Preconditioned Conjugate Gradients, Radial Basis Functions and Toeplitz Matrices

B. J. C. Baxter
Department of Mathematics, Imperial College
London SW7 2BZ, England
b.baxter@ic.ac.uk
www.ma.ic.ac.uk/∼baxter

Radial basis functions provide highly useful and flexible interpolants to multivariate functions. Further, they are beginning to be used in the numerical solution of partial differential equations. Unfortunately, their construction requires the solution of a dense linear system. Therefore much attention has been given to iterative methods. In this paper, we present a highly efficient preconditioner for the conjugate gradient solution of the interpolation equations generated by gridded data. Thus our method applies to the corresponding Toeplitz matrices. The number of iterations required to achieve a given tolerance is independent of the number of variables.

1. Introduction

A radial basis function approximation has the form

$$s(x) = \sum_{j=1}^{n} y_j \, \varphi(\|x - x_j\|), \qquad x \in \mathbb{R}^d,$$

where $\varphi\colon [0,\infty) \to \mathbb{R}$ is some given function, $(y_j)_1^n$ are real coefficients, and the centres $(x_j)_1^n$ are points in $\mathbb{R}^d$; the norm $\|\cdot\|$ will be Euclidean throughout this study. For a wide class of functions φ, it is known that the interpolation matrix

$$A = (\varphi(\|x_j - x_k\|))_{j,k=1}^n$$

is invertible. This matrix is typically full, which fact has encouraged the study of iterative methods. For example, highly promising results have been published in the use of radial basis functions in collocation and Galerkin methods for the numerical solution of partial differential equations (see Franke and Schaback (1998) and Wendland (1999)), but direct solution limits their applicability to fairly small problems. The use of the preconditioned conjugate gradient algorithm was pioneered by Dyn, Levin and Rippa (1986), and some stunning results for scattered data were presented recently

in Faul and Powell (1999), although the rapid convergence described there is not fully understood. Therefore we study the highly structured case when the data form a finite regular grid. The conjugate gradient algorithm has been applied to Toeplitz matrices with some success; see, for instance, Chan and Strang (1989). However, since our matrices are usually *not* positive definite and often possess elements that grow away from the diagonal, the preconditioners of Chan and Strang (1989) are not suitable. However, the matrices have the property that their inverses tractable more tractable. Specifically, the detailed study of the spectra of the associated Toeplitz operators presented in Baxter (1992) and Baxter (1994) allows us to create highly efficient preconditioners by inverting relatively small finite sections of the bi-infinite symmetric Toeplitz operator, and this construct is also easily understood via Toeplitz theory.

Let n be a positive integer and let A_n be the symmetric Toeplitz matrix given by

$$A_n = (\varphi(j-k))_{j,k=-n}^{n}, \tag{1.1}$$

where $\varphi\colon \mathbb{R} \to \mathbb{R}$ is either a Gaussian ($\varphi(x) = \exp(-\lambda x^2)$ for some positive constant λ) or a multiquadric ($\varphi(x) = (x^2 + c^2)^{1/2}$ for some real constant c). In this paper we construct efficient preconditioners for the conjugate gradient solution of the linear system

$$A_n x = f, \qquad f \in \mathbb{R}^{2n+1}, \tag{1.2}$$

when φ is a Gaussian, or the augmented linear system

$$\begin{aligned} A_n x + e y &= f, & (1.3)\\ e^T x &= 0, & (1.4) \end{aligned}$$

when φ is a multiquadric. Here $e = [1, 1, \ldots, 1]^T \in \mathbb{R}^{2n+1}$ and $y \in \mathbb{R}$. Section 2 describes the construction for the Gaussian and Section 3 deals with the multiquadric. Of course, we exploit the Toeplitz structure of A_n to perform a matrix-vector multiplication in $\mathcal{O}(n \log n)$ operations whilst storing $\mathcal{O}(n)$ real numbers. Further, we shall see numerically that the number of iterations required to achieve a solution of (1.2) or (1.4) to within a given tolerance is independent of n. The MATLAB software used can be obtained from my homepage.

Our method applies to many other radial basis functions, such as the inverse multiquadric ($\varphi(x) = (x^2 + c^2)^{-1/2}$) and the thin plate spline ($\varphi(x) = x^2 \log|x|$). However, we concentrate on the Gaussian and the multiquadric because they exhibit most of the important features of our approach in a concrete setting. Similarly we treat the one-dimensional problem merely to avoid complication; the multidimensional case is a rather slight generaliza-

tion of this work. Let us remark that the analogue of (1.1) is the operator

$$A_n^{(d)} = (\varphi(j-k))_{j,k\in[-n,n]^d}, \tag{1.5}$$

and we shall still call $A_n^{(d)}$ a Toeplitz matrix. Moreover the matrix-vector multiplication

$$A_n^{(d)} x = \left(\sum_{k\in[-n,n]^d} \varphi(\|j-k\|) x_k \right)_{j\in[-n,n]^d}, \tag{1.6}$$

where $\|\cdot\|$ is the Euclidean norm and $x = (x_j)_{j\in[-n,n]^d}$, can still be calculated in $\mathcal{O}(N \log N)$ operations, where $N = (2n+1)^d$, requiring $\mathcal{O}(N)$ real numbers to be stored. This trick is a simple extension of the Toeplitz matrix-vector multiplication method when $d = 1$.

2. The Gaussian

It is well-known that the Gaussian generates a positive definite interpolation matrix, and its functional decay is so rapid that preconditioning the conjugate gradient algorithm is not necessary. However, it provides a useful model problem that we shall describe here before developing the ideas further in the following section.

Our treatment of the preconditioned conjugate gradient (PCG) method follows Section 10.3 of Golub and Van Loan (1989), and we begin with a general description. We let n be a positive integer and $A \in \mathbb{R}^{n\times n}$ be an arbitrary symmetric positive definite matrix. For any nonsingular symmetric matrix $P \in \mathbb{R}^{n\times n}$ and $b \in \mathbb{R}^n$ we can use the following iteration to solve the linear system $PAPx = Pb$.

Algorithm 2.1. Choose any x_0 in $\mathbb{R}^n$. Set $r_0 = Pb - PAPx_0$ and $d_0 = r_0$.
For $k = 0, 1, 2, \ldots$ do begin
$a_k = r_k^T r_k / d_k^T PAPd_k$
$x_{k+1} = x_k + a_k d_k$
$r_{k+1} = r_k - a_k PAPd_k$
$b_k = r_{k+1}^T r_{k+1} / r_k^T r_k$
$d_{k+1} = r_{k+1} + b_k d_k$
Stop if $\|r_{k+1}\|$ or $\|d_{k+1}\|$ is sufficiently small.
end.

In order to simplify Algorithm 2.1 define

$$C = P^2, \qquad \xi_k = Px_k, \qquad r_k = P\rho_k \qquad \text{and} \qquad \delta_k = Pd_k. \tag{2.1}$$

Substituting in Algorithm 2.1 we obtain the following method.

Algorithm 2.2. Choose any ξ_0 in $\mathbb{R}^n$. Set $\rho_0 = b - A\xi_0$, $\delta_0 = C\rho_0$.

For $k = 0, 1, 2, \ldots$ do begin
$a_k = \rho_k^T C \rho_k / \delta_k^T A \delta_k$
$\xi_{k+1} = \xi_k + a_k \delta_k$
$\rho_{k+1} = \rho_k - a_k A \delta_k$
$b_k = \rho_{k+1}^T C \rho_{k+1} / \rho_k^T C \rho_k$
$\delta_{k+1} = C \rho_{k+1} + b_k \delta_k$
Stop if $\|\rho_{k+1}\|$ or $\|\delta_{k+1}\|$ is sufficiently small.
end.

It is Algorithm 2.2 that we shall consider as our PCG method in this section, and we shall call C the preconditioner. We see that the only restriction on C is that it must be a symmetric positive definite matrix, but we observe that the spectrum of CA should consist of a small number of clusters, preferably one cluster concentrated at one. At this point, we also mention that the condition number of CA is not a reliable guide to the efficacy of our preconditioner. For example, consider the two cases when (i) CA has only two different eigenvalues, say 1 and $100,000$, and (ii) when CA has eigenvalues uniformly distributed in the interval $[1, 100]$. The former has the larger condition number but, in exact arithmetic, the answer will be achieved in two steps, whereas the number of steps can be as high as n in the latter case. Thus the term "preconditioner" is sometimes inappropriate, although its usage has become standard.

In this paper we concentrate on preconditioners for the Toeplitz matrices generated by radial basis function interpolation on a (finite) regular grid. Accordingly, we let A be the matrix A_n of (1.1) and let $\varphi(x) = \exp(-x^2)$. Thus A_n is positive definite and can be embedded in the bi-infinite symmetric Toeplitz matrix

$$A_\infty = (\varphi(j-k))_{j,k \in \mathcal{Z}}\,. \tag{2.2}$$

The classical theory of Toeplitz operators (see, for instance, Grenander and Szegő (1984)) and the work of Baxter (1994) provide the relations

$$\text{Sp } A_n \subset \text{Sp } A_\infty = [\sigma(\pi), \sigma(0)] \subset (0, \infty), \tag{2.3}$$

where σ is the symbol function

$$\sigma(\xi) = \sum_{k \in \mathcal{Z}} \hat{\varphi}(\xi + 2\pi k), \qquad \xi \in \mathbb{R}, \tag{2.4}$$

and Sp A_∞ denotes the spectrum of the operator A_∞. Further, Theorem 9 of Buhmann and Micchelli (1991) allows us to conclude that, for any fixed integers j and k, we have

$$\lim_{n \to \infty} (A_n^{-1})_{j,k} = (A_\infty^{-1})_{j,k}. \tag{2.5}$$

It was equations (2.3) and (2.5) which led us to investigate the possibility

of using some of the elements of A_n^{-1} for a relatively small value of n to construct preconditioners for A_N, where N may be much larger than n. Specifically, let us choose integers $0 < m \leq n$ and define the sequence

$$c_j = (A_n^{-1})_{j0}, \qquad j = -m, \ldots, m. \tag{2.6}$$

We now let C_N be the $(2N+1) \times (2N+1)$ banded symmetric Toeplitz matrix

$$C_N = \begin{pmatrix} c_0 & \cdots & c_m & & & \\ \vdots & \ddots & & \ddots & & \\ c_m & & & & & \\ & \ddots & & & & c_m \\ & & & & & \vdots \\ & & & c_m & \cdots & c_0 \end{pmatrix}. \tag{2.7}$$

We claim that, for sufficiently large m and n, C_N provides an excellent preconditioner when $A = A_N$ in Algorithm 2.2. Before discussing any theoretical motivation for this choice of preconditioner, we present an example. We let $n = 64$, $m = 9$ and $N = 32,768$. Constructing A_n and calculating the elements $\{(A_n^{-1})_{j0} : j = 0, 1, \ldots, m\}$ we find that

$$\begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_9 \end{pmatrix} = \begin{pmatrix} 1.4301 \times 10^{0} \\ -5.9563 \times 10^{-1} \\ 2.2265 \times 10^{-1} \\ -8.2083 \times 10^{-2} \\ 3.0205 \times 10^{-2} \\ -1.1112 \times 10^{-2} \\ 4.0880 \times 10^{-3} \\ -1.5039 \times 10^{-3} \\ 5.5325 \times 10^{-4} \\ -2.0353 \times 10^{-4} \end{pmatrix}. \tag{2.8}$$

Now C_N can be embedded in the bi-infinite Toeplitz matrix C_∞ defined by

$$(C_\infty)_{jk} = \begin{cases} c_{j-k}, & |j-k| \leq m, \\ 0, & |j-k| > m, \end{cases} \tag{2.9}$$

and the symbol for this operator is the trigonometric polynomial

$$\sigma_{C_\infty}(\xi) = \sum_{j=-m}^{m} c_j e^{ij\xi}, \qquad \xi \in \mathbb{R}. \tag{2.10}$$

In Figure 2.1 we display a graph of σ_{C_∞} for $0 \leq \xi \leq 2\pi$, and it is clearly a positive function. Thus the relations

$$\text{Sp } C_N \subset \text{Sp } C_\infty = \{\sigma_{C_\infty}(\xi) : \xi \in [0, 2\pi]\} \subset (0, \infty) \tag{2.11}$$

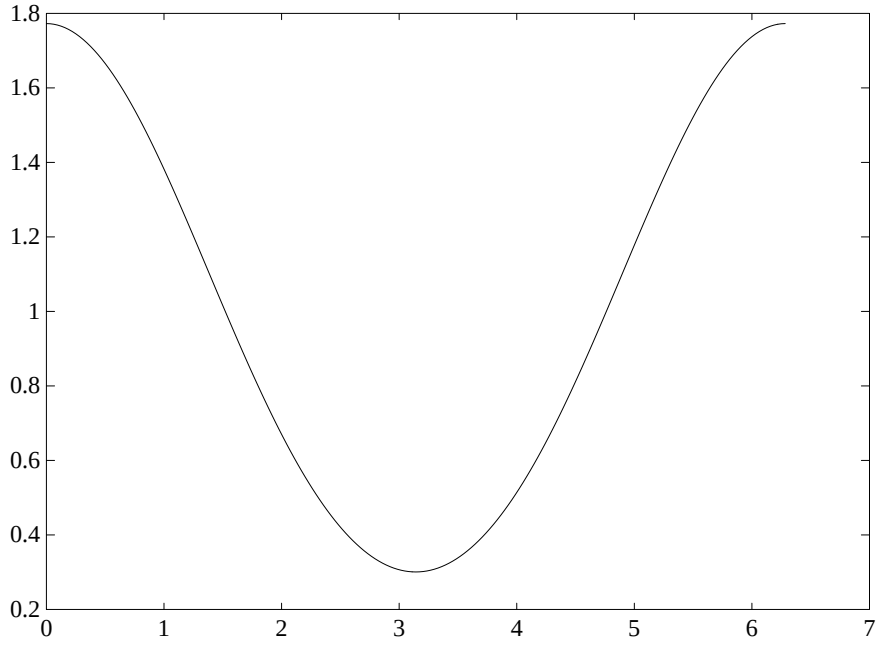

Fig. 2.1. The symbol function for C_∞.

imply that C_N is positive definite. Hence it is suitable to use C_N as the preconditioner in Algorithm 2.2. Our aim in this example is to compare this choice of preconditioner with the use of the identity matrix as the preconditioner. To this end, we let the elements of the vector b of Algorithm 2.2 be random real numbers uniformly distributed in the interval $[-1, 1]$. Applying Algorithm 2.2 using the identity matrix as the preconditioner provides the results of Table 2.1. Table 2.2 contains the analogous results using (2.7) and (2.8). In both cases the iterations were stopped when the residual vector satisfied the bound $\|r_{k+1}\|/\|b\| < 10^{-13}$. The behaviour shown in the tables is typical; we find that the number of steps required is independent of N and b.

Why should (2.7) and (2.8) provide a good preconditioner? Let us consider the bi-infinite Toeplitz matrix $C_\infty A_\infty$. The spectrum of this operator is given by

$$\text{Sp } C_\infty A_\infty = \{\sigma_{C_\infty}(\xi)\sigma(\xi) : \xi \in [0, 2\pi]\}, \tag{2.12}$$

where σ is given by (2.4) and σ_{C_∞} by (2.10). Therefore in order to concen-

Table 2.1. *No preconditioning*

Iteration	Error
1	2.797904×10^{1}
10	1.214777×10^{-2}
20	1.886333×10^{-6}
30	2.945903×10^{-10}
33	2.144110×10^{-11}
34	8.935534×10^{-12}

Table 2.2. *Using (2.7) and (2.8) as the preconditioner*

Iteration	Error
1	2.315776×10^{-1}
2	1.915017×10^{-3}
3	1.514617×10^{-7}
4	1.365228×10^{-11}
5	1.716123×10^{-15}

trate Sp $C_\infty A_\infty$ at unity we must have

$$\sigma_{C_\infty}(\xi)\sigma(\xi) \approx 1, \qquad \xi \in [0, 2\pi]. \tag{2.13}$$

In other words, we want σ_{C_∞} to be a trigonometric polynomial approximating the continuous function $1/\sigma$. Now if the Fourier series of $1/\sigma$ is given by

$$\sigma^{-1}(\xi) = \sum_{j \in \mathcal{Z}} \gamma_j e^{ij\xi}, \qquad \xi \in \mathbb{R}, \tag{2.14}$$

then its Fourier coefficients $(\gamma_j)_{j \in \mathcal{Z}}$ are the coefficients of the cardinal function χ for the integer grid, that is

$$\chi(x) = \sum_{j \in \mathcal{Z}} \gamma_j \varphi(x - j), \qquad x \in \mathbb{R}, \tag{2.15}$$

and

$$\chi(k) = \delta_{0k}, \qquad k \in \mathcal{Z}. \tag{2.16}$$

(See, for instance, Buhmann (1990).) Recalling (2.5), we deduce that one way to calculate approximate values of the coefficients $(\gamma_j)_{j \in \mathcal{Z}}$ is to solve

the linear system

$$A_n c^{(n)} = e^0, \tag{2.17}$$

where $e^0 = (\delta_{j0})_{j=-n}^{n} \in \mathbb{R}^{2n+1}$. We now set

$$c_j = c_j^{(n)}, \qquad 0 \le j \le m, \tag{2.18}$$

and we observe that the symbol function σ for the Gaussian is a theta function (see Baxter (1994), Section 2). Thus σ is a positive continuous function whose Fourier series is absolutely convergent. Hence $1/\sigma$ is a positive continuous function and Wiener's lemma (Rudin (1973)) implies the absolute convergence, and therefore the uniform convergence, of its Fourier series. We deduce that the symbol function σ_{C_∞} can be chosen to approximate $1/\sigma$ to within any required accuracy. More formally we have the

Lemma 2.1. Given any $\epsilon > 0$, there are positive integers m and n_0 such that

$$\left|\sigma(\xi) \sum_{j=-m}^{m} c_j^{(n)} e^{ij\xi} - 1\right| \le \epsilon, \qquad \xi \in [0, 2\pi], \tag{2.19}$$

for every $n \ge n_0$, where $c^{(n)} = (c_j^{(n)})_{j=-n}^{n}$ is given by (2.17).

Proof. The uniform convergence of the Fourier series for σ^{-1} implies that we can choose m such that

$$\left|\sigma(\xi) \sum_{j=-m}^{m} \gamma_j e^{ij\xi} - 1\right| \le \epsilon, \qquad \xi \in [0, 2\pi]. \tag{2.20}$$

By (2.5), we can also choose n_0 such that $\max\{|\gamma_j - c_j^{(n)}| : j = -m, \dots, m\} \le \epsilon$, when $n \ge n_0$. Then we have

$$\begin{aligned} &\left|\sigma(\xi) \sum_{j=-m}^{m} c_j^{(n)} e^{ij\xi} - 1\right| \\ &\le \left|\sigma(\xi) \sum_{j=-m}^{m} \gamma_j e^{ij\xi} - 1\right| + \left|\sigma(\xi) \sum_{j=-m}^{m} (\gamma_j - c_j^{(n)}) e^{ij\xi}\right| \\ &\le \epsilon[1 + (2m+1)\|\sigma\|_\infty]. \end{aligned} \tag{2.21}$$

Since ϵ is arbitrary the proof is complete. □

3. The Multiquadric

The multiquadric interpolation matrix

$$A = \Big(\varphi(\|x_j - x_k\|)\Big)_{j,k=1}^{n}, \tag{3.1}$$

where $\varphi(r) = (r^2 + c^2)^{1/2}$ and $(x_j)_{j=1}^n$ are points in $\mathbb{R}^d$, is not positive definite. In Micchelli (1986), it was shown to be *almost negative definite*, that is for any real numbers $(y_j)_{j=1}^n$ satisfying $\sum y_j = 0$ we have

$$\sum_{j,k=1}^{n} y_j y_k \varphi(\|x_j - x_k\|) \leq 0. \tag{3.2}$$

Furthermore, inequality (3.2) is strict when $n \geq 2$, the points $(x_j)_{j=1}^n$ are all different, and $\sum |y_j| > 0$. In other words, A is negative definite on the subspace $\langle e \rangle^\perp$, where $e = [1, 1, \ldots, 1]^T \in \mathbb{R}^n$.

Of course we cannot apply Algorithms 2.1 and 2.2 in this case. However, we can use the almost negative definiteness of A to solve a closely related linearly constrained quadratic programming problem:

$$\begin{aligned} &\text{minimize} \quad \frac{1}{2}\xi^T A\xi - b^T\xi \\ &\text{subject to} \quad e^T\xi = 0, \end{aligned} \tag{3.3}$$

where b can be any element of $\mathbb{R}^n$. Standard theory of Lagrange multipliers guarantees the existence of a unique pair of vectors $\xi^* \in \mathbb{R}^n$ and $\eta^* \in \mathbb{R}^m$ satisfying the equations

$$\begin{aligned} A\xi^* + e\eta^* &= b, \\ e^T\xi^* &= 0, \end{aligned} \tag{3.4}$$

where η^* is the Lagrange multiplier vector for the constrained optimization problem (3.3). We do not go into further detail on this point because the nonsingularity of the matrix

$$\begin{pmatrix} A & e \\ e^T & 0 \end{pmatrix} \tag{3.5}$$

is well-known (see, for instance, Powell (1990)). Instead we observe that one way to solve (3.4) is to apply the following modification of Algorithm 2.1 to (3.3).

Algorithm 3.1. Let P be any symmetric $n \times n$ matrix such that $\ker P = \langle e \rangle$.

Set $x_0 = 0$, $r_0 = Pb - PAPx_0$, $d_0 = r_0$.

For $k = 0, 1, 2, \ldots$ do begin

$a_k = r_k^T r_k / d_k^T PAPd_k$

$x_{k+1} = x_k + a_k d_k$

$r_{k+1} = r_k - a_k PAPd_k$

$b_k = r_{k+1}^T r_{k+1} / r_k^T r_k$

$d_{k+1} = r_{k+1} + b_k d_k$

Stop if $\|r_{k+1}\|$ or $\|d_{k+1}\|$ is sufficiently small.
end.

We observe that Algorithm 3.1 solves the linearly constrained optimization problem

$$\begin{aligned} &\text{minimize} && \frac{1}{2}x^T PAPx - b^T Px \\ &\text{subject to} && e^T x = 0. \end{aligned} \tag{3.6}$$

Moreover, the following elementary lemma implies that the solutions $\xi^*$of (3.4) and x^* of (3.6) are related by the equations $\xi^* = Px^*$.

Lemma 3.1. Let S be any symmetric $n \times n$ matrix and let $K = \ker S$. Then $S : K^\perp \to K^\perp$ is a bijection. In other words, given any $b \in K^\perp$ there is precisely one $a \in K^\perp$ such that

$$Sa = b. \tag{3.7}$$

Proof. For any $n \times n$ matrix M we have the equation

$$\mathbb{R}^n = \ker M \oplus \text{Im } M^T.$$

Consequently the symmetric matrix S satisfies

$$\mathbb{R}^n = \ker S \oplus \text{Im } S,$$

whence Im $S = K^\perp$. Hence for every $b \in K^\perp$ there exists $\alpha \in \mathbb{R}^n$ such that $S\alpha = b$. Now we can write $\alpha = a + \beta$, where $a \in K^\perp$ and $\beta \in K$ are uniquely determined by α. Thus $Sa = S\alpha = b$, and (3.7) has a solution. If $a' \in K^\perp$ also satifies (3.7), then their difference $a - a'$ lies in the intersection $K \cap K^\perp = \{0\}$, which settles the uniqueness of a. □

Setting $P = S$ and $K = \langle e \rangle$ in Lemma 3.1 we deduce that there is exactly one $x^* \in \langle e \rangle^\perp$ such that

$$PAPx^* = Pb,$$

and PAP is negative definite when restricted to the subspace $\langle e \rangle^\perp$. Following the development of Section 2, we define

$$C = P^2, \qquad \xi_k = Px_k, \qquad \text{and} \qquad \delta_k = Pd_k, \tag{3.8}$$

as in equation (2.1). However, we cannot define ρ_k by (2.1) because P is singular. One solution, advocated by Dyn, Levin and Rippa (1986), is to use the recurrence for (ρ_k) embodied in Algorithm 2.1 without further ado.

Algorithm 3.2. Choose any ξ_0 in $\langle e \rangle^\perp$. Set $\rho_0 = b - A\xi_0$ and $\delta_0 = C\rho_0$.

For $k = 0, 1, 2, \ldots$ do begin
$a_k = \rho_k^T C\rho_k / \delta_k^T A\delta_k$
$\xi_{k+1} = \xi_k + a_k\delta_k$
$\rho_{k+1} = \rho_k - a_k A\delta_k$
$b_k = \rho_{k+1}^T C\rho_{k+1} / \rho_k^T C\rho_k$
$\delta_{k+1} = C\rho_{k+1} + b_k\delta_k$
Stop if $\|\rho_{k+1}\|$ or $\|\delta_{k+1}\|$ is sufficiently small.
end.

However this algorithm is unstable in finite precision arithmetic, as we shall see in our main example below. One modification that successfully avoids instability is to force the condition

$$\rho_k \in \langle e\rangle^\perp \tag{3.9}$$

to hold for all k. Now Lemma 3.1 implies the existence of exactly one vector $\rho_k \in \langle e\rangle^\perp$ for which $P\rho_k = r_k$. Therefore, defining Q to be the orthogonal projection onto $\langle e\rangle^\perp$, that is $Q : x \mapsto x - e(e^T x)/(e^T e)$, we obtain

Algorithm 3.3. Choose any ξ_0 in $\langle e\rangle^\perp$. Set $\rho_0 = Q(b - A\xi_0)$, $\delta_0 = C\rho_0$.
For $k = 0, 1, 2, \ldots$ do begin
$a_k = \rho_k^T C\rho_k / \delta_k^T A\delta_k$
$\xi_{k+1} = \xi_k + a_k\delta_k$
$\rho_{k+1} = Q(\rho_k - a_k A\delta_k)$
$b_k = \rho_{k+1}^T C\rho_{k+1} / \rho_k^T C\rho_k$
$\delta_{k+1} = C\rho_{k+1} + b_k\delta_k$
Stop if $\|\rho_{k+1}\|$ or $\|\delta_{k+1}\|$ is sufficiently small.
end.

We see that the only restriction on C is that it must be a non-negative definite symmetric matrix such that $\ker C = \langle e\rangle$. It is easy to construct such a matrix given a positive definite symmetric matrix D by a rank one modification:

$$C = D - \frac{(De)(De)^T}{e^T De}. \tag{3.10}$$

The Cauchy-Schwarz inequality implies that $x^T Cx \geq 0$ with equality if and only if $x \in \langle e\rangle$. Of course we do not need to form C explicitly, since $C : x \mapsto Dx - (e^T Dx/e^T De)De$. Before constructing D we consider the spectral properties of $A_\infty = (\varphi(j-k))_{j,k\in\mathcal{Z}}$ in more detail.

A minor modification to Proposition 5.2.2 of Baxter (1992) yields the following useful result. Let us say that a complex sequence $(y_j)_{\mathcal{Z}}$ is *zero-summing* if it is finitely supported and satisfies $\sum y_j = 0$. The *symbol function*

$$\sigma(\xi) = \sum_{k\in\mathcal{Z}} \hat{\varphi}(\xi + 2\pi k), \qquad \xi \in \mathbb{R}, \tag{3.11}$$

now requires the distributional Fourier transform of the multiquadric. In the univariate case, this is given by

$$\hat{\varphi}(\xi) = -(2c/|\xi|)K_1(c|\xi|), \qquad \xi \in \mathbb{R} \setminus \{0\}, \tag{3.12}$$

where K_1 is a modified Bessel function. The symbol function is studied extensively in Baxter (1994).

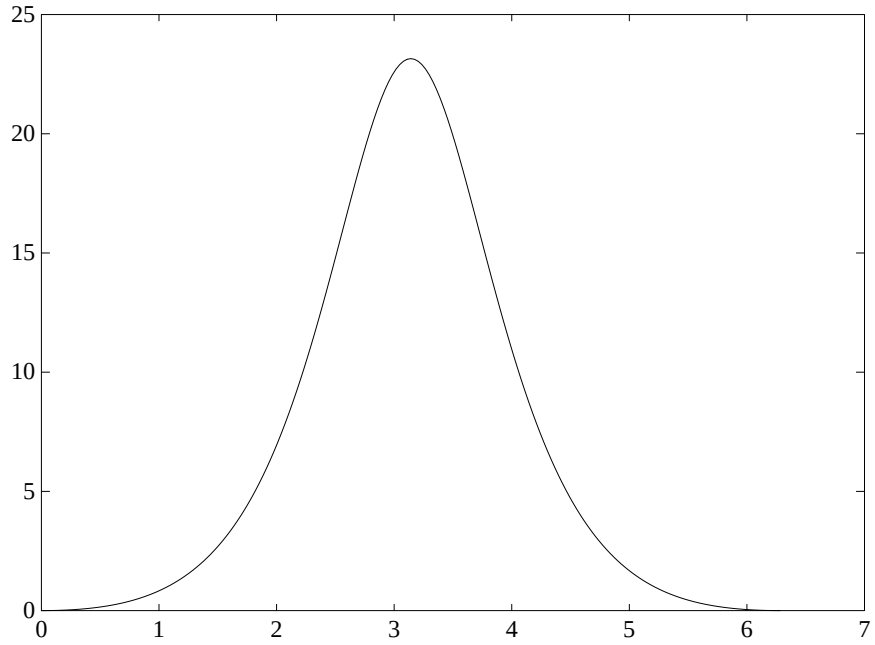

Fig. 3.2. The reciprocal symbol function $1/\sigma$ for the multiquadric.

Proposition 3.2. For every $\eta \in (0, 2\pi)$ we can find a set $\{(y_j^{(n)})_{j\in\mathcal{Z}} : n = 1, 2, \ldots\}$ of zero-summing sequences such that

$$\lim_{n\to\infty} \sum_{j,k\in\mathcal{Z}} y_j^{(n)} \overline{y_k^{(n)}} \varphi(j-k) \Big/ \sum_{j\in\mathcal{Z}} |y_j^{(n)}|^2 = \sigma(\eta). \tag{3.13}$$

Proof. We adopt the proof technique of Proposition 5.2.2 of Baxter (1992).

For each positive integer n we define the trigonometric polynomial

$$L_n(\xi) = n^{-1/2} \sum_{k=0}^{n-1} e^{ik\xi}, \qquad \xi \in \mathbb{R},$$

and we recall from Section 2 of Baxter (1994) that

$$K_n(\xi) = \frac{\sin^2 n\xi/2}{n \sin^2 \xi/2} = |L_n(\xi)|^2, \tag{3.14}$$

where K_n is the nth degree Fejér kernel. We now choose $(y_j^{(n)})_{j\in\mathcal{Z}}$ to be the Fourier coefficients of the trigonometric polynomial $\xi \mapsto L_n(\xi - \eta)\sin\xi/2$, which implies the relation

$$\left| \sum_{j\in\mathcal{Z}} y_j^{(n)} e^{ij\xi} \right|^2 = \sin^2 \xi/2 \; K_n(\xi - \eta),$$

and we see that $(y_j^{(n)})_{j\in\mathcal{Z}}$ is a zero-summing sequence. By the Parseval relation we have

$$\sum_{j\in\mathcal{Z}} |y_j^{(n)}|^2 = (2\pi)^{-1} \int_0^{2\pi} \sin^2 \xi/2 \; K_n(\xi - \eta)\, d\xi \tag{3.15}$$

and the approximate identity property of the Fejér kernel (Zygmund (1988), p. 86) implies that

$$\sin^2 \eta/2 = \lim_{n\to\infty} (2\pi)^{-1} \int_0^{2\pi} \sin^2 \xi/2 \; K_n(\xi - \eta)\, d\xi = \lim_{n\to\infty} \sum_{j\in\mathcal{Z}} |y_j^{(n)}|^2.$$

Further, because σ is continuous on $(0, 2\pi)$ (Baxter (1994), Section 4.4), we have

$$\begin{aligned} \sin^2 \eta/2 \; \sigma(\eta) &= \lim_{n\to\infty} (2\pi)^{-1} \int_0^{2\pi} \sin^2 \xi/2 \; K_n(\xi - \eta)\sigma(\xi)\, d\xi \\ &= \lim_{n\to\infty} \sum_{j,k\in\mathcal{Z}} y_j^{(n)} \overline{y_k^{(n)}} \varphi(j-k). \end{aligned}$$

□

Thus we have shown that, just as in the classical theory of Toeplitz operators (Grenander and Szegő (1984)), everything depends on the range of values of the symbol function σ. Because σ inherits the double pole that $\hat{\varphi}$ enjoys at zero, we have $\sigma\colon (0, 2\pi) \mapsto (\sigma(\pi), \infty)$. In Figure 3.2 we display the function σ^{-1}.

Now let m be a positive integer and let $(d_j)_{j=-m}^{m}$ be an even sequence of real numbers. We define a bi-infinite banded symmetric Toeplitz matrix

D_∞ by the equations

$$(D_\infty)_{jk} = \begin{cases} d_{j-k}, & |j-k| \le m, \\ 0, & \text{otherwise}. \end{cases} \tag{3.16}$$

Thus $(D_\infty A_\infty)_{jk} = \psi(j-k)$ where $\psi(x) = \sum_{l=-m}^{m} d_l \varphi(x-l)$. Further

$$\sum_{j,k\in\mathcal{Z}} y_j \overline{y_k} \psi(j-k) = (2\pi)^{-1} \int_0^{2\pi} \Big| \sum_{j\in\mathcal{Z}} y_j e^{ij\xi} \Big|^2 \sigma_{D_\infty}(\xi)\sigma(\xi)\, d\xi, \tag{3.17}$$

where the symbol function σ_{D_∞} for the Toeplitz operator D_∞ is given by

$$\sigma_{D_\infty}(\xi) = \sum_{j=-m}^{m} d_j e^{ij\xi}, \xi \in \mathbb{R}. \tag{3.18}$$

Now the function $\sigma\sigma_{D_\infty}$ is continuous for $\xi \in (0, 2\pi)$, so the argument of Proposition 3.2 also shows that, for every $\eta \in (0, 2\pi)$, we can find a set $\{(y_j^{(n)})_{j\in\mathcal{Z}} : n = 1, 2, \dots\}$ of zero-summing sequences such that

$$\lim_{n\to\infty} \frac{\sum_{j,k\in\mathcal{Z}} y_j^{(n)} \overline{y_k^{(n)}} \psi(j-k)}{\sum_{j\in\mathcal{Z}} |y_j^{(n)}|^2} = \sigma_{D_\infty}(\eta)\sigma(\eta). \tag{3.19}$$

A good preconditioner must ensure that $\{\sigma_{D_\infty}(\xi)\sigma(\xi) : \xi \in (0, 2\pi)\}$ is a bounded set. Because of the form of σ_{D_∞} we have the equation

$$\sum_{j=-m}^{m} d_j = 0. \tag{3.20}$$

Moreover, as in Section 2, we want the approximation

$$\sigma_{D_\infty}(\xi)\sigma(\xi) \approx 1, \qquad \xi \in (0, 2\pi), \tag{3.21}$$

and we need σ_{D_∞} to be a non-negative trigonometric polynomial which is positive almost everywhere, which ensures that every one of its principal minors is positive definite.

Let us define

$$c_j^{(n)} = -\left(A_n^{-1}\right)_{j0}, \qquad j = -m, \dots, m, \tag{3.22}$$

and

$$\sigma^{-1}(\xi) = \sum_{j\in\mathcal{Z}} \gamma_j e^{ij\xi}, \qquad \xi \in \mathbb{R}. \tag{3.23}$$

Then Theorem 9 of Buhmann and Micchelli (1991) states that

$$\lim_{n\to\infty} c_j^{(n)} = \gamma_j, \tag{3.24}$$

for any given fixed integer j. We shall use this fact to construct a suitable

Table 3.3. *Preconditioned CG – $m = 9$, $n = 64$, $N = 2,048$*

Iteration	Error
1	3.975553×10^{4}
2	8.703344×10^{-1}
3	2.463390×10^{-2}
4	8.741920×10^{-3}
5	3.650521×10^{-4}
6	5.029770×10^{-6}
7	1.204610×10^{-5}
8	1.141872×10^{-7}
9	1.872273×10^{-9}
10	1.197310×10^{-9}
11	3.103685×10^{-11}

Table 3.4. *Preconditioned CG – $m = 9$, $n = 64$, $N = 32,768$*

Iteration	Error
1	2.103778×10^{5}
2	4.287497×10^{0}
3	5.163441×10^{-1}
4	1.010665×10^{-1}
5	1.845113×10^{-3}
6	3.404016×10^{-3}
7	3.341912×10^{-5}
8	6.523212×10^{-7}
9	1.677274×10^{-5}
10	1.035225×10^{-8}
11	1.900395×10^{-10}

σ_{D_∞}. First we subtract a multiple of the vector $[1, \ldots, 1]^T \in \mathbb{R}^{2m+1}$ from $(c_j^{(n)})_{j=-m}^{m}$ to form a new vector $(d_j)_{j=-m}^{m}$ satisfying $\sum d_j = 0$, and we observe that, by (3.24), $\sigma_{D_\infty}(\xi)$ is one-signed for all sufficiently large values of n. For the numerical experiments here, we have chosen $n = 64$ and $m = 9$.

Thus, given

$$A_N = \Big(\varphi(j-k)\Big)_{j,k=-N}^{N}$$

for any $N \geq n$, we let D_N be any $(2N+1) \times (2N+1)$ principal submatrix

Table 3.5. *Preconditioned CG – $m = 1$, $n = 64$, $N = 8,192$*

Iteration	Error
1	2.645008×10^{4}
10	8.632419×10^{0}
20	9.210298×10^{-1}
30	7.695337×10^{-1}
40	3.187051×10^{-5}
50	5.061053×10^{-7}
60	7.596739×10^{-9}
70	1.200700×10^{-10}
73	3.539988×10^{-11}
74	1.992376×10^{-11}

of D_∞ and define the preconditioner C_N by the equation

$$C_N = D_N - \frac{(D_N e)(D_N e)^T}{e^T D_N e}, \tag{3.25}$$

where $e = [1, \ldots, 1]^T \in \mathbb{R}^{2N+1}$. We reiterate that we actually compute the matrix-vector product $C_N x$ by the operations $x \mapsto D_N x - (e^T D_N x / e^T D_N e)e$ rather than by storing the elements of C_N in memory.

C_N provides an excellent preconditioner. Tables 3.3 and 3.4 illustrate its use when Algorithm 3.3 is applied to the linear system

$$\begin{aligned} A_N x + ey &= b, \\ e^T x &= 0, \end{aligned} \tag{3.26}$$

when $N = 2,048$ and $N = 32,768$ respectively. Here $y \in \mathbb{R}$, $e = [1, \ldots, 1]^T \in \mathbb{R}^{2N+1}$ and $b \in \mathbb{R}^{2N+1}$ consists of pseudo-random real numbers uniformly distributed in the interval $[-1, 1]$. Again, this behaviour is typical and all our numerical experiments indicate that the number of steps is independent of N. We remind the reader that the error shown is $\|\rho_{k+1}\|$, but that the iterations are stopped when either $\|\rho_{k+1}\|$ or $\|\delta_{k+1}\|$ is less than $10^{-13}\|b\|$, where we are using the notation of Algorithm 3.3.

It is interesting to compare Table 3.3 with Table 3.5. Here we have chosen $m = 1$, and the preconditioner is essentially a multiple of the second divided difference preconditioner advocated by Dyn, Levin and Rippa (1986). Indeed, we find that $d_0 = 7.8538$ and $d_1 = d_{-1} = -3.9269$. We see that its behaviour is clearly inferior to the preconditioner generated by choosing $m = 9$. Furthermore, this is to be expected, because we are choosing a smaller finite section to approximate the reciprocal of the symbol function.

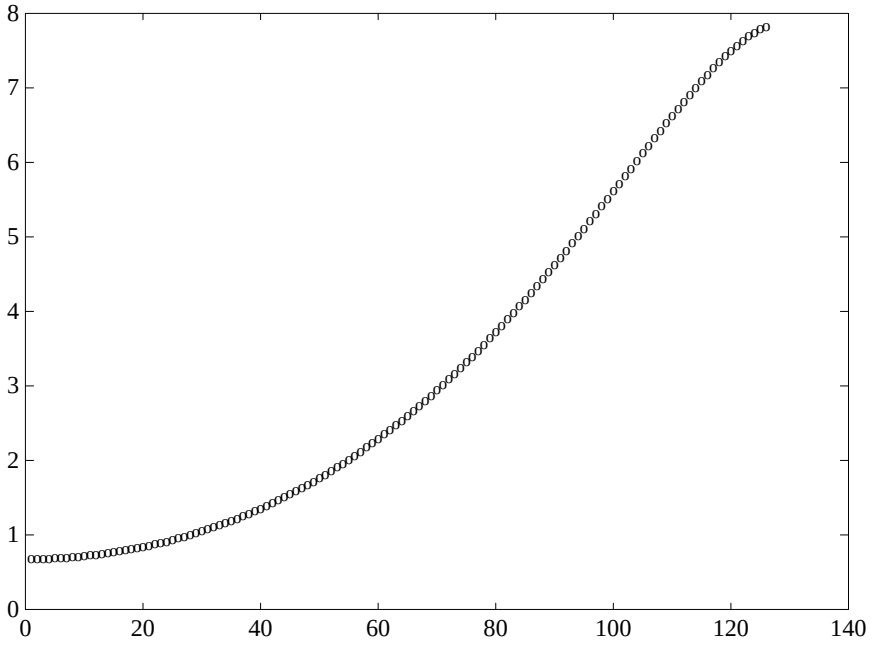

Fig. 3.3. The spectrum of $C_n A_n$ for $m = 1$ and $n = 64$.

However, because $\sigma_{D_\infty}(\xi)$ is a multiple of $\sin^2 \xi/2$, this preconditioner still possesses the property that $\{\sigma_{D_\infty}(\xi)\sigma(\xi) : \xi \in (0, 2\pi)\}$ is a bounded set of real numbers.

It is also interesting to compare the spectra of $C_n A_n$ for $n = 64$ and $m = 1$ and $m = 9$. Accordingly, Figures 3.3 and 3.4 display all but the largest nonzero eigenvalues of $C_n A_n$ for $m = 1$ and $m = 6$ respectively. The largest eigenvalues are 502.6097 and 288.1872, respectively, and these were omitted from the plots in order to reveal detail at smaller scales. We see that the clustering of the spectrum when $m = 9$ is excellent.

The final topic in this section demonstrates the instability of Algorithm 3.2 when compared with Algorithm 3.3. We refer the reader to Table 3.6, where we have chosen $m = 9$, $n = N = 64$, and setting $b = [1, 4, 9, \ldots, N^2]^T$. The iterations for Algorithm 3.3, displayed in Table 3.6, were stopped at iteration 108. For Algorithm 3.2, iterations were stopped when either $\|\rho_{k+1}\|$ or $\|\delta_{k+1}\|$ became smaller than $10^{-13}\|b\|$. It is useful to display the norm of $\|\delta_k\|$ rather than $\|\rho_k\|$ in this case. We see that the two algorithms almost agree on the early interations, but that Algorithm 3.2 soon begins

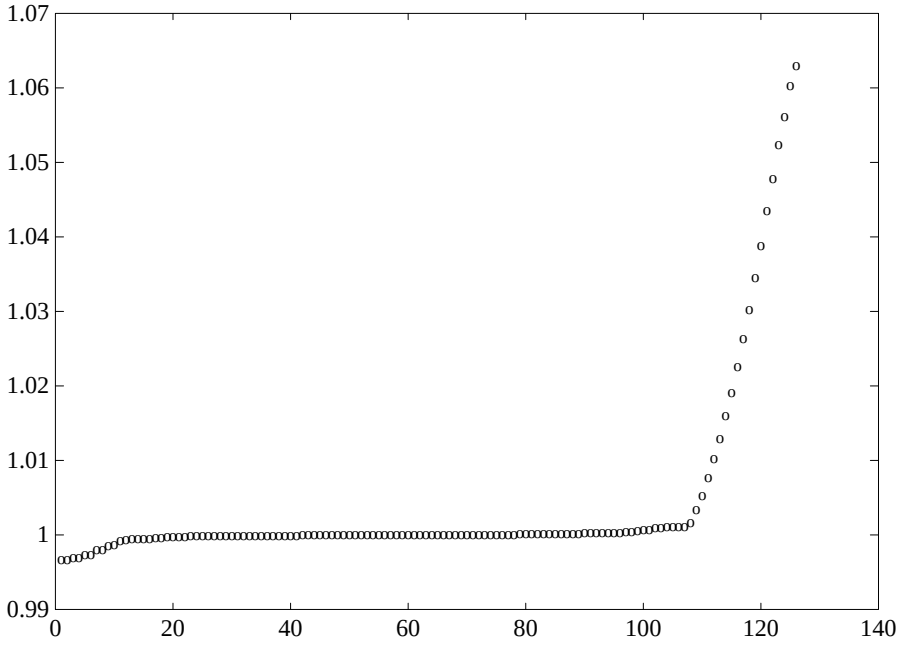

Fig. 3.4. The spectrum of $C_n A_n$ for $m = 9$ and $n = 64$.

cycling, and no convergence seems to occur. Thus when ρ_k can leave the required subspace due to finite precision arithmetic, it is possible to attain non-descent directions.

REFERENCES

B. J. C. Baxter (1992a), "On the asymptotic behaviour of the span of translates of the multiquadric $\varphi(r) = (r^2 + c^2)^{1/2}$ as $c \to \infty$", *Comput. Math. Applic.* **24**, 1–6.

B. J. C. Baxter (1992b), *The Interpolation Theory of Radial Basis Functions*, PhD thesis, University of Cambridge.

B. J. C. Baxter (1994), "Norm estimates for inverses of Toeplitz distance matrices", *J. Approx. Theory* **79**, 222–242.

M. D. Buhmann and C. A. Micchelli (1991), "Multiply monotone functions for cardinal interpolation", *Advances in Applied Mathematics* **12**, 358–386.

R. Chan and G. Strang (1989), "Toeplitz equations by conjugate gradients with circulant preconditioner", *SIAM J. Sci. Stat. Comp.* **10**, 104–119.

N. D. Dyn, D. Levin and S. Rippa (1986), "Numerical procedures for surface fitting of scattered data by radial functions", *SIAM J. Sci. Stat. Comput.* **7**, 639–659.

Table 3.6. *Algorithms 3.3a & b – $m = 1$, $n = 64$, $N = 64$, $b = [1, 4, \ldots, N^2]^T$.*

Iteration	$\|\delta_k\|$ – 3.3a	$\|\delta_k\|$ – 3.3b
1	4.436896×10^{4}	4.436896×10^{4}
2	2.083079×10^{2}	2.083079×10^{2}
3	2.339595×10^{0}	2.339595×10^{0}
4	1.206045×10^{-1}	1.206041×10^{-1}
5	1.698965×10^{-3}	1.597317×10^{-3}
6	6.537466×10^{-2}	6.512586×10^{-2}
7	1.879294×10^{-4}	9.254943×10^{-6}
8	2.767714×10^{-2}	1.984033×10^{-7}
9	3.453789×10^{-4}	
10	1.914126×10^{-3}	
20	4.628447×10^{-1}	
30	3.696474×10^{-0}	
40	$8.061922 \times 10^{+3}$	
50	2.155310×10^{0}	
100	3.374467×10^{-1}	

A. C. Faul and M. J. D. Powell, "Krylov subspace methods for radial basis function interpolation", DAMTP Report 1999/NA11, University of Cambridge.

C. Franke and R. Schaback (1998), "Solving partial differential equations by collocation using radial basis functions", *Appl. Math. Comp.* **93**, 73–82.

G. H. Golub and C. F. Van Loan (1989), *Matrix Computations*, The John Hopkins University Press (Baltimore).

U. Grenander and G. Szegő (1984), *Toeplitz Forms*, Chelsea (New York).

M. J. D. Powell (1992), "The theory of radial basis function approximation in 1990", in *Advances in Numerical Analysis II: Wavelets, Subdivision Algorithms and Radial Functions*, ed. W. A. Light, Oxford University Press (Oxford), 105–210.

W. Rudin (1973), *Functional Analysis*, McGraw Hill (New York).

H. Wendland (1999), *Meshless Galerkin methods using radial basis functions*, *Math. Comp.* **68**, 1521–1531.

A. Zygmund (1979), *Trigonometric Series*, Volumes I and II, Cambridge University Press (Cambridge).