TECHNICAL MEMORANDUM

808

"Time"-parallel diffusion-based correlation operators

A. T. Weaver[1], S. Gürol[1], J. Tshimanga[1], M. Chrust, A. Piacentini[1]

Research Department

[1]CERFACS / CECI CNRS UMR 5318, Toulouse, France

Submitted to Quarterly Journal of the Royal Meteorological Society

August 2017

European Centre for Medium-Range Weather Forecasts
Europäisches Zentrum für mittelfristige Wettervorhersage
Centre européen pour les prévisions météorologiques à moyen terme

Series: ECMWF Technical Memoranda

A full list of ECMWF Publications can be found on our web site under:
http://www.ecmwf.int/en/research/publications

Contact: library@ecmwf.int

European Centre for Medium-Range Weather Forecasts
Shinfield Park, Reading, RG2 9AX, England

Abstract

Correlation operators based on the solution of an implicitly formulated diffusion equation can be implemented numerically using the Chebyshev iteration method. The attractive properties of the algorithm for modelling correlation functions on high-performance computers have been discussed in a recent paper. The current paper describes a straightforward variant of that algorithm that allows the matrix-vector products involved in the sequential pseudo-time diffusion process to be performed in parallel. Contrary to the original algorithm, which requires solving a sequence of linear systems involving a symmetric positive-definite (SPD) matrix, the "time"-parallel algorithm requires solving a single linear system involving a nonsymmetric positive-definite (NSPD) matrix. The key information required by the Chebyshev iteration for solving the NSPD problem is an estimate of the extreme eigenvalues of the NSPD matrix. For the problem under consideration, the extreme eigenvalues of the NSPD matrix are the same as those of the original SPD matrix, and can be pre-computed using a Lanczos algorithm. The convergence properties of the algorithm are studied from a theoretical perspective and using numerical experiments with a diffusion-based covariance model employed with a variational data assimilation system for the global ocean. Results suggest that time-parallelization can reduce the run-time of an implicit diffusion-based correlation operator by greater than a factor of two. It can be implemented practically using a hybrid parallelization approach that combines Message Passing Interface tasks in the spatial domain with Open Multi-Processing threads spanning the pseudo-time dimension. The sensitivity of the results to preconditioning, to the choice of first guess and to the stopping criterion is discussed.

1 Introduction

Correlation operators are central to variational data assimilation. In three- and four-dimensional variational data assimilation (3D-Var and 4D-Var), they are required for modelling background error (Berre, 2015). In weak-constraint 4D-Var, they are needed for representing errors in the numerical model as well as background error (Rabier and Fisher, 2015). Developing correlation operators, and inverse correlation operators, for representing observation error is an area attracting increasing attention for improving the use of satellite data in 3D-Var and 4D-Var (Desroziers, 2015). Correlation operators are also fundamental to ensemble-variational assimilation (EnVar) where they are needed to localize low-rank sample covariance matrices constructed from ensemble perturbations (Lorenc, 2015). For Monte Carlo applications, the "square-root" of a correlation operator provides a practical tool for generating random samples that can be used to perturb model input parameters. With these different applications comes the need to develop general correlation operators that can represent diverse error structures, are appropriate for systems with large state vectors, and can execute efficiently on high-performance computers.

Correlation operators derived from diffusion-like operators have been developed extensively over the years for ocean and atmospheric data assimilation (Weaver and Courtier, 2001; Dobricic and Pinardi, 2008; Carrier and Ngodock, 2010; Purser *et al.*, 2003a,b). Diffusion-based correlation operators have also been developed in other fields, such as seismic inversion (Bui-Thanh *et al.*, 2013), uncertainty quantification (Gmeiner *et al.*, 2017) and geostatistics (Lindgren *et al.*, 2011). Diffusion operators can be formulated using explicit or implicit discretization schemes, but implicit schemes are more flexible and robust for correlation modelling (Mirouze and Weaver, 2010; Weaver and Mirouze, 2013). The current study focuses on implicitly formulated diffusion operators and their application to variational data assimilation in global ocean models.

Weaver *et al.* (2016) (hereafter WTP16) proposed the use of an iterative algorithm based on the Chebyshev iteration (CI) method (Gutknecht and Röllin, 2002) to provide an approximate solution of each symmetric positive-definite (SPD) linear system in the M-step sequence of an implicitly formulated dif-

fusion operator. CI has several attractive properties for the problem under consideration.

1. WTP16 show that the convergence properties of CI are similar, albeit slightly inferior, to those of conjugate gradients (CG) for the implicit diffusion problem under consideration. CG requires the evaluation of inner products on each iteration, for estimating the algorithm parameters and, possibly, for re-orthogonalizing residual vectors to combat the effects of round-off error. In a parallel domain decomposition, these inner products require costly global Message Passing Interface (MPI) communications, in addition to the local MPI communications before each system matrix application. CI requires only local MPI communications and is thus better suited for high-performance computers.

2. CI has small memory requirements and is straightforward to implement as it only requires the ability to perform matrix-vector multiplications with the implicit diffusion system matrix. This can be a major practical advantage over other methods such as multigrid or direct solvers that require structural knowledge of the system matrix and/or the availability of a hierarchy of grids. This is the case for global ocean models where complex boundary geometry and non-standard grids make these alternative methods less practical.

3. CI is a linear solver. Fixing the total number of CI iterations (K) and formulating the complete correlation operator as a product $\mathbf{U}\mathbf{U}^{\mathrm{T}}$, where CI is used to solve the sequence of $M/2$ linear systems in $\mathbf{U}$ and the adjoint of CI is used to solve the sequence of $M/2$ linear systems in $\mathbf{U}^{\mathrm{T}}$, allows the intrinsic SPD property of a correlation matrix to be preserved numerically regardless of the desired solution accuracy (value of K). WTP16 show that adequate solutions can be obtained using a modest convergence criterion (value of K). The SPD-preserving formulation is thus especially convenient for reducing the cost of the operator without introducing numerical inaccuracies in the gradient of the cost function required by the variational minimization algorithm.

CI is a parameter-dependent method that requires estimates of the extreme eigenvalues of the implicit diffusion system matrix. These can be estimated prior to entering the variational minimization algorithm using a Lanczos method (Saad, 2003, pp. 185–186). Alternatively, the simpler power method (Golub and Van Loan, 1996, pp. 330–332) can be used to estimate the maximum eigenvalue, which is sufficient since the minimum eigenvalue of the system matrix is approximately equal to one.

The implicit diffusion operator described in WTP16 is now a standard component of NEMOVAR (Mogensen *et al.*, 2012; Balmaseda *et al.*, 2013; Waters *et al.*, 2015), a variational data assimilation platform for the Nucleus for European Modelling of the Ocean (NEMO) model (Madec, 2008). The new operator produces more realistic correlations near complex boundaries and has improved scalability properties compared to the previous implicit diffusion operator that was built from a product of one-dimensional implicit diffusion (recursive filter-like) components. Nevertheless, further improvements are necessary to reduce the computational cost of the correlation model for applications involving high-resolution models or "large" correlation length scales. Large length scales are typical in a multi-scale covariance model formulation (Mirouze *et al.*, 2016), and are often required in the localization operator of EnVar, especially when using a large ensemble size (Ménétrier and Auligné, 2015) or scale-dependent localization (Buehner and Shlyaeva, 2015). Large length scales combined with a high-resolution model will increase the condition number of the system matrix, with the consequence of slowing down, possibly significantly, the convergence rate of CI.

There are different ways the cost of the correlation operator can be reduced while still remaining within the framework of CI. One possibility is to employ selective use of single precision in the CI solver. Preconditioning is another possibility, with domain decomposition preconditioners, such as Additive

Schwarz or Restricted Additive Schwarz (Cai and Sarkis, 1999), seemingly well suited for the problem at hand. Furthermore, for large-scale correlations, the diffusion computations could be done more economically on a grid with resolution comparable to the scale itself, which could be much coarser than the resolution of the native analysis grid.

In this paper we explore another approach, which aims at improving the parallel aspects of the algorithm to allow for faster execution of the correlation operator on high-performance computers. In particular, we show how the implicit diffusion problem can be reformulated in a way that allows the costly matrix-vector products involved in the M-step (pseudo-time) diffusion processes to be performed in parallel on each iteration of the CI solver. The approach falls into the general class of time-parallel algorithms, whose development is an active area of research in numerical modelling (for a review see Gander (2015)) and more recently in data assimilation (Rao and Sandu, 2016; Fisher and Gürol, 2017). The time-parallel formulation of implicit diffusion presented in this paper builds on an idea described in (Zhu, 1994, chapter 5), which first involves recasting the sequential M-step implicit diffusion operator in terms of a single nonsymmetric positive-definite (NSPD) linear system. The new aspects in this paper are the application of this technique to correlation modelling as well as the use of CI to solve the resulting NSPD system. Furthermore, we complement the latter with a theoretical analysis of the convergence properties of CI for this particular NSPD problem.

The structure of the paper is as follows. The formulation of a correlation matrix in terms of an implicitly formulated diffusion operator is described in section 2. The beginning of this section contains background material required to develop the NSPD formulation that is presented towards the end of the section. The CI algorithm used to solve the NSPD problem is described in section 3. Different aspects of the algorithm are discussed, including its theoretical convergence properties, and the choice of preconditioners and first guess. Numerical experiments using the NEMOVAR system are presented in section 4. The NSPD formulation is evaluated in terms of convergence behaviour and solution accuracy, and compared to the standard SPD formulation, which is the benchmark. Section 5 gives a summary and discussion. Appendix A describes a minor correction to the CI algorithm that was derived and applied in WTP16. Appendix B outlines the mathematical basis that ensures convergence of CI for the NSPD problem under consideration. Appendix C describes the properties of a block-diagonal preconditioning matrix that can be considered as a cheaper alternative to the ideal, block lower triangular form of the preconditioning matrix used in the experiments.

2 Diffusion-based correlation operators

2.1 General formulation and background material

The starting point is the general factored form of a univariate $N \times N$ correlation matrix $\mathbf{C}$ constructed from a self-adjoint filter $\mathbf{L} = \mathbf{L}^{1/2}\mathbf{L}^{1/2}$ (see WTP16):

$$\mathbf{C} = \mathbf{\Gamma}^{1/2}\mathbf{L}^{1/2}\mathbf{W}^{-1}\left(\mathbf{L}^{1/2}\right)^{\mathrm{T}}\mathbf{\Gamma}^{1/2} \tag{1}$$

where $\mathbf{W}$ defines the matrix of grid-dependent weights associated with the discrete inner product with respect to which $\mathbf{L}$ is self-adjoint ($\mathbf{L} = \mathbf{W}^{-1}\mathbf{L}^{\mathrm{T}}\mathbf{W}$), and $\mathbf{\Gamma} = \mathbf{\Gamma}^{1/2}\mathbf{\Gamma}^{1/2}$ is a diagonal matrix of normalization factors to ensure that the diagonal elements of $\mathbf{C}$ are approximately equal to one. In this paper we consider the class of self-adjoint filters built from an elliptic operator that describes implicit diffusion over a pseudo-time step interval $0 \leq m \leq M$, where m is an integer and M is taken to be even in order to obtain a practical expression for the operator $\mathbf{L}^{1/2}$.

For the implicit diffusion-based filters, $\mathbf{L}^{1/2}$ and $\left(\mathbf{L}^{1/2}\right)^{\mathrm{T}} = \left(\mathbf{L}^{\mathrm{T}}\right)^{1/2}$ have the specific form

$$\left.\begin{aligned} \mathbf{L}^{1/2} &= \left(\mathbf{A}^{-1}\right)^{M/2} \\ \left(\mathbf{L}^{1/2}\right)^{\mathrm{T}} &= \left(\left(\mathbf{A}^{\mathrm{T}}\right)^{-1}\right)^{M/2} \end{aligned}\right\} \tag{2}$$

where the elements of the self-adjoint matrix $\mathbf{A}$ are associated with the discrete representation of the operator $1-\nabla\cdot\boldsymbol{\kappa}\nabla$ involving the gradient operator ∇, divergence operator $\nabla\cdot$, and diffusion tensor $\boldsymbol{\kappa}$. The elements of $\boldsymbol{\kappa}$ have physical units of length squared. The correlation functions described by the implicit diffusion model are closely related to those from the Matérn family in $\mathbb{R}^d$ where d determines the dimensionality of the gradient and divergence operators (Guttorp and Gneiting, 2006; Weaver and Mirouze, 2013). If $\boldsymbol{\kappa}$ is constant and boundaries are ignored then the kernel (correlation function) of the correlation operator $\mathbf{C}$ is given by

$$c_d(r) = \frac{2^{1-M+d/2}}{\Gamma(M-d/2)}\, r^{M-d/2}\, K_{M-d/2}(r) \tag{3}$$

where $r = \sqrt{(\mathbf{x}-\mathbf{x}')^{\mathrm{T}}\,\boldsymbol{\kappa}^{-1}\,(\mathbf{x}-\mathbf{x}')}$ is a nondimensional distance measure between points $\mathbf{x}$ and $\mathbf{x}'$ in $\mathbb{R}^d$, $K_{M-d/2}(\cdot)$ is the Bessel function of the second kind of order $M-d/2$, and $\Gamma(\cdot)$ is the Gamma function. For this special case, the normalization matrix in Eq. (1) is simply $\boldsymbol{\Gamma} = \gamma_d \mathbf{I}_N$ where $\mathbf{I}_N$ is the $N\times N$ identity matrix, and

$$\gamma_d = 2^d \pi^{d/2} \frac{\Gamma(M)}{\Gamma(M-d/2)} \sqrt{|\boldsymbol{\kappa}|},$$

$|\boldsymbol{\kappa}|$ denoting the determinant of $\boldsymbol{\kappa}$.

The directional length scales of the correlation functions are controlled by $\boldsymbol{\kappa}$, which is a $d\times d$ symmetric positive-definite (and hence invertible) matrix. The fatness of the tails of the correlation functions, which in spectral space is related to the decay rate of the correlation spectrum at high wavenumbers, is controlled by the pseudo-time parameter M. There is a simple relationship between $\boldsymbol{\kappa}$ and the inverse of the tensor $\boldsymbol{D}^{-1}$ describing the curvature of the correlation function (3) near its peak (Weaver and Mirouze, 2013):

$$\boldsymbol{\kappa} = \left(\frac{1}{2M-d-2}\right)\boldsymbol{D}. \tag{4}$$

Michel *et al.* (2016) refer to $\boldsymbol{D}^{-1}$ as the local correlation tensor (LCT), while Weaver and Mirouze (2013) refer to the inverse of the LCT ($\boldsymbol{D}$) as the Daley tensor in view of its interpretation as a generalized length scale following a conventional definition given in the classical textbook by Daley (1991). The LCT or Daley tensor is of significant practical interest since it can be estimated from an ensemble of simulated errors (Belo Pereira and Berre, 2006; Weaver and Mirouze, 2013; Yaremchuk and Nechaev, 2013; Michel *et al.*, 2016). Equation (4) then provides a way of defining the diffusion tensor given an estimate of $\boldsymbol{D}$ and a specific value of M. Equation (4) can also be used approximately to introduce spatial dependence in $\boldsymbol{\kappa}$ when spatially varying estimates of $\boldsymbol{D}$ are available.

In the rest of the paper, we are concerned with finding efficient algorithms for performing matrix-vector multiplications with $\mathbf{L}^{1/2}$, and the corresponding adjoint operator $\left(\mathbf{L}^{1/2}\right)^{\mathrm{T}}$. Using the self-adjointness identity, $\mathbf{A} = \mathbf{W}^{-1}\,\mathbf{A}^{\mathrm{T}}\,\mathbf{W}$, it is possible to transform (2) into the equivalent form (see WTP16)

$$\left.\begin{aligned} \mathbf{L}^{1/2} &= \mathbf{W}^{-1/2}\,\widehat{\mathbf{L}}^{1/2}\,\mathbf{W}^{1/2} \\ \left(\mathbf{L}^{1/2}\right)^{\mathrm{T}} &= \mathbf{W}^{1/2}\,\widehat{\mathbf{L}}^{1/2}\,\mathbf{W}^{-1/2} \end{aligned}\right\}, \tag{5}$$

which involve *symmetric* matrices

$$\left.\begin{array}{rcl} \widehat{\mathbf{L}}^{1/2} & = & \left(\widehat{\mathbf{A}}^{-1}\right)^{M/2} = \left(\widehat{\mathbf{L}}^{1/2}\right)^{\mathrm{T}} \\ \widehat{\mathbf{A}} & = & \mathbf{W}^{1/2}\mathbf{A}\,\mathbf{W}^{-1/2} = \widehat{\mathbf{A}}^{\mathrm{T}} \end{array}\right\}. \tag{6}$$

This alternative representation, which involves a single inversion of the SPD matrix $\widehat{\mathbf{A}}$ instead of a pair of inversions of self-adjoint positive-definite matrices $\mathbf{A}$ and $\mathbf{A}^{\mathrm{T}}$, is more convenient for the iterative algorithms considered in this paper.

2.2 Symmetric and nonsymmetric formulations

We will work primarily with the operator $\widehat{\mathbf{L}}^{1/2}$ in what follows, which will require frequent reference to the half-interval parameter $M/2$. To lighten notation, we introduce the following definition:

$$M' \equiv M/2.$$

Let $\{\boldsymbol{\psi}_m\}$, $m = 1,\ldots,M'$, denote the sequence of N-dimensional vectors generated by the M'-step diffusion operator $\widehat{\mathbf{L}}^{1/2}$ acting on a given input vector $\boldsymbol{\psi}_0$. To apply $\widehat{\mathbf{L}}^{1/2}$, WTP16 propose an iterative approach to solve each of the SPD linear systems involved in the M'-sequence of implicit diffusion steps:

$$\left.\begin{array}{rcl} \widehat{\mathbf{A}}\boldsymbol{\psi}_1 & = & \boldsymbol{\psi}_0 \\ \widehat{\mathbf{A}}\boldsymbol{\psi}_2 & = & \boldsymbol{\psi}_1 \\ & \vdots & \\ \widehat{\mathbf{A}}\boldsymbol{\psi}_{M'} & = & \boldsymbol{\psi}_{M'-1} \end{array}\right\}. \tag{7}$$

Note that $\widehat{\mathbf{A}}$ itself depends on M through the parameterization (4) and that the condition number of $\widehat{\mathbf{A}}$ will increase when the value of M decreases (see Fig. 5a in WTP16). Apart from sharing the same $N \times N$ SPD matrix $\widehat{\mathbf{A}}$, the equations in (7) are linked in the sense that the solution of one equation constitutes the right-hand side (rhs) of the next equation in the sequence. The algorithm is purely sequential as the equations need to be solved one after another, following the sequence of increasing pseudo-time steps.

Our goal in this paper is to break this temporal chaining between the consecutive solution steps of the system in order to allow the matrix-vector products involving $\widehat{\mathbf{A}}$ to be carried out in parallel. For this purpose, we start by reformulating (7) as a *single* large linear system. Following Zhu (1994), this can be done by recasting the system (7) as

$$\left.\begin{array}{rcl} \widehat{\mathbf{A}}\boldsymbol{\psi}_1 & = & \boldsymbol{\psi}_0 \\ -\boldsymbol{\psi}_1 + \widehat{\mathbf{A}}\boldsymbol{\psi}_2 & = & \mathbf{0} \\ & \vdots & \\ -\boldsymbol{\psi}_{M'-1} + \widehat{\mathbf{A}}\boldsymbol{\psi}_{M'} & = & \mathbf{0} \end{array}\right\},$$

or, in compact form,

$$\boldsymbol{\mathcal{A}}\boldsymbol{\Psi} = \boldsymbol{\zeta} \tag{8}$$

where

$$\boldsymbol{\Psi} = \begin{pmatrix} \boldsymbol{\psi}_1 \\ \boldsymbol{\psi}_2 \\ \vdots \\ \boldsymbol{\psi}_{M'} \end{pmatrix}, \quad \boldsymbol{\zeta} = \begin{pmatrix} \boldsymbol{\psi}_0 \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{pmatrix}$$

and

$$\boldsymbol{\mathcal{A}} = \begin{pmatrix} \widehat{\mathbf{A}} & & & \\ -\mathbf{I}_N & \widehat{\mathbf{A}} & & \\ & \ddots & \ddots & \\ & & -\mathbf{I}_N & \widehat{\mathbf{A}} \end{pmatrix}. \tag{9}$$

The system matrix (9) is NSPD and of size $NM' \times NM'$. In what follows, and for reasons that will become clearer later in the paper when discussing the actual solution algorithm, we will refer to Eq. (8) as a *parallel* linear system.

The solution of Eq. (8) is a vector that concatenates the solutions of the individual equations in system (7). The solution of interest is the final state $\boldsymbol{\psi}_{M'}$ produced by the implicit diffusion process acting on the initial state $\boldsymbol{\psi}_0$. To emphasize this point, we can write Eq. (8) explicitly as a transformation from $\boldsymbol{\psi}_0$ to $\boldsymbol{\psi}_{M'}$ using the notation

$$\left.\begin{aligned} \boldsymbol{\mathcal{A}}\boldsymbol{\Psi} &= \mathbf{E}\,\boldsymbol{\psi}_0 \\ \boldsymbol{\psi}_{M'} &= \mathbf{F}^{\mathrm{T}}\boldsymbol{\Psi} \end{aligned}\right\}, \tag{10}$$

where

$$\mathbf{E}\,\boldsymbol{\psi}_0 = \boldsymbol{\zeta},$$

$\mathbf{E}$ and $\mathbf{F}$ being rectangular matrices of size $NM' \times N$, given by

$$\mathbf{E} = \begin{pmatrix} \mathbf{I}_N \\ \mathbf{0}_N \\ \vdots \\ \mathbf{0}_N \end{pmatrix} \quad \text{and} \quad \mathbf{F} = \begin{pmatrix} \mathbf{0}_N \\ \vdots \\ \mathbf{0}_N \\ \mathbf{I}_N \end{pmatrix}.$$

From Eq. (10), the operator $\widehat{\mathbf{L}}^{1/2}$ in Eq. (1) can be formally identified as

$$\widehat{\mathbf{L}}^{1/2} = \left(\widehat{\mathbf{L}}^{1/2}\right)^{\mathrm{T}} = \mathbf{F}^{\mathrm{T}}\,\boldsymbol{\mathcal{A}}^{-1}\,\mathbf{E}. \tag{11}$$

It is straightforward to show that $\boldsymbol{\mathcal{A}}^{-1}$ has the following block lower triangular structure:

$$\boldsymbol{\mathcal{A}}^{-1} = \begin{pmatrix} \widehat{\mathbf{A}}^{-1} & & & \\ \left(\widehat{\mathbf{A}}^{-1}\right)^2 & \widehat{\mathbf{A}}^{-1} & & \\ \vdots & \vdots & \ddots & \\ \left(\widehat{\mathbf{A}}^{-1}\right)^{M'} & \left(\widehat{\mathbf{A}}^{-1}\right)^{M'-1} & \cdots & \widehat{\mathbf{A}}^{-1} \end{pmatrix}. \tag{12}$$

The action of the extension operator $\mathbf{E}$ and restriction operator $\mathbf{F}^{\mathrm{T}}$ in Eq. (11) is to extract the element in the lower left-hand corner of the matrix (12), which shows that $\widehat{\mathbf{L}}^{1/2}$ defined by Eq. (11) is equivalent to $\widehat{\mathbf{L}}^{1/2}$ in Eq. (6).

An important point for the iterative solution methods considered in the next section is that the NSPD matrix $\boldsymbol{\mathcal{A}}$ and SPD matrix $\widehat{\mathbf{A}}$ have the same eigenvalues. This result follows from the particular block triangular structure of $\boldsymbol{\mathcal{A}}$, which leads to a simple relationship between the characteristic polynomials of $\boldsymbol{\mathcal{A}}$ and $\widehat{\mathbf{A}}$:

$$\det\left(\boldsymbol{\mathcal{A}} - \lambda\,\mathbf{I}_{NM'}\right) = \left(\det\left(\widehat{\mathbf{A}} - \lambda\,\mathbf{I}_N\right)\right)^{M'} \tag{13}$$

where $\det(\mathbf{P})$ stands for the determinant of the square matrix $\mathbf{P}$, and λ is an eigenvalue. Equation (13) implies that the eigenvalues of $\boldsymbol{\mathcal{A}}$ are identical to those of $\widehat{\mathbf{A}}$, with algebraic multiplicity at least equal to M'.

2.3 Hybrid time-sequential/time-parallel diffusion

It is possible to define $\widehat{\mathbf{L}}^{1/2}$ (and $\left(\widehat{\mathbf{L}}^{1/2}\right)^{\mathrm{T}}$) as a combination of time-sequential and time-parallel operations using the formulation

$$\left.\begin{aligned} \widehat{\mathbf{L}}^{1/2} &= \prod_{l=1}^{L} \left(\mathbf{F}_{m_l}^{\mathrm{T}} \boldsymbol{\mathcal{A}}_{m_l}^{-1} \mathbf{E}_{m_l}\right) \\ \left(\widehat{\mathbf{L}}^{1/2}\right)^{\mathrm{T}} &= \prod_{l=L}^{1} \left(\mathbf{E}_{m_l}^{\mathrm{T}} \left(\boldsymbol{\mathcal{A}}_{m_l}^{\mathrm{T}}\right)^{-1} \mathbf{F}_{m_l}\right) \end{aligned}\right\} \tag{14}$$

where L defines the number of sequential steps ($1 \leq L \leq M'$), and m_l is an integer ($1 \leq m_l \leq M'$) that defines the block size of the $m_l N \times m_l N$ matrix $\boldsymbol{\mathcal{A}}_{m_l}$ and of the $m_l N \times N$ matrices $\mathbf{E}_{m_l}$ and $\mathbf{F}_{m_l}$. For a given M, the values of m_l and L must satisfy $\sum_{l=1}^{L} m_l = M'$. The purely time-sequential formulation corresponds to $L = M'$ and $m_l = 1$, $l = 1, \ldots, M'$, while the purely time-parallel formulation corresponds to $L = 1$ and $m_1 = M'$. Since the number of blocks can be different on each step, the order of the operations is important and must be reversed in $\left(\widehat{\mathbf{L}}^{1/2}\right)^{\mathrm{T}}$ in order to maintain symmetry. This is ensured in Eq. (14) by reversing the bounds of the multiplication index l in the expression for $\left(\widehat{\mathbf{L}}^{1/2}\right)^{\mathrm{T}}$.

3 The solution algorithm

3.1 The Chebyshev iteration

The solution algorithm considered in this study is CI. WTP16 discuss its advantages over other methods for solving the linear systems in (7) within the context of correlation modelling for variational data assimilation. Here, we summarize the salient features of the method that are relevant to the current study.

First, consider the sequential system (7). The iterate $\boldsymbol{\psi}_m^k$ produced by CI at iteration k on the mth step of the sequence is such that its error $\mathbf{e}_m^k = \boldsymbol{\psi}_m^k - \boldsymbol{\psi}_m^*$ with respect to the exact solution on the mth step, $\boldsymbol{\psi}_m^* = \widehat{\mathbf{A}}^{-1} \boldsymbol{\psi}_{m-1}$, is

$$\mathbf{e}_m^k = \varphi_k(\widehat{\mathbf{A}})\, \mathbf{e}_m^0 \tag{15}$$

where φ_k is the polynomial that is the unique solution of

$$\min_{\substack{\varphi(0)=1 \\ \deg(\varphi) \leq k}} \max_{\lambda \in [\lambda_{\min}, \lambda_{\max}]} |\varphi(\lambda)|.$$

In words, φ_k is the polynomial of degree at most k and with unit value at the origin, which has the smallest maximum amplitude on the interval bounded by the minimum ($\lambda_{\min}$) and maximum ($\lambda_{\max}$) eigenvalues of $\widehat{\mathbf{A}}$. The *kth shifted and scaled Chebyshev polynomial of the first kind* provides the unique solution of this problem (Gergelits and Strakoš, 2014). Equation (15) can also be expressed as

$$\mathbf{r}_m^k = \varphi_k(\widehat{\mathbf{A}})\, \mathbf{r}_m^0 \tag{16}$$

where

$$\mathbf{r}_m^k = \widehat{\mathbf{A}} \boldsymbol{\psi}_m^k - \boldsymbol{\psi}_{m-1} = \widehat{\mathbf{A}}\, \mathbf{e}_m^k$$

is the residual vector.

The practical implementation of CI involves exploiting recurrence relations such that the polynomials φ_k themselves never need to be computed explicitly (see appendix A in WTP16, and appendix A in this paper which describes a minor correction to the algorithm presented in WTP16). All that is needed are estimates $\theta_{\min}$ and $\theta_{\max}$ of the extreme eigenvalues $\lambda_{\min}$ and $\lambda_{\max}$ on which the parameters of CI depend. The extreme eigenvalues can be estimated accurately using a Lanczos algorithm.

The convergence rate of CI is independent of the rhs. Exploiting this property, WTP16 recommend using CI with a fixed number of iterations (K) to solve each of the M' linear systems in $\widehat{\mathbf{L}}^{1/2}$ and using the *adjoint* of CI with the same number of iterations to solve the M' linear systems in $\left(\widehat{\mathbf{L}}^{1/2}\right)^{\mathrm{T}}$. A modest value of K (e.g., chosen to achieve a 10^{-3} or 10^{-4} reduction in the norm of the residual) is generally sufficient to obtain an acceptable representation of the correlation matrix (see Fig. 10 in WTP16). The fixed-iteration implementation of CI, together with its adjoint, allows the SPD property of $\mathbf{C}$ to be preserved exactly in numerical applications. This feature is especially convenient for reducing the computational cost of CI without introducing numerical inaccuracies in the gradient-based minimization algorithms used in variational assimilation. Furthermore, using a stopping criterion based on a fixed number of iterations rather than the residual norm implies that CI is entirely free of scalar product evaluations. As such, in a parallel domain decomposition, CI does not require any costly global MPI communications.

CI may be used to solve NSPD linear systems as well as SPD linear systems, provided that the eigenvalues lie in the open right half of the complex plane (Manteuffel, 1977). For the time-parallel implicit diffusion problem, the eigenvalues are real and, as discussed earlier, identical to those of the SPD matrix $\widehat{\mathbf{A}}$. For the augmented system, the iterate $\boldsymbol{\Psi}_k$ produced by CI at iteration k is such that (cf. Eq. (15))[1]

$$\boldsymbol{\mathcal{E}}_k \;=\; \varphi_k(\boldsymbol{\mathcal{A}})\,\boldsymbol{\mathcal{E}}_0 \tag{17}$$

where $\boldsymbol{\mathcal{E}}_k = \boldsymbol{\Psi}_k - \boldsymbol{\Psi}^*$ is the error with respect to the exact solution $\boldsymbol{\Psi}^* = \boldsymbol{\mathcal{A}}^{-1}\boldsymbol{\zeta}$. Furthermore, and in analogy with Eq. (16),

$$\boldsymbol{\Xi}_k \;=\; \varphi_k(\boldsymbol{\mathcal{A}})\,\boldsymbol{\Xi}_0 \tag{18}$$

where

$$\boldsymbol{\Xi}_k \;=\; \boldsymbol{\mathcal{A}}\boldsymbol{\Psi}_k - \boldsymbol{\zeta} \;=\; \begin{pmatrix} \boldsymbol{\xi}_1^k \\ \vdots \\ \boldsymbol{\xi}_{M'}^k \end{pmatrix}$$

is the residual vector on the kth iteration.

The CI algorithm applied to the NSPD linear system is presented in Algorithm 1. This algorithm is similar to Algorithm 2 in WTP16 but is written in terms of the NSPD matrix and with a corrected value for the β_1 coefficient (see appendix A for an explanation). The convergence properties of Algorithm 1 are discussed in the next section, with mathematical details given in appendix B. When the stopping criterion is based on a fixed number of iterations K, Algorithm 2 can be used to compute the α_k and β_k coefficients prior to entering the main computational loop, which is convenient for constructing the adjoint of CI. The resulting fixed-iteration version of CI is given in Algorithm 3. Further discussion about the adjoint of the algorithm is deferred until section 3.5.

The application of $\boldsymbol{\mathcal{A}}$ is the most costly step in Algorithm 3. This cost is dominated by the application of the M' block-matrices $\widehat{\mathbf{A}}$ and by the local MPI communications that are required, before $\widehat{\mathbf{A}}$ is applied,

[1] Note the notation here where a subscript (superscript) denotes the CI iteration counter k on vectors associated with the NSPD (SPD) system. The subscript on vectors associated with the SPD system denotes the step counter m.

Algorithm 1 Chebyshev Iteration

$\boldsymbol{\Xi}_0 = \boldsymbol{\mathcal{A}}\boldsymbol{\Psi}_0 - \boldsymbol{\zeta} :=$ initial residual (input)
$\mathbf{p}_0 = -\boldsymbol{\Xi}_0$
$\sigma = (\theta_{\max} + \theta_{\min})/2$
$\delta = (\theta_{\max} - \theta_{\min})/2$
$\alpha_0 = 1/\sigma$
$\beta_1 = (\delta\,\alpha_0)^2/2$
$k = 0$
while *stopping criterion not satisfied* **do**
  $\mathbf{q}_k = \boldsymbol{\mathcal{A}}\mathbf{p}_k$
  $\boldsymbol{\Psi}_{k+1} = \boldsymbol{\Psi}_k + \alpha_k \mathbf{p}_k$
  $\boldsymbol{\Xi}_{k+1} = \boldsymbol{\Xi}_k + \alpha_k \mathbf{q}_k$
  $\mathbf{p}_{k+1} = -\boldsymbol{\Xi}_{k+1} + \beta_{k+1}\mathbf{p}_k$
  $k = k + 1$
  $\alpha_k = 1/(\sigma - \beta_k/\alpha_{k-1})$
  $\beta_{k+1} = (\delta\,\alpha_k/2)^2$
end while

Algorithm 2 Chebyshev iteration parameters for fixed K

$\sigma = (\theta_{\max} + \theta_{\min})/2$
$\delta = (\theta_{\max} - \theta_{\min})/2$
$\alpha_0 = 1/\sigma$
$\beta_1 = (\delta\,\alpha_0)^2/2$
for $k = 1, \ldots, K-1$ **do**
  $\alpha_k = 1/(\sigma - \beta_k/\alpha_{k-1})$
  $\beta_{k+1} = (\delta\,\alpha_k/2)^2$
end for

Algorithm 3 Chebyshev iteration with fixed K

Initialize parameters using Algorithm 2
$\boldsymbol{\Xi}_0 = \boldsymbol{\mathcal{A}}\boldsymbol{\Psi}_0 - \boldsymbol{\zeta} :=$ initial residual (input)
$\mathbf{p}_0 = -\boldsymbol{\Xi}_0$
for $k = 0, \ldots, K-1$ **do**
  $\mathbf{q}_k = \boldsymbol{\mathcal{A}}\mathbf{p}_k$
  $\boldsymbol{\Psi}_{k+1} = \boldsymbol{\Psi}_k + \alpha_k \mathbf{p}_k$
  $\boldsymbol{\Xi}_{k+1} = \boldsymbol{\Xi}_k + \alpha_k \mathbf{q}_k$
  $\mathbf{p}_{k+1} = -\boldsymbol{\Xi}_{k+1} + \beta_{k+1}\mathbf{p}_k$
end for
$\boldsymbol{\Psi}_K :=$ solution (output)

to update the halos of the spatial subdomains in a parallel decomposition (see appendix C in WTP16). The application of $\widehat{\mathbf{A}}$ and the local MPI communications are applied separately to each subset of the input vector that corresponds to a particular time level. These operations can be performed in parallel to reduce the overall run-time. This is the key point that makes Algorithm 3 potentially advantageous compared to the time-sequential diffusion algorithm based on CI which was studied in WTP16. In order to be advantageous, however, Algorithm 3 must have adequate convergence properties for the NSPD linear system under consideration. Furthermore, the matrix-vector products must be implemented in parallel in a cost-effective way. This requires careful consideration of technical aspects of the parallel computing environment and data assimilation system. The remainder of the paper will focus on the first point (convergence properties). Some discussion of the second point (parallel implementation) is given section 4.8.

3.2 Convergence properties

Algorithm 3 generates a sequence of residual vectors following relation (18). For a *general* polynomial-based iteration, φ_k is chosen such that

$$\|\boldsymbol{\Xi}_k\| \;\leq\; \|\varphi_k(\boldsymbol{\mathcal{A}})\|\,\|\boldsymbol{\Xi}_0\| \tag{19}$$

is small, where $\|\cdot\|$ denotes the 2-norm. If $\boldsymbol{\mathcal{A}}$ is diagonalizable then $\|\varphi_k(\boldsymbol{\mathcal{A}})\|$ converges to zero uniformly as $k\to\infty$ if and only if $|\varphi_k(\lambda_n)|\to 0$ as $k\to\infty$ for every eigenvalue λ_n of $\boldsymbol{\mathcal{A}}$ (Manteuffel, 1977). Since the NSPD matrix (9) is not diagonalizable, convergence of a polynomial-based method such as CI is not ensured by only verifying convergence of the polynomials.

In order to understand the convergence properties of a nondiagonalizable matrix, we need to examine the Jordan form of $\boldsymbol{\mathcal{A}}$. Theorem 2.2 of Manteuffel (1977) states that the convergence of a general polynomial-based iteration applied to the nondiagonalizable matrix $\boldsymbol{\mathcal{A}}$ requires that the $M'-1$ derivatives of the polynomials, as well as the polynomials themselves, converge to zero uniformly as $k\to\infty$ for every eigenvalue λ_n of $\boldsymbol{\mathcal{A}}$. The shifted and scaled Chebyshev polynomials satisfy these conditions (Manteuffel, 1975, Theorem 2.12, p. 27). This ensures that Algorithm 3 will converge, despite the fact that $\boldsymbol{\mathcal{A}}$ is nondiagonalizable.

From Eq. (19), it can be seen that $\|\varphi_k(\boldsymbol{\mathcal{A}})\|$ gives a measure of the amount by which the norm of the error is reduced after k iterations. A specific reduction is achieved by choosing k such that

$$\|\boldsymbol{\Xi}_k\| \;\leq\; \varepsilon\,\|\boldsymbol{\Xi}_0\|$$

where ε is the tolerance, and $0<\|\varphi_k(\boldsymbol{\mathcal{A}})\|\leq\varepsilon<1$.

For the SPD system, the number of iterations (k_s) required to reach a given tolerance ε is bounded by (Axelsson, 1996, p. 181)

$$k_\mathrm{s} \;\leq\; \frac{1}{2}\sqrt{\chi}\,\ln\!\left(\frac{2}{\varepsilon}\right) \tag{20}$$

where

$$\chi \;=\; \frac{\lambda_\mathrm{max}}{\lambda_\mathrm{min}}$$

is the condition number of $\widehat{\mathbf{A}}$. Equation (20) is derived under the assumptions that $\varepsilon\ll 1$ and $\chi\gg 1$. Experiments presented in section 4, however, suggest that this equation provides a sharp upper bound on k_s even when these assumptions are only modestly satisfied.

For the NSPD system, $\|\varphi_k(\boldsymbol{\mathcal{A}})\|$ behaves like $\left|\varphi_k^{(M'-1)}(\lambda_n)\right|/(M'-1)!$ as $k\to\infty$, rather than like $|\varphi_k(\lambda_n)|$ as in the SPD case, where the superscript (j) on φ_k denotes the jth derivative of φ_k. This result is derived in appendix B (see Theorem 2). The dependence of $\|\varphi_k(\boldsymbol{\mathcal{A}})\|$ on the derivatives of the Chebyshev polynomials slows down the rate of convergence of CI for the NSPD system. Manteuffel (1975) shows that

$$|\varphi_k^{(j)}(\lambda_n)| \le \eta(\lambda_n, j)\, k^j\, |\varphi_k(\lambda_n)| \tag{21}$$

where $\eta(\lambda_n, j)$ is a constant that depends only on λ_n and the order of the derivative (j). From the term k^j, it is clear that convergence will be degraded for increasing j.

3.3 Preconditioning

The convergence rate of CI may be accelerated using a preconditioning matrix $\boldsymbol{\mathcal{P}} \approx \boldsymbol{\mathcal{A}}^{-1}$. It can be applied to the augmented system either as a *left preconditioner*,

$$\boldsymbol{\mathcal{P}}\boldsymbol{\mathcal{A}}\boldsymbol{\Psi} = \boldsymbol{\mathcal{P}}\boldsymbol{\zeta},$$

or as a *right preconditioner*,

$$\left.\begin{aligned} \boldsymbol{\mathcal{A}}\boldsymbol{\mathcal{P}}\boldsymbol{\Psi}' &= \boldsymbol{\zeta} \\ \boldsymbol{\Psi} &= \boldsymbol{\mathcal{P}}\boldsymbol{\Psi}' \end{aligned}\right\}.$$

Equation (12) suggests that $\boldsymbol{\mathcal{P}}$ should have the block lower triangular matrix form

$$\boldsymbol{\mathcal{P}} = \begin{pmatrix} \mathbf{P} & & & \\ \mathbf{P}^2 & \mathbf{P} & & \\ \vdots & \vdots & \ddots & \\ \mathbf{P}^{M'} & \mathbf{P}^{M'-1} & \cdots & \mathbf{P} \end{pmatrix} \tag{22}$$

where $\mathbf{P} \approx \widehat{\mathbf{A}}^{-1}$. The ideal preconditioner is $\boldsymbol{\mathcal{P}} = \boldsymbol{\mathcal{A}}^{-1}$, which corresponds to setting $\mathbf{P} = \widehat{\mathbf{A}}^{-1}$ in Eq. (22). For this ideal case, the CI parameters are $(\sigma, \delta) = (\alpha_0, \beta_1) = (1, 0)$ since $\lambda_{\max} = \lambda_{\min} = 1$, and convergence to the exact solution is achieved in a single iteration from an arbitrary first guess.

The preconditioned NSPD matrices $\boldsymbol{\mathcal{P}}\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{A}}\boldsymbol{\mathcal{P}}$ have the same eigenvalues. Moreover, their eigenvalues are multiples of those of the preconditioned SPD matrices $\mathbf{P}\widehat{\mathbf{A}}$ and $\widehat{\mathbf{A}}\mathbf{P}$ as can be readily deduced by examining the characteristic polynomial of $\boldsymbol{\mathcal{P}}\boldsymbol{\mathcal{A}}$ (or $\boldsymbol{\mathcal{A}}\boldsymbol{\mathcal{P}}$). This has two consequences. First, the extreme eigenvalues of $\boldsymbol{\mathcal{P}}\boldsymbol{\mathcal{A}}$ (or $\boldsymbol{\mathcal{A}}\boldsymbol{\mathcal{P}}$) required by CI can be computed by applying the Lanczos algorithm to the simpler matrix $\mathbf{P}\widehat{\mathbf{A}}$ (or $\widehat{\mathbf{A}}\mathbf{P}$). If $\mathbf{P}$ can be factored as $\mathbf{P} = \mathbf{P}^{1/2}\left(\mathbf{P}^{1/2}\right)^{\mathrm{T}}$ then the eigenvalues can also be computed by applying the Lanczos algorithm to the symmetric matrix $\left(\mathbf{P}^{1/2}\right)^{\mathrm{T}}\widehat{\mathbf{A}}\mathbf{P}^{1/2}$. The second consequence is that left and right preconditioning lead to identical iterates with CI. This occurs since CI depends only on the (extreme) eigenvalues of the preconditioned matrix, which are identical for the left- and right-preconditioned matrices. The right-preconditioned CI algorithm is presented in Algorithm 4. The first guess in Algorithm 4 is defined as a transformation of the rhs, the details of which are discussed in the next section.

The direct use of Eq. (22) in the preconditioning transformation $\boldsymbol{\Psi} = \boldsymbol{\mathcal{P}}\boldsymbol{\Psi}'$ leads to operations of the form

$$\boldsymbol{\psi}_m = \mathbf{P}^m\boldsymbol{\psi}'_1 + \cdots + \mathbf{P}\boldsymbol{\psi}'_m \quad \text{for } m = 1, \ldots, M'.$$

These computations can be expensive as they involve $\sum_{i=1}^{m} i$ applications of $\mathbf{P}$ and require accessing the input states $\boldsymbol{\psi}'_i$ on all previous time levels $1 \le i \le m$. The recursive formula

$$\boldsymbol{\psi}_m = \mathbf{P}\left(\boldsymbol{\psi}_{m-1} + \boldsymbol{\psi}'_m\right) \quad \text{for } m = 1, \ldots, M', \tag{23}$$

Algorithm 4 Right-preconditioned Chebyshev iteration

Initialize parameters using Algorithm 2
$\boldsymbol{\zeta}$:= rhs (input)
$\boldsymbol{\Psi}_0 := \mathcal{G}\boldsymbol{\zeta}$
$\boldsymbol{\Xi}_0 = \mathcal{A}\boldsymbol{\Psi}_0 - \boldsymbol{\zeta}$
$\mathbf{p}_0 = -\boldsymbol{\Xi}_0$
for $k = 0, \ldots, K-1$ **do**
  $\mathbf{u}_k = \mathcal{P}\mathbf{p}_k$
  $\mathbf{q}_k = \mathcal{A}\mathbf{u}_k$
  $\boldsymbol{\Psi}'_{k+1} = \boldsymbol{\Psi}'_k + \alpha_k \mathbf{p}_k$
  $\boldsymbol{\Xi}_{k+1} = \boldsymbol{\Xi}_k + \alpha_k \mathbf{q}_k$
  $\mathbf{p}_{k+1} = -\boldsymbol{\Xi}_{k+1} + \beta_{k+1}\mathbf{p}_k$
end for
$\boldsymbol{\Psi}_K = \mathcal{P}\boldsymbol{\Psi}'_K$:= solution (output)

with $\boldsymbol{\psi}_0 = \mathbf{0}$, provides a much more efficient algorithm to apply $\mathcal{P}$. On a given time level, Eq. (23) requires the input state on that level and only one application of $\mathbf{P}$. However, whereas the application of the M $\widehat{\mathbf{A}}$ blocks in $\mathcal{A}$ (Eq. (9)) can be done in parallel, the application of the $\mathbf{P}$ blocks in $\mathcal{P}$ via Eq. (23) is *sequential* since the state on time level $m-1$ must be evaluated before the state on time level m can be evaluated. The application of $\mathbf{P}$ cannot be too costly if we are to benefit from applying the blocks of $\mathcal{A}$ in parallel. For this reason, we consider only simple preconditioners in what follows.

We will use the subscript notation $\mathcal{P}_{\mathbf{P}^{-1}}$ when we need to distinguish preconditioners $\mathcal{P}$ that have different blocks $\mathbf{P}$. Otherwise, we will omit the subscript to simplify notation. In section 4 we will examine the effect of two simple preconditioners of the form (22). The first ($\mathcal{P}_{\mathbf{I}}$) employs $\mathbf{P} = \mathbf{I}_N$; the second ($\mathcal{P}_{\mathbf{D}}$) employs $\mathbf{P} = \mathbf{D}^{-1}$ where $\mathbf{D}$ is a diagonal matrix defined by the diagonal of $\widehat{\mathbf{A}}$.

Rather than preconditioning the augmented system directly, another approach is to precondition the sequence of SPD systems in (7), using a left or right preconditioning matrix $\mathbf{P} \approx \widehat{\mathbf{A}}^{-1}$, and to form the augmented system from the resulting preconditioned sequence. This procedure leads to a simpler, block-diagonal form of the preconditioner:

$$\widetilde{\mathcal{P}} = \begin{pmatrix} \mathbf{P} & & \\ & \ddots & \\ & & \mathbf{P} \end{pmatrix}. \tag{24}$$

However, as shown in appendix C, Eq. (24) is not ideal in the sense that employing it with the ideal block matrix $\mathbf{P} = \widehat{\mathbf{A}}^{-1}$ results in convergence of CI in M' iterations (not one). This block-diagonal preconditioner will be denoted by $\widetilde{\mathcal{P}}_{\mathbf{P}^{-1}}$ when we need to identify the block matrix $\mathbf{P}$.

3.4 First guess

The first guess affects the residual vector used to initialize the CI algorithm. An appropriate choice of first guess (initial residual vector) can improve the accuracy of the iterative solution (see Eqs (15)-(16) and Eqs (17)-(18)), especially when the iterative algorithm is employed with a modest stopping criterion. For the case where the linear systems in (7) are solved in sequence, the solution of each system may be taken as a good initial iterate for the next system in the sequence. This corresponds to choosing the first guess to be the rhs of each member of the sequence.

For the augmented system, using the rhs as first guess results in only the first element of the augmented system having a non-zero value. The preconditioners presented in the previous section can be used to provide a potentially better guess. Since $\mathcal{P}$ is constructed such that $\mathcal{P} \approx \mathcal{A}^{-1}$, it suggests that the first guess can be chosen as

$$\boldsymbol{\Psi}_0 = \mathcal{G}\boldsymbol{\zeta} = \begin{pmatrix} \mathbf{P}\boldsymbol{\psi}_0 \\ \mathbf{P}^2\boldsymbol{\psi}_0 \\ \vdots \\ \mathbf{P}^{M'}\boldsymbol{\psi}_0 \end{pmatrix} \quad \text{if} \quad \mathcal{G} = \mathcal{P}, \tag{25}$$

or

$$\boldsymbol{\Psi}_0 = \mathcal{G}\boldsymbol{\zeta} = \begin{pmatrix} \mathbf{P}\boldsymbol{\psi}_0 \\ \mathbf{0}_N \\ \vdots \\ \mathbf{0}_N \end{pmatrix} \quad \text{if} \quad \mathcal{G} = \widetilde{\mathcal{P}}. \tag{26}$$

We distinguish $\mathcal{G}$ used to compute the first guess from the transformation $\mathcal{P}$ used for the preconditioner, as we can choose them to be different. Likewise, for the SPD system, the first guess can be chosen as

$$\boldsymbol{\psi}^0_m = \mathbf{G}\,\boldsymbol{\psi}_{m-1} \quad \text{for} \quad m = 1, \ldots, M' \tag{27}$$

where $\mathbf{G}$ is a preconditioner for the SPD system. In section 4 we will examine the effect of different choices of first-guess fields for the SPD and NSPD systems (see Tables 1 and 2).

3.5 Adjoint of the Chebyshev iteration

A correlation model would typically employ Algorithm 4 for the solver in $\widehat{\mathbf{L}}^{1/2}$ and the adjoint of Algorithm 4 for the solver in $\left(\widehat{\mathbf{L}}^{1/2}\right)^{\mathrm{T}}$. The adjoint of Algorithm 4 is given in Algorithm 5. It is derived by transposing each line of Algorithm 4 and reversing the order of the operations, as would be done following a standard automatic differentiation procedure. Algorithm 4 is initialized with the rhs, whereas Algorithm 5 is initialized with the adjoint vector associated with the "solution". The exactness of the adjoint algorithm is checked numerically by verifying that, for vectors $\boldsymbol{\psi}_0 = \mathbf{W}^{1/2}\mathbf{v}_1$ and $\boldsymbol{\psi}_1 = \mathbf{W}^{1/2}\mathbf{v}_2$ with vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ arbitrary and dimensionless, the inner product $\left(\widehat{\mathbf{L}}^{1/2}\boldsymbol{\psi}_0\right)^{\mathrm{T}}\boldsymbol{\psi}_1 = \boldsymbol{\psi}_0^{\mathrm{T}}\left(\widehat{\mathbf{L}}^{1/2}\right)^{\mathrm{T}}\boldsymbol{\psi}_1$ is satisfied to machine precision, where the left- and right-hand sides of the identity employ the forward and adjoint solvers, respectively.

4 Numerical experiments

4.1 Experimental design

The experimental design is similar to that of WTP16. The methods described in this paper have been implemented in NEMOVAR. Numerical experiments using this system are described in this section. The ocean model configuration is global, with an average resolution of about 1° except within a few degrees of the Equator where the meridional resolution is increased to around $1/3°$. The configuration is identical to the one used by Balmaseda *et al.* (2013).

In this paper the iterative solvers are applied to a correlation model based on a fully 3D implicit diffusion operator. This is the main difference between the experimental set-up here and that of WTP16 where

Algorithm 5 Adjoint of Algorithm 4

Initialize parameters using Algorithm 2
$\boldsymbol{\Psi}_K :=$ adjoint "solution" (input)
$\boldsymbol{\Xi}_K = \mathbf{p}_K = \mathbf{0}$
for $k = K-1,\ldots,0$ **do**
  $\boldsymbol{\Xi}_k = \boldsymbol{\Xi}_{k+1} - \mathbf{p}_{k+1}$
  $\mathbf{q}_k = \alpha_k \boldsymbol{\Xi}_k$
  $\mathbf{u}_k = \boldsymbol{\mathcal{A}}^{\mathrm{T}} \mathbf{q}_k$
  $\mathbf{s}_k = \boldsymbol{\mathcal{P}}^{\mathrm{T}} \mathbf{u}_k$
  $\mathbf{p}_k = \beta_{k+1} \mathbf{p}_{k+1} + \alpha_k \boldsymbol{\Psi}_K + \mathbf{s}_k$
end for
$\overline{\boldsymbol{\Xi}}_0 = \boldsymbol{\Xi}_0 - \mathbf{p}_0$
$\boldsymbol{\Psi}_0 = \boldsymbol{\Psi}_K + \boldsymbol{\mathcal{A}}^{\mathrm{T}} \overline{\boldsymbol{\Xi}}_0$
$\boldsymbol{\zeta} = -\overline{\boldsymbol{\Xi}}_0 + \boldsymbol{\mathcal{G}}^{\mathrm{T}} \boldsymbol{\Psi}_0 :=$ adjoint "rhs" (output)

experiments were performed with a 2D implicit diffusion operator. The diffusion operator is formulated in general orthogonal curvilinear coordinates and discretized using centred finite differences. Neumann boundary conditions are applied at coastlines, at the ocean-atmosphere interface, and at the irregular ocean bottom defined by bathymetry.

The diffusion tensor $\boldsymbol{\kappa}$ is taken to be diagonal. The diagonal elements κ_1, κ_2 and κ_3 control the directional length-scales of the correlation function along the coordinate axes (horizontal and vertical). In the orthogonal curvilinear coordinate system of the global model, the horizontal axes roughly correspond to the geographical axes λ and ϕ on the (assumed) spherical Earth, except from mid- to high latitudes in the Northern Hemisphere (NH) where they deviate from the geographical axes to account for the two NH poles located over North America and Asia on the tri-polar grid. The vertical (z-)axis is everywhere perpendicular to the sphere.

An idealized distribution of values for the 3D diffusion tensor elements is considered for the experiments here. The diagonal elements are taken to be spatially dependent with their value at each grid-point (i,j,k) defined according to Eq. (4) (with $d=3$). Specifically, we take $\kappa_1(i,j) = D_1^2/(2M-5)$, $\kappa_2(i,j) = D_2^2/(2M-5)$ and $\kappa_3(k) = D_3^2/(2M-5)$, where $D_1 = \rho_1 e_1$, $D_2 = \rho_2 e_2$ and $D_3 = \rho_3 e_3$ are directional length-scales; $e_1 = e_1(i,j)$, $e_2 = e_2(i,j)$ and $e_3 = e_3(k)$ are the local scale factors associated with the curvilinear grid; and ρ_1, ρ_2 and ρ_3 are positive proportionality constants. For the experiments, values of $\rho_1 = \rho_2 = 5$ and $\rho_3 = 1.5$ are used. Within $R = 500$ km of coastlines, the horizontal length scales are reduced according to $D_1 \mapsto \max(D_1\, r_{\mathrm{coa}}/R, e_1)$ and $D_2 \mapsto \max(D_2\, r_{\mathrm{coa}}/R, e_2)$ where r_{coa} is the chordal distance to the nearest coastal point. The local grid-steps e_1 and e_2 define the minimum-allowed zonal and meridional length-scales, respectively.

Algorithms 4 and 5 terminate after a fixed number of iterations K. In order to determine K, CI is first applied using a stopping criterion based on a specified convergence tolerance on the residual norm (see below). Unless stated otherwise, this stopping criterion will be used in the remainder of this paper to study and compare the convergence properties of CI for the SPD and NSPD systems.

The stopping criterion for each member m of the sequence of SPD linear systems (7) is defined according to the reduction of the ratio of the 2-norm of the residual on the kth iteration to the 2-norm of the initial residual:

$$\varepsilon_m^k = \frac{\left\| \widehat{\mathbf{A}} \boldsymbol{\psi}_m^k - \boldsymbol{\psi}_{m-1}^K \right\|_2}{\left\| \widehat{\mathbf{A}} \boldsymbol{\psi}_m^0 - \boldsymbol{\psi}_{m-1}^K \right\|_2} \tag{28}$$

where $\boldsymbol{\psi}^{K}_{m-1}$ denotes the approximate solution of the linear system associated with the previous member in the sequence. The stopping criterion for the NSPD system is also based on the reduction of the ratio of the 2-norms of the kth residual and initial residual:

$$\varepsilon_k = \frac{\left\| \boldsymbol{\mathcal{A}}\boldsymbol{\Psi}_k - \boldsymbol{\zeta} \right\|_2}{\left\| \boldsymbol{\mathcal{A}}\boldsymbol{\Psi}_0 - \boldsymbol{\zeta} \right\|_2}. \tag{29}$$

Contrary to Eq. (28), which measures the convergence of a given member of the sequence, Eq. (29) is a global measure of the convergence of all members in the pseudo-time interval. Sensitivity experiments using alternative stopping criteria are described in section 4.6.

4.2 Convergence properties

We start by considering the convergence behaviour of CI when solving the SPD system

$$\widehat{\mathbf{A}}\boldsymbol{\psi}_1 = \boldsymbol{\psi}_0 \quad \text{with} \quad \boldsymbol{\psi}^0_1 = \mathbf{0}. \tag{30}$$

Equation (30) can be considered as the first linear system ($m = 1$) in the sequence (7) associated with the operator $\widehat{\mathbf{L}}^{1/2}$ (Eq. (5)). The rhs is defined as $\boldsymbol{\psi}_0 = \mathbf{W}^{1/2}\mathbf{v}$ where $\mathbf{v}$ is a normally distributed random vector with zero mean and unit variance. The first guess ($\boldsymbol{\psi}^0_1$) is a zero field. Figure 1 shows the relative residual norm ε^k_m as a function of CI iteration for values of M' ranging from 2 to 10. The lowest value ($M' = 2$ or $M = 4$) corresponds to a fat-tailed covariance operator and the highest value ($M' = 10$ or $M = 20$) to a near-Gaussian covariance operator. The operational implementations of NEMOVAR at ECMWF and Met Office employ a value in between these extreme values ($M' = 5$ or $M = 10$). The same rhs has been used for all the experiments that use different values of M'. The convergence rate is linear and increases as M' increases, consistent with the behaviour of the condition number of $\widehat{\mathbf{A}}$, which decreases as M' increases (cf. Fig. 5 in WTP16). The convergence curves resulting from solving the remaining SPD systems in sequence,

$$\widehat{\mathbf{A}}\boldsymbol{\psi}_m = \boldsymbol{\psi}_{m-1} \quad \text{with} \quad \boldsymbol{\psi}^0_m = \mathbf{0}; \quad m = 2, \ldots, M', \tag{31}$$

are virtually identical to those for $m = 1$.

The solid lines in Fig. 1 show the corresponding values predicted by the rhs of the inequality (20). The logarithmic identity $\ln(y/x) = \log_{10}(y/x)\ln(10)$ has been used to express the rhs of (20) in terms of $\log_{10}(\varepsilon)$ in accordance with the vertical scale used in Fig. 1. The analytical estimates are in very close agreement with the actual values. The inequality (20) thus provides a tight upper bound on the number of iterations required to achieve a specified reduction of the relative residual norm. This suggests that, for the SPD system, the rhs of (20) can be used directly to determine K for the correlation model, without the need to apply CI first in a trial problem as recommended by WTP16.

The left panel in Fig. 2 shows the evolution of the relative residual norm ε_k as a function of CI iteration when solving the NSPD system (10):

$$\left.\begin{aligned} \boldsymbol{\mathcal{A}}\boldsymbol{\Psi} &= \mathbf{E}\boldsymbol{\psi}_0 \quad \text{with} \quad \boldsymbol{\Psi}_0 = \mathbf{0} \\ \boldsymbol{\psi}_{M'} &= \mathbf{F}^{\mathrm{T}}\boldsymbol{\Psi} \end{aligned}\right\} \tag{32}$$

where $\boldsymbol{\psi}_0$ is the same vector used for the experiments with the SPD system. The first guess ($\boldsymbol{\Psi}_0$) is a zero field as in the SPD case. The convergence of the NSPD system is slower than for the SPD system as it depends on the derivatives of the Chebyshev polynomials as well as the Chebyshev polynomials

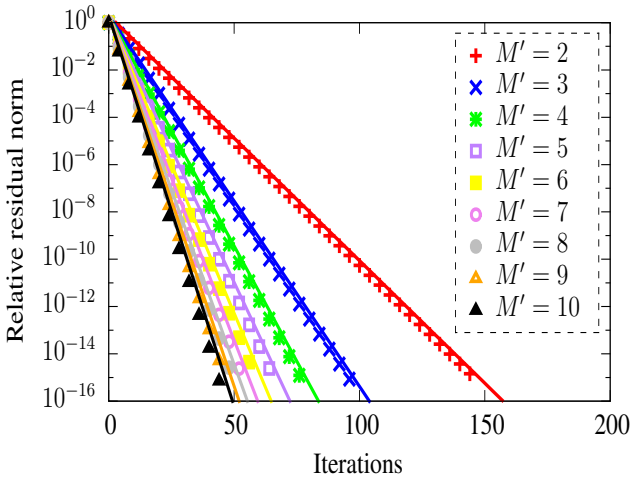


Figure 1: The residual norm relative to the initial residual norm as a function of CI iteration. The plotted symbols are the actual values obtained by solving the SPD system (30) for different values of M'. The plotted lines are the values estimated from the analytical expression (20). The vertical axis uses a logarithmic scale.

themselves (see Eqs (56) and (57) in appendix B). For increasing values of M', it displays an increasingly nonlinear behaviour in the early iterations, but a faster convergence rate in the later iterations. The nonlinear behaviour in the early iterations can be understood from the inequality (21), which will be influenced by the multiplicative term $\eta(\lambda_n, j)k^j$, particularly when $|\varphi_k(\lambda_n)|$ is not small as is the case in the early iterations. The faster convergence rate for larger M' when k is large is presumably related to the fact that the conditioning of the block components $\widehat{\mathbf{A}}$ improves as M' increases, and this aspect becomes dominant in determining the convergence behaviour in the later iterations.

The nonlinear behaviour is more clearly visible in the right panel in Fig. 2, which highlights the initial convergence behaviour. The scale on the vertical axis covers four orders of magnitude, which is an acceptable tolerance for a correlation model, as discussed earlier in section 3.1 and in WTP16. In the remainder of the paper, we will focus on this range when discussing the convergence properties of the solver.

Figure 3(a) shows the total number of CI iterations required to solve the sequence of M' SPD systems to a tolerance of $\varepsilon_m^k < 10^{-4}$ (black triangle symbols). The total number increases as M' increases even though each individual system requires fewer iterations to solve as M' increases. The corresponding values required to solve the NSPD system (32) to a tolerance of $\varepsilon_k < 10^{-4}$ are also shown (blue plus symbols). The total iteration counts remain relatively stable for different values of M', contrary to what is observed with the SPD system.

Figure 3(b) compares the total number of $\widehat{\mathbf{A}}$-matrix-vector products as a function of M'. From the perspective of Fig. 3(b), the NSPD formulation is of no interest compared to the SPD formulation especially for large values of M'. However, if the $\widehat{\mathbf{A}}$-matrix-vector products can be perfectly parallelized in the "time" dimension (m) on each CI iteration then the overall run-time can be reduced by a factor equal to the ratio of the total iteration counts required by the SPD and NSPD formulations. Figure 3(a) then suggests that the NSPD formulation can provide a potential speed-up by a factor with an upper bound between two and three for the values of M' considered. From this "time"-parallel perspective, there is a

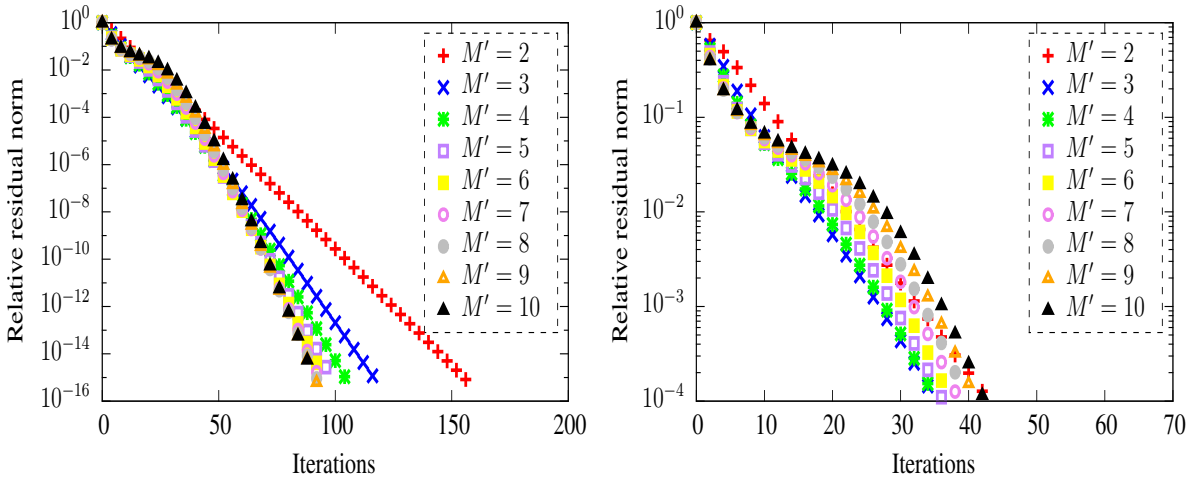


Figure 2: The residual norm relative to the initial residual norm as a function of CI iteration when solving the NSPD system (32) for different values of M'. The rhs is $\boldsymbol{\zeta} = \mathbf{E}\boldsymbol{\psi}_0$ where $\boldsymbol{\psi}_0$ is the same as that used for solving the SPD system in Fig. 1. The vertical axis uses a logarithmic scale. The right panel shows the same curves as the left panel but with a refined vertical scale to focus on the initial convergence behaviour.

clear advantage with the NSPD formulation, as discussed further in sections 4.7 and 4.8.

4.3 Sensitivity to the initial residual

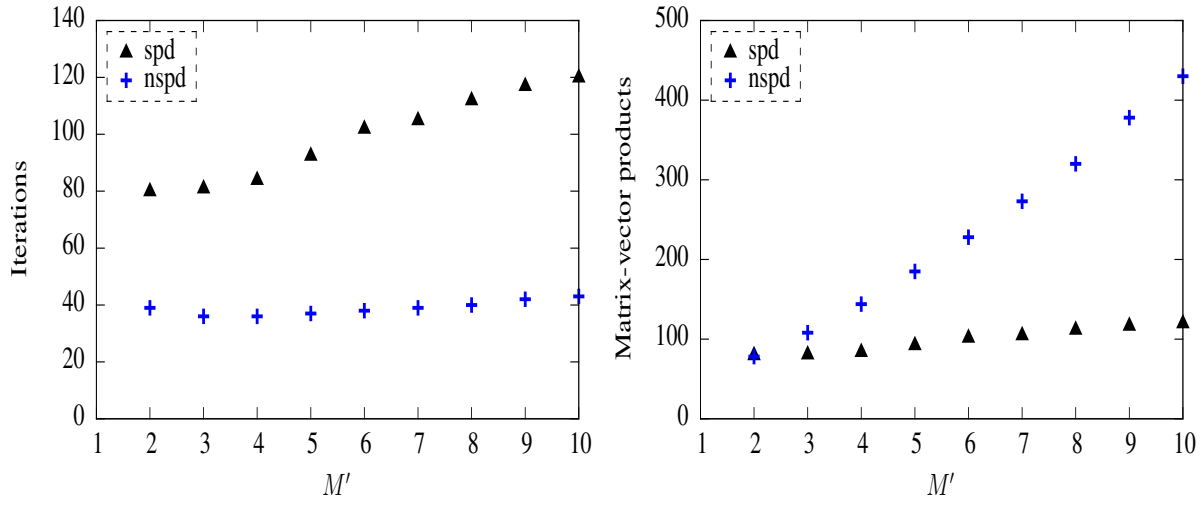
For the SPD systems, the CI convergence properties are insensitive to the initial residual. This is not the case with the NSPD system, however, as illustrated in this section. The initial residual depends on the rhs and the choice of first guess.

We consider initially the sensitivity to changes to the rhs, while continuing to use a zero field for the first guess. Figure 4 shows the convergence curves when the NSPD system is solved with a rhs taken to be the solution $\boldsymbol{\psi}_{M'} = \mathbf{F}^{\mathrm{T}}\boldsymbol{\Psi}$ of Eq. (32):

$$\left.\begin{aligned} \boldsymbol{\mathcal{A}}\boldsymbol{\Psi} &= \mathbf{E}\boldsymbol{\psi}_{M'} \quad \text{with} \quad \boldsymbol{\Psi}_0 = \mathbf{0} \\ \boldsymbol{\psi}_M &= \mathbf{F}^{\mathrm{T}}\boldsymbol{\Psi}. \end{aligned}\right\}. \tag{33}$$

The solution of Eq. (33) corresponds to the result of applying the complete diffusion operator $\widehat{\mathbf{L}} = \widehat{\mathbf{L}}^{1/2}\widehat{\mathbf{L}}^{1/2}$. The plateau-like feature in the relative residual norm appears earlier in the iterative process and the total iteration counts increase as a result (cf. right panel in Fig. 2).

Figures 5(a) and 5(b) show the evolution of the relative residual when CI is initialized with a first guess taken to be a non-zero transformation $\mathcal{P}_{\mathrm{I}}$ of the rhs. The different panels are associated with each of the



(a) Total iteration counts for $\widehat{\mathbf{L}}^{1/2}$

(b) Total $\widehat{\mathbf{A}}$-matrix-vector products for $\widehat{\mathbf{L}}^{1/2}$

Figure 3: (a) The total number of CI iterations required to solve the M' SPD systems (black diamond symbols) and the single NSPD system with M' time-blocks (blue plus symbols) in $\widehat{\mathbf{L}}^{1/2}$. A convergence tolerance of $\varepsilon_m^k < 10^{-4}$ (Eq. (28)) is used for the SPD system, and $\varepsilon_k < 10^{-4}$ (Eq. (29)) for the NSPD system. The first guess is a zero field. (b) The total number of $\widehat{\mathbf{A}}$-matrix-vector products associated with the iteration counts in (a). Values are plotted as a function of M'. The ratio of the iterations required for the SPD problem and those required for the NSPD problem in panel (a) indicates the potential run-time speed-up that can be achieved by performing the M' $\widehat{\mathbf{A}}$-matrix-vector products in parallel.

NSPD systems in the two-member sequence describing the complete diffusion operator $\widehat{\mathbf{L}} = \widehat{\mathbf{L}}^{1/2}\widehat{\mathbf{L}}^{1/2}$:

$$\left.\begin{aligned} \boldsymbol{\mathcal{A}}\boldsymbol{\Psi} &= \mathbf{E}\,\boldsymbol{\psi}_0 \quad \text{with} \quad \boldsymbol{\Psi}_0 = \boldsymbol{\mathcal{P}}_{\mathbf{I}}\mathbf{E}\,\boldsymbol{\psi}_0 \\ \boldsymbol{\psi}_{M'} &= \mathbf{F}^{\mathrm{T}}\boldsymbol{\Psi} \end{aligned}\right\}, \tag{34}$$

$$\left.\begin{aligned} \boldsymbol{\mathcal{A}}\boldsymbol{\Psi} &= \mathbf{E}\,\boldsymbol{\psi}_{M'} \quad \text{with} \quad \boldsymbol{\Psi}_0 = \boldsymbol{\mathcal{P}}_{\mathbf{I}}\mathbf{E}\,\boldsymbol{\psi}_{M'} \\ \boldsymbol{\psi}_{M} &= \mathbf{F}^{\mathrm{T}}\boldsymbol{\Psi} \end{aligned}\right\}. \tag{35}$$

Using the transformation $\boldsymbol{\mathcal{P}}_{\mathbf{I}}$ to define $\boldsymbol{\Psi}_0$ amounts to initializing the state on each time level with the non-zero rhs available on the first step of each sequence (see Eq. (25)). This choice of first guess results in a systematic reduction in the number of iterations compared to the number required with a zero first guess, for all values of M' considered except for $M' = 2$ which requires slightly more iterations for the first NSPD system (cf. blue plus and red triangle symbols in Figs 6(a) and 6(b)). Other choices of first guess were less successful in reducing the number of iteration counts (see Table 2).

Since the convergence behaviour of the NSPD system in the early iterations is sensitive to the initial residual, it raises the question of how to define the value of the fixed iteration K in Algorithms 4 and 5. For a fixed tolerance, the number of iterations required to solve the first and second NSPD systems in the factored form of $\widehat{\mathbf{L}}$ can differ by several iterations, with the second system generally requiring more iterations than the first system for the stopping criterion (29) (see the values of K_1 and K_2 in Table 2). A reasonable strategy is to choose K as the arithmetic mean of these two values, and to use this value of K in both the forward and adjoint CI algorithms to enforce exact numerical symmetry of $\widehat{\mathbf{L}}$. In variational data assimilation, K must be diagnosed prior to entering the minimization (inner) loop.

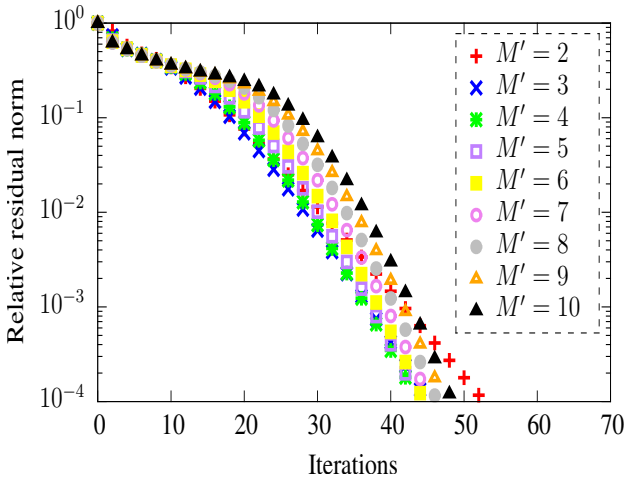


Figure 4: As the right panel in Fig. 2 but for the NSPD system (33) which has rhs $\boldsymbol{\zeta} = \mathbf{E}\boldsymbol{\psi}_{M'}$ where $\boldsymbol{\psi}_{M'}$ is the solution of the NSPD system (32). This figure thus illustrates the convergence behaviour of the NSPD system in the second component of the complete diffusion operator $\widehat{\mathbf{L}} = \widehat{\mathbf{L}}^{1/2}\widehat{\mathbf{L}}^{1/2}$.

4.4 Solution accuracy

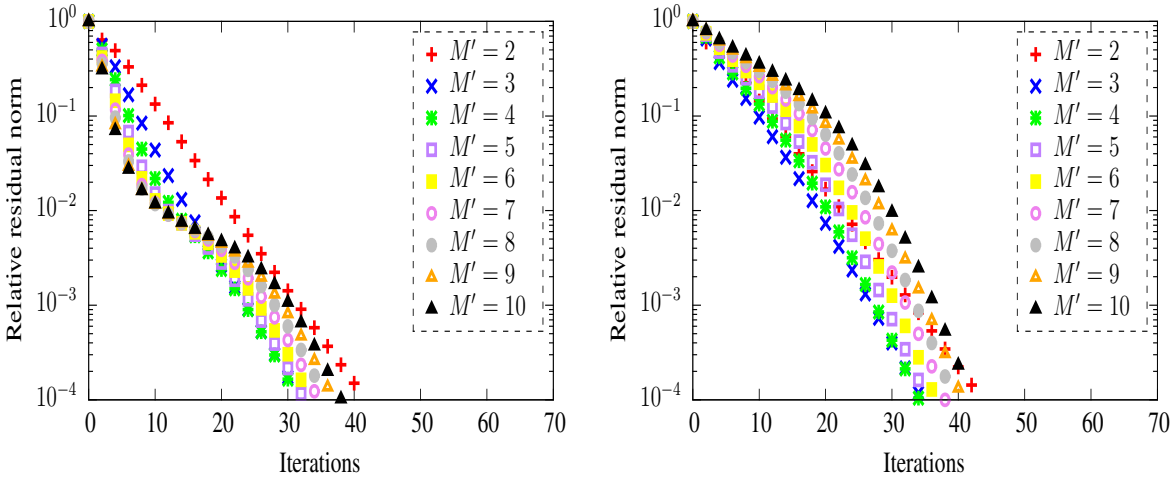
The solution of interest is the final state $\boldsymbol{\psi}_M$ obtained by applying the full $\widehat{\mathbf{L}}$ operator to $\boldsymbol{\psi}_0$. To assess the accuracy of the solution for a given stopping criterion, we examine the relative error norm

$$\varepsilon^* = \frac{\left\| \boldsymbol{\psi}_M - \boldsymbol{\psi}_M^* \right\|_2}{\left\| \boldsymbol{\psi}_M^* \right\|_2} \tag{36}$$

where $\boldsymbol{\psi}_M^*$ is the "exact" solution on step M, obtained by solving each member of the SPD sequence to a very high accuracy, here taken as $\varepsilon_m^k < 10^{-15}$. Solving the corresponding NSPD systems to an accuracy of $\varepsilon_k < 10^{-15}$ results in a nearly identical solution to $\boldsymbol{\psi}_M^*$. It is more interesting to assess the solution accuracy when a more moderate tolerance is specified, similar to one that would be used in practice for the correlation model. Consider the case when the stopping tolerances ε_m^k and ε_k are set to 10^{-4}. The relative error ε^* is shown for different M in Fig. 7 for both the SPD-based and NSPD-based formulations of $\widehat{\mathbf{L}}$. The results are displayed for two choices of the first guess: a zero field and a non-zero field determined from the rhs (see figure legend). The non-zero first-guess fields that have been employed in these figures correspond to the ones that gave best results of all those considered (see Tables 1 and 2).

Several features can be observed from these figures. With a zero first guess, the SPD formulation produces a more accurate solution than the NSPD formulation (cf. black triangle and blue cross symbols). Whereas the solution errors with the NSPD formulation are relatively stable as a function of M, those with the SPD formulation (and zero first guess) increase steadily with M, probably due to an accumulation of errors affecting the rhs of the sequence of SPD systems. When $M = 20$ the SPD and NSPD formulations have similar accuracy.

Compared to the results with zero first guess, the use of the rhs as first guess with the SPD system leads to a significantly more accurate solution (up to an order of magnitude) and successfully counteracts the accumulated-error effect that occurs with large values of M (cf. black diamonds and orange circles). The accuracy of the iterative solution of the NSPD system is also significantly improved when using a first guess that exploits information about the rhs (cf. red triangle with blue cross symbols). Here, results



(a) $\boldsymbol{\zeta} = \mathbf{E}\boldsymbol{\psi}_0$ (b) $\boldsymbol{\zeta} = \mathbf{E}\boldsymbol{\psi}_{M'}$

Figure 5: (a) As in the right panel in Fig. 2 but with the first guess defined as $\boldsymbol{\Psi}_0 = \mathcal{P}_\mathrm{I}\mathbf{E}\boldsymbol{\psi}_0$ (see Eq. (34)). (b) As Fig. 4 but with the first guess defined as $\boldsymbol{\Psi}_0 = \mathcal{P}_\mathrm{I}\mathbf{E}\boldsymbol{\psi}_{M'}$ (see Eq. (35)).

with $\boldsymbol{\Psi}_0 = \mathcal{P}_\mathrm{I}\boldsymbol{\zeta}$ are displayed; other choices of non-zero first guess were also beneficial but not to the same extent (see Table 2).

For both the SPD- and NSPD-based formulations, these results suggest that the computational cost of the algorithm can be reduced by relaxing the stopping tolerance (reducing the number of iterations), while still achieving an acceptable solution accuracy using a non-zero first guess. This is evident from Tables 1 and 2, which show that the solution error resulting from using an increased tolerance (10^{-3}) and rhs-based first guess is comparable to the error obtained using a reduced tolerance (10^{-4}) and zero first guess (cf. ε^* in rows 1, 2 and 3 in Table 1 for the SPD formulation; ε^* in rows 1, 4 and 5 in Table 2 for the NSPD formulation).

4.5 Preconditioning

In this section we examine the impact of employing the simple preconditioners described in section 3.3. First, consider the diagonal preconditioner $\mathbf{D}^{-1}$ for the SPD problem. This preconditioner does not change the total iteration counts compared to the unpreconditioned case (see Table 1). The convergence properties of CI for the SPD problem are linked to the condition number of the SPD matrix (Gergelits and Strakoš, 2014). In the experiments, the condition number estimated from the extremal Ritz values is unaffected by this preconditioner (not shown). This can be explained by the following simplified analysis.

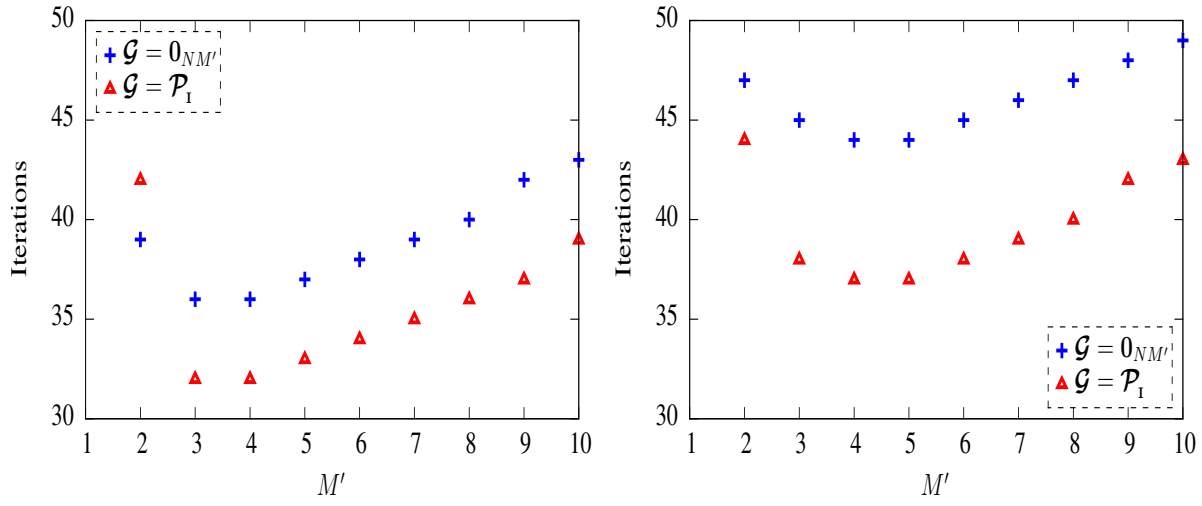
Consider the time-implicit discretization of the 3D diffusion equation with constant diffusion coefficients $(\kappa_1, \kappa_2, \kappa_3)$. Assuming that the spatial derivatives are discretized using centred finite differences and that the grid steps (e_1, e_2, e_3) are constant then it is straightforward to compute the eigenvalues of the associated system matrix following the approach described in section 3.2 of WTP16. The extreme eigenvalues are $\lambda_{\min} \approx 1$ and $\lambda_{\max} \approx 1 + 4X$ where $X = \kappa_1/e_1^2 + \kappa_2/e_2^2 + \kappa_3/e_3^2$. The condition number is then $\chi \approx 1 + 4X$. When the system matrix is preconditioned by $\mathbf{D}^{-1}$, the extreme eigenvalues become

$\mathbf{G}$	$\mathbf{P}$	ε_m^k	ε^*	K	$\widehat{\mathbf{A}}$-prod
$\mathbf{0}_N$	$\mathbf{I}_N$	10^{-4}	8.0×10^{-4}	18	180
$\mathbf{I}_N$	$\mathbf{I}_N$	10^{-4}	1.1×10^{-4}	18	180
$\mathbf{I}_N$	$\mathbf{I}_N$	10^{-3}	9.7×10^{-4}	14	140
$\mathbf{D}^{-1}$	$\mathbf{I}_N$	10^{-4}	5.6×10^{-4}	18	180
$\mathbf{I}_N$	$\mathbf{D}^{-1}$	10^{-4}	6.0×10^{-4}	18	180

Table 1: A summary of results from experiments with the SPD formulation of $\widehat{\mathbf{L}}$ (Eq. (7)) and different first-guess fields, as defined by the transformation (27) acting on the rhs (first column), and different preconditioners (second column) for the case $M = 10$. The third column gives the convergence tolerance (Eq. (28)). The fourth column gives the solution accuracy ε^* achieved after the final (Mth) diffusion step (Eq. (36)). The fifth column gives the number of iterations required to satisfy the tolerance indicated in the third column. The sixth column gives the total number of $\widehat{\mathbf{A}}$-matrix-vector products ($= K \times M$) required by $\widehat{\mathbf{L}}$.

$\mathcal{G}$	$\mathcal{P}$	ε_k	ε^*	K_1	K_2	$\widehat{\mathbf{A}}$-prod
$\mathbf{0}_{NM'}$	$\mathbf{I}_{NM'}$	10^{-4}	2.3×10^{-3}	37	44	405
$\mathbf{I}_{NM'}$	$\mathbf{I}_{NM'}$	10^{-4}	0.5×10^{-3}	36	38	370
$\widetilde{\mathcal{P}}_{\mathrm{D}}$	$\mathbf{I}_{NM'}$	10^{-4}	1.3×10^{-3}	38	44	410
$\mathcal{P}_{\mathrm{I}}$	$\mathbf{I}_{NM'}$	10^{-4}	0.3×10^{-3}	33	37	350
$\mathcal{P}_{\mathrm{I}}$	$\mathbf{I}_{NM'}$	10^{-3}	4.7×10^{-3}	25	34	295
$\mathcal{P}_{\mathrm{D}}$	$\mathbf{I}_{NM'}$	10^{-4}	0.9×10^{-3}	39	45	420
$\mathcal{P}_{\mathrm{I}}$	$\widetilde{\mathcal{P}}_{\mathrm{D}}$	10^{-4}	0.3×10^{-3}	33	37	350
$\mathcal{P}_{\mathrm{I}}$	$\mathcal{P}_{\mathrm{I}}$	10^{-4}	0.1×10^{-3}	61	67	640
$\mathcal{P}_{\mathrm{I}}$	$\mathcal{P}_{\mathrm{D}}$	10^{-4}	0.2×10^{-3}	33	36	345
$\mathcal{P}_{\mathrm{I}}$	$\mathcal{P}_{\mathrm{D}}$	10^{-3}	1.8×10^{-3}	26	34	300

Table 2: A summary of results from experiments with the NSPD formulation of $\widehat{\mathbf{L}}$ (Eq. (10)) and different first guess fields, as defined by the transformations (25) or (26) acting on the rhs (first column), and different preconditioners (second column) for the case $M = 10$. The third column indicates the specified convergence tolerance (Eq. (29)). The fourth column gives the solution accuracy ε^* after the final (Mth) diffusion step (Eq. (36)). The fifth (sixth) column gives the number of iterations required to reach the specified tolerance for the first (second) NSPD system associated with the square-root factors of $\widehat{\mathbf{L}}$. The last column gives the total number of $\widehat{\mathbf{A}}$-matrix-vector products ($= (K_1 + K_2) \times M'$) required by $\widehat{\mathbf{L}}$.



(a) NSPD systems (32) and (34)

(b) NSPD systems (33) and (35)

Figure 6: The total number of CI iterations required to solve (a) the first NSPD system and (b) the second NSPD system in the factored operator $\widehat{\mathbf{L}} = \widehat{\mathbf{L}}^{1/2}\widehat{\mathbf{L}}^{1/2}$ where each NSPD matrix has M' time-blocks. A stopping criterion of $\varepsilon_k < 10^{-4}$ is used. Values are plotted as a function of M'. The different symbols correspond to different choices of the first guess as indicated by the transformation in the legend (see Eq. (25)).

$\lambda_{\min} \approx 1/(1+2X)$ and $\lambda_{\max} \approx (1+4X)/(1+2X)$, which are identical to those of the unpreconditioned system up to a constant scaling factor $1/(1+2X)$. The condition number, however, remains unchanged.

For the NSPD problem, the results with different preconditioners for the case $M' = 5$ are summarized in Table 2. The diagonal preconditioner $\widetilde{\mathcal{P}}_{\mathrm{D}}$ has no impact as might be expected in view of the results above for the diagonally preconditioned SPD problem. The simple block lower triangular preconditioner $\mathcal{P}_{\mathrm{I}}$ has a detrimental impact as can be seen by the increased number of iterations required to satisfy the specified tolerance. Nevertheless, the solution is more accurate than the solutions obtained with the other preconditioners, which suggests that the stopping criterion is inappropriate with this preconditioner. This point will be revisited in the next section.

The block lower triangular preconditioner $\mathcal{P}_{\mathrm{D}}$ has only a small impact for the case $M' = 5$ considered in the table. Contrary to $\mathcal{P}_{\mathrm{I}}$, however, it does not degrade performance. In this respect, it is likely that the block matrices $\mathbf{D}^{-m'}$, for $m' < m$, in $\mathcal{P}_{\mathrm{D}}$ are important for properly scaling (downweighting) the contributions from the components $\boldsymbol{\psi}_{m'}$ on previous steps m' in determining the effect of the preconditioner on the component $\boldsymbol{\psi}_m$ on the current step m.

The impact of $\mathcal{P}_{\mathrm{D}}$ for other values of M' is examined in Figures 8(a) and 8(b), which show, as a function of M', the total number of iterations required to solve each of the coupled systems associated with $\widehat{\mathbf{L}} = \widehat{\mathbf{L}}^{1/2}\widehat{\mathbf{L}}^{1/2}$:

$$\left.\begin{aligned} \mathcal{A}\mathcal{P}_{\mathrm{D}}\boldsymbol{\Psi}' &= \mathbf{E}\boldsymbol{\psi}_0 \;\text{ with }\; \boldsymbol{\Psi}'_0 = \mathcal{P}_{\mathrm{D}}^{-1}\mathcal{P}_{\mathrm{I}}\mathbf{E}\boldsymbol{\psi}_0 \\ \boldsymbol{\psi}_{M'} &= \mathbf{F}^{\mathrm{T}}\mathcal{P}_{\mathrm{D}}\boldsymbol{\Psi}' \end{aligned}\right\}, \tag{37}$$

$$\left.\begin{aligned} \mathcal{A}\mathcal{P}_{\mathrm{D}}\boldsymbol{\Psi}' &= \mathbf{E}\boldsymbol{\psi}_{M'} \;\text{ with }\; \boldsymbol{\Psi}'_0 = \mathcal{P}_{\mathrm{D}}^{-1}\mathcal{P}_{\mathrm{I}}\mathbf{E}\boldsymbol{\psi}_{M'} \\ \boldsymbol{\psi}_{M} &= \mathbf{F}^{\mathrm{T}}\mathcal{P}_{\mathrm{D}}\boldsymbol{\Psi}' \end{aligned}\right\}. \tag{38}$$

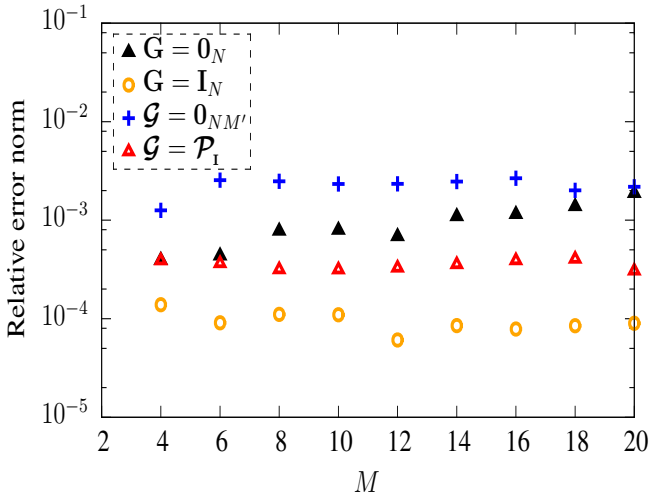


Figure 7: The relative norm of the error in the solution on the final step (Eq. (36)) when solving the sequence of M SPD systems with a stopping criterion of $\varepsilon_m^k < 10^{-4}$ (Eq. (28)), and when solving the sequence of two NSPD systems with a stopping criterion of $\varepsilon_k < 10^{-4}$ (Eq. (29)). The different symbols correspond to different choices of the first guess as indicated by the transformation in the legend (see Eq. (25)). The true solution is obtained by solving the sequence of M SPD systems to a very high accuracy ($\varepsilon_m^k < 10^{-15}$). Values are plotted as a function of M. The vertical axes use a logarithmic scale.

These are similar to systems (34) and (35) but are right-preconditioned by $\mathcal{P}_{\mathrm{D}}$. The inverse transformation $\mathcal{P}_{\mathrm{D}}^{-1}$ has been introduced so that the first guess is $\boldsymbol{\Psi}_0 = \mathcal{P}_{\mathrm{I}}\mathbf{E}\,\boldsymbol{\psi}_0$, as desired. Notice, however, that $\mathcal{P}_{\mathrm{D}}^{-1}$ is not required by Algorithm 4 since this algorithm is initialized directly with the residual $\boldsymbol{\Xi}_0$. Figures 8(a) and 8(b) indicate that $\mathcal{P}_{\mathrm{D}}$ is particularly beneficial for values of $M' > 5$ and somewhat neutral for values of $M' \leq 5$. Figures 9(a) and 9(b) show that the convergence curves with $\mathcal{P}_{\mathrm{D}}$ are noticeably more linear than the corresponding curves for the unpreconditioned problem for all values of M' considered (cf. Figs 5(a) and 5(b)).

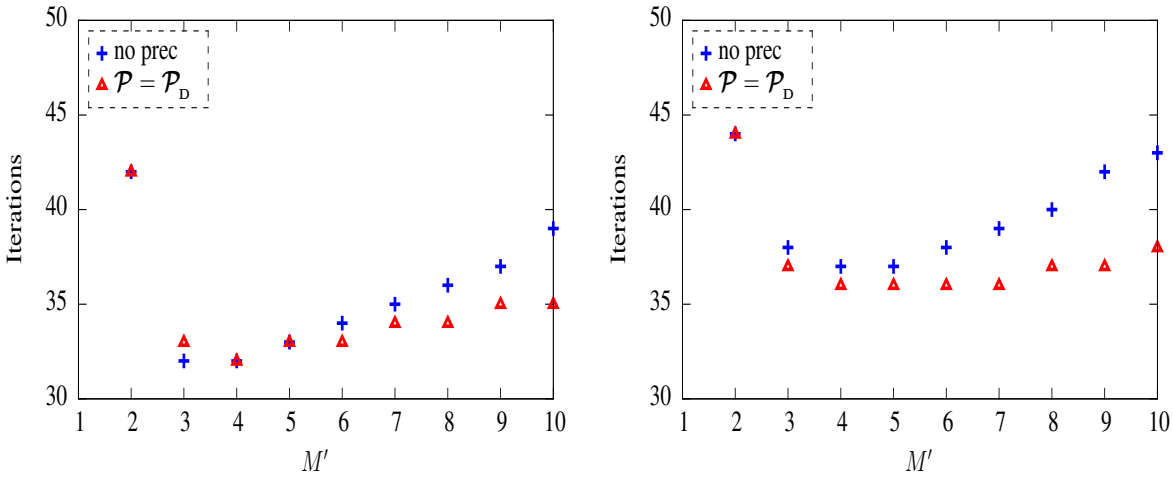
4.6 Stopping criterion to determine K

Results from the previous section suggest that there is no clear relationship between the residual 2-norm and the solution accuracy as defined by ε^* in Eq. (36). As such, a stopping criterion based on the residual 2-norm might not always be appropriate for estimating the number of CI iterations K required to solve the NSPD system to a desired accuracy. In particular, results showed that it is clearly inappropriate with the (albeit poor) right-preconditioner $\mathcal{P}_{\mathrm{I}}$.

Other stopping criteria are possible. For the preconditioned problem, it is possible to define the stopping criterion based on the $\mathcal{P}^{\mathrm{T}}\mathcal{P}$-norm of the residual instead of the 2-norm of the residual. This is a natural stopping criterion with the left-preconditioner since

$$\left\|\mathcal{P}\mathcal{A}\boldsymbol{\Psi}_k - \mathcal{P}\boldsymbol{\zeta}\right\|_2 = \left\|\mathcal{A}\boldsymbol{\Psi}_k - \boldsymbol{\zeta}\right\|_{\mathcal{P}^{\mathrm{T}}\mathcal{P}} \tag{39}$$

where $\|(\cdot)\|_{\mathcal{P}^{\mathrm{T}}\mathcal{P}} = \sqrt{(\cdot)^{\mathrm{T}}\mathcal{P}^{\mathrm{T}}\mathcal{P}(\cdot)}$. If $\mathcal{P} = \mathcal{A}^{-1}$ then $\|\mathcal{A}\boldsymbol{\Psi}_k - \boldsymbol{\zeta}\|_{\mathcal{P}^{\mathrm{T}}\mathcal{P}} = \|\boldsymbol{\Psi}_k - \boldsymbol{\Psi}^*\|_2$ where $\boldsymbol{\Psi}^* = \mathcal{A}^{-1}\boldsymbol{\zeta}$ is the exact solution. This limiting case shows that the $\mathcal{P}^{\mathrm{T}}\mathcal{P}$-norm is an approximate measure of the 2-norm of the solution error. Another possibility is to use a stopping criterion based on the 2-norm of



(a) NSPD systems (34) and (37)

(b) NSPD systems (35) and (38)

Figure 8: As Figs 6(a) and 6(b) but with preconditioning (red triangles). For comparison, the values without preconditioning ($\mathcal{P} = \mathbf{I}_{NM'}$) are also shown (blue crosses). In both experiments, the first guess is defined according to the rhs transformation $\mathcal{G} = \mathcal{P}_1$. The blue crosses are equivalent to the red triangles in Figs 6(a) and 6(b).

the *actual* solution error, which first requires an accurate solve of the system to estimate $\boldsymbol{\Psi}^*$. Since we are only interested in the solution on the final step M', we may also consider the targeted variant of the solution error, $\left\| \boldsymbol{\psi}^k_{M'} - \boldsymbol{\psi}^*_{M'} \right\|_2$.

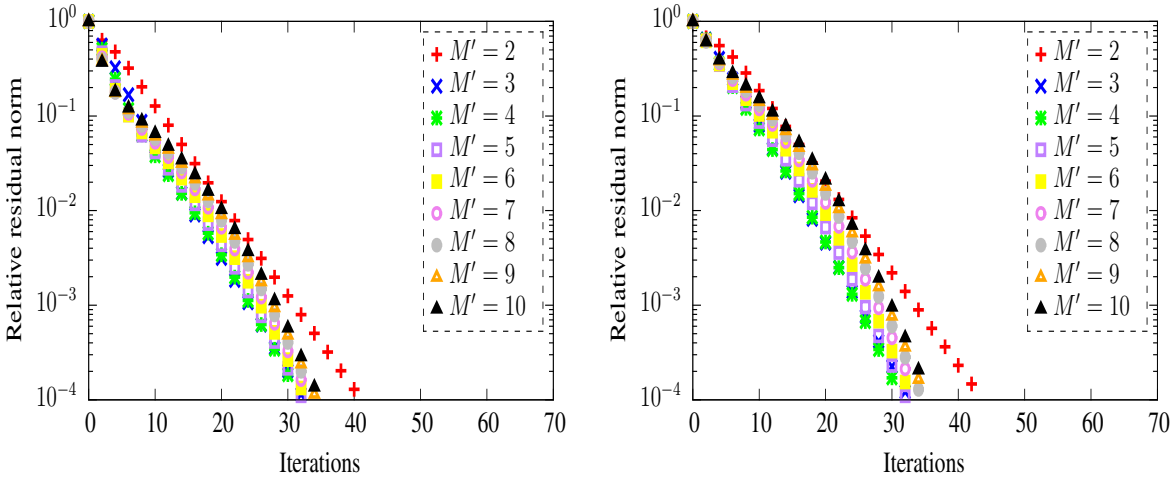
The different choices of the stopping criterion that have been considered are summarized in Table 3. They are distinguished by the expression for the tolerance $\varepsilon^{(n)}_k = a/b$ where the argument of the norm in a is based on either the residual or solution error. For the residual-based norms, the argument of the norm in b is either the initial residual or rhs. For the solution-error-based norms, the argument of the norm in b is either the initial solution error or the exact solution. With a zero first guess, the initial residual (initial solution error) and rhs (solution) are equivalent. With a non-zero first guess, measuring a relative to the initial residual norm or initial solution error norm is consistent with the minimization properties of CI (see section 5), while measuring a relative to the norm of the signal (rhs or solution) would seem preferable when the first guess is already "close" to the solution.

Table 4 summarizes the results from using the different stopping criteria to solve Eqs (37) and (38) to a tolerance of $\varepsilon^{(n)}_k < 10^{-3}$. The basic experiment is the same as the one associated with the last row in Table 2 that uses the baseline stopping criterion (Eq. (29)). Those results have been repeated in the first row of Table 4 for easy reference. The different stopping criteria produce different total iteration counts, with those based on the solution-error norm leading to fewer iterations than those based on the residual norm (cf. cases $n = \{1,3,4\}$ and cases $n = \{2,5,7\}$). Satisfying the targeted solution-error norm (case $n = 8$) requires the most iterations and as a result leads to the smallest error ε^*. Note that for this experiment, ε^* is equivalent to the stopping criterion itself.

When the normalization factor b is defined by the initial residual norm (cases $n = \{1,3\}$) or initial error norm (cases $n = \{5,7\}$), fewer iterations are required to solve the first system than the second system ($K_1 < K_2$). The opposite ($K_2 < K_1$) occurs when b is defined by the norm of the rhs (cases $n = \{2,4\}$) or the norm of the exact solution (cases $n = \{6,8\}$), presumably since the first guess for the second system,

$\varepsilon_k^{(n)} = a/b$		
(n)	a	b
(1)	$\left\|\boldsymbol{\mathcal{A}}\boldsymbol{\Psi}_k - \boldsymbol{\zeta}\right\|_2$	$\left\|\boldsymbol{\mathcal{A}}\boldsymbol{\Psi}_0 - \boldsymbol{\zeta}\right\|_2$
(2)	$\left\|\boldsymbol{\mathcal{A}}\boldsymbol{\Psi}_k - \boldsymbol{\zeta}\right\|_2$	$\left\|\boldsymbol{\zeta}\right\|_2$
(3)	$\left\|\boldsymbol{\mathcal{A}}\boldsymbol{\Psi}_k - \boldsymbol{\zeta}\right\|_{\boldsymbol{\mathcal{P}}^{\mathrm{T}}\boldsymbol{\mathcal{P}}}$	$\left\|\boldsymbol{\mathcal{A}}\boldsymbol{\Psi}_0 - \boldsymbol{\zeta}\right\|_{\boldsymbol{\mathcal{P}}^{\mathrm{T}}\boldsymbol{\mathcal{P}}}$
(4)	$\left\|\boldsymbol{\mathcal{A}}\boldsymbol{\Psi}_k - \boldsymbol{\zeta}\right\|_{\boldsymbol{\mathcal{P}}^{\mathrm{T}}\boldsymbol{\mathcal{P}}}$	$\left\|\boldsymbol{\zeta}\right\|_{\boldsymbol{\mathcal{P}}^{\mathrm{T}}\boldsymbol{\mathcal{P}}}$
(5)	$\left\|\boldsymbol{\Psi}_k - \boldsymbol{\Psi}^*\right\|_2$	$\left\|\boldsymbol{\Psi}_0 - \boldsymbol{\Psi}^*\right\|_2$
(6)	$\left\|\boldsymbol{\Psi}_k - \boldsymbol{\Psi}^*\right\|_2$	$\left\|\boldsymbol{\Psi}^*\right\|_2$
(7)	$\left\|\boldsymbol{\psi}_{M'}^k - \boldsymbol{\psi}_{M'}^*\right\|_2$	$\left\|\boldsymbol{\psi}_{M'}^0 - \boldsymbol{\psi}_{M'}^*\right\|_2$
(8)	$\left\|\boldsymbol{\psi}_{M'}^k - \boldsymbol{\psi}_{M'}^*\right\|_2$	$\left\|\boldsymbol{\psi}_{M'}^*\right\|_2$

Table 3: A summary of the stopping criteria that have been used for determining the number of fixed iterations K for solving the NSPD system with CI. Note that $\varepsilon_k^{(1)} = \varepsilon_k$ in Eq. (29).



(a) $\boldsymbol{\zeta} = \mathbf{E}\boldsymbol{\psi}_0$ (b) $\boldsymbol{\zeta} = \mathbf{E}\boldsymbol{\psi}_{M'}$

Figure 9: (a) As Figs 5(a) and 5(b) but with $\mathcal{P}_{\mathrm{D}}$ used as preconditioning matrix (see Eqs (37) and (38)).

which is based on the solution after $M/2$ steps, is already a decent estimate of the solution of the full M-step diffusion problem. Interestingly, however, the average values of K_1 and K_2 are the same or similar with both types of normalization. For the practical purpose of determining K for the correlation model, both types of normalization can then be expected to give similar results.

4.7 Hybrid time-sequential/time-parallel formulation

Equation (14) describes a hybrid formulation of $\widehat{\mathbf{L}}^{1/2}$ involving a combination of time-sequential and time-parallel operations. Results using the allowable hybrid combinations when $M' = 5$ are displayed in Table 5, where the second column indicates the number of time-blocks m_l in the NSPD matrices $\boldsymbol{\mathcal{A}}_{m_l}$ on each of the $l = 1, \ldots, L$ sequential steps. In order to quantify the potential run-time improvement that can be achieved by parallelizing the $\widehat{\mathbf{A}}$-matrix-vector products in the different formulations, we define the following average speed-up factor:

$$\text{speed-up} = \frac{1}{L}\sum_{l=1}^{L}\left(\frac{m_l K_{\mathrm{s}}}{\widehat{K}_{m_l}}\right) \tag{40}$$

where K_{s} is the number of iterations required to solve the system with the SPD matrix ($K_{\mathrm{s}} = 18$ in Table 5), and $\widehat{K}_{m_l}$ is the number of iterations required to solve the system with the NSPD matrix $\boldsymbol{\mathcal{A}}_{m_l}$. The speed-up factor is bounded below by 1 with the purely time-sequential formulation and bounded above by $M'K_{\mathrm{s}}/\widehat{K}_{M'}$ with the purely time-parallel formulation. The maximum speed-up factor for $M' = 5$ is 2.7 and reaches 2.9 when $\widehat{K}_{M'}$ is determined using the solution-error stopping criterion $\varepsilon_k^{(5)}$ (in this case the solver terminates after 31 iterations instead of 34 iterations when using the residual stopping criterion $\varepsilon_k^{(1)}$). Table 5 shows that useful speed-up factors can also be achieved with the hybrid formulations $1 < L < M'$.

Figure 10 shows the reduction of the relative residual norm for the systems with different values of m_l. The convergence behaviour is very similar during the first 5 iterations, which results in a one-order of

$\varepsilon_k^{(n)}$	ε^*	K_1	K_2	K	$\widehat{\mathbf{A}}$-prod
$\varepsilon_k^{(1)}$	1.8×10^{-3}	26	34	30	300
$\varepsilon_k^{(2)}$	1.2×10^{-3}	34	26	30	300
$\varepsilon_k^{(3)}$	2.7×10^{-3}	25	30	27	275
$\varepsilon_k^{(4)}$	4.3×10^{-3}	33	22	27	275
$\varepsilon_k^{(5)}$	4.3×10^{-3}	24	27	25	255
$\varepsilon_k^{(6)}$	2.6×10^{-3}	30	24	27	270
$\varepsilon_k^{(7)}$	2.3×10^{-3}	26	28	27	270
$\varepsilon_k^{(8)}$	0.8×10^{-3}	35	27	31	310

Table 4: Results from solving Eqs (37) and (38) (where $\mathcal{G}=\mathcal{P}_{\mathbf{I}}$ and $\mathcal{P}=\mathcal{P}_{\mathbf{D}}$) for the case $M=10$, using the different stopping criteria described in Table 3. A tolerance of $\varepsilon_k^{(n)}<10^{-3}$ has been used in all $n=1,\ldots,8$ cases. Columns 2-4 and 6 are as described in the caption for Table 2. Column 5 is the average of K_1 and K_2.

L	m_l	$\widehat{K}_{m_l}$	speed-up
5	1,1,1,1,1	18,18,18,18,18	1
4	1,1,1,2	18,18,18,21	1.2
3	1,2,2	18,20,21	1.5
3	1,1,3	18,18,23	1.6
2	2,3	20,23	2.1
2	1,4	18,28	1.8
1	5	34	2.7

Table 5: A summary of results from experiments with the hybrid sequential/parallel formulation of $\widehat{\mathbf{L}}^{1/2}$ for the case $M'=5$. The first guess and preconditioning transformations are $\mathcal{G}=\mathcal{P}_{\mathbf{I}}$ and $\mathcal{P}=\mathcal{P}_{\mathbf{D}}$. The first column indicates the number of sequential steps (Eq. (14)). The second column indicates the number of time-blocks in the NSPD matrix on each of the $l=1,\ldots,L$ sequential steps. The third column gives the number of iterations required to satisfy the tolerance $\varepsilon_k<10^{-4}$ on each of the sequential steps. It is taken as the average of the number of iterations required on the corresponding sequential step of each square-root component $\widehat{\mathbf{L}}^{1/2}$ of $\widehat{\mathbf{L}}$. The fourth column indicates the potential speed-up factor that can be achieved with time-parallelization (Eq. (40)).

magnitude reduction in the residual norm. The curves are approximately linear for values of $m_l \leq 3$. Note that modifying m_l affects the structure of $\boldsymbol{\mathcal{A}}_{m_l}$ but does not affect the conditioning of the block components $\widehat{\mathbf{A}}$ of $\boldsymbol{\mathcal{A}}_{m_l}$ as they depend on the diffusion tensor (4) which is a function of the fixed value M (not m_l). This explains why fewer iterations are required to solve the $\boldsymbol{\mathcal{A}}_{m_l}$-matrix systems in the hybrid formulation, for the cases $1 < m_l < M'$, than the systems with the equivalent number of time-blocks associated with Figs 8(a) (cf. total iteration counts for values of $M' = 2$, 3 and 4).

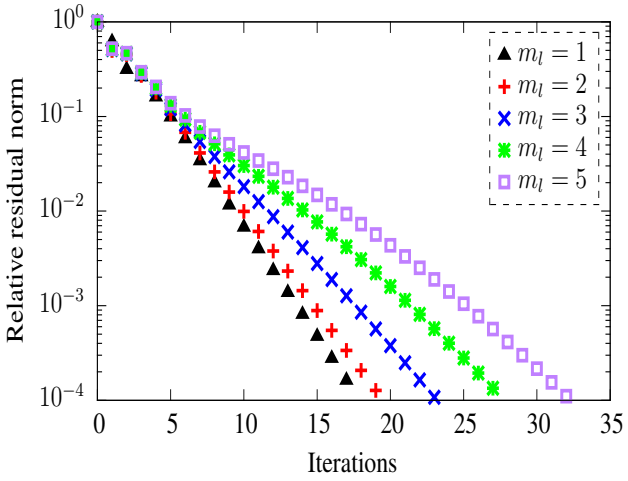


Figure 10: The residual norm relative to the initial residual norm as a function of CI iteration when the NSPD matrix $\boldsymbol{\mathcal{A}}_{m_l}$ is composed of identical $\widehat{\mathbf{A}}$ blocks but different number of time levels m_l. The $\widehat{\mathbf{A}}$ blocks are associated with the diffusion operator with $M = 10$. The first guess and preconditioner transformations are defined by $\boldsymbol{\mathcal{G}} = \boldsymbol{\mathcal{P}}_{\mathrm{I}}$ and $\boldsymbol{\mathcal{P}} = \boldsymbol{\mathcal{P}}_{\mathrm{D}}$. The rhs is the same in all experiments.

4.8 Parallel implementation in NEMOVAR

The NSPD formulation of an implicitly formulated diffusion equation employed to model a correlation operator has been demonstrated to provide a potential speed-up of up to a factor between two and three. Considering that the time spent on the execution of the correlation operator in the current ECMWF operational 3D-Var (NEMOVAR) system with horizontal grid resolution of $1/4^\circ$ amounts to approximately two thirds of the total run-time, this would translate to a potential application speed-up of up to 1.8, assuming a perfect parallel implementation. Such improvement can be considered to be critical for enabling production of a future high-resolution eddy-resolving ocean analysis in an operational environment with tight run-time constraints. Here, we limit ourselves to a discussion of possible implementation strategies of the time-parallel scheme.

The implicitly formulated SPD diffusion operator $\widehat{\mathbf{A}}$ is parallelized in the spatial dimension. The global domain is decomposed into a structured grid of m_{s} sub-domains with each block belonging to one MPI task. The operator $\widehat{\mathbf{A}}$ is applied in parallel on each sub-domain and message passing is employed for exchanging halos during each Chebyshev iteration (see appendix C in WTP16). It should be noted that message passing is also used for both local and global communications in other components of the 3D-Var system, which are also parallelized. The parallel implementation of $\boldsymbol{\mathcal{A}}$ could be considered in combination with the scheme used to parallelize the $\widehat{\mathbf{A}}$ operator in the spatial domain. The total number of tasks p would be split between the spatial m_{s} and temporal m_{t} dimensions such that $m_{\mathrm{s}} m_{\mathrm{t}} = p$. In

practice, there would be both computational and communication overhead associated with the parallel implementation in the pseudo-time and space domains. This becomes apparent when one considers the sub-diagonal block of $\mathcal{A}$ or the implementation of the preconditioner. This strategy for implementing time-parallelization would lead to the problem of determining the optimal pair (m_s, m_t) of space-time sub-domain decomposition that maximizes the computation-to-communication ratio. The pairs $(p,1)$ and $\left(p/M', M'\right)$ correspond, respectively, to the limiting cases of no parallelization and complete parallelization in the pseudo-time domain. This approach would be fairly straightforward to implement, but far from optimal for other components of NEMOVAR, where the processors allocated in the pseudo-time dimension would be idle. This may still be acceptable for specific applications, such as the estimation of normalization factors using a Monte Carlo (randomization) technique (Weaver and Courtier, 2001), which are dominated by the cost of the diffusion operator.

An alternative parallel implementation would require introducing a hybrid parallelization model combining MPI tasks in the spatial dimension with Open Multi-Processing (OpenMP) threads spanning the pseudo-time dimension. In practice, this would amount to adding an outer dimension to the arrays employed by the diffusion operator and corresponding to the pseudo-time. OpenMP directives would need to be inserted only at a high level in loops over this new parallel dimension. A practical advantage of such approach becomes apparent when considering preconditioning or the operations associated with the sub-diagonal block of $\mathcal{A}$. Since all the time-levels are visible to a given MPI task assigned to a spatial sub-domain, no message passing would be required in their implementation. Attention would need to be paid to the MPI communications between OpenMP threads. If the message passing for each time level was to be performed in parallel, a thread safe MPI implementation would need to be available. Alternatively the message passing would have to be performed by the master thread and would become a sequential operation. A hybrid implementation also opens up the possibility to parallelize other components of the data assimilation system, allowing for a potentially more uniform acceleration of the code.

5 Summary and Discussion

In this paper we have shown how a diffusion-based correlation operator can be formulated implicitly in a way that avoids the sequential "time"-stepping of conventional diffusion. The advantage of this approach is that it opens the way to perform in parallel the costly matrix-vector operations required on each step of the diffusion process. The new formulation requires solving a single linear system involving a nonsymmetric positive definite (NSPD) matrix. Following WTP16 who advocated the use of the Chebyshev iteration (CI) to solve the symmetric positive-definite (SPD) matrix systems of the sequential diffusion-based correlation operator, we have proposed here to use the same algorithm to solve the NSPD system. Linearity and the absence of global communications are the two properties of CI that make it an attractive algorithm for the problem under consideration. In this paper, however, the problem is complicated by the nonsymmetric and nondiagonalizable properties of the system matrix that must be inverted. Understanding the convergence properties of CI for this problem is therefore vital to determine whether a "time"-parallelized NSPD-based system can be a viable alternative to the sequential SPD-based system. This has been the main purpose of the paper.

The new formulation of the correlation operator has been implemented in the NEMOVAR global ocean variational data assimilation system. Based on numerical experiments with that system and supported by theoretical results, we can summarize the main findings of this study as follows.

- *Convergence*. Even though the NSPD matrix is nondiagonalizable, CI is guaranteed to converge (Manteuffel, 1975, 1977). For the SPD system, the convergence rate depends on the properties of

the shifted and scaled Chebyshev polynomials. For the NSPD system, the convergence rate also depends on the properties of the derivatives of the shifted and scaled Chebyshev polynomials. As a result, CI applied to the NSPD system converges more slowly than CI applied to the SPD system.

- *Practical convergence*. A high-precision solution is not required to obtain an adequate representation of correlation functions for data assimilation. Reducing the 2-norm of the residual or solution error by three- to four-orders of magnitude is generally adequate. Therefore, in the numerical experiments, we focused on evaluating the convergence properties of CI for achieving an accuracy in this range.

- *Eigenvalues*. The NSPD matrix under consideration has a special structure such that its eigenvalues are real and given by those of the SPD matrix with algebraic multiplicity of M where M is the number of implicit iterations. The extreme eigenvalues required by CI can therefore be estimated using an eigenvalue algorithm (e.g., Lanczos) applied to the simpler SPD matrix.

- *Initial residual*. When CI is applied to the NSPD system, it has a nonlinear convergence behaviour, especially in the early iterations, and strong sensitivity to the initial residual, contrary to what occurs when CI is applied to the SPD system. The nonlinear behaviour increases as M increases. For the SPD system, the number of iterations required to reduce the residual 2-norm (relative to the initial residual 2-norm) can be accurately estimated using an analytical relationship from Axelsson (1996). For the NSPD system, no such simple relationship exists to the best of our knowledge.

- *First guess*. For the SPD problem, defining the first guess to be the right-hand side instead of a zero field results in nearly a one-order of magnitude improvement in the accuracy of the solution when seeking a four-order of magnitude reduction of the residual 2-norm relative to its initial value. A similar relative improvement in solution accuracy can be obtained for the NSPD problem using a first guess whose value at each "time" level is equal to the right-hand side of the SPD problem. For the NSPD problem, this simple choice of first guess also results in a reduction in the number of iterations for the modest convergence tolerance considered.

- *Preconditioning*. The structure of the exact inverse of the NSPD matrix suggests that a preconditioner should have a block lower triangular form where the block matrices involve powers of the approximate inverse of the SPD matrix. Defining the block matrices as powers of the inverse of the diagonal of the SPD matrix results in an inexpensive preconditioner that is particularly effective for accelerating convergence when $M > 10$. Results with the preconditioner for values of $M \leq 10$ can also accelerate convergence, although the gain is less clear and may not offset the extra cost of applying the preconditioner. For the SPD problem, the diagonal preconditioner has no impact.

- *Stopping criterion*. To preserve symmetry, the correlation model should be formulated in "square-root" factors where each factor represents $M/2$ steps of the diffusion operator and involves the solution of an NSPD system. CI should be employed with the same fixed number of iterations, K, when solving each NSPD system. To determine an appropriate value of K, CI is first applied to a trial problem with a particular stopping criterion. Here, we proposed to define K as the arithmetic mean of the number of iterations required to solve each NSPD system in succession; i.e., to apply the full M-step diffusion operator. While the 2-norm of the residual mostly gave an adequate measure of the solution error, it was argued that a more reliable stopping criterion should be based on the 2-norm of the actual solution error, especially in the presence of a general preconditioner. With a non-zero first guess, measuring the reduction of the 2-norm of the solution error with respect to either the 2-norm of the initial solution error or the 2-norm of the solution itself was shown to lead to a similar value of K, when computed as an average of the values required for solving each system.

- *Parallel speed-up*. In terms of the total number of applications of the SPD matrix, solving the single NSPD system with CI is more costly than solving the sequence of SPD systems with CI. However, solving the NSPD system requires fewer iterations than solving the sequence of SPD systems, thus providing scope for reducing run-time by performing the matrix-vector computations in parallel. In particular, in the case of a perfectly "time"-parallelized NSPD system, we have shown that there is potential to reduce the diffusion operator run-time by a factor between two and three. Different possible ways of implementing a "time"-parallel diffusion operator have been discussed within the context of NEMOVAR. Most promising is the use of a hybrid parallelization approach that combines MPI tasks in the spatial domain with OpenMP threads spanning the pseudo-time dimension. This will be investigated with NEMOVAR in future work.

Acknowledgements

The authors gratefully acknowledge the support of the ERA-CLIM2 project of the European Framework Programme 7 (Grant Agreement No. 607029), the AVENUE project of the RTRA STAE Foundation, and the French national programme LEFE/INSU. This Technical Memorandum is also registered as a CERFACS Technical Report, reference TR-PA-17-119.

Appendix A: Correction to the CI algorithm presented in WTP16

Orthogonal polynomials like the shifted and scaled Chebsyhev polynomials satisfy a general three-term recurrence relation (Eq. (A1) in WTP16):

$$\begin{aligned} \varphi_{-1}(t) &= 0, \\ \varphi_0(t) &= 1, \\ \varphi_{k+1}(t) &= \left(\frac{t-\mu_{k+1}}{\nu_{k+1}}\right)\varphi_k(t) - \left(\frac{\tau_{k+1}}{\nu_{k+1}}\right)\varphi_{k-1}(t), \end{aligned} \tag{41}$$

where $k \geq 0$. The coefficients ν_{k+1}, μ_{k+1} and τ_{k+1} are related to the step sizes α_k and β_{k+1} in the CI algorithm according to (Eqs (A4), (A13) and (A14) in WTP16)

$$\alpha_k = -\frac{1}{\nu_{k+1}}, \tag{42}$$

$$\beta_{k+1} = \frac{\tau_{k+2}}{\nu_{k+1}}, \tag{43}$$

$$\tau_{k+1} = -\left(\mu_{k+1} + \nu_{k+1}\right). \tag{44}$$

Here, we are interested in the specific case $k = 0$, which was overlooked in WTP16. For $k = 0$, the third equation in (41) becomes

$$\varphi_1(t) = \frac{t-\mu_1}{\nu_1}. \tag{45}$$

The shifted and scaled Chebyshev polynomials are given by

$$\varphi_k(t) = \frac{T_k(\chi_t)}{T_k(\chi_0)} \tag{46}$$

where

$$\chi_t \;=\; \chi(t) \;:=\; \frac{t-\sigma}{\delta}, \tag{47}$$

and σ and δ are the CI parameters defined in lines 1 and 2 of Algorithm 2. Since $T_1(t) = t$, equating Eqs (45) and (46) for $k = 1$ gives

$$\frac{t-\mu_1}{\nu_1} = \frac{T_1(\chi_t)}{T_1(\chi_0)} = \frac{(t-\sigma)/\delta}{-\sigma/\delta} = \frac{t-\sigma}{-\sigma},$$

implying that $\mu_1 = \sigma$ and $\nu_1 = -\sigma$, and hence from Eqs (42) and (44) that

$$\alpha_0 \;=\; \frac{1}{\sigma}, \tag{48}$$

$$\beta_1 \;=\; -\frac{\tau_2}{\sigma}. \tag{49}$$

To determine τ_2, we use the third expression in (A29) from WTP16 with $k = 2$:

$$\tau_2 \;=\; \frac{\delta}{2}\left(\frac{T_0(\chi_0)}{T_1(\chi_0)}\right) = -\frac{\delta^2}{2\sigma} \tag{50}$$

since $T_0(t) = 1$. Substituting Eq. (50) in Eq. (49) and using Eq. (48) gives

$$\beta_1 = \frac{(\delta\,\alpha_0)^2}{2}. \tag{51}$$

The β_{k+1} coefficients for $k > 0$ are the same as given in WTP16:

$$\beta_{k+1} = \left(\frac{\delta\,\alpha_k}{2}\right)^2. \tag{52}$$

Comparing Eqs (51) and (52) shows that the correct β_1 coefficient is a factor of 2 larger than the coefficient used in WTP16. The corrected algorithm is consistent with Algorithm 5 in Gutknecht and Röllin (2002).

The impact of the correction is minor in terms of total iteration counts, but does lead to a more consistent convergence behaviour of the solver between steps in the $m = 1,\ldots,M'$ sequence and better agreement with the theoretical bound k_s (Eq. 20).

Appendix B: Convergence properties of the Chebyshev iteration

The $N\times N$ SPD matrix $\widehat{\mathbf{A}}$ can be written in terms of its $n = 1,\ldots,N$ eigenpairs $(\lambda_n, \mathbf{v}_n)$ as

$$\widehat{\mathbf{A}} = \sum_{n=1}^{N} \lambda_n \mathbf{v}_n \mathbf{v}_n^{\mathrm{T}}.$$

The eigenvalues are assumed to be ordered from smallest λ_1 to largest λ_N. The $M'N/2\times M'N/2$ NSPD matrix $\boldsymbol{\mathcal{A}}$ (Eq. (9)) can be expressed in Jordan block form as (Saad, 2003, pp. 15–16)

$$\boldsymbol{\mathcal{A}} \;=\; \mathbf{Z}\mathbf{J}\mathbf{Z}^{-1} \tag{53}$$

where $\mathbf{J} = \mathrm{diag}(\mathbf{J}_1, \mathbf{J}_2, \ldots, \mathbf{J}_N)$, the $M' \times M'$ matrix $\mathbf{J}_n$ being the Jordan block associated with eigenvalue λ_n, $n = 1, \ldots, N$:

$$\mathbf{J}_n \equiv \mathbf{J}_n(\lambda_n) = \begin{pmatrix} \lambda_n & 1 & & \\ & \lambda_n & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_n \end{pmatrix}. \tag{54}$$

Note that although $\boldsymbol{\mathcal{A}}$ is nonsymmetric, and hence possesses potentially complex eigenvalues, its specific structure gives rise to a spectrum of real eigenvalues, given by those of the SPD matrix $\widehat{\mathbf{A}}$ with algebraic multiplicity of M'.

From Eq. (53), a general polynomial of $\boldsymbol{\mathcal{A}}$ (here the Chebyshev polynomial $\varphi_k(\boldsymbol{\mathcal{A}})$), can be expressed as (Higham, 2008, p. 3, Definition 1.2)

$$\varphi_k(\boldsymbol{\mathcal{A}}) = \mathbf{Z}\,\mathrm{diag}\big(\varphi_k(\mathbf{J}_n)\big)\,\mathbf{Z}^{-1} \tag{55}$$

where

$$\varphi_k(\mathbf{J}_n) = \begin{pmatrix} \varphi_k(\lambda_n) & \varphi_k'(\lambda_n) & \cdots & \frac{1}{(M'-1)!}\varphi_k^{(M'-1)}(\lambda_n) \\ & \varphi_k(\lambda_n) & \ddots & \vdots \\ & & \ddots & \varphi_k'(\lambda_n) \\ & & & \varphi_k(\lambda_n) \end{pmatrix},$$

$\varphi_k^{(m)}(\lambda_n)$ denoting the mth derivative of φ_k evaluated at λ_n. Since these derivatives exist, $\varphi_k(\boldsymbol{\mathcal{A}})$ is said to be *defined on the spectrum of* $\boldsymbol{\mathcal{A}}$ (Higham, 2008, p. 3, Definition 1.1).

From the analysis above, it is apparent that the convergence of CI for a nondiagonalizable matrix requires convergence of the derivatives of the polynomial as well the polynomial itself, as stated by the following (Manteuffel, 1977, Theorem 2.2):

Theorem 1 *If λ_n is an eigenvalue of $\boldsymbol{\mathcal{A}}$ with invariant subspace of dimension d_n, then $\|\varphi_k(\boldsymbol{\mathcal{A}})\| \to 0$ as $k \to \infty$ if and only if $\varphi_k^{(j)}(\lambda_n) \to 0$ as $k \to 0$ for every $j < d_n$, for each eigenvalue λ_n.*

In our case, $d_n = M'$. Manteuffel (1975) shows that Theorem 1 is indeed satisfied by the shifted and scaled Chebyshev polynomials.

By exploiting the special structure of $\boldsymbol{\mathcal{A}}$ in our problem, the matrix $\mathbf{Z}$ can be shown to be a unitary matrix ($\mathbf{Z}^{-1} = \mathbf{Z}^{\mathrm{T}}$) of the form

$$\mathbf{Z} = \begin{pmatrix} \mathbf{Z}_1 & \mathbf{Z}_2 & \cdots & \mathbf{Z}_N \end{pmatrix}$$

where the block $NM' \times M'$ matrix $\mathbf{Z}_n$ is associated with the eigenvector $\mathbf{v}_n$ of $\widehat{\mathbf{A}}$:

$$\mathbf{Z}_n = \begin{pmatrix} & & (-1)^{(M'-1)}\mathbf{v}_n \\ & \iddots & \\ -\mathbf{v}_n & & \\ \mathbf{v}_n & & \end{pmatrix}.$$

Using Eq. (55) and properties of the 2-norm (Meyer, 2000, p. 283), we can then write

$$\|\varphi_k(\mathcal{A})\| = \|\mathbf{Z}\,\varphi_k(\mathbf{J})\,\mathbf{Z}^{\mathrm{T}}\| = \|\varphi_k(\mathbf{J})\| \tag{56}$$

where

$$\|\varphi_k(\mathbf{J})\| = \max_n \|\varphi_k(\mathbf{J}_n)\|,\ n = 1,\ldots,N.$$

The next theorem states some convergence properties for $\|\varphi_k(\mathbf{J})\|$.

Theorem 2 *Let the Jordan block $\mathbf{J}_n$ associated with eigenvalue λ_n, $n = 1,\ldots,N$, be defined by Eq. (54). Then,*

$$\lim_{k\to\infty} \frac{\|\varphi_k(\mathbf{J}_n)\|}{\left|\frac{\varphi_k^{(M'-1)}(\lambda_n)}{(M'-1)!}\right|} = 1. \tag{57}$$

Proof. $\varphi_k(\mathbf{J}_n)$ can be expressed as a Taylor series (Higham, 2008, p. 4):

$$\varphi_k(\mathbf{J}_n) = \varphi_k(\lambda_n)\mathbf{I}_n + \varphi_k'(\lambda_n)\mathbf{N}_n + \cdots + \frac{\varphi_k^{(M'-1)}(\lambda_n)}{(M'-1)!}\mathbf{N}_n^{M'-1} \tag{58}$$

where $\mathbf{J}_n = \lambda_n\mathbf{I}_n + \mathbf{N}_n$. From Eq. (54), the matrix $\mathbf{N}_n$ has zero elements except for a superdiagonal of 1s, and satisfies $\mathbf{N}_n^{M'} = \mathbf{0}$. Dividing Eq. (58) by the coefficient of the last term in the series gives

$$\left(\frac{(M'-1)!}{\varphi_k^{(M'-1)}(\lambda_n)}\right)\varphi_k(\mathbf{J}_n) = \frac{(M'-1)!\,\varphi_k(\lambda_n)}{\varphi_k^{(M'-1)}(\lambda_n)}\mathbf{I}_n + \frac{(M'-1)!\,\varphi_k'(\lambda_n)}{\varphi_k^{(M'-1)}(\lambda_n)}\mathbf{N}_n + \cdots$$
$$+ \frac{(M'-1)!\,\varphi_k^{(j)}(\lambda_n)}{j!\,\varphi_k^{(M'-1)}(\lambda_n)}\mathbf{N}_n^j + \cdots + \mathbf{N}_n^{M'-1}.$$

Taking the norm of the above expression and noting that $\|\mathbf{N}_n^j\| = 1$ for $j < M'$, and $|\varphi_k^{(j)}(\lambda_n)/\varphi_k^{(M'-1)}(\lambda_n)| \to 0$ as $k \to \infty$ for $j < M'-1$, we obtain the result (57). □

Appendix C: Block-diagonal preconditioning of the NSPD system

Consider the sequence of SPD systems in (7) where each member in the sequence is preconditioned to the left by $\mathbf{P} \approx \widehat{\mathbf{A}}^{-1}$. An augmented NSPD system can be formed from the resulting preconditioned sequence as

$$\widetilde{\mathcal{P}}\mathcal{A}\boldsymbol{\Psi} = \widetilde{\mathcal{P}}\boldsymbol{\zeta}, \tag{59}$$

where $\widetilde{\mathcal{P}}$ is defined by Eq. (24). As with $\mathcal{P}\mathcal{A}$, the eigenvalues of $\widetilde{\mathcal{P}}\mathcal{A}$ are multiples of the eigenvalues of $\mathbf{PA}$. Compared to the block lower triangular preconditioner $\mathcal{P}$, the block-diagonal preconditioner $\widetilde{\mathcal{P}}$ has the advantage that its M blocks can be applied in parallel.

An obvious drawback with Eq. (24) is that employing it with the ideal preconditioner for the SPD matrix ($\mathbf{P} = \widehat{\mathbf{A}}^{-1}$) does not result in an ideal preconditioner for Eq. (59). To see this, it is instructive to examine the update equation on the first iteration of Algorithm 3 with $\widetilde{\mathcal{P}}$ as a preconditioner. Given a first guess $\boldsymbol{\Psi}_0$, the update equation for $\boldsymbol{\Psi}$ is

$$\boldsymbol{\Psi}_1 = \boldsymbol{\Psi}_0 - \alpha_0\left(\widetilde{\mathcal{P}}\mathcal{A}\boldsymbol{\Psi}_0 - \widetilde{\mathcal{P}}\boldsymbol{\zeta}\right)$$

where the term in parentheses corresponds to the initial search direction ($\mathbf{p}_0$). The update equations for the individual elements of $\boldsymbol{\Psi}_1$ are

$$\left.\begin{array}{rcl}
\boldsymbol{\psi}_1^1 &=& \boldsymbol{\psi}_1^0 - \alpha_0\big(\qquad\qquad \mathbf{P}\widehat{\mathbf{A}}\boldsymbol{\psi}_1^0 - \mathbf{P}\boldsymbol{\psi}_0 \big) \\
\boldsymbol{\psi}_2^1 &=& \boldsymbol{\psi}_2^0 - \alpha_0\big(-\mathbf{P}\boldsymbol{\psi}_1^0 + \mathbf{P}\widehat{\mathbf{A}}\boldsymbol{\psi}_2^0 \qquad\big) \\
&\vdots& \\
\boldsymbol{\psi}_{M'}^1 &=& \boldsymbol{\psi}_{M'}^0 - \alpha_0\big(-\mathbf{P}\boldsymbol{\psi}_{M'-1}^0 + \mathbf{P}\widehat{\mathbf{A}}\boldsymbol{\psi}_{M'}^0 \qquad\big)
\end{array}\right\}.$$

It is easy to see that with $\mathbf{P} = \widehat{\mathbf{A}}^{-1}$, and hence $\alpha_0 = 1$, only the first component $\boldsymbol{\psi}_1^1$ converges to the exact solution ($\boldsymbol{\psi}_1^1 = \widehat{\mathbf{A}}^{-1}\,\boldsymbol{\psi}_0$). The exact solutions for the other states $\boldsymbol{\psi}_m$, $m = 2,\ldots,M'$, will be retrieved successively after M' iterations. This number is related to the special structure of the preconditioned system matrix, as explained below.

The matrix $\widetilde{\mathcal{P}}\mathcal{A}$ has block-matrix form similar to $\mathcal{A}$:

$$\widetilde{\mathcal{P}}\mathcal{A} = \begin{pmatrix}
\mathbf{P}\widehat{\mathbf{A}} & & & \\
-\mathbf{P} & \mathbf{P}\widehat{\mathbf{A}} & & \\
& \ddots & \ddots & \\
& & -\mathbf{P} & \mathbf{P}\widehat{\mathbf{A}}
\end{pmatrix}.$$

When $\mathbf{P} = \widehat{\mathbf{A}}^{-1}$, the eigenvalues of the matrix $\widetilde{\mathcal{P}}_{\widehat{\mathbf{A}}}\mathcal{A}$ are all equal to 1 with algebraic multiplicity of $N \times M'$. Furthermore, $\widetilde{\mathcal{P}}_{\widehat{\mathbf{A}}}\mathcal{A} - \mathbf{I}_{NM'}$ is a nilpotent matrix; i.e.,

$$\rho(\widetilde{\mathcal{P}}_{\widehat{\mathbf{A}}}\mathcal{A}) \equiv \big(\widetilde{\mathcal{P}}_{\widehat{\mathbf{A}}}\mathcal{A} - \mathbf{I}_{NM'}\big)^{M'} = \mathbf{0},$$

implying that M' is the degree of the *minimal polynomial* of $\widetilde{\mathcal{P}}_{\widehat{\mathbf{A}}}\mathcal{A}$. Algorithm 3 will then stop at iteration M' since

$$\|\boldsymbol{\Xi}_{M'}\| \leq \|\varphi_{M'}(\widetilde{\mathcal{P}}_{\widehat{\mathbf{A}}}\mathcal{A})\|\,\|\boldsymbol{\Xi}_0\| \leq \|\rho(\widetilde{\mathcal{P}}_{\widehat{\mathbf{A}}}\mathcal{A})\|\,\|\boldsymbol{\Xi}_0\| = 0.$$

References

Axelsson O. 1996. *Iterative Solution Methods*. Cambridge University Press: Cambridge, UK.

Balmaseda MA, Mogensen K, Weaver AT. 2013. Evaluation of the ECMWF Ocean Reanalysis ORAS4. *Q. J. R. Meteorol. Soc.* **139**: 1132–1161.

Belo Pereira M, Berre L. 2006. The use of an ensemble approach to study the background error covariances in a global NWP model. *Mon. Weather Rev.* **134**: 2466–2489.

Berre L. 2015. Background error covariances: estimation and specification. In: *Advanced Data Assimilation for Geosciences*, Blayo E, Boquet M, Cosme E, Cugliandolo LF (eds), Oxford University Press: Oxford, UK, pp. 185–208.

Buehner M, Shlyaeva A. 2015. Scale-dependent background-error covariance localisation. *Tellus A.* **67**: 28 027.

Bui-Thanh T, Ghattas O, Martin J, Stadler G. 2013. A computational framework for infinite-dimensional Bayesian inverse problems. Part I: The linearized case, with application to global seismic inversion. *SIAM J. Sci. Comput.* **35**: A2494–A2523.

Cai XC, Sarkis A. 1999. Restricted additive Schwarz preconditioner for general sparse linear systems. *SIAM J. Sci. Comput.* **21**: 792–797.

Carrier MJ, Ngodock H. 2010. Background-error correlation model based on the implicit solution of a diffusion equation. *Ocean Model.* **35**: 45–53.

Daley R. 1991. *Atmospheric Data Analysis*. Cambridge Atmospheric and Space Sciences Series, Cambridge University Press: Cambridge, UK.

Desroziers G. 2015. Observation error specifications. In: *Advanced Data Assimilation for Geosciences*, Blayo E, Boquet M, Cosme E, Cugliandolo LF (eds), Oxford University Press: Oxford, UK, pp. 209–228.

Dobricic S, Pinardi N. 2008. An oceanographic three-dimensional variational data assimilation scheme. *Ocean Model.* **22**: 89–105.

Fisher M, Gürol S. 2017. Parallelization in the time dimension of four-dimensional variational data assimilation. *Q. J. R. Meteorol. Soc.* **143**: 1136–1147.

Gander MJ. 2015. 50 years of time parallel time integration. In: *Multiple Shooting and Time Domain Decomposition Methods*, Carraro T, Geiger M, Körkel S, Rannacher R (eds), Springer: New York, NY, pp. 69–113.

Gergelits T, Strakoš Z. 2014. Composite convergence bounds based on Chebyshev polynomials and finite precision conjugate gradient computations. *Numer. Algorithms.* **65**: 759–782.

Gmeiner B, Drzisga D, Rüde U, Scheichl R, Wohlmuth B. 2017. Scheduling massively parallel multigrid for multilevel Monte Carlo methods. *SIAM J. Sci. Comput.* (In press).

Golub GH, Van Loan CF. 1996. *Matrix Computations*. Johns Hopkins University Press: Baltimore, MD, 3rd edn.

Gutknecht M, Röllin S. 2002. The Chebyshev iteration revisited. *Parallel Comput.* **28**: 263–283.

Guttorp P, Gneiting T. 2006. Miscellanea studies in the history of probability and statistics XLIX: on the Matérn correlation family. *Biometrika.* **93**: 989–995.

Higham N. 2008. *Functions of Matrices*. Society for Industrial and Applied Mathematics: Philadelphia, PA.

Lindgren F, Rue H, Lindström J. 2011. An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach. *J. Roy. Stat. Soc.: Series B Stat. Method.* **73**: 423–398.

Lorenc A. 2015. Four-dimensional variational data assimilation. In: *Advanced Data Assimilation for Geosciences*, Blayo E, Boquet M, Cosme E, Cugliandolo LF (eds), Oxford University Press: Oxford, UK, pp. 31–73.

Madec G. 2008. *NEMO Ocean Engine*. Note du Pôle de modélisation, Institut Pierre-Simon Laplace IPSL), France, No 27, ISSN No 1288-1619: France.

Manteuffel TA. 1975. An iterative method for solving nonsymmetric linear systems with dynamic estimation of parameters. Technical Report UIUCDS-R-75-758, Department of Computer Science, Univ. of Illinois, Urbana, IL.

Manteuffel TA. 1977. The Tchebychev iteration for nonsymmetric linear systems. *Numer. Math.* **28**: 307–327.

Ménétrier B, Auligné T. 2015. Optimized localization and hybridization to filter ensemble-based covariances. *Mon. Weather Rev.* **143**: 3931–3947.

Meyer CD. 2000. *Matrix Analysis and Applied Linear Algebra*. Society for Industrial and Applied Mathematics: Philadelphia, PA.

Michel Y, Ménétrier B, Montmerle T. 2016. Objective filtering of the local correlation tensor. *Q. J. R. Meteorol. Soc.* **142**: 2314–2323.

Mirouze I, Blockley EW, Lea DJ, Martin MJ, Bell MJ. 2016. A multiple length scale correlation operator for ocean data assimilation. *Tellus A.* **68**: 29 744.

Mirouze I, Weaver AT. 2010. Representation of correlation functions in variational assimilation using an implicit diffusion operator. *Q. J. R. Meteorol. Soc.* **136**: 1421–1443.

Mogensen K, Balmaseda MA, Weaver AT. 2012. The NEMOVAR ocean assimilation system and its implementation for ECMWF's ocean analysis for System 4. Technical Memorandum 668, ECMWF, Reading, UK. Also registered as a CERFACS Technical Report No. TR-CMGC-12-30.

Purser RJ, Wu WS, Parrish DF, Roberts NM. 2003a. Numerical aspects of the application of recursive filters to variational statistical analysis. Part I: Spatially homogeneous and isotropic Gaussian covariances. *Mon. Weather Rev.* **131**: 1524–1535.

Purser RJ, Wu WS, Parrish DF, Roberts NM. 2003b. Numerical aspects of the application of recursive filters to variational statistical analysis. Part II: spatially inhomogeneous and anisotropic general covariances. *Mon. Weather Rev.* **131**: 1536–1548.

Rabier F, Fisher M. 2015. Data assimilation in meteorology. In: *Advanced Data Assimilation for Geosciences*, Blayo E, Boquet M, Cosme E, Cugliandolo LF (eds), Oxford University Press: Oxford, UK, pp. 433–459.

Rao V, Sandu A. 2016. A time-parallel approach to strong-constraint four-dimensional variational data assimilation. *J. Comput. Phys.* **331**: 583–593.

Saad Y. 2003. *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics: Philadelphia, PA, 2nd edn.

Waters J, Lea DJ, Martin MJ, Mirouze I, Weaver AT, While J. 2015. Implementing a variational data assimilation system in an operational 1/4 degree global ocean model. *Q. J. R. Meteorol. Soc.* **141**: 333–349.

Weaver AT, Courtier P. 2001. Correlation modelling on the sphere using a generalized diffusion equation. *Q. J. R. Meteorol. Soc.* **127**: 1815–1846.

Weaver AT, Mirouze I. 2013. On the diffusion equation and its application to isotropic and anisotropic correlation modelling in variational assimilation. *Q. J. R. Meteorol. Soc.* **139**: 242–260.

Weaver AT, Tshimanga J, Piacentini A. 2016. Correlation operators based on an implicitly formulated diffusion equation solved with the Chebyshev iteration. *Q. J. R. Meteorol. Soc.* **142**: 455–471.

Yaremchuk M, Nechaev D. 2013. Covariance localization with the diffusion-based correlations models. *Mon. Weather Rev.* **141**: 848–860.

Zhu J. 1994. *Solving Partial Differential Equations on Parallel Computers*. World Scientific: Singapore.