbf{y}.$$

The solution $\widehat{\mathbf{y}}$ of the preconditioned problem (3.27) is related to the solution $\widehat{\mathbf{x}}$ of the original problem by

$$\widehat{\mathbf{x}} = \widetilde{\mathsf{L}}^{-T}\widehat{\mathbf{y}}.$$

If the CG algorithm is applied directly to the preconditioned problem (3.27) with a given $\mathbf{y}^0$, then the algorithm is initialized by

$$\mathbf{y}^0 = \widetilde{\mathsf{L}}^T\mathbf{x}^0, \quad \bar{\mathbf{g}}^0 = \widetilde{\mathsf{L}}^{-1}\mathsf{A}\widetilde{\mathsf{L}}^{-T}\mathbf{y}^0 - \widetilde{\mathsf{L}}^{-1}\mathbf{b} = \widetilde{\mathsf{L}}^{-1}\mathbf{g}^0, \quad \text{and} \quad \bar{\mathbf{p}}^1 = \bar{\mathbf{g}}^0;$$

the iterates are defined by

$$\begin{aligned}
\bar{\alpha}_k &= \|\bar{\mathbf{g}}^{k-1}\|^2/(\bar{\mathbf{p}}^k)^T\widetilde{\mathsf{L}}^{-1}\mathsf{A}\widetilde{\mathsf{L}}^{-T}\bar{\mathbf{p}}^k,\\
\mathbf{y}^k &= \mathbf{y}^{k-1} - \bar{\alpha}_k\bar{\mathbf{p}}^k,\\
\bar{\mathbf{g}}^k &= \bar{\mathbf{g}}^{k-1} - \bar{\alpha}_k\widetilde{\mathsf{L}}^{-1}\mathsf{A}\widetilde{\mathsf{L}}^{-T}\bar{\mathbf{p}}^k,\\
\bar{\beta}_k &= \|\bar{\mathbf{g}}^k\|^2/\|\bar{\mathbf{g}}^{k-1}\|^2,\\
\bar{\mathbf{p}}^{k+1} &= \bar{\mathbf{g}}^k + \bar{\beta}_k\bar{\mathbf{p}}^k.
\end{aligned}$$

Substituting

$$\mathbf{y}^k = \widetilde{\mathsf{L}}^T\mathbf{x}^k, \quad \bar{\mathbf{g}}^k = \widetilde{\mathsf{L}}^{-1}\mathbf{g}^k, \quad \text{and} \quad \bar{\mathbf{p}}^k = \widetilde{\mathsf{L}}^T\mathbf{p}^k,$$

and denoting

$$\mathbf{z}^k = \widetilde{\mathsf{L}}^{-T}\widetilde{\mathsf{L}}^{-1}\mathbf{g}^k = \mathsf{M}^{-1}\mathbf{g}^k,$$

we obtain the *preconditioned conjugate gradient algorithm* (PCG) in the original variables.

**Algorithm 3.3. Preconditioned conjugate gradient method (PCG).**

```
Given a symmetric positive definite matrix $\mathsf{A} \in \mathbb{R}^{n\times n}$, its symmetric positive
definite approximation $\mathsf{M} \in \mathbb{R}^{n\times n}$, and $\mathbf{b} \in \mathbb{R}^n$.
Step 0. {Initialization.}
        Choose $\mathbf{x}^0 \in \mathbb{R}^n$, set $\mathbf{g}^0 = \mathsf{A}\mathbf{x}^0 - \mathbf{b}$, $\mathbf{z}^0 = \mathsf{M}^{-1}\mathbf{g}^0$, $\mathbf{p}^1 = \mathbf{z}^0$, $k = 1$
Step 1. {Conjugate gradient loop.}
        while $\|\mathbf{g}^{k-1}\|$ is not small
          $\alpha_k = (\mathbf{z}^{k-1})^T\mathbf{g}^{k-1}/(\mathbf{p}^k)^T\mathsf{A}\mathbf{p}^k$
          $\mathbf{x}^k = \mathbf{x}^{k-1} - \alpha_k\mathbf{p}^k$
          $\mathbf{g}^k = \mathbf{g}^{k-1} - \alpha_k\mathsf{A}\mathbf{p}^k$
          $\mathbf{z}^k = \mathsf{M}^{-1}\mathbf{g}^k$
          $\beta_k = (\mathbf{z}^k)^T\mathbf{g}^k/(\mathbf{z}^{k-1})^T\mathbf{g}^{k-1}$
          $\mathbf{p}^{k+1} = \mathbf{z}^k + \beta_k\mathbf{p}^k$
          $k = k + 1$
        end while
Step 2. {Return a (possibly approximate) solution.}
          $\widetilde{\mathbf{x}} = \mathbf{x}^k$
```

Notice that the PCG algorithm does not exploit explicitly the Cholesky factorization of the preconditioner $\mathsf{M}$. The *pseudoresiduals* $\mathbf{z}^k$ are typically obtained by solving $\mathsf{M}\mathbf{z}^k = \mathbf{g}^k$. If $\mathsf{M}$ is a good approximation of $\mathsf{A}$, then $\mathbf{z}^k$ is close to the error vector $\mathbf{e}^k$. The rate of convergence of the PCG algorithm depends on the condition number of the Hessian of the transformed function $\bar{f}$, i.e., on $\kappa(\mathsf{M}^{-1}\mathsf{A}) = \kappa(\widetilde{\mathsf{L}}^{-1}\mathsf{A}\widetilde{\mathsf{L}}^{-T})$. Thus the efficiency of the preconditioned conjugate gradient method depends critically on the choice of a preconditioner, which should balance the cost of its application with the preconditioning effect. We refer interested readers to specialized books like Saad [163] or Axelsson [4] for more information. Since the choice of the preconditioner is problem dependent, we limit our attention here to the brief discussion of a few general strategies.

The most simple preconditioners may be defined by means of the decomposition

$$\mathsf{A} = \mathsf{D} + \mathsf{E} + \mathsf{E}^T,$$

where $\mathsf{D}$ is the diagonal of $\mathsf{A}$ and $\mathsf{E}$ is its strict lower part with the entries $[\mathsf{E}]_{ij} = [\mathsf{A}]_{ij}$ for $i > j$ and $[\mathsf{E}]_{ij} = 0$ otherwise.

The *Jacobi preconditioner* $\mathsf{M}_J = \mathsf{D}$ is the easiest one to implement, but its efficiency is very limited. Better approximation of $\mathsf{A}$ can be achieved by choosing the *block diagonal Jacobi preconditioner*

$$\mathsf{M}_{BJ} = \begin{bmatrix} \mathsf{A}_{11} & \mathsf{O} & \dots & \mathsf{O} \\ \mathsf{O} & \mathsf{A}_{22} & \dots & \mathsf{O} \\ . & . & \dots & . \\ \mathsf{O} & \mathsf{O} & \dots & \mathsf{A}_{kk} \end{bmatrix}, \tag{3.28}$$

where $\mathsf{A}_{ii}$ are diagonal blocks of $\mathsf{A}$ (see, e.g., Greenbaum [106, Sect. 10.5]). The *pseudoresiduals* $\mathbf{z}^k$ are typically obtained by solving $\mathsf{A}_{ii}\mathbf{z}_i^k = \mathbf{g}_i^k$.

Good results may be often achieved with the *symmetric Gauss-Seidel* preconditioner

$$\mathsf{M}_{SGS} = (\mathsf{D}+\mathsf{E})\mathsf{D}^{-1}(\mathsf{D}+\mathsf{E}^T).$$

Notice that

$$\widetilde{\mathsf{L}} = (\mathsf{D}+\mathsf{E})\mathsf{D}^{-\frac{1}{2}}$$

is a regular lower triangular matrix, so we have the triangular factorization

$$\mathsf{M}_{SGS} = \widetilde{\mathsf{L}}\widetilde{\mathsf{L}}^T$$

for free.

More generally, the factorized preconditioners can be produced by *incomplete Cholesky (IC)* which neglects some fill-in elements in the factor $\mathsf{L}$. When the elements of $\mathsf{L}$ are neglected because they are smaller than a certain threshold, the factorization is called "IC-by-value", and when they are omitted because they do not belong to a certain sparsity pattern, we have "IC-by-position". See for example Axelsson [4] or Saad [163]. The drawback of this method is that it can fail on the generation of diagonal entries.

## 3.7 Preconditioning by Conjugate Projector

So far we have assumed that the preconditioners to a symmetric positive definite matrix $\mathsf{A}$ are nonsingular matrices that approximate $\mathsf{A}$. In this section we describe an alternative strategy which is useful when we are able to find the minimizer $\mathbf{x}^0$ of $f$ over a subspace $\mathcal{U}$ of $\mathbb{R}^n$. We shall show that in this case we can get the preconditioning effect by reducing the conjugate gradient iterations to the conjugate complement of $\mathcal{U}$.

### 3.7.1 Conjugate Projectors

Our main tools will be the projectors with conjugate range and kernel. We shall use the basic relations introduced in Sect. 1.3 and some observations that we review in this subsection.

Let $\mathsf{A} \in \mathbb{R}^{n\times n}$ be a symmetric positive definite matrix. A projector $\mathsf{P}$ is an *A-conjugate projector* or briefly a *conjugate projector* if Im$\mathsf{P}$ is $\mathsf{A}$-conjugate to Ker$\mathsf{P}$, or equivalently

$$\mathsf{P}^T\mathsf{A}(\mathsf{I}-\mathsf{P}) = \mathsf{P}^T\mathsf{A} - \mathsf{P}^T\mathsf{A}\mathsf{P} = \mathsf{O}.$$

It follows that $\mathsf{Q} = \mathsf{I}-\mathsf{P}$ is also a conjugate projector,

$$\mathsf{P}^T\mathsf{A} = \mathsf{A}\mathsf{P} = \mathsf{P}^T\mathsf{A}\mathsf{P}, \quad \text{and} \quad \mathsf{Q}^T\mathsf{A} = \mathsf{A}\mathsf{Q} = \mathsf{Q}^T\mathsf{A}\mathsf{Q}. \tag{3.29}$$

Let us denote $\mathcal{V} = \mathrm{Im}\mathsf{Q}$. If $\mathbf{x} \in \mathsf{A}\mathcal{V}$, then $\mathbf{y} = \mathsf{Q}\mathbf{x}$ satisfies $\mathbf{y} \in \mathcal{V}$ and

$$\mathsf{Q}^T\mathsf{A}\mathsf{Q}\mathbf{x} = \mathsf{A}\mathsf{Q}\mathbf{x} = \mathsf{A}\mathbf{y},$$

so that

$$\mathsf{Q}^T\mathsf{A}\mathsf{Q}(\mathsf{A}\mathcal{V}) \subseteq \mathsf{A}\mathcal{V}. \tag{3.30}$$

Thus $\mathsf{A}\mathcal{V}$ is an invariant subspace of $\mathsf{Q}^T\mathsf{A}\mathsf{Q}$.

The following lemma shows that the mapping which assigns to each $\mathbf{x} \in \mathsf{A}\mathcal{V}$ the vector $\mathsf{Q}\mathbf{x} \in \mathcal{V}$ is expansive as in Fig. 3.4.

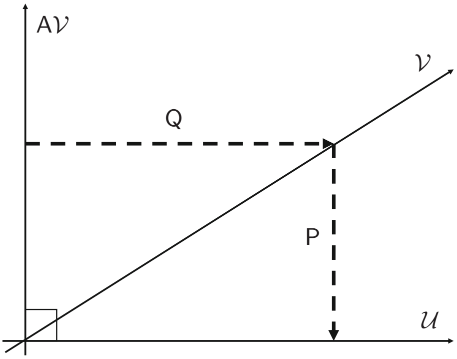

**Fig. 3.4.** Geometric illustration of Lemma 3.4

**Lemma 3.4.** *Let* $\mathsf{Q}$ *denote a conjugate projector on* $\mathcal{V}$ *and* $\mathbf{x} \in \mathsf{A}\mathcal{V}$*. Then*

$$\|\mathsf{Q}\mathbf{x}\| \ge \|\mathbf{x}\|.$$

*Proof.* For any $\mathbf{x} \in \mathsf{A}\mathcal{V}$, there is $\mathbf{y} \in \mathbb{R}^n$ such that $\mathbf{x} = \mathsf{A}\mathsf{Q}\mathbf{y}$. It follows that

$$\mathsf{Q}^T\mathbf{x} = \mathsf{Q}^T\mathsf{A}\mathsf{Q}\mathbf{y} = \mathsf{A}\mathsf{Q}\mathbf{y} = \mathbf{x}.$$

Since $\mathbf{x}^T\mathsf{Q}\mathbf{x} = \mathbf{x}^T\mathsf{Q}^T\mathbf{x} = \|\mathbf{x}\|^2$, we have

$$\|\mathsf{Q}\mathbf{x}\|^2 = \mathbf{x}^T\mathsf{Q}^T\mathsf{Q}\mathbf{x} = \mathbf{x}^T\left((\mathsf{Q}^T - \mathsf{I}) + \mathsf{I}\right)\left((\mathsf{Q} - \mathsf{I}) + \mathsf{I}\right)\mathbf{x} = \|(\mathsf{Q} - \mathsf{I})\mathbf{x}\|^2 + \|\mathbf{x}\|^2.$$

□

### 3.7.2 Minimization in Subspace

Let us assume that $\mathcal{U}$ is the subspace spanned by the columns of a full column rank matrix $\mathsf{U} \in \mathbb{R}^{n\times n}$ and notice that $\mathsf{U}^T\mathsf{A}\mathsf{U}$ is regular. Indeed, if $\mathsf{U}^T\mathsf{A}\mathsf{U}\mathbf{x} = \mathbf{o}$, then

$$\|\mathsf{U}\mathbf{x}\|_{\mathsf{A}}^2 = \mathbf{x}^T(\mathsf{U}^T\mathsf{A}\mathsf{U}\mathbf{x}) = 0,$$

so $\mathbf{x} = \mathbf{o}$ by the assumptions that $\mathsf{U}$ is the full column rank matrix and $\mathsf{A}$ is SPD. Thus we can define

$$\mathsf{P} = \mathsf{U}(\mathsf{U}^T\mathsf{A}\mathsf{U})^{-1}\mathsf{U}^T\mathsf{A}.$$

It is easy to check directly that $\mathsf{P}$ is a conjugate projector onto $\mathcal{U}$ as

$$\mathsf{P}^2 = \mathsf{U}(\mathsf{U}^T\mathsf{A}\mathsf{U})^{-1}\mathsf{U}^T\mathsf{A}\mathsf{U}(\mathsf{U}^T\mathsf{A}\mathsf{U})^{-1}\mathsf{U}^T\mathsf{A} = \mathsf{P}$$

and

$$\mathsf{P}^T\mathsf{A}(\mathsf{I}-\mathsf{P}) = \mathsf{A}\mathsf{U}(\mathsf{U}^T\mathsf{A}\mathsf{U})^{-1}\mathsf{U}^T\mathsf{A}(\mathsf{I}-\mathsf{U}(\mathsf{U}^T\mathsf{A}\mathsf{U})^{-1}\mathsf{U}^T\mathsf{A}) = \mathsf{O}.$$

Since any vector $\mathbf{x} \in \mathcal{U}$ can be written in the form $\mathbf{x} = \mathsf{U}\mathbf{y}$, $\mathbf{y} \in \mathbb{R}^m$, and

$$\mathsf{P}\mathbf{x} = \mathsf{U}(\mathsf{U}^T\mathsf{A}\mathsf{U})^{-1}\mathsf{U}^T\mathsf{A}\mathsf{U}\mathbf{y} = \mathsf{U}\mathbf{y} = \mathbf{x},$$

it follows that

$$\mathcal{U} = \mathrm{Im}\mathsf{P}.$$

The conjugate projector $\mathsf{P}$ onto $\mathcal{U}$ can be used for the solution of

$$\min_{\mathbf{x}\in\mathcal{U}} f(\mathbf{x}) = \min_{\mathbf{y}\in R^m} f(\mathsf{U}\mathbf{y}) = \min_{\mathbf{y}\in R^m} \frac{1}{2}\mathbf{y}^T\mathsf{U}^T\mathsf{A}\mathsf{U}\mathbf{y} - \mathbf{b}^T\mathsf{U}\mathbf{y}.$$

Using the gradient argument of Proposition 2.1, we get that the minimizer $\mathbf{y}^0$ of the latter problem satisfies

$$\mathsf{U}^T\mathsf{A}\mathsf{U}\mathbf{y}^0 = \mathsf{U}^T\mathbf{b}, \tag{3.31}$$

so that the minimizer $\mathbf{x}^0$ of $f$ over $\mathcal{U}$ satisfies

$$\mathbf{x}^0 = \mathsf{U}\mathbf{y}^0 = \mathsf{U}(\mathsf{U}^T\mathsf{A}\mathsf{U})^{-1}\mathsf{U}^T\mathbf{b} = \mathsf{P}\mathsf{A}^{-1}\mathbf{b}. \tag{3.32}$$

Our assumption concerning the ability to find the minimum of $f$ over $\mathcal{U}$ effectively amounts to the assumption that we are able to solve (3.31). Notice that we can evaluate the product $\mathsf{P}\mathsf{A}^{-1}\mathbf{b}$ without solving any system of linear equations with the matrix $\mathsf{A}$.

### 3.7.3 Conjugate Gradients in Conjugate Complement

In the next step we shall use the conjugate projectors $\mathsf{P}$ and $\mathsf{Q} = \mathsf{I} - \mathsf{P}$ to decompose our minimization problem (3.1) into the minimization on $\mathcal{U}$ and the minimization on $\mathcal{V} = \mathrm{Im}\mathsf{Q}$. We shall use the conjugate gradient method to solve the latter problem.

Two observations are needed to exploit the special structure of our problem. First, using Lemma 3.4, $\dim\mathcal{V} = \dim\mathsf{A}\mathcal{V}$, and (1.2), we get that the mapping which assigns to each $\mathbf{x} \in \mathsf{A}\mathcal{V}$ a vector $\mathsf{Q}\mathbf{x} \in \mathcal{V}$ is an isomorphism, so that

$$\mathsf{Q}(\mathsf{A}\mathcal{V}) = \mathcal{V}.$$

Second, using (3.29) and (3.32), we get

$$\mathbf{g}^0 = \mathsf{A}\mathbf{x}^0 - \mathbf{b} = \mathsf{A}\mathsf{P}\mathsf{A}^{-1}\mathbf{b} - \mathbf{b} = \mathsf{P}^T\mathbf{b} - \mathbf{b} = -\mathsf{Q}^T\mathbf{b}. \tag{3.33}$$

Using both observations, we get

$$\begin{aligned}\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}) &= \min_{\mathbf{y}\in\mathcal{U},\mathbf{z}\in\mathcal{V}} f(\mathbf{y}+\mathbf{z}) = \min_{\mathbf{y}\in\mathcal{U}} f(\mathbf{y}) + \min_{\mathbf{z}\in\mathcal{V}} f(\mathbf{z})\\ &= f(\mathbf{x}^0) + \min_{\mathbf{z}\in\mathcal{V}} f(\mathbf{z}) = f(\mathbf{x}^0) + \min_{\mathbf{x}\in\mathsf{A}\mathcal{V}} \frac{1}{2}\mathbf{x}^T\mathsf{Q}^T\mathsf{A}\mathsf{Q}\mathbf{x} - \mathbf{b}^T\mathsf{Q}\mathbf{x}\\ &= f(\mathbf{x}^0) + \min_{\mathbf{x}\in\mathsf{A}\mathcal{V}} \frac{1}{2}\mathbf{x}^T\mathsf{Q}^T\mathsf{A}\mathsf{Q}\mathbf{x} + \left(\mathbf{g}^0\right)^T\mathbf{x},\end{aligned} \tag{3.34}$$

where $\mathbf{x}^0$ is determined by (3.32).

It remains to solve the minimization problem (3.34). First observe that using Lemma 3.4, we get that $\mathsf{Q}^T\mathsf{A}\mathsf{Q}|\mathsf{A}\mathcal{V}$ is positive definite. Since by (3.33) $\mathbf{g}^0 \in \mathrm{Im}\mathsf{Q}^T$,

$$\mathrm{Im}\mathsf{Q}^T = \mathrm{Im}(\mathsf{Q}^T\mathsf{A}) = \mathrm{Im}(\mathsf{A}\mathsf{Q}) = \mathsf{A}\mathcal{V}, \tag{3.35}$$

and $\mathsf{A}\mathcal{V}$ is an invariant subspace of $\mathsf{Q}^T\mathsf{A}\mathsf{Q}$, we can use the procedure described in Sect. 3.2 to generate $\mathsf{Q}^T\mathsf{A}\mathsf{Q}$-conjugate vectors $\mathbf{p}^1, \dots, \mathbf{p}^k$ of

$$\mathcal{K}^k = \mathcal{K}^k(\mathsf{Q}^T\mathsf{A}\mathsf{Q}, \mathbf{g}^0) = \mathrm{Span}\{\mathbf{g}^0, \mathsf{Q}^T\mathsf{A}\mathsf{Q}\mathbf{g}^0, \dots, (\mathsf{Q}^T\mathsf{A}\mathsf{Q})^{k-1}\mathbf{g}^0\}.$$

It simply follows that

$$\mathbf{q}^1 = \mathsf{Q}\mathbf{p}^1, \mathbf{q}^2 = \mathsf{Q}\mathbf{p}^2, \dots$$

are $\mathsf{A}$-conjugate vectors of $\mathcal{V}$. Using (3.14), $\mathbf{p}^i \in \mathsf{A}\mathcal{V}$, and Lemma 3.4, it is easy to see that

$$\|\mathbf{q}^k\| \geq \|\mathbf{p}^k\| \geq \|\mathbf{g}^{k-1}\|,$$

so that we can generate a new conjugate direction $\mathbf{q}^k$ whenever $\mathbf{g}^{k-1} \neq \mathbf{o}$. We can sum up the most important properties of the CG algorithm with the preconditioning by the conjugate projector into the following theorem.

**Theorem 3.5.** *Let $\mathbf{x}^k$ be generated by Algorithm 3.4 to find the solution $\widehat{\mathbf{x}}$ of (3.1) with a full column rank matrix $\mathsf{U} \in \mathbb{R}^{n\times m}$. Then the algorithm is well defined and there is $k \leq n - m$ such that $\mathbf{x}^k = \widehat{\mathbf{x}}$. Moreover, the following statements hold for $i = 1, \dots, k$:*

*(i)* $f(\mathbf{x}^i) = \min\{f(\mathbf{x}) : \mathbf{x} \in \mathcal{U} + \mathsf{Q}\mathcal{K}^i(\mathsf{Q}^T\mathsf{A}\mathsf{Q}, \mathbf{g}^0)\}$.
*(ii)* $\|\mathbf{q}^i\| \geq \|\mathbf{g}^{i-1}\|$.
*(iii)* $(\mathbf{q}^i)^T\mathsf{A}\mathbf{q}^j = 0$ *for* $i > j$.
*(iv)* $(\mathbf{q}^i)^T\mathsf{A}\mathbf{x} = 0$ *for* $\mathbf{x} \in \mathcal{U}$.

The complete conjugate gradient algorithm with the preconditioning by the conjugate projector reads as follows:

**Algorithm 3.4. Conjugate gradients with projector preconditioning (CGPP).**

```
Given a symmetric positive definite matrix $A \in \mathbb{R}^{n\times n}$, a full column rank matrix
$U \in \mathbb{R}^{n\times m}$, and $\mathbf{b} \in \mathbb{R}^n$.

Step 0. {Initialization.}
        $P = U(U^TAU)^{-1}U^TA,\ Q = I - P$
        $\mathbf{x}^0 = PA^{-1}\mathbf{b} = U(U^TAU)^{-1}U^T\mathbf{b}$
        $k = 1,\ \mathbf{g}^0 = A\mathbf{x}^0 - \mathbf{b},\ \mathbf{q}^1 = Q\mathbf{g}^0$

Step 1. {Conjugate gradient loop. }
        while $\|\mathbf{g}^{k-1}\| > 0$
          $\alpha_k = (\mathbf{g}^{k-1})^T\mathbf{q}^k/(\mathbf{q}^k)^TA\mathbf{q}^k$
          $\mathbf{x}^k = \mathbf{x}^{k-1} - \alpha_k\mathbf{q}^k$
          $\mathbf{g}^k = \mathbf{g}^{k-1} - \alpha_kA\mathbf{q}^k$
          $\beta_k = (\mathbf{g}^k)^TA\mathbf{q}^k/(\mathbf{q}^k)^TA\mathbf{q}^k$
          $\mathbf{q}^{k+1} = Q\mathbf{g}^k + \beta_k\mathbf{q}^k$
          $k = k + 1$
        end while

Step 2. {Return a (possibly approximate) solution.}
          $\widetilde{\mathbf{x}} = \mathbf{x}^k$
```

### 3.7.4 Preconditioning Effect

As we have seen in the previous section, the iterations of Algorithm 3.4 may be considered as the conjugate gradient iterations for the minimization of

$$f_{0,Q}(\mathbf{x}) = \frac{1}{2}\mathbf{x}^TQ^TAQ\mathbf{x} + (\mathbf{g}^0)^T\mathbf{x}$$

that generate the iterations

$$\mathbf{x}^k \in \mathcal{K}^k(Q^TAQ, \mathbf{g}^0) \subseteq A\mathcal{V}.$$

Thus only the positive definite restriction $Q^TAQ|A\mathcal{V}$ of $Q^TAQ$ to $A\mathcal{V}$ takes part in the process of solution, and the rate of convergence may be estimated by the spectral condition number $\kappa(Q^TAQ|A\mathcal{V})$ of $Q^TAQ|A\mathcal{V}$.

It is rather easy to see that

$$\kappa(Q^TAQ|A\mathcal{V}) \le \kappa(A).$$

Indeed, denoting by $\lambda_1 \ge \cdots \ge \lambda_n$ the eigenvalues of $A$, we can observe that if $\mathbf{x} \in A\mathcal{V}$ and $\|\mathbf{x}\| = 1$, then by Lemma 3.4

$$\mathbf{x}^TQ^TAQ\mathbf{x} \ge (Q\mathbf{x})^TA(Q\mathbf{x})/\|Q\mathbf{x}\|^2 \ge \lambda_n$$

and

$$\mathbf{x}^TQ^TAQ\mathbf{x} \le \mathbf{x}^TQ^TAQ\mathbf{x} + \mathbf{x}^TP^TAP\mathbf{x} = \mathbf{x}^TA\mathbf{x} \le \lambda_1. \tag{3.36}$$

To see the preconditioning effect of the algorithm in more detail, let us denote by $\mathcal{E}$ the $m$-dimensional subspace spanned by the eigenvectors corresponding to the $m$ smallest eigenvalues $\lambda_{n-m+1} \geq \cdots \geq \lambda_n$ of $\mathsf{A}$, and let $\mathsf{R}_{\mathsf{A}\mathcal{U}}$ and $\mathsf{R}_{\mathcal{E}}$ denote the orthogonal projectors on $\mathsf{A}\mathcal{U}$ and $\mathcal{E}$, respectively. Let

$$\gamma = \|\mathsf{R}_{\mathsf{A}\mathcal{U}} - \mathsf{R}_{\mathcal{E}}\|$$

denote the *gap* between $\mathsf{A}\mathcal{U}$ and $\mathcal{E}$. It can be evaluated provided we have matrices $\mathsf{U}$ and $\mathsf{E}$ whose columns form the orthonormal bases of $\mathsf{A}\mathcal{U}$ and $\mathcal{E}$, respectively. It is known [170] that if $\sigma$ is the least singular value of $\mathsf{U}^T\mathsf{E}$, then

$$\gamma = \sqrt{1-\sigma^2} \leq 1.$$

**Theorem 3.6.** *Let* $\mathcal{U}, \mathcal{V}, \mathsf{Q}$ *be those of Algorithm 3.4, let* $\lambda_1 \geq \cdots \geq \lambda_n$ *denote the eigenvalues of* $\mathsf{A}$*, and let* $\overline{\lambda}_{\min}$ *denote the least nonzero eigenvalue of* $\mathsf{Q}^T\mathsf{A}\mathsf{Q}$*. Then*

$$\lambda_n \leq \sqrt{(1-\gamma^2)\lambda_{n-m}^2 + \gamma^2\lambda_n^2} \leq \overline{\lambda}_{\min} \tag{3.37}$$

*and*

$$\kappa(\mathsf{Q}^T\mathsf{A}\mathsf{Q}|\mathsf{A}\mathcal{V}) \leq \frac{\lambda_1}{\sqrt{(1-\gamma^2)\lambda_{n-m}^2 + \gamma^2\lambda_n^2}}.$$

*Proof.* Let $\mathbf{x} \in \mathsf{A}\mathcal{V}$, $\|\mathbf{x}\| = 1$, so that $\|\mathsf{Q}\mathbf{x}\| \geq 1$ by Lemma 3.4. Observing that $\mathrm{Im}\mathsf{R}_{\mathcal{E}}$ and $\mathrm{Im}(\mathsf{I} - \mathsf{R}_{\mathcal{E}})$ are orthogonal invariant subspaces of $\mathsf{A}$, we get that

$$\begin{aligned} \|\mathsf{A}\mathsf{Q}\mathbf{x}\|^2 &= \|\mathsf{A}(\mathsf{I}-\mathsf{R}_{\mathcal{E}})\mathsf{Q}\mathbf{x}\|^2 + \|\mathsf{A}\mathsf{R}_{\mathcal{E}}\mathsf{Q}\mathbf{x}\|^2 \\ &\geq \lambda_{n-m}^2\|(\mathsf{I}-\mathsf{R}_{\mathcal{E}})\mathsf{Q}\mathbf{x}\|^2 + \lambda_n^2\|\mathsf{R}_{\mathcal{E}}\mathsf{Q}\mathbf{x}\|^2 \\ &\geq \left(\lambda_{n-m}^2\|(\mathsf{I}-\mathsf{R}_{\mathcal{E}})\mathsf{Q}\mathbf{x}\|^2 + \lambda_n^2\|\mathsf{R}_{\mathcal{E}}\mathsf{Q}\mathbf{x}\|^2\right)/\|\mathsf{Q}\mathbf{x}\|^2 \\ &\geq \lambda_{n-m}^2(1-\xi^2) + \lambda_n^2\xi^2, \end{aligned} \tag{3.38}$$

where $\xi = \|\mathsf{Q}\mathbf{x}\|^{-1}\|\mathsf{R}_{\mathcal{E}}\mathsf{Q}\mathbf{x}\|$. We have used that

$$\|(\mathsf{I}-\mathsf{R}_{\mathcal{E}})\mathsf{Q}\mathbf{x}\|^2 + \|\mathsf{R}_{\mathcal{E}}\mathsf{Q}\mathbf{x}\|^2 = \|\mathsf{Q}\mathbf{x}\|^2.$$

Since $\mathrm{Im}\mathsf{Q} = \mathcal{V}$, it follows by the definition of $\mathsf{R}_{\mathsf{A}\mathcal{U}}$ that $\mathsf{R}_{\mathsf{A}\mathcal{U}}\mathsf{Q} = \mathsf{O}$ and

$$\xi = \|\mathsf{Q}\mathbf{x}\|^{-1}\|(\mathsf{R}_{\mathcal{E}} - \mathsf{R}_{\mathsf{A}\mathcal{U}})\mathsf{Q}\mathbf{x}\| \leq \gamma.$$

As the last expression in (3.38) is a decreasing function of $\xi$ for $\xi \geq 0$, it follows that

$$\|\mathsf{Q}^T\mathsf{A}\mathsf{Q}\mathbf{x}\|^2 = \|\mathsf{A}\mathsf{Q}\mathbf{x}\|^2 \geq \lambda_{n-m}^2(1-\gamma^2) + \lambda_n^2\gamma^2.$$

The rest is an easy consequence of (3.36). □

The above theorem suggests that the preconditioning by the conjugate projector is efficient when $\mathcal{U}$ approximates the subspace spanned by the eigenvectors which correspond to the smallest eigenvalues of $\mathsf{A}$. If $\mathsf{U}^T\mathsf{E}$ is nonsingular and $\lambda_n < \lambda_{n-m}$, then $\gamma < 1$ and

$$\kappa(\mathsf{Q}^T\mathsf{A}\mathsf{Q}|\mathsf{A}\mathcal{V}) < \kappa(\mathsf{A}).$$

If the minimization problem arises from the discretization of elliptic partial differential equations, than $\mathsf{U}$ can be obtained by aggregation. It turns out that even the subspace with a very small dimension can considerably improve the rate of convergence. See Sect. 3.10.1 for a numerical example.

## 3.8 Conjugate Gradients for More General Problems

Let $\mathsf{A}$ be only positive semidefinite, so that the cost function $f$ is convex but not strictly convex, and let the unconstrained minimization problem (3.1) be well posed, i.e., $\mathbf{b} \in \mathrm{Im}\mathsf{A}$ by Proposition 2.1.

If we start the conjugate gradient algorithm from arbitrary $\mathbf{x}^0 \in \mathbb{R}^n$, then the gradient $\mathbf{g}^0$ and the Krylov space $\mathcal{K}^n(\mathsf{A}, \mathbf{g}^0)$ satisfy

$$\mathbf{g}^0 \in \mathrm{Im}\mathsf{A} \quad \text{and} \quad \mathcal{K}^n(\mathsf{A}, \mathbf{g}^0) \subseteq \mathrm{Im}\mathsf{A}.$$

Since the CG method picks the conjugate directions from $\mathcal{K}^n(\mathsf{A}, \mathbf{g}^0)$, it follows that the method works only on the range of $\mathsf{A}$. Thus the algorithm generates the iterates $\mathbf{x}^k$ which converge to a solution $\overline{\mathbf{x}}$ with the rate of convergence which can be described by the distribution of the spectrum of the restriction $\mathsf{A}|\mathcal{K}^n(\mathsf{A}, \mathbf{g}^0)$. Observing that the least eigenvalue of $\mathsf{A}|\mathcal{K}^n(\mathsf{A}, \mathbf{g}^0)$ is bounded from below by the least nonzero eigenvalue $\overline{\lambda}_{\min}$ of $\mathsf{A}$, we get the error estimate

$$\|\mathbf{e}^k\|_{\mathsf{A}} \le 2 \left( \frac{\sqrt{\overline{\kappa}(\mathsf{A})} - 1}{\sqrt{\overline{\kappa}(\mathsf{A})} + 1} \right)^k \|\mathbf{e}^0\|_{\mathsf{A}}, \tag{3.39}$$

where $\overline{\kappa}(\mathsf{A})$ denotes the *regular spectral condition number* of $\mathsf{A}$ defined by

$$\overline{\kappa}(\mathsf{A}) = \kappa(\mathsf{A}|\mathrm{Im}\mathsf{A}) = \lambda_{\max}/\overline{\lambda}_{\min}.$$

Let $\mathsf{P}$ and $\mathsf{Q} = \mathsf{I} - \mathsf{P}$ denote the orthogonal projectors on $\mathrm{Im}\mathsf{A}$ and $\mathrm{Ker}\mathsf{A}$, respectively, so that

$$\overline{\mathbf{x}} = \mathsf{P}\overline{\mathbf{x}} + \mathsf{Q}\overline{\mathbf{x}}.$$

Since the reduction $\mathsf{A}|\mathrm{Im}\mathsf{A}$ is nonsingular, it follows that there is a unique solution $\overline{\mathbf{x}}_{\mathrm{LS}} \in \mathrm{Im}\mathsf{A}$ of (3.1), and by Proposition 2.1 any solution $\overline{\mathbf{x}}$ satisfies

$$\overline{\mathbf{x}} = \widehat{\mathbf{x}}_{\mathrm{LS}} + \mathbf{d}, \quad \mathbf{d} \in \mathrm{Ker}\mathsf{A}.$$

Thus if $\overline{\mathbf{x}}$ is a solution of (3.1), then $\mathsf{P}\overline{\mathbf{x}} = \overline{\mathbf{x}}_{\mathrm{LS}}$ and

$$\mathsf{Q}\overline{\mathbf{x}} = \mathsf{Q}\mathbf{x}^0.$$

It follows that $\overline{\mathbf{x}}_{\mathrm{LS}}$ is the least square solution of $\mathsf{A}\mathbf{x} = \mathbf{b}$, and to get it by the conjugate gradient algorithm, it is enough to take $\mathbf{x}^0 \in \mathrm{Im}\mathsf{A}$.

If $\mathsf{A}$ is indefinite, then, using the arguments of Sect. 3.2, it is easy to check that the conjugate gradient method still generates conjugate directions, but it fails when $(\mathbf{p}^k)^T \mathsf{A}\mathbf{p}^k = 0$. The latter case may happen with $\mathbf{p}^k \notin \mathrm{Ker}\mathsf{A}$, as in

$$[1,\ 1]\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}\begin{bmatrix} 1 \\ 1 \end{bmatrix} = 0.$$

It follows that there is no guarantee that the CG Algorithm 3.1 is able, at least without modifications, to find a stationary point of $f$.

## 3.9 Convergence in Presence of Rounding Errors

The elegant mathematical theory presented above assumes implementation of the conjugate gradient algorithm in exact arithmetic and captures well the performance of only a limited number of conjugate gradient iterations in computer arithmetics. Since we are going to use the conjugate gradient method mainly for a low-precision approximation of well-conditioned auxiliary problems, we shall base our exposition on this theory in what follows. However, it is still useful to be aware of possible effects of rounding errors that accompany any computer implementation of the conjugate gradient algorithm.

It has been known since the introduction of the CG method and the Lanczos method [140], which generates the same iterates, that, when used in finite precision arithmetic, the vectors generated by these algorithms can seriously violate their theoretical properties. In particular, it has been observed that the evaluated gradients can lose their orthogonality after as small a number of iterations as twenty, and that nearly dependent conjugate directions can be generated. In spite of these effects, it has been observed that the conjugate gradient method still converges in finite precision arithmetic, but that the convergence is delayed [105, 107].

Undesirable effects of the rounding errors can be reduced by reorthogonalization. A simple analysis reveals that the full reorthogonalization of the gradients is costly and requires large memory. A key to an efficient implementation of the reorthogonalization is based on observation that accumulation of the rounding errors has a regular pattern, namely, that large perturbations of the generated vectors belong to the space generated by the eigenvectors of $\mathsf{A}$ which can be approximated well by the vectors from the current Krylov space. This has led to the efficient implementation of the conjugate gradient method based on the *selective orthogonalization* proposed by Parlett and Scott [158]. More details and information about the effects of rounding errors and implementation of the conjugate gradient method in finite arithmetic can be found in the comprehensive review paper by Meurant and Strakoš [152].

## 3.10 Numerical Experiments

Here we illustrate the performance of the CG algorithm and the effect of preconditioning on the solution of an ill-conditioned benchmark and a class of well-conditioned problems. The latter was proposed to resemble the class of problems arising from application of the multigrid or domain decomposition methods to the elliptic partial differential equations. The cost functions $f_{\mathrm{L},h}$ and $f_{\mathrm{LW},h}$ introduced here are used in Sects. 4.8, 5.11, and 6.12 as benchmarks for the solution of constrained problems, so that we can assess additional complexity arising from implementation of various constraints and better understand our algorithms. Moreover, using the same cost functions in our benchmarks considerably simplifies their implementation.

### 3.10.1 Basic CG and Preconditioning

Let $\Omega = (0,1) \times (0,1)$ denote an open domain with the boundary $\Gamma$ and its part $\Gamma_u = \{0\} \times [0,1]$. Let $H^1(\Omega)$ denote the Sobolev space of the first order in the space $L^2(\Omega)$ of functions on $\Omega$ whose squares are integrable in the Lebesgue sense, let

$$\mathcal{V} = \{u \in H^1(\Omega) : \ u = 0 \ \text{ on } \ \Gamma_u\},$$

and let us define for any $u \in H^1(\Omega)$

$$f_{\mathrm{L}}(u) = \frac{1}{2}\int_{\Omega} \|\nabla u(x)\|^2 \mathrm{d}\Omega + \int_{\Omega} u \mathrm{d}\Omega.$$

Thus we can define the continuous problem to find

$$\min_{u \in \mathcal{V}} f_{\mathrm{L}}(u). \tag{3.40}$$

Our ill-conditioned benchmark was obtained from (3.40) by the finite element discretization using a regular grid defined by the discretization parameter $h$ and linear elements. The Dirichlet conditions were enhanced into the Hessian $\mathsf{A}_{\mathrm{L},h}$ of the discretized cost function $f_{\mathrm{L},h}$, so that $\mathsf{A}_{\mathrm{L},h} \in \mathbb{R}^{n \times n}$ is positive definite, $n = p(p-1)$, and $p = 1/h + 1$. Moreover, $\mathsf{A}_{\mathrm{L},h}$ is known to be ill-conditioned with the condition number $\kappa(\mathsf{A}_{\mathrm{L},h}) \approx h^{-2}$. The computations were carried out with $h = 1/32$, which corresponds to $n = 1056$ unknowns.

We used the benchmark to compare the performance of CG, CG with SSOR preconditioning, and CG with preconditioning by the conjugate projector. To define the conjugate projector, we decomposed the domain into $4 \times 4$ squares with typically $8 \times 8$ variables which were aggregated by means of the matrix $\mathsf{U}$ with 16 columns.

The graph of the norm of the gradient (vertical axis) against the number of iterations for the basic CG algorithm (CG), the CG algorithm with SSOR preconditioning (CG–SSOR), and the CG algorithm with preconditioning by the conjugate projector (CG–CP) is in Fig. 3.5. We can see that though the performance of the CG algorithm is poor if the Hessian of the cost function is ill-conditioned, it can be considerably improved by preconditioning.

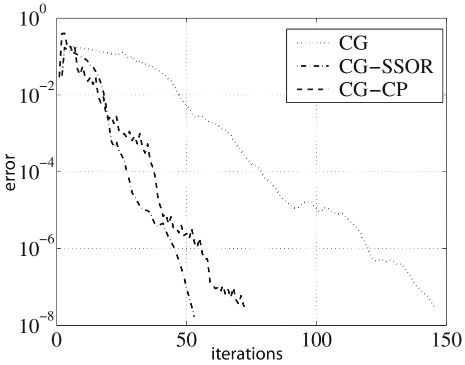

**Fig. 3.5.** Convergence of CG, CG–SSOR, and CG–CP algorithms

### 3.10.2 Numerical Demonstration of Optimality

To illustrate the concept of optimality, let us consider the class of problems to minimize

$$f_{\mathrm{LW},h}(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathsf{A}_{\mathrm{LW},h}\mathbf{x} - \mathbf{b}_{\mathrm{LW},h}^T\mathbf{x},$$

where

$$\mathsf{A}_{\mathrm{LW},h} = \mathsf{A}_{\mathrm{L},h} + 2\mathsf{I}, \quad [\mathbf{b}_{\mathrm{LW},h}]_i = -1, \quad i = 1, \ldots, n, \quad n = 1/h + 1.$$

The class of problems can be given a mechanical interpretation associated to the expanding spring systems on Winkler's foundation. Using Gershgorin's theorem, it can be proved that the spectrum of the Hessian $\mathsf{A}_{\mathrm{LW},h}$ of $f_{\mathrm{LW},h}$ is located in the interval $[2, 10]$, so that $\kappa(\mathsf{A}_{\mathrm{LW},h}) \le 5$.

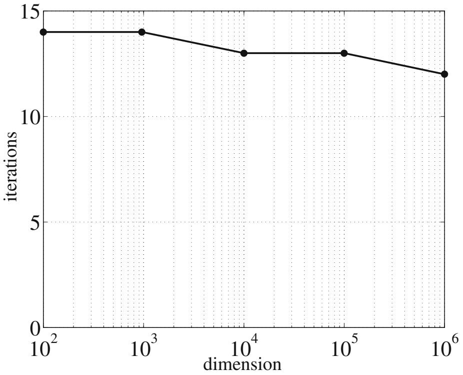

**Fig. 3.6.** Optimality of CG for a class of well-conditioned problems

In Fig. 3.6, we can see the numbers of CG iterations $k_n$ (vertical axis) that were necessary to reduce the norm of the gradient by $10^{-6}$ for the problems with the dimension $n$ ranging from 100 to 1000000. The dimension $n$ on the horizontal axis is in the logarithmic scale. We can see that $k_n$ varies mildly with varying $n$, in agreement with the theory developed in Sect. 3.5. Moreover, since the cost of the matrix–vector multiplications is in our case proportional to the dimension $n$ of the matrix $\mathsf{A}_{\mathrm{LW},h}$, it follows that the cost of the solution is also proportional to $n$.

## 3.11 Comments and Conclusions

The development of the conjugate gradient method was preceded by the method of conjugate directions [92]. If the conjugate directions are generated by means of a suitable matrix decomposition, the method can be considered as a variant of the direct methods of Sect. 1.5 (see, e.g., [169]).

Since its introduction in the early 1950s by Hestenes and Stiefel [117], a lot of research related to the development of the CG method has been carried out, so that there are many references concerning this subject. We refer an interested reader to the textbooks and research monographs by Saad [163], van der Vorst [178], Greenbaum [106], Hackbusch [110], Chen [21], and Axelsson [4] for more information. A comprehensive account of development of the CG method up to 1989 may be found in the paper by Golub and O'Leary [102]. Most of the research is concentrated on the development and analysis of preconditioners.

Preconditioning by conjugate projector presented in Sect. 3.7 was introduced by Dostál [39]. The same preconditioning with different analysis was presented independently by Marchuk and Kuznetsov [150] as the *conjugate gradients in subspace* or the *generalized conjugate gradient method* and by Nicolaides [154] as the *deflation method.*

Finding at each step the minimum over the subspace generated by all the previous search directions, the conjugate gradient method exploits all the information gathered during the previous iterations. To use this feature in the algorithms for the solution of constrained problems, it is important to generate long uninterrupted sequences of the conjugate gradient iterations. This strategy also supports exploitation of yet another unique feature of the conjugate gradient method, namely, its self-preconditioning capabilities that were described by van der Sluis and van der Vorst [168]. The latter property can also be described in terms of the preconditioning by the conjugate projector. Indeed, if $\mathsf{Q}_k$ denotes the conjugate projector onto the conjugate complement $\mathcal{V}$ of $\mathcal{U} = \mathrm{Span}\{\mathbf{p}_1, \dots, \mathbf{p}_{\mathrm{k}}\}$, then it is possible to give the bound on the rate of convergence of the conjugate gradient method starting from $\mathbf{x}_{k+1}$ in terms of the regular condition number $\overline{\kappa}_k = \overline{\kappa}(\mathsf{Q}_k^T \mathsf{A} \mathsf{Q}_k|\mathcal{V})$ of $\mathsf{Q}_k^T \mathsf{A} \mathsf{Q}_k|\mathcal{V}$ and observe that $\overline{\kappa}_k$ decreases with the increasing $k$.

For the solution of large problems, the basic CG algorithm is most successful when it is combined with the preconditioning which exploits additional information about $\mathsf{A}$, often obtained by tracing its generation. Thus the *multigrid* (see, e.g., Hackbusch [109] or Trottenberg et al. [176]) or *FETI* (see, e.g., Farhat, Mandel, and Roux [85], or Toselli and Widlund [175])-based preconditioners for the solution of problems arising from the discretization of elliptic partial differential equations exploit the information about the original continuous problems so efficiently that the discretized problems can be solved at a cost proportional to the number of unknowns. It follows that the conjugate gradient method should outperform direct solvers at least for some large problems. Special preconditioners for singular or nearly singular systems arising in optimization were proposed, e.g., by Hager [114].