Additive Schwarz preconditioners for interpolation of divergence-free vector fields on spheres

Q. T. Le Gia[1] T. Tran[2]

(Received 29 January 2011; revised 22 August 2011)

Abstract

The linear system arising from the interpolation problem of surface divergence-free vector fields using radial basis functions tends to be ill-conditioned when the separation radius of the scattered data is small. When the surface under consideration is the unit sphere, we introduce a preconditioner based on the additive Schwarz method to accelerate the solution process. Theoretical estimates for the condition number of the preconditioned matrix are given. Numerical experiments using scattered data from the MAGSAT satellite show the effectiveness of our preconditioner.

Contents

1 Introduction

Fitting divergence-free tangent vector fields to scattered data has applications in some important partial differential equations used in weather forecasting models. In the barotropic vorticity equation on the surface of the sphere, which provides a good model for 500 mb short term weather forecasts in mid-latitudes [2, pp. 108–110], the velocity is required to be surface divergence free. The nonlinear flow of an incompressible fluid in a single hydrostatic atmospheric layer is described by the shallow water wave equations on the surface of a rotating sphere. The incompressibility assumption gives rises to the constraint that the velocity field is surface divergence free.

Interpolation of divergence-free vector fields using radial basis functions (RBFs) was introduced recently by Narcowich et al. [5]. The interpolant is constructed from surface divergence-free RBFs, which handle scattered data effectively. To construct the interpolant, one needs to solve a linear system, which is often ill-conditioned when the separation radius of the scattered data set is small.

We introduce a preconditioner based on the additive Schwarz method to

accelerate the solution process. We developed preconditioners of a similar type for interpolation of scalar functions and pseudo-differential equations on the unit sphere [3, 4].

2 Divergence free RBFs on spheres

Suppose $\psi : S \times S \to \mathbb{R}$ is a strictly positive definite kernel on the unit sphere $S \subset \mathbb{R}^3$. That is, ψ is continuous, $\psi(\mathbf{x}, \mathbf{y}) = \psi(\mathbf{y}, \mathbf{x})$ for all $\mathbf{x}, \mathbf{y} \in S$, and for any set of distinct points $\{\mathbf{x}_1, \ldots, \mathbf{x}_M\}$ the $M \times M$ matrix $[\psi(\mathbf{x}_i, \mathbf{x}_j)]$ is positive definite. A convenient way to define ψ is via a radial basis function Φ as

$$\psi(\mathbf{x}, \mathbf{y}) = \Phi(\mathbf{x} - \mathbf{y}) = \phi(\|\mathbf{x} - \mathbf{y}\|), \quad \mathbf{x}, \mathbf{y} \in S, \tag{1}$$

where $\phi : [0, \infty) \to \mathbb{R}$ is a positive definite function [6]. Hence the kernel ψ is of the form

$$\psi(\mathbf{x}, \mathbf{y}) = \rho(\mathbf{x} \cdot \mathbf{y}) \quad \text{for } \mathbf{x}, \mathbf{y} \in S, \tag{2}$$

where $\rho : [-1, 1] \to \mathbb{R}$ is defined by $\rho(t) = \phi(\sqrt{2 - 2t})$.

Using the kernel ψ, we define the divergence free RBF as follows. First, we define

$$\Psi(\mathbf{x}, \mathbf{y}) = \mathrm{Curl}_{\mathbf{x}}[\mathrm{Curl}_{\mathbf{y}}\, \psi(\mathbf{x}, \mathbf{y})]^T, \tag{3}$$

in which $\mathrm{Curl}_{\mathbf{x}}$ and $\mathrm{Curl}_{\mathbf{y}}$ indicate the Curl operator with respect to variables $\mathbf{x}$ and $\mathbf{y}$, respectively. Here, $\mathrm{Curl} = \mathbf{n} \times \nabla^*$, where ∇^* denotes the surface gradient and $\mathbf{n}$ is the outward unit normal to S.

For fixed points $\mathbf{x}, \mathbf{y} \in S$, the matrix $\Psi(\mathbf{x}, \mathbf{y})$ represents a linear transformation which maps tangent vectors based at the point $\mathbf{y}$ to tangent vectors based at the point $\mathbf{x}$. In other words, if $\mathbf{t}_{\mathbf{y}}$ is a tangent vector belonging to the tangent space $TS_{\mathbf{y}}$, then $\Psi(\mathbf{x}, \mathbf{y})\mathbf{t}_{\mathbf{y}}$ is a tangent vector belonging to $TS_{\mathbf{x}}$. The kernel Ψ is surface divergence free [5]; that is, for a tangent vector $\mathbf{t}_{\mathbf{y}}$ based at a point $\mathbf{y} \in S$, there holds

$$\mathrm{Div}(\Psi(\mathbf{x}, \mathbf{y})\mathbf{t}_{\mathbf{y}}) = 0.$$

For a fixed $\mathbf{y} \in S$, the vector function $\Psi(\cdot, \mathbf{y})\mathbf{t}_{\mathbf{y}}$ is called a divergence-free RBF.

The kernel $\Psi(\mathbf{x}, \mathbf{y})$ is used to interpolate surface divergence free vector fields on the unit sphere. Suppose $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ is a set of scattered points on the unit sphere and $\mathbf{F} : S \to \mathbb{R}^3$ is a continuous surface divergence-free vector field whose values at the $\mathbf{x}_j$ are known. Then the interpolation problem is to find $I_X\mathbf{F}$ in

$$V_X := \left\{ \sum_{k=1}^{N} \Psi(\mathbf{x}, \mathbf{x}_k)\mathbf{s}_k : \mathbf{s}_k \in TS_{\mathbf{x}_k} \text{ and } \mathbf{x}_k \in X \right\} \tag{4}$$

so that

$$I_X\mathbf{F}(\mathbf{x}_j) = \mathbf{F}(\mathbf{x}_j) \quad \text{for } j = 1, \ldots, N\,. \tag{5}$$

By choosing an orthonormal basis $\{\boldsymbol{e}_k, \widetilde{\boldsymbol{e}}_k\}$ for the tangent space $TS_{\mathbf{x}_k}$ we write $I_X\mathbf{F}$ as

$$I_X\mathbf{F}(\mathbf{x}) = \sum_{k=1}^{N} \Psi(\mathbf{x}, \mathbf{x}_k)(c_k\boldsymbol{e}_k + \widetilde{c}_k\widetilde{\boldsymbol{e}}_k), \quad c_k, \widetilde{c}_k \in \mathbb{R}\,, \tag{6}$$

and $\mathbf{F}(\mathbf{x}_j)$ as

$$\mathbf{F}(\mathbf{x}_j) = d_j\boldsymbol{e}_j + \widetilde{d}_j\widetilde{\boldsymbol{e}}_j\,, \quad d_k, \widetilde{d}_k \in \mathbb{R}\,. \tag{7}$$

In particular, in the implementation we choose $\boldsymbol{e}_k = \boldsymbol{e}|_{x=x_k}$ and $\widetilde{\boldsymbol{e}}_k = \widetilde{\boldsymbol{e}}|_{x=x_k}$ where

$$\boldsymbol{e} = \frac{\partial \mathbf{p}}{\partial \theta}\,, \quad \widetilde{\boldsymbol{e}} = \frac{1}{\sin\theta}\frac{\partial \mathbf{p}}{\partial \phi} \quad \text{and} \quad \mathbf{p} = (\sin\theta\cos\phi, \sin\theta\sin\phi, \cos\theta)^T.$$

From (5), (6) and (7), we obtain

$$\sum_{k=1}^{N} \underbrace{\begin{pmatrix} \boldsymbol{e}_j^T \\ \widetilde{\boldsymbol{e}}_j^T \end{pmatrix} \Psi(\mathbf{x}_j, \mathbf{x}_k)(\boldsymbol{e}_k \quad \widetilde{\boldsymbol{e}}_k)}_{A_{jk}} \begin{pmatrix} c_k \\ \widetilde{c}_k \end{pmatrix} = \begin{pmatrix} d_j \\ \widetilde{d}_j \end{pmatrix}, \quad j = 1, \ldots, N\,. \tag{8}$$

Let $\mathbf{A}$ be the $2N \times 2N$ matrix composed of the blocks A_{jk} in (8), let $\mathbf{c} = (c_1, \widetilde{c}_1, \ldots, c_N, \widetilde{c}_N)^T$, and let $\mathbf{d} = (d_1, \widetilde{d}_1, \ldots, d_N, \widetilde{d}_N)^T$. Then the system (8) becomes

$$\mathbf{A}\mathbf{c} = \mathbf{d}\,. \tag{9}$$

A convenient way to compute $\Psi(\mathbf{x}_j, \mathbf{x}_k)$ in the entries of $\mathbf{A}$ is via the matrix Q_x satisfying $Q_x\mathbf{y} = \mathbf{x} \times \mathbf{y}$. Then $\mathrm{Curl}_x = Q_x\nabla$, and from (1) and (3) we obtain

$$\Psi(\mathbf{x}, \mathbf{y}) = Q_x(-\nabla\nabla^T\Phi(\mathbf{x} - \mathbf{y}))Q_y^T\,, \quad \mathbf{x}, \mathbf{y} \in S\,. \tag{10}$$

By (1), the jk-component of the Hessian matrix $\nabla\nabla^T\Phi(\mathbf{x})$ is

$$\frac{\partial^2\Phi}{\partial x_j \partial x_k} = \delta_{j,k}\frac{1}{r}\phi'(r) + x_j x_k \frac{1}{r}\left(\frac{1}{r}\phi'(r)\right)' =: \delta_{j,k}F(r) + x_j x_k G(r), \quad r = \|\mathbf{x}\|.$$

From this we have

$$\nabla\nabla^T\Phi(\mathbf{x} - \mathbf{y}) = F(r)I + G(r)(\mathbf{x} - \mathbf{y})(\mathbf{x} - \mathbf{y})^T, \quad r = \|\mathbf{x} - \mathbf{y}\|,$$

where I is the identity matrix. Thus

$$\Psi(\mathbf{x}, \mathbf{y}) = F(r)(\mathbf{y}\mathbf{x}^T - \mathbf{y}^T\mathbf{x}I) - G(r)(\mathbf{x} \times \mathbf{y})(\mathbf{x} \times \mathbf{y})^T. \tag{11}$$

Setting $\widetilde{\mathbf{s}} = \mathbf{x} \times \mathbf{s}$, from (11) we obtain

$$\Psi(\mathbf{x}, \mathbf{y})\mathbf{s} = \mathbf{x} \times \left[F(r)\widetilde{\mathbf{s}} + G(r)(\mathbf{x} - \mathbf{y})(\mathbf{x} - \mathbf{y})^T\widetilde{\mathbf{s}}\right]. \tag{12}$$

With $r_{jk} = \|\mathbf{x}_j - \mathbf{x}_k\|$, the 2×2 matrix A_{jk} from (8) is explicitly

$$A_{jk} = F(r_{jk})\begin{pmatrix} -\widetilde{\boldsymbol{e}}_j \cdot \widetilde{\boldsymbol{e}}_k & \widetilde{\boldsymbol{e}}_j \cdot \boldsymbol{e}_k \\ \boldsymbol{e}_j \cdot \widetilde{\boldsymbol{e}}_k & -\boldsymbol{e}_j \cdot \boldsymbol{e}_k \end{pmatrix} + G(r_{jk})\begin{pmatrix} \widetilde{\boldsymbol{e}}_j \cdot \mathbf{x}_k \\ -\boldsymbol{e}_j \cdot \mathbf{x}_k \end{pmatrix}(\mathbf{x}_j \cdot \widetilde{\boldsymbol{e}}_k \quad -\mathbf{x}_j \cdot \boldsymbol{e}_k). \tag{13}$$

Formula (12) can be used to compute $I_X\mathbf{F}$ after $\mathbf{c}$ is found.

3 Additive Schwarz method

In order to apply the additive Schwarz method, we need to write the interpolation problem (5) in a variational form. To this end, we introduce a norm and an inner product.

Given a divergence free vector field $\mathbf{F}$ there is a scalar stream function f such that $\mathbf{F} = \operatorname{Curl} f$ (recall $\operatorname{Curl} = \mathbf{n} \times \nabla^*$). We define the norm

$$\|\mathbf{F}\|_\Psi := \|f\|_\Psi = \left(\sum_{\ell=1}^{\infty} \sum_{m=1}^{2\ell+1} \frac{|\widehat{f}_{\ell,m}|^2}{\widehat{\psi}(\ell)} \right)^{1/2}, \quad \widehat{f}_{\ell,m} = \int_S f Y_{\ell,m} \, dS, \tag{14}$$

where $Y_{\ell,m}$ is the spherical harmonic and $\widehat{\psi}(\ell)$ is the Fourier–Legendre coefficient of ρ [4, formula (3.2)]. We then define

$$\mathcal{N}_\Psi := \{\mathbf{F} = \operatorname{Curl} f : f \in C^1(S) \text{ and } \|\mathbf{F}\|_\Psi < \infty\}$$

which is a Hilbert space with respect to the inner product

$$\langle \mathbf{F}, \mathbf{G} \rangle_\Psi := \langle f, g \rangle_\Psi = \sum_{\ell=1}^{\infty} \sum_{m=1}^{2\ell+1} \frac{\widehat{f}_{\ell,m} \widehat{g}_{\ell,m}}{\widehat{\psi}(\ell)}, \quad \mathbf{F} = \operatorname{Curl} f, \quad \mathbf{G} = \operatorname{Curl} g.$$

For our analysis, we assume further that for some $\tau > 1$,

$$\widehat{\psi}(\ell) \sim [1 + \ell(\ell+1)]^{-(\tau+1)}, \quad \ell = 0, 1, \ldots. \tag{15}$$

Then $\mathcal{N}_\Psi$ defined above is a reproducing kernel Hilbert space associated with the kernel Ψ which is contained in the space of continuous vector fields on S. More precisely [1, Theorem 2.2], Ψ is the reproducing kernel for $\mathcal{N}_\Psi$ in the sense that, for all $\mathbf{F} \in \mathcal{N}_\Psi$,

$$\langle \mathbf{F}, \Psi(\cdot, \mathbf{x}) \mathbf{t}_\mathbf{x} \rangle_\Psi = \mathbf{t}_\mathbf{x}^T \mathbf{F}(\mathbf{x}), \quad \text{for all } \mathbf{x} \in S, \tag{16}$$

where $\mathbf{t}_\mathbf{x}$ is any tangent vector based at $\mathbf{x}$.

Using the reproducing property (16) of the kernel Ψ, we write the interpolation equation (5) as

$$\langle I_X \mathbf{F}, \Psi(\cdot, \mathbf{x}_j)\mathbf{t}_{\mathbf{x}_j}\rangle_\Psi = \langle \mathbf{F}, \Psi(\cdot, \mathbf{x}_j)\mathbf{t}_{\mathbf{x}_j}\rangle_\Psi\,, \quad j = 1, \ldots, N\,. \tag{17}$$

Since V_X is spanned by $\Psi(\cdot, \mathbf{x}_j)\mathbf{e}_j$ and $\Psi(\cdot, \mathbf{x}_j)\widetilde{\mathbf{e}}_j$ for $j = 1, \ldots, N$, (17) is equivalent to the variational form

$$\langle I_X \mathbf{F}, \mathbf{G}\rangle_\Psi = \langle \mathbf{F}, \mathbf{G}\rangle_\Psi\,, \quad \text{for all } \mathbf{G} \in V_X\,. \tag{18}$$

Additive Schwarz methods provide fast solutions to equation (18) by solving, in parallel, problems of smaller size. Let the space V_X be decomposed as

$$V_X = V_0 + \cdots + V_J\,, \tag{19}$$

where V_j, for $j = 0, \ldots, J$, are subspaces of V_X. Let $P_j : V_X \to V_j$, for $j = 0, \ldots, J$, be projections defined by

$$\langle P_j \boldsymbol{v}, \boldsymbol{w}\rangle_\Psi = \langle \boldsymbol{v}, \boldsymbol{w}\rangle_\Psi\,, \quad \text{for all } \boldsymbol{v} \in V_X\,,\ \boldsymbol{w} \in V_j\,. \tag{20}$$

Defining

$$P := P_0 + \cdots + P_J\,, \tag{21}$$

then the additive Schwarz method for equation (18) consists in solving, by an iterative method, the equation

$$P I_X \mathbf{F} = \mathbf{g}\,, \tag{22}$$

where the right-hand side is $\mathbf{g} = \sum_{j=0}^{J} \mathbf{g}_j$, with $\mathbf{g}_j \in V_j$ being solutions of

$$\langle \mathbf{g}_j, \boldsymbol{w}\rangle_\Psi = \langle \mathbf{F}, \boldsymbol{w}\rangle_\Psi\,, \quad \text{for all } \boldsymbol{w} \in V_j\,. \tag{23}$$

The equivalence of (18) and (22) is well known [4]. A practical method to solve (22) is the conjugate gradient method; the additive Schwarz method (Algorithm 1) can be viewed as a preconditioned conjugate gradient method.

In the following, we present a decomposition of V_X into a sum of subspaces as in (19).

Let a spherical cap of radius α centred at $\mathbf{p} \in S$ be defined as

$$C(\mathbf{p}, \alpha) := \{\mathbf{x} \in S : \theta(\mathbf{p}, \mathbf{x}) < \alpha\}. \tag{24}$$

where $\theta(\mathbf{p}, \mathbf{x}) = \cos^{-1}(\mathbf{p} \cdot \mathbf{x})$ is the geodesic distance between two points $\mathbf{x}, \mathbf{p} \in S$. Let α now be a fixed number satisfying $0 < \alpha < \pi/3$ and let $X_0 := \{\mathbf{p}_j : j = 1, \ldots, J\}$ be a subset of X such that

$$X = \bigcup_{j=1}^{J} \left[\overline{C(\mathbf{p}_j, \alpha)} \cap X\right]. \tag{25}$$

For $j = 1, \ldots, J$, the subset X_j is defined as

$$X_j := \{\mathbf{x}_k \in X : \theta(\mathbf{x}_k, \mathbf{p}_j) \leqslant \alpha\} = \overline{C(\mathbf{p}_j, \alpha)} \cap X. \tag{26}$$

The sets X_j may have different numbers of elements and may overlap each other. Because of (25), X is decomposed into J overlapping subsets X_j, $j = 1, \ldots, J$.

We define $V_j = V_{X_j}$, $j = 0, \ldots, J$; that is,

$$V_j = \text{ span } \{\Psi(\mathbf{x}, \mathbf{x}_k)\boldsymbol{e}_k, \Psi(\mathbf{x}, \mathbf{x}_k)\widetilde{\boldsymbol{e}}_k : \mathbf{x}_k \in X_j\},$$

so that $V_X = V_0 + \cdots + V_J$. The Schwarz operator P is then defined by (20) and (21). We gave an algorithm to construct the sets X_j, $j = 1, \ldots, J$, satisfying (26) [4, Section 6].

In the following, we describe the preconditioned conjugate gradient method based on the Schwarz operator P. For $i = 0, \ldots, J$, let $\mathbf{A}_i$ be the restriction of the matrix $\mathbf{A}$ onto each subspace V_i, that is, $\mathbf{A}_i$ is a submatrix of size $2\,\mathrm{card}(X_i) \times 2\,\mathrm{card}(X_i)$ given by $\mathbf{A}_i = [A_{jk}]$, where A_{jk} is the 2×2 block matrix defined for $\mathbf{x}_j, \mathbf{x}_k \in X_i$ as given in (13).

For $k = 0, \dots, J$, let I_k be an ordered subset of $\{1, \dots, N\}$ such that $\boldsymbol{x}_m \in X_k$ if and only if $m \in I_k$. The cardinality of the set I_k is denoted by s_k and the rth element of the set I_k is denoted by $I_k(r)$. For a given vector $\boldsymbol{v} = (v_1, \widetilde{v}_1, \dots, v_N, \widetilde{v}_N)^T$, the restriction map $R_k : \mathbb{R}^{2N} \to \mathbb{R}^{2s_k}$ is defined as

$$R_k \boldsymbol{v} = (v_{I_k(1)}, \widetilde{v}_{I_k(1)}, \dots, v_{I_k(s_k)}, \widetilde{v}_{I_k(s_k)})^T.$$

Conversely, for a vector $\boldsymbol{u} = (u_1, \widetilde{u}_1, \dots, u_{s_k}, \widetilde{u}_{s_k})^T$, the extension map $R_k^T : \mathbb{R}^{2s_k} \to \mathbb{R}^{2N}$ is defined by $R_k^T \boldsymbol{u} = (v_1, \widetilde{v}_1, \dots, v_N, \widetilde{v}_N)^T$, where

$$(v_j, \widetilde{v}_j) = \begin{cases} (u_r, \widetilde{u}_r), & \text{if } j = I_k(r) \text{ for some } r \in \{1, \dots, s_k\}, \\ 0, & \text{if } j \notin I_k. \end{cases}$$

Algorithm 1 shows a pseudocode for the preconditioned conjugate gradient method.

4 Estimates of the condition number

Let φ_k and $\widetilde{\varphi}_k$ be defined so that $\operatorname{Curl} \varphi_k = \Psi(\cdot, \boldsymbol{x}_k)\boldsymbol{e}_k$ and $\operatorname{Curl} \widetilde{\varphi}_k = \Psi(\cdot, \boldsymbol{x}_k)\widetilde{\boldsymbol{e}}_k$. These stream functions also have compact supports. Let

$$W_X = \operatorname{span}\{\varphi_k,\ \widetilde{\varphi}_k\ : \boldsymbol{x}_k \in X\} \quad \text{and} \quad W_j = \operatorname{span}\{\varphi_k,\ \widetilde{\varphi}_k\ : \boldsymbol{x}_k \in X_j\}.$$

Assume the functions in W_j have supports in Γ_j, where Γ_j is a spherical cap centred at $\boldsymbol{p}_j$. We make the following assumption:

Assumption 1. *We can partition the index set* $1, \dots, J$ *into* M *(for* $1 \leqslant M \leqslant J$*) sets* J_m*, for* $1 \leqslant m \leqslant M$*, such that if* $i, j \in J_m$ *and* $i \neq j$ *then* $\Gamma_i \cap \Gamma_j = \emptyset$.

Under Assumption 1, the following results hold.

Algorithm 1: Preconditioned conjugate gradient method.

input : The scattered set X on the sphere, the values of the vector field $\mathbf{F}$ at X, and the desired accuracy ϵ.
output: The divergence free RBF approximation of $\mathbf{F}$

Partition the scattered set X into $X_0 \cup \cdots \cup X_J$
Compute the coordinates d_j and $\widetilde{d}_j$ of the vector field $\mathbf{F}$
$\mathbf{r} = [d_j \ \widetilde{d}_j]_{j=1}^{N}$
$\mathbf{p} = 0$
$\mathbf{c} = 0$
$\mathrm{iter} = 0$
while $\|\mathbf{r}\| > \epsilon$ **do**
    for $j = 1$ **to** J **do**
        $\mathbf{p} = \mathbf{p} + R_j^T \mathbf{A}_j^{-1} R_j \mathbf{r}$
    end
    $\mathbf{p} = \mathbf{p} + R_0^T \mathbf{A}_0^{-1} R_0 \mathbf{r}$
    if $\mathrm{iter} > 0$ **then**
        $\zeta_0 = \zeta_1$
    end
    $\zeta_1 = \mathbf{p} \cdot \mathbf{r}$
    $\mathrm{iter} = \mathrm{iter} + 1$
    if $\mathrm{iter} = 1$ **then**
        $\mathbf{p}_1 = \mathbf{p}$
    end
    else
        $\mathbf{p}_1 = \mathbf{p} + (\zeta_1/\zeta_0)\mathbf{p}_1$
    end
    $\gamma = (\mathbf{r} \cdot \mathbf{p})/(\mathbf{p}_1 \cdot \mathbf{A}\mathbf{p}_1)$
    $\mathbf{r} = \mathbf{r} - \gamma \mathbf{A}\mathbf{p}_1$
    $\mathbf{c} = \mathbf{c} + \gamma \mathbf{p}_1$
end
Construct the approximate solution using (6) and (12).

Lemma 2. *There exists a positive constant* c *independent of the set* X *such that for any* $\mathbf{u} \in V_X$ *satisfying* $\mathbf{u} = \sum_{j=0}^{J} \mathbf{u}_j$ *with* $\mathbf{u}_j \in V_j$ *for* $j = 0, \dots, J$,

$$\langle \mathbf{u}, \mathbf{u} \rangle_\Psi \leqslant cM \sum_{j=0}^{J} \langle \mathbf{u}_j, \mathbf{u}_j \rangle_\Psi .$$

Proof: Since $\mathbf{u} \in V_X$ is divergence free, there is a stream function $u \in W_X$ such that $\mathbf{u} = \operatorname{Curl} u$. Similarly, for every $\mathbf{u}_j \in V_j$ there is a stream function $u_j \in W_j$ such that $\mathbf{u}_j = \operatorname{Curl} u_j$. Le Gia and Tran [4, Lemma 5.2] gave, under the assumption (15),

$$\langle u, u \rangle_\psi \leqslant cM \sum_{j=0}^{J} \langle u_j, u_j \rangle_\psi .$$

This estimate and definition (14) give the desired result. ♠

Lemma 3. *For any* $\mathbf{u} \in V_X$ *there exist* $\mathbf{u}_j \in V_j$, $j = 0, \dots, J$, *satisfying* $\mathbf{u} = \sum_{j=0}^{J} \mathbf{u}_j$ *and*

$$\sum_{j=0}^{J} \langle \mathbf{u}_j, \mathbf{u}_j \rangle_\Psi \leqslant \left(1 + \frac{J}{(1 - \|\widetilde{Q}\|_\Psi)^2} \right) \langle \mathbf{u}, \mathbf{u} \rangle_\Psi ,$$

where $\widetilde{Q} = Q_J \cdots Q_1$ *and*

$$\|Q\|_\Psi = \sup\{\|\widetilde{Q}\mathbf{v}\|_\Psi : \mathbf{v} \in V_X \text{ and } \|\mathbf{v}\|_\Psi \leqslant 1\}.$$

Here Q_j *is the orthogonal projection from* V_X *to* $V_j^\perp$ *with respect to* $\langle \cdot, \cdot \rangle_\Psi$, *where* $V_j^\perp := \{\mathbf{u} \in \mathcal{N}_\Psi : \langle \mathbf{u}, \mathbf{w} \rangle_\Psi = 0 \quad \text{for all } \mathbf{w} \in V_j\}$.

Proof: The proof follows in the same manner as Lemma 2, using an external lemma by Le Gia and Tran [4, Lemma 5.3]. ♠

Lemmas 2 and 3 allow us to estimate the condition number of P, namely $\kappa(P) = \lambda_{\max}(P)/\lambda_{\min}(P)$ where $\lambda_{\max}(P)$ and $\lambda_{\min}(P)$ are the maximum eigenvalue and the minimum eigenvalue of P, respectively.

Theorem 4. *Under Assumption 1, the condition number* $\kappa(P)$ *of the additive Schwarz operator* P *is bounded by*

$$\kappa(P) \leqslant cM \left(1 + \frac{J}{(1 - \|\widetilde{Q}\|_\psi)^2}\right),$$

where c *is a constant independent of* M, J *and the set* X. *Here, the operator* $\widetilde{Q}$ *is defined in Lemma 3.*

Proof: From Lemmas 2 and 3, and the standard analysis for domain decomposition methods [4], we obtain $\lambda_{\max}(P) \leqslant cM$ and

$$\lambda_{\min}^{-1}(P) \leqslant \left(1 + \frac{J}{(1 - \|\widetilde{Q}\|_\psi)^2}\right).$$

Therefore the result of the theorem is obtained. ♠

5 Numerical experiments

We present numerical experiments based on globally scattered data extracted from a very large data set collected by the NASA satellite MAGSAT. Given a positive real number q, different sets X of scattered points are extracted from the original data set so that the separation radius q_X is not less than q.

TABLE 1: Radial basis functions used in the numerical experiments.

$\phi(r)$	τ
$(1-r)_+^6(35r^2+18r+3)$	5/2
$(1-r)_+^8(32r^3+25r^2+8r+1)$	7/2

The separation radius of a set $X = \{x_1, \ldots, x_N\} \subset S$ is defined by $q_X = 0.5 \min_{i \neq j} \cos^{-1}(x_i \cdot x_j)$. The number of points and the separation radius q_X of each data set are listed in Table 2. The exact vector field is

$$F = \frac{1}{\sin\theta}\frac{\partial F}{\partial \phi} e - \frac{\partial F}{\partial \theta}\widetilde{e},$$

where F is the following stream function

$$F = 4\cos(\alpha)\cos(\theta) - 4\sin(\alpha)\sin(\theta)\cos(\phi), \quad \alpha = \pi/4,$$

and the tangent vectors are

$$e = (\cos\theta\cos\phi, \cos\theta\sin\phi, -\sin\theta)^T \quad \text{and} \quad \widetilde{e} = (-\sin\phi, \cos\phi, 0)^T.$$

We use the compactly supported radial basis functions [6] listed in Table 1 to define the kernel Ψ. Tables 2 and 3–4 give numerical results for the conjugate gradient (CG) method and preconditioned CG.

As shown in Table 2, when N increases and q_X decreases, the condition number $\kappa(A)$ of the matrix A increases, and hence the CPU time (in seconds) increases significantly. When the preconditioner introduced in Algorithm 1 is applied, as can be seen in Tables 3–4, the condition numbers of the preconditioned systems $\kappa(P)$ are much smaller than the condition numbers of the original interpolation matrix $\kappa(A)$. As a consequence, the number of iterations and CPU times in solving the linear systems are reduced dramatically. This shows the effectiveness of the preconditioner.

TABLE 2: Conjugate gradient method (without preconditioners).

τ	N	q_X	λ_{min}	λ_{max}	$\kappa(A)$	CPU	ITER
5/2	7663	$\pi/200$	7.4E−3	8.8E+3	1.2E+6	4507.6	2526
	10443	$\pi/240$	4.6E−3	1.2E+4	2.7E+6	7300.2	3413
7/2	7763	$\pi/200$	7.0E−5	3.0E+3	4.3E+7	8706.9	8948
	10443	$\pi/240$	8.6E−5	4.1E+3	4.8E+7	14647.8	8805

References

[1] E. Fuselier, F. Narcowich, J. D. Ward, and G. Wright. Error and stability estimates for surface-divergence free RBF interpolants on the sphere. *Math. Comp.*, 78:2157–2186, 2009. doi:10.1090/S0025-5718-09-02214-5. C747

[2] J. R. Holton. *An Introduction to Dynamic Meteorology*. Academic Press, San Francisco, 3rd ed., 1992. doi:10.1119/1.1987371. C743

[3] Q. T. Le Gia, I. H. Sloan, and T. Tran. Overlapping additive Schwarz preconditioners for elliptic PDEs on the unit sphere. *Math. Comp.*, 78:79–101, 2009. doi:10.1090/S0025-5718-08-02150-9. C744

[4] Q. T. Le Gia and T. Tran. An overlapping additive Schwarz preconditioner for interpolation on the unit sphere with spherical radial basis functions. *J. of Complexity*, 26:552–573, 2010. doi:10.1016/j.jco.2010.06.003. C744, C747, C748, C749, C752, C753

[5] F. J. Narcowich, J. D. Ward, and G. B. Wright. Divergence-free RBFs on surfaces. *J. Fourier Anal. Appl.*, 13:643–663, 2007. doi:10.1007/s00041-006-6903-2. C743, C744

[6] H. Wendland. *Scattered Data Approximation*. Cambridge University Press, Cambridge, 2005. doi:10.2277/0521843359. C744, C754

TABLE 3: Preconditioned conjugate gradient method for $\tau = 5/2$.

N	$\cos\alpha$	$\cos\beta$	J	$\lambda_{\min}$	$\lambda_{\max}$	$\kappa(P)$	CPU	ITER
7663	0.95	0.46	84	9.6E−4	11.02	1.1E+4	785.6	296
7663	0.94	−0.54	76	1.9E−3	10.99	5.8E+3	519.8	177
7663	0.93	−0.60	65	4.6E−3	10.58	2.3E+3	459.2	145
7663	0.92	−0.35	56	7.1E−4	10.21	1.4E+4	631.8	177
7663	0.91	−0.70	50	2.9E−3	9.35	3.2E+3	514.8	130
7663	0.90	−0.15	45	8.9E−4	9.20	1.0E+4	753.2	175
7663	0.89	−0.62	42	5.1E−2	9.32	1.8E+2	231.4	50
7663	0.88	−0.61	37	5.5E−3	8.52	1.6E+3	370.9	74
7663	0.87	−0.56	37	3.5E−3	9.26	2.6E+3	571.2	101
7663	0.86	−0.73	30	1.1E−2	8.14	7.1E+2	369.4	67
7663	0.85	−0.62	33	3.4E−2	8.67	2.6E+2	345.6	52
10443	0.95	−0.07	86	4.6E−4	11.53	2.5E+4	767.8	250
10443	0.94	−0.35	75	4.8E−4	11.47	2.4E+4	648.9	195
10443	0.93	−0.60	65	1.6E−3	10.50	6.6E+3	644.6	186
10443	0.92	−0.35	55	7.5E−4	10.25	1.4E+4	623.2	165
10443	0.91	−0.66	48	4.9E−3	9.71	2.0E+3	378.8	101
10443	0.90	−0.63	45	2.8E−2	9.21	3.3E+2	277.5	57
10443	0.89	−0.66	39	1.4E−3	8.46	6.0E+3	886.9	170
10443	0.88	−0.69	37	1.8E−3	8.52	4.7E+3	840.0	123
10443	0.87	−0.69	37	6.2E−3	8.88	1.4E+3	386.8	63
10443	0.86	−0.58	33	2.3E−3	8.44	3.7E+3	720.8	94
10443	0.85	−0.79	30	4.8E−3	8.26	1.7E+3	637.6	82

TABLE 4: Preconditioned conjugate gradient method for $\tau = 7/2$.

N	cos α	cos β	J	λ_{min}	λ_{max}	κ(P)	CPU	ITER
7663	0.95	0.46	84	1.2E−4	11.23	9.3E+4	620.3	482
7663	0.94	−0.54	76	4.3E−4	11.15	2.6E+4	408.7	278
7663	0.93	−0.60	65	1.5E−4	10.79	7.2E+4	652.2	396
7663	0.92	−0.35	56	2.2E−3	10.45	4.8E+3	351.6	189
7663	0.91	−0.70	50	6.2E−4	9.62	1.5E+4	457.9	218
7663	0.90	−0.15	45	3.1E−4	9.41	3.1E+4	650.5	277
7663	0.89	−0.62	42	7.6E−3	9.54	1.3E+3	252.0	95
7663	0.88	−0.61	37	2.6E−3	8.74	3.3E+3	306.8	107
7663	0.87	−0.56	37	6.8E−4	9.50	1.4E+4	501.7	150
7663	0.86	−0.73	30	2.4E−3	8.27	3.5E+3	451.3	138
7663	0.85	−0.62	33	1.2E−2	8.92	7.6E+2	312.2	78
10443	0.95	−0.07	86	1.5E−3	11.75	7.6E+3	453.4	212
10443	0.94	−0.35	75	7.2E−4	11.76	1.6E+4	606.4	246
10443	0.93	−0.60	65	2.8E−4	10.69	3.8E+4	773.2	236
10443	0.92	−0.35	55	9.8E−4	10.61	1.1E+4	763.6	207
10443	0.91	−0.66	48	1.9E−3	9.94	5.4E+3	639.8	156
10443	0.90	−0.63	45	6.8E−3	9.40	1.4E+3	409.2	87
10443	0.89	−0.66	39	2.2E−4	8.58	3.9E+4	1549.9	305
10443	0.88	−0.69	37	2.3E−4	8.74	3.9E+4	1336.5	210
10443	0.87	−0.69	37	3.4E−3	9.10	2.7E+3	538.4	86
10443	0.86	−0.58	33	4.0E−3	8.59	2.1E+3	763.9	101
10443	0.85	−0.79	30	4.4E−3	8.51	2.0E+3	856.5	115

Author addresses

1. **Q. T. Le Gia**, School of Mathematics and Statistics, University of New South Wales, Sydney, AUSTRALIA.
mailto:qlegia@unsw.edu.au

2. **T. Tran**, School of Mathematics and Statistics, University of New South Wales, Sydney, AUSTRALIA.
mailto:thanh.tran@unsw.edu.au