GOLUB – VAN LOAN MATRIX COMPUTATIONS 3rd ed.

A. Dax (1990). "The Convergence of Linear Stationary Iterative Processes for Solving Singular Unstructured Systems of Linear Equations," *SIAM Review 32*, 611–635.

Finally, the effect of rounding errors on the methods of this section are treated in

H. Wozniakowski (1978). "Roundoff-Error Analysis of Iterations for Large Linear Systems," *Numer. Math. 30*, 301–314.
P.A. Knight (1993). "Error Analysis of Stationary Iteration and Associated Problems," Ph.D. thesis, Department of Mathematics, University of Manchester, England.

10.2 The Conjugate Gradient Method

A difficulty associated with the SOR, Chebyshev semi-iterative, and related methods is that they depend upon parameters that are sometimes hard to choose properly. For example, the Chebyshev acceleration scheme needs good estimates of the largest and smallest eigenvalue of the underlying iteration matrix $M^{-1}N$. Unless this matrix is sufficiently structured, it may be analytically impossible and/or computationally expensive to do this.

In this section, we present a method without this difficulty for the symmetric positive definite $Ax = b$ problem, the well-known Hestenes-Stiefel conjugate gradient method. We derived this method in §9.3.1 from the Lanczos algorithm. The derivation now is from a different point of view and it will set the stage for various important generalizations in §10.3 and §10.4.

10.2.1 Steepest Descent

The starting point in the derivation is to consider how we might go about minimizing the function

$$\phi(x) \;=\; \frac{1}{2}x^TAx - x^Tb$$

where $b \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n\times n}$ is assumed to be positive definite and symmetric. The minimum value of $\phi(x)$ is $-b^TA^{-1}b/2$, achieved by setting $x = A^{-1}b$. Thus, minimizing ϕ and solving $Ax = b$ are equivalent problems if A is symmetric positive definite.

One of the simplest strategies for minimizing ϕ is the *method of steepest descent*. At a current point x_c the function ϕ decreases most rapidly in the direction of the negative gradient: $-\nabla\phi(x_c) = b - Ax_c$. We call

$$r_c = b - Ax_c$$

the *residual* of x_c. If the residual is nonzero, then there exists a positive α such that $\phi(x_c + \alpha r_c) < \phi(x_c)$. In the method of steepest descent (with exact line search) we set $\alpha = r_c^Tr_c/r_c^TAr_c$ thereby minimizing

$$\phi(x_c + \alpha r_c) \;=\; \phi(x_c) - \alpha r_c^Tr_c + \frac{1}{2}\alpha^2 r_c^TAr_c.$$

This gives

```
$x_0$ = initial guess
$r_0 = b - Ax_0$
$k = 0$
while $r_k \neq 0$
    $k = k + 1$
    $\alpha_k = r_{k-1}^Tr_{k-1}/r_{k-1}^TAr_{k-1}$
    $x_k = x_{k-1} + \alpha_k r_{k-1}$
    $r_k = b - Ax_k$
end
```
(10.2.1)

It can be shown that

$$\left(\phi(x_k) + \frac{1}{2}b^TA^{-1}b\right) \;\le\; \left(1 - \frac{1}{\kappa_2(A)}\right)\left(\phi(x_{k-1}) + \frac{1}{2}b^TA^{-1}b\right) \qquad (10.2.2)$$

which implies global convergence. Unfortunately, the rate of convergence may be prohibitively slow if the condition $\kappa_2(A) = \lambda_1(A)/\lambda_n(A)$ is large. Geometrically this means that the level curves of ϕ are very elongated hyperellipsoids and minimization corresponds to finding the lowest point in a relatively flat, steep-sided valley. In steepest descent, we are forced to traverse back and forth *across* the valley rather than *down* the valley. Stated another way, the gradient directions that arise during the iteration are not different enough.

10.2.2 General Search Directions

To avoid the pitfalls of steepest descent, we consider the successive minimization of ϕ along a set of directions $\{p_1, p_2, \ldots\}$ that do not necessarily correspond to the residuals $\{r_0, r_1, \ldots\}$. It is easy to show that $\phi(x_{k-1} + \alpha p_k)$ is minimized by setting

$$\alpha \;=\; \alpha_k \;=\; p_k^Tr_{k-1}/p_k^TAp_k.$$

With this choice it can be shown that

$$\phi(x_{k-1} + \alpha_kp_k) \;=\; \phi(x_{k-1}) - \frac{1}{2}\frac{(p_k^Tr_{k-1})^2}{p_k^TAp_k}. \qquad (10.2.3)$$

To ensure a reduction in the size of ϕ we insist that p_k *not* be orthogonal to r_{k-1}. This leads to the following framework:

```
$x_0 =$ initial guess
$r_0 = b - Ax_0$
$k = 0$
while $r_k \neq 0$
      $k = k + 1$
      Choose a direction $p_k$ such that $p_k^T r_{k-1} \neq 0$.        (10.2.4)
      $\alpha_k = p_k^T r_{k-1} / p_k^T A p_k$
      $x_k = x_{k-1} + \alpha_k p_k$
      $r_k = b - Ax_k$
end
```

Note that

$$x_k \in x_0 + \text{span}\{p_1, \ldots, p_k\} \equiv \{x_0 + \gamma_1 p_1 + \cdots + \gamma_k p_k : \gamma_i \in \mathbb{R}\}.$$

Our goal is to choose the search directions in a way that guarantees convergence without the shortcomings of steepest descent.

10.2.3 A-Conjugate Search Directions

If the search directions are linearly independent *and* x_k solves the problem

$$\min_{x \in x_0 + \text{span}\{p_1, \ldots, p_k\}} \phi(x) \qquad (10.2.5)$$

for $k = 1, 2, \ldots$, then convergence is guaranteed in at most n steps. This is because x_n minimizes ϕ over $\mathbb{R}^n$ and therefore satisfies $Ax_n = b$.

However, for this to be a viable approach the search directions must have the property that it is "easy" to compute x_k given x_{k-1}. Let us see what this says about the determination of p_k. If

$$x_k = x_0 + P_{k-1} y + \alpha p_k$$

where $P_{k-1} = [\, p_1, \ldots, p_{k-1} \,]$, $y \in \mathbb{R}^{k-1}$, and $\alpha \in \mathbb{R}$, then

$$\phi(x_k) = \phi(x_0 + P_{k-1} y) + \alpha y^T P_{k-1}^T A p_k + \frac{\alpha^2}{2} p_k^T A p_k - \alpha p_k^T r_0.$$

If $p_k \in \text{span}\{Ap_1, \ldots, Ap_{k-1}\}^\perp$, then the cross term $\alpha y^T P_{k-1}^T A p_k$ is zero and the search for the minimizing x_k splits into a pair of uncoupled minimizations, one for y and one for α:

$$\begin{aligned}
\min_{x_k \in x_0 + \text{span}\{p_1, \ldots, p_k\}} \phi(x_k) &= \min_{y, \alpha} \; \phi(x_0 + P_{k-1} y + \alpha p_k) \\
&= \min_{y, \alpha} \left(\phi(x_0 + P_{k-1} y) + \frac{\alpha^2}{2} p_k^T A p_k - \alpha p_k^T r_0 \right) \\
&= \min_{y} \; \phi(x_0 + P_{k-1} y) + \min_{\alpha} \left(\frac{\alpha^2}{2} p_k^T A p_k - \alpha p_k^T r_0 \right).
\end{aligned}$$

Note that if y_{k-1} solves the first min problem then $x_{k-1} = x_0 + P_{k-1} y_{k-1}$ minimizes ϕ over $x_0 + \text{span}\{p_1, \ldots, p_{k-1}\}$. The solution to the α min problem is given by $\alpha_k = p_k^T r_0 / p_k^T A p_k$. Note that because of A-conjugacy,

$$\begin{aligned}
p_k^T r_{k-1} &= p_k^T (b - Ax_{k-1}) \\
&= p_k^T (b - A(x_0 + P_{k-1} y_{k-1})) = p_k^T r_0.
\end{aligned}$$

With these results it follows that $x_k = x_{k-1} + \alpha_k p_k$ and we obtain the following instance of (10.2.4):

```
$x_0 =$ initial guess
$k = 0$
$r_0 = b - Ax_0$
while $r_k \neq 0$
      $k = k + 1$
      Choose $p_k \in \text{span}\{Ap_1, \ldots, Ap_{k-1}\}^\perp$ so $p_k^T r_{k-1} \neq 0$.    (10.2.6)
      $\alpha_k = p_k^T r_{k-1} / p_k^T A p_k$
      $x_k = x_{k-1} + \alpha_k p_k$
      $r_k = b - Ax_k$
end
```

The following lemma shows that it is possible to find the search directions with the required properties.

Lemma 10.2.1 *If $r_{k-1} \neq 0$, then there exists a $p_k \in$ span$\{Ap_1, \ldots, Ap_{k-1}\}^\perp$ such that $p_k^T r_{k-1} \neq 0$.*

Proof. For the case $k = 1$, set $p_1 = r_0$. If $k > 1$, then since $r_{k-1} \neq 0$ it follows that

$$\begin{aligned}
A^{-1} b \notin x_0 + \text{span}\{p_1, \ldots, p_{k-1}\} \;&\Rightarrow\; b \notin Ax_0 + \text{span}\{Ap_1, \ldots, Ap_{k-1}\} \\
&\Rightarrow\; r_0 \notin \text{span}\{Ap_1, \ldots, Ap_{k-1}\}.
\end{aligned}$$

Thus there exists a $p \in \text{span}\{Ap_1, \ldots, Ap_{k-1}\}^\perp$ such that $p^T r_0 \neq 0$. But $x_{k-1} \in x_0 + \text{span}\{p_1, \ldots, p_{k-1}\}$ and so $r_{k-1} \in r_0 + \text{span}\{Ap_1, \ldots, Ap_{k-1}\}$. It follows that $p^T r_{k-1} = p^T r_0 \neq 0$. □

The search directions in (10.2.6) are said to be *A-conjugate* because $p_i^T A p_j = 0$ for all $i \neq j$. Note that if $P_k = [\, p_1, \ldots, p_k \,]$ is the matrix of these vectors, then

$$P_k^T A P_k = \text{diag}(p_1^T A p_1, \ldots, p_k^T A p_k)$$

is nonsingular since A is positive definite and the search directions are nonzero. It follows that P_k has full column rank. This guarantees convergence in (10.2.6) in at most n steps because x_n (if we get that far) minimizes $\phi(x)$ over $\text{ran}(P_n) = \mathbb{R}^n$.

10.2.4 Choosing a Best Search Direction

A way to combine the positive aspects of steepest descent and A-conjugate searching is to choose p_k in (10.2.6) to be the closest vector to r_{k-1} that is A-conjugate to $p_1, \ldots, p_{k-1}$. This defines "version zero" of the *method of conjugate gradients*:

```
$x_0$ = initial guess
$k = 0$
$r_0 = b - Ax_0$
while $r_k \neq 0$
      $k = k + 1$
      if $k = 1$
            $p_1 = r_0$
      else
            Let $p_k$ minimize $\| p - r_{k-1} \|_2$ over all vectors                    (10.2.7)
                  $p \subset \mathrm{span}\{Ap_1, \ldots, Ap_{k-1}\}^{\perp}$
      end
      $\alpha_k = p_k^T r_{k-1} / p_k^T A p_k$
      $x_k = x_{k-1} + \alpha_k p_k$
      $r_k = b - Ax_k$
end
$x = x_k$
```

To make this an effective sparse $Ax = b$ solver, we need an efficient method for computing p_k. A considerable amount of analysis is required to develop the final recursions. The first step is to show that p_k is the minimum residual of a certain least squares problem.

Lemma 10.2.2 *For $k \geq 2$ the vectors p_k generated by (10.2.7) satisfy*

$$p_k = r_{k-1} - AP_{k-1}z_{k-1},$$

where $P_{k-1} = [p_1, \ldots, p_{k-1}]$ and z_{k-1} solves $\displaystyle\min_{z \in \mathbb{R}^{k-1}} \| r_{k-1} - AP_{k-1}z \|_2$.

Proof. Suppose z_{k-1} solves the above LS problem and let p be the associated minimum residual:

$$p = r_{k-1} - AP_{k-1}z_{k-1}.$$

It follows that $p^T AP_{k-1} = 0$. Moreover, $p = [I - (AP_{k-1})(AP_{k-1})^+]r_{k-1}$ is the orthogonal projection of r_{k-1} into $\mathrm{ran}(AP_{k-1})^{\perp}$ and so it is the closest vector in $\mathrm{ran}(AP_{k-1})^{\perp}$ to r_{k-1}. Thus, $p = p_k$. □

With this result we can establish a number of important relationships between the residuals r_k, the search directions p_k, and the Krylov subspaces

$$\mathcal{K}(r_0, A, k) = \mathrm{span}\{r_0, Ar_0, \ldots, A^{k-1}r_0\}.$$

Theorem 10.2.3 *After k iterations in (10.2.7) we have*

$$r_k = r_{k-1} - \alpha_k A p_k \tag{10.2.8}$$

$$P_k^T r_k = 0 \tag{10.2.9}$$

$$\mathrm{span}\{p_1, \ldots, p_k\} = \mathrm{span}\{r_0, \ldots, r_{k-1}\} = \mathcal{K}(r_0, A, k) \tag{10.2.10}$$

and the residuals $r_0, \ldots, r_k$ are mutually orthogonal.

Proof. Equation (10.2.8) follows by applying A to both sides of $x_k = x_{k-1} + \alpha_k p_k$ and using the definition of the residual.

To prove (10.2.9), we recall that $x_k = x_0 + P_k y_k$ where y_k is the minimizer of

$$\phi(x_0 + P_k y) = \phi(x_0) + \frac{1}{2}y^T(P_k^T A P_k)y - y^T P_k(b - Ax_0).$$

But this means that y_k solves the linear system $(P_k^T A P_k)y = P_k^T(b - Ax_0)$. Thus

$$0 = P_k^T(b - Ax_0) - P_k^T A P_k y_k = P_k^T(b - A(x_0 + P_k y_k)) = P_k^T r_k.$$

To prove (10.2.10) we note from (10.2.8) that

$$\{Ap_1, \ldots, Ap_{k-1}\} \subseteq \mathrm{span}\{r_0, \ldots, r_{k-1}\}$$

and so from Lemma 10.2.2,

$$p_k = r_{k-1} - [Ap_1, \ldots, Ap_{k-1}]z_{k-1} \in \mathrm{span}\{r_0, \ldots, r_{k-1}\}$$

It follows that

$$[p_1, \ldots, p_k] = [r_0, \ldots, r_{k-1}]T$$

for some upper triangular T. Since the search directions are independent, T is nonsingular. This shows

$$\mathrm{span}\{p_1, \ldots, p_k\} = \mathrm{span}\{r_0, \ldots, r_{k-1}\}.$$

Using (10.2.8) we see that

$$r_k \in \mathrm{span}\{r_{k-1}, Ap_k\} \subseteq \mathrm{span}\{r_{k-1}, Ar_0, \ldots, Ar_{k-1}\}.$$

The Krylov space connection in (10.2.10)follows from this by induction.

Finally, to establish the mutual orthogonality of the residuals, we note from (10.2.9) that r_k is orthogonal to any vector in the range of P_k. But from (10.2.10) this subspace contains $r_0, \ldots, r_{k-1}$. □

Using these facts we next show that p_k is a simple linear combination of its predecessor p_{k-1} and the "current" residual r_{k-1}.

Corollary 10.2.4 *The residuals and search directions in (10.2.7) have the property that $p_k \in \text{span}\{p_{k-1}, r_{k-1}\}$ for $k \geq 2$.*

Proof. If $k = 2$, then from (10.2.10) $p_2 \in \text{span}\{r_0, r_1\}$. But $p_1 = r_0$ and so p_2 is a linear combination of p_1 and r_1.

If $k > 2$, then partition the vector z_{k-1} of Lemma 10.2.2 as

$$z_{k-1} = \begin{array}{cc} \left[\begin{array}{c} w \\ \mu \end{array}\right] & \begin{array}{c} k-2 \\ 1 \end{array} \end{array}.$$

Using the identity $r_{k-1} = r_{k-2} - \alpha_{k-1}Ap_{k-1}$, we see that

$$\begin{aligned} p_k &= r_{k-1} - AP_{k-1}z_{k-1} = r_{k-1} - AP_{k-2}w - \mu Ap_{k-1} \\ &= \left(1 + \frac{\mu}{\alpha_{k-1}}\right) r_{k-1} + s_{k-1} \end{aligned}$$

where

$$\begin{aligned} s_{k-1} &= -\frac{\mu}{\alpha_{k-1}} r_{k-2} - AP_{k-2}w \\ &\in \text{span}\{r_{k-2}, AP_{k-2}w\} \\ &\subseteq \text{span}\{r_{k-2}, Ap_1, \ldots, Ap_{k-2}\} \\ &\subseteq \text{span}\{r_1, \ldots, r_{k-2}\} \end{aligned}$$

Because the r_i are mutually orthogonal, it follows that s_{k-1} and r_{k-1} are orthogonal to each other. Thus, the least squares problem of Lemma 10.2.2 boils down to choosing μ and w such that

$$\| p_k \|_2^2 = \left(1 + \frac{\mu}{\alpha_{k-1}}\right)^2 \| r_{k-1} \|_2^2 + \| s_{k-1} \|_2^2$$

is minimum. Since the 2-norm of $r_{k-2} - AP_{k-2}z$ is minimized by z_{k-2} giving residual p_{k-1}, it follows that s_{k-1} is a multiple of p_{k-1}. Consequently, $p_k \in \text{span}\{r_{k-1}, p_{k-1}\}$. □

We are now set to derive a very simple expression for p_k. Without loss of generality we may assume from Corollary 10.2.4 that

$$p_k = r_{k-1} + \beta_k p_{k-1}.$$

Since $p_{k-1}^T Ap_k = 0$ it follows that

$$\beta_k = -\frac{p_{k-1}^T A r_{k-1}}{p_{k-1}^T A p_{k-1}}$$

This leads us to "version 1" of the conjugate gradient method:

```
$x_0 =$ initial guess
$k = 0$
$r_0 = b - Ax_0$
while $r_k \neq 0$
    $k = k + 1$
    if $k = 1$
        $p_1 = r_0$
    else
        $\beta_k = -p_{k-1}^T A r_{k-1} / p_{k-1}^T A p_{k-1}$
        $p_k = r_{k-1} + \beta_k p_{k-1}$
    end                                                    (10.2.11)
    $\alpha_k = p_k^T r_{k-1} / p_k^T A p_k$
    $x_k = x_{k-1} + \alpha_k p_k$
    $r_k = b - Ax_k$
end
$x = x_k$
```

In this implementation, the method requires three separate matrix-vector multiplications per step. However, by computing residuals recursively via $r_k = r_{k-1} - \alpha_k Ap_k$ and substituting

$$r_{k-1}^T r_{k-1} = -\alpha_{k-1} r_{k-1}^T A p_{k-1} \qquad (10.2.12)$$

and

$$r_{k-2}^T r_{k-2} = \alpha_{k-1} p_{k-1}^T A p_{k-1} \qquad (10.2.13)$$

into the formula for β_k, we obtain the following more efficient version:

Algorithm 10.2.1 [Conjugate Gradients] If $A \in \mathbb{R}^{n \times n}$ is symmetric positive definite, $b \in \mathbb{R}^n$, and $x_0 \in \mathbb{R}^n$ is an initial guess ($Ax_0 \approx b$), then the following algorithm computes $x \in \mathbb{R}^n$ so $Ax = b$.

```
$k = 0$
$r_0 = b - Ax_0$
while $r_k \neq 0$
    $k = k + 1$
    if $k = 1$
        $p_1 = r_0$
    else
        $\beta_k = r_{k-1}^T r_{k-1} / r_{k-2}^T r_{k-2}$
        $p_k = r_{k-1} + \beta_k p_{k-1}$
    end
    $\alpha_k = r_{k-1}^T r_{k-1} / p_k^T A p_k$
    $x_k = x_{k-1} + \alpha_k p_k$
    $r_k = r_{k-1} - \alpha_k A p_k$
end
$x = x_k$
```

This procedure is essentially the form of the conjugate gradient algorithm that appears in the original paper by Hestenes and Stiefel (1952). Note that only one matrix-vector multiplication is required per iteration.

10.2.5 The Lanczos Connection

In §9.3.1 we derived the conjugate gradient method from the Lanczos algorithm. Now let us look at the connections between these two algorithms in the reverse direction by "deriving" the Lanczos process from conjugate gradients. Define the matrix of residuals $R_k \in \mathbb{R}^{n\times k}$ by

$$R_k = [\, r_0, \ldots, r_{k-1}\,]$$

and the upper bidiagonal matrix $B_k \in \mathbb{R}^{k\times k}$ by

$$B_k = \begin{bmatrix} 1 & -\beta_2 & 0 & \cdots & 0 \\ 0 & 1 & -\beta_3 & & \vdots \\ & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & -\beta_k \\ 0 & \cdots & & 0 & 1 \end{bmatrix}.$$

From the equations $p_i = r_{i-1} + \beta_i p_{i-1}$, $i = 2{:}k$, and $p_1 = r_0$ it follows that $R_k = P_k B_k$. Since the columns of $P_k = [\, p_1, \ldots, p_k\,]$ are A-conjugate, we see that $R_k^T A R_k \;=\; B_k^T \text{diag}(p_1^T A p_1, \ldots, p_k^T A p_k) B_k$ is tridiagonal. From (10.2.10) it follows that if

$$\Delta \;=\; \text{diag}(\rho_0, \ldots, \rho_{k-1}) \qquad \rho_i = \| r_i \|_2$$

then the columns of $R_k \Delta^{-1}$ form an orthonormal basis for the subspace $\text{span}\{r_0, Ar_0, \ldots, A^{k-1} r_0\}$. Consequently, the columns of this matrix are essentially the Lanczos vectors of Algorithm 9.3.1, i.e.,

$$q_i = \pm r_{i-1}/\rho_{i-1} \qquad i = 1{:}k.$$

Moreover, the tridiagonal matrix associated with these Lanczos vectors is given by

$$T_k \;=\; \Delta^{-1} B_k^T \text{diag}(p_i^T A p_i) B_k \Delta^{-1}. \tag{10.2.14}$$

The diagonal and subdiagonal of this matrix involve quantities that are readily available during the conjugate gradient iteration. Thus, we can obtain good estimates of A's extremal eigenvalues (and condition number) as we generate the x_k in Algorithm 10.2.1.

10.2.6 Some Practical Details

The termination criteria in Algorithm 10.2.1 is unrealistic. Rounding errors lead to a loss of orthogonality among the residuals and finite termination is not mathematically guaranteed. Moreover, when the conjugate gradient method is applied, n is usually so big that $O(n)$ iterations represents an unacceptable amount of work. As a consequence of these observations, it is customary to regard the method as a genuinely iterative technique with termination based upon an iteration maximum k_{max} and the residual norm. This leads to the following practical version of Algorithm 10.2.1:

```
$x$ = initial guess
$k = 0$
$r = b - Ax_0$
$\rho_0 = \| r \|_2^2$
while ( $\sqrt{\rho_k} > \epsilon \| b \|_2$ ) $\wedge$ ( $k < k_{max}$ )
    $k = k + 1$
    if $k = 1$
        $p = r$
    else
        $\beta_k = \rho_{k-1}/\rho_{k-2}$                    (10.2.16)
        $p = r + \beta_k p$
    end
    $w = Ap$
    $\alpha_k = \rho_{k-1}/p^T w$
    $x = x + \alpha_k p$
    $r = r - \alpha_k w$
    $\rho_k = \| r \|_2^2$
end
```

This algorithm requires one matrix-vector multiplication and $10n$ flops per iteration. Notice that just four n-vectors of storage are essential: x, r, p, and w. The subscripting of the scalars is not necessary and is only done here to facilitate comparison with Algorithm 10.2.1.

It is also possible to base the termination criteria on heuristic estimates of the error $A^{-1} r_k$ by approximating $\| A^{-1} \|_2$ with the reciprocal of the smallest eigenvalue of the tridiagonal matrix T_k given in (10.2.14).

The idea of regarding conjugate gradients as an iterative method began with Reid (1971). The iterative point of view is useful but then the *rate* of convergence is central to the method's success.

10.2.7 Convergence Properties

We conclude this section by examining the convergence of the conjugate gradient iterates $\{x_k\}$. Two results are given and they both say that the method performs well when A is near the identity either in the sense of a low rank perturbation or in the sense of norm.

Theorem 10.2.5 *If $A = I + B$ is an n-by-n symmetric positive definite matrix and $\text{rank}(B) = r$ then Algorithm 10.2.1 converges in at most $r+1$ steps.*

Proof. The dimension of

$$\text{span}\{r_0, Ar_0, \ldots, A^{k-1}r_0\} = \text{span}\{r_0, Br_0, \ldots, B^{k-1}r_0\}$$

cannot exceed $r+1$. Since $p_1, \ldots, p_k$ span this subspace and are independent, the iteration cannot progress beyond $r+1$ steps. □

An important metatheorem follows from this result:

- If A is close to a rank r correction to the identity, then Algorithm 10.2.1 almost converges after $r+1$ steps.

We show how this heuristic can be exploited in the next section.

An error bound of a different flavor can be obtained in terms of the A-norm which we define as follows:

$$\| w \|_A = \sqrt{w^T A w}.$$

Theorem 10.2.6 *Suppose $A \in \mathbb{R}^{n \times n}$ is symmetric positive definite and $b \in \mathbb{R}$. If Algorithm 10.2.1 produces iterates $\{x_k\}$ and $\kappa = \kappa_2(A)$ then*

$$\| x - x_k \|_A \le 2 \| x - x_0 \|_A \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^k .$$

Proof. See Luenberger (1973, p.187). □

The accuracy of the $\{x_k\}$ is often much better than this theorem predicts. However, a heuristic version of Theorem 10.2.6 turns out to be very useful:

- The conjugate gradient method converges very fast in the A-norm if $\kappa_2(A) \approx 1$.

In the next section we show how we can sometimes convert a given $Ax = b$ problem into a related $\tilde{A}\tilde{x} = \tilde{b}$ problem with $\tilde{A}$ being close to the identity.

Problems

P10.2.1 Verify that the residuals in (10.2.1) satisfy $r_i^T r_j = 0$ whenever $j = i + 1$.

P10.2.2 Verify (10.2.2).

P10.2.3 Verify (10.2.3).

P10.2.4 Verify (10.2.12) and (10.2.13).

P10.2.5 Give formula for the entries of the tridiagonal matrix T_k in (10.2.14).

P10.2.6 Compare the work and storage requirements associated with the practical implementation of Algorithms 9.3.1 and 10.2.1.

P10.2.7 Show that if $A \in \mathbb{R}^{n \times n}$ is symmetric positive definite and has k distinct eigenvalues, then the conjugate gradient method does not require more than $k+1$ steps to converge.

P10.2.8 Use Theorem 10.2.5 to verify that

$$\| x_k - A^{-1}b \|_2 \le 2\sqrt{\kappa} \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^k \| x_0 - A^{-1}b \|_2.$$

Notes and References for Sec. 10.2

The conjugate gradient method is a member of a larger class of methods that are referred to as *conjugate direction* algorithms. In a conjugate direction algorithm the search directions are all B-conjugate for some suitably chosen matrix B. A discussion of these methods appears in

J.E. Dennis Jr. and K. Turner (1987). "Generalized Conjugate Directions," *Lin. Alg. and Its Applic. 88/89*, 187–209.

G.W. Stewart (1973). "Conjugate Direction Methods for Solving Systems of Linear Equations," *Numer. Math. 21*, 284–97.

Some historical and unifying perspectives are offered in

G. Golub and D. O'Leary (1989). "Some History of the Conjugate Gradient and Lanczos Methods," *SIAM Review 31*, 50–102.

M.R. Hestenes (1990). "Conjugacy and Gradients," in *A History of Scientific Computing*, Addison-Wesley, Reading, MA.

S. Ashby, T.A. Manteuffel, and P.E. Saylor (1992). "A Taxonomy for Conjugate Gradient Methods," *SIAM J. Numer. Anal. 27*, 1542–1568.

The classic reference for the conjugate gradient method is

M.R. Hestenes and E. Stiefel (1952). "Methods of Conjugate Gradients for Solving Linear Systems," *J. Res. Nat. Bur. Stand. 49*, 409–36.

An exact arithmetic analysis of the method may be found in chapter 2 of

M.R. Hestenes (1980). *Conjugate Direction Methods in Optimization*, Springer-Verlag, Berlin.

See also

O. Axelsson (1977). "Solution of Linear Systems of Equations: Iterative Methods," in *Sparse Matrix Techniques: Copenhagen, 1976*, ed. V.A. Barker, Springer-Verlag, Berlin.

For a discussion of conjugate gradient convergence behavior, see

D. G. Luenberger (1973). *Introduction to Linear and Nonlinear Programming*, Addison-Wesley, New York.

A. van der Sluis and H.A. Van Der Vorst (1986). "The Rate of Convergence of Conjugate Gradients," *Numer. Math. 48*, 543–560.