Journal of Computational Science xxx (2017) xxx–xxx

Contents lists available at ScienceDirect

Journal of Computational Science

journal homepage: www.elsevier.com/locate/jocs

A parallel implementation of the ensemble Kalman filter based on modified Cholesky decomposition

Elias D. Nino-Ruiz [a,b,*], Adrian Sandu [b], Xinwei Deng [c]

[a] Department of Computer Science, Universidad del Norte, Barranquilla, Colombia
[b] Computational Science Laboratory, Department of Computer Science, Virginia Tech, Blacksburg, VA 24060, USA
[c] Department of Statistics, Virginia Tech, Blacksburg, VA 24060, USA

ARTICLE INFO

Article history:
Received 30 May 2016
Received in revised form 8 December 2016
Accepted 7 April 2017
Available online xxx

JEL classification:
62L20
62M05
62M20
62P35

Keywords:
Ensemble Kalman filter
Covariance matrix estimation
Local domain analysis

ABSTRACT

This paper discusses an efficient parallel implementation of the ensemble Kalman filter based on the modified Cholesky decomposition. The proposed implementation starts with decomposing the domain into sub-domains. In each sub-domain a sparse estimation of the inverse background error covariance matrix is computed via a modified Cholesky decomposition; the estimates are computed concurrently on separate processors. The sparsity of this estimator is dictated by the conditional independence of model components for some radius of influence. Then, the assimilation step is carried out in parallel without the need of inter-processor communication. Once the local analysis states are computed, the analysis sub-domains are mapped back onto the global domain to obtain the analysis ensemble. Computational experiments are performed using the Atmospheric General Circulation Model (SPEEDY) with the T-63 resolution on the Blueridge cluster at Virginia Tech. The number of processors used in the experiments ranges from 96 to 2048. The proposed implementation outperforms in terms of accuracy the well-known local ensemble transform Kalman filter (LETKF) for all the model variables. The computational time of the proposed implementation is similar to that of the parallel LETKF method (where no covariance estimation is performed). Finally, for the largest number of processors, the proposed parallel implementation is 400 times faster than the serial version of the proposed method.

1. Introduction

In operational data assimilation, sequential and variational methods are required to posses the ability of being performed in parallel [1–3]. This obeys to current atmospheric and oceanic model resolutions in which the total number of components arises to the order of millions and the daily information to be assimilated in the order of terabytes [4,5]. Thus, serial data assimilation methods are impractical under realistic operational scenarios. In sequential data assimilation, one of the best parallel ensemble Kalman filter (EnKF) implementations is the local ensemble transform Kalman filter (LETKF) [6]. This method is based on domain localization given a radius of influence ζ. Usually, the assimilation process is performed for each model component in parallel making use of a deterministic formulation of the EnKF in the ensemble space. In this formulation, the unknown background error covariance matrix is estimated by the rank-deficient ensemble covariance matrix which, in ensemble space, is well-defined. The LETKF relies in the assumption that local domain analyses avoid the impact of spurious correlations, for instance, by considering only small values for ζ. However, in operational data assimilation, ζ can be large owing to circumstances such as sparse observational networks and/or long distance data error correlations (i.e., pressure fields) In such cases, the accuracy of the LETKF can be negatively impacted owing to spurious correlations.

We think there is an opportunity to provide a more robust parallel ensemble Kalman filter implementation via a better estimation of background error correlations. When two model components (i.e., grid points) are assumed to be conditionally independent, their corresponding entry in the estimated inverse background error covariance matrix is zero. Conditionally dependence/independence of model components can be forced making use of local domain analyses. For instance, when the distance of two model components in physical space is larger than ζ, their corresponding entry in the inverse background error covariance matrix is zero. This can be exploited in order to obtain sparse estimators of such matrix which implies huge savings in terms of memory and computations. Even more, high performance computing can be used in order to speedup the assimilation process: the global domain can be decomposed according to an available number

* Corresponding author at: Department of Computer Science, Universidad del Norte, Barranquilla, Colombia.
E-mail addresses: enino@uninorte.edu.co, enino@vt.edu (E.D. Nino-Ruiz), asandu7@vt.edu (A. Sandu), xdeng@vt.edu (X. Deng).

http://dx.doi.org/10.1016/j.jocs.2017.04.005
1877-7503/

of processors, for all processors, local inverse background error covariance matrices are estimated and then, the stochastic EnKF formulation [7] can be used in order to compute local domain analyses. The local analyses and then mapped back onto the global domain from which the global analysis state is obtained.

This paper is organized as follows. In Section 2 basic concepts regarding sequential data assimilation and covariance matrix estimation are presented, in Section 3 a parallel implementation of the ensemble Kalman filter based on the modified Cholesky decomposition is proposed; experimental results are discussed in Section 4 and future research directions are presented in Section 5. Conclusions are drawn in Section 6.

2. Preliminaries

2.1. Modified Cholesky decomposition

Let $\mathbf{S} = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_N\} \in \mathbb{R}^{n\times N}$, the matrix whose columns are nth dimensional random Gaussian vectors with probability distribution $\mathcal{N}(\mathbf{0}_n, \mathbf{Q})$, where the number of columns N denotes the number of samples. Denote by $\mathbf{x}^{[j]} \in \mathbb{R}^{N\times 1}$, the vector holding the jth component across all the columns of $\mathbf{S}$, for $2 \leq j \leq n$. The modified Cholesky decomposition [8] arises from regressing each variable $\mathbf{x}^{[j]}$ on its predecessors $\mathbf{x}^{[j-1]}, \mathbf{x}^{[j-2]}, \ldots, \mathbf{x}^{[1]}$, that is, fitting regressions:

$$\mathbf{x}^{[j]} = \sum_{q=1}^{j-1} \beta_{jq} \cdot \mathbf{x}^{[q]} + \varepsilon^{[j]} \in \mathbb{R}^{N\times 1}, \tag{1}$$

where $\varepsilon^{[j]}$ denotes the error in the regression of the jth component. Let $\mathbf{D}_{jj} = \left\{\mathbf{var}\left(\varepsilon^{[j]}\right)\right\} \in \mathbb{R}^{n\times n}$ be the diagonal matrix of error variances and let $\mathbf{T}_{jq} = \{-\beta_{jq}\} \in \mathbb{R}^{n\times n}$ denote the unitary lower-triangular matrix containing the negative value of regression coefficients, for $2 \leq q < j \leq n$. An approximation of the inverse covariance matrix $\mathbf{Q}^{-1} \in \mathbb{R}^{n\times n}$ reads:

$$\mathbf{Q}^{-1} \approx \hat{\mathbf{Q}}^{-1} = \mathbf{T}^T \cdot \mathbf{D}^{-1} \cdot \mathbf{T}, \tag{2}$$

and making use of basic linear algebra, an approximation of $\mathbf{Q} \in \mathbb{R}^{n\times n}$ is:

$$\mathbf{Q} \approx \hat{\mathbf{Q}} = \mathbf{T}^{-1} \cdot \mathbf{D} \cdot \mathbf{T}^{-T}. \tag{3}$$

2.2. Local ensemble transform Kalman filter

Localization is commonly used in the context of sequential data assimilation in order to mitigate the impact of spurious correlations in the assimilation process. In general, two forms of localization methods are used: covariance matrix localization and domain localization, both have proven to be equivalent [9]. In practice, covariance matrix localization can be very difficult owing to the explicit representation in memory of the ensemble covariance matrix. On the other hand, domain localization methods avoid spurious correlations by considering only observations within a given radius of influence ζ: in the two-dimensional case, each model component is surrounded by a local box of dimension $(2\cdot\zeta+1, 2\cdot\zeta+1)$ and the information within the scope of ζ (observed components and background error correlations) is used in the assimilation process and conversely, the information out the local box is discarded. In Fig. 1, local boxes for different radii of influence ζ are shown. The red grid point is the one to be assimilated, blue points are used in the assimilation process while black points are discarded. Based on this idea, the local ensemble transform Kalman filter is proposed (LETKF) [10].

The global formulation of the LETKF is defined as follows: for a given background ensemble

$$\mathbf{X}^b = \left[\mathbf{x}^{b[1]}, \mathbf{x}^{b[2]}, \ldots, \mathbf{x}^{b[N]}\right] \in \mathbb{R}^{n\times N}, \tag{4}$$

and ensemble perturbation matrix

$$\mathbf{U}^b = \mathbf{X}^b - \bar{\mathbf{x}}^b \otimes \mathbf{1}_N^T \in \mathbb{R}^{n\times N}, \tag{5}$$

where n is the number of model components, N is the ensemble size, $\mathbf{x}^{b[i]} \in \mathbb{R}^{n\times 1}$ is the ith ensemble member, for $1 \leq i \leq N$, $\bar{\mathbf{x}}^b$ is the ensemble mean, $\mathbf{1}_N$ is the Nth dimensional vector whose components are all ones and $\otimes$ denotes the outer product of two vectors, an estimated of the analysis error covariance matrix in the ensemble space reads:

$$\hat{\mathbf{P}}^a = \left[(N-1)\cdot\mathbf{I}_{N\times N} + \mathbf{Z}^T \cdot \mathbf{R}^{-1} \cdot \mathbf{Z}\right]^{-1} \tag{6a}$$

where $\mathbf{Z} = \mathbf{H}\cdot\mathbf{U}^b \in \mathbb{R}^{m\times N}$, $\mathbf{H} \in \mathbb{R}^{m\times n}$ is the linear observational operator, m is the number of observed components and, $\mathbf{R} \in \mathbb{R}^{m\times m}$ is the estimated data error covariance matrix. The optimal weights in such space reads:

$$\mathbf{r}^a = \hat{\mathbf{P}}^a \cdot \mathbf{Z}^T \cdot \mathbf{R}^{-1} \cdot \left[\mathbf{y} - \mathbf{H}\cdot\bar{\mathbf{x}}^b\right], \tag{6b}$$

therefore, the optimal perturbations can be computed as follows:

$$\mathbf{W}^a = \mathbf{r}^a \otimes \mathbf{1}_N^T + \left[(N-1)\cdot\hat{\mathbf{P}}^a\right]^{1/2} \in \mathbb{R}^{N\times N} \tag{6c}$$

from which, in model space, the analysis reads:

$$\mathbf{X}^a = \bar{\mathbf{x}}^b \otimes \mathbf{1}_N^T + \mathbf{U}\cdot\mathbf{W}^a \in \mathbb{R}^{n\times N}. \tag{6d}$$

The set of equations (6) are applied to each model component in order to compute the global analysis state.

2.3. Ensemble Kalman filter based on modified Cholesky

In [11], the modified Cholesky decomposition is used in order to obtain sparse estimators of the inverse background error covariance matrix. The columns of matrix (5) are assumed normally distributed with moments:

$$\mathbf{u}^{b[i]} \sim \mathcal{N}(\mathbf{0}_n, \mathbf{B}), \quad \text{for } 1 \leq i \leq N, \tag{7}$$

where $\mathbf{B} \in \mathbb{R}^{n\times n}$ is the true unknown background error covariance matrix. Denote by $\mathbf{x}^{[j]} \in \mathbb{R}^{N\times 1}$ the vector holding the jth model component across all the columns of matrix (5), for $1 \leq j \leq n$, following the analysis of Section 2.1, i.e., $\mathbf{S} = \mathbf{U}$, an estimate of the inverse background error covariance matrix reads:

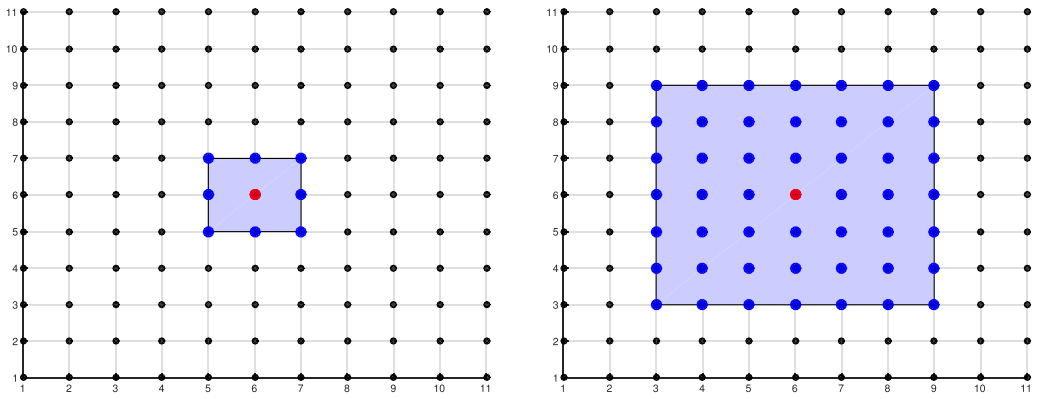
$$\mathbf{B}^{-1} \approx \hat{\mathbf{B}}^{-1} = \mathbf{T}^T \cdot \mathbf{D}^{-1} \cdot \mathbf{T} \in \mathbb{R}^{n\times n}, \tag{8}$$

and similar to (3),

$$\mathbf{B} \approx \hat{\mathbf{B}} = \mathbf{T}^{-1} \cdot \mathbf{D} \cdot \mathbf{T}^{-T} \in \mathbb{R}^{n\times n}. \tag{9}$$

Based on (1), the resulting estimator $\hat{\mathbf{B}}^{-1}$ can be dense. This implies no conditional independence of model components in space which, in practice, can be quite unrealistic for model variables such as wind components, specific humidity and temperature. Thus, a more realistic approximation of $\mathbf{B}^{-1}$ implies a sparse estimator $\hat{\mathbf{B}}^{-1}$. Readily, the structure of $\hat{\mathbf{B}}^{-1}$ depends on the structure of $\mathbf{T}$ this is, on the non-zero coefficients from the regression problems (1). Consequently, if we want to force a particular structure on $\hat{\mathbf{B}}^{-1}$ some of the coefficients in (1) must be set to zero. Thus, we can condition the predecessors of a particular model component to be inside the scope of some radius ζ. This will depend on the manner how the model components are labeled. In practice, row-major and column-major formats are commonly used in the context of data assimilation but, other formats can be used in order to exploit particular features of model discretizations and/or dynamics. For instance, making use of row-major format, consider we want to compute the corresponding set of coefficients for the grid point 6

Please cite this article in press as: E.D. Nino-Ruiz, et al., A parallel implementation of the ensemble Kalman filter based on modified Cholesky decomposition, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.04.005



(a) $\zeta = 1$ (b) $\zeta = 3$

Fig. 1. Local boxes for different radius of influence ζ.

in Fig. 2 for $\zeta = 1$. The local box surrounding the grid point 6 provides the model components inside the scope of ζ. Readily, the predecessors of 6 are the model components labeled from 1 to 5 according to the labelling system utilized.

In general, the analysis increments of the EnKF reads:

$$\mathbf{X}^a = \mathbf{X}^b + \delta\mathbf{X}^a \in \mathbb{R}^{n\times N}, \tag{10}$$

where δX^a is known as the analysis increment. According to the primal formulation of the EnKF, $\hat{\mathbf{B}}^{-1}$ is used in order to compute the analysis correction:

$$\delta\mathbf{X} = \left[\hat{\mathbf{B}}^{-1} + \mathbf{H}^T\cdot\mathbf{R}^{-1}\cdot\mathbf{H}\right]^{-1}\cdot\mathbf{H}^T\cdot\mathbf{R}^{-1}\cdot\left[\mathbf{Y}^s - \mathbf{H}\cdot\mathbf{X}^b\right] \in \mathbb{R}^{n\times N} \tag{11}$$

while, in the dual formulation $\hat{\mathbf{B}}$ is implicitly used:

$$\delta\mathbf{X} = \mathbf{X}\cdot\mathbf{V}^T\cdot\left[\mathbf{R} + \mathbf{V}\cdot\mathbf{V}^T\right]^{-1}\cdot\left[\mathbf{Y}^s - \mathbf{H}\cdot\mathbf{X}^b\right] \in \mathbb{R}^{n\times N}, \tag{12}$$

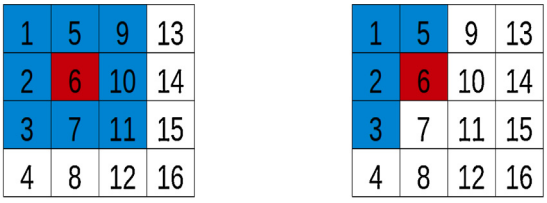
where

$$\mathbf{T}\cdot\mathbf{X} = \mathbf{D}^{1/2} \in \mathbb{R}^{n\times n}, \tag{13}$$

$\mathbf{Y}^s \in \mathbb{R}^{m\times N}$ is the matrix of perturbed observation with data error distribution $\mathcal{N}(\mathbf{0}_m, \mathbf{R})$, and $\mathbf{V} = \mathbf{H}\cdot\mathbf{X} \in \mathbb{R}^{m\times n}$. The primal approach can be employed making use of iterative solvers in order to solve the implicit linear system in (11). On the other hand, the dual approach relies most of its computation in the solution of the unitary triangular linear system in (13). In general, there are good linear solvers in the current literature, some of them well-known and used in operational data assimilation such as the case of LAPACK [12] and CuBLAS [13]. Compact representation of matrices can be used as well in order to exploit the structures of $\hat{\mathbf{B}}^{-1}$ and $\mathbf{T}$ in terms of memory allocation.

3. Proposed parallel implementation of the ensemble Kalman filter based on modified Cholesky decomposition

We consider the use of domain decomposition in order to reduce the dimension of the data assimilation problem. To start, the domain is split according to a given number of sub-domains. Typically, the number of sub-domains matches the number of threads/processors involved in the assimilation process. With no loose of generality, consider the number of sub-domains Δ to be a multiple of n. The total number of model components at each sub-domain is n/Δ but, in order to estimate $\hat{\mathbf{B}}^{-1}$, boundary information is needed which adds $(2\cdot\zeta+1)^2$ model grid points to the procedure of background covariance matrix estimation. If we consider Δ sub-domains, at the kth sub-domain, for $1 \le k \le \Delta$, the analysis reads:



(a) In blue, local box for the model component 6 when $\zeta = 1$. (b) In blue, predecessors of the model component 6 for $\zeta = 1$.

Fig. 2. Local model components (local box) and local predecessors for the model component 6 when $\zeta = 1$. Column-major ordering is utilized to label the model components.

$$\begin{aligned}\mathbf{X}^a_{[k]} = \; & \mathbf{X}^b_{[k]} + \hat{\mathbf{B}}_{[k]}\cdot\mathbf{H}^T_{[k]}\cdot\left[\mathbf{R}_{[k]} + \mathbf{H}_{[k]}\cdot\hat{\mathbf{B}}_{[k]}\cdot\mathbf{H}^T_{[k]}\right] \\ & \cdot\left[\mathbf{Y}^s_{[k]} - \mathbf{H}_{[k]}\cdot\mathbf{X}^b_{[k]}\right] \in \mathbb{R}^{n_{sd}\times N},\end{aligned} \tag{14}$$

where $n_{sd} = n/\Delta + (2\cdot\zeta+1)^2$, and at sub-domain k: $\mathbf{X}^b_{[k]}$ are the model components, $\mathbf{H}_{[k]} \in \mathbb{R}^{m_{sd}\times n_{sd}}$ is the linear observational operator, m_{sd} is the number of observed components in the sub-domain, $\mathbf{Y}^s_{[k]} \in \mathbb{R}^{m_{sd}\times N}$ is the sub-set of perturbed observations, $\hat{\mathbf{B}}^{-1}_{[k]} \in \mathbb{R}^{n_{sd}\times n_{sd}}$ is the local inverse estimation of the background error covariance matrix and $\mathbf{R}_{[k]} \in \mathbb{R}^{m_{sd}\times m_{sd}}$ is the local data-error covariance information. Thus, for all $1 \le k \le \Delta$, the analysis sub-domains (14) are computed, the $(2\cdot\zeta+1)^2$ boundary points are discarded and then, n/Δ analysis points are mapped back onto the global domain. Readily, the dual approach can be used as well. One desired property of the proposed EnKF implementation is that boundary information is not exchanged during the assimilation process, each sub-domain works independently in the estimation of $\hat{\mathbf{B}}^{-1}_{[k]}$ and posterior assimilation of $\mathbf{Y}^s_{[k]}$. In the Algorithm 1, the parallel ensemble Kalman filter based on modified Cholesky decomposition is detailed. The analysis step of this method is shown in the Algorithm 2 wherein, the model state is divided according to the number of sub-domains Δ and then, in parallel, information of the background ensemble, the observed components, the observation operator, the estimated data error correlations at each sub-domain are utilized in order to perform the local assimilations. The analysis sub-domains are then merged into the global analysis state as can be seen in line 9 of the Algorithm 2. This is done as follows:

1. Once each processor finishes the assimilation step, the boundary information is discarded and the analysis sub-domain is sent to the main thread.
2. The main thread receives the analysis sub-domains from the different processors.

Please cite this article in press as: E.D. Nino-Ruiz, et al., A parallel implementation of the ensemble Kalman filter based on modified Cholesky decomposition, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.04.005

3. The local analysis are positioned in their corresponding places of the global domain.

Notice, atomicity is not needed for this operation since analysis sub-domains do not intersect owing to all information concerning to boundaries is discarded after the assimilation step. The local assimilation process is detailed in the Algorithm 3.

Algorithm 1.

Algorithm 1 Parallel ensemble Kalman filter based on modified Cholesky decomposition (PAR-EnKF-MC)

Require: Initial background ensemble $\mathbf{X}^b = \left[\mathbf{x}^{b[1]}, \mathbf{x}^{b[2]}, \ldots, \mathbf{x}^{b[N]}\right] \in \mathbb{R}^{n\times N}$.
Ensure: Analysis ensemble at each assimilation time.
1: **while** There are observations to be assimilated **do**
2: Retrieve $\mathbf{y}$.
3: $\mathbf{Y}^s \leftarrow$ **create_perturbed_observations**$(\mathbf{y}, \mathbf{R})$
4: $\mathbf{X}^a \leftarrow$ **perform_assimilation**$(\mathbf{X}^b, \mathbf{Y}^s, \mathbf{R}, \mathbf{H})$ ▷ Parallel analysis step
5: **for all** $k \leftarrow 1 \rightarrow N$ **do** ▷ Parallel forecast step
6: $\mathbf{x}^{b[k]} \leftarrow \mathcal{M}_{t_{previous}\rightarrow t_{current}}(\mathbf{x}^{a[k]})$
7: **end for**
8: **end while**

Algorithm 2.

Algorithm 2 Assimilation step for the PAR-EnKF-MC

Require: Background ensemble $\mathbf{X}^b \in \mathbb{R}^{n\times N}$, perturbed observations $\mathbf{Y}^s \in \mathbb{R}^{m\times N}$, linearized observation operator $\mathbf{H} \in \mathbb{R}^{m\times N}$, estimated data error covariance matrix $\mathbf{R} \in \mathbb{R}^{m\times m}$.
Ensure: Analysis ensemble $\mathbf{X}^a \in \mathbb{R}^{n\times N}$.
1: **procedure** PERFOFM_ASSIMILATION$(\mathbf{X}^b, \mathbf{Y}^s, \mathbf{R}, \mathbf{H})$ ▷ Ensemble members are stored columnwise
2: Decompose the model states $\mathbf{X}^b$ into Δ sub-domains
3: **for all** $k \leftarrow 1 \rightarrow \Delta$ **do**
4: $\mathbf{X}^b_{[k]} \leftarrow$ **components_from_domain_k**$(\mathbf{X}^b, k)$
5: $\mathbf{H}_{[k]} \leftarrow$ **components_from_domain_k**$(\mathbf{H}, k)$
6: $\mathbf{Y}^s_{[k]} \leftarrow$ **components_from_domain_k**$(\mathbf{Y}^s, k)$
7: $\mathbf{R}_{[k]} \leftarrow$ **components_from_domain_k**$(\mathbf{R}, k)$
8: $\mathbf{X}^a_{[k]} \leftarrow$ **perform_local_assimilation**$(\mathbf{X}^b, \mathbf{Y}^s, \mathbf{R}, \mathbf{H})$
9: $\mathbf{X}^a \leftarrow$ **build_analysis_state**$(\mathbf{X}^a, \mathbf{X}^a_{[k]}, k)$
10: **end for**
11: **return** $\mathbf{X}^a$ ▷ The analysis ensemble is $\mathbf{X}^a$.
12: **end procedure**

Algorithm 3.

Algorithm 3 Local assimilation method

Require: Local background ensemble $\mathbf{X}^b_l \in \mathbb{R}^{n_{sd}\times N}$, local perturbed observations $\mathbf{Y}^s_l \in \mathbb{R}^{m_{sd}\times N}$, local linearized observation operator $\mathbf{H}_l \in \mathbb{R}^{m_{sd}\times N}$, local estimated data error covariance matrix $\mathbf{R}_l \in \mathbb{R}^{m_{sd}\times m}$.
Ensure: Analysis ensemble $\mathbf{X}^a_l \in \mathbb{R}^{n_{sd}\times N}$.
1: **procedure** PERFORM_LOCAL_ASSIMILATION$(\mathbf{X}^b_l, \mathbf{Y}^s_l, \mathbf{R}_l, \mathbf{H}_l)$ ▷ Ensemble members are stored columnwise
2: Estimate $\widehat{\mathbf{B}}_l^{-1}$ based on the samples $\mathbf{X}^b_l$.
3: Perform the assimilation,

$$\mathbf{X}^a_l \leftarrow \mathbf{X}^b_l + \left[\widehat{\mathbf{B}}_l^{-1} + \mathbf{H}_l^T \cdot \mathbf{R}_l^{-1} \cdot \mathbf{H}_l\right]^{-1} \cdot \left[\mathbf{Y}^s_l - \mathbf{H}_l \cdot \mathbf{X}^b_l\right]$$

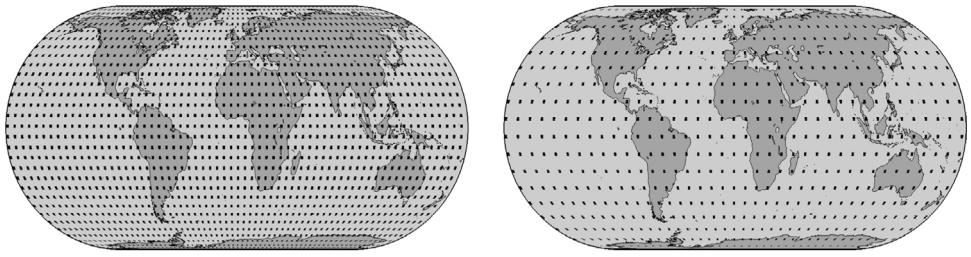
4: **return** $\mathbf{X}^a_l$ ▷ The local analysis ensemble is $\mathbf{X}^a$.
5: **end procedure**

We are now ready to test our proposed parallel implementation of EnKF based on modified Cholesky decomposition.

4. Experimental settings

In this section we study the performance of the proposed parallel ensemble Kalman filter based on modified Cholesky decomposition (PAR-EnKF-MC). The experiments are performed

Please cite this article in press as: E.D. Nino-Ruiz, et al., A parallel implementation of the ensemble Kalman filter based on modified Cholesky decomposition, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.04.005



(a) $p = 12\%$ (b) $p = 4\%$

Fig. 3. Observational networks for different values of p. Dark dots denote the location of the observed components. The observed model variables are the zonal and the meridional wind components, the specific humidity, and the temperature.

using the atmospheric general circulation model SPEEDY [14,15]. SPEEDY is a hydrostatic, spectral coordinate, spectral transform model in the vorticity-divergence form, with semi-implicit treatment of gravity waves. The number of layers in the SPEEDY model is 8 and the T-63 model resolution (192 × 96 grids) is used for the horizontal space discretization of each layer. Four model variables are part of the assimilation process: the temperature (K), the zonal and the meridional wind components (m/s), and the specific humidity (g/kg). The total number of model components is n = 589,824. The number of ensemble members is N = 94 for all the scenarios. The model state space is approximately 6274 times larger than the number of ensemble members ($n \gg N$). The tests are performed on the super computer Blueridge cluster at the university of Virginia Tech. BlueRidge is a 408-node Cray CS-300 cluster. Each node is outfitted with two octa-core Intel Sandy Bridge CPUs and 64 GB of memory, for a total of 6528 cores and 27.3 TB of memory systemwide.

Starting with the state of the system $\mathbf{x}^{\text{ref}}_{-3}$ at time t_{-3}, the model solution $\mathbf{x}^{\text{ref}}_{-3}$ is propagated in time over one year:

$$\mathbf{x}^{\text{ref}}_{-2} = \mathcal{M}_{t_{-3}\rightarrow t_{-2}}\left(\mathbf{x}^{\text{ref}}_{-3}\right).$$

The reference solution $\mathbf{x}^{\text{ref}}_{-2}$ is used to build a perturbed background solution:

$$\hat{\mathbf{x}}^{b}_{-2} = \mathbf{x}^{\text{ref}}_{-2} + \boldsymbol{\epsilon}^{b}_{-2},\ \boldsymbol{\epsilon}^{b}_{-2} \sim \mathcal{N}\left(\mathbf{0}_n,\ \underset{i}{\text{diag}}\left\{\left(0.05\ \{\mathbf{x}^{\text{ref}}_{-2}\}_i\right)^2\right\}\right). \tag{15}$$

The perturbed background solution is propagated over another year to obtain the background solution at time t_{-1}:

$$\mathbf{x}^{b}_{-1} = \mathcal{M}_{t_{-2}\rightarrow t_{-1}}\left(\hat{\mathbf{x}}^{b}_{-2}\right). \tag{16}$$

This model propagation attenuates the random noise introduced in (15) and makes the background state (16) consistent with the physics of the SPEEDY model. Then, the background state (16) is utilized in order to build an ensemble of perturbed background states:

$$\hat{\mathbf{x}}^{b[i]}_{-1} = \mathbf{x}^{b}_{-1} + \boldsymbol{\epsilon}^{b}_{-1},\ \boldsymbol{\epsilon}^{b}_{-1} \sim \mathcal{N}\left(\mathbf{0}_n,\ \underset{i}{\text{diag}}\left\{\left(0.05\ \{\mathbf{x}^{b}_{-1}\}_i\right)^2\right\}\right),$$
$$1 \le i \le N, \tag{17}$$

from which, after three months of model propagation, the initial ensemble is obtained at time t_0:

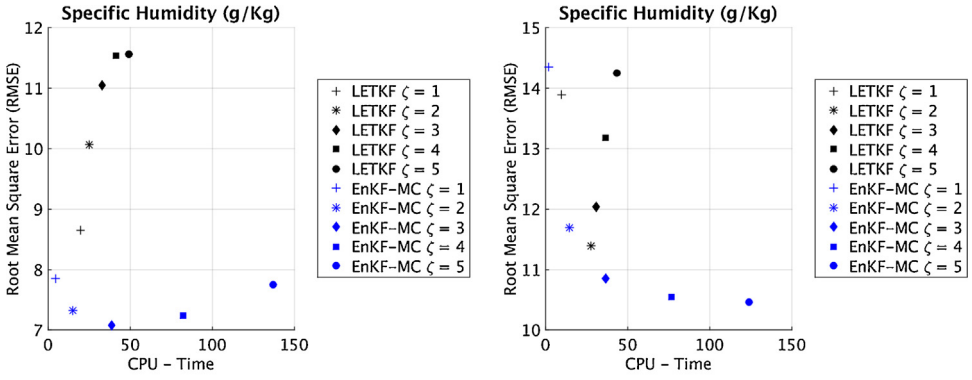
$$\mathbf{x}^{b[i]}_{0} = \mathcal{M}_{t_{-1}\rightarrow t_0}\left(\hat{\mathbf{x}}^{b[i]}_{-1}\right).$$

Again, the model propagation of the perturbed ensemble ensures that the ensemble members are consistent with the physics of the numerical model.

The experiments are performed over a period of 24 days, where observations are taken every 2 days (M = 12). At time k synthetic observations are built as follows:

$$\mathbf{y}_k = \mathbf{H}_k \cdot \mathbf{x}^{\text{ref}}_k + \boldsymbol{\epsilon}_k,\ \boldsymbol{\epsilon}_k \sim \mathcal{N}(\mathbf{0}_m,\ \mathbf{R}_k),\quad \mathbf{R}_k = \text{diag}_i\left\{\left(0.01\ \{\mathbf{H}_k\ \mathbf{x}^{\text{ref}}_k\}_i\right)^2\right\}.$$

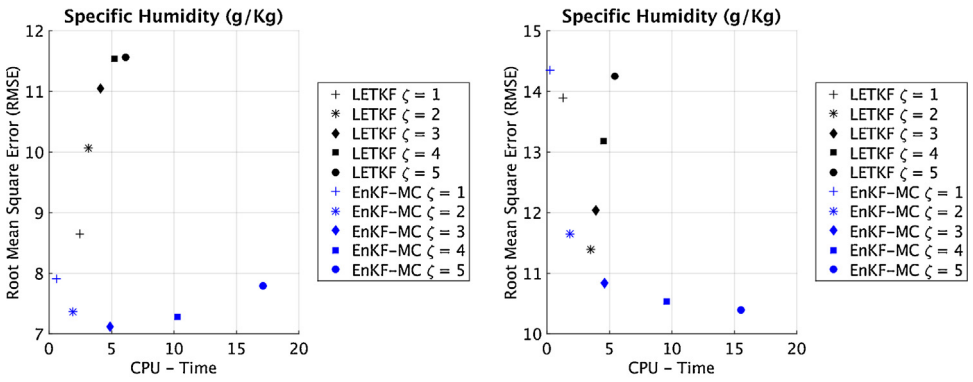
The observation operators $\mathbf{H}_k$ are fixed throughout the time interval. We perform experiments with several operators characterized by different proportions p of observed components from the model state $\mathbf{x}^{\text{ref}}_k$ ($m \approx p \cdot n$). We consider four different values for p: 0.50, 0.12, 0.06 and 0.04 which represent 50%, 12%, 6% and 4% of the total number of model components, respectively. Some of the observational networks used during the experiments are shown in Fig. 3 with their corresponding percentage of observed components from the model state.



(a) $p \sim 50\%$ (b) $p \sim 4\%$

Fig. 4. Relation between CPU-time (s) and accuracy of the compared EnKF implementations for different radii of influence when the number of computing nodes is 6 (96 processors).

Please cite this article in press as: E.D. Nino-Ruiz, et al., A parallel implementation of the ensemble Kalman filter based on modified Cholesky decomposition, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.04.005



(a) $p \sim 50\%$ (b) $p \sim 4\%$

Fig. 5. Relation between CPU-time (s) and accuracy of the compared EnKF implementations for different radii of influence when the number of computing nodes is 48 (768 processors).

The analyses of the PAR-EnKF-MC are compared against those obtained making use of the LETKF implementation proposed by Hunt et al in [16,17]. The analysis accuracy is measured by the root mean square error (RMSE)

$$\text{RMSE} = \sqrt{\frac{1}{M} \cdot \sum_{k=1}^{M} \left[\mathbf{x}_k^{\text{ref}} - \mathbf{x}_k^{\text{a}}\right]^T \cdot \left[\mathbf{x}_k^{\text{ref}} - \mathbf{x}_k^{\text{a}}\right]} \tag{18}$$

where $\mathbf{x}^{\text{ref}} \in \mathbb{R}^{n\times 1}$ and $\mathbf{x}_k^{\text{a}} \in \mathbb{R}^{n\times 1}$ are the reference and the analysis solutions at time k, respectively, and M is the number of assimilation times.

During the assimilation steps, the data error covariance matrices $\mathbf{R}_k$ are used and therefore, no representativeness errors are involved during the assimilation. The different EnKF implementations are performed making use of FORTRAN and specialized libraries such as BLAS and LAPACK are used in order to perform the algebraic computations.

4.1. Influence of the localization radius on analysis accuracy

We study the accuracy of the proposed PAR-EnKF-MC and the LETKF implementations for different radii of influence. The relations between the accuracy of the methods and the radii for 96 and for 768 processors are shown in Figs. 4 and 5, respectively. The results reveal that the accuracy of the PAR-EnKF-MC formulation can be improved by increasing the radius of influence ζ. This implies that the impact of spurious correlations is mitigated when background error correlations are estimated via the modified Cholesky decomposition. However, the larger the radius of influence, the larger the local data assimilation problem to solve. This will demand more computational time which can be mitigated by increasing the number of processors during the assimilation step. On the other hand, in the LETKF context, since background error correlations are estimated based on the empirical moments of the ensemble, spurious correlations affect the analysis when $\zeta > 2$. Consequently, localization radius sizes beyond this value decreases the performance of the LETKF.

4.2. Computational times for different numbers of processors

We compare the elapsed times and the accuracy of both implementations when the number of processors (sub-domains) is increased. We vary the number of compute nodes from 6 (96 processors) to 128 (2,048 processors), fix the radius of influence at $\zeta = 5$, and use an observational network with $p = 4\%$. The elapsed times for different numbers of computing nodes for the PAR-EnKF-MC and LETKF are shown in Fig. 6. As expected, the elapsed time of the LETKF is smaller than that of PAR-EnKF-MC formulation since no covariance estimation is performed. Nevertheless, the difference between the elapsed times is small (in the order of seconds), while the PAR-EnKF-MC results are more accurate than those obtained by the LETKF. Note that for this experiment scalability is lost due to the limited model resolution in respect to the communication costs. For instance, information of analysis sub-domains must travel across network connections in order to build the global analysis state. For small sub-domains (large number of processors), the latency is larger than the processing time required for computing the local analysis at each processor. Thus we expect more speed-up as the model resolution is increased.

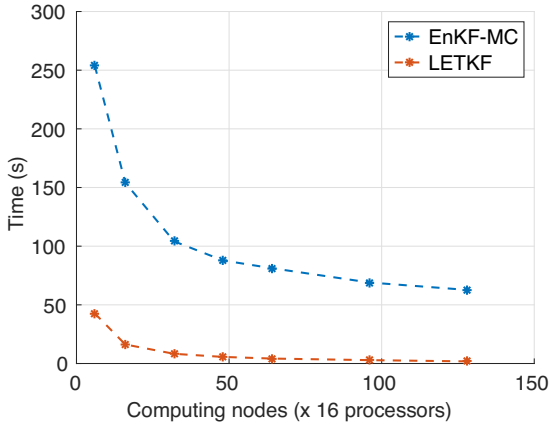


Fig. 6. Elapsed times of the PAR-EnKF-MC and LETKF for different number of compute nodes (×16 processors).

4.3. Influence of the number of processors (sub-domains) on accuracy of PAR-EnKF-MC analyses

An important concern to address in the PAR-EnKF-MC formulation is how its accuracy is impacted when the number of processors (sub-domains) is increased. As we mentioned before, the model domain is decomposed in order to speedup computations but not for increasing the accuracy of the method (i.e., the impact of spurious correlations can be small for small sub-domain sizes) Two main reasons are that we have a well-conditioned estimated of $\mathbf{B}^{-1}$ and even more, the conditional independence of model components makes the sub-domain size to have no impact in the accuracy of the PAR-EnKF-MC. As can be seen in Fig. 7, for the specific humidity variable and values of ζ and p, the PAR-EnKF-MC provides almost

Please cite this article in press as: E.D. Nino-Ruiz, et al., A parallel implementation of the ensemble Kalman filter based on modified Cholesky decomposition, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.04.005

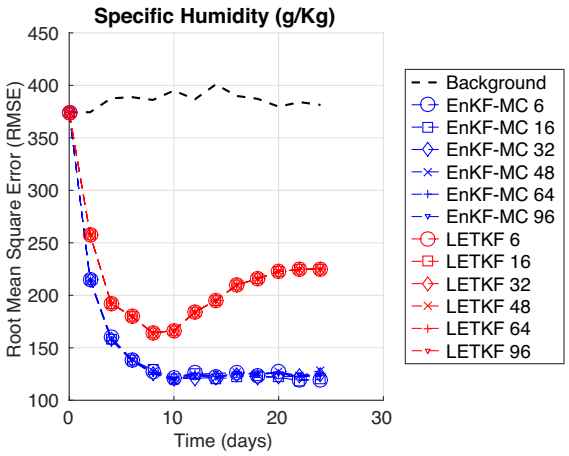


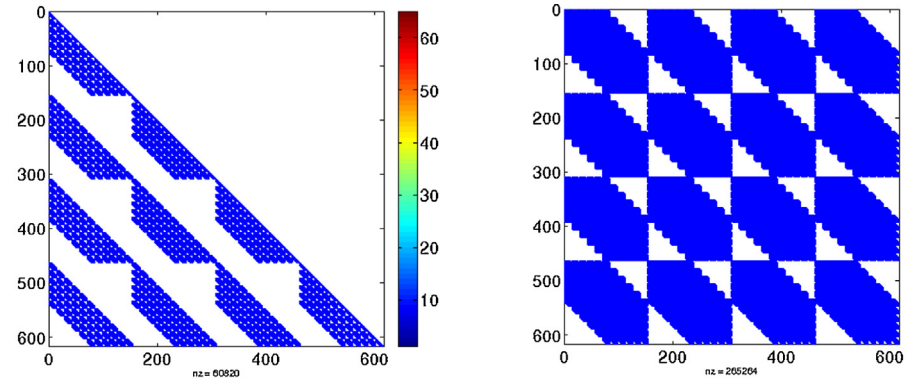
Fig. 7. RMSE of the LETKF and PAR-EnKF-MC implementations for the specific humidity (*sh*) for different numbers of compute nodes. The number of compute nodes is next to the method name.

the same accurate results among all configurations. The small variations in the RMSE values of the PAR-EnKF-MC obey to the synthetic data built at different processors during the assimilation step. For instance, the random number generators used in the experiments depends on the processors id and therefore, the exact synthetic data is not replicated when the number of processors is changed. In the LETKF context we obtain the exact same results for all configurations since it is a deterministic filter and even more, the assimilation is performed for each grid point in the sub-domain.

Lastly, Fig. 8 shows an estimate of a local inverse background error covariance matrix for some sub-domain. Fig. 8a shows the non-zero coefficients in that particular sub-domain, Fig. 8b reflects the structure of $\hat{\mathbf{B}}^{-1}$ based on $\mathbf{T}$. Fig. 8c and d shows the estimated background error covariance matrix $\hat{\mathbf{B}}$ from two different perspectives. As is expected, the correlations are dissipated in space but, they still quite large as can be seen in Fig. 8d. Intuitively, when the sub-domain size is small, high correlations are present between model components owing to their proximity. On the other hand, when the sub-domain size is large, more dissipation is expected on the correlation waves of $\hat{\mathbf{B}}$.

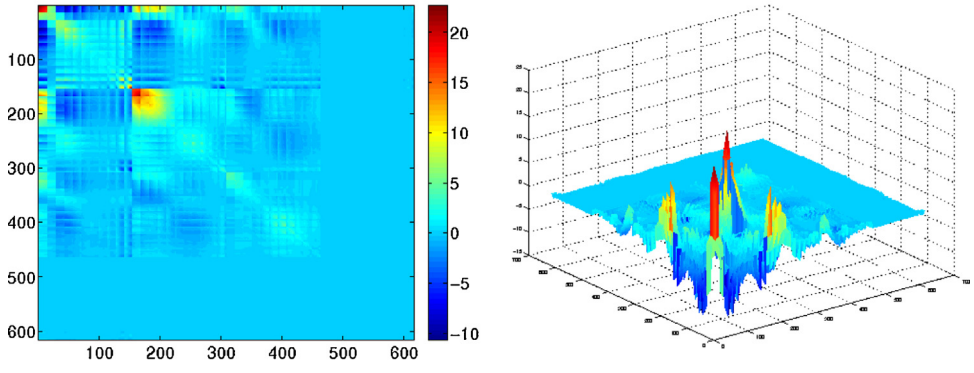
5. Future work

We think there is an opportunity to exploit even more high performance computing tools in the context of PAR-EnKF-MC. Here, most of the computational time is spent in the estimation of the coefficients in (1). The approximation of those coefficients is performed making use of the singular value decomposition (SVD) SVD implementations are highly proposed in the context of accelerating devices such as Many Core Intel (MIC) [18] and the Compute Unified Device Architecture (CUDA) [19]. Since the analysis corrections are computed at each sub-domain independently, each processor (sub-domain) can submit to a given device the information needed in order to solve the linear regression problem (1). Once the solution is computed, the device returns the coefficients to the processor



(a) Structure of $\mathbf{T}$

(b) Suctructure of $\widehat{\mathbf{B}}^{-1}$



(c) $\widehat{\mathbf{B}}$

(d) Surface of $\widehat{\mathbf{B}}$

Fig. 8. Structures of $\mathbf{T}$ and $\hat{\mathbf{B}}^{-1}$ for a radius of influence of $r = 5$. The contourf and surface of $\hat{\mathbf{B}}$ are shown as well. The vector state reads $\mathbf{x} = [u, v, T, sh]^T$.

Please cite this article in press as: E.D. Nino-Ruiz, et al., A parallel implementation of the ensemble Kalman filter based on modified Cholesky decomposition, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.04.005

which assembles the received information in **T**. Generally speaking the process is as follows:

- The domain is split according to Δ processors (sub-domains)
- At each sub-domain a local inverse estimation of the background error covariance matrix is computed:
  - Submit the vectors $\mathbf{x}^{[.]}$ to the assigned device in order to compute the weights in the linear regression (1).
  - In the device, compute the coefficients making use of SVD.
  - The subdomain receives the coefficients from the device.
- The non-zero coefficients are placed in their respective positions in **T**.
- Continue until the coefficients for all local components have been computed.
- Perform the local assimilation.

6. Conclusions

An efficient and parallel implementation of the ensemble Kalman filter based on a modified Cholesky decomposition is proposed. The method exploits the conditional independence of model components in order to obtain sparse estimators of $\mathbf{B}^{-1}$ via the modified Cholesky decomposition. High performance computing can be used in order to speedup the assimilation process: the global domain is decomposed according to the number of processors (sub-domains), at each sub-domain a local estimator of the inverse background error covariance matrix is computed and the local assimilation process is carried out. Each sub-domain is then mapped back onto the global domain where then, the global analysis is obtained. The proposed EnKF implementation is compared against the well-known local ensemble transform Kalman filter (LETKF) making use of the Atmospheric General Circulation Model (SPEEDY) with the T-63 resolution in the super computer cluster Blueridge at Virginia Tech. The number of processors is ranged from 96 to 2048. The accuracy of the proposed EnKF outperforms that of the LETKF. Even more, the computational time of the proposed implementation differs in seconds of the parallel LETKF method in which no covariance estimation is performed. Finally, for the largest number of processors, the proposed method is 400 times faster than its serial theoretical implementation.

Acknowledgements

This work was supported in part by awards NSFCCF-1218454, AFOSRFA9550-12-1-0293-DEF, and by the Computational Science Laboratory at Virginia Tech.

References

[1] L. Nerger, W. Hiller, Software for ensemble-based data assimilation systems – implementation strategies and scalability, Comput. Geosci. 55 (2013) 110–118.
[2] V. Rao, A. Sandu, A time-parallel approach to strong-constraint four-dimensional variational data assimilation, J. Comput. Phys. 313 (2016) 583–593.
[3] Y. Liu, A. Weerts, M. Clark, H. Hendricks Franssen, S. Kumar, H. Moradkhani, D. Seo, D. Schwanenberg, P. Smith, A. Van Dijk, et al., Advancing data assimilation in operational hydrologic forecasting: progresses, challenges, and emerging opportunities, Hydrol. Earth Syst. Sci. 16 (10) (2012).
[4] M. Zupanski, Theoretical and practical issues of ensemble data assimilation in weather and climate, in: S. Park, L. Xu (Eds.), Data Assimilation for Atmospheric, Oceanic and Hydrologic Applications, Springer, Berlin/Heidelberg, 2009, pp. 67–84, http://dx.doi.org/10.1007/978-3-540-71056-1_3.
[5] A.A. Emerick, A.C. Reynolds, Ensemble smoother with multiple data assimilation, Comput. Geosci. 55 (2013) 3–15, http://dx.doi.org/10.1007/978-3-540-71056-1_3.
[6] E. Ott, B.R. Hunt, I. Szunyogh, A.V. Zimin, E.J. Kostelich, M. Corazza, E. Kalnay, D.J. Patil, J.A. Yorke, A local ensemble Kalman filter for atmospheric data assimilation, Tellus A 56 (5) (2004) 415–428, http://dx.doi.org/10.1111/j.1600-0870.2004.00076.x.
[7] G. Evensen, The ensemble Kalman filter: theoretical formulation and practical implementation, Ocean Dyn. 53 (4) (2003) 343–367, http://dx.doi.org/10.1007/s10236-003-0036-9.
[8] P.J. Bickel, E. Levina, Regularized estimation of large covariance matrices, Ann. Stat. 36 (1) (2008) 199–227, http://dx.doi.org/10.1214/009053607000000758.
[9] P. Sakov, L. Bertino, Relation between two common localisation methods for the EnKF, Comput. Geosci. 15 (2) (2011) 225–237, http://dx.doi.org/10.1007/s10596-010-9202-6.
[10] B.R. Hunt, E.J. Kostelich, I. Szunyogh, Efficient data assimilation for spatiotemporal chaos: a local ensemble transform Kalman filter, Phys. D: Nonlinear Phenom. 230 (1) (2007) 112–126.
[11] E.D. Nino, A. Sandu, W. Deng, An ensemble Kalman filter implementation based on modified Cholesky decomposition for inverse covariance matrix estimation, arXiv preprint arXiv:1572695.
[12] E. Anderson, Z. Bai, J. Dongarra, A. Greenbaum, A. McKenney, J. Du Croz, S. Hammerling, J. Demmel, C. Bischof, D. Sorensen, LAPACK: a portable linear algebra library for high-performance computers, in: Proceedings of the 1990 ACM/IEEE Conference on Supercomputing, Supercomputing '90, IEEE Computer Society Press, Los Alamitos, CA, USA, 1990, pp. 2–11 http://dl.acm.org/citation.cfm?id=110382.110385.
[13] L.S. Blackford, J. Demmel, J. Dongarra, I. Duff, S. Hammarling, G. Henry, M. Heroux, L. Kaufman, A. Lumsdaine, A. Petitet, R. Pozo, K. Remington, R.C. Whaley, An updated set of basic linear algebra subprograms (BLAS), ACM Trans. Math. Softw. 28 (2001) 135–151.
[14] F. Molteni, Atmospheric simulations using a GCM with simplified physical parametrizations. I: Model climatology and variability in multi-decadal experiments, Climate Dyn. 20 (2–3) (2003) 175–191, http://dx.doi.org/10.1007/s00382-002-0268-2.
[15] F. Kucharski, F. Molteni, A. Bracco, Decadal interactions between the western tropical pacific and the north Atlantic oscillation, Climate Dyn. 26 (1) (2006) 79–91, http://dx.doi.org/10.1007/s00382-005-0085-5.
[16] K. Terasaki, M. Sawada, T. Miyoshi, Local ensemble transform Kalman filter experiments with the nonhydrostatic icosahedral atmospheric model NICAM, SOLA 11 (2015) 23–26.
[17] T. Miyoshi, M. Kunii, The local ensemble transform Kalman filter with the weather research and forecasting model: experiments with real observations, Pure Appl. Geophys. 169 (3) (2012) 321–333.
[18] M. Huang, C. Lai, X. Shi, Z. Hao, H. You, Study of parallel programming models on computer clusters with Intel MIC coprocessors, Int. J. High Perform. Comput. Appl. (2015), http://dx.doi.org/10.1177/1094342015580864, arXiv:http://hpc.sagepub.com/content/early/2015/04/10/1094342015580864.full.pdf+html, http://hpc.sagepub.com/content/early/2015/04/10/1094342015580864.abstract.
[19] S. Lahabar, P. Narayanan, Singular value decomposition on GPU using CUDA, IEEE International Symposium on Parallel Distributed Processing, 2009. IPDPS 2009 (2009) 1–10, http://dx.doi.org/10.1109/IPDPS.2009.5161058.

Please cite this article in press as: E.D. Nino-Ruiz, et al., A parallel implementation of the ensemble Kalman filter based on modified Cholesky decomposition, J. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.jocs.2017.04.005