Define

$$\Delta H_k = H_{k+1} - H_k, \quad \gamma_k = \nabla f(x_{k+1}) - \nabla f(x_k), \quad \delta_k = x_{k+1} - x_k.$$

Then the quasi-Newton relation is satisfied by the following updating rules.

1. *Rank-one correction scheme*: $\Delta H_k = \frac{(\delta_k - H_k\gamma_k)(\delta_k - H_k\gamma_k)^T}{\langle \delta_k - H_k\gamma_k, \gamma_k\rangle}$.
2. *Davidon–Fletcher–Powell scheme (DFP)*: $\Delta H_k = \frac{\delta_k\delta_k^T}{\langle \gamma_k, \delta_k\rangle} - \frac{H_k\gamma_k\gamma_k^T H_k}{\langle H_k\gamma_k, \gamma_k\rangle}$.
3. *Broyden–Fletcher–Goldfarb–Shanno scheme (BFGS)*:

$$\Delta H_k = \beta_k \frac{\delta_k\delta_k^T}{\langle \gamma_k, \delta_k\rangle} - \frac{H_k\gamma_k\delta_k^T + \delta_k\gamma_k^T H_k}{\langle \gamma_k, \delta_k\rangle},$$

where $\beta_k = 1 + \langle H_k\gamma_k, \gamma_k\rangle / \langle \gamma_k, \delta_k\rangle$.

Clearly, there are many other possibilities. From the computational point of view, BFGS is considered to be the most stable scheme.

Note that for quadratic functions, the variable metric methods usually terminate in at most n iterations. In a neighborhood of a strict local minimum x^* they demonstrate a *superlinear* rate of convergence: for any $x_0 \in \mathbb{R}^n$ close enough to x^* there exists a number N such that for all $k \geq N$ we have

$$\| x_{k+1} - x^* \| \leq \text{const} \cdot \| x_k - x^* \| \cdot \| x_{k-n} - x^* \|$$

(the proofs are very long and technical). As far as the worst-case global convergence is concerned, these methods are not better than the Gradient Method.

In the variable metric schemes it is necessary to store and update a symmetric $(n \times n)$-matrix. Thus, each iteration needs $O(n^2)$ auxiliary arithmetic operations. This feature is considered as one of the main drawbacks of the variable metric methods. It stimulated the interest in *conjugate gradient* schemes which have a much lower complexity of each iteration. We discuss these schemes in Sect. 1.3.2.

1.3.2 Conjugate Gradients

Conjugate gradient methods were initially proposed for minimizing quadratic functions. Consider the problem

$$\min_{x \in \mathbb{R}^n} f(x) \tag{1.3.2}$$

with $f(x) = \alpha + \langle a, x\rangle + \frac{1}{2}\langle Ax, x\rangle$ and $A = A^T \succ 0$. We have already seen that the solution of this problem is $x^* = -A^{-1}a$. Therefore, our objective function can be written in the following form:

$$f(x) = \alpha + \langle a, x\rangle + \tfrac{1}{2}\langle Ax, x\rangle \;=\; \alpha - \langle Ax^*, x\rangle + \tfrac{1}{2}\langle Ax, x\rangle$$

$$= \alpha - \tfrac{1}{2}\langle Ax^*, x^*\rangle + \tfrac{1}{2}\langle A(x - x^*), x - x^*\rangle.$$

Thus, $f^* = \alpha - \frac{1}{2}\langle Ax^*, x^*\rangle$ and $\nabla f(x) = A(x - x^*)$.

Suppose we are given a starting point $x_0 \in \mathbb{R}^n$. Consider the linear *Krylov* subspaces

$$\mathscr{L}_k = \mathrm{Lin}\{A(x_0 - x^*), \ldots, A^k(x_0 - x^*)\}, \quad k \geq 1,$$

where A^k is the kth power of matrix A. A sequence of points $\{x_k\}$ is generated by the *Conjugate Gradient Method* in accordance with the following rule.

$$\boxed{x_k = \arg\min\{f(x) \mid \; x \in x_0 + \mathscr{L}_k\}, \; k \geq 1.} \tag{1.3.3}$$

This definition looks quite artificial. However, later we will see that this method can be written in a pure "algorithmic" form. We need representation (1.3.3) only for theoretical analysis.

Lemma 1.3.1 *For any $k \geq 1$ we have $\mathscr{L}_k = \mathrm{Lin}\{\nabla f(x_0), \ldots, \nabla f(x_{k-1})\}$.*

Proof For $k = 1$, the statement is true since $\nabla f(x_0) = A(x_0 - x^*)$. Suppose that it is valid for some $k \geq 1$. Consider a point

$$x_k = x_0 + \sum_{i=1}^{k} \lambda^{(i)} A^i (x_0 - x^*) \in x_0 + \mathscr{L}_k$$

with some $\lambda \in \mathbb{R}^k$. Then

$$\nabla f(x_k) = A(x_0 - x^*) + \sum_{i=1}^{k} \lambda^{(i)} A^{i+1}(x_0 - x^*) = y + \lambda^{(k)} A^{k+1}(x_0 - x^*),$$

for a certain y from $\mathscr{L}_k$. Thus,

$$\mathscr{L}_{k+1} \equiv \mathrm{Lin}\{\mathscr{L}_k \bigcup A^{k+1}(x_0 - x^*)\} = \mathrm{Lin}\{\mathscr{L}_k \bigcup \nabla f(x_k)\}$$

$$= \mathrm{Lin}\{\nabla f(x_0), \ldots, \nabla f(x_k)\}. \qquad \square$$

The next result helps us to understand the behavior of the sequence $\{x_k\}$.

Lemma 1.3.2 *For any k, $i \geq 0$, $k \neq i$ we have $\langle \nabla f(x_k), \nabla f(x_i) \rangle = 0$.*

Proof Let $k > i$. Consider the function

$$\phi(\lambda) = f\left(x_0 + \sum_{j=1}^{k} \lambda^{(j)} \nabla f(x_{j-1})\right), \quad \lambda \in \mathbb{R}^k.$$

In view of Lemma 1.3.1, for some $\lambda_* \in \mathbb{R}^k$ we have $x_k = x_0 + \sum\limits_{j=1}^{k} \lambda_*^{(j)} \nabla f(x_{j-1})$. However, by definition, x_k is the minimum point of $f(\cdot)$ on $x_0 + \mathscr{L}_k$. Therefore $\nabla \phi(\lambda_*) = 0$. It remains to compute the components of the gradient:

$$0 = \frac{\partial \phi(\lambda_*)}{\partial \lambda^{(j)}} = \langle \nabla f(x_k), \nabla f(x_{j-1}) \rangle, \quad j = 1, \ldots, k. \quad \square$$

This lemma has two evident consequences.

Corollary 1.3.1 *The sequence generated by the Conjugate Gradient Method for problem (1.3.2) is finite.*

Proof Indeed, the number of nonzero orthogonal directions in $\mathbb{R}^n$ cannot exceed n. $\square$

Corollary 1.3.2 *For any $p \in \mathscr{L}_k$, $k \geq 1$, we have $\langle \nabla f(x_k), p \rangle = 0$.* $\square$

The last auxiliary result explains the name of the method. Let $\delta_i = x_{i+1} - x_i$. It is clear that $\mathscr{L}_k = \mathrm{Lin}\{\delta_0, \ldots, \delta_{k-1}\}$.

Lemma 1.3.3 *For any $k, i \geq 0$, $k \neq i$, we have $\langle A\delta_k, \delta_i \rangle = 0$.*

(Such directions are called *conjugate* with respect to A.)

Proof Without loss of generality, we can assume that $k > i$. Then

$$\langle A\delta_k, \delta_i \rangle = \langle A(x_{k+1} - x_k), \delta_i \rangle = \langle \nabla f(x_{k+1}) - \nabla f(x_k), \delta_i \rangle = 0$$

since $\delta_i = x_{i+1} - x_i \in \mathscr{L}_{i+1} \subseteq \mathscr{L}_k \subseteq \mathscr{L}_{k+1}$. $\square$

Let us show how we can write down the Conjugate Gradient Method in a more algorithmic form. Since $\mathscr{L}_k = \mathrm{Lin}\{\delta_0, \ldots, \delta_{k-1}\}$, we can represent x_{k+1} as follows:

$$x_{k+1} = x_k - h_k \nabla f(x_k) + \sum_{j=0}^{k-1} \lambda^{(j)} \delta_j.$$

In our notation, this is

$$\delta_k = -h_k \nabla f(x_k) + \sum_{j=0}^{k-1} \lambda^{(j)} \delta_j. \tag{1.3.4}$$

Let us compute the coefficients in this representation. Multiplying (1.3.4) by A and δ_i, $0 \le i \le k-1$, and using Lemma 1.3.3, we obtain

$$\begin{aligned} 0 = \langle A\delta_k, \delta_i \rangle &= -h_k \langle A \nabla f(x_k), \delta_i \rangle + \sum_{j=0}^{k-1} \lambda^{(j)} \langle A\delta_j, \delta_i \rangle \\ &= -h_k \langle A \nabla f(x_k), \delta_i \rangle + \lambda^{(i)} \langle A\delta_i, \delta_i \rangle \\ &= -h_k \langle \nabla f(x_k), A\delta_i \rangle + \lambda^{(i)} \langle A\delta_i, \delta_i \rangle \\ &= -h_k \langle \nabla f(x_k), \nabla f(x_{i+1}) - \nabla f(x_i) \rangle + \lambda^{(i)} \langle A\delta_i, \delta_i \rangle. \end{aligned}$$

Hence, in view of Lemma 1.3.2, $\lambda_i = 0$ for $i < k-1$. For $i = k-1$, we have

$$\lambda^{(k-1)} = \frac{h_k \|\nabla f(x_k)\|^2}{\langle A\delta_{k-1}, \delta_{k-1} \rangle} = \frac{h_k \|\nabla f(x_k)\|^2}{\langle \nabla f(x_k) - \nabla f(x_{k-1}), \delta_{k-1} \rangle}.$$

Thus, $x_{k+1} = x_k - h_k p_k$, where

$$p_k = \nabla f(x_k) - \frac{\|\nabla f(x_k)\|^2 \delta_{k-1}}{\langle \nabla f(x_k) - \nabla f(x_{k-1}), \delta_{k-1} \rangle} = \nabla f(x_k) - \frac{\|\nabla f(x_k)\|^2 p_{k-1}}{\langle \nabla f(x_k) - \nabla f(x_{k-1}), p_{k-1} \rangle}$$

since $\delta_{k-1} = -h_{k-1} p_{k-1}$ by definition of the directions $\{p_k\}$.

Note that we managed to write down the Conjugate Gradient Method in terms of the gradients of the objective function $f(\cdot)$. This provides us with the possibility of *formally* applying this scheme to minimize a general nonlinear function. Of course, such an extension destroys all properties of the process which are specific for quadratic functions. However, in a neighborhood of a strict local minimum, the objective function is close to quadratic. Therefore, asymptotically this method should be fast.

Let us present a general scheme of the Conjugate Gradient Method for minimizing a general nonlinear function.

Conjugate Gradient Method
0. Let $x_0 \in \mathbb{R}^n$. Compute $f(x_0)$, $\nabla f(x_0)$. Set $p_0 = \nabla f(x_0)$.
1. kth iteration ($k \geq 0$). (a) Find $x_{k+1} = x_k - h_k p_k$(by "exact" line search). (b) Compute $f(x_{k+1})$ and $\nabla f(x_{k+1})$. (c) Compute the coefficient β_k. (d) Define $p_{k+1} = \nabla f(x_{k+1}) - \beta_k p_k$.

In this scheme, we have not yet specified the coefficient β_k. In fact, there exist many different formulas for this coefficient. All of them give the same results on quadratic functions. However, in the general nonlinear case, they generate different sequences. Let us present the three most popular expressions.

1. *Dai–Yuan*: $\beta_k = \frac{\|\nabla f(x_{k+1})\|^2}{\langle \nabla f(x_{k+1}) - \nabla f(x_k), p_k \rangle}$.
2. *Fletcher–Rieves*: $\beta_k = -\frac{\|\nabla f(x_{k+1})\|^2}{\|\nabla f(x_k)\|^2}$.
3. *Polak–Ribbiere*: $\beta_k = -\frac{\langle \nabla f(x_{k+1}), \nabla f(x_{k+1}) - \nabla f(x_k) \rangle}{\|\nabla f(x_k)\|^2}$.

Recall that in the quadratic case, the Conjugate Gradient Method terminates in n iterations (or less). Algorithmically, this means that $p_n = 0$. In the general nonlinear case, this is not true. However, after n iterations, this direction loses its interpretation. Therefore, in all practical schemes, there exists a *restarting* strategy, which at some moment sets $\beta_k = 0$ (usually after every n iterations). This ensures the global convergence of the process (since we have the usual gradient step just after the restart, and all other iterations decrease the value of the objective function). In a neighborhood of a strict minimum, the conjugate gradient schemes demonstrate a local n-step quadratic convergence:

$$\| x_n - x^* \| \leq \text{const} \cdot \| x_0 - x^* \|^2 .$$

Note that this local convergence is slower than that of the variable metric methods. However, the conjugate gradient methods have the advantage of cheap iteration. As far as the global convergence is concerned, these schemes, in general, are not better than the simplest Gradient Method.