Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Mathematical Sciences

Håkon Gryvill

Hierarchical ensemble Kalman filter formulated with sparse matrices

Master's thesis in Applied Physics and Mathematics
Supervisor: Håkon Tjelmeland
June 2020

NTNU
Norwegian University of Science and Technology

Håkon Gryvill

Hierarchical ensemble Kalman filter formulated with sparse matrices

Master's thesis in Applied Physics and Mathematics
Supervisor: Håkon Tjelmeland
June 2020

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Mathematical Sciences

NTNU
Norwegian University of
Science and Technology

Preface

This thesis is a part of the Master's degree in Industrial Mathematics at the Department of Mathematical sciences, at the University of Science and Technology (NTNU). The report was written over a time span of 21 weeks during the spring of 2020.

I would like to thank my supervisor, Håkon Tjelmeland, for his mentoring during the last year. By bringing up discussions regarding the topics presented in this thesis, he has helped me understand the ideas behind the various concepts. Additionally, he has given me valuable feedback on my drafts, which has been very helpful to me. I would especially like to thank him for his supervision during the time period when the coronavirus prevented us to meet in person.

Håkon Gryvill
June 2020
Trondheim, Norway

Abstract

The hierarchical ensemble Kalman filter (HEnKF), introduced in Omre and Myrseth (2010), is an extension of the ensemble Kalman filter (EnKF). By imposing a hierarchical model on the state space variables, HEnKF has shown to yield more robust results than EnKF. However, as a consequence of this, HEnKF is computationally demanding, especially for high-dimensional systems. By imposing a Gaussian Markov random field (GMRF) on the state space variables, we are able to reduce the computational cost of HEnKF.

In this thesis, we propose a new prior distribution for the model parameters of the state space variables, where we assume a GMRF. We argue that we are able to reduce computational cost of HEnKF, by utilizing the sparse matrix structure provided by the GMRF. Two numerical examples are presented, where results provided by the prior distribution originally used in HEnKF are compared to results provided by the prior distribution presented in this thesis.

In both of the numerical examples, the prior distribution presented in this thesis is able to provide a considerable reduction in computational demand, compared to the prior distribution originally used in HEnKF. The prior distribution introduced in this thesis is also able to produce reliable results in both examples, even when the state space variable is high-dimensional, while the quality of the results provided by the prior distribution originally used in HEnKF decreases as the dimension of the state space variable increases. The theory presented in this thesis suggests that the computational complexity of HEnKF applying the presented prior distribution is linear as a function of the dimension of the state space variable. From the numerical results presented in this thesis, we observe that the computational complexity is somewhat higher.

Sammendrag

Hierarkisk ensemble Kalmanfilter (HEnKF), introdusert i Omre and Myrseth (2010), er en utvidelse av ensemble Kalmanfilter (EnKF). Ved å ilegge en hierarkisk modell på state space-variablene har HEnKF vist å gi mer robuste resultater enn EnKF. En konsekvens av dette er imidlertid at HEnKF er beregningsmessig krevende, spesielt for høydimensjonale systemer. Ved å ilegge et Gaussisk Markovfelt (GMRF) på state space-variablene er vi i stand til å redusere den beregningsmessige kostnaden ved HEnKF.

I denne oppgaven foreslår vi en ny apriorifordeling for modellparameterne til state space-variablene, hvor vi antar en GMRF. Vi argumenterer for at vi er stand til å redusere den beregningsmessige kostnaden i HEnKF, ved å anvende den glisne matrisestrukturen vi oppnår gjennom en GMRF. To numeriske eksempler er presentert, hvor resultater anskaffet med apriorifordelingen opprinnelig brukt i HEnKF er sammenlignet med resultater anskaffet med apriorifordelingen presentert i denne oppgaven.

I begge de numeriske eksemplene er apriorifordelingen presentert i denne oppgaven i stand til å redusere den beregningsmessige kostnaden, sammenlignet med apriorifordelingen opprinnelig brukt i HEnKF. Apriorifordelingen introdusert i denne oppgaven i stand til å produsere pålitelige resultater i begge eksempler, selv når state space-variabelen er høydimensjonal, mens kvaliteten på resultatene med apriorifordelingen opprinnelig brukt i HEnKF avtar når dimensjonen på state space-variabelen øker. Teorien presentert i denne oppgaven antyder at den beregningsmessige kompleksiteten for HEnKF med den presenterte apriorifordelingen er lineær som funksjon av dimensjonen på state space-variabelen. Fra de numeriske eksemplene presentert i denne oppgaven observerer vi en beregningsmessig kompleksitet som er noe høyere.

1 Introduction

In many cases, we wish to predict the outcome of a future event. This event can be described by a latent variable, denoted $\boldsymbol{x}_{T+1}$, where T is the current time-step. Suppose that this latent variable is related to the latent variables from the previous time-steps, $\boldsymbol{x}_0, \ldots, \boldsymbol{x}_T$. Further, suppose that each of these latent variables $\boldsymbol{x}_t$ give rise to an observation $\boldsymbol{d}_t$, for $t = 0, \ldots, T$. Our primary objective is to assess the one-step forecasting problem, which is to predict the latent variable $\boldsymbol{x}_{T+1}$ given the set of observations $\boldsymbol{d}_0, \ldots, \boldsymbol{d}_T$.

The state space model (Brockwell and Davis, 1991, chap. 12.1) defines a set of properties regarding the relationship between the observations $\boldsymbol{d}_0, \ldots, \boldsymbol{d}_T$ and latent variables $\boldsymbol{x}_0, \ldots, \boldsymbol{x}_{T+1}$. By assuming a few additional properties regarding Gaussianity and linearity, the Kalman filter, introduced in Kalman (1960), is able to produce an analytic solution to the one-step forecasting problem. This filter was originally applied in the tracking of spacecrafts, where $\boldsymbol{x}_t$ defines the position of the spacecraft at time-step t, and where the observation $\boldsymbol{d}_t$ defines its velocity and azimuth at the same time-step. The relationship between the observations and the latent variables are defined by the motion equations.

However, when at least one of the assumed properties are invalid, the Kalman filter is analytically unfeasible, and the filter must be approximated. There exists several approximations, such as the extended Kalman filter (Gordon et al., 1993), the randomized maximum likelihood filter (Oliver, 1996), the particle filter (Arulampalam et al., 2002) and the unscented Kalman filter (Julier and Uhlmann, 1997). The ensemble Kalman filter, introduced in Evensen (1994), approximates the Kalman filter when the relationship between two consecutive latent variables is assumed Gaussian and nonlinear. Applications of the ensemble Kalman filter includes meteorology, see Houtekamer and Mitchell (1997).

Although the ensemble Kalman filter has proven to yield reliable results in many applications, the filter struggles with a few artifacts. These artifacts are mainly related to estimation of the covariance matrix. The hierarchical ensemble Kalman filter, introduced in Omre and Myrseth (2010), manages to reduce the impact of these artifacts, by imposing a hierarchical model on the latent variables. This entails enforcing prior distributions on the model parameters. However, the hierarchical ensemble Kalman filter faces some issues regarding computational demands. These issues partially stem from the choice of prior distributions on the model parameters, and become more severe as the dimensions of the problem increase.

We believe that by imposing a different set of prior distributions on the model parameters, we are able to reduce the computational demands of the hierarchical ensemble Kalman filter, without notably reducing the quality of the results. By assuming that the elements of the latent variables follow a Markov structure, we are able to obtain a sparse precision matrix, i.e. the inverse of the covariance matrix. This enables us to reduce the number of computations performed in the hierarchical ensemble Kalman filter. The extent of this reduction relies on the structure of the Markov structure imposed on the latent variables. Since we are able to change the structure of the Markov chain as we choose, we are given a

wide range of options. That is, we are given more freedom to find a suitable hierarchical model for each situation.

In Section 2, we present the theory necessary to understand the new concepts presented in this report, which is introduced in Section 3. Section 4 introduces the setup of the numerical examples performed in this report. The results and the discussion of these are presented in Section 5. Lastly, Section 6 gives a small summary and conclusion of the results.

2 Background

This section presents the theory that is necessary for understanding the new material introduced in Section 3. Section 2.1 presents the state space model, which sculpts the foundation of our problem setup. The linear Gaussian state space model, which is a special case of the state space model, is introduced in Section 2.2. Section 2.3 presents the Kalman filter, while Section 2.4 introduces an approximation of the Kalman filter, namely the ensemble Kalman filter. The hierarchical ensemble Kalman filter is presented in Section 2.5, which is an extension of the ensemble Kalman filter. Section 2.6 introduces the Gaussian Markov random field, while Section 2.7 discusses how Cholesky decomposition can be applied to efficiently compute equations involving band matrices. Sections 2.5, 2.6 and 2.7 are especially important, as the new material presented in Section 3 combines the theory introduced in these sections.

2.1 State space model

This section presents the state space model (SSM), which is the underlying model for the problem we consider. The state space model is both defined mathematically and illustrated visually. For more information about the state space model, see Omre and Myrseth (2011) and Brockwell and Davis (1991).

The state space model consists of two sequences that are linked together; a sequence of observations, denoted $\boldsymbol{d}_{0:T} = (\boldsymbol{d}_0, \boldsymbol{d}_1, \ldots, \boldsymbol{d}_T)$, and a sequence of latent variables $\boldsymbol{x}_{0:T+1} = (\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_{T+1})$. The vector $\boldsymbol{d}_t$ is of length D, $\boldsymbol{d}_t \in \mathbb{R}^D$, and denotes the observation at an arbitrary time-step t. Similarly, $\boldsymbol{x}_t$ is a vector of length K, $\boldsymbol{x}_t \in \mathbb{R}^K$, and denotes the latent variable at time t. Throughout this report, vectors are printed in bold, and the index t is exclusively used for denoting time-step. Also, we choose to write $\boldsymbol{d}_{0:T}$ and $\boldsymbol{x}_{0:T+1}$ in bold, as it is more reasonable to consider these as sequences of vectors, rather than matrices.

Shumway and Stoffer (2016, chap. 6.2) explain that estimating $\boldsymbol{x}_t|\boldsymbol{d}_{0:T}$ is called *forecasting* for $t > T$, while it is called *filtering* for $t = T$, and *smoothing* for $t < T$. In this report, our main objective is to assess the one-step forecasting problem, i.e. estimating $\boldsymbol{x}_{T+1}$ given all of the observations available at time T, $\boldsymbol{d}_{0:T} = (\boldsymbol{d}_0, \boldsymbol{d}_1, \ldots, \boldsymbol{d}_T)$. We also want to assess the filtering problem, i.e. estimate $\boldsymbol{x}_t$ given $\boldsymbol{d}_{0:t}$ for $t = 0, \ldots, T$. Assessment of the forecasting and filtering

problems are discussed further in Section 2.3. In the following, we describe the properties of the state space model.

We first consider the sequence of latent variables, $\boldsymbol{x}_{0:T+1} = (\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_{T+1})$. Each latent variable $\boldsymbol{x}_{t+1}$, given $\boldsymbol{x}_t$, is conditionally independent of the previous latent variables, for $t = 1, \ldots, T+1$. That is, the sequence of latent variables $\boldsymbol{x}_{0:T+1}$ follows a first order Markov chain

$$f(\boldsymbol{x}_{t+1}|\boldsymbol{x}_{0:t}) = f(\boldsymbol{x}_{t+1}|\boldsymbol{x}_t). \tag{1}$$

Alternatively, we can define a *forward*-function, denoted $w(.)$, which specifies the relationship between two consecutive latent variables

$$\boldsymbol{x}_{t+1} = w_t(\boldsymbol{x}_t, \boldsymbol{v}_t), \tag{2}$$

where $\boldsymbol{v}_t$ is a noise term. This is called the *forward* model. Note that the state space model makes no distributional assumptions about the noise term $\boldsymbol{v}_t$.

We now proceed to define the relationship between the observations $\boldsymbol{d}_{0:T}$ and the latent variables $\boldsymbol{x}_{0:T+1}$. First, each observation $\boldsymbol{d}_t$ is conditionally independent of the remaining observations, given $\boldsymbol{x}_t$, for $t = 0, \ldots, T$. To denote every observation in $\boldsymbol{d}_{0:T}$ except $\boldsymbol{d}_t$, we write $\boldsymbol{d}_{-t}$. Similarly, we denote every latent variable in $\boldsymbol{x}_{0:T+1}$ except $\boldsymbol{x}_t$ as $\boldsymbol{x}_{-t}$. This enables us to describe the conditional independence between the observations as follows

$$f(\boldsymbol{d}_t|\boldsymbol{d}_{-t}, \boldsymbol{x}_{0:T+1}) = f(\boldsymbol{d}_t|\boldsymbol{x}_{0:T+1}), \quad t = 0, \ldots, T. \tag{3}$$

Second, $\boldsymbol{d}_t$ is conditionally independent of the remaining latent variables given $\boldsymbol{x}_t$. We use the term *single-state* dependence to describe this property. The single-state dependence can be formulated as

$$f(\boldsymbol{d}_t|\boldsymbol{x}_{0:T+1}) = f(\boldsymbol{d}_t|\boldsymbol{x}_t), \quad t = 0, \ldots, T. \tag{4}$$

Note that the right hand side of (3) equals the left hand side of (4). These two properties enables us to write the following expression

$$f(\boldsymbol{d}_{0:T}|\boldsymbol{x}_{0:T+1}) = \prod_{t=0}^{T} f(\boldsymbol{d}_t|\boldsymbol{x}_t), \tag{5}$$

which we denote as the likelihood model.

Figure 1 illustrates the connection between the two sequences $\boldsymbol{d}_{0:T}$ and $\boldsymbol{x}_{0:T+1}$. The edges indicate the causal dependencies between the latent variables and the observations, and within the latent variables. We see that there are only edges between a latent variable and the previous. This indicates the Markov property between the latent variables, (2). We also see that there are no edges between observations, only between a latent variable and the observation at the same time-step. This enlightens the conditional independence between the observations, (3) and the single-state dependence, (4). Note that no distributional assumptions are made in the state space model. In the following section, we present the linear Gaussian state-space model, which makes further assumptions about the likelihood and forward models.

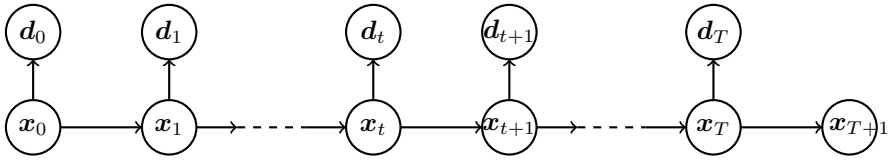


Figure 1: A hidden Markov model. The edges illustrate the stochastic dependencies between the nodes. The latent variables $\boldsymbol{x}_t, t = 0, \ldots, T+1$ are unobserved, while the observations $\boldsymbol{d}_t, t = 0, \ldots, T$ are observed.

2.2 Linear Gaussian state space model

This section presents the linear Gaussian state space model, which is a special case of the state space model. We assume the same model as in Section 2.1. For more information about the linear Gaussian state space model, see Omre and Myrseth (2011) and Brockwell and Davis (1991).

The linear Gaussian state space model assumes a state space model, as specified in Section 2.1. We recall that this model satisfies (1) to (5), and is illustrated in Figure 1. However, the linear Gaussian state space model makes further assumptions about the connections between the observations and latent variables. In the following, we present these properties.

First, we assume that the pdf of the latent variable in the initial time-step is Gaussian, i.e.

$$\boldsymbol{x}_0 \sim N(\boldsymbol{\mu}_0, \Sigma_0), \tag{6}$$

where $\boldsymbol{\mu}_0 \in \mathbb{R}^K$ is the mean vector and $\Sigma_0 \in \mathbb{R}^{K \times K}$ is the covariance matrix of $\boldsymbol{x}_0$. Second, we make further assumptions about the forward model, presented in (2). We assume that the latent variable $\boldsymbol{x}_{t+1}$ given $\boldsymbol{x}_t$, is normally distributed around a linear transformation of $\boldsymbol{x}_t$. That is,

$$\boldsymbol{x}_{t+1}|\boldsymbol{x}_t \sim N(A_t\boldsymbol{x}_t, \Sigma_x), \tag{7}$$

for some matrix $A_t \in \mathbb{R}^{K \times K}$, and some covariance matrix $\Sigma_x \in \mathbb{R}^{K \times K}$. Since the pdf of $\boldsymbol{x}_{t+1}|\boldsymbol{x}_t$ is Gaussian around a linear transformation of $\boldsymbol{x}_t$, we say that the forward model is *Gauss-linear*. By defining a random variable $\boldsymbol{v}_t \sim N(\boldsymbol{0}, \Sigma_x)$, we are able to make an alternative formulation of the forward model

$$\boldsymbol{x}_{t+1} = A_t\boldsymbol{x}_t + \boldsymbol{v}_t. \tag{8}$$

Second, we make assumptions about the likelihood model for $\boldsymbol{d}_t$ given $\boldsymbol{x}_t$. We assume that $\boldsymbol{d}_t$ given $\boldsymbol{x}_t$ is Gauss-linear as well. That is,

$$\boldsymbol{d}_t|\boldsymbol{x}_t \sim N(H\boldsymbol{x}_t, \Sigma_d), \tag{9}$$

for some matrix $H \in \mathbb{R}^{D \times K}$, and some covariance matrix $\Sigma_d \in \mathbb{R}^{D \times D}$. Alternatively, by defining a variable $\boldsymbol{u}_t \sim N(\boldsymbol{0}, \Sigma_d)$, we can formulate this property as

$$\boldsymbol{d}_t = H\boldsymbol{x}_t + \boldsymbol{u}_t. \tag{10}$$

In the following section, we present an algorithm for computing $f(\boldsymbol{x}_{T+1}|\boldsymbol{d}_{0:T})$. This algorithm relies on the properties in (6), (7) and (9) in order to provide results that are analytically tractable.

2.3 Kalman filter

This section presents the Kalman filter, and explains the filter step-by-step. We outline the presentation of the filter in an algorithm. We assume the same model as specified in Sections 2.1 and 2.2. For more information about the Kalman filter, see Kalman (1960), Shumway and Stoffer (2016) and Omre and Myrseth (2011).

The Kalman filter, introduced in Kalman (1960), is a recursive algorithm that creates estimates of a latent variable $\boldsymbol{x}_t$, given the observations $\boldsymbol{d}_{0:t} = (\boldsymbol{d}_0, \dots, \boldsymbol{d}_t)$. More specifically, the filter is able to provide an analytic expression for $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$, for all $t = 0, \dots, T$ for linear Gaussian state space models. We recall that this is called *filtering* (Shumway and Stoffer, 2016, chap. 6.2). The Kalman filter is also able to solve the one-step forecasting problem, i.e. to compute an analytical expression for $f(\boldsymbol{x}_{T+1}|\boldsymbol{d}_{0:T})$, which is our primary objective. Before we introduce the Kalman filter, we present two theorems that are necessary for proving the Kalman filter.

Theorem 1. *Assume that* (3) *and* (9) *hold, and*

$$\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1} \sim N(\boldsymbol{\mu}_{t|0:t-1}, \Sigma_{t|0:t-1}), \tag{11}$$

then

$$\boldsymbol{x}_t|\boldsymbol{d}_{0:t} \sim N(\boldsymbol{\mu}_{t|0:t}, \Sigma_{t|0:t}), \tag{12}$$

where

$$\boldsymbol{\mu}_{t|0:t} = \boldsymbol{\mu}_{t|0:t-1} + K_{KF}(\boldsymbol{d}_t - H\boldsymbol{\mu}_{t|0:t-1}) \tag{13}$$

$$\Sigma_{t|0:t} = (I - K_{KF}H)\Sigma_{t|0:t-1} \tag{14}$$

and where $K_{KF} = \Sigma_{t|0:t-1}H^\intercal(H\Sigma_{t|0:t-1}H^\intercal + \Sigma_d)^{-1}$. *The matrices* H *and* Σ_d *are defined in* (9).

The full proof of Theorem 1 is presented in Appendix A. In the following, we present a sketch of the proof to give the reader a better understanding of the Kalman filter. To derive the distribution for $\boldsymbol{x}_t|\boldsymbol{d}_{0:t}$, we apply Bayes' rule

$$f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}) = \frac{f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})f(\boldsymbol{d}_t|\boldsymbol{x}_t, \boldsymbol{d}_{0:t-1})}{f(\boldsymbol{d}_t|\boldsymbol{d}_{0:t-1})}. \tag{15}$$

Since we assume that the observations are conditionally independent, (3), we have that

$$f(\boldsymbol{d}_t|\boldsymbol{x}_t, \boldsymbol{d}_{0:t-1}) = f(\boldsymbol{d}_t|\boldsymbol{x}_t). \tag{16}$$

Thus,

$$f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}) = \frac{f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})f(\boldsymbol{d}_t|\boldsymbol{x}_t)}{f(\boldsymbol{d}_t|\boldsymbol{d}_{0:t-1})}. \tag{17}$$

The expression for $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})$ is given in (11), while the expression for $f(\boldsymbol{d}_t|\boldsymbol{x}_t)$ is given in (9). By inserting these expressions into (17), we notice that $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$ is Gaussian with the parameters defined in (13) and (14). Now we proceed to the second theorem necessary to prove the Kalman filter.

Theorem 2. *Assume that* (1) *and* (7) *hold, and*

$$\boldsymbol{x}_t|\boldsymbol{d}_{0:t} \sim N(\boldsymbol{\mu}_{t|0:t}, \Sigma_{t|0:t}), \tag{18}$$

then

$$\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t} \sim N(\boldsymbol{\mu}_{t+1|0:t}, \Sigma_{t+1|0:t}), \tag{19}$$

where

$$\boldsymbol{\mu}_{t+1|0:t} = A_t\boldsymbol{\mu}_{t|0:t}, \tag{20}$$

$$\Sigma_{t+1|0:t} = \Sigma_x + A_t\Sigma_{t|0:t}A_t^\intercal, \tag{21}$$

where A_t *and* Σ_x *are defined in* (7).

The full proof of Theorem 2 is given in Appendix B. In the following, we present a sketch of the proof to give the reader some notion of how the Kalman filter works. The distribution for $\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t}$ can be assessed by computing

$$f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t}) = \int f(\boldsymbol{x}_{t+1}, \boldsymbol{x}_t|\boldsymbol{d}_{0:t})d\boldsymbol{x}_t. \tag{22}$$

By rewriting the integrand, we have

$$f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t}) = \int f(\boldsymbol{x}_{t+1}|\boldsymbol{x}_t, \boldsymbol{d}_{0:t})f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})d\boldsymbol{x}_t. \tag{23}$$

Since we assume that the latent variables follow a first order Markov chain, (1), we have

$$f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t}) = \int f(\boldsymbol{x}_{t+1}|\boldsymbol{x}_t)f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})d\boldsymbol{x}_t. \tag{24}$$

The first factor in the integrand is given in (7), while the second factor is given in (18). By inserting these expressions into the equation above, we see that $\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t}$ is Gaussian with the parameters given in (20) and (21). The expressions for the filtering and one-step forecasting distributions are presented in the following theorem.

Theorem 3. *Assume a linear Gaussian state space model, that is, where* (1), (3), (4), (6), (7) *and* (9) *hold, then*

$$\boldsymbol{x}_t|\boldsymbol{d}_{0:t} \sim N(\boldsymbol{\mu}_{t|0:t}, \Sigma_{t|0:t}), \tag{25}$$

$$\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t} \sim N(\boldsymbol{\mu}_{t+1|0:t}, \Sigma_{t+1|0:t}), \tag{26}$$

for $t = 0, \ldots, T$, where

$$\boldsymbol{\mu}_{t|0:t} = \boldsymbol{\mu}_{t|0:t-1} + K_{KF}(\boldsymbol{d}_t - H\boldsymbol{\mu}_{t|0:t-1}), \tag{27}$$

$$\Sigma_{t|0:t} = (I - K_{KF}H)\Sigma_{t|0:t-1}, \tag{28}$$

$$\boldsymbol{\mu}_{t+1|0:t} = A_t\boldsymbol{\mu}_{t|0:t}, \tag{29}$$

$$\Sigma_{t+1|0:t} = A_t\Sigma_{t|0:t}A_t^\intercal + \Sigma_x, \tag{30}$$

where $K_{KF} = \Sigma_{t|0:t-1}H^\intercal(H\Sigma_{t|0:t-1}H + \Sigma_d)^{-1}$ is the Kalman gain matrix for the Kalman filter, $\boldsymbol{\mu}_{0|0:-1} = \boldsymbol{\mu}_0$ and $\Sigma_{0|0:-1} = \Sigma_0$ are defined in (6)*, A_t and Σ_x are defined in* (7)*, and H and Σ_d are defined in* (9)*.*

Proof. In order to prove this theorem, i.e. to prove that (25) and (26) hold for $t = 0, \ldots, T$, we structure the proof as a "proof by induction". We first prove that the theorem holds for the initial time-step $t = 0$. We then prove that the theorem holds for time-step t if we assume that it holds for $t-1$. In the following, we prove that (25) and (26) hold for $t = 0$.

In order to prove that (25) holds for $t = 0$, we apply Theorem 1. Since we assume that (3) and (9) hold, and that $\boldsymbol{x}_0 \sim N(\boldsymbol{\mu}_0, \Sigma_0)$ from (6), we have from Theorem 1 that

$$\boldsymbol{x}_0|\boldsymbol{d}_0 \sim N(\boldsymbol{\mu}_{0|0:0}, \Sigma_{0|0:0}), \tag{31}$$

where

$$\boldsymbol{\mu}_{0|0:0} = \boldsymbol{\mu}_{0|0:-1} + K_{\mathrm{KF}}(\boldsymbol{d}_0 - H\boldsymbol{\mu}_{0|0:-1}), \tag{32}$$

$$\Sigma_{0|0:0} = (I - K_{\mathrm{KF}}H)\Sigma_{0|0:-1}, \tag{33}$$

where $\boldsymbol{\mu}_{0|0:-1} = \boldsymbol{\mu}_0$ and $\Sigma_{0|0:-1} = \Sigma_0$. That is, (25) holds for $t = 0$. In order to prove (26) for $t = 0$, we apply Theorem 2. Since we assume that (1) and (7) hold, we have from Theorem 2 that

$$\boldsymbol{x}_1|\boldsymbol{d}_0 \sim N(\boldsymbol{\mu}_{1|0:0}, \Sigma_{1|0:0}), \tag{34}$$

where

$$\boldsymbol{\mu}_{1|0:0} = A_0\boldsymbol{\mu}_{0|0:0}, \tag{35}$$

$$\Sigma_{1|0:0} = A_0\Sigma_{0|0:0}A_0^\intercal + \Sigma_x. \tag{36}$$

That is, (26) holds for $t = 0$, which means that the theorem holds for $t = 0$.

We now want to prove that Theorem 3 holds for time-step t if we assume that it holds for $t-1$. We do this step-wise, by first proving that (25) holds, and then proving that (26) consequently also holds. By assuming that the theorem holds for time-step $t-1$, we have from (26) that

$$\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1} \sim N(\boldsymbol{\mu}_{t|0:t-1}, \Sigma_{t|0:t-1}). \tag{37}$$

Since we assume that (3) and (9) hold, we have from Theorem 1 that

$$\boldsymbol{x}_t|\boldsymbol{d}_{0:t} \sim N(\boldsymbol{\mu}_{t|0:t}, \Sigma_{t|0:t}), \tag{38}$$

where

$$\boldsymbol{\mu}_{t|0:t} = \boldsymbol{\mu}_{t|0:t-1} + K_{\mathrm{KF}}(\boldsymbol{d}_t - H\boldsymbol{\mu}_{t|0:t-1}), \tag{39}$$

$$\Sigma_{t|0:t} = (I - K_{\mathrm{KF}}H)\Sigma_{t|0:t-1}. \tag{40}$$

This means that (25) holds.

Now that an expression for $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$ is accessible, we are able to prove that (26) also holds. Since we assume that (1) and (7) hold, we are able to apply Theorem 2 to prove (26). From Theorem 2, we have that

$$\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t} \sim N(\boldsymbol{\mu}_{t+1|0:t}, \Sigma_{t+1|0:t}), \tag{41}$$

where

$$\boldsymbol{\mu}_{t+1|0:t} = A_t\boldsymbol{\mu}_{t|0:t}, \tag{42}$$

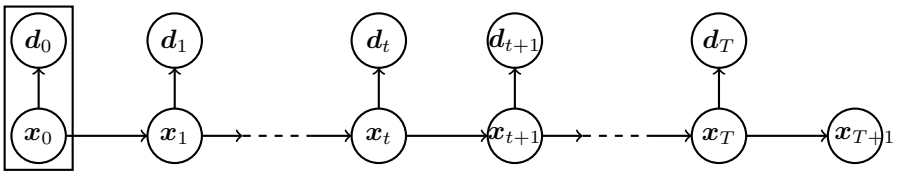
$$\Sigma_{t+1|0:t} = A_t\Sigma_{t|0:t}A_t^\intercal + \Sigma_x, \tag{43}$$

which means that (26) holds. This completes the proof of Theorem 3. □

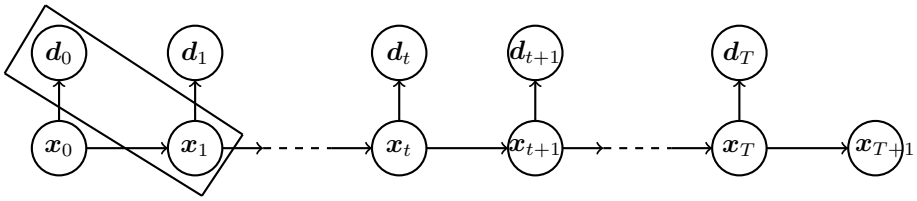
In Theorem 3, we see that both the filtering and forecasting distributions are Gaussian, which means that assessing the distributions only consists of computing the expressions for the mean vector and covariance matrix. In addition, we see that the parameters of the filtering distribution $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$, given in (27) and (28), are functions of $\boldsymbol{\mu}_{t|0:t-1}$ and $\Sigma_{t|0:t-1}$ only. Hence, $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$ can be assessed through the forecasting distribution at time-step $t-1$, $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})$. Similarly, we see that the expressions for the distribution parameters of the forecasting distribution $f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t})$, given in (29) and (30), are functions of $\boldsymbol{\mu}_{t|0:t}$ and $\Sigma_{t|0:t}$ only. That is, the forecasting distribution at time-step t, $f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t})$, can be assessed through the filtering distribution at time-step t, $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$. Thus, it is possible to formulate the Kalman filter as a recursive algorithm.

The algorithm for the Kalman filter alternates between an *update-step* and a *forecast-step*, which are both performed once for each time-step $t = 0, \ldots, T$. In the update-step at time-step t, we assess the filter distribution $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$, given in (25). In practice this entails computing (27) and (28). In the forecast-step at time-step t, we assess the forecasting distribution $f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t})$, given in (26), through the distribution acquired in the update-step, $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$. This is done by computing (29) and (30).

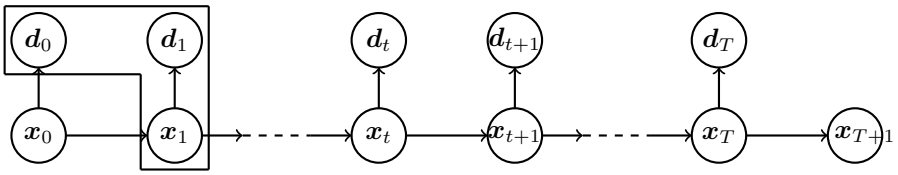
The three first steps of the Kalman filter, in addition to the last step, is illustrated in Figure 2. Figure 2(a) illustrates the update-step in the first iteration, i.e. for $t = 0$, where the distribution $f(\boldsymbol{x}_0|\boldsymbol{d}_0)$ is assessed. The forecast-step for $t = 0$ is visualized in Figure 2(b), where $f(\boldsymbol{x}_1|\boldsymbol{d}_0)$ is computed. Figure 2(c) illustrates the update-step for $t = 1$, where $f(\boldsymbol{x}_1|\boldsymbol{d}_{0:1})$ is assessed. Similarly, Figure 2(d) visualizes the forecast-step in the last iteration, i.e. for $t = T$, where we compute $f(\boldsymbol{x}_{T+1}|\boldsymbol{d}_{0:T})$. Algorithm 1 presents the Kalman filter as an algorithm. In the following section, we consider an approximation of the Kalman filter, when the linear Gaussian state space model is invalid.



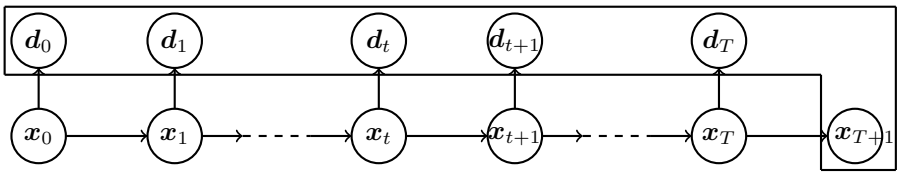
(a) The update-step in the first iteration of the Kalman filter. We condition $\boldsymbol{x}_0$ on the first observation $\boldsymbol{d}_0$.



(b) The forecast-step in the first iteration of the Kalman filter algorithm. We forecast $\boldsymbol{x}_1$, given the observation $\boldsymbol{d}_0$.



(c) The update-step in the second iteration of the Kalman filter. We condition $\boldsymbol{x}_1$ on $\boldsymbol{d}_1$.



(d) The forecast-step in the last iteration of the Kalman filter. We forecast $\boldsymbol{x}_{T+1}$, given the observations $\boldsymbol{d}_0, \ldots, \boldsymbol{d}_T$.

Figure 2: The three first plots visualize the three first steps of the Kalman filter algorithm. The last plot visualizes the last step. The squares indicate which observations that each latent variable is conditioned on.

Initiate $\boldsymbol{\mu}_0$ and Σ_0 from (6)
for $t = 0, \ldots, T$ **do**
  Update:
  Calculate $\boldsymbol{\mu}_{t|0:t}$ using (27)
  Calculate $\Sigma_{t|0:t}$ using (28)
  Forecast:
  Calculate $\boldsymbol{\mu}_{t+1|0:t}$ using (29)
  Calculate $\Sigma_{t+1|0:t}$ using (30)
end

Algorithm 1: Algorithm for the Kalman Filter. The pdf for the initial state, $f(\boldsymbol{x}_0)$, is assumed known. The subscripts on $\boldsymbol{\mu}$ and Σ indicate which latent variable that is considered and which observations that are conditioned on.

2.4 Ensemble Kalman filter

This section presents the ensemble Kalman filter, which is an approximation of the Kalman filter, presented in 2.3. We first explain the background for the development of the ensemble Kalman filter, and proceed with a presentation of the filter. Lastly, we present the filter in the form of an algorithm. For more information about the ensemble Kalman filter, see Evensen (1994) and Omre and Myrseth (2011).

Section 2.3 presents the Kalman filter, and explains how the filter solves the filtering and one-step forecasting problems. In the proof of Theorem 3, we notice that all of the properties in the state space model and linear Gaussian state space model are necessary in the Kalman filter. In the following, we do not assume that the forward model, (2), is Gauss-linear, as in the linear Gaussian state space model. However, we assume that the remaining properties of the model are valid, i.e. (1), (3), (4), (6) and (9) hold. Thus, the Kalman filter has to be approximated. A possible solution is the extended Kalman filter, which replaces the nonlinear forward function with a linear approximation. However, extended Kalman filter is unfeasible for high-dimensional systems, which is unfortunate as we mainly focus on high-dimensional systems in this report. Another option is the ensemble Kalman filter, which also approximates the Kalman filter, and is well-suited for high-dimensional systems. The ensemble Kalman filter represents the filtering and forecasting distributions through a set of realizations, rather than through a mean vector and covariance matrix. In the following, we give a detailed description of the ensemble Kalman filter.

The ensemble Kalman filter and the Kalman filter are structurally similar. Both filters assess $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$ and $f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t})$ for $t = 0, \ldots, T$ through a recursive algorithm, alternating between an update-step and a forecast-step. However, since EnKF represents the distributions through a set of realizations, the update and forward-steps in the two filters differ. In the following, we introduce some necessary notation before we consider the update- and forward-steps of the ensemble Kalman filter.

Recall that the Kalman filter assesses the forecasting distribution $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})$

by computing its mean vector $\boldsymbol{\mu}_{t|0:t-1}$ and $\Sigma_{t|0:t-1}$, in (29) and (30), respectively. The ensemble Kalman filter provides a Gaussian approximation of $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})$ through a set of approximately independent realizations, which are thought to represent the distribution. Each realization is called an *ensemble member*, and we denote the jth ensemble member by $\boldsymbol{\chi}_{t,j} \in \mathbb{R}^K$. For the remainder of this report, we let the index j denote the ensemble member. The number of ensemble members is called the *ensemble size* and we denote this by J. The ensemble members are generated independently from the pdf of the latent variable in the initial time-step, i.e. $\boldsymbol{\chi}_{0,j} \sim f(\boldsymbol{x}_0)$ for $j = 1, \ldots, J$, and then adjusted according to the likelihood and forward models, in order to represent the correct distribution at all times. The set of ensemble members is called an *ensemble*, and the we denote the ensemble representing $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})$ as χ_t. The ensemble χ_t is a matrix of consisting of J column vectors, where $\boldsymbol{\chi}_{t,j}$ is the jth column vector of χ_t, i.e. $\chi_t \in \mathbb{R}^{K\times J}$. The ensemble is represented as a matrix, rather than a set of ensemble members. Representing the ensemble as a matrix enables us to extensively simplify the expressions that are presented in Section 3. In the following, we first consider the update-step of the ensemble Kalman filter.

As for the Kalman filter, the objective of the update-step is to assess the filtering distribution $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$. EnKF assesses the filtering distribution by first approximating the forecasting distribution $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})$ through the ensemble. The approximation of $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})$ is a Gaussian pdf $N(\boldsymbol{\mu}_{t|0:t-1}, \Sigma_{t|0:t-1})$, where $\boldsymbol{\mu}_{t|0:t-1}$ and $\Sigma_{t|0:t-1}$ are unknown parameters. These parameters are approximated through the ensemble χ_t. The mean $\boldsymbol{\mu}_{t|0:t-1}$ is estimated by the average of the ensemble members

$$\hat{\boldsymbol{\mu}}_{t|0:t-1} = \frac{1}{J}\sum_{j=1}^{J} \boldsymbol{\chi}_{t,j}. \tag{44}$$

Similarly, we use the sample covariance of the ensemble to estimate $\Sigma_{t|0:t-1}$

$$\hat{\Sigma}_{t|0:t-1} = \frac{1}{J-1}\sum_{j=1}^{J} (\boldsymbol{\chi}_{t,j} - \hat{\boldsymbol{\mu}}_{t|0:t-1})(\boldsymbol{\chi}_{t,j} - \hat{\boldsymbol{\mu}}_{t|0:t-1})^\intercal. \tag{45}$$

We denote the Gaussian pdf $N(\hat{\boldsymbol{\mu}}_{t|0:t-1}, \hat{\Sigma}_{t|0:t-1})$ as $\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}, \chi_t)$, and it is used as the Gaussian approximation of $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})$. Note that the Gaussian approximation is conditioned on χ_t, because $\hat{\boldsymbol{\mu}}_{t|0:t-1}$ and $\hat{\Sigma}_{t|0:t-1}$ are calculated through the ensemble.

Now that a Gaussian approximation of $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})$ is available, we are able to approximate $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$. We approximate $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$ by a Gaussian approximation denoted $\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}, \chi_t)$. As in Section 2.3, we apply Bayes' rule to derive the expressions for the distribution parameters

$$\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}, \chi_t) = \frac{\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}, \chi_t) f(\boldsymbol{d}_t|\boldsymbol{x}_t, \boldsymbol{d}_{0:t-1})}{f(\boldsymbol{d}_t|\boldsymbol{d}_{0:t-1})}. \tag{46}$$

Due to conditional independence between observations, (3), we have that $f(\boldsymbol{d}_t|\boldsymbol{x}_t, \boldsymbol{d}_{0:t-1}) = f(\boldsymbol{d}_t|\boldsymbol{x}_t)$. We have that the prior $\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}, \chi_t)$ is Gaussian

with parameters defined in (44) and (45), and that $f(\boldsymbol{d}_t|\boldsymbol{x}_t)$ is defined in (9). By inserting these pdfs into (46), we have that the posterior $\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}, \chi_t)$ is a Gaussian pdf $N(\hat{\boldsymbol{\mu}}_{t|0:t}, \hat{\Sigma}_{t|0:t})$. The full derivation of the distribution is presented in Appendix C, and the expressions for the parameters are only presented here

$$\hat{\boldsymbol{\mu}}_{t|0:t} = \hat{\boldsymbol{\mu}}_{t|0:t-1} + K_{\text{EnKF}}(\boldsymbol{d}_t - H\hat{\boldsymbol{\mu}}_{t|0:t-1}) \tag{47}$$

$$\hat{\Sigma}_{t|0:t} = (I - K_{\text{EnKF}}H)\hat{\Sigma}_{t|0:t-1}, \tag{48}$$

where $K_{\text{EnKF}} = \hat{\Sigma}_{t|0:t-1}H^\intercal(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)^{-1}$ is the Kalman gain matrix for EnKF. Note that the expressions are identical to the expressions in (39) and (40), except that $\boldsymbol{\mu}_{t|0:t-1}$ and $\Sigma_{t|0:t-1}$ are replaced with $\hat{\boldsymbol{\mu}}_{t|0:t-1}$ and $\hat{\Sigma}_{t|0:t-1}$. This completes the assessment of $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$.

In order to ensure that the ensemble represents the filter distribution $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$, the ensemble members need to be adjusted to the observation in the current time-step, $\boldsymbol{d}_t$. By the term "adjusted", we mean that if the ensemble is a set of realizations from $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})$, the adjusted ensemble is a set of realizations from $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$. If the ensemble is not adjusted to $\boldsymbol{d}_t$, the ensemble Kalman filter fails to approximate the Kalman filter. The following expression adjusts the existing ensemble members on the current observation, $\boldsymbol{d}_t$

$$\tilde{\boldsymbol{\chi}}_{t,j} = \boldsymbol{\chi}_{t,j} + K_{\text{EnKF}}(\boldsymbol{d}_t + \boldsymbol{u}_{t,j} - H\boldsymbol{\chi}_{t,j}), \tag{49}$$

where $\boldsymbol{u}_{t,j} \sim N(\boldsymbol{0}, \Sigma_d)$ and where the tilde indicates that the jth ensemble member $\tilde{\boldsymbol{\chi}}_{t,j}$ is adjusted according to $\boldsymbol{d}_{0:t}$. Similarly, $\tilde{\chi}_t$ denotes the ensemble that represents $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$. If χ_t is a set of realizations from $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})$, (49) guarantees that $\tilde{\chi}_t$ is a set of realizations from $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$.

An alternative approach to adjusting the existing ensemble members to $\boldsymbol{d}_t$ in (49), is to sample new ensemble members from $\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}, \chi_t)$. This enables us to provide independent ensemble members that are adjusted according to $\boldsymbol{d}_t$. However, because of the uncertainty regarding the validity of the Gaussian approximation, we want to avoid sampling from $\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}, \chi_t)$ at all costs. That is, the Gaussian approximation $\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}, \chi_t)$ is utilized to approximate the filter distribution only. This completes the update-step of the ensemble Kalman filter. In the following we consider the forecast-step.

The forecast-step in the ensemble Kalman filter resembles the forecast-step in the traditional Kalman filter. Recall that the objective of the forecast-step at time t in the Kalman filter is to provide an expression for the forecasting distribution $f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t})$. In practice, this entails we compute the distribution parameters of $f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t})$ by (29) and (30). However, since the forward model is assumed to be nonlinear, these expressions are invalid in EnKF. Since EnKF represents $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$ through an ensemble, EnKF applies the formulation of the forward model from (2) on each ensemble member individually, in order to obtain an ensemble that represents $f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t})$. That is,

$$\boldsymbol{\chi}_{t+1,j} = w_t(\tilde{\boldsymbol{\chi}}_{t,j}, \boldsymbol{v}_{t,j}), \quad j = 1, \ldots, J, \tag{50}$$

where $\boldsymbol{v}_{t,j} \sim N(\boldsymbol{0}, \Sigma_x)$. This completes the forecast-step in EnKF. The ensemble Kalman filter is summarised in Algorithm 2.

Initiate $\chi_0 = \{\boldsymbol{\chi}_{0,j} \sim f(\boldsymbol{x}_0), j = 1, \ldots, J\}$
for $t = 0, \ldots, T$ **do**
  Update:
  Calculate $\hat{\boldsymbol{\mu}}_{t|0:t-1}$ using (44)
  Calculate $\hat{\Sigma}_{t|0:t-1}$ using (45)
  Calculate $\hat{\boldsymbol{\mu}}_{t|0:t}$ using (47)
  Calculate $\hat{\Sigma}_{t|0:t}$ using (48)
  Calculate $\tilde{\boldsymbol{\chi}}_{t,j}$ using (49) for $j = 1, \ldots, J$
  Forecast:
  Calculate $\boldsymbol{\chi}_{t+1,j}$ using (50) for $j = 1, \ldots, J$
end

Algorithm 2: Algorithm for the ensemble Kalman Filter. The pdf for the initial state, $f(\boldsymbol{x}_0)$, is assumed known.

2.5 Hierarchical ensemble Kalman filter

The hierarchical ensemble Kalman filter (HEnKF), introduced in Omre and Myrseth (2010), is an extension of the ensemble Kalman filter, presented in Section 2.4. This section explains the background for the development of HEnKF, and provides a presentation of the filter. We assume the same model as in Section 2.4, that is, a state-space model where (1), (3), (4), (6) and (9) apply.

The ensemble Kalman filter (EnKF), presented in Section 2.4 has proven to be a powerful tool for making estimates on a latent variable given a set of observations. However, EnKF fails to account for the estimation uncertainty in $\hat{\Sigma}_{t|0:t-1}$, which is the estimate of $\Sigma_{t|0:t-1}$. In addition, $\hat{\Sigma}_{t|0:t-1}$ fails to contain the same amount of information about the distribution as $\Sigma_{t|0:t-1}$. This is because of rank deficiency, which occurs when the ensemble size J is smaller than both K and D, which we recall are the dimensions of the state space variable $\boldsymbol{x}_t$ and the observation $\boldsymbol{d}_t$, respectively. As a result of this, Omre and Myrseth (2010) suggests enforcing a hierarchical model on $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})$ in the update-step, which entails imposing prior distributions for $\boldsymbol{\mu}_{t|0:t-1}$ and $\Sigma_{t|0:t-1}$. This enables us to account for the estimation uncertainty in $\Sigma_{t|0:t-1}$. By imposing a suitable prior distribution for $\Sigma_{t|0:t-1}$, we are able to remove the issue of rank deficiency completely. The variables $\boldsymbol{\mu}_{t|0:t-1}$ and $\Sigma_{t|0:t-1}$ are mentioned extensively in this chapter, which is why we simplify the notation by denoting these variables as $\boldsymbol{\mu}_t$ and Σ_t, respectively, for the remainder of this chapter. In the following, we present a detailed description of the hierarchical ensemble Kalman filter.

The structure of the hierarchical ensemble Kalman filter resembles the structure of ensemble Kalman filter. Both filters alternates between an update-step and a forecast-step, and represent the distributions through an ensemble. However, since the hierarchical ensemble Kalman filter enforces a hierarchical model on the state variables, adjustments has to be made for the update-step. It also entails that HEnKF has to assign prior distributions for $\boldsymbol{\mu}_t$ and Σ_t as a part of the initialization. In the following, we first consider the update-step.

The update-step in HEnKF consists of deriving the distribution $f(\boldsymbol{\mu}_t, \Sigma_t|\chi_t)$, and then adjusting the ensemble members to the current observation $\boldsymbol{d}_t$, as in EnKF. First, we consider the derivation of $f(\boldsymbol{\mu}_t, \Sigma_t|\chi_t)$. An expression for $f(\boldsymbol{\mu}_t, \Sigma_t|\chi_t)$ can be found by using Bayes' rule and the prior distribution $f(\boldsymbol{\mu}_t, \Sigma_t)$

$$f(\boldsymbol{\mu}_t, \Sigma_t|\chi_t) = \frac{f(\boldsymbol{\mu}_t, \Sigma_t)\hat{f}(\chi_t|\boldsymbol{\mu}_t, \Sigma_t)}{f(\chi_t)}. \tag{51}$$

HEnKF chooses the prior distribution $f(\boldsymbol{\mu}_t, \Sigma_t)$ such that deriving the expression for $f(\boldsymbol{\mu}_t, \Sigma_t|\chi_t)$ is analytically tractable. This is possible by choosing a prior distribution such that $f(\boldsymbol{\mu}_t, \Sigma_t)$ is a conjugate prior to the likelihood $\hat{f}(\chi_t|\boldsymbol{\mu}_t, \Sigma_t)$, i.e. the prior $f(\boldsymbol{\mu}_t, \Sigma_t)$ and the posterior $f(\boldsymbol{\mu}_t, \Sigma_t|\chi_t)$ belong to the same family of distributions (Gamerman and Lopes, 2006, p. 50). In order to choose a suitable prior distribution, we first need to assess the likelihood $\hat{f}(\chi_t|\boldsymbol{\mu}_t, \Sigma_t)$.

The ensemble members are considered to be independent, thus we have that

$$\hat{f}(\chi_t|\boldsymbol{\mu}_t, \Sigma_t) = \prod_{j=1}^{J} \hat{f}(\boldsymbol{\chi}_{t,j}|\boldsymbol{\mu}_t, \Sigma_t). \tag{52}$$

Each ensemble member $\boldsymbol{\chi}_{t,j}$ is considered to be realizations from the pdf of $\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}$. Since $\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}, \boldsymbol{\mu}_t, \Sigma_t \sim N(\boldsymbol{\mu}_t, \Sigma_t)$, we have that $\hat{f}(\chi_t|\boldsymbol{\mu}_t, \Sigma_t)$ is Gaussian. That is, the joint prior distribution $f(\boldsymbol{\mu}_t, \Sigma_t)$ should belong to the Gaussian conjugate family, see Gelman et al. (2003). Omre and Myrseth (2010) chooses the prior to belong to the normal-inverse-Wishart distribution. In the following, we present the expressions for the prior and posterior distributions $f(\boldsymbol{\mu}_t, \Sigma_t|\chi_t)$.

We write the prior distribution as $f(\boldsymbol{\mu}_t, \Sigma_t) = f(\boldsymbol{\mu}_t|\Sigma_t)f(\Sigma_t)$. The following distributions are assigned to $\boldsymbol{\mu}_t|\Sigma_t$ and Σ_t

$$\boldsymbol{\mu}_t|\Sigma_t \sim N(\boldsymbol{\xi}_t, \alpha_t\Sigma_t), \tag{53}$$

$$\Sigma_t \sim \text{IW}(\Psi_t, \nu_t), \tag{54}$$

where $\alpha_t > 0$ is the variance scaling parameter, and $\text{IW}(\Psi_t, \nu_t)$ is the inverse-Wishart distribution with matrix parameter $\Psi_t \in \mathbb{R}^{K\times K}$ and ν_t degrees of freedom. The expression for the posterior distributions are only presented here (Omre and Myrseth, 2010). The posterior distributions are as follows

$$\boldsymbol{\mu}_t|\Sigma_t, \chi_t \sim N(\boldsymbol{\xi}_t^*, \alpha_t^*\Sigma_t) \tag{55}$$

$$\Sigma_t|\chi_t \sim \text{IW}(\Psi_t^*, \nu_t^*) \tag{56}$$

where

$$\boldsymbol{\xi}_t^* = \frac{1}{1+J\alpha_t}\boldsymbol{\xi}_t + \frac{J\alpha_t}{1+J\alpha_t}\hat{\boldsymbol{\mu}}_{t|0:t-1}, \tag{57}$$

$$\alpha_t^* = \frac{\alpha_t}{1+J\alpha_t}, \tag{58}$$

$$\Psi_t^* = \Psi_t + (J-1)\hat{\Sigma}_{t|0:t-1} + \frac{J}{1+J\alpha_t}(\hat{\boldsymbol{\mu}}_{t|0:t-1} - \boldsymbol{\xi}_t)(\hat{\boldsymbol{\mu}}_{t|0:t-1} - \boldsymbol{\xi}_t)^\intercal, \tag{59}$$

$$\nu_t^* = \nu_t + J, \tag{60}$$

where $\hat{\boldsymbol{\mu}}_{t|0:t-1}$ and $\hat{\Sigma}_{t|0:t-1}$ are defined in (47) and (48), respectively.

Note that the parameters are conditioned on χ_t, which means that the distribution parameters are adjusted according to $\boldsymbol{d}_{0:t-1}$. That is, the distribution parameters will be adjusted according to $\boldsymbol{d}_t$ in the the following time-step, when $\boldsymbol{\mu}_{t+1}$ and Σ_{t+1} are adjusted to χ_{t+1}. Also note that the parameters in the posterior distributions are weighted averages of the prior parameters and ensemble members. This means that the prior is assigned much weight when the ensemble size J is small, and that the prior is assigned less weight when J is large. When $J \to \infty$, the prior is assigned no weight, and HEnKF coincides with EnKF.

Note that HEnKF adjusts $\boldsymbol{\mu}_t$ and Σ_t to χ_t by deriving a distribution $f(\boldsymbol{\mu}_t, \Sigma_t|\chi_t)$, as an alternative to the updating in EnKF, which is to compute (47) and (48). The HEnKF-approach allows us to combine our prior assumptions about $\boldsymbol{\mu}_t$ and Σ_t with knowledge about the observations $\boldsymbol{d}_{0:t-1}$ through the ensemble χ_t, as we can see in (57) to (60). By imposing a sensible prior distribution for Σ_t, HEnKF is also able to guarantee that Σ_t remains full-rank. Also, choosing a prior $f(\boldsymbol{\mu}_t, \Sigma_t)$ from the Gaussian conjugate family reduces the amount of computations that has to be performed, since computing the normalizing constant for $f(\boldsymbol{\mu}_t, \Sigma_t|\chi_t)$ is avoided, which in many cases is difficult to compute.

It should also be noted that the normal-inverse-Wishart distribution is considered to be the standard choice of conjugate prior distribution when the likelihood is multivariate normal, which is arguably the reason why Omre and Myrseth (2010) chose this distribution. Later, we shall see that the choice of this distribution give rise to some issues regarding computability. This completes the derivation of $f(\boldsymbol{\mu}_t, \Sigma_t|\chi_t)$. In the following, we consider the adjustment of the ensemble members to the current observation, $\boldsymbol{d}_t$.

The adjusting of the ensemble members resembles the adjusting in EnKF. Recall that in EnKF, each ensemble member is adjusted based on $\hat{\Sigma}_{t|0:t-1}$, see (49). However, since HEnKF considers Σ_t as a stochastic variable, HEnKF generates a sample of Σ_t from $f(\Sigma_t|\chi_t)$, in order to perform the updating of the ensemble members. That is, HEnKF adjusts each ensemble member on a different sample of Σ_t. The generated sample of Σ_t that is applied to updating ensemble member number j is denoted $\Sigma_{t,j}$, i.e. $\Sigma_{t,j} \sim f(\Sigma_t|\chi_t)$. The updating is as follows

$$\tilde{\boldsymbol{\chi}}_{t,j} = \boldsymbol{\chi}_{t,j} + \Sigma_{t,j} H^{\intercal} (H \Sigma_{t,j} H^{\intercal} + \Sigma_d)^{-1} (\boldsymbol{d}_t + \boldsymbol{u}_{t,j} - H \boldsymbol{\chi}_{t,j}), \tag{61}$$

where $\boldsymbol{u}_{t,j} \sim N(\boldsymbol{0}, \Sigma_d)$ and $\Sigma_{t,j} \sim \text{IW}(\Psi_t^*, \nu_t^*)$. As in (49), by computing the ensemble $\tilde{\chi}_t$ through (61), the ensemble $\tilde{\chi}_t$ is a set of realizations from $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$ if χ_t is a set of realizations from $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})$. This completes the update-step of HEnKF, and we proceed to the forecast-step.

The forecast-step in HEnKF resembles the forecast-step in EnKF. However, in addition to forwarding the ensemble members, the distribution parameters for $\boldsymbol{\mu}_t$ and Σ_t has to be forwarded as well. For more details about the updating of $\boldsymbol{\mu}_t$ and Σ_t, see Omre and Myrseth (2010). An important detail in the forwarding

of the distribution parameters, is that the forecast-step preserves the distribution families of $\boldsymbol{\mu}_t$ and Σ_t. This is important because it means that the distribution families of both $\boldsymbol{\mu}_t$ and Σ_t are preserved throughout the entire filter algorithm, since both the update- and forward-steps preserve the distribution families. This reduces the computational demands of the filter extensively, as it reduces the updating and forecasting of $\boldsymbol{\mu}_t$ and Σ_t to computing their parameter values, and reassessing the distribution families of $\boldsymbol{\mu}_t$ and Σ_t in each time-step is omitted. The forwarding of the ensemble members is identical to the forwarding in EnKF, see (50). This completes the presentation of the hierarchical ensemble Kalman filter.

2.6 Gaussian Markov random fields

This section presents Gaussian Markov random fields (GMRF), which is an important part of the new theory presented in Section 3. For more information about Gaussian Markov random fields, see Rue and Held (2005).

Before we present GMRF, we introduce the notation that is necessary to present GMRF in a digestible manner. Consider a stochastic variable $\boldsymbol{x}$, which is a vector of length K, i.e. $\boldsymbol{x} \in \mathbb{R}^K$. We assume that $\boldsymbol{x}$ is normally distributed with mean vector $\boldsymbol{\mu}$ and covariance matrix Σ, i.e. $\boldsymbol{x} \sim N(\boldsymbol{\mu}, \Sigma)$. We denote the kth element of $\boldsymbol{x}$ as x^k, i.e. $\boldsymbol{x} = (x^1, \dots, x^k, \dots, x^K)$. In order to denote all elements of $\boldsymbol{x}$ between the ith and kth element, we write $\boldsymbol{x}^{\{i+1:k-1\}}$. If we want to denote all elements of $\boldsymbol{x}$, except the ith and kth element, we write $\boldsymbol{x}^{-\{i,k\}}$. That is, if we want to denote that the ith and kth element of $\boldsymbol{x}$ are conditionally independent given the remaining elements, we write

$$x^i \perp x^k | \boldsymbol{x}^{-\{i,k\}}. \tag{62}$$

We let Q denote the precision matrix of $\boldsymbol{x}$, which is the inverse of the covariance matrix, i.e. $Q = \Sigma^{-1}$. Further, we denote the (i,k)th entry of Q with a superscript, i.e. $Q^{i,k}$.

In the following, we define the *graph* associated to $\boldsymbol{x}$. The graph associated to $\boldsymbol{x}$ consists of K nodes, where node k is associated to the kth element of $\boldsymbol{x}$. The graph consists of a set of edges, where each edge connects two of the nodes in the graph. The following definition determines whether there is an edge between two nodes or not. There exists no edge between node i and k if and only if

$$x^i \perp x^k | \boldsymbol{x}^{-\{i,k\}}. \tag{63}$$

That is, if x^i and x^k are conditionally independent given the remaining elements of $\boldsymbol{x}$, then there is no edge between nodes i and k, and vice versa.

Further, we define the term *sequential neighbour*. Each node k has a set of sequential neighbours, which we denote Λ_k. We define Λ_k as the smallest subset of $\{1, \dots, k-1\}$ such that $f(x^k | \boldsymbol{x}^{\{1:k-1\}}) = f(x^k | \boldsymbol{x}^{\Lambda_k})$ holds, where $\boldsymbol{x}^{\Lambda_k}$ is the vector containing the stochastic variables associated with the nodes in Λ_k. From the definition of Λ_k, we have that

$$x^k \perp \boldsymbol{x}^{\{1:k-1\} \setminus \Lambda_k} | \boldsymbol{x}^{\Lambda_k}. \tag{64}$$

That is, x^k is conditionally independent of $\boldsymbol{x}^{\{1:k-1\}\setminus\Lambda_k}$ given $\boldsymbol{x}^{\Lambda_k}$. To denote the number of sequential neighbours of node k, we write $|\Lambda_k|$. The set Λ_k is ordered in ascending order, which enables us to enumerate the nodes in Λ_k from 1 to $|\Lambda_k|$. We denote the lth sequential neighbour of node k as $\Lambda_k(l)$, and we denote the stochastic variable associated to the lth sequential neighbour of node k as $x^{\Lambda_k(l)}$. If node i is a sequential neighbour of node k, i.e. $i \in \Lambda_k$, then $\Lambda_k^{-1}(i)$ denotes the index of node i in Λ_k. That is, if node i is the lth sequential neighbour of node k, i.e. $\Lambda_k(l) = i$, then $\Lambda_k^{-1}(i) = l$. This means that $\Lambda_k(\Lambda_k^{-1}(i)) = i$.

We present a theorem that provides a useful connection between the sequential neighbourhood and conditional independence.

Theorem 4. *Assume $1 \leq i < k \leq K$. We then have*

$$i \notin \Lambda_k \text{ and } \nexists s \in \{k+1, \ldots, K\} : i, k \in \Lambda_s \iff x^i \perp x^k | \boldsymbol{x}^{-\{i,k\}}. \tag{65}$$

The proof is presented in Appendix E. The theorem states that if either $i \in \Lambda_k$ or there exists a sequential neighbourhood Λ_s such that $i, k \in \Lambda_s$, then x^i and x^k are not conditionally independent given the remaining elements of $\boldsymbol{x}$. That is, there are two possibilities as to why there is an edge between node i and k. Either node i is a sequential neighbour of node k, i.e. $i \in \Lambda_k$, or both nodes are sequential neighbours of the same node, $i, k \in \Lambda_s$ for some $s \in \{k+1, K\}$. In the following, we discuss a theorem that connects the presented theory to the sparsity of the precision matrix Q.

From Theorem 2.2 in Rue and Held (2005), we have that $Q^{i,k} = 0$ if and only if the ith and kth element of $\boldsymbol{x}$ are conditionally independent given the remaining elements,

$$x^i \perp x^k | \boldsymbol{x}^{-\{i,k\}} \iff Q^{i,k} = 0. \tag{66}$$

Hence we have a useful connection between the graph associated to $\boldsymbol{x}$, and the structure of the precision matrix Q. If there is no edge between node i and k, then (63) holds. This implies that $Q^{i,k} = 0$. Conversely, if $Q^{i,k} = 0$, then there is no edge between node i and k. This enables us to assess the sparsity of Q through the graph. In the following, we apply this theorem on an example.

Assume that $\boldsymbol{x}$ follows a second order Markov chain,

$$x^k \perp \boldsymbol{x}^{\{1,\ldots,k-3\}} | \boldsymbol{x}^{\{k-2,k-1\}}. \tag{67}$$

That is, the kth element of $\boldsymbol{x}$ is conditionally independent of the $k-3$ first elements, given the two elements before x^k. We can also represent the conditional dependencies in $\boldsymbol{x}$ by a graph. Since $\boldsymbol{x}$ follows a second order Markov chain, the sequential neighbourhood for node k is

$$\Lambda_k = \{k-2, k-1\} \quad \text{for } k > 2. \tag{68}$$

That is $|\Lambda_k| = 2$ if $k > 2$, and $|\Lambda_k| = k-1$ if $k \leq 2$. Thus, there exist edges between node k and the two previous nodes. Since node k is a sequential neighbour of node $k+1$ and $k+2$, there are also edges between node k and the

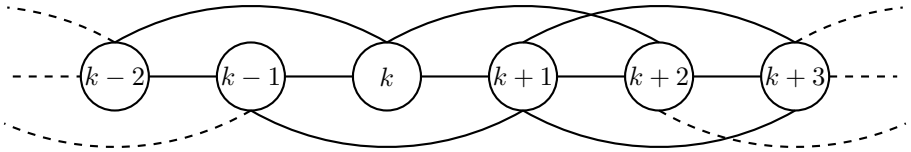


Figure 3: An excerpt of the graph for $\boldsymbol{x}$. There is an edge between each node and its four closest nodes.

two following nodes. Figure 3 shows a part of the graph associated to $\boldsymbol{x}$. We see that there is only an edge between a node and its four closest nodes. In the following, we show how the Markov property on $\boldsymbol{x}$ influences the structure of the precision matrix Q in this example.

By combining (66) and (68), we have that

$$Q^{i,k} = 0 \quad \text{for } |i-k| > 2. \tag{69}$$

That is, Q has the following structure

$$Q = \begin{pmatrix} \bullet & \bullet & \bullet & & & & & & & & & \\ \bullet & \bullet & \bullet & \bullet & & & & & & & & \\ \bullet & \bullet & \bullet & \bullet & \bullet & & & & & & & \\ & \bullet & \bullet & \bullet & \bullet & \bullet & & & & & & \\ & & \bullet & \bullet & \bullet & \bullet & \bullet & & & & & \\ & & & \bullet & \bullet & \bullet & \bullet & \bullet & & & & \\ & & & & \bullet & \bullet & \bullet & \bullet & \bullet & & & \\ & & & & & \ddots & & & & & & \\ & & & & & & \bullet & \bullet & \bullet & \bullet & \bullet & \\ & & & & & & & \bullet & \bullet & \bullet & \bullet & \bullet \\ & & & & & & & & \bullet & \bullet & \bullet & \bullet \\ & & & & & & & & & \bullet & \bullet & \bullet \end{pmatrix}, \tag{70}$$

where $\bullet$ represents the nonzero entries of Q. That is, only the elements on the diagonal, and the two first lower and upper diagonals are nonzero, which means that in total 5 diagonals are nonzero. The *bandwidth* of a matrix Q is defined as

$$\max\{|i-k| : Q^{i,k} \neq 0\}. \tag{71}$$

In our example, Q is a band matrix with bandwidth two. In general, if $\boldsymbol{x}$ follows an mth order Markov chain, Q has $2m+1$ nonzero-diagonals and bandwidth m. This result is very useful, as band matrices have many beneficial properties regarding computational complexity (Rue and Held, 2005, chap. 1.2.1), which is utilized in the next section.

2.7 Applying Cholesky decomposition on band matrices

This section discusses how we in certain cases are able to apply Cholesky decomposition to efficiently compute expressions involving band matrices. This is a crucial part of the new theory presented in Section 3. For more information, see Rue and Held (2005).

Let $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$ be vectors of length K, i.e. $\boldsymbol{z}_1, \boldsymbol{z}_2 \in \mathbb{R}^K$. We assume that $\boldsymbol{z}_1$ is unknown and that $\boldsymbol{z}_2$ is known. Further, assume that Q is a known symmetric positive-definite matrix of size $K \times K$. Our objective is to compute

$$\boldsymbol{z}_1 = Q^{-1}\boldsymbol{z}_2. \tag{72}$$

as efficiently as possible. It is obviously possible to compute the right hand side directly, that is, by first inverting Q, and then compute $Q^{-1}\boldsymbol{z}_2$. Another option is to use Cholesky decomposition. In the following, we first describe how Cholesky decomposition is applied to compute (72), when Q is a full matrix. Further, we discuss how Cholesky decomposition enables us to reduce the computational complexity of computing (72) when we assume that Q is a band matrix.

We first utilize that Q is a symmetric positive-definite matrix. For every symmetric and positive-definite matrix, there exists a lower triangular matrix L such that $Q = LL^\intercal$. By combining this expression with (72), we have

$$\boldsymbol{z}_1 = L^{-\intercal}L^{-1}\boldsymbol{z}_2. \tag{73}$$

We now want to split up the right hand side into two equations. We define a vector $\boldsymbol{z}_3 \in \mathbb{R}^K$, such that

$$\boldsymbol{z}_3 = L^{-1}\boldsymbol{z}_2, \tag{74}$$

$$\boldsymbol{z}_1 = L^{-\intercal}\boldsymbol{z}_3. \tag{75}$$

Note that inserting the expression for $\boldsymbol{z}_3$ in (74) into (75) yields (73). In order to compute $\boldsymbol{z}_1$, we first compute $\boldsymbol{z}_3$ from (74), and then compute $\boldsymbol{z}_1$ from (75). In the following, we first consider the computation of (74).

We rewrite (74) as

$$\boldsymbol{z}_2 = L\boldsymbol{z}_3. \tag{76}$$

Since L is a lower triangular matrix, L has the following structure

$$L = \begin{pmatrix} \bullet & & & & & & & \\ \bullet & \bullet & & & & & & \\ \bullet & \bullet & \bullet & & & & & \\ \bullet & \bullet & \bullet & \bullet & & & & \\ & & & & \ddots & & & \\ \bullet & \bullet & \dots & \dots & \bullet & \bullet & & \\ \bullet & \bullet & \dots & \dots & \bullet & \bullet & \bullet & \\ \bullet & \bullet & \dots & \dots & \bullet & \bullet & \bullet & \bullet \end{pmatrix}, \tag{77}$$

where $\bullet$ denotes the nonzero entries of L. In order to utilize the structure of L, we compute (76) row-by-row. That is, we compute $\boldsymbol{z}_2$ recursively, by first

solving the first row of the matrix equation $\boldsymbol{z}_2 = L\boldsymbol{z}_3$, and then repeat this for every row down to the last row. In the following, we discuss this in detail.

Since L is lower diagonal, L only has one nonzero entry on the first row, see (77). Thus, the equation for the first row of $\boldsymbol{z}_2 = L\boldsymbol{z}_3$ becomes

$$z_2^1 = L^{1,1} z_3^1, \tag{78}$$

where z_2^1 and z_3^1 are the first elements of $\boldsymbol{z}_2$ and $\boldsymbol{z}_3$, respectively. Since $\boldsymbol{z}_2$ is known, we have that $z_3^1 = L^{1,1}/z_2^1$. Now that z_3^1 is known, we are able to solve the second row of $\boldsymbol{z}_2 = L\boldsymbol{z}_3$ efficiently. The Cholesky decomposition L has at most two nonzero entries on the second row, as seen in (77). Thus, the equation for the second row becomes

$$z_2^2 = L^{2,1} z_3^1 + L^{2,2} z_3^2. \tag{79}$$

Since z_3^1 is known, we are able to solve the equation for z_3^2

$$z_3^2 = \frac{1}{L^{2,2}}(z_2^2 - L^{2,1} z_3^1). \tag{80}$$

Similarly, we are able to compute z_3^3, z_3^4, etc, recursively. In general, the expression for the lth element of $\boldsymbol{z}_3$ becomes

$$z_3^l = \frac{1}{L^{l,l}}(z_2^l - \sum_{s=1}^{l-1} L^{l,s} z_3^s). \tag{81}$$

Now that $\boldsymbol{z}_3$ is computed, we are able to compute $\boldsymbol{z}_1$ from (75). We apply the same logic to solve this equation as for solving (74). By rewriting (75), we have

$$\boldsymbol{z}_3 = L^\intercal \boldsymbol{z}_1. \tag{82}$$

As for (76), we compute (82) row-by-row. However, since $L^\intercal$ is upper diagonal, we iterate in the opposite direction than for (76). That is, we first compute the last row of $\boldsymbol{z}_3 = L^\intercal \boldsymbol{z}_1$, and iterate backwards up to the first row of $\boldsymbol{z}_1$. The expression for the lth element of $\boldsymbol{z}_1$ is as follows

$$z_1^l = \frac{1}{L^{l,l}}(z_3^l - \sum_{s=l+1}^{K} L^{s,l} z_1^s). \tag{83}$$

In the following we compare the computational complexity of computing (72) brute-force, to the presented Cholesky decomposition-approach.

In general, we have that inverting Q has a computational complexity of $\mathcal{O}(K^3)$. The computational complexity of multiplying Q^{-1} with $\boldsymbol{z}_2$ is $\mathcal{O}(K^3)$, which means that the overall complexity of computing (72) brute-force is $\mathcal{O}(K^3)$. The computational complexity of Cholesky decomposing Q is $\mathcal{O}(K^3)$. By computing $\boldsymbol{z}_3$ through (81), the computational complexity is $\mathcal{O}(K^2)$, because computing each of the K elements of $\boldsymbol{z}_3$ has a computational complexity of $\mathcal{O}(K)$. Similarly, the computational complexity of computing $\boldsymbol{z}_1$ from (83) is $\mathcal{O}(K^2)$. Thus, the

overall complexity of computing (72) with Cholesky decomposition is $\mathcal{O}(K^3)$. Hence Cholesky decomposition is of little use when Q is assumed to be a full matrix. In the following, we make further assumptions about the sparsity of Q, and discuss how this affects the computational complexity of computing (72).

We now assume that Q is a band matrix with *bandwidth* m, where the term bandwidth is defined in (71). The computational complexity can be reduced by utilizing the fact that Q is a band matrix with known bandwidth. Appendix D shows that L only has nonzero entries on the diagonal and on the m first lower diagonals. For example, if the bandwidth of Q is two, i.e. $m = 2$, L has the following structure

$$L = \begin{pmatrix} \bullet & & & & & & & \\ \bullet & \bullet & & & & & & \\ \bullet & \bullet & \bullet & & & & & \\ & \bullet & \bullet & \bullet & & & & \\ & & & & \ddots & & & \\ & & & \bullet & \bullet & \bullet & & \\ & & & & \bullet & \bullet & \bullet & \\ & & & & & \bullet & \bullet & \bullet \end{pmatrix}, \tag{84}$$

where $\bullet$ indicate the nonzero entries of L. In the following, we show how we are able to compute (72) more efficiently, when Q has bandwidth m.

As previously, we compute (72) by solving (74) and (75). Recall that (74) is solved by computing $\boldsymbol{z}_3$ elementwise through (81). However, because of the structure of L, we are able to simplify this expression. Since Q has bandwidth m, L has at most $m + 1$ nonzero entries on each row. The nonzero entries are the diagonal entry, and the first m entries located to the left of the diagonal. For example, we can see from (84) that each row has at most 3 nonzero entries when $m = 2$. Thus, (81) can be simplified to

$$z_3^l = \frac{1}{L^{l,l}}(z_2^l - \sum_{s=l-m}^{l-1} L^{l,s} z_3^s), \tag{85}$$

where we notice that the sum goes from $l - m$ to $l - 1$, while the sum in (81) goes from 1 to $l - 1$.

Similarly, we are able to compute (75) more efficiently as well. Each row of $L^\intercal$ has at most $m + 1$ nonzero entries, and these elements are located on the diagonal and to the right of the diagonal. Hence, (83) can be simplified to

$$z_1^l = \frac{1}{L^{l,l}}(z_3^l - \sum_{s=l+1}^{l+m} L^{s,l} z_1^s), \tag{86}$$

where we notice that the sum goes from $l + 1$ to $l + m$, while the sum in (83) goes from $l + 1$ to K. Algorithm 3 presents a summary of computing (72) with Cholesky decomposition when Q is a band matrix with bandwidth

Compute Cholesky decomposition, $Q = LL^\intercal$.
Solve $\boldsymbol{z}_2 = L\boldsymbol{z}_3$ for $\boldsymbol{z}_3$:
for $l = 1, \ldots, K$ **do**
  Compute z_3^l using (85)
end
Solve $\boldsymbol{z}_3 = L^\intercal \boldsymbol{z}_1$ for $\boldsymbol{z}_1$:
for $l = K, \ldots, 1$ **do**
  Compute z_1^l using (86)
end

Algorithm 3: Algorithm for computing $\boldsymbol{z}_1 = Q^{-1}\boldsymbol{z}_2$ with Cholesky decomposition when Q has bandwidth m.

m. In the following, we discuss the computational complexity of the Cholesky decomposition-approach for band matrices.

As mentioned, the computational complexity of Cholesky decomposing Q is $\mathcal{O}(K^3)$. However, since Q is a band matrix with bandwidth m, we have that L has at most $m+1$ entries on each row. Thus, the number of nonzero entries in L is $\mathcal{O}(Km)$. Computing each entry has a computational complexity of $\mathcal{O}(m)$. Thus, Cholesky decomposing Q has a computational complexity of $\mathcal{O}(Km^2)$. In addition, we are able to reduce the computational complexity of computing (81) and (83), by rather computing (85) and (86). The computational complexity of computing (85) is $\mathcal{O}(Km)$ for each element of $\boldsymbol{z}_3$, which is the same as for computing (86). Thus, the overall computational complexity of utilizing Cholesky decomposition to compute (72) is $\mathcal{O}(Km^2)$. As mentioned, the brute-force approach has a computational complexity of $\mathcal{O}(K^3)$. In our numerical examples, we have that $K \gg m$. Thus, Cholesky decomposition extensively reduces the computational complexity.

3 Alternative prior distribution for HEnKF

In this section, we propose a new family of prior distributions for the hierarchical ensemble Kalman filter, followed by a presentation of the corresponding posterior distributions and the precision matrix for the latent variable $\boldsymbol{x}_t$. Lastly, we show how the proposed prior distributions can be applied to reduce the computational costs of the hierarchical ensemble Kalman filter.

3.1 New prior distribution for HEnKF

Recall that the ensemble Kalman filter, presented in Section 2.4, fails to account for the estimation uncertainty in the estimation of the covariance matrix. The hierarchical ensemble Kalman filter, presented in Section 2.5, solves this by imposing a prior distribution on the model parameters. However, this increases the computational cost of the update step, which is apparent by comparing (49) to (61). In the following, we aim to reduce the computational demands of HEnKF,

while still obtaining reliable results. That is, we impose a prior distribution on the model parameters that enables us to reduce the computational cost of computing (61). We believe that this prior distribution is able to provide reliable results by inheriting some of the beneficial properties from the hierarchical model, such as accounting for estimation uncertainty. In the following, we apply (66) and Theorem 4 from Section 2.6 to construct a new prior distribution.

Recall that we in Section 2.5 assume that $\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}$ is normally distributed with mean vector $\boldsymbol{\mu}_t$ and covariance matrix Σ_t, where the distributions for $\boldsymbol{\mu}_t|\Sigma_t$ and Σ_t are defined in (53) and (54), respectively. In the following, we assume that $\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}$ is a Gaussian Markov random field, as presented in Section 2.6. That is, we assume that there is a graph associated to $\boldsymbol{x}_t$, where the kth element of $\boldsymbol{x}_t$ is associated to node k in the graph. We assume that node k is assigned a sequential neighbourhood that satisfies (64), which we denote Λ_k. Further, we assume that Λ_k is time-invariant, which is why Λ_k is denoted without the subscript t. We denote the stochastic variables related to the set of sequential neighbours of node k as $\boldsymbol{x}_t^{\Lambda_k}$, i.e. $\boldsymbol{x}_t^{\Lambda_k} \in \mathbb{R}^{|\Lambda_k|}$. In the following, we introduce the model parameters for $\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}$.

We denote the model parameters of $\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}$ as $\boldsymbol{\eta}_t$ and $\boldsymbol{\phi}_t$. We assume that $\boldsymbol{\phi}_t$ is a vector of length K, i.e. $\boldsymbol{\phi}_t \in \mathbb{R}^K$, and we denote the kth element of $\boldsymbol{\phi}_t$ as ϕ_t^k. Further, we assume that $\boldsymbol{\eta}_t$ is a collection of K vectors, and we denote the kth vector by $\boldsymbol{\eta}_t^k$, i.e. $\boldsymbol{\eta}_t = (\boldsymbol{\eta}_t^1, \dots, \boldsymbol{\eta}_t^k, \dots, \boldsymbol{\eta}_t^K)$. We have that $\boldsymbol{\eta}_t^k$ is a vector of length $|\Lambda_k| + 1$, i.e. $\boldsymbol{\eta}_t^k \in \mathbb{R}^{|\Lambda_k|+1}$, where we recall that $|\Lambda_k|$ is the number of sequential neighbours for node k. We denote the lth element in $\boldsymbol{\eta}_t^k$ as $\eta_t^{k,l}$. Since $\boldsymbol{\eta}_t^k$ is of length $|\Lambda_k| + 1$, the collection of vectors in $\boldsymbol{\eta}_t$ are generally not of the same size. Hence, it is not reasonable to represent $\boldsymbol{\eta}_t$ as a matrix, but rather as a collection of vectors, which is why we denote $\boldsymbol{\eta}_t$ in bold. Before we introduce the prior distributions for $\boldsymbol{\phi}_t$ and $\boldsymbol{\eta}_t|\boldsymbol{\phi}_t$, we present the distribution for $\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}, \boldsymbol{\eta}_t, \boldsymbol{\phi}_t$.

We have that

$$f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}, \boldsymbol{\eta}_t, \boldsymbol{\phi}_t) = \prod_{k=1}^{K} f(x_t^k|\boldsymbol{x}_t^{\{1:k-1\}}, \boldsymbol{d}_{0:t-1}, \boldsymbol{\eta}_t, \boldsymbol{\phi}_t). \tag{87}$$

Since we assume that $\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}, \boldsymbol{\eta}_t, \boldsymbol{\phi}_t$ is a GMRF, we apply (64) to simplify the expression above

$$f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}, \boldsymbol{\eta}_t, \boldsymbol{\phi}_t) = \prod_{k=1}^{K} f(x_t^k|\boldsymbol{x}_t^{\Lambda_k}, \boldsymbol{d}_{0:t-1}, \boldsymbol{\eta}_t, \boldsymbol{\phi}_t). \tag{88}$$

We also assume that

$$f(x_t^k|\boldsymbol{x}_t^{\Lambda_k}, \boldsymbol{d}_{0:t-1}, \boldsymbol{\eta}_t, \boldsymbol{\phi}_t) = f(x_t^k|\boldsymbol{x}_t^{\Lambda_k}, \boldsymbol{d}_{0:t-1}, \boldsymbol{\eta}_t^k, \phi_t^k). \tag{89}$$

Further, we assume that

$$x_t^k|\boldsymbol{x}_t^{\Lambda_k}, \boldsymbol{d}_{0:t-1}, \boldsymbol{\eta}_t^k, \phi_t^k \sim N\left(\eta_t^{k,1} + \sum_{l=1}^{|\Lambda_k|} x_t^{\Lambda_k(l)} \eta_t^{k,l+1}, \phi_t^k\right), \tag{90}$$

where $\eta_t^{k,l+1}$ is element $l+1$ in the vector η_t^k. By writing the mean as a dot product of two vectors, we have

$$x_t^k | \boldsymbol{x}_t^{\Lambda_k}, \boldsymbol{d}_{0:t-1}, \boldsymbol{\eta}_t^k, \phi_t^k \sim N\left(\left(1, \left(\boldsymbol{x}_t^{\Lambda_k}\right)^\intercal\right) \cdot \boldsymbol{\eta}_t^k, \phi_t^k\right). \tag{91}$$

This completes the presentation of the pdf of $\boldsymbol{x}_t | \boldsymbol{d}_{0:t-1}, \boldsymbol{\eta}_t, \boldsymbol{\phi}_t$.

Recall that we impose prior distributions on the model parameters in Section 2.5. Similarly, we now want to impose prior distributions on $\boldsymbol{\phi}_t$ and $\boldsymbol{\eta}_t | \boldsymbol{\phi}_t$. We first present the prior distribution for $\boldsymbol{\phi}_t$. We assume that the elements of $\boldsymbol{\phi}_t$ are a priori independent

$$f(\boldsymbol{\phi}_t) = \prod_{k=1}^{K} f(\phi_t^k). \tag{92}$$

We assign the following distribution to each element ϕ_t^k

$$\phi_t^k \sim \text{InvGam}(\alpha_t^k, \beta_t^k). \tag{93}$$

That is, ϕ_t^k is inverse-gamma distributed with parameters α_t^k and β_t^k. In the prior distribution for $\boldsymbol{\eta}_t | \boldsymbol{\phi}_t$, we assume that

$$f(\boldsymbol{\eta}_t | \boldsymbol{\phi}_t) = \prod_{k=1}^{K} f(\boldsymbol{\eta}_t^k | \phi_t^k) \tag{94}$$

and

$$\boldsymbol{\eta}_t^k | \phi_t^k \sim N(\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k}, \phi_t^k \Sigma_{\boldsymbol{\eta}_t^k}), \tag{95}$$

for some $\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k} \in \mathbb{R}^{|\Lambda_k|+1}$, and $\Sigma_{\boldsymbol{\eta}_t^k} \in \mathbb{R}^{(|\Lambda_k|+1)\times(|\Lambda_k|+1)}$. This completes the presentation of the prior distribution. Note that the prior $f(\boldsymbol{\eta}_t, \boldsymbol{\phi}_t)$ is normal-inverse-gamma distributed, which is in the Gaussian conjugate family. This means that the distribution $f(\boldsymbol{\eta}_t, \boldsymbol{\phi}_t)$ is a conjugate prior of the likelihood $f(\boldsymbol{x}_t | \boldsymbol{d}_{0:t-1}, \boldsymbol{\eta}_t, \boldsymbol{\phi}_t)$, which again means that $f(\boldsymbol{\eta}_t, \boldsymbol{\phi}_t | \boldsymbol{x}_t)$ is normal-inverse-gamma distributed as well.

3.2 New posterior distribution

In Section 3.1 we present a family of alternative prior distributions for the hierarchical ensemble Kalman filter. In the following, we present the corresponding posterior distributions $f(\boldsymbol{\phi}_t | \chi_t)$ and $f(\boldsymbol{\eta}_t | \boldsymbol{\phi}_t, \chi_t)$. Note the distributions are only presented here, while the full derivation of the posterior distributions are presented in Appendix F. We assume the same structure as in Section 3.1, that is, we assume a sequential neighbourhood between the elements of the latent variable $\boldsymbol{x}_t$. Recall that in the update-step in HEnKF, we assume that each ensemble member $\boldsymbol{\chi}_{t,j}$ is distributed according to $f(\boldsymbol{x}_t | \boldsymbol{d}_{0:t-1}, \boldsymbol{\eta}_t, \boldsymbol{\phi}_t)$. Thus

$$f(\boldsymbol{\chi}_{t,j} | \boldsymbol{\eta}_t, \phi_t) = \prod_{k=1}^{K} f(\chi_{t,j}^k | \boldsymbol{\chi}_{t,j}^{\Lambda_k}, \boldsymbol{d}_{0:t-1}, \boldsymbol{\eta}_t^k, \phi_t^k), \tag{96}$$

where $\boldsymbol{\chi}_{t,j}^{\Lambda_k}$ is the vector containing the values of $\boldsymbol{\chi}_{t,j}$ related to the sequential neighbours of the kth node, i.e. $\boldsymbol{\chi}_{t,j}^{\Lambda_k} \in \mathbb{R}^{|\Lambda_k|}$. We also assume that the ensemble members are independent. Thus, the distribution for the entire ensemble χ_t is

$$f(\chi_t|\boldsymbol{\eta}_t, \boldsymbol{\phi}_t) = \prod_{j=1}^{J} f(\boldsymbol{\chi}_{t,j}|\boldsymbol{\eta}_t, \boldsymbol{\phi}_t) = \prod_{j=1}^{J}\prod_{k=1}^{K} f(\chi_{t,j}^k|\boldsymbol{\chi}_{t,j}^{\Lambda_k}, \boldsymbol{d}_{0:t-1}, \boldsymbol{\eta}_t^k.\phi_t^k), \tag{97}$$

By (91), we have that

$$\chi_{t,j}^k|\boldsymbol{\chi}_{t,j}^{\Lambda_k}, \boldsymbol{d}_{0:t-1}, \boldsymbol{\eta}_t^k, \phi_t^k \sim N\left(\left(1, \left(\boldsymbol{\chi}_{t,j}^{\Lambda_k}\right)^\intercal\right) \cdot \boldsymbol{\eta}_t^k, \phi_t^k\right). \tag{98}$$

As mentioned, the full derivation of the posterior distributions are presented in Appendix F. However, a brief summary of the derivation is presented here. In order to find the distribution for $\boldsymbol{\phi}_t|\chi_t$, we apply Bayes' rule

$$f(\boldsymbol{\phi}_t|\chi_t) = \frac{f(\boldsymbol{\phi}_t)f(\chi_t|\boldsymbol{\phi}_t)}{f(\chi_t)} \propto f(\boldsymbol{\phi}_t)f(\chi_t|\boldsymbol{\phi}_t). \tag{99}$$

Further, we have that

$$f(\boldsymbol{\phi}_t)f(\chi_t|\boldsymbol{\phi}_t) = \int f(\boldsymbol{\phi}_t)f(\chi_t|\boldsymbol{\eta}_t, \boldsymbol{\phi}_t)f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t)d\boldsymbol{\eta}_t. \tag{100}$$

The first factor in the integrand is given by (92) and (93), the second factor is given by (97) and (98), while the third factor is defined in (94) and (95). By inserting these expressions into (100), we get that $f(\boldsymbol{\phi}_t|\chi_t)$ has the shape of a product of inverse-gamma distributions. More specifically, we have that

$$f(\boldsymbol{\phi}_t|\chi_t) = \prod_{k=1}^{K} f(\phi_t^k|\chi_t), \tag{101}$$

where

$$f(\phi_t^k|\chi_t) = \text{InvGam}(\tilde{\alpha}_t^k, \tilde{\beta}_t^k), \tag{102}$$

$$\tilde{\alpha}_t^k = \alpha_t^k + \frac{J}{2} \tag{103}$$

$$\tilde{\beta}_t^k = \left(\frac{1}{\beta_t^k} + \frac{1}{2}\left(\gamma_t^k - (\boldsymbol{\rho}_t^k)^\intercal \left(\Theta_t^k\right)^{-1} \boldsymbol{\rho}_t^k\right)\right)^{-1}. \tag{104}$$

where γ_t^k, $\boldsymbol{\rho}_t^k$ and Θ_t^k are defined as

$$\gamma_t^k = \boldsymbol{\mu}_{\eta_t^k}^\intercal \left(\Sigma_{\eta_t^k}\right)^{-1} \boldsymbol{\mu}_{\eta_t^k} + \left(\boldsymbol{\chi}_t^k\right)^\intercal \cdot \boldsymbol{\chi}_t^k, \tag{105}$$

$$\boldsymbol{\rho}_t^k = \left(\Sigma_{\eta_t^k}\right)^{-1} \boldsymbol{\mu}_{\eta_t^k} + \left(\mathbf{1}_J, \left(\chi_t^{\Lambda_k}\right)^\intercal\right)^\intercal \boldsymbol{\chi}_t^k \tag{106}$$

$$\Theta_t^k = \left(\Sigma_{\eta_t^k}\right)^{-1} + \left(\mathbf{1}_J, \left(\chi_t^{\Lambda_k}\right)^\intercal\right)^\intercal \cdot \left(\mathbf{1}_J, \left(\chi_t^{\Lambda_k}\right)^\intercal\right), \tag{107}$$

where $\mathbf{1}_J \in \mathbb{R}^{J\times 1}$ is a column-vector of length J consisting of ones. The vector $\boldsymbol{\chi}_t^k$ consists of the kth element of every ensemble member. That is, $\boldsymbol{\chi}_t^k$ is the kth row-vector of χ_t, i.e. $\boldsymbol{\chi}_t^k \in \mathbb{R}^J$. The matrix $\chi_t^{\Lambda_k}$ consists of $\boldsymbol{\chi}_{t,j}^{\Lambda_k}$ for $j = 1, \dots J$. That is, $\boldsymbol{\chi}_{t,j}^{\Lambda_k}$ is the jth column vector of $\chi_t^{\Lambda_k}$, i.e. $\chi_t^{\Lambda_k} \in \mathbb{R}^{|\Lambda_k|\times J}$. Note that the last term in Θ_t^k is an outer product. From (101), we have that elements of $\boldsymbol{\phi}_t$ are a posteriori independent. This is advantageous, as it enables us to sample the elements of $\boldsymbol{\phi}_t|\chi_t$ separately. In the following, we proceed to the derivation of the distribution for $\boldsymbol{\eta}_t|\boldsymbol{\phi}_t, \chi_t$.

The full derivation of the distribution for $\boldsymbol{\eta}_t|\boldsymbol{\phi}_t, \chi_t$ is presented in Appendix F. However, a summary of the derivation is presented here. The distribution for $\boldsymbol{\eta}_t|\boldsymbol{\phi}_t, \chi_t$ is derived by applying Bayes' rule

$$f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t, \chi_t) = \frac{f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t) f(\chi_t|\boldsymbol{\phi}_t, \boldsymbol{\eta}_t)}{f(\chi_t|\boldsymbol{\phi}_t)} \propto f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t) f(\chi_t|\boldsymbol{\phi}_t, \boldsymbol{\eta}_t), \tag{108}$$

where the first factor is defined by (94) and (95), while the second factor is defined by (97) and (98). By inserting these expressions, we get that $f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t, \chi_t)$ has the shape of a product of normal distributions. More specifically, we have

$$f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t, \chi_t) = \prod_{k=1}^{K} f(\boldsymbol{\eta}_t^k|\phi_t^k, \chi_t), \tag{109}$$

where

$$\boldsymbol{\eta}_t^k|\phi_t^k, \chi_t \sim N\left(\left(\Theta_t^k\right)^{-1} \boldsymbol{\rho}_t^k, \phi_t^k \left(\Theta_t^k\right)^{-1}\right). \tag{110}$$

From (109), we see that the elements of $\boldsymbol{\eta}_t$ are a posteriori independent. This enables us to sample the elements of $\boldsymbol{\eta}_t|\chi_t$ separately, which is advantageous. The distribution for $\boldsymbol{\eta}_t, \boldsymbol{\phi}_t|\chi_t$ is then

$$f(\boldsymbol{\eta}_t, \boldsymbol{\phi}_t|\chi_t) = f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t, \chi_t) f(\boldsymbol{\phi}_t|\chi_t) = \prod_{k=1}^{K} f(\boldsymbol{\eta}_t^k|\phi_t^k, \chi_t) f(\phi_t^k|\chi_t). \tag{111}$$

3.3 Precision matrix

Recall that in HEnKF, presented in Section 2.5, we compute (61) in order to adjust the ensemble χ_t to the current observation. In order to compute (61), we sample $\Sigma_{t,j}$ from $f(\Sigma_t|\chi_t)$ for each ensemble member $\boldsymbol{\chi}_{t,j}$, where we recall that Σ_t is the covariance matrix of $\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}$. Section 3.1 presents alternative prior distributions for the model parameters of $\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}$. Consequently, $\Sigma_{t,j}$ must be sampled from the posterior distribution corresponding to the new prior distribution. This posterior distribution is presented in Section 3.2. In the following, we show how the inverse of $\Sigma_{t,j}$ can be computed based on samples from the posterior distributions. More precisely, we present expressions connecting

the precision matrix $Q_{t,j} = \Sigma_{t,j}^{-1}$, to samples from $f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t, \chi_t)$ and $f(\boldsymbol{\phi}_t|\chi_t)$. We assume the same structure as in Sections 3.1 and 3.2. That is, we assume that $\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}$ is a GMRF and that there exists a sequential neighbourhood for each element of $\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}$. We only present the expressions for the entries of Q_t here, while the full derivation is presented in Appendix G.

Before we present the expression for the precision matrix, we need to define a set $\tilde{\Lambda}_k$. We define $\tilde{\Lambda}_k$ as the set of nodes that have node k as their sequential neighbour. That is, if node k is a sequential neighbour of $l > k$, i.e. $k \in \Lambda_l$, then $l \in \tilde{\Lambda}_k$. Hence Λ_k is a subset of the subsequent nodes, i.e. $\tilde{\Lambda}_k \subseteq \{k+1, \ldots, K\}$. In the following, we present the expressions for the nonzero entries of the precision matrix $Q_{t,j}$. We first consider the diagonal entries. The (k,k)th entry of $Q_{t,j}$ is defined as

$$Q_{t,j}^{k,k} = \frac{1}{\phi_{t,j}^k} + \sum_{l \in \tilde{\Lambda}_k} \frac{(\eta_{t,j}^{l,\Lambda_l^{-1}(k)+1})^2}{\phi_{t,j}^l}, \tag{112}$$

where $\boldsymbol{\phi}_{t,j}$ and $\boldsymbol{\eta}_{t,j}$ are samples from $f(\boldsymbol{\phi}_t|\chi_t)$ and $f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t, \chi_t)$, respectively, as defined in (101) and (109), and that $\eta_{t,j}^{l,\Lambda_l^{-1}(k)+1}$ is element number $\Lambda_l^{-1}(k)+1$ in the vector $\boldsymbol{\eta}_{t,j}^l$. We now proceed to the off-diagonal entries of $Q_{t,j}$.

In the following, we assume $k < l$. By combining Theorem 4 with (66), we are able to decide which off-diagonal entries of $Q_{t,j}$ that are zero. We have that

$$k \notin \Lambda_l \text{ and } \nexists s \in \{k+1, \ldots, K\} : l, k \in \Lambda_s \iff Q_{t,j}^{l,k} = 0 \text{ for } k < l. \tag{113}$$

That is, there are two possibilities as to why an off-diagonal element $Q_{t,j}^{l,k}$ is nonzero. Either node $k \in \Lambda_l$, or there exists a node s such that $k, l \in \Lambda_s$. It is also possible that both are true simultaneously. The set $\tilde{\Lambda}_k \cap \tilde{\Lambda}_l$ is the set of all nodes that have both node k and l as their sequential neighbours. That is, if $l, k \in \Lambda_s$, then $s \in \tilde{\Lambda}_k \cap \tilde{\Lambda}_l$. In the following, we present the expression for an off-diagonal entry $Q_{t,j}^{l,k}$ for three different cases. The first case is when $k \in \Lambda_l$ and $\tilde{\Lambda}_k \cap \tilde{\Lambda}_l = \emptyset$. The second case is when $k \notin \Lambda_l$ and $\tilde{\Lambda}_k \cap \tilde{\Lambda}_l \neq \emptyset$. The third case is when $k \in \Lambda_l$ and $\tilde{\Lambda}_k \cap \tilde{\Lambda}_l \neq \emptyset$. We consider the first case first.

If $k \in \Lambda_l$ and node k and l are not sequential neighbours of the same nodes, i.e. $\tilde{\Lambda}_k \cap \tilde{\Lambda}_l = \emptyset$, then

$$Q_{t,j}^{l,k} = Q_{t,j}^{k,l} = -\frac{\eta_{t,j}^{l,\Lambda_l^{-1}(k)+1}}{\phi_{t,j}^l}. \tag{114}$$

In the second case, that is when $k \notin \Lambda_l$, and $\tilde{\Lambda}_k \cap \tilde{\Lambda}_l \neq \emptyset$, we have

$$Q_{t,j}^{l,k} = Q_{t,j}^{k,l} = \sum_{s \in \tilde{\Lambda}_k \cap \tilde{\Lambda}_l} \frac{\eta_{t,j}^{s,\Lambda_s^{-1}(l)+1} \eta_{t,j}^{s,\Lambda_s^{-1}(k)+1}}{\phi_{t,j}^s}. \tag{115}$$

In the third case, when $k \in \Lambda_l$ and $\tilde{\Lambda}_k \cap \tilde{\Lambda}_l \neq \emptyset$, the expression for $Q_{t,j}^{l,k}$ is

the sum of the two expressions above

$$Q_{t,j}^{l,k} = Q_{t,j}^{k,l} = -\frac{\eta_{t,j}^{l,\Lambda_l^{-1}(k)+1}}{\phi_{t,j}^{l}} + \sum_{s \in \tilde{\Lambda}_k \cap \tilde{\Lambda}_l} \frac{\eta_{t,j}^{s,\Lambda_s^{-1}(l)+1} \eta_{t,j}^{s,\Lambda_s^{-1}(k)+1}}{\phi_{t,j}^{s}}. \tag{116}$$

From (116) we see that sparsity of the precision matrix $Q_{t,j}$ is determined by Λ_k for $k = 1, \ldots, K$. If we choose $\Lambda_k = \emptyset$ for $k = 1, \ldots, K$, the precision matrix is diagonal. On the other extreme, if we choose $\Lambda_k = \{1, \ldots, k-1\}$ for $k = 1, \ldots, K$, the precision matrix is full. Thus, by adjusting the set of sequential neighbours for each node, we are able to modify the precision matrix as we choose. This completes the presentation of the precision matrix. In the following, we show how we can utilize the sparsity of the precision matrix to reduce the computational cost of the hierarchical ensemble Kalman filter.

3.4 Reduction in computational cost

In Section 3.3, we present expressions connecting the entries of the matrix $Q_{t,j}$ to a sample from $f(\boldsymbol{\eta}_t, \boldsymbol{\phi}_t | \chi_t)$, where we recall that $Q_{t,j} = \Sigma_{t,j}^{-1}$ is a sampled precision matrix of $\boldsymbol{x}_t | \boldsymbol{d}_{0:t-1}$ and where $\boldsymbol{\eta}_t$ and $\boldsymbol{\phi}_t$ are the model parameters of $\boldsymbol{x}_t | \boldsymbol{d}_{0:t-1}$. As mentioned at the end of Section 3.3, the sparsity of $Q_{t,j}$ is determined by the sequential neighbourhood imposed on the elements of $\boldsymbol{x}_t | \boldsymbol{d}_{0:t-1}$. In this section, we explain how we are able to reduce the computational cost of the hierarchical ensemble Kalman filter, from Section 2.5, when we assume that $Q_{t,j}$ is sparse. This is done by combining the theory from Sections 2.7 and 3.3. More specifically, we consider how the computational costs of computing (61) can be reduced by applying Cholesky decomposition. We assume the same structure as in Sections 3.1 to 3.3.

Our aim is to compute the last term in (61) as efficiently as possible. That is, computing

$$\Sigma_{t,j} H^\intercal (H \Sigma_{t,j} H^\intercal + \Sigma_d)^{-1} (\boldsymbol{d}_t + \boldsymbol{u}_{t,j} - H \boldsymbol{\chi}_{t,j}). \tag{117}$$

We denote this expression as $\Delta\chi_{t,j}$, i.e.

$$\Delta\chi_{t,j} = \Sigma_{t,j} H^\intercal (H \Sigma_{t,j} H^\intercal + \Sigma_d)^{-1} (\boldsymbol{d}_t + \boldsymbol{u}_{t,j} - H \boldsymbol{\chi}_{t,j}). \tag{118}$$

In the following, we present a procedure for computing $\Delta\chi_{t,j}$ that is more efficient than the brute-force approach, i.e. computing the inverse of $H\Sigma_{t,j}H^\intercal + \Sigma_d$ directly.

First, we use that $\Sigma_{t,j} = Q_{t,j}^{-1}$ and $\Sigma_d = Q_d^{-1}$

$$\Delta\chi_{t,j} = Q_{t,j}^{-1} H^\intercal (H Q_{t,j}^{-1} H^\intercal + Q_d^{-1})^{-1} (\boldsymbol{d}_t + \boldsymbol{u}_{t,j} - H \boldsymbol{\chi}_{t,j}). \tag{119}$$

We apply the Sherman-Morrison-Woodbury matrix identity (Woodbury, 1950) on $(HQ_{t,j}^{-1}H^\intercal + \Sigma_d)^{-1}$.

$$(H Q_{t,j}^{-1} H^\intercal + Q_d^{-1})^{-1} = Q_d - Q_d H (Q_{t,j} + H^\intercal Q_d H)^{-1} H^\intercal Q_d. \tag{120}$$

Thus we want to evaluate

$$\Delta\chi_{t,j} = Q_{t,j}^{-1}H^\intercal(Q_d - Q_dH(Q_{t,j} + H^\intercal Q_dH)^{-1}H^\intercal Q_d)(\boldsymbol{d}_t + \boldsymbol{u}_{t,j} - H\boldsymbol{\chi}_{t,j}). \quad (121)$$

We denote $\boldsymbol{r}_1 = \boldsymbol{d}_t + \boldsymbol{u}_{t,j} - H\boldsymbol{\chi}_{t,j}$

$$\Delta\chi_{t,j} = Q_{t,j}^{-1}H^\intercal(Q_d - Q_dH(Q_{t,j} + H^\intercal Q_dH)^{-1}H^\intercal Q_d)\boldsymbol{r}_1 \quad (122)$$

We split up the outer parentheses, and we obtain

$$\Delta\chi_{t,j} = Q_{t,j}^{-1}H^\intercal Q_d\boldsymbol{r}_1 - Q_{t,j}^{-1}H^\intercal Q_dH(Q_{t,j} + H^\intercal Q_dH)^{-1}H^\intercal Q_d\boldsymbol{r}_1. \quad (123)$$

Our aim is to apply Cholesky decomposition to reduce the computational cost of computing the expression above. In order to achieve this, we have to make additional assumptions about the sparsity of H and Q_d. That is, we assume that H and Q_d are structured such that $Q_{t,j} + H^\intercal Q_dH$ is a sparse matrix with known bandwidth. In the following, we first consider the first term in (123).

In order to compute $Q_{t,j}^{-1}H^\intercal Q_d\boldsymbol{r}_1$, we make use of the following fact. The computational complexity of computing AB, where $A \in \mathbb{R}^{u\times v}$ and $B \in \mathbb{R}^{v\times w}$ is $\mathcal{O}(uvw)$. Thus, in order to compute $Q_{t,j}^{-1}H^\intercal Q_d\boldsymbol{r}_1$ as efficiently as possible, we should always multiply the last two factors first, $Q_{t,j}^{-1}(H^\intercal(Q_d\boldsymbol{r}_1))$. In order to prevent inverting $Q_{t,j}$ directly, we make use of the theory presented in Section 2.7. That is, we define $\boldsymbol{r}_2 = HQ_d\boldsymbol{r}_1$ and $\boldsymbol{r}_3 = Q_{t,j}^{-1}HQ_d\boldsymbol{r}_1$, and apply Algorithm 3 to compute $\boldsymbol{r}_3 = Q_{t,j}^{-1}\boldsymbol{r}_2$.

In order to compute the second term of (123), we use the same strategy. We define the vector $\boldsymbol{r}_4$ as

$$\boldsymbol{r}_4 = Q_{t,j}^{-1}H^\intercal Q_dH(Q_{t,j} + H^\intercal Q_dH)^{-1}H^\intercal Q_d\boldsymbol{r}_1. \quad (124)$$

We first compute $\boldsymbol{r}_5 = (Q_{t,j} + H^\intercal Q_dH)^{-1}\boldsymbol{r}_2 = (Q_{t,j} + H^\intercal Q_dH)^{-1}H^\intercal Q_d\boldsymbol{r}_1$. In order to compute $\boldsymbol{r}_5$, we apply Algorithm 3. We then compute $\boldsymbol{r}_6 = H^\intercal Q_dH\boldsymbol{r}_5$. Lastly, we compute $\boldsymbol{r}_4 = Q_{t,j}^{-1}\boldsymbol{r}_6$ through Algorithm 3. Algorithm 4 presents a summary of computing $\Delta\chi_{t,j}$ using the presented procedure.

Compute $\boldsymbol{r}_1 = \boldsymbol{d}_t + \boldsymbol{u}_{t,j} - H\boldsymbol{\chi}_{t,j}$
Compute $\boldsymbol{r}_2 = H^\intercal Q_d\boldsymbol{r}_1$
Compute $\boldsymbol{r}_3 = Q_{t,j}^{-1}\boldsymbol{r}_2$ using Algorithm 3
Compute $\boldsymbol{r}_5 = (Q_{t,j} + H^\intercal Q_dH)^{-1}\boldsymbol{r}_2$ using Algorithm 3
Compute $\boldsymbol{r}_6 = H^\intercal Q_dH\boldsymbol{r}_5$
Compute $\boldsymbol{r}_4 = Q_{t,j}^{-1}\boldsymbol{r}_6$ using Algorithm 3
Return $\Delta\chi_{t,j} = \boldsymbol{r}_3 - \boldsymbol{r}_4$

Algorithm 4: Algorithm for computing (118) using Cholesky decomposition.

In Section 2.7 we argue that the computational complexity of Cholesky decomposing a $K \times K$-matrix with bandwidth m can be reduced from $\mathcal{O}(K^3)$ to $\mathcal{O}(Km^2)$. If we assume that H and Σ_d are diagonal matrices, we argue that the computational complexity of computing $\Delta\chi_{t,j}$ through Algorithm 4 also can be reduced to $\mathcal{O}(Km^2)$.

4 Implementation and experimental setup

In this section, we first give a brief explanation of how the hierarchical ensemble Kalman filter, from Section 2.5, is implemented with the prior distribution presented in Section 3.1. We then present two test cases where the results provided by our proposed prior distribution are compared against results provided by the prior distribution introduced in Omre and Myrseth (2010).

4.1 Implementation

In this section we explain how we implement the hierarchical ensemble Kalman filter, introduced in Omre and Myrseth (2010), with the prior distribution presented in Section 3.1. We first present some of the packages that are used in the implementation, and how they are utilized to reduce the computational cost. We then explain how we apply object-oriented programming to simplify the implementation of the hierarchical ensemble Kalman filter. Lastly, we explain how the observations in the numerical examples are generated.

We implement the hierarchical ensemble Kalman filter in `Python`. Several packages are used extensively in the implementation, in order to reduce the computational cost. The package `numpy` is used to represent matrices and vectors, in addition to sampling realizations from the normal distribution. We notice that sampling from the normal distribution is a major contributing factor to the overall computational cost, especially for high-dimensional systems. This stems from the way computers simulate realizations from the normal distribution. If $\boldsymbol{x} \sim N(\boldsymbol{\mu}, \Sigma)$, where $\boldsymbol{x}, \boldsymbol{\mu} \in \mathbb{R}^K$ and $\Sigma \in \mathbb{R}^{K \times K}$, we have that $\boldsymbol{x}$ is simulated by computing

$$\boldsymbol{x} = \boldsymbol{\mu} + B\boldsymbol{z}, \tag{125}$$

where $BB^\intercal = \Sigma$ and $\boldsymbol{z} \sim N(\boldsymbol{0}, I_K)$. There are several ways to compute B. In `numpy`, the default is computing B using singular value decomposition. However, by computing B using Cholesky decomposition, we are able to reduce the computational cost.

We use the package `scipy` for inverting sparse matrices and sampling from the inverse-gamma distribution. This package is used in combination with the package `sksparse.cholmod` to perform Cholesky decomposition on sparse matrices. With this package we are able to utilize that the structure of the matrix is known. Also, this package enables us to store the structure of the sparse matrices. Hence the structure of the sparse matrices only needs to be assessed once, which enables us to only compute the nonzero entries of the sparse matrices.

In order to implement a Gaussian Markov random field, as presented in Section 2.6, we construct a class `Neighbourhood`. Recall that in Section 2.6, we assign a sequential neighbourhood to each element of the state vector $\boldsymbol{x}_t$. Each instance of the class `Neighbourhood` represents a node $k \in \{1, \dots, K\}$ in the graph associated to $\boldsymbol{x}_t$, and stores the sequential neighbours of node k. Recall from Section 2.4 that we approximate the one-step forecasting distribution

$f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})$ by an ensemble χ_t. We implement a class `Ensemble`, in order to represent the ensemble as a matrix. That is, each ensemble member is represented as a vector, and the ensemble members are column vectors of the ensemble matrix. A subclass of `Ensemble` is also implemented, called `Ensemble_matrix_form`. This class inherits its attributes from `Ensemble`, and represents each ensemble member as a matrix, rather than a vector. This means that the ensemble is represented as a sequence of matrices, i.e. a *tensor*. Each instance of `Ensemble_matrix_form` stores an ensemble on both matrix and tensor form. This is implemented such that when the ensemble is updated on matrix form, the tensor representation is updated accordingly, and vice versa. This is very useful for the Markov structure imposed on the state-space variables in Section 4.3.

Recall from Section 2.1 that the filtering and forecasting problems are assessed through a set of observations $\boldsymbol{d}_t$ in each time-step. In order to create a reference solution in each time-step, we generate the initial reference solution $\boldsymbol{x}_0$ from (6), and then compute $\boldsymbol{x}_t$ using (2). The observations $\boldsymbol{d}_t$ are then generated through (10). In order to compare the results provided by the two prior distributions to the same reference solution, we store the reference solutions $\boldsymbol{x}_0, \dots, \boldsymbol{x}_{T+1}$ in `txt`-files. In the first numerical example, presented in Section 4.2, we assume that the forward function, (8), is deterministic. Thus, we only need to store the reference solution from the initial time-step, $\boldsymbol{x}_0$, because the reference solutions for time-steps $t = 1, \dots T+1$ can be acquired by applying the forward function in (2) on $\boldsymbol{x}_0$.

4.2 First numerical example

This section presents the setup of the first numerical example. This numerical example is largely based on the first numerical example presented in Omre and Myrseth (2010). The aim of the numerical examples is to compare results provided by the prior distribution used in Omre and Myrseth (2010) to the results provided by the prior distribution presented in Section 3.1. We first present the numerical values for the constants, vectors and matrices defined in the state space model, Section 2.1. Further, we define the numerical values for the parameters in the prior distributions presented in Sections 2.5 and 3.1.

For the likelihood model, presented in (10), we set $H = I_K$ and $\Sigma_d = \sigma_d^2 I_K$, with $\sigma_d^2 = 20$. Further, we set $T = 10$, which means that our objective is to assess $\boldsymbol{x}_{11}|\boldsymbol{d}_0, \dots, \boldsymbol{d}_{10}$. The reference solution in the initial time-step, is simulated by (6) with $\boldsymbol{\mu}_0 = \boldsymbol{0}$ and $\Sigma_0 = 20C$, where C is the correlation matrix, and is defined as

$$c(\Delta) = e^{-3\Delta/20}, \tag{126}$$

where Δ is the interdistance between the elements of $\boldsymbol{x}_0$. The reference solution $\boldsymbol{x}_t$ is computed by (8), where $\boldsymbol{v}_t = \boldsymbol{0}$ and where A_t is a $K \times K$-matrix. The forward function A_t forwards the 10 elements in $\boldsymbol{x}_t$ from $5t-4$ to $5t+5$ by a "10-node sliding average". That is,

$$x_{t+1}^k = \frac{1}{10} \sum_{s=k-4}^{k+5} x_t^s \quad \text{for } k \in \{5t-4, 5t+5\}. \tag{127}$$

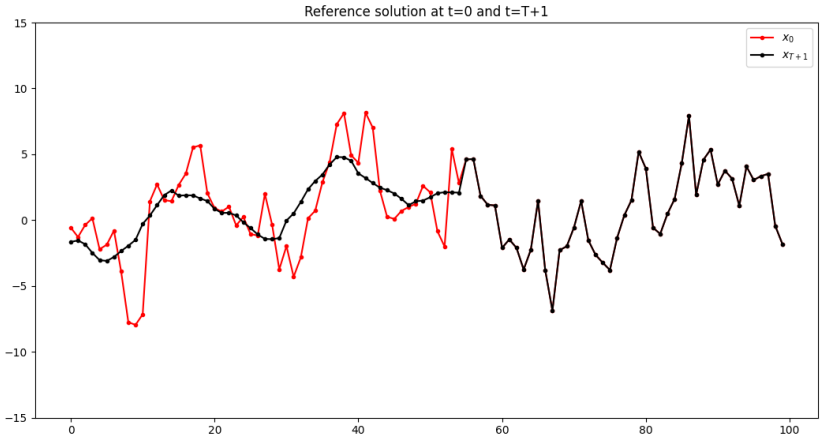


Figure 4: The red and black lines display the reference solutions at time-steps $t = 0$ and $t = T + 1 = 11$, respectively, for $K = 100$. For K between 56 and 100, the reference solutions coincide.

In the case when $k < 5$, we take the average of the nodes from 1 to $k + 5$, i.e.

$$x_{t+1}^k = \frac{1}{k+5} \sum_{s=1}^{k+5} x_t^s \quad \text{for } k < 5. \tag{128}$$

Similarly, when $k > K - 5$, we have

$$x_{t+1}^k = \frac{1}{K-k+6} \sum_{s=k-4}^{K} x_t^s \quad \text{for } k > K - 5. \tag{129}$$

The remaining elements of $\boldsymbol{x}_{t+1}$ are unchanged. That is, the forward function smooths the 10 elements in $\boldsymbol{x}_t$ between $5t - 4$ and $5t + 5$ in each iteration, while the remaining elements remain unchanged. The reference solutions at $t = 0$ and $t = 11$, i.e. realizations of $\boldsymbol{x}_0$ and $\boldsymbol{x}_{11}$, for $K = 100$ are visualized in red and black, respectively, in Figure 4. We notice that there are less fluctuations in the reference solution at $t = 11$ and in the reference solution at $t = 0$ in the first 55 elements, while the last 45 elements are identical. That is, the first half of the reference solution at $t = 11$ is smoother than the second half. This coincides with the forward function in (127). Also note that the forward function is linear. Since the forward function is linear, the model assumptions in the linear Gaussian state space model, presented in Section 2.2, holds, and hence the reference solution is Gaussian.

As also discussed above, our objective is to assess the computational cost of HEnKF for the two prior distributions presented in Sections 2.5 and 3.1. Hence

we run HEnKF for different values of K, which we recall is the dimension of $\boldsymbol{x}_t$. We run HEnKF for a set of values of K, ranging between $K = 100$ and $K = 10000$. The ensemble size, defined in Section 2.4, is set to $J = 10$.

We now set the values for the hyperparameters for the prior distributions presented in Omre and Myrseth (2010). Recall that the prior distributions for $\boldsymbol{\mu}_t|\Sigma_t$ and Σ_t are defined in (53) and (54), respectively. The values we set for the parameters of prior distributions resembles the values set in the first numerical example in Omre and Myrseth (2010). We set $\mathrm{E}(\boldsymbol{\mu}_t) = \boldsymbol{\xi}_t = \mathbf{0}, \alpha_t = 500, \Psi_t = (\nu_t - K - 1)\Sigma_0$ and $\nu_t = K + 30$.

In Section 3.1, we present the prior distributions for $\boldsymbol{\phi}_t$ and $\boldsymbol{\eta}_t|\boldsymbol{\phi}_t$, defined in (93) and (94), respectively. We now present the parameter values for these distributions. Recall that we impose a sequential neighbourhood Λ_k on each element of $\boldsymbol{x}_t$, i.e. x_t^k. In order to assess how the performance of the prior distributions depends on the choice of sequential neighbourhood, we present three different examples. In each example, we impose a different sequential neighbourhood on the elements of $\boldsymbol{x}_t$. In the following, we present the values of the hyperparameters in the three different examples.

In the first example, we set the following parameter values. We set $\Lambda_k = \{k-1\}$, i.e. a first order Markov chain. We set $\mathrm{Cov}(\boldsymbol{\eta}_t^k|\phi_t^k) = \Sigma_{\boldsymbol{\eta}_t^k} = \sigma^2_{\boldsymbol{\eta}_t^k} I_{|\Lambda_k|+1}$, with $\sigma^2_{\boldsymbol{\eta}_t^k} = 100$ for $k \in \{1, \ldots, K\}$. In the second example, we set $\Lambda_k = \{k-2, k-1\}$, i.e. a second order Markov chain. As for the first example, we set $\mathrm{Cov}(\boldsymbol{\eta}_t^k|\phi_t^k) = \Sigma_{\boldsymbol{\eta}_t^k} = \sigma^2_{\boldsymbol{\eta}_t^k} I_{|\Lambda_k|+1}$, with $\sigma^2_{\boldsymbol{\eta}_t^k} = 100$ for $k \in \{1, \ldots, K\}$. In the third example, we set $\Lambda_k = \{k-5, \ldots, k-1\}$, i.e. a fifth order Markov chain. We set $\Sigma_{\boldsymbol{\eta}_t^k} = \mathrm{diag}(100, 5, \ldots, 5)$, where $\Sigma_{\boldsymbol{\eta}_t^k} \in \mathbb{R}^{(|\Lambda_k|+1)\times(|\Lambda_k|+1)}$. That is, the first diagonal element is 100, i.e. $\Sigma^{1,1}_{\boldsymbol{\eta}_t^k} = 100$, while the remaining diagonal elements are set to 5. Notice that we in all three examples set the elements of $\boldsymbol{\eta}_t^k$ to be a priori independent. In all of the three examples we set $\mathrm{E}(\boldsymbol{\eta}_t^k|\phi_t^k) = \boldsymbol{\mu}_{\boldsymbol{\eta}_t^k} = \mathbf{0}$ for $k \in \{1, \ldots, K\}$. We also set the same values for the hyperparameters of ϕ_t^k in all three examples. We set $\alpha_t^k = 2.5$ and $\beta_t^k = 7.5$ for $k \in \{1, \ldots, K\}$.

4.3 Second numerical example

In this section, we present the setup of the second numerical example in this report. The aim of this numerical example is to compare the results provided by the prior distribution presented in Omre and Myrseth (2010) to the results provided by the prior distribution introduced in Section 3.1 on a nonlinear state space model, where we recall that the state space model is defined in Section 2.1. Additionally, we represent the state space vector $\boldsymbol{x}_t$ by a two-dimensional grid. For the prior distribution presented in 3.1, we apply a Markov structure on the state space variable $\boldsymbol{x}_t$ that adapts to the grid. This enables us to assess how the run time of HEnKF increases as the bandwidth of the precision matrix of $\boldsymbol{x}_t$ increases. As in Section 4.2, we first present the numerical values for the constants, vectors and matrices in the state space model. We then define the

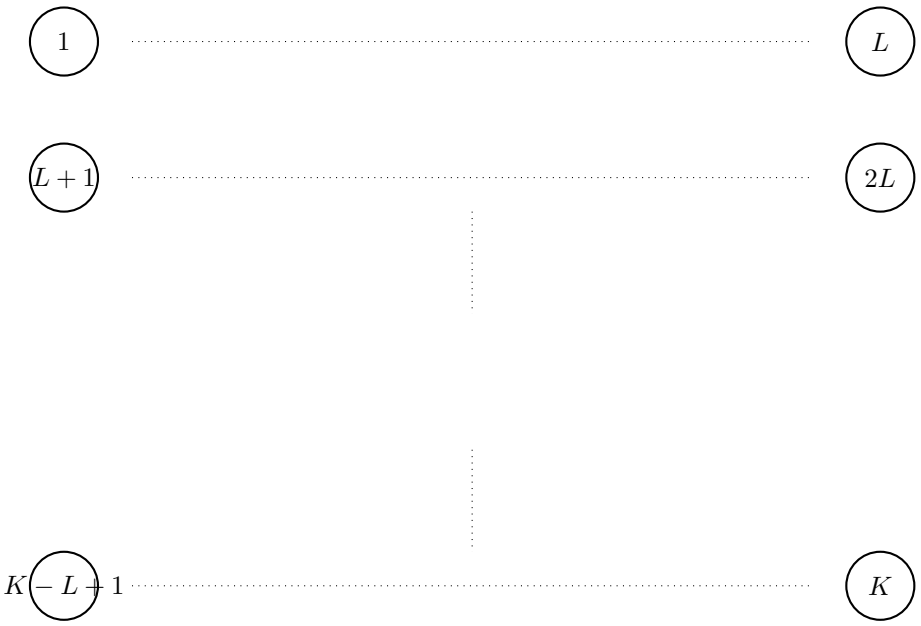


Figure 5: The grid associated to the state space vector $\boldsymbol{x}_t$. Node k in the grid is associated to the kth element in the space vector $\boldsymbol{x}_t$. The grid is quadratic with length and width L and enumerated from left to right, top down.

numerical values for the hyperparameters in the two prior distributions that are compared.

In the likelihood model, from (10), we set $H = I_K$ and $\Sigma_d = \sigma_d^2 I_K$ with $\sigma_d^2 = 2$, where K is the dimension of the state space variable $\boldsymbol{x}_t$. Further, we set $T = 4$. That is, our aim is to estimate the one-step forecasting distribution $f(\boldsymbol{x}_5|\boldsymbol{d}_{0:4})$. The reference solution in the initial time-step, which is a realization of $\boldsymbol{x}_0$, is sampled from a multivariate standard normal distribution. Recall from Section 2.6, that the conditional dependencies between the elements of the state space vector $\boldsymbol{x}_t$ is represented by a graph. Node k in the graph represents the kth element in $\boldsymbol{x}_t$, and Λ_k represents the set of sequential neighbours for node k. In this example, we visualize the graph associated to $\boldsymbol{x}_t$ as a two-dimensional grid of nodes, where the length and the width of the grid are L, as visualized in Figure 5. That is, the grid is quadratic with dimensions $L \times L$ and $K = L^2$. The nodes in the graph are enumerated row-by-row, from left to right. That is, node 1 is in the upper left corner, while node L is in the upper right corner. Node $K - L$ is in the lower left corner, while node $K = L^2$ is in the lower right corner. In the following, we present the forward function.

The forward function takes the K nodes in the graph, and chooses $m < K$ of these nodes randomly. In this example, we set $m = K/10$. For each of these nodes, we denote a as the average of the nodes inside a 5×5-window centered

around the randomly chosen node. As an illustrative example, the nodes inside the 5×5-window around node k are colored yellow in Figure 6. We denote the number of nodes inside this window as n. In the case when a randomly chosen node is along one of the edges or corners of the grid, such as node 1, we take the average of the nodes inside the 5×5-window that are also inside the grid. That is, for node 1, we take the average of the $n = 9$ closest nodes. Further, we define $r = a\sqrt{n}$. We then compute $u = \Phi(r)$, where Φ is the cdf of the standard normal distribution. Further, we compute $s = F_t^{-1}(u, \tau)$, where F_t^{-1} is the inverse cdf of the t-distribution with τ degrees of freedom. In this example, we set $\tau = 500$. To ensure that this variable is scaled correctly, we compute

$$w = \frac{s}{\sqrt{\tau/(\tau+1)}}. \tag{130}$$

The value in the randomly chosen node is then set to w. That is, the forward function assumes that r is standard normally distributed, and then transforms r into a t-distributed variable with τ degrees of freedom. The motivation behind the choice of forward function is that the distribution for the elements of $\boldsymbol{x}_t$ becomes more heavy-tailed for each iteration, since the t-distribution has heavier tails than the standard normal distribution. This function also enables us to compare the two prior distribution on a nonlinear state space model. The reference solutions at $t = 0$ and $t = T + 1 = 5$, i.e. realizations of $\boldsymbol{x}_0$ and $\boldsymbol{x}_5$, are visualized in Figure 7, in red and black, respectively. We notice that the reference solution at $t = 5$ is somewhat more heavy-tailed than the reference solution at $t = 0$.

As in the first numerical example, introduced in Section 4.2, the reference solution $\boldsymbol{x}_t$ is computed by (2). Since the forward function is nonlinear, the reference solution is generally not Gaussian. As mentioned in Section 4.1, we store the reference solutions in order to compare the two prior distributions to the same reference solutions. The ensemble size, defined in Section 2.4, is set to $J = 10$.

We now set the values for the hyperparameters in the prior distributions presented in Omre and Myrseth (2010). Recall that the prior distributions for $\boldsymbol{\mu}_t|\Sigma_t$ and Σ_t are defined in (53) and (54), respectively. We set $\mathrm{E}(\boldsymbol{\mu}_t) = \boldsymbol{\xi}_t = \mathbf{0}$, $\alpha_t = 500$, $\Psi_t = (\nu_0 - K - 1)\Sigma_0$ and $\nu_t = K + 30$.

In Section 3.1, we introduced prior distributions for $\boldsymbol{\phi}_t$ and $\boldsymbol{\eta}_t|\boldsymbol{\phi}_t$, defined in (93) and (94), respectively. We now set the parameters values for these distributions. Recall from Section 2.6 that we assign a sequential neighbourhood Λ_k for each node k in the graph associated to $\boldsymbol{x}_t$. We assume that this graph is identical to the grid applied in the forward function. That is, the grid is quadratic with length and width L. The sequential neighbourhood for node k is defined as $\Lambda_k = \{k - L - 1, k - L, k - 1\}$. We visualize a part of the grid and the sequential neighbours of node k in Figure 8, where node k is colored green and its sequential neighbours are colored yellow. We draw edges between the nodes, in accordance with the theory presented in Section 2.6. We draw green edges between node k and the nodes in Λ_k, and we draw yellow edges between all pair of nodes in Λ_k. Further, we set $\Sigma_{\boldsymbol{\eta}_t^k} = \mathrm{diag}(100, 5, \ldots, 5)$,

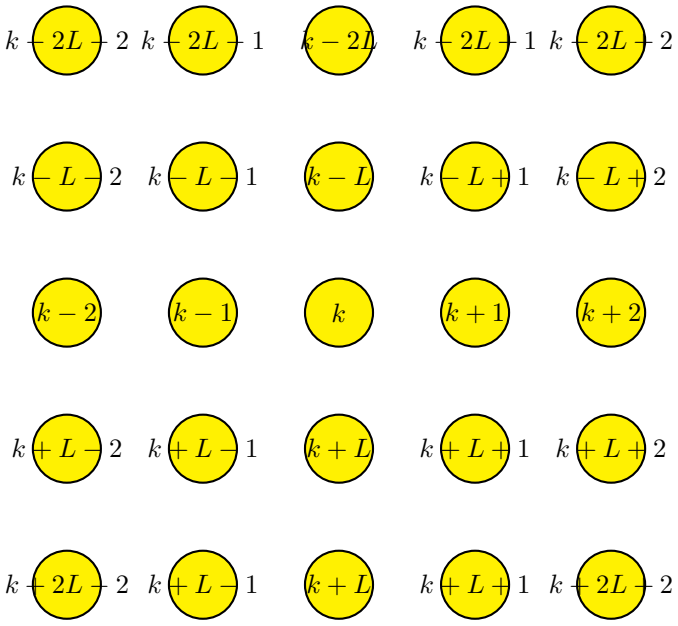


Figure 6: Each node represents an element of the state space vector $\boldsymbol{x}_t$. The yellow nodes represents the 5×5-window centered around node k.

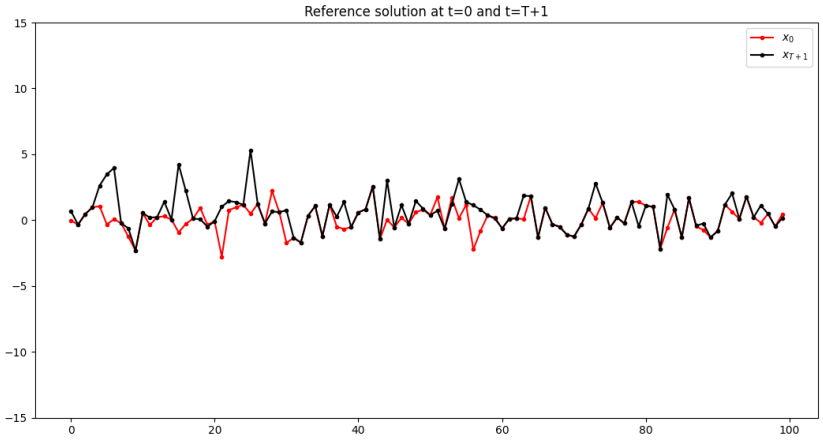


Figure 7: The red and black lines visualize the reference solutions at time-steps $t = 0$ and $t = T + 1 = 5$, respectively. That is, realizations of $\boldsymbol{x}_0$ and $\boldsymbol{x}_5$ for $K = 100$.

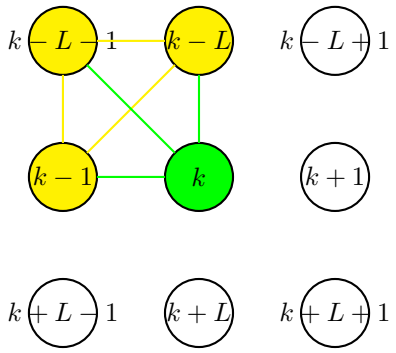


Figure 8: Each node represents an element of the state space vector $\boldsymbol{x}_t$. The yellow nodes represent the sequential neighbourhood of node k, denoted Λ_k. The edges indicate the conditional dependencies between the nodes, in accordance with the theory presented in Section 2.6. That is, the green edges indicate that $x_t^r \not\perp x_t^k | \boldsymbol{x}_t^{-\{k,r\}}$ for some $r \in \Lambda_k$, while the yellow edges indicate that $x_t^r \not\perp x_t^s | \boldsymbol{x}_t^{-\{r,s\}}$ for some $r, s \in \Lambda_k$.

where $\Sigma_{\boldsymbol{\eta}_t^k} \in \mathbb{R}^{(|\Lambda_k|+1)\times(|\Lambda_k|+1)}$. That is, the first diagonal element is 100, i.e. $\Sigma_{\boldsymbol{\eta}_t^k}^{1,1} = 100$, while the remaining diagonal elements are set to 5. Further, we set $\mathrm{E}(\boldsymbol{\eta}_t^k | \phi_t^k) = \boldsymbol{\mu}_{\boldsymbol{\eta}_t^k} = \mathbf{0}$ for $k \in \{1, \ldots, K\}$. For the hyperparameters of ϕ_t^k, we set $\alpha_t^k = 2.5$ and $\beta_t^k = 7.5$ for $k \in \{1, \ldots, K\}$. Note that we set the same parameter values for $t = 0, \ldots T+1$.

5 Results and discussion

In this section, we present the results from the numerical examples described in Section 4. Sections 5.1 and 5.2 present the results from the numerical examples, introduced in Sections 4.2 and 4.3, respectively. Our overall objective is to compare the results provided by the prior distributions from Omre and Myrseth (2010), which we present in Section 2.5, to the results provided by the prior distributions introduced in Section 3.1, by considering their computational costs and the quality of the results. In both of the numerical examples, the ensemble size is set to $J = 10$.

5.1 First numerical example

As mentioned, we compare the results provided by the prior distributions introduced in Section 2.5 to the results provided by the prior distributions presented in Section 3.1. In the following, we refer to the prior distributions presented in Section 2.5 as the *old* prior distributions, while the prior distributions presented in Section 3.1 are referred to as the *new* prior distributions. Recall from Section

4.2, that we compare the results provided by the two prior distributions for different values of K, where K is the dimension of the state space vector $\boldsymbol{x}_t$. Also recall from Section 4.2 that we run HEnKF with the new prior distributions for three different sets of prior assumptions. In the three examples, we assume that the state space vector $\boldsymbol{x}_t$ follows a first, second and fifth order Markov chain, respectively. That is, we compare results provided by four different prior assumptions in total; one using the old prior distributions, and three using the new.

We want to compare the two prior distributions by comparing their computational cost, and by comparing the quality of the results. In order to assess the quality of the results, we present a set of plots. In each plot, we visualize the reference solution at time-step $t = 11$, which is a realization from $f(\boldsymbol{x}_{11})$. The realization is displayed along with a 95% empirical prediction interval of $\boldsymbol{x}_{11}|\boldsymbol{d}_{0:10}$. This interval is based on the ensemble χ_{11}, which we recall is an approximation of the one-step forecasting distribution $f(\boldsymbol{x}_{11}|\boldsymbol{d}_{0:10})$. The ensemble is created using one of the four different prior assumptions. In the following, we present the prediction intervals created for $K = 100$.

Figure 9 visualizes the reference solution at $t = 11$ along with a 95% empirical prediction interval created with the old prior distribution for $K = 100$. Figures 10, 11 and 12 illustrate 95% empirical prediction intervals created with the new prior distribution, for a first, second and fifth order Markov chain, respectively. By comparing the four prediction intervals, we notice that all of the prediction intervals manage to capture the fluctuations in the first half of the reference solution quite well. For the second half, we notice that the old prior distribution and the new prior distribution with a fifth order Markov chain provide shorter prediction intervals that fluctuate rapidly, which seize to capture the sharp movements in the reference solution. We observe that the prediction intervals provided by the new prior distribution with first and second order Markov chains are smoother, and do not capture the movements in the second half of the reference solution. The prediction intervals seize to contain almost the entire reference solution, nonetheless.

Figures 13 to 16 visualize the same as Figures 9 to 12, for $K = 1000$, rather than $K = 100$. Note that we only display the first 100 elements, as displaying all 1000 elements yields plots that are practically impossible to interpret. From Figures 13 to 16, we observe similar effects as for $K = 100$. That is, all four prediction intervals manage to capture the movements in the first half, while only the prediction intervals provided by the old prior distribution and the new prior distribution with a fifth order Markov chain capture the movements in the second half. However, the prediction intervals provided by the new prior distribution with first and second order Markov chains still manage to contain the reference solution. Also here we notice that the new prior distribution with first and second order Markov chains provide smoother prediction intervals than the fifth order Markov chain and the old prior distribution.

Figures 17 to 20 visualize the same as Figures 13 to 16, however for $K = 10000$. Here we notice similar effects as for $K = 100$ and $K = 1000$. That is, all of the four prior assumption provide prediction intervals that capture the movements in

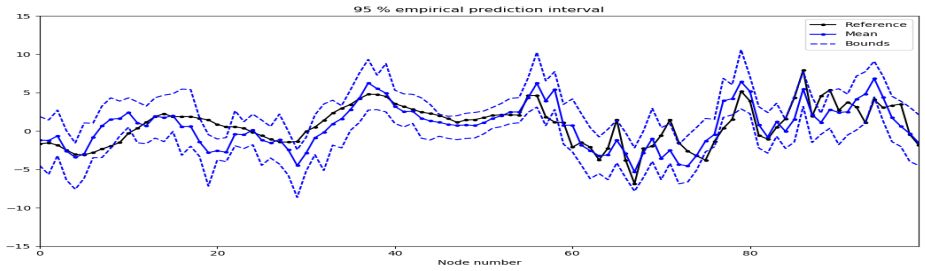


Figure 9: Old prior, $K = 100$.

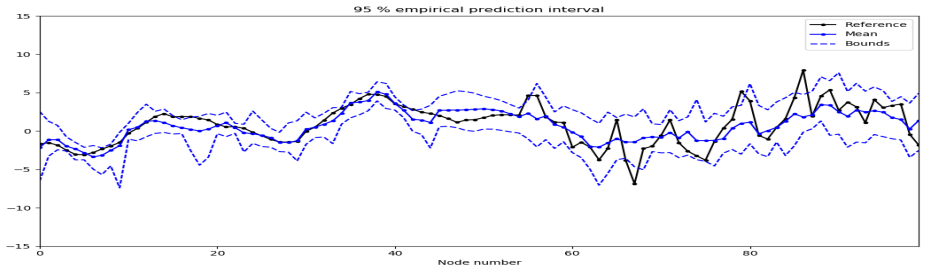


Figure 10: New prior, first order Markov chain, $K = 100$.

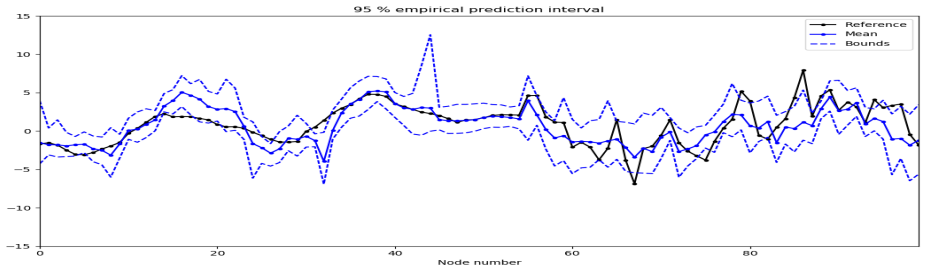


Figure 11: New prior, second order Markov chain, $K = 100$.

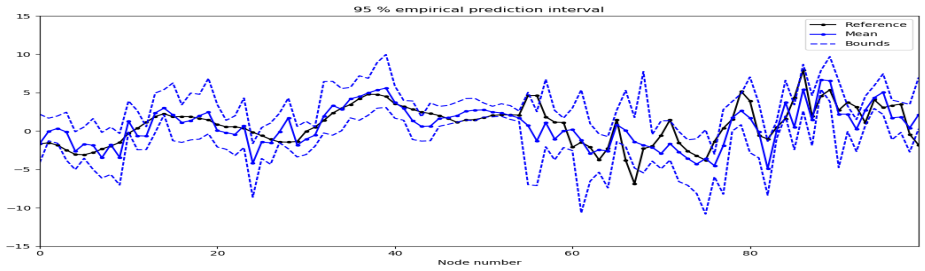


Figure 12: New prior, fifth order Markov chain, $K = 100$. The black line represents the reference solution at $t = 11$ for $K = 100$. The continuous, blue line represents the mean of the ensemble χ_{11} for $K = 100$. The dashed, blue lines represent the bounds of the 95% empirical prediction interval based on the ensemble.

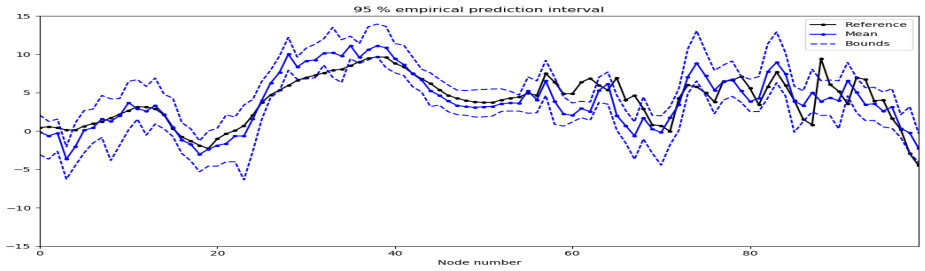


Figure 13: Old prior, $K = 1000$.

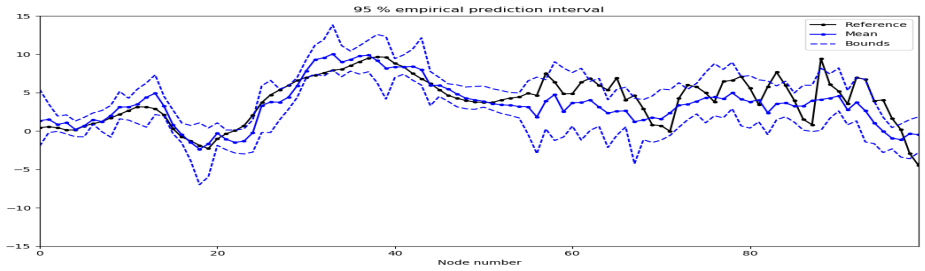


Figure 14: New prior, first order Markov chain, $K = 1000$.

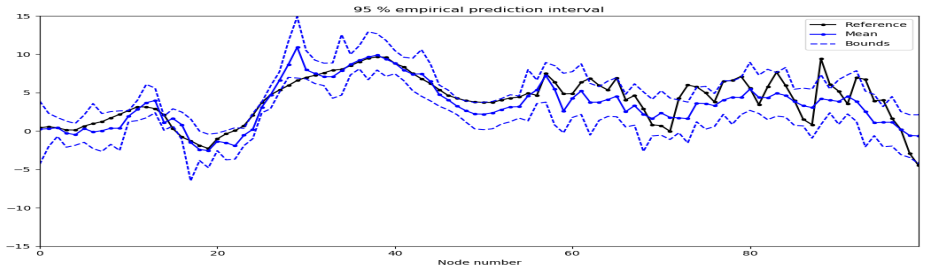


Figure 15: New prior, second order Markov chain, $K = 1000$.

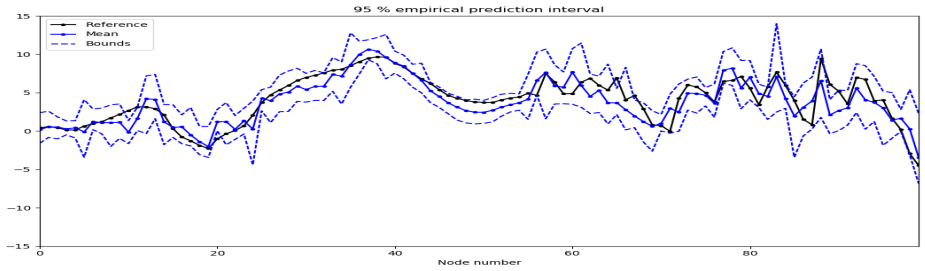


Figure 16: New prior, fifth order Markov chain, $K = 1000$. The black line represents the reference solution at $t = 11$ for $K = 1000$. The continuous, blue line represents the mean of the ensemble χ_{11} for $K = 1000$. The dashed, blue lines represent the bounds of the 95% empirical prediction interval based on the ensemble.

the first half quite well. In the second half, the old prior distribution and the new prior distribution with a fifth order provide shorter prediction intervals. However, we notice that the prediction intervals provided by new prior distribution with a fifth order Markov chain better capture the rapid fluctuations in the second half of the reference solution. This effect is especially visible in the area around $K = 60$ and $K = 70$. The prediction intervals created with the new prior distribution using first and second order Markov chains do not capture the rapid fluctuation in the second half, but still seize to contain almost the entire reference solution. As for $K = 100$ and $K = 1000$, we notice that the new prior distribution with first and second order Markov chains provide smoother prediction intervals than the fifth order Markov chain.

In general, from Figures 9 to 20, we observe that the old prior distribution and the new prior distribution with a fifth order Markov chain provide short prediction intervals that capture the movements in reference solution well, compared to the new prior distribution with first and second order Markov chains. It should be noted that this numerical example also was performed with values of K other than the ones presented here, and that the figures from these examples shows similar effects. A plausible explanation to this effect can be found by considering the number of parameters in the two prior distributions. In the following, we discuss how the number of parameters in the two prior distributions influence the prediction intervals.

The old prior distribution has $\mathcal{O}(K^2)$ parameters, as seen in (53) and (54), while we see from (94) and (95) that the new prior distribution has $\mathcal{O}(Km^2)$ parameters, where m is the order of the Markov chain imposed on $\boldsymbol{x}_t$. Recall from Section 2.5 that these parameters are adjusted to account for the information in the observations, through the likelihood model. That is, for the old prior distribution, there are $\mathcal{O}(K^2)$ parameters that are adjusted, while there are $\mathcal{O}(Km^2)$ parameters in the new prior distribution that are adjusted. Since $m^2 < K$ in this example, there are more parameters in the old prior distribution that are being adjusted to the information in the observations. This suggests that the ensemble provided by the old prior distribution is more flexible to adjusting to the observations, since the old prior distribution provides the observations with more parameters to adjust. That is, this suggests that the prediction intervals provided by the old prior distribution are more agile. This is a plausible explanation to why these prediction intervals capture the movements in the reference solution well, compared to the prediction intervals provided by the new prior distribution with first and second order Markov chains. We also notice from the figures that the new prior distribution with a fifth order Markov chain provides prediction intervals that captures the fluctuations in the reference solution. A plausible explanation is that the ensemble provided by this distribution has a sufficient amount of parameters to adjust to the observations.

From Figures 9, 13 and 17, we also notice that prediction intervals provided by the old prior distribution become shorter and smoother as K increases. These effects are also visible in the prediction intervals created for the values of K other than the ones presented here. A plausible explanation can be found by considering the posterior distribution corresponding to the old prior distribution.

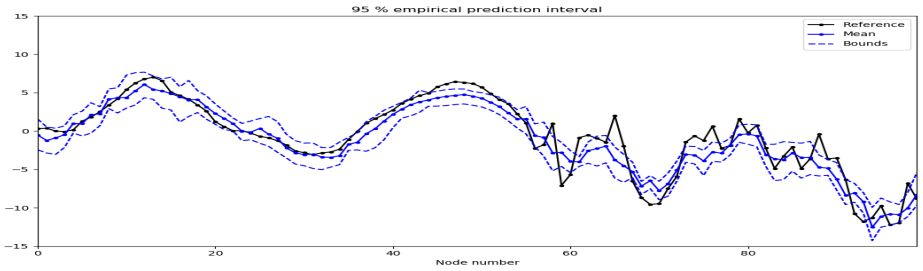


Figure 17: Old prior, $K = 10000$.

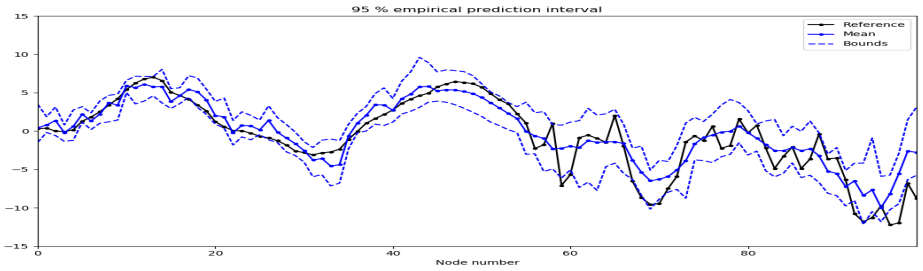


Figure 18: New prior, first order Markov chain, $K = 10000$.

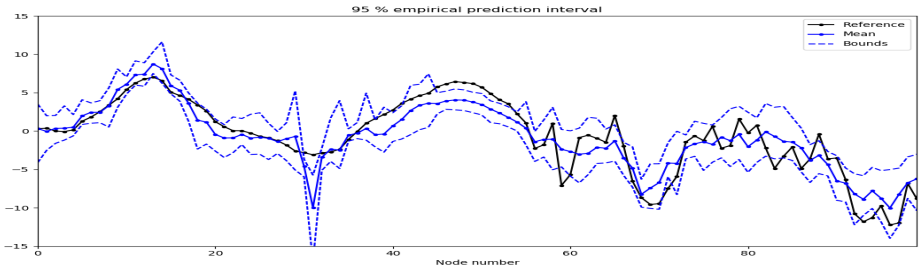


Figure 19: New prior, second order Markov chain, $K = 10000$.

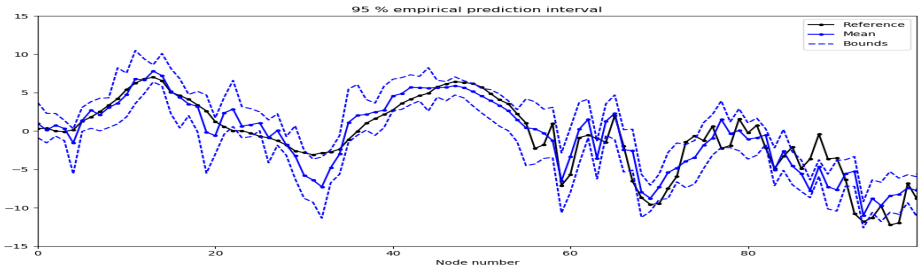


Figure 20: New prior, fifth order Markov chain, $K = 10000$. The black line represents the reference solution at $t = 11$ for $K = 10000$. The continuous, blue line represents the mean of the ensemble χ_{11} for $K = 10000$. The dashed, blue lines represent the bounds of the 95% empirical prediction interval based on the ensemble.

Recall that we in (59) compute Ψ_t^*, which is the matrix parameter in the posterior distribution for Σ_t, i.e. $f(\Sigma_t|\chi_t)$. In the expression for Ψ_t^*, we notice that the expression involves $\hat{\Sigma}_{t|0:t-1}$, where $\hat{\Sigma}_{t|0:t-1}$ is defined in (45) as the empirical covariance matrix based on the ensemble χ_t. If the number of entries in $\hat{\Sigma}_{t|0:t-1}$ is much larger than the number of entries in χ_t, the variance of the estimator $\hat{\Sigma}_{t|0:t-1}$ becomes large. In this example, we have that $J = 10$, which means that the ensemble is a matrix of size $K \times 10$, i.e. $\chi_t \in \mathbb{R}^{K\times 10}$. However, the number of entries in $\hat{\Sigma}_{t|0:t-1}$ is K^2, since $\hat{\Sigma}_{t|0:t-1} \in \mathbb{R}^{K\times K}$. Since $K^2 \gg 10K$, the number of entries to be estimated is much larger than the number of entries in χ_t. When the variance of the estimator is high, there is a chance that $\hat{\Sigma}_{t|0:t-1}$ is an overestimate of the true covariance matrix of $\boldsymbol{x}_t$. This means that the estimator $\hat{\Sigma}_{t|0:t-1}$ assumes that the correlation between the elements of $\boldsymbol{x}_t$ is higher than what is inherent in the true covariance of $\boldsymbol{x}_t$. This suggests that the elements of each ensemble member become more correlated, and that the resulting prediction intervals become smoother. Also, if $\hat{\Sigma}_{t|0:t-1}$ is an overestimate of the true covariance matrix, this suggests that the information contained in $\boldsymbol{d}_t$ is more spread out among the elements in each ensemble member. That is, $\boldsymbol{\chi}_{t,j}$ relies more on the information in the observations, which suggests that the prediction intervals shrink. Note that if this effect is caused by the estimation variance of $\hat{\Sigma}_{t|0:t-1}$, this issue can be solved by increasing the ensemble size J, which causes the estimation variance of $\hat{\Sigma}_{t|0:t-1}$ to decrease.

From the presented figures, we do not observe that the prediction intervals provided by the new prior distributions become shorter as K increases, as we observe for the old prior distribution. A plausible explanation to this is that the new prior distribution imposes a Markov chain on $\boldsymbol{x}_t$, which reduces the chance of overestimating the correlation between the elements of $\boldsymbol{x}_t$. That is, the Markov assumption prevents two elements of $\boldsymbol{x}_t$ far away from each other to be conditionally dependent given the remaining elements of $\boldsymbol{x}_t$. This is also visible by considering the precision matrix Q_t. For the new prior distribution, most of the entries in Q_t are zero, which enforces conditional independence between most of the elements of $\boldsymbol{x}_t$. The old prior distribution, however, assumes that Σ_t is a full matrix, which makes it possible for the correlation between the elements of $\boldsymbol{x}_t$ to be overestimated.

As mentioned, we also want to compare the computational cost of HEnKF using the different prior distributions. This is done by counting the CPU-time for each of the four examples, for a set of different values of K between 100 and 10000. Figure 21 shows the CPU-time of HEnKF using the different prior distributions as a function of K. The blue, yellow and green lines display the CPU-time when imposing a first, second and fifth order Markov chain on $\boldsymbol{x}_t$, respectively. The red line displays the CPU-time for the old prior distribution. We expect the CPU-time to increase when the Markov order increases, as the number of computations in each matrix multiplication increases. However, we see that the CPU-time for the first and second order Markov chains are practically identical, and that the CPU-time for the fifth order Markov chain is only slightly larger. We also see that the new prior distribution provide a large reduction

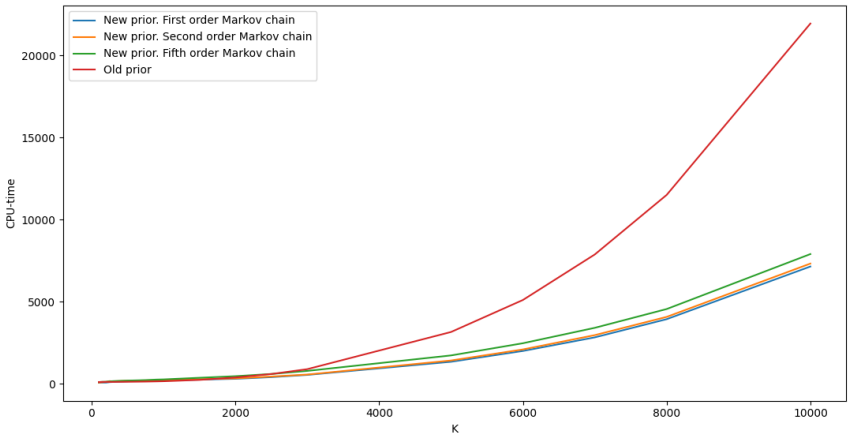


Figure 21: Comparison of CPU-times as a function of the dimension K. The red line represents the CPU-time in seconds as a function of K for the old prior distribution. The blue, yellow and green lines represent the new prior distributions for a first, second and fifth order Markov chain, respectively.

in computational cost, compared to the old prior distribution. For $K = 10000$, the CPU-time using the old prior distribution is close to 22000 seconds, while the CPU-time when imposing a first order Markov chain is approximately 7100 seconds. It should also be noted that the CPU-time varies slightly for each run. However, for $K = 10000$, this variation is small.

In Section 2.7, we state the computational complexity of Cholesky decomposing a $K \times K$-matrix can be reduced from $\mathcal{O}(K^3)$ to $\mathcal{O}(Km^2)$ by applying Algorithm 3, when we assume that the matrix is sparse and has known bandwidth m. From Section 2.6, we have that the bandwidth of the precision matrix of $\boldsymbol{x}_t$, denoted Q_t, is equal to the order of the Markov chain imposed on $\boldsymbol{x}_t$. Thus, Algorithm 4 suggests that (61) can be computed with a computational complexity that is linear as a function of K, since m is constant as a function of K. If every computation in HEnKF could be performed with a computational complexity of $\mathcal{O}(K)$ or less, we would expect the CPU-time to increase linearly as a function of K. However, from Figure 21, we see that the CPU-time is not linear as a function of K, but suggests that the computational complexity is higher than $\mathcal{O}(K)$. This might be due to the sampling of the ensemble members in the initial time-step $\boldsymbol{\chi}_{0,j}$, that are sampled from $N(\mathbf{0}, \Sigma_0)$. Recall that Σ_0, in (126), is a full matrix, and that sampling from a multivariate normal distribution involves computing B such that $BB^\intercal = \Sigma_0$. The computation of B can generally not be performed with a computational complexity of $\mathcal{O}(K)$ or less.

5.2 Second numerical example

As mentioned in Section 4.3, we compare the prior distribution presented in Omre and Myrseth (2010) to the prior distribution introduced in Section 3.1 on a nonlinear state space model, where the state space model is defined in Section 2.1. This enables us to assess the quality of the results provided by the two prior distributions when the assumptions of a linear Gaussian state space model, Section 2.2, do not hold. As in Section 5.1, we refer to the prior distribution in Omre and Myrseth (2010) as the *old* prior distribution, while the prior distribution from Section 3.1 are referred to as the *new* prior distribution.

The numerical performance of the two prior distributions are compared in the same way as in Section 5.1. That is, we compare the CPU-time for HEnKF using the two prior distributions, and we assess the quality of the results. The quality of the results are assessed by comparing the 95% empirical prediction intervals provided by the ensemble at time-step $t = T + 1 = 5$, χ_5, to the reference solution at the same time-step. Recall from Section 2.4 that χ_5 approximates the one-step forecasting distribution $f(\boldsymbol{x}_5|\boldsymbol{d}_{0:4})$. We compare the results for $K = 100$, 4900 and 10000. In the following, we first consider $K = 100$.

Figure 22 displays the prediction interval provided by the old prior distribution for $K = 100$. Similarly, the prediction intervals provided by the new prior distribution are visualized in Figure 23 for $K = 100$. We notice that the prediction intervals provided by both prior distributions capture almost the entire reference solution. In addition, we notice the new prior distribution provides longer prediction intervals than the old prior distribution.

Figures 24 and 25 present the prediction intervals provided by the old and new prior distributions, respectively, for $K = 4900$. Note that the figures only visualize the 100 elements of $\boldsymbol{x}_t$ between 2400 and 2500, as displaying all 4900 elements yields plots that are impossible to interpret. By comparing the two intervals, we see that the prediction interval provided by the new prior distribution better captures the movements in the reference solution, compared to the old prior distribution. We also notice that the new prior distributions provide longer prediction intervals than the old prior distribution.

Figures 26 and 27 display the prediction intervals provided by the old and new prior distributions, respectively, for $K = 10000$. Note that we display the 100 elements between 4950 and 5050. Here we notice that the old prior distribution fails to produce a prediction interval that captures the movements in the reference solution. However, we see in Figure 27 that the prediction intervals provided by the new prior distribution almost contain the entire reference solution. We also notice that the old prior distribution provides a much smaller prediction interval than the new prior distribution.

In general, we notice that both prior distributions provide reliable prediction intervals for $K = 100$. We also observe that the new prior distribution manages to provide reliable prediction intervals as K increases, while the prediction intervals provided by the old prior distribution becomes shorter decreases as K increases, which prevents the prediction intervals to capture the reference solution. This numerical example was also performed with values of K other than the ones

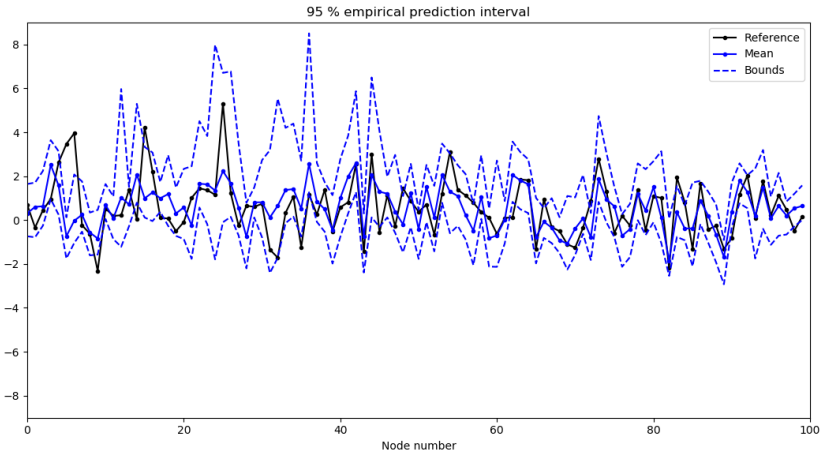


Figure 22: Old prior, $K = 100$. The black line represents the reference solution at $t = 5$ for $K = 100$. The continuous, blue line represents the mean of the ensemble χ_5, which is created with the old prior distribution for $K = 100$. The dashed, blue lines represent the bounds of the 95% empirical prediction interval based on the ensemble.

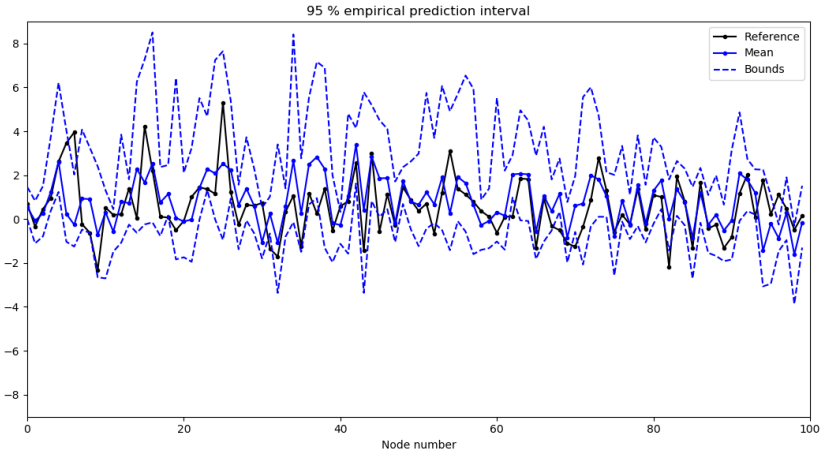


Figure 23: New prior, $K = 100$. The black line represents the reference solution at $t = 5$ for $K = 100$. The continuous, blue line represents the mean of the ensemble χ_5, which is created with the new prior distribution for $K = 100$. The dashed, blue lines represent the bounds of the 95% empirical prediction interval based on the ensemble.

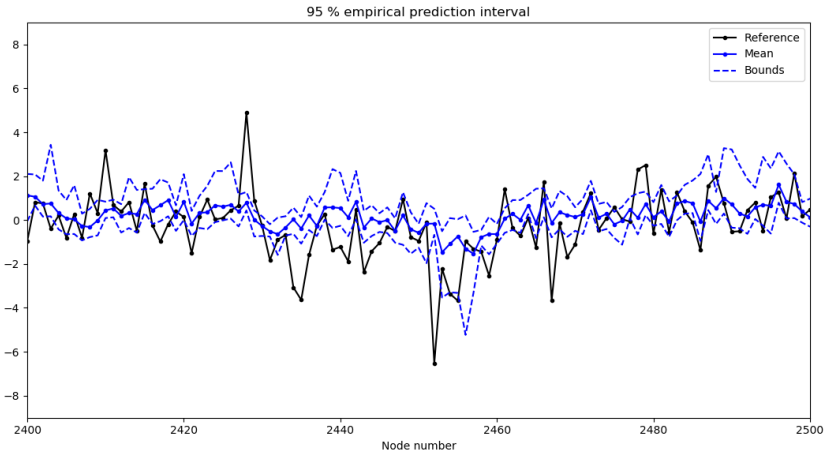


Figure 24: Old prior, $K = 4900$. The black line represents the reference solution at $t = 5$ for $K = 4900$. The continuous, blue line represents the mean of the ensemble χ_5, which is created with the old prior distribution for $K = 4900$. The dashed, blue lines represent the bounds of the 95% empirical prediction interval based on the ensemble.

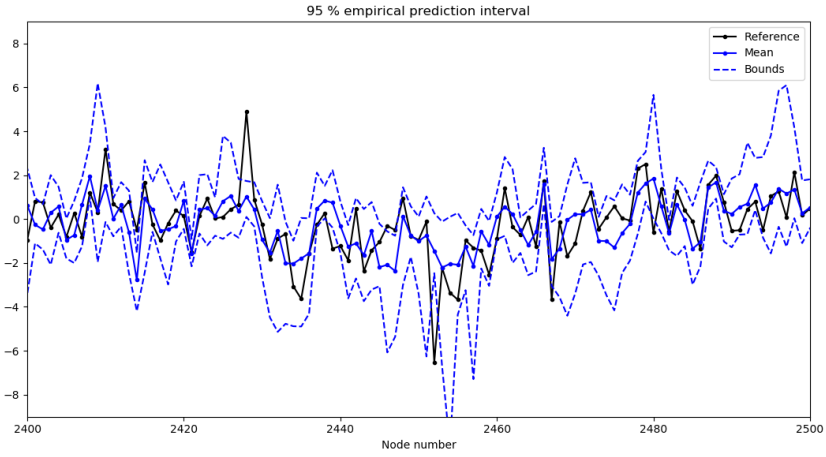


Figure 25: New prior, $K = 4900$. The black line represents the reference solution at $t = 5$ for $K = 4900$. The continuous, blue line represents the mean of the ensemble χ_5, which is created with the new prior distribution for $K = 4900$. The dashed, blue lines represent the bounds of the 95% empirical prediction interval based on the ensemble.

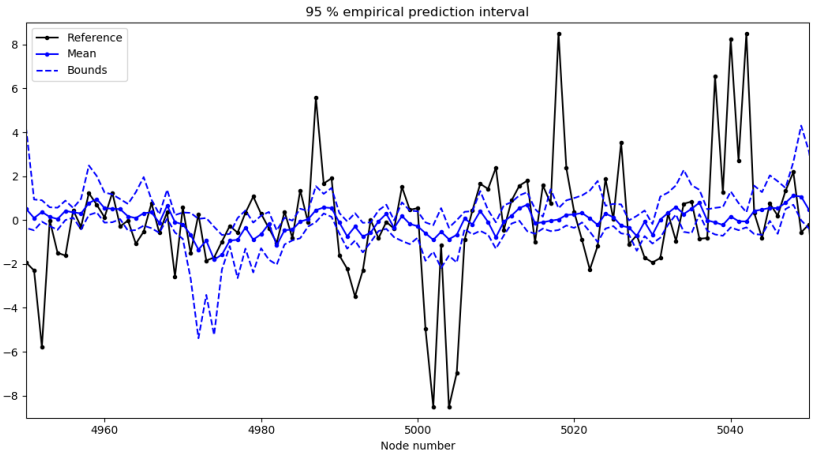


Figure 26: Old prior, $K = 10000$. The black line represents the reference solution at $t = 5$ for $K = 10000$. The continuous, blue line represents the mean of the ensemble χ_5, which is created with the old prior distribution for $K = 10000$. The dashed, blue lines represent the bounds of the 95% empirical prediction interval based on the ensemble.

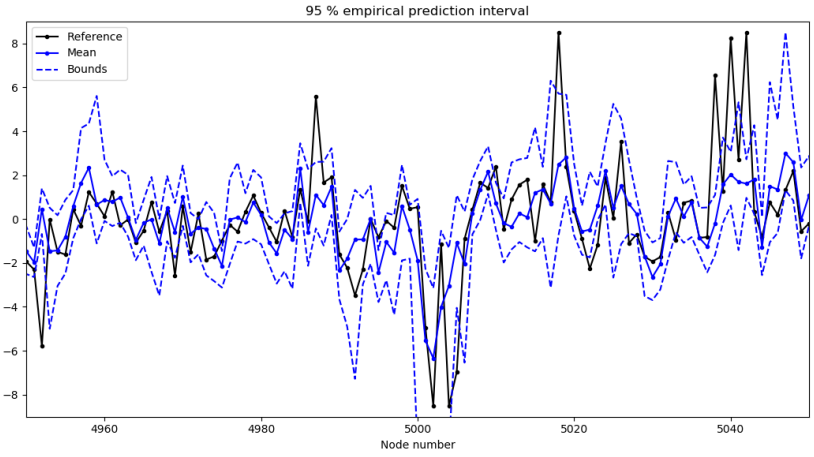


Figure 27: New prior, $K = 10000$. The black line represents the reference solution at $t = 5$ for $K = 10000$. The continuous, blue line represents the mean of the ensemble χ_5, which is created with the new prior distribution for $K = 10000$. The dashed, blue lines represent the bounds of the 95% empirical prediction interval based on the ensemble.

presented here, and the results from these examples suggest the same effects. A plausible explanation to this, is that the old prior distribution estimates the correlation between the elements of $\boldsymbol{x}_t$ to be higher than what is inherent in the true covariance matrix of $\boldsymbol{x}_t$, as discussed in Section 5.1. This might cause the prediction intervals to become shorter. A plausible explanation why the new prior distribution provides reliable results, is related to the Markov structure imposed on $\boldsymbol{x}_t$. First, the Markov property puts restrictions on the conditional dependencies between the elements of $\boldsymbol{x}_t$, as discussed in Section 5.1. Second, since the Markov structure imposed on $\boldsymbol{x}_t$, i.e. $\Lambda_k = \{k-L-1, k-L, k-1\}$, adapts to the grid associated to $\boldsymbol{x}_t$, the new prior distribution is provided information about the grid applied in the state space model. The old prior distribution makes no assumptions about the grid, which might suggests that the new prior distribution is able to utilize the information about the grid to produce more reliable results.

As for the first numerical example, we compare the CPU-time of HEnKF for the two prior distributions. In practice, this is done by measuring the CPU-time for both prior distributions for a set of different values of K, ranging between 100 and 10000. Figure 28 displays the CPU-time as a function of K, where the CPU-time for the old prior distribution is displayed by the yellow line, while the CPU-time for the new prior distribution is the blue line. We notice that the new prior distribution provide a considerable reduction in CPU-time. For $K = 10000$, we have that the CPU-time using the old prior distribution is approximately 10000 seconds, while the CPU-time using the new prior distribution is around 3600 seconds.

In this numerical example, the new prior distribution assumes that the sequential neighbourhood for the kth element of $\boldsymbol{x}_t$ is $\Lambda_k = \{k-L-1, k-L, k-1\}$, where L is the width and length of the grid associated to $\boldsymbol{x}_t$, i.e. $L^2 = K$. Recall that the precision matrix of $\boldsymbol{x}_t$ is denoted Q_t, and that the structure of Q_t is related to the sequential neighbourhood, as specified in Section 3.3. Since the sequential neighbourhood Λ_k changes with L, the bandwidth of Q_t, denoted m, also changes with L. In Appendix H, we prove that $m = L+1 = \sqrt{K}+1$. Figure 28 suggests that the new prior distribution is able to provide a considerable reduction in computational cost, even though the bandwidth m of Q_t increases as K increases.

As stated in Section 2.7, the computational complexity of Cholesky decomposing a $K \times K$-matrix with bandwidth m can be reduced to $\mathcal{O}(Km^2)$ by applying Algorithm 3. In this example we have that $m = \sqrt{K}+1$, which means that the computational complexity becomes $\mathcal{O}(K(\sqrt{K}+1)^2) = \mathcal{O}(K^2)$. Algorithm 4 then suggests that (61) can be computed with a computational complexity of $\mathcal{O}(K^2)$. This suggests that if every computation in HEnKF could be performed with a computational complexity of $\mathcal{O}(K^2)$ or less, the CPU-time would increase quadratically as a function of K. That is, we would expect the square root of the CPU-time to increase linearly as a function of K. Figure 29 visualizes the square root of the CPU-time for the new prior distribution as a function of K. The figure suggests that the square root of the CPU-time is almost linear in K, which again suggests that the computational complexity is approximately

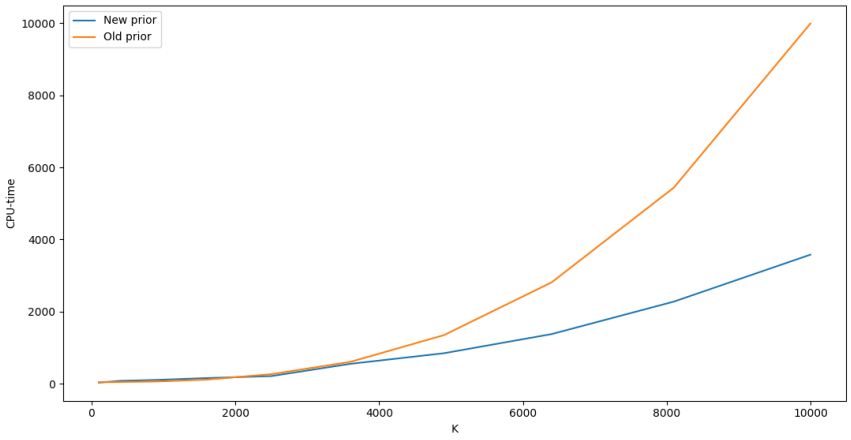


Figure 28: Comparison of CPU-times as a function of the dimension K. The yellow line represents the CPU-time in seconds for the old prior distribution as a function of K. The blue line represents the CPU-time for the new prior distribution.

$\mathcal{O}(K^2)$.

6 Closing remarks

In this report, we propose a prior distribution to be applied in the hierarchical ensemble Kalman filter (HEnKF), as an alternative to the prior distribution applied in Omre and Myrseth (2010). The proposed prior distribution is chosen with the intent to reduce the computational cost of HEnKF. The aim of this report is to compare the presented prior distribution to the prior distribution used in Omre and Myrseth (2010), by comparing the computational costs and their numerical results.

Two numerical examples are performed. In both examples, the proposed prior distribution provides a considerable reduction in computational demands. In the nonlinear example, the presented prior distribution produces reliable results that better adapts to the dimension of the problem, compared to the prior distribution applied in Omre and Myrseth (2010).

Further research on this topic could include applying the proposed prior distribution on larger models than the ones presented in this thesis, for example on large grids in three dimensions. In addition, the proposed prior distribution could be applied on state space models different from the ones presented in the numerical examples, in order to test if the computational complexity could be reduced as suggested in thesis. Also, the proposed prior distribution could be

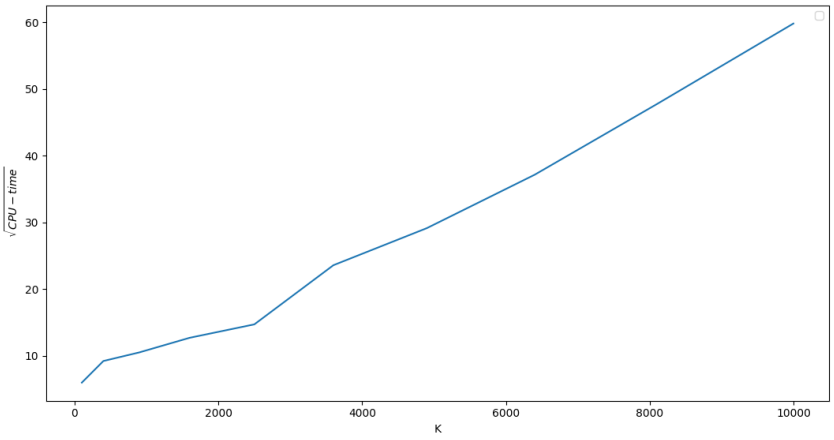


Figure 29: The blue line represents the square root of the CPU-time as a function of K, for the new prior distribution.

applied on more realistic examples, for example on real-world data, or on data that resembles reality.

References

Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T. (2002). "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking." *IEEE Transactions on Signal Processing*, **50**, 174–188.

Brockwell, P. J. and Davis, R. A. (1991). *Time Series: Theory and Methods*. 2nd ed. Springer.

Evensen, G. (1994). "Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics." *Journal of Geophysical research*, **99**, 10143–10162.

Gamerman, D. and Lopes, H. F. (2006). *Markov Chain Monte Carlo*. Chapman & Hall/CRC.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (2003). *Bayesian Data Analysis*. 2nd ed. Chapman & Hall/CRC.

Gordon, N. J., Salmond, D. J., and Smith, A. F. M. (1993). "Novel-approach to nonlinear non-Gaussian Bayesian state estimation." *IEE Proceedings-F*, **140**, 107–113.

Houtekamer, P. L. and Mitchell, H. L. (1997). "Data assimilation using an ensemble Kalman filter technique." *Monthly Weather Review*, **126**, 796–811.

Julier, S. and Uhlmann, J. (1997). "A new extension of the Kalman filter to nonlinear systems." *Proceeding of the Society of Photo-Optical Instrumentation Engineers*, **3068**, 182–193.

Kalman, R. E. (1960). "A new approach to linear filtering and prediction problems." *Journal of Basic Engineering*, 35–45.

Oliver, D. S. (1996). "On conditional simulation to inaccurate data." *Journal of Mathematical Geology*, **28**, 811–817.

Omre, H. and Myrseth, I. (2010). "Hierarchical ensemble Kalman filter." *SPE Journal*, **15**, 569–580.

— (2011). "The ensemble Kalman filter and related filters." In *Large-Scale Inverse Problems and Quantification of Uncertainty*, eds. L. Biegler, G. Biros, O. Ghattas, M. Heinkenschloss, D. Keyes, B. Mallick, Y. Marzouk, L. Tenorio, B. van Bloemen Waanders, and K. Willcox, chap. 11. John Wiley & Sons.

Rue, H. and Held, L. (2005). *Gaussian Markov Random Fields*. Chapman & Hall/CRC.

Shumway, R. H. and Stoffer, D. S. (2016). *Time Series Analysis and Its Applications*. 4th ed. Springer.

Woodbury, M. A. (1950). "Inverting modified matrices." *Memorandum report 42, Statistical Research group, Princeton University, Princeton, NJ*.

Appendices

A Proof of Theorem 1

In this section we prove Theorem 1. That is, if (3) and (9) hold, and the prior $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})$ is defined as in (11), the posterior distribution is

$$\boldsymbol{x}_t|\boldsymbol{d}_{0:t} \sim N(\boldsymbol{\mu}_{t|0:t}, \Sigma_{t|0:t}), \tag{131}$$

where $\boldsymbol{\mu}_{t|0:t}$ and $\Sigma_{t|0:t}$ are defined in (13) and (14), respectively.

Proof. By Bayes' rule, we have

$$f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}) = \frac{f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})f(\boldsymbol{d}_t|\boldsymbol{x}_t, \boldsymbol{d}_{0:t-1})}{f(\boldsymbol{d}_t|\boldsymbol{d}_{0:t-1})}. \tag{132}$$

We are able to simplify the second factor in the numerator. Since we assume that (3) holds, the observation $\boldsymbol{d}_t$ is conditionally independent of the previous observations $\boldsymbol{d}_{0:t-1}$ given $\boldsymbol{x}_t$. We have that

$$f(\boldsymbol{d}_t|\boldsymbol{x}_t, \boldsymbol{d}_{0:t-1}) = f(\boldsymbol{d}_t|\boldsymbol{x}_t). \tag{133}$$

In addition, since $f(\boldsymbol{d}_t|\boldsymbol{d}_{0:t-1})$ is not a function of $\boldsymbol{x}_t$, we have that

$$f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}) \propto f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}) f(\boldsymbol{d}_t|\boldsymbol{x}_t). \tag{134}$$

In the following, we prove that $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$ has the form of a Gaussian distribution, and hence calculating the normalizing constant $f(\boldsymbol{d}_t|\boldsymbol{d}_{0:t-1})$ in (132) is unnecessary. By combining the equation above with (11) and (9), we have that

$$f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}) \propto e^{-\frac{1}{2}(\boldsymbol{x}_t - \boldsymbol{\mu}_{t|0:t-1})^\intercal \Sigma_{t|0:t-1}^{-1} (\boldsymbol{x}_t - \boldsymbol{\mu}_{t|0:t-1})} e^{-\frac{1}{2}(\boldsymbol{d}_t - H\boldsymbol{x}_t)^\intercal \Sigma_d^{-1} (\boldsymbol{d}_t - H\boldsymbol{x}_t)}. \tag{135}$$

By expanding the parentheses and omitting the terms not containing $\boldsymbol{x}_t$, and we obtain

$$\begin{aligned} f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}) \propto & e^{-\frac{1}{2}(\boldsymbol{x}_t^\intercal \Sigma_{t|0:t-1}^{-1} \boldsymbol{x}_t - \boldsymbol{\mu}_{t|0:t-1}^\intercal \Sigma_{t|0:t-1}^{-1} \boldsymbol{x}_t - \boldsymbol{x}_t^\intercal \Sigma_{t|0:t-1}^{-1} \boldsymbol{\mu}_{t|0:t-1})} \\ & \cdot e^{-\frac{1}{2}((H\boldsymbol{x}_t)^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t - \boldsymbol{d}_t^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t - (H\boldsymbol{x}_t)^\intercal \Sigma_d^{-1} \boldsymbol{d}_t)}. \end{aligned} \tag{136}$$

All of the terms in the exponents are scalars. This means that the each term is equal to its own transpose, which enables us to transpose each term as we choose. That is,

$$\boldsymbol{\mu}_{t|0:t-1}^\intercal \Sigma_{t|0:t-1}^{-1} \boldsymbol{x}_t = (\boldsymbol{\mu}_{t|0:t-1}^\intercal \Sigma_{t|0:t-1}^{-1} \boldsymbol{x}_t)^\intercal = \boldsymbol{x}_t^\intercal \Sigma_{t|0:t-1}^{-1} \boldsymbol{\mu}_{t|0:t-1}, \tag{137}$$

$$\boldsymbol{d}_t^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t = (\boldsymbol{d}_t^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t)^\intercal = (H\boldsymbol{x}_t)^\intercal \Sigma_d^{-1} \boldsymbol{d}_t, \tag{138}$$

where we have that $\Sigma_{t|0:t-1}^{-1} = \Sigma_{t|0:t-1}^{-\intercal}$ and $\Sigma_d^{-1} = \Sigma_d^{-\intercal}$, since covariance matrices are symmetric. This means that (136) simplifies to

$$f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}) \propto e^{-\frac{1}{2}(\boldsymbol{x}_t^\intercal \Sigma_{t|0:t-1}^{-1} \boldsymbol{x}_t + (H\boldsymbol{x}_t)^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t - 2\boldsymbol{\mu}_{t|0:t-1}^\intercal \Sigma_{t|0:t-1}^{-1} \boldsymbol{x}_t - 2\boldsymbol{d}_t^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t)}. \tag{139}$$

By noting that $(H\boldsymbol{x}_t)^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t = \boldsymbol{x}_t^\intercal H^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t$, the first two terms in the exponent can be rewritten as

$$\boldsymbol{x}_t^\intercal \Sigma_{t|0:t-1}^{-1} \boldsymbol{x}_t + (H\boldsymbol{x}_t)^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t = \boldsymbol{x}_t^\intercal (\Sigma_{t|0:t-1}^{-1} + H^\intercal \Sigma_d^{-1} H)\boldsymbol{x}_t. \tag{140}$$

Similarly, the last two terms in the exponent in (139) can be written as

$$-2\boldsymbol{\mu}_{t|0:t-1}^\intercal \Sigma_{t|0:t-1}^{-1} \boldsymbol{x}_t - 2\boldsymbol{d}_t^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t = -2(\boldsymbol{\mu}_{t|0:t-1}^\intercal \Sigma_{t|0:t-1}^{-1} + \boldsymbol{d}_t^\intercal \Sigma_d^{-1} H)\boldsymbol{x}_t. \tag{141}$$

The enables us to write (139) as

$$f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}) \propto e^{-\frac{1}{2}(\boldsymbol{x}_t^\intercal (\Sigma_{t|0:t-1}^{-1} + H^\intercal \Sigma_d^{-1} H)\boldsymbol{x}_t - 2(\boldsymbol{\mu}_{t|0:t-1}^\intercal \Sigma_{t|0:t-1}^{-1} + \boldsymbol{d}_t^\intercal \Sigma_d^{-1} H)\boldsymbol{x}_t)} \tag{142}$$

We define the matrix C and the vector $\boldsymbol{D}$ as

$$C = \Sigma_{t|0:t-1}^{-1} + H^\intercal \Sigma_d^{-1} H, \tag{143}$$

$$\boldsymbol{D} = (\boldsymbol{\mu}_{t|0:t-1}^\intercal \Sigma_{t|0:t-1}^{-1} + \boldsymbol{d}_t^\intercal \Sigma_d^{-1} H)^\intercal. \tag{144}$$

Thus, (142) becomes

$$f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}) \propto e^{-\frac{1}{2}(\boldsymbol{x}_t^\intercal C \boldsymbol{x}_t - 2\boldsymbol{D}^\intercal \boldsymbol{x}_t)} = e^{-\frac{1}{2}\boldsymbol{x}_t^\intercal C \boldsymbol{x}_t + \boldsymbol{D}^\intercal \boldsymbol{x}_t}. \tag{145}$$

From Definition 2.2 in Rue and Held (2005), we have that

$$\boldsymbol{x}_t|\boldsymbol{d}_{0:t} \sim N_C(\boldsymbol{D}, C), \tag{146}$$

where N_C denotes the normal distribution with canonical parametrization. From Rue and Held (2005) we also have that

$$N_C(\boldsymbol{D}, C) \overset{d}{=} N(C^{-1}\boldsymbol{D}, C^{-1}), \tag{147}$$

where the "d" indicates that the two distributions are equal in distribution. By combining the two expressions above, we have

$$\boldsymbol{x}_t|\boldsymbol{d}_{0:t} \sim N_C(\boldsymbol{D}, C) \overset{d}{=} N(C^{-1}\boldsymbol{D}, C^{-1}). \tag{148}$$

That is,

$$\boldsymbol{\mu}_{t|0:t} = C^{-1}\boldsymbol{D} \tag{149}$$

$$\Sigma_{t|0:t} = C^{-1} \tag{150}$$

In the following, we first derive the expression for $\boldsymbol{\mu}_{t|0:t}$.

From (143) and (144), we have that

$$\boldsymbol{\mu}_{t|0:t} = C^{-1}\boldsymbol{D} = (\Sigma_{t|0:t-1}^{-1} + H^\intercal \Sigma_d^{-1} H)^{-1}(\boldsymbol{\mu}_{t|0:t-1}^\intercal \Sigma_{t|0:t-1}^{-1} + \boldsymbol{d}_t^\intercal \Sigma_d^{-1} H)^\intercal \tag{151}$$

By the Sherman-Morrison-Woodbury matrix identity (Woodbury, 1950), we have

$$\begin{aligned}&(\Sigma_{t|0:t-1}^{-1} + H^\intercal \Sigma_d^{-1} H)^{-1} \\ =&\Sigma_{t|0:t-1} - \Sigma_{t|0:t-1} H^\intercal (H \Sigma_{t|0:t-1} H^\intercal + \Sigma_d)^{-1} H \Sigma_{t|0:t-1}.\end{aligned} \tag{152}$$

By applying this matrix identity on (151), we have

$$\begin{aligned}\boldsymbol{\mu}_{t|0:t} =&(\Sigma_{t|0:t-1} - \Sigma_{t|0:t-1} H^\intercal (H \Sigma_{t|0:t-1} H^\intercal + \Sigma_d)^{-1} H \Sigma_{t|0:t-1}) \\ &\cdot (\boldsymbol{\mu}_{t|0:t-1}^\intercal \Sigma_{t|0:t-1}^{-1} + \boldsymbol{d}_t^\intercal \Sigma_d^{-1} H)^\intercal\end{aligned} \tag{153}$$

By expanding the parentheses, we have

$$\begin{aligned}&\boldsymbol{\mu}_{t|0:t} = \boldsymbol{\mu}_{t|0:t-1} - \Sigma_{t|0:t-1} H^\intercal (H \Sigma_{t|0:t-1} H^\intercal + \Sigma_d)^{-1} H \boldsymbol{\mu}_{t|0:t-1} \\ &+ \Sigma_{t|0:t-1} H^\intercal \Sigma_d^{-1} \boldsymbol{d}_t - \Sigma_{t|0:t-1} H^\intercal (H \Sigma_{t|0:t-1} H^\intercal + \Sigma_d)^{-1} H \Sigma_{t|0:t-1} H^\intercal \Sigma_d^{-1} \boldsymbol{d}_t\end{aligned} \tag{154}$$

By rewriting the first term on the second row as

$$\begin{aligned}&\Sigma_{t|0:t-1}H^\intercal\Sigma_d^{-1}\boldsymbol{d}_t\\=&\Sigma_{t|0:t-1}H^\intercal(H\Sigma_{t|0:t-1}H^\intercal+\Sigma_d)^{-1}(H\Sigma_{t|0:t-1}H^\intercal+\Sigma_d)\Sigma_d^{-1}\boldsymbol{d}_t,\end{aligned} \tag{155}$$

the expression above becomes

$$\begin{aligned}\boldsymbol{\mu}_{t|0:t}=&\boldsymbol{\mu}_{t|0:t-1}-\Sigma_{t|0:t-1}H^\intercal(H\Sigma_{t|0:t-1}H^\intercal+\Sigma_d)^{-1}H\boldsymbol{\mu}_{t|0:t-1}\\&+\Sigma_{t|0:t-1}H^\intercal(H\Sigma_{t|0:t-1}H^\intercal+\Sigma_d)^{-1}(H\Sigma_{t|0:t-1}H^\intercal+\Sigma_d)\Sigma_d^{-1}\boldsymbol{d}_t\\&-\Sigma_{t|0:t-1}H^\intercal(H\Sigma_{t|0:t-1}H^\intercal+\Sigma_d)^{-1}H\Sigma_{t|0:t-1}H^\intercal\Sigma_d^{-1}\boldsymbol{d}_t\end{aligned} \tag{156}$$

By expanding the factor $(H\Sigma_{t|0:t-1}H^\intercal+\Sigma_d)$ on the second row, we have

$$\begin{aligned}\boldsymbol{\mu}_{t|0:t}=&\boldsymbol{\mu}_{t|0:t-1}-\Sigma_{t|0:t-1}H^\intercal(H\Sigma_{t|0:t-1}H^\intercal+\Sigma_d)^{-1}H\boldsymbol{\mu}_{t|0:t-1}\\&+\Sigma_{t|0:t-1}H^\intercal(H\Sigma_{t|0:t-1}H^\intercal+\Sigma_d)^{-1}H\Sigma_{t|0:t-1}H^\intercal\Sigma_d^{-1}\boldsymbol{d}_t\\&+\Sigma_{t|0:t-1}H^\intercal(H\Sigma_{t|0:t-1}H^\intercal+\Sigma_d)^{-1}\Sigma_d\Sigma_d^{-1}\boldsymbol{d}_t\\&-\Sigma_{t|0:t-1}H^\intercal(H\Sigma_{t|0:t-1}H^\intercal+\Sigma_d)^{-1}H\Sigma_{t|0:t-1}H^\intercal\Sigma_d^{-1}\boldsymbol{d}_t\end{aligned} \tag{157}$$

The second and fourth row cancel. The factor $\Sigma_d\Sigma_d^{-1}$ on the third row also cancels, and we have

$$\begin{aligned}\boldsymbol{\mu}_{t|0:t}=&\boldsymbol{\mu}_{t|0:t-1}-\Sigma_{t|0:t-1}H^\intercal(H\Sigma_{t|0:t-1}H^\intercal+\Sigma_d)^{-1}H\boldsymbol{\mu}_{t|0:t-1}\\&+\Sigma_{t|0:t-1}H^\intercal(H\Sigma_{t|0:t-1}H^\intercal+\Sigma_d)^{-1}\boldsymbol{d}_t\end{aligned} \tag{158}$$

Recall from Theorem 1 that K_{KF} is defined as

$$K_{\text{KF}}=\Sigma_{t|0:t-1}H^\intercal(H\Sigma_{t|0:t-1}H^\intercal+\Sigma_d)^{-1}. \tag{159}$$

Hence, the expression for $\boldsymbol{\mu}_{t|0:t}$ becomes

$$\boldsymbol{\mu}_{t|0:t}=\boldsymbol{\mu}_{t|0:t-1}-K_{\text{KF}}(\boldsymbol{d}_t-H\boldsymbol{\mu}_{t|0:t-1}) \tag{160}$$

Which is what we set out to prove. We proceed to the expression for $\Sigma_{t|0:t}$

$$\Sigma_{t|0:t}=C^{-1}=(\Sigma_{t|0:t-1}^{-1}+H^\intercal\Sigma_d^{-1}H)^{-1} \tag{161}$$

By (152), we have

$$\Sigma_{t|0:t}=\Sigma_{t|0:t-1}-\Sigma_{t|0:t-1}H^\intercal(H\Sigma_{t|0:t-1}H^\intercal+\Sigma_d)^{-1}H\Sigma_{t|0:t-1} \tag{162}$$

From (159), we have that

$$\Sigma_{t|0:t}=\Sigma_{t|0:t-1}-K_{\text{KF}}H\Sigma_{t|0:t-1}=(I-K_{\text{KF}}H)\Sigma_{t|0:t-1}. \tag{163}$$

Which is what we set out to prove. This completes the proof of Theorem 1. □

B Proof of Theorem 2

In the following, we prove Theorem 2. That is, we prove that $f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t})$ is Gaussian with parameters defined in (20) and (21), when (1) and (7) hold, and when $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$ is Gaussian as defined in (18).

Proof. We have that

$$\begin{aligned} f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t}) &= \int f(\boldsymbol{x}_{t+1}, \boldsymbol{x}_t|\boldsymbol{d}_{0:t})d\boldsymbol{x}_t \\ &= \int f(\boldsymbol{x}_{t+1}|\boldsymbol{x}_t, \boldsymbol{d}_{0:t})f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})d\boldsymbol{x}_t \end{aligned} \tag{164}$$

Since the latent variables follows a first order Markov chain, (1), we have that $f(\boldsymbol{x}_{t+1}|\boldsymbol{x}_t, \boldsymbol{d}_{0:t}) = f(\boldsymbol{x}_{t+1}|\boldsymbol{x}_t)$. Thus the integral above becomes

$$f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t}) = \int f(\boldsymbol{x}_{t+1}|\boldsymbol{x}_t)f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})d\boldsymbol{x}_t. \tag{165}$$

The first factor in the integrand is given in (7), while the second factor is given in (18). The expression then becomes

$$\begin{aligned} f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t}) = \int &\frac{1}{(2\pi)^{K/2}|\Sigma_x|^{K/2}} e^{-\frac{1}{2}(\boldsymbol{x}_{t+1}-A_t\boldsymbol{x}_t)^\intercal \Sigma_x^{-1}(\boldsymbol{x}_{t+1}-A_t\boldsymbol{x}_t)} \\ \cdot &\frac{1}{(2\pi)^{K/2}|\Sigma_{t|0:t}|^{K/2}} e^{-\frac{1}{2}(\boldsymbol{x}_t-\boldsymbol{\mu}_{t|0:t})^\intercal \Sigma_{t|0:t}^{-1}(\boldsymbol{x}_t-\boldsymbol{\mu}_{t|0:t})} d\boldsymbol{x}_t \end{aligned} \tag{166}$$

We omit all factors not containing either $\boldsymbol{x}_{t+1}$ or $\boldsymbol{x}_t$, and put the factor only containing $\boldsymbol{x}_{t+1}$ outside the integral.

$$\begin{aligned} f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t}) \propto e^{-\frac{1}{2}\boldsymbol{x}_{t+1}^\intercal \Sigma_x^{-1}\boldsymbol{x}_{t+1}} \int &e^{-\frac{1}{2}\boldsymbol{x}_t^\intercal A_t^\intercal \Sigma_x^{-1} A_t \boldsymbol{x}_t} \\ \cdot &e^{-\frac{1}{2}\boldsymbol{x}_t^\intercal \Sigma_{t|0:t}^{-1}\boldsymbol{x}_t + \boldsymbol{x}_{t+1}^\intercal \Sigma_x^{-1} A_t \boldsymbol{x}_t + \boldsymbol{\mu}_{t|0:t}^\intercal \Sigma_{t|0:t}^{-1}\boldsymbol{x}_t} d\boldsymbol{x}_t. \end{aligned} \tag{167}$$

We define a matrix C and a vector $\boldsymbol{D}$

$$C = A_t^\intercal \Sigma_x^{-1} A_t + \Sigma_{t|0:t}^{-1}, \tag{168}$$

$$\boldsymbol{D} = (\boldsymbol{x}_{t+1}^\intercal \Sigma_x^{-1} A_t + \boldsymbol{\mu}_{t|0:t}^\intercal \Sigma_{t|0:t}^{-1})^\intercal. \tag{169}$$

This enables us to simplify the expression for $f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t})$

$$f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t}) \propto e^{-\frac{1}{2}\boldsymbol{x}_{t+1}^\intercal \Sigma_x^{-1}\boldsymbol{x}_{t+1}} \int e^{-\frac{1}{2}\boldsymbol{x}_t^\intercal C \boldsymbol{x}_t + \boldsymbol{D}^\intercal \boldsymbol{x}_t} d\boldsymbol{x}_t. \tag{170}$$

From Definition 2.2 in Rue and Held (2005), we have that the integrand has the form of a Gaussian pdf with canonical parametrization, i.e. $N_C(\boldsymbol{D}, C)$, where N_C denotes the normal distribution with canonical parametrization. From Rue and Held (2005) we also have that

$$N_C(\boldsymbol{D}, C) \overset{d}{=} N(C^{-1}\boldsymbol{D}, C^{-1}), \tag{171}$$

where the "d" indicates that the two distributions are equal in distribution. This enables us to rewrite the integrand as

$$f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t}) \propto e^{-\frac{1}{2}\boldsymbol{x}_{t+1}^\intercal \Sigma_x^{-1}\boldsymbol{x}_{t+1}+\frac{1}{2}\boldsymbol{D}^\intercal C^{-1}\boldsymbol{D}} \int e^{-\frac{1}{2}(\boldsymbol{x}_t - C^{-1}\boldsymbol{D})^\intercal C(\boldsymbol{x}_t - C^{-1}\boldsymbol{D})} d\boldsymbol{x}_t. \tag{172}$$

The integral simplifies to $(2\pi)^{K/2}|C|^{-1/2}$. This factor is omitted, since it is not a function of $\boldsymbol{x}_{t+1}$. By inserting the expressions for C and $\boldsymbol{D}$, we obtain

$$f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t}) \propto e^{-\frac{1}{2}\boldsymbol{x}_{t+1}^\intercal \Sigma_x^{-1}\boldsymbol{x}_{t+1}} \cdot e^{\frac{1}{2}(\boldsymbol{x}_{t+1}^\intercal \Sigma_x^{-1}A_t + \boldsymbol{\mu}_{t|0:t}^\intercal \Sigma_{t|0:t}^{-1})(A_t^\intercal \Sigma_x^{-1}A_t + \Sigma_{t|0:t}^{-1})^{-1}(\boldsymbol{x}_{t+1}^\intercal \Sigma_x^{-1}A_t + \boldsymbol{\mu}_{t|0:t}^\intercal \Sigma_{t|0:t}^{-1})^\intercal} \tag{173}$$

By multiplying the parentheses in the exponent and omitting the term not containing $\boldsymbol{x}_{t+1}$, we obtain

$$f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t}) \propto e^{-\frac{1}{2}\boldsymbol{x}_{t+1}^\intercal \Sigma_x^{-1}\boldsymbol{x}_{t+1}+\frac{1}{2}\boldsymbol{x}_{t+1}^\intercal \Sigma_x^{-1}A_t(A_t^\intercal \Sigma_x^{-1}A_t+\Sigma_{t|0:t}^{-1})^{-1}A_t^\intercal \Sigma_x^{-1}\boldsymbol{x}_{t+1}} \cdot e^{\boldsymbol{\mu}_{t|0:t}^\intercal \Sigma_{t|0:t}^{-1}(A_t^\intercal \Sigma_x^{-1}A_t+\Sigma_{t|0:t}^{-1})^{-1}A_t^\intercal \Sigma_x^{-1}\boldsymbol{x}_{t+1}} \tag{174}$$

where we utilize that $\Sigma_x^{-1} = \Sigma_x^{-\intercal}$. By the Sherman-Morrison-Woodbury matrix identity (Woodbury, 1950), we have that

$$\Sigma_x^{-1}A_t(A_t^\intercal \Sigma_x^{-1}A_t + \Sigma_{t|0:t}^{-1})^{-1}A_t^\intercal \Sigma_x^{-1} = \Sigma_x^{-1} - (\Sigma_x + A_t\Sigma_{t|0:t}A_t^\intercal)^{-1} \tag{175}$$

By replacing the left hand side with the right hand side in the expression for $f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t})$ above, we obtain

$$f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t}) \propto e^{-\frac{1}{2}\boldsymbol{x}_{t+1}^\intercal \Sigma_x^{-1}\boldsymbol{x}_{t+1}+\frac{1}{2}\boldsymbol{x}_{t+1}^\intercal (\Sigma_x^{-1}-(\Sigma_x+A_t\Sigma_{t|0:t}A_t^\intercal)^{-1})\boldsymbol{x}_{t+1}} \cdot e^{\boldsymbol{\mu}_{t|0:t}^\intercal \Sigma_{t|0:t}^{-1}(A_t^\intercal \Sigma_x^{-1}A_t+\Sigma_{t|0:t}^{-1})^{-1}A_t^\intercal \Sigma_x^{-1}\boldsymbol{x}_{t+1}} \tag{176}$$

Two of the terms in the exponent on the first line cancel, and we obtain

$$f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t}) \propto e^{-\frac{1}{2}\boldsymbol{x}_{t+1}^\intercal (\Sigma_x+A_t\Sigma_{t|0:t}A_t^\intercal)^{-1}\boldsymbol{x}_{t+1}} \cdot e^{\boldsymbol{\mu}_{t|0:t}^\intercal \Sigma_{t|0:t}^{-1}(A_t^\intercal \Sigma_x^{-1}A_t+\Sigma_{t|0:t}^{-1})^{-1}A_t^\intercal \Sigma_x^{-1}\boldsymbol{x}_{t+1}} \tag{177}$$

We define a matrix E and a vector $\boldsymbol{F}$

$$E = (\Sigma_x + A_t\Sigma_{t|0:t}A_t^\intercal)^{-1} \tag{178}$$

$$\boldsymbol{F} = (\boldsymbol{\mu}_{t|0:t}^\intercal \Sigma_{t|0:t}^{-1}(A_t^\intercal \Sigma_x^{-1}A_t + \Sigma_{t|0:t}^{-1})^{-1}A_t^\intercal \Sigma_x^{-1})^\intercal \tag{179}$$

Thus, we have

$$f(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t}) \propto e^{-\frac{1}{2}\boldsymbol{x}_{t+1}^\intercal E\boldsymbol{x}_{t+1}+\boldsymbol{F}^\intercal \boldsymbol{x}_{t+1}} \tag{180}$$

From Definition 2.2 in Rue and Held (2005), we have that

$$\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t} \sim N_C(\boldsymbol{F}, E), \tag{181}$$

where N_C denotes the normal distribution with canonical parametrization. From Rue and Held (2005) we also have that

$$N_C(\boldsymbol{F}, E) \overset{d}{=} N(E^{-1}\boldsymbol{F}, E^{-1}), \tag{182}$$

where the "d" indicates that the two distributions are equal in distribution. By combining the two expressions above, we have

$$\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t} \sim N_C(\boldsymbol{F}, E) = N(E^{-1}\boldsymbol{F}, E^{-1}). \tag{183}$$

That is,

$$\boldsymbol{\mu}_{t+1|0:t} = E^{-1}\boldsymbol{F} \tag{184}$$

$$\Sigma_{t+1|0:t} = E^{-1} \tag{185}$$

In the following, we first derive the expression for $\boldsymbol{\mu}_{t+1|0:t}$.

By inserting the expressions for E and $\boldsymbol{F}$, we have that

$$\begin{aligned}
&\boldsymbol{\mu}_{t+1|0:t} = E^{-1}\boldsymbol{F} \\
&= (\Sigma_x + A_t\Sigma_{t|0:t}A_t^\intercal)(\boldsymbol{\mu}_{t|0:t}^\intercal\Sigma_{t|0:t}^{-1}(A_t^\intercal\Sigma_x^{-1}A_t + \Sigma_{t|0:t}^{-1})^{-1}A_t^\intercal\Sigma_x^{-1})^\intercal \\
&= (\Sigma_x + A_t\Sigma_{t|0:t}A_t^\intercal)\Sigma_x^{-1}A_t(A_t^\intercal\Sigma_x^{-1}A_t + \Sigma_{t|0:t}^{-1})^{-1}\Sigma_{t|0:t}^{-1}\boldsymbol{\mu}_{t|0:t},
\end{aligned} \tag{186}$$

where we utilized that

$$\Sigma_{t|0:t}^{-1} = \Sigma_{t|0:t}^{-\intercal}, \tag{187}$$

$$(A_t^\intercal\Sigma_x^{-1}A_t + \Sigma_{t|0:t}^{-1})^{-1} = (A_t^\intercal\Sigma_x^{-1}A_t + \Sigma_{t|0:t}^{-1})^{-\intercal}. \tag{188}$$

By applying the Sherman-Morrison-Woodbury matrix identity (Woodbury, 1950), we have

$$\begin{aligned}
&(A_t^\intercal\Sigma_x^{-1}A_t + \Sigma_{t|0:t}^{-1})^{-1} \\
&= \Sigma_{t|0:t} - \Sigma_{t|0:t}A_t^\intercal(A_t\Sigma_{t|0:t}A_t^\intercal + \Sigma_x)^{-1}A_t\Sigma_{t|0:t}
\end{aligned} \tag{189}$$

By replacing the right hand side with the left hand side in the expression above, we have

$$\begin{aligned}
\boldsymbol{\mu}_{t+1|0:t} =&(\Sigma_x + A_t\Sigma_{t|0:t}A_t^\intercal)\Sigma_x^{-1}A_t \\
&\cdot(\Sigma_{t|0:t} - \Sigma_{t|0:t}A_t^\intercal(A_t\Sigma_{t|0:t}A_t^\intercal + \Sigma_x)^{-1}A_t\Sigma_{t|0:t}) \\
&\cdot\Sigma_{t|0:t}^{-1}\boldsymbol{\mu}_{t|0:t}.
\end{aligned} \tag{190}$$

We see that $\Sigma_{t|0:t}$ cancels out

$$\begin{aligned}
\boldsymbol{\mu}_{t+1|0:t} =&(\Sigma_x + A_t\Sigma_{t|0:t}A_t^\intercal)\Sigma_x^{-1}A_t \\
&\cdot(I - \Sigma_{t|0:t}A_t^\intercal(A_t\Sigma_{t|0:t}A_t^\intercal + \Sigma_x)^{-1}A_t)\boldsymbol{\mu}_{t|0:t}.
\end{aligned} \tag{191}$$

Further calculations yield

$$\begin{aligned}
\boldsymbol{\mu}_{t+1|0:t} =&(A_t + A_t\Sigma_{t|0:t}A_t^\intercal\Sigma_x^{-1}A_t) \\
&\cdot(I - \Sigma_{t|0:t}A_t^\intercal(A_t\Sigma_{t|0:t}A_t^\intercal + \Sigma_x)^{-1}A_t)\boldsymbol{\mu}_{t|0:t}.
\end{aligned} \tag{192}$$

By expanding the parenthesis on the first line and the outer parenthesis on the second line, we have

$$\begin{aligned}\boldsymbol{\mu}_{t+1|0:t} &= (A_t + A_t\Sigma_{t|0:t}A_t^\intercal\Sigma_x^{-1}A_t)\boldsymbol{\mu}_{t|0:t} \\ &- (A_t + A_t\Sigma_{t|0:t}A_t^\intercal\Sigma_x^{-1}A_t)\Sigma_{t|0:t}A_t^\intercal(A\Sigma_{t|0:t}A_t^\intercal + \Sigma_x)^{-1}A_t\boldsymbol{\mu}_{t|0:t}.\end{aligned} \tag{193}$$

We have that

$$\begin{aligned}&(A_t + A_t\Sigma_{t|0:t}A_t^\intercal\Sigma_x^{-1}A_t)\Sigma_{t|0:t}A_t^\intercal \\ =&(A_t\Sigma_{t|0:t}A_t^\intercal + A_t\Sigma_{t|0:t}A_t^\intercal\Sigma_x^{-1}A_t\Sigma_{t|0:t}A_t^\intercal) \\ =&A_t\Sigma_{t|0:t}A_t^\intercal(I + \Sigma_x^{-1}A_t\Sigma_{t|0:t}A_t^\intercal).\end{aligned} \tag{194}$$

Thus, by inserting this expression on the second line in (193), we have

$$\begin{aligned}\boldsymbol{\mu}_{t+1|0:t} &= A_t\boldsymbol{\mu}_{t|0:t} + A_t\Sigma_{t|0:t}A_t^\intercal\Sigma_x^{-1}A_t\boldsymbol{\mu}_{t|0:t} \\ &- A_t\Sigma_{t|0:t}A_t^\intercal(I + \Sigma_x^{-1}A_t\Sigma_{t|0:t}A_t^\intercal)(A_t\Sigma_{t|0:t}A_t^\intercal + \Sigma_x)^{-1}A_t\boldsymbol{\mu}_{t|0:t}.\end{aligned} \tag{195}$$

By multiplying with Σ_x and Σ_x^{-1} on the second line, we obtain

$$\begin{aligned}\boldsymbol{\mu}_{t+1|0:t} &= A_t\boldsymbol{\mu}_{t|0:t} + A_t\Sigma_{t|0:t}A_t^\intercal\Sigma_x^{-1}A_t\boldsymbol{\mu}_{t|0:t} \\ &- A_t\Sigma_{t|0:t}A_t^\intercal\Sigma_x^{-1}(\Sigma_x + A_t\Sigma_{t|0:t}A_t^\intercal)(A_t\Sigma_{t|0:t}A_t^\intercal + \Sigma_x)^{-1}A_t\boldsymbol{\mu}_{t|0:t}\end{aligned} \tag{196}$$

The parentheses on the second line cancel out

$$\begin{aligned}\boldsymbol{\mu}_{t+1|0:t} =&A_t\boldsymbol{\mu}_{t|0:t} + A_t\Sigma_{t|0:t}A_t^\intercal\Sigma_x^{-1}A_t\boldsymbol{\mu}_{t|0:t} \\ &- A_t\Sigma_{t|0:t}A_t^\intercal\Sigma_x^{-1}A_t\boldsymbol{\mu}_{t|0:t}\end{aligned} \tag{197}$$

The last two terms cancel out, and we have

$$\boldsymbol{\mu}_{t+1|0:t} = A_t\boldsymbol{\mu}_{t|0:t} \tag{198}$$

Which is what we set out to prove. We now consider $\Sigma_{t+1|0:t}$.

The expression for $\Sigma_{t+1|0:t}$ is

$$\Sigma_{t+1|0:t} = E^{-1} = \Sigma_x + A_t\Sigma_{t|0:t}A_t^\intercal \tag{199}$$

Thus, we have that

$$\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t} \sim N(A_t\boldsymbol{\mu}_{t|0:t}, \Sigma_x + A_t\Sigma_{t|0:t}A_t^\intercal), \tag{200}$$

which completes the proof. □

C Derivation of filter distribution in EnKF

Our objective is to derive the expressions for the parameters of $\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t})$, namely $\hat{\boldsymbol{\mu}}_{t|0:t}$ and $\hat{\Sigma}_{t|0:t}$, presented in (47) and (48)

$$\hat{\boldsymbol{\mu}}_{t|0:t} = \hat{\boldsymbol{\mu}}_{t|0:t-1} + K_{\text{EnKF}}(\boldsymbol{d}_t - H\hat{\boldsymbol{\mu}}_{t|0:t-1}), \tag{201}$$

$$\hat{\Sigma}_{t|0:t} = (I - K_{\text{EnKF}}H)\hat{\Sigma}_{t|0:t-1}, \tag{202}$$

where $\hat{\boldsymbol{\mu}}_{t|0:t-1}$ and $\hat{\Sigma}_{t|0:t-1}$ are the distribution parameters of $\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}, \chi_t)$ and $K_{\text{EnKF}} = \hat{\Sigma}_{t|0:t-1}H^\intercal(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)^{-1}$.

By Bayes' rule, we have

$$\hat{f}(\boldsymbol{x}_{t+1}|\boldsymbol{d}_{0:t}, \chi_t) = \frac{\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}, \chi_t) f(\boldsymbol{d}_t|\boldsymbol{x}_t, \boldsymbol{d}_{0:t-1})}{f(\boldsymbol{d}_t|\boldsymbol{d}_{0:t-1})}. \tag{203}$$

We are able to simplify the second factor in the numerator. Since we assume that (3) holds, the observation $\boldsymbol{d}_t$ is conditionally independent of the previous observations $\boldsymbol{d}_{0:t-1}$ given $\boldsymbol{x}_t$. We have that

$$f(\boldsymbol{d}_t|\boldsymbol{x}_t, \boldsymbol{d}_{0:t-1}) = f(\boldsymbol{d}_t|\boldsymbol{x}_t). \tag{204}$$

In addition, since $f(\boldsymbol{d}_t|\boldsymbol{d}_{0:t-1})$ is not a function of $\boldsymbol{x}_t$, we have that

$$\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}, \chi_t) \propto \hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}, \chi_t) f(\boldsymbol{d}_t|\boldsymbol{x}_t). \tag{205}$$

In the following, we show that $\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}, \chi_t)$ has the form of a Gaussian distribution, and hence calculating the normalizing constant $f(\boldsymbol{d}_t|\boldsymbol{d}_{0:t-1})$ in (203) is unnecessary. We have that $\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}, \chi_t)$ is normally distributed with mean $\hat{\boldsymbol{\mu}}_{t|0:t-1}$ and covariance matrix $\hat{\Sigma}_{t|0:t-1}$, and from (9) we have that $\boldsymbol{d}_t|\boldsymbol{x}_t \sim N(H\boldsymbol{x}_t, \Sigma_d)$. Thus

$$\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}, \chi_t) \propto e^{-\frac{1}{2}(\boldsymbol{x}_t - \hat{\boldsymbol{\mu}}_{t|0:t-1})^\intercal \hat{\Sigma}^{-1}_{t|0:t-1}(\boldsymbol{x}_t - \hat{\boldsymbol{\mu}}_{t|0:t-1})} e^{-\frac{1}{2}(\boldsymbol{d}_t - H\boldsymbol{x}_t)^\intercal \Sigma_d^{-1}(\boldsymbol{d}_t - H\boldsymbol{x}_t)} \tag{206}$$

By expanding the parentheses and omitting the terms not containing $\boldsymbol{x}_t$, we obtain

$$\begin{aligned}\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}, \chi_t) \propto & e^{-\frac{1}{2}(\boldsymbol{x}_t^\intercal \hat{\Sigma}^{-1}_{t|0:t-1}\boldsymbol{x}_t - \hat{\boldsymbol{\mu}}^\intercal_{t|0:t-1}\hat{\Sigma}^{-1}_{t|0:t-1}\boldsymbol{x}_t - \boldsymbol{x}_t^\intercal \hat{\Sigma}^{-1}_{t|0:t-1}\hat{\boldsymbol{\mu}}_{t|0:t-1})} \\ & \cdot e^{-\frac{1}{2}((H\boldsymbol{x}_t)^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t - \boldsymbol{d}_t^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t - (H\boldsymbol{x}_t)^\intercal \Sigma_d^{-1}\boldsymbol{d}_t)}\end{aligned} \tag{207}$$

All of the terms in the exponents are scalars. This means that the each term is equal to its own transpose, which enables us to transpose each term as we choose. Thus,

$$\hat{\boldsymbol{\mu}}^\intercal_{t|0:t-1}\hat{\Sigma}^{-1}_{t|0:t-1}\boldsymbol{x}_t = (\hat{\boldsymbol{\mu}}^\intercal_{t|0:t-1}\hat{\Sigma}^{-1}_{t|0:t-1}\boldsymbol{x}_t)^\intercal = \boldsymbol{x}_t^\intercal \hat{\Sigma}^{-1}_{t|0:t-1}\hat{\boldsymbol{\mu}}_{t|0:t-1}, \tag{208}$$

$$\boldsymbol{d}_t^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t = (\boldsymbol{d}_t^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t)^\intercal = (H\boldsymbol{x}_t)^\intercal \Sigma_d^{-1}\boldsymbol{d}_t, \tag{209}$$

where we have that $\hat{\Sigma}^{-1}_{t|0:t-1} = \hat{\Sigma}^{-\intercal}_{t|0:t-1}$ and $\Sigma_d^{-1} = \Sigma_d^{-\intercal}$, since covariance matrices are symmetric. This means that (207) simplifies to

$$\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t}, \chi_t) \propto e^{-\frac{1}{2}(\boldsymbol{x}_t^\intercal \hat{\Sigma}^{-1}_{t|0:t-1}\boldsymbol{x}_t + (H\boldsymbol{x}_t)^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t - 2\hat{\boldsymbol{\mu}}^\intercal_{t|0:t-1}\hat{\Sigma}^{-1}_{t|0:t-1}\boldsymbol{x}_t - 2\boldsymbol{d}_t^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t)} \tag{210}$$

By noting that $(H\boldsymbol{x}_t)^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t = \boldsymbol{x}_t^\intercal H^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t$, the first two terms in the exponent can be rewritten as

$$\boldsymbol{x}_t^\intercal \hat{\Sigma}^{-1}_{t|0:t-1}\boldsymbol{x}_t + (H\boldsymbol{x}_t)^\intercal \Sigma_d^{-1} H\boldsymbol{x}_t = \boldsymbol{x}_t^\intercal (\hat{\Sigma}^{-1}_{t|0:t-1} + H^\intercal \Sigma_d^{-1} H)\boldsymbol{x}_t. \tag{211}$$

Similarly, the last two terms in the exponent in (210) can be written as

$$-2\hat{\boldsymbol{\mu}}_{t|0:t-1}^{\intercal}\hat{\Sigma}_{t|0:t-1}^{-1}\boldsymbol{x}_t - 2\boldsymbol{d}_t^{\intercal}\Sigma_d^{-1}H\boldsymbol{x}_t = -2(\hat{\boldsymbol{\mu}}_{t|0:t-1}^{\intercal}\hat{\Sigma}_{t|0:t-1}^{-1} + \boldsymbol{d}_t^{\intercal}\Sigma_d^{-1}H)\boldsymbol{x}_t. \quad (212)$$

The enables us to write (210) as

$$\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t},\chi_t) \propto e^{-\frac{1}{2}(\boldsymbol{x}_t^{\intercal}(\hat{\Sigma}_{t|0:t-1}^{-1}+H^{\intercal}\Sigma_d^{-1}H)\boldsymbol{x}_t - 2(\hat{\boldsymbol{\mu}}_{t|0:t-1}^{\intercal}\hat{\Sigma}_{t|0:t-1}^{-1}+\boldsymbol{d}_t^{\intercal}\Sigma_d^{-1}H)\boldsymbol{x}_t)} \quad (213)$$

We define the matrix C and the vector $\boldsymbol{D}$ as

$$C = \hat{\Sigma}_{t|0:t-1}^{-1} + H^{\intercal}\Sigma_d^{-1}H, \quad (214)$$

$$\boldsymbol{D} = (\hat{\boldsymbol{\mu}}_{t|0:t-1}^{\intercal}\hat{\Sigma}_{t|0:t-1}^{-1} + \boldsymbol{d}_t^{\intercal}\Sigma_d^{-1}H)^{\intercal}. \quad (215)$$

Thus, (213) becomes

$$\hat{f}(\boldsymbol{x}_t|\boldsymbol{d}_{0:t},\chi_t) \propto e^{-\frac{1}{2}(\boldsymbol{x}_t^{\intercal}C\boldsymbol{x}_t - 2\boldsymbol{D}^{\intercal}\boldsymbol{x}_t)} = e^{-\frac{1}{2}\boldsymbol{x}_t^{\intercal}C\boldsymbol{x}_t + \boldsymbol{D}^{\intercal}\boldsymbol{x}_t}. \quad (216)$$

From Definition 2.2 in Rue and Held (2005), we have that

$$\boldsymbol{x}_t|\boldsymbol{d}_{0:t} \sim N_C(\boldsymbol{D}, C), \quad (217)$$

where N_C denotes the normal distribution with canonical parametrization. From Rue and Held (2005) we also have that

$$N_C(\boldsymbol{D}, C) \overset{d}{=} N(C^{-1}\boldsymbol{D}, C^{-1}). \quad (218)$$

where the "d" indicates that the two distributions are equal in distribution. By combining the two expressions above, we have

$$\boldsymbol{x}_t|\boldsymbol{d}_{0:t} \sim N_C(\boldsymbol{D}, C) \overset{d}{=} N(C^{-1}\boldsymbol{D}, C^{-1}). \quad (219)$$

That is,

$$\hat{\boldsymbol{\mu}}_{t|0:t} = C^{-1}\boldsymbol{D} \quad (220)$$

$$\hat{\Sigma}_{t|0:t} = C^{-1} \quad (221)$$

In the following, we first derive the expression for $\hat{\boldsymbol{\mu}}_{t|0:t}$.

From (214) and (215), we have that

$$\hat{\boldsymbol{\mu}}_{t|0:t} = C^{-1}\boldsymbol{D} = (\hat{\Sigma}_{t|0:t-1}^{-1} + H^{\intercal}\Sigma_d^{-1}H)^{-1}(\hat{\boldsymbol{\mu}}_{t|0:t-1}^{\intercal}\hat{\Sigma}_{t|0:t-1}^{-1} + \boldsymbol{d}_t^{\intercal}\Sigma_d^{-1}H)^{\intercal} \quad (222)$$

By the Sherman-Morrison-Woodbury matrix identity (Woodbury, 1950), we have that

$$\begin{aligned}&(\hat{\Sigma}_{t|0:t-1}^{-1} + H^{\intercal}\Sigma_d^{-1}H)^{-1}\\ =&\hat{\Sigma}_{t|0:t-1} - \hat{\Sigma}_{t|0:t-1}H^{\intercal}(H\hat{\Sigma}_{t|0:t-1}H^{\intercal} + \Sigma_d)^{-1}H\hat{\Sigma}_{t|0:t-1}.\end{aligned} \quad (223)$$

By applying this matrix identity on (222), we have

$$\begin{aligned}\hat{\boldsymbol{\mu}}_{t|0:t} =&(\hat{\Sigma}_{t|0:t-1} - \hat{\Sigma}_{t|0:t-1}H^\intercal(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)^{-1}H\hat{\Sigma}_{t|0:t-1}) \\ &\cdot (\hat{\boldsymbol{\mu}}_{t|0:t-1}^\intercal\hat{\Sigma}_{t|0:t-1}^{-1} + \boldsymbol{d}_t^\intercal\Sigma_d^{-1}H)^\intercal\end{aligned} \tag{224}$$

By expanding the parentheses, we have

$$\begin{aligned}&\hat{\boldsymbol{\mu}}_{t|0:t} = \hat{\boldsymbol{\mu}}_{t|0:t-1} - \hat{\Sigma}_{t|0:t-1}H^\intercal(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)^{-1}H\hat{\boldsymbol{\mu}}_{t|0:t-1} \\ &+ \hat{\Sigma}_{t|0:t-1}H^\intercal\Sigma_d^{-1}\boldsymbol{d}_t - \hat{\Sigma}_{t|0:t-1}H^\intercal(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)^{-1}H\hat{\Sigma}_{t|0:t-1}H^\intercal\Sigma_d^{-1}\boldsymbol{d}_t.\end{aligned} \tag{225}$$

By rewriting the first term on the second row as

$$\begin{aligned}&\hat{\Sigma}_{t|0:t-1}H^\intercal\Sigma_d^{-1}\boldsymbol{d}_t \\ =&\hat{\Sigma}_{t|0:t-1}H^\intercal(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)^{-1}(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)\Sigma_d^{-1}\boldsymbol{d}_t,\end{aligned} \tag{226}$$

the expression above becomes

$$\begin{aligned}&\hat{\boldsymbol{\mu}}_{t|0:t} = \hat{\boldsymbol{\mu}}_{t|0:t-1} - \hat{\Sigma}_{t|0:t-1}H^\intercal(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)^{-1}H\hat{\boldsymbol{\mu}}_{t|0:t-1} \\ &+ \hat{\Sigma}_{t|0:t-1}H^\intercal(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)^{-1}(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)\Sigma_d^{-1}\boldsymbol{d}_t \\ &- \hat{\Sigma}_{t|0:t-1}H^\intercal(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)^{-1}H\hat{\Sigma}_{t|0:t-1}H^\intercal\Sigma_d^{-1}\boldsymbol{d}_t.\end{aligned} \tag{227}$$

By expanding the factor $(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)$ on the second row, we have

$$\begin{aligned}&\hat{\boldsymbol{\mu}}_{t|0:t} = \hat{\boldsymbol{\mu}}_{t|0:t-1} - \hat{\Sigma}_{t|0:t-1}H^\intercal(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)^{-1}H\hat{\boldsymbol{\mu}}_{t|0:t-1} \\ &+ \hat{\Sigma}_{t|0:t-1}H^\intercal(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)^{-1}H\hat{\Sigma}_{t|0:t-1}H^\intercal\Sigma_d^{-1}\boldsymbol{d}_t \\ &+ \hat{\Sigma}_{t|0:t-1}H^\intercal(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)^{-1}\Sigma_d\Sigma_d^{-1}\boldsymbol{d}_t \\ &- \hat{\Sigma}_{t|0:t-1}H^\intercal(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)^{-1}H\hat{\Sigma}_{t|0:t-1}H^\intercal\Sigma_d^{-1}\boldsymbol{d}_t.\end{aligned} \tag{228}$$

The second and fourth row cancel. The factor $\Sigma_d\Sigma_d^{-1}$ on the third row also cancels, and we have

$$\begin{aligned}&\hat{\boldsymbol{\mu}}_{t|0:t} = \hat{\boldsymbol{\mu}}_{t|0:t-1} - \hat{\Sigma}_{t|0:t-1}H^\intercal(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)^{-1}H\hat{\boldsymbol{\mu}}_{t|0:t-1} \\ &+ \hat{\Sigma}_{t|0:t-1}H^\intercal(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)^{-1}\boldsymbol{d}_t.\end{aligned} \tag{229}$$

Recall that K_{EnKF} is defined as

$$K_{\text{EnKF}} = \hat{\Sigma}_{t|0:t-1}H^\intercal(H\hat{\Sigma}_{t|0:t-1}H^\intercal + \Sigma_d)^{-1}. \tag{230}$$

Hence, the expression for $\hat{\boldsymbol{\mu}}_{t|0:t}$ becomes

$$\hat{\boldsymbol{\mu}}_{t|0:t} = \hat{\boldsymbol{\mu}}_{t|0:t-1} - K_{\text{EnKF}}(\boldsymbol{d}_t - H\hat{\boldsymbol{\mu}}_{t|0:t-1}). \tag{231}$$

Which is what we set out to prove. We proceed to the expression for $\hat{\Sigma}_{t|0:t}$

$$\hat{\Sigma}_{t|0:t} = C^{-1} = (\hat{\Sigma}_{t|0:t-1}^{-1} + H^\intercal\Sigma_d^{-1}H)^{-1}. \tag{232}$$

By (223), we have

$$\hat{\Sigma}_{t|0:t} = \hat{\Sigma}_{t|0:t-1} - \hat{\Sigma}_{t|0:t-1} H^\intercal (H \hat{\Sigma}_{t|0:t-1} H^\intercal + \Sigma_d)^{-1} H \hat{\Sigma}_{t|0:t-1}. \tag{233}$$

By using (230)

$$\hat{\Sigma}_{t|0:t} = \hat{\Sigma}_{t|0:t-1} - K_{\mathrm{EnKF}} H \hat{\Sigma}_{t|0:t-1} = (I - K_{\mathrm{KF}} H) \hat{\Sigma}_{t|0:t-1}. \tag{234}$$

This expression coincides with the expression in (202). This completes the derivation of the expressions for the parameters of $\hat{f}(\boldsymbol{x}_t | \boldsymbol{d}_{0:t})$.

D Proof of structure in Cholesky factorization

Section 2.7 states that the Cholesky factorization of a band matrix with bandwidth m is lower-triangular with nonzero entries located only on the main diagonal and the m first lower diagonals, see (77). This section proves that this is the case for all band matrices with bandwidth m.

We have that Q is a band matrix with bandwidth m. That is, we only have nonzero entries on the main diagonal and the m first upper and lower diagonals. This can be formulated as

$$Q^{i,k} = 0 \quad \forall i, k : |i - k| > m, \tag{235}$$

where $Q^{i,k}$ is the (i,k)th entry of Q. Section 2.7 states that Q is positive definite and symmetric. Hence, there exists a unique lower triangular matrix L such that $Q = LL^\intercal$. We want to prove that

$$L^{i,k} = 0 \quad \forall i, k : |i - k| > m. \tag{236}$$

Proof. This is trivial for the entries above the main diagonal, since L is lower triangular, and hence we can omit the absolute value around $i - k$. That is, we assume $i > k$ without loss of generality.

$$L^{i,k} = 0 \quad \forall i, k : i - k > m. \tag{237}$$

We prove this expression for an arbitrary row of L, namely the ith row. Without loss of generality, we assume that $i > m$. Rewriting (237) yields

$$L^{i,k} = 0 \quad \forall k : k < i - m. \tag{238}$$

That is, we want to prove that $L^{i,k} = 0$ for all k such that $1 \leq k < i - m$. In order to prove this, we make use of the following formula for calculating the off-diagonal entries of L

$$L^{i,k} = \frac{1}{L^{k,k}} (Q^{i,k} - \sum_{s=1}^{k-1} L^{i,s} L^{k,s}), \quad k < i. \tag{239}$$

In order to prove that $L^{i,k} = 0$ for all k such that $1 < k \leq i - m$, we first prove that $Q^{i,k} = 0$ for $k < i - m$.

From (235), we have that $Q^{i,k} = 0$ when $|i-k| > m$. Since we consider the entries below the diagonal, we have $i > k$. This entails that we can remove the absolute value around $|i-k|$. That is, $Q^{i,k} = 0$ for $i - k > m$ or equivalently $k < i - m$. Hence, in the case when $k < i - m$, we are able to simplify (239) to

$$L^{i,k} = \frac{1}{L^{k,k}}(-\sum_{s=1}^{k-1} L^{i,s}L^{k,s}), \quad k < i - m. \tag{240}$$

In order to prove that $L^{i,k} = 0$, we structure the proof as a "proof by induction", by first proving that $L^{i,1} = 0$, and then proving that $L^{i,k} = 0$ if $L^{i,1} = \cdots = L^{i,k-2} = L^{i,k-1} = 0$. Proving that $L^{i,1} = 0$ is trivial, since the summation in (240) goes from $s = 1$ to $k - 1$, and hence the sum is zero when $k = 1$.

We now assume that $L^{i,1} = \cdots = L^{i,k-2} = L^{i,k-1} = 0$, and we want to prove that $L^{i,k} = 0$ for $k < i - m$. By writing out the summation in (240), we have

$$L^{i,k} = \frac{1}{L^{k,k}}(-L^{i,1}L^{k,1} - L^{i,2}L^{k,2} \cdots - L^{i,k-1}L^{k,k-1}), \quad k < i - m. \tag{241}$$

Since we assume $L^{i,1} = \cdots = L^{i,k-2} = L^{i,k-1} = 0$, we have that the second factor in every term is zero. Hence $L^{i,k} = 0$. This proves that (236) holds, which completes the proof. □

E Proof of Theorem 4

In the following, we prove Theorem 4 presented in Section 2.6. That is, we prove that: Assume $1 \leq i < k \leq K$. We then have

$$i \notin \Lambda_k \text{ and } \nexists s \in \{k+1, \ldots, K\} : i, k \in \Lambda_s \iff x^i \perp x^k | \boldsymbol{x}^{-\{i,k\}}. \tag{242}$$

Proof. From Theorem 2.1 in Rue and Held (2005), we have that

$$x^i \perp x^k | \boldsymbol{x}^{-\{i,k\}} \iff f(\boldsymbol{x}) = g(x^{-\{i\}})h(x^{-\{k\}}), \tag{243}$$

for some functions g and h. Since we assume that $\boldsymbol{x}$ is a GMRF, the pdf of $\boldsymbol{x}$ can be written as

$$f(\boldsymbol{x}) = \prod_{l=1}^{K} f(x^l | x^{\{1:l-1\}}) = \prod_{l=1}^{K} f(x^l | x^{\Lambda_l}) \propto \prod_{l=1}^{K} e^{-\frac{1}{2\sigma_l^2}(x^l - \boldsymbol{D}_l^\intercal \boldsymbol{x}^{\Lambda_l} - C_l)^2} \tag{244}$$

for some constants C_l and $\sigma_l^2 > 0$, and some vector $\boldsymbol{D}_l \in \mathbb{R}^{|\Lambda_l|}$. We can rewrite this as

$$\begin{aligned} f(\boldsymbol{x}) \propto & e^{-\frac{1}{2\sigma_k^2}(x^k - \boldsymbol{D}_k^\intercal \boldsymbol{x}^{\Lambda_k} - C_k)^2} e^{-\frac{1}{2\sigma_s^2}(x^s - \boldsymbol{D}_s^\intercal \boldsymbol{x}^{\Lambda_s} - C_s)^2} \\ & \cdot \prod_{\substack{l=1 \\ l \neq s \\ l \neq k}}^{K} e^{-\frac{1}{2\sigma_l^2}(x^l - \boldsymbol{D}_l^\intercal \boldsymbol{x}^{\Lambda_l} - C_l)^2}. \end{aligned} \tag{245}$$

From the first line in the expression above, we see that if $i \in \Lambda_k$, we get a factor $e^{Mx^i x^k}$, for some constant $M \neq 0$. Similarly, we see that if $i, k \in \Lambda_s$ for some s, we obtain a factor $e^{Nx^i x^k}$ for some $N \neq 0$. That is, we have that $f(\boldsymbol{x}) \neq g(x^{-\{i\}})h(x^{-\{k\}})$, for some functions g and h. This entails that $x^i \not\perp x^k | \boldsymbol{x}^{-\{i,k\}}$. From (243) we see that $f(\boldsymbol{x}) = g(x^{-\{i\}})h(x^{-\{k\}})$ if and only if $i \notin \Lambda_k$ and $i, k \notin \Lambda_s$. That is, (243) holds if and only if $i \notin \Lambda_k$ and $\nexists s : i, k \in \Lambda_s$. □

F Derivation of posteriors in Section 3.2

Recall that Section 3.1 presents prior distributions for $\boldsymbol{\eta}_t|\boldsymbol{\phi}_t$ and $\boldsymbol{\phi}_t$. In the following, we present the derivation of the corresponding posterior distributions for $\boldsymbol{\eta}_t|\boldsymbol{\phi}_t$ and $\boldsymbol{\phi}_t$, which is presented in Section 3.2. That is, the distributions for $\boldsymbol{\eta}_t|\boldsymbol{\phi}_t, \chi_t$ and $\boldsymbol{\phi}_t|\chi_t$, where we recall from Section 2.4 that χ_t is the ensemble representing $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})$.

We first present the derivation of $f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t, \chi_t)$. By Bayes' rule, we have

$$f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t, \chi_t) = \frac{f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t)f(\chi_t|\boldsymbol{\eta}_t, \boldsymbol{\phi}_t)}{f(\chi_t|\boldsymbol{\phi}_t)} \tag{246}$$

We omit the denominator, which is constant as a function of $\boldsymbol{\eta}_t$:

$$f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t, \chi_t) \propto f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t)f(\chi_t|\boldsymbol{\eta}_t, \boldsymbol{\phi}_t). \tag{247}$$

Since the elements of $\boldsymbol{\eta}_t|\boldsymbol{\phi}_t$ are a priori independent, (94), we have

$$f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t, \chi_t) \propto f(\chi_t|\boldsymbol{\eta}_t, \boldsymbol{\phi}_t) \prod_{k=1}^{K} f(\boldsymbol{\eta}_t^k|\boldsymbol{\phi}_t) \tag{248}$$

From (94), we have that $f(\boldsymbol{\eta}_t^k|\boldsymbol{\phi}_t) = f(\boldsymbol{\eta}_t^k|\phi_t^k)$. In addition, we have that

$$f(\chi_t|\boldsymbol{\eta}_t, \boldsymbol{\phi}_t) = \prod_{k=1}^{K} f(\boldsymbol{\chi}_t^k|\boldsymbol{\eta}_t^k, \phi_t^k, \chi_t^{\Lambda_k}). \tag{249}$$

Thus, we can write

$$f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t, \chi_t) \propto \prod_{k=1}^{K} f(\boldsymbol{\eta}_t^k|\phi_t^k)f(\boldsymbol{\chi}_t^k|\boldsymbol{\eta}_t^k, \phi_t^k, \chi_t^{\Lambda_k}). \tag{250}$$

In the following, we consider the second factor in the expression above.

For each element in $\boldsymbol{x}_t$, we have $x_t^k \sim N((1, (\boldsymbol{x}_t^{\Lambda_k})^\intercal) \cdot \boldsymbol{\eta}_t^k, \phi_t^k)$. We assume that the ensemble members are distributed according to $f(\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1})$, i.e. $\chi_{t,j}^k \sim N((1, (\boldsymbol{\chi}_{t,j}^{\Lambda_k})^\intercal) \cdot \boldsymbol{\eta}_t^k, \phi_t^k)$. Since all of the ensemble members are assumed independent, we have that $\boldsymbol{\chi}_t^k \sim N\left(\left(\mathbf{1}_J, (\chi_t^{\Lambda_k})^\intercal\right) \cdot \boldsymbol{\eta}_t^k, \phi_t^k I_J\right)$, where I_J is the

identity matrix of size J and $\mathbf{1}_J$ is a vector of length J consisting of ones. That is,

$$f(\boldsymbol{\chi}_t^k|\boldsymbol{\eta}_t^k,\phi_t^k,\chi_t^{\Lambda_k}) \propto \frac{1}{|\phi_t^k|^{J/2}}\cdot e^{-\frac{1}{2}\left(\boldsymbol{\chi}_t^k-\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)\cdot\boldsymbol{\eta}_t^k\right)^\intercal(\phi_t^k I_J)^{-1}\left(\boldsymbol{\chi}_t^k-\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)\cdot\boldsymbol{\eta}_t^k\right)}. \tag{251}$$

By expanding the parentheses in the expression above, we have

$$\begin{aligned} f(\boldsymbol{\chi}_t^k|\boldsymbol{\eta}_t^k,\phi_t^k,\chi_t^{\Lambda_k}) \propto \frac{1}{|\phi_t^k|^{J/2}}\cdot e^{-\frac{1}{2}(\boldsymbol{\chi}_t^k)^\intercal(\phi_t^k I_J)^{-1}\boldsymbol{\chi}_t^k+(\boldsymbol{\chi}_t^k)^\intercal(\phi_t^k I_J)^{-1}\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)\cdot\boldsymbol{\eta}_t^k} \\ \cdot e^{-\frac{1}{2}\left(\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)\cdot\boldsymbol{\eta}_t^k\right)^\intercal(\phi_t^k I_J)^{-1}\left(\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)\cdot\boldsymbol{\eta}_t^k\right)} \end{aligned} \tag{252}$$

Since $(\phi_t^k I_J)^{-1} = (\phi_t^k)^{-1} I_J$, we can write

$$\begin{aligned} f(\boldsymbol{\chi}_t^k|\boldsymbol{\eta}_t^k,\phi_t^k,\chi_t^{\Lambda_k}) \propto \frac{1}{|\phi_t^k|^{J/2}}\cdot e^{-\frac{1}{2\phi_t^k}(\boldsymbol{\chi}_t^k)^\intercal\boldsymbol{\chi}_t^k+\frac{1}{\phi_t^k}(\boldsymbol{\chi}_t^k)^\intercal\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)\cdot\boldsymbol{\eta}_t^k} \\ \cdot e^{-\frac{1}{2\phi_t^k}\left(\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)\cdot\boldsymbol{\eta}_t^k\right)^\intercal\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)\cdot\boldsymbol{\eta}_t^k} \end{aligned} \tag{253}$$

The exponent on the last row can be rewritten as

$$\begin{aligned} &-\frac{1}{2\phi_t^k}\left(\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)\cdot\boldsymbol{\eta}_t^k\right)^\intercal\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)\cdot\boldsymbol{\eta}_t^k \\ &=-\frac{1}{2\phi_t^k}(\boldsymbol{\eta}_t^k)^\intercal\cdot\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)^\intercal\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)\cdot\boldsymbol{\eta}_t^k. \end{aligned} \tag{254}$$

Thus,

$$\begin{aligned} f(\boldsymbol{\chi}_t^k|\boldsymbol{\eta}_t^k,\phi_t^k,\chi_t^{\Lambda_k}) \propto \frac{1}{|\phi_t^k|^{J/2}}\cdot e^{-\frac{1}{2\phi_t^k}(\boldsymbol{\chi}_t^k)^\intercal\boldsymbol{\chi}_t^k+\frac{1}{\phi_t^k}(\boldsymbol{\chi}_t^k)^\intercal\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)\cdot\boldsymbol{\eta}_t^k} \\ \cdot e^{-\frac{1}{2\phi_t^k}(\boldsymbol{\eta}_t^k)^\intercal\cdot\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)^\intercal\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)\cdot\boldsymbol{\eta}_t^k}. \end{aligned} \tag{255}$$

In the following, we consider the first factor in (250).

We have that

$$\boldsymbol{\eta}_t^k|\phi_t^k \sim N(\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k},\phi_t^k\Sigma_{\eta_t^k}). \tag{256}$$

That is

$$f(\boldsymbol{\eta}_t^k|\phi_t^k) \propto \frac{1}{|\phi_t^k|^{(|\Lambda_k|+1)/2}}\cdot e^{-\frac{1}{2}(\boldsymbol{\eta}_t^k-\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k})^\intercal(\phi_t^k\Sigma_{\eta_t^k})^{-1}(\boldsymbol{\eta}_t^k-\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k})}. \tag{257}$$

We have that

$$(\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k})^\intercal(\phi_t^k\Sigma_{\eta_t^k})^{-1}\boldsymbol{\eta}_t^k = (\boldsymbol{\eta}_t^k)^\intercal(\phi_t^k\Sigma_{\eta_t^k})^{-1}\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k}. \tag{258}$$

We proceed by expanding the parentheses in (257)

$$\begin{aligned} f(\boldsymbol{\eta}_t^k|\phi_t^k) \propto \frac{1}{|\phi_t^k|^{(|\Lambda_k|+1)/2}}\cdot e^{-\frac{1}{2}(\boldsymbol{\eta}_t^k)^\intercal(\phi_t^k\Sigma_{\eta_t^k})^{-1}\boldsymbol{\eta}_t^k} \\ \cdot e^{(\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k})^\intercal(\phi_t^k\Sigma_{\eta_t^k})^{-1}\boldsymbol{\eta}_t^k-\frac{1}{2}(\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k})^\intercal(\phi_t^k\Sigma_{\eta_t^k})^{-1}\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k}} \end{aligned} \tag{259}$$

By inserting (255) and (259) into (250), we have

$$
\begin{aligned}
f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t,\chi_t) &\propto \prod_{k=1}^{K} f(\boldsymbol{\eta}_t^k|\phi_t^k) f(\boldsymbol{\chi}_t^k|\boldsymbol{\eta}_t^k,\phi_t^k,\chi_t^{\Lambda_k}) \\
&\propto \prod_{k=1}^{K} e^{-\frac{1}{2}(\boldsymbol{\eta}_t^k)^\intercal(\phi_t^k\Sigma_{\boldsymbol{\eta}_t^k})^{-1}\boldsymbol{\eta}_t^k - \frac{1}{2\phi_t^k}(\boldsymbol{\eta}_t^k)^\intercal\cdot\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)^\intercal\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)\cdot\boldsymbol{\eta}_t^k} \\
&\quad \cdot e^{(\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k})^\intercal(\phi_t^k\Sigma_{\boldsymbol{\eta}_t^k})^{-1}\boldsymbol{\eta}_t^k + \frac{1}{\phi_t^k}(\boldsymbol{\chi}_t^k)^\intercal\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)\cdot\boldsymbol{\eta}_t^k} \\
&\quad \cdot \frac{1}{|\phi_t^k|^{(J+|\Lambda_k|+1)/2}} \cdot e^{-\frac{1}{2}(\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k})^\intercal(\phi_t^k\Sigma_{\boldsymbol{\eta}_t^k})^{-1}\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k} - \frac{1}{2\phi_t^k}(\boldsymbol{\chi}_t^k)^\intercal\boldsymbol{\chi}_t^k}.
\end{aligned}
\tag{260}
$$

Note that the terms in the exponent on the second line are quadratic as a function of $\boldsymbol{\eta}_t^k$, while the terms on the third line are linear in $\boldsymbol{\eta}_t^k$. The terms on the last line do not contain $\boldsymbol{\eta}_t^k$. We rearrange the terms on each line

$$
\begin{aligned}
f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t,\chi_t) &\propto \prod_{k=1}^{K} f(\boldsymbol{\eta}_t^k|\phi_t^k) f(\boldsymbol{\chi}_t^k|\boldsymbol{\eta}_t^k,\phi_t^k,\chi_t^{\Lambda_k}) \\
&\propto \prod_{k=1}^{K} e^{-\frac{1}{2}(\boldsymbol{\eta}_t^k)^\intercal\left(\frac{1}{\phi_t^k}(\Sigma_{\boldsymbol{\eta}_t^k})^{-1} + \frac{1}{\phi_t^k}\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)^\intercal\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)\right)\boldsymbol{\eta}_t^k} \\
&\quad \cdot e^{\left(\frac{1}{\phi_t^k}(\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k})^\intercal(\Sigma_{\boldsymbol{\eta}_t^k})^{-1} + \frac{1}{\phi_t^k}(\boldsymbol{\chi}_t^k)^\intercal\cdot\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)\right)\boldsymbol{\eta}_t^k} \\
&\quad \cdot \frac{1}{|\phi_t^k|^{(J+|\Lambda_k|+1)/2}} \cdot e^{-\frac{1}{2\phi_t^k}(\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k})^\intercal(\Sigma_{\boldsymbol{\eta}_t^k})^{-1}\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k} - \frac{1}{2\phi_t^k}(\boldsymbol{\chi}_t^k)^\intercal\boldsymbol{\chi}_t^k}.
\end{aligned}
\tag{261}
$$

In order to simplify the expression, we define

$$
\Theta_t^k = (\Sigma_{\boldsymbol{\eta}_t^k})^{-1} + \left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)^\intercal\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right) \tag{262}
$$

$$
\boldsymbol{\rho}_t^k = \left((\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k})^\intercal(\Sigma_{\boldsymbol{\eta}_t^k})^{-1} + (\boldsymbol{\chi}_t^k)^\intercal\cdot\left(\mathbf{1}_J,(\chi_t^{\Lambda_k})^\intercal\right)\right)^\intercal \tag{263}
$$

$$
\gamma_t^k = (\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k})^\intercal(\Sigma_{\boldsymbol{\eta}_t^k})^{-1}\boldsymbol{\mu}_{\boldsymbol{\eta}_t^k} - (\boldsymbol{\chi}_t^k)^\intercal\boldsymbol{\chi}_t^k \tag{264}
$$

Now, we can write (261) as

$$
\begin{aligned}
f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t,\chi_t) &\propto \prod_{k=1}^{K} f(\boldsymbol{\eta}_t^k|\phi_t^k) f(\boldsymbol{\chi}_t^k|\boldsymbol{\eta}_t^k,\phi_t^k,\chi_t^{\Lambda_k}) \\
&\propto \prod_{k=1}^{K} \frac{1}{|\phi_t^k|^{(J+|\Lambda_k|+1)/2}} \cdot e^{-\frac{1}{2\phi_t^k}(\boldsymbol{\eta}_t^k)^\intercal\Theta_t^k\boldsymbol{\eta}_t^k + \frac{1}{\phi_t^k}(\boldsymbol{\rho}_t^k)^\intercal\boldsymbol{\eta}_t^k - \frac{1}{2\phi_t^k}\gamma_t^k}
\end{aligned}
\tag{265}
$$

Since γ_t^k and the first factor not are functions of $\boldsymbol{\eta}_t^k$, we omit these. We notice that the expression has the form of a normal distribution with canonical

parametrization. By rearranging the terms, we have

$$f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t,\chi_t) \propto \prod_{k=1}^{K} e^{\frac{1}{2\phi_t^k}(\boldsymbol{\rho}_t^k)^\intercal(\Theta_t^k)^{-1}\boldsymbol{\rho}_t^k - \frac{1}{2}\left(\boldsymbol{\eta}_t^k-(\Theta_t^k)^{-1}\boldsymbol{\rho}_t^k\right)^\intercal\left(\frac{1}{\phi_t^k}\Theta_t^k\right)\left(\boldsymbol{\eta}_t^k-(\Theta_t^k)^{-1}\boldsymbol{\rho}_t^k\right)} \tag{266}$$

The first term is constant as a function of $\boldsymbol{\eta}_t^k$, and thus we can omit it. The second term has the form of a normal distribution,

$$\boldsymbol{\eta}_t^k|\boldsymbol{\phi}_t,\chi_t \sim N\left((\Theta_t^k)^{-1}\boldsymbol{\rho}_t^k, \left(\frac{1}{\phi_t^k}\Theta_t^k\right)^{-1}\right). \tag{267}$$

Which completes the derivation of the expression for the posterior distribution for $\boldsymbol{\eta}_t^k|\boldsymbol{\phi}_t$. We now go on with the derivation of the expression for the posterior derivation of $\boldsymbol{\phi}_t$.

We start by using Bayes' rule

$$f(\boldsymbol{\phi}_t|\chi_t) = \frac{f(\boldsymbol{\phi}_t)f(\chi_t|\boldsymbol{\phi}_t)}{f(\chi_t)}. \tag{268}$$

The expression in the denominator is not a function of $\boldsymbol{\phi}_t$, and thus we omit it,

$$f(\boldsymbol{\phi}_t|\chi_t) \propto f(\boldsymbol{\phi}_t)f(\chi_t|\boldsymbol{\phi}_t) = f(\chi_t,\boldsymbol{\phi}_t). \tag{269}$$

In order to calculate this expression, we use the law of total probability

$$f(\boldsymbol{\phi}_t|\chi_t) \propto \int f(\chi_t,\boldsymbol{\eta}_t,\boldsymbol{\phi}_t)d\boldsymbol{\eta}_t. \tag{270}$$

We apply the formula for conditional probability

$$f(\boldsymbol{\phi}_t|\chi_t) \propto \int f(\chi_t|\boldsymbol{\eta}_t,\boldsymbol{\phi}_t)f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t)f(\boldsymbol{\phi}_t)d\boldsymbol{\eta}_t. \tag{271}$$

We put the last factor outside the integral

$$f(\boldsymbol{\phi}_t|\chi_t) \propto f(\boldsymbol{\phi}_t)\int f(\chi_t|\boldsymbol{\eta}_t,\boldsymbol{\phi}_t)f(\boldsymbol{\eta}_t|\boldsymbol{\phi}_t)d\boldsymbol{\eta}_t. \tag{272}$$

In order to treat the integrand, we utilize the fact that the elements of $\boldsymbol{\eta}_t|\boldsymbol{\phi}_t$ are a priori independent, (94), and the GMRF imposed on χ_t

$$f(\boldsymbol{\phi}_t|\chi_t) \propto f(\boldsymbol{\phi}_t)\int\cdots\int\prod_{k=1}^{K} f(\boldsymbol{\chi}_t^k|\boldsymbol{\eta}_t^k,\phi_t^k,\chi_t^{\Lambda_k})f(\boldsymbol{\eta}_t^k|\phi_t^k)d\boldsymbol{\eta}_t^k, \tag{273}$$

where we have K integrals in the expression above. We can swap the ordering of the product-sign and the integral-sign without affecting the result

$$f(\boldsymbol{\phi}_t|\chi_t) \propto f(\boldsymbol{\phi}_t)\prod_{k=1}^{K}\int f(\boldsymbol{\chi}_t^k|\boldsymbol{\eta}_t^k,\phi_t^k,\chi_t^{\Lambda_k})f(\boldsymbol{\eta}_t^k|\phi_t^k)d\boldsymbol{\eta}_t^k. \tag{274}$$

We notice that the integrand is identical to the expression in (265).

$$f(\boldsymbol{\phi}_t|\chi_t) \propto f(\boldsymbol{\phi}_t) \prod_{k=1}^{K} \int \frac{e^{-\frac{1}{2\phi_t^k}(\boldsymbol{\eta}_t^k)^\intercal \Theta_t^k \boldsymbol{\eta}_t^k + \frac{1}{\phi_t^k}(\boldsymbol{\rho}_t^k)^\intercal \boldsymbol{\eta}_t^k - \frac{1}{2\phi_t^k}\gamma_t^k}}{(\phi_t^k)^{(|\Lambda_k|+J+1)/2}} d\boldsymbol{\eta}_t^k. \tag{275}$$

We put the last term in the exponent outside the integral, and rearrange the integrand factors

$$\begin{aligned} f(\boldsymbol{\phi}_t|\chi_t) \propto & f(\boldsymbol{\phi}_t) \prod_{k=1}^{K} \frac{1}{(\phi_t^k)^{(|\Lambda_k|+J+1)/2}} e^{-\frac{1}{2\phi_t^k}\gamma_t^k} e^{\frac{1}{2\phi_t^k}(\boldsymbol{\rho}_t^k)^\intercal (\Theta_t^k)^{-1} \boldsymbol{\rho}_t^k} \\ & \int e^{-\frac{1}{2}\left(\boldsymbol{\eta}_t^k - (\Theta_t^k)^{-1}\boldsymbol{\rho}_t^k\right)^\intercal \left(\frac{1}{\phi_t^k} I_{|\Lambda_k|+1}\right) \Theta_t^k \left(\boldsymbol{\eta}_t^k - (\Theta_t^k)^{-1}\boldsymbol{\rho}_t^k\right)} d\boldsymbol{\eta}_t^k. \end{aligned} \tag{276}$$

We notice that the integrand has the shape of a multivariate normal distribution. Since the normalizing constant is not contained in the integral, we multiply with the normalizing constant, and the expression becomes

$$f(\boldsymbol{\phi}_t|\chi_t) \propto f(\boldsymbol{\phi}_t) \prod_{k=1}^{K} \frac{\left|\left(\frac{1}{\phi_t^k} I_{|\Lambda_k|+1}\right) \Theta_t^k\right|^{-1/2}}{(\phi_t^k)^{(|\Lambda_k|+J+1)/2}} e^{-\frac{1}{2\phi_t^k}\gamma_t^k} e^{\frac{1}{2\phi_t^k}(\boldsymbol{\rho}_t^k)^\intercal (\Theta_t^k)^{-1} \boldsymbol{\rho}_t^k} \tag{277}$$

We omit $|\Theta_t^k|^{-1/2}$, since it is constant as a function of $\boldsymbol{\phi}_t$. Since $\phi_t^k > 0$, we have

$$\begin{aligned} f(\boldsymbol{\phi}_t|\chi_t) &\propto f(\boldsymbol{\phi}_t) \prod_{k=1}^{K} \frac{(\phi_t^k)^{(\Lambda_k+1)/2}}{(\phi_t^k)^{(|\Lambda_k|+J+1)/2}} e^{-\frac{1}{2\phi_t^k}\gamma_t^k} e^{\frac{1}{2\phi_t^k}(\boldsymbol{\rho}_t^k)^\intercal (\Theta_t^k)^{-1} \boldsymbol{\rho}_t^k} \\ &= f(\boldsymbol{\phi}_t) \prod_{k=1}^{K} \frac{1}{(\phi_t^k)^{J/2}} e^{-\frac{1}{2\phi_t^k}\gamma_t^k} e^{\frac{1}{2\phi_t^k}(\boldsymbol{\rho}_t^k)^\intercal (\Theta_t^k)^{-1} \rho_t^k} \end{aligned} \tag{278}$$

We now consider the first factor. Since the elements of $\boldsymbol{\phi}_t$ are a priori independent, (92), and that $\phi_t^k \sim \text{InvGam}(\alpha_t^k, \beta_t^k)$, we have

$$f(\boldsymbol{\phi}_t) = \prod_{k=1}^{K} f(\phi_t^k) = \prod_{k=1}^{K} \frac{1}{(\beta_t^k)^{\alpha_t^k} \Gamma(\alpha_t^k)} \frac{e^{-1/(\phi_t^k \beta_t^k)}}{(\phi_t^k)^{\alpha_t^k+1}} \propto \frac{e^{-1/(\phi_t^k \beta_t^k)}}{(\phi_t^k)^{\alpha_t^k+1}}. \tag{279}$$

The distribution for $\boldsymbol{\phi}_t|\chi_t$ is then

$$\begin{aligned} f(\boldsymbol{\phi}_t|\chi_t) \propto & \prod_{k=1}^{K} \frac{e^{-1/(\phi_t^k \beta_t^k)}}{(\phi_t^k)^{\alpha_t^k+1}} \frac{1}{(\phi_t^k)^{J/2}} e^{-\frac{1}{2\phi_t^k}\gamma_t^k} e^{\frac{1}{2\phi_t^k}(\boldsymbol{\rho}_t^k)^\intercal (\Theta_t^k)^{-1} \boldsymbol{\rho}_t^k} \\ & \prod_{k=1}^{K} \frac{1}{(\phi_t^k)^{\alpha_t^k+J/2+1}} e^{-\frac{1}{\phi_t^k}\left(\frac{1}{\beta_t^k} + \frac{1}{2}\gamma_t^k - \frac{1}{2}(\boldsymbol{\rho}_t^k)^\intercal (\Theta_t^k)^{-1} \boldsymbol{\rho}_t^k\right)} \end{aligned} \tag{280}$$

We see that $f(\boldsymbol{\phi}_t|\chi_t)$ has the form of an inverse gamma distribution with parameters

$$\tilde{\alpha}_t^k = \alpha_t^k + \frac{J}{2} \tag{281}$$

$$\tilde{\beta}_t^k = \left(\frac{1}{\beta_t^k} + \frac{1}{2}(\gamma_t^k - (\boldsymbol{\rho}_t^k)^\intercal(\Theta_t^k)^{-1}\boldsymbol{\rho}_t^k)\right)^{-1}. \tag{282}$$

This completes the derivation of the posterior distributions.

G Derivation of precision matrix in Section 3.3

Recall from (61), that a covariance matrix $\Sigma_{t,j}$ is sampled from the posterior $f(\Sigma_t|\chi_t)$, in order to adjust ensemble member $\boldsymbol{\chi}_{t,j}$ to the observation $\boldsymbol{d}_t$. In Sections and 3.1 and 3.2, we introduce a new prior distribution and its corresponding posterior distribution, respectively. Consequently, $\Sigma_{t,j}$ must be sampled from the new posterior distribution $f(\boldsymbol{\eta}_t, \boldsymbol{\phi}_t|\chi_t)$. In this section we derive the expressions for the elements of the precision matrix $Q_{t,j} = \Sigma_{t,j}^{-1}$ for $\boldsymbol{x}_t|\boldsymbol{d}_{0:t-1}, \boldsymbol{\eta}_t, \boldsymbol{\phi}_t$. For simplicity, we omit the indexes t and j, since they remain unchanged and are of little interest throughout the derivation. The precision matrix Q can be derived by noting that the pdf of $\boldsymbol{x}|\boldsymbol{\eta}, \boldsymbol{\phi}$ can be written in two different ways. First, we can write

$$f(\boldsymbol{x}|\boldsymbol{\eta}, \boldsymbol{\phi}) \propto e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\intercal Q(\boldsymbol{x}-\boldsymbol{\mu})}. \tag{283}$$

Further rewriting yields

$$f(\boldsymbol{x}|\boldsymbol{\eta}, \boldsymbol{\phi}) \propto e^{-\frac{1}{2}\sum_{k=1}^K\sum_{i=1}^K x^k x^i Q^{k,i} - x^k\mu^i Q^{k,i} - x^i\mu^k Q^{k,i} + \mu^k\mu^i Q^{i,k}}. \tag{284}$$

Alternatively, we can write the pdf as

$$f(\boldsymbol{x}|\boldsymbol{\eta}, \boldsymbol{\phi}) = \prod_{l=1}^K f(\boldsymbol{x}^l|\boldsymbol{\eta}^l, \phi^l, \boldsymbol{x}^{\Lambda_l}) \propto \prod_{l=1}^K e^{-\frac{1}{2\phi^l}(x^l - (1,(\boldsymbol{x}^{\Lambda_l})^\intercal)\boldsymbol{\eta}^l)^2}. \tag{285}$$

By expanding the brackets, we obtain

$$\begin{aligned} &f(\boldsymbol{x}|\boldsymbol{\eta}, \boldsymbol{\phi}) \propto e^{-\frac{1}{2}\frac{1}{\phi^k}(x^k)^2 - \frac{2}{\phi^k}(1,(\boldsymbol{x}^{\Lambda_k})^\intercal)\boldsymbol{\eta}^k)x^k + \frac{1}{\phi^k}((1,(\boldsymbol{x}^{\Lambda_k})^\intercal)\boldsymbol{\eta}^k)^2} \\ &e^{-\frac{1}{2}\frac{1}{\phi^s}(x^k)^2 - \frac{2}{\phi^s}(1,(\boldsymbol{x}^{\Lambda_s})^\intercal)\boldsymbol{\eta}^s)x^s + \frac{1}{\phi^s}((1,(\boldsymbol{x}^{\Lambda_s})^\intercal)\boldsymbol{\eta}^s)^2} \\ &\prod_{\substack{l=1\\ l\neq k\\ l\neq s}}^K e^{-\frac{1}{2}\frac{1}{\phi^l}(x^l)^2 - \frac{2}{\phi^l}(1,(\boldsymbol{x}^{\Lambda_l})^\intercal)\boldsymbol{\eta}^l)x^l + \frac{1}{\phi^l}((1,(\boldsymbol{x}^{\Lambda_l})^\intercal)\boldsymbol{\eta}^l)^2}. \end{aligned} \tag{286}$$

By comparing the terms in the exponents of (284) and (286) that are quadratic in $\boldsymbol{x}$, we are able to derive the expression for Q. We first consider the terms that contain $(x^k)^2$.

From (284), we have that the only term that contains $(x^k)^2$ is $Q^{k,k}(x^k)^2$. In (286), we see that $(x^k)^2$ appears in the first term on the first line, $\frac{1}{\phi^k}(x^k)^2$. We must also consider the nodes that have node k as their sequential neighbour. If node k is a sequential neighbour of node s, i.e. $s \in \tilde{\Lambda}_k$, the third term on the second line of (286) contains $(x^k)^2$. Recall that the index of node k in Λ_s is denoted $\Lambda_s^{-1}(k)$. From (286), we see that there is a dot product between $(1, (\boldsymbol{x}^{\Lambda_s})^\intercal)$ and $\boldsymbol{\eta}^s$. Since x^k is the $\Lambda_s^{-1}(k)$th element in $\boldsymbol{x}^{\Lambda_k}$, we have that x^k is multiplied with the $\Lambda_s^{-1}(k)+1$th element in $\boldsymbol{\eta}^s$. Thus, we have the term $\frac{1}{\phi^s}(x^k \eta^{s,\Lambda_s^{-1}(k)+1})^2$ in the exponent of (286). We must also take into account that there could be several nodes in $\tilde{\Lambda}_k$. Thus, we have

$$Q^{k,k} = \frac{1}{\phi^k} + \sum_{s \in \tilde{\Lambda}_k} \frac{1}{\phi^s}(\eta^{s,\Lambda_s^{-1}(k)+1})^2. \tag{287}$$

This completes the derivation of the diagonal entries of Q. In the following, we consider the terms containing $x^k x^s$.

We assume that $s > k$. From Theorem 4, we have that $Q^{k,s} \neq 0$ if and only if either $k \in \Lambda_s$ or $k, s \in \Lambda_l$ for some $l = \{s+1, \dots, K\}$, or both. Recall that we can rewrite the statement $k, s \in \Lambda_l$ as $l \in \tilde{\Lambda}_k \cap \tilde{\Lambda}_s$. That is, $\tilde{\Lambda}_k \cap \tilde{\Lambda}_s$ denotes the set of nodes that has both nodes k and s as their sequential neighbours, i.e. $k, s \in \Lambda_l \implies l \in \tilde{\Lambda}_k \cap \tilde{\Lambda}_s$. In the following, we consider the three cases in which $Q^{k,s} \neq 0$. The first case is when $k \in \Lambda_s$ and $\tilde{\Lambda}_k \cap \tilde{\Lambda}_s = \emptyset$. The second case is when $k \notin \Lambda_s$ and $\tilde{\Lambda}_k \cap \tilde{\Lambda}_s \neq \emptyset$. The third case is when $k \in \Lambda_s$ and $\tilde{\Lambda}_k \cap \tilde{\Lambda}_s \neq \emptyset$. In the following, we consider the first case.

We assume that $k \in \Lambda_s$ and $\tilde{\Lambda}_k \cap \tilde{\Lambda}_s = \emptyset$. From (286), we have that $x^k x^s$ appears in the second term on the second line, i.e. in $-\frac{2}{\phi^s}\eta^{s,\Lambda_s^{-1}(k)+1} x^s x^k$. From (284), we have that $x^s x^k$ appears in the term $Q^{k,s} x^k x^s$ and in the term $Q^{s,k} x^s x^k$. Since Q is symmetric, we have $Q^{k,s} = Q^{s,k}$. By comparing the terms in (284) and (286), that contain $x^k x^s$ in the first case, we have

$$\begin{aligned} Q^{s,k} + Q^{k,s} = 2Q^{k,s} = 2Q^{s,k} &= -\frac{2}{\phi^s}\eta^{s,\Lambda_s^{-1}(k)+1}, \\ Q^{k,s} = Q^{s,k} &= -\frac{1}{\phi^s}\eta^{s,\Lambda_s^{-1}(k)+1}. \end{aligned} \tag{288}$$

We now consider the second case, when we assume that $k \notin \Lambda_s$ and that $\tilde{\Lambda}_k \cap \tilde{\Lambda}_s \neq \emptyset$. We consider $l \in \tilde{\Lambda}_k \cap \tilde{\Lambda}_s$. That is, we consider the l that satisfies $s, k \in \Lambda_l$. The last term on the third line in (286) can be written as

$$\frac{1}{\phi^l}((1, (\boldsymbol{x}^{\Lambda_l})^\intercal)\boldsymbol{\eta}^l)^2 \tag{289}$$

$$= \frac{1}{\phi^l}(\eta^{l,1} + x^{\Lambda_l(1)}\eta^{l,2} + \dots + x^{\Lambda_l^{-1}(\Lambda_l(k))}\eta^{l,\Lambda_l^{-1}(k)+1} \tag{290}$$

$$+ \dots + x^{\Lambda_l^{-1}(\Lambda_l(s))}\eta^{l,\Lambda_l^{-1}(s)+1} + \dots x^{\Lambda_l(|\Lambda_l|)}\eta^{l,|\Lambda_l|+1})^2. \tag{291}$$

We have that $x^{\Lambda_k^{-1}(\Lambda_k(k))} = x^k$ and $x^{\Lambda_k^{-1}(\Lambda_k(s))} = x^s$. Thus, the expression above becomes

$$\frac{1}{\phi^l}((1, (\boldsymbol{x}^{\Lambda_l})^\intercal)\boldsymbol{\eta}^l)^2 \tag{292}$$

$$= \frac{1}{\phi^l}(\eta^{l,1} + x^{\Lambda_l(1)}\eta^{l,2} + \cdots + x^k\eta^{l,\Lambda_l^{-1}(k)+1} \tag{293}$$

$$+ \cdots + x^s\eta^{l,\Lambda_l^{-1}(s)+1} + \ldots x^{\Lambda_l(|\Lambda_l|)}\eta^{l,|\Lambda_l|+1})^2. \tag{294}$$

By expanding the parenthesis, we see that the term $2x^k\eta^{l,\Lambda_l^{-1}(k)+1}x^s\eta^{l,\Lambda_l^{-1}(s)+1}$ appears. The factor x^kx^s appears in (284) in $Q^{k,s}x^kx^s$ and in $Q^{s,k}x^kx^s$. Because Q is symmetric, we have $Q^{k,s} = Q^{s,k}$. Thus, we have that $Q^{k,s} + Q^{s,k} = 2Q^{k,s} = 2Q^{s,k}$. By comparing the terms in (284) and (286), we have

$$Q^{k,s} + Q^{s,k} = 2Q^{k,s} = \sum_{l\in\tilde{\Lambda}_k\cap\tilde{\Lambda}_s} 2\eta^{l,\Lambda_l^{-1}(k)+1}\eta^{l,\Lambda_l^{-1}(s)+1}, \tag{295}$$

$$Q^{k,s} = Q^{s,k} = \sum_{l\in\tilde{\Lambda}_k\cap\tilde{\Lambda}_s} \eta^{l,\Lambda_l^{-1}(k)+1}\eta^{l,\Lambda_l^{-1}(s)+1}, \tag{296}$$

where account for all l that satisfies $s, k \in \Lambda_l$.

For the third case, we assume that $k \in \Lambda_s$ and $\tilde{\Lambda}_k \cap \tilde{\Lambda}_s \neq \emptyset$. In this case, the expression $Q^{k,s}$ is the sum of the two previous cases. That is,

$$Q^{k,s} = Q^{s,k} = -\frac{1}{\phi^s}\eta^{s,\Lambda_s^{-1}(k)+1} + \sum_{l\in\tilde{\Lambda}_k\cap\tilde{\Lambda}_s} \eta^{l,\Lambda_l^{-1}(k)+1}\eta^{l,\Lambda_l^{-1}(s)+1}. \tag{297}$$

This completes the derivation of the expression for the elements of the precision matrix Q.

H Proof of bandwidth-dimension relation in Section 5.2

In Section 4.3, we impose the following sequential neighbourhood on the state space variable $\boldsymbol{x}_t$, $\Lambda_k = \{k-L-1, k-L, k-1\}$, where $K = L^2$ is the dimension of $\boldsymbol{x}_t$. We state in Section 5.2 that $m = \sqrt{K} + 1$, where m is the bandwidth of the matrix Q_t, the precision matrix of $\boldsymbol{x}_t$. In the following, we prove this statement.

Proof. In order to derive the relationship between K and m, we first assess the sparsity of Q_t by considering the sequential neighbourhood Λ_k. Figure 30 displays some of the nodes associated to $\boldsymbol{x}_t$. The figure only displays the edges connected to node k, and the nodes connected to node k. That is, the nodes in the figure are the only nodes that are not conditionally independent of node k. We divide the nodes in Figure 30 into three groups, represented by the three colors in the figure. The yellow nodes are the sequential neighbours of node

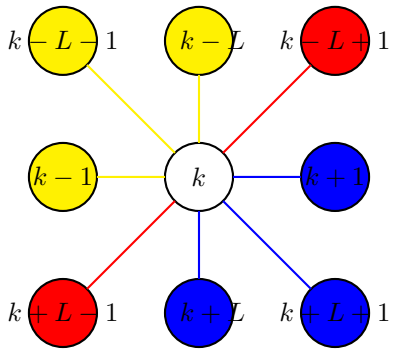
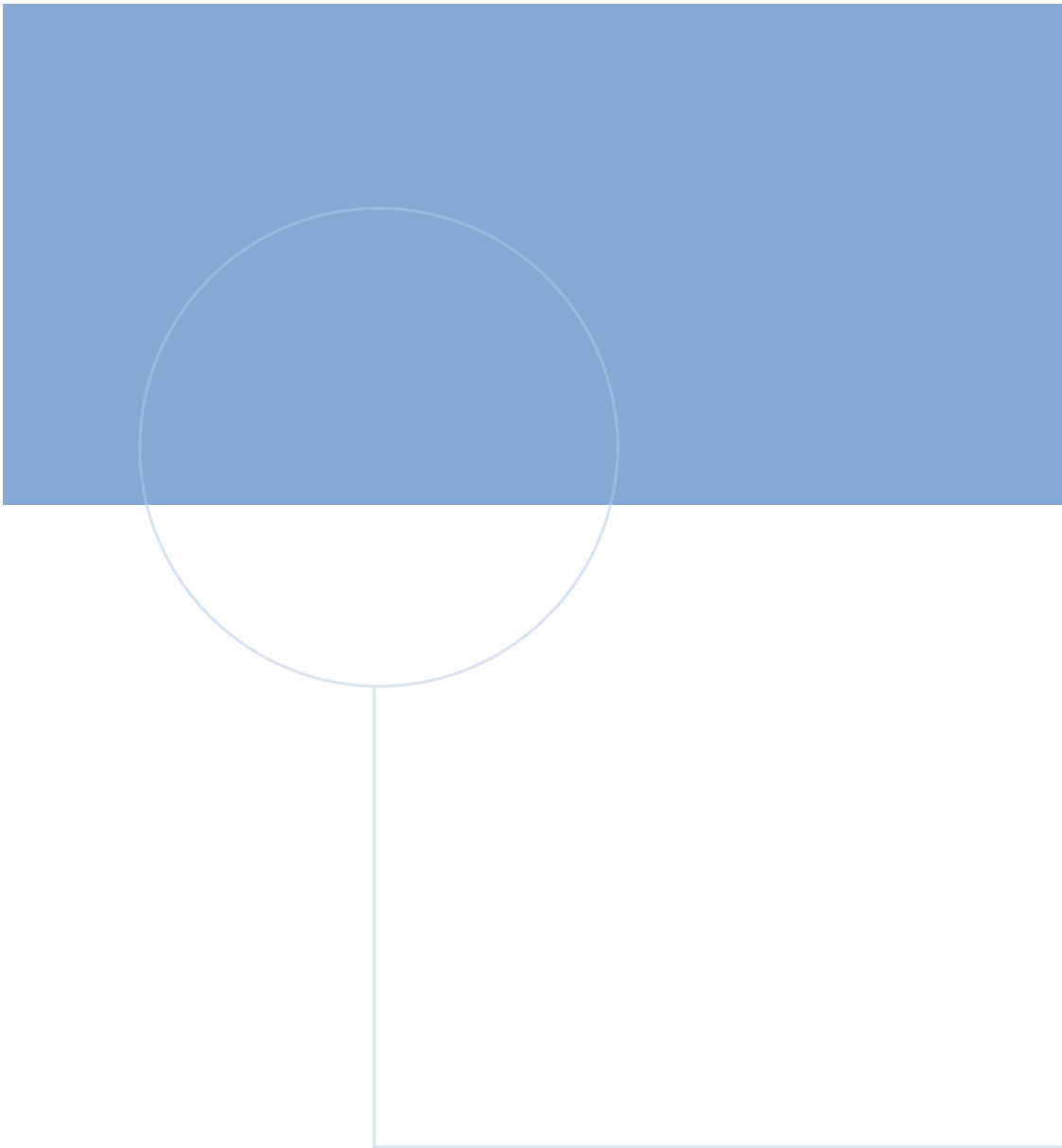


Figure 30: Each node represents an element of the state space vector $\boldsymbol{x}_t$. The edges indicate the conditional dependencies between the nodes if we assume that the sequential neighbour for node k is $\Lambda_k = \{k-L-1, k-L, k-1\}$. The yellow nodes represent the sequential neighbourhood of node k, denoted Λ_k. The blue nodes represent the nodes that has node k as one of their sequential neighbours, $\tilde{\Lambda}_k$. The red nodes represent the nodes that are sequential neighbours of at least one of the nodes in $\tilde{\Lambda}_k$. Note that the only edges included in the graph are the edges connected to node k.

k, i.e. all nodes in the set Λ_k, while the blue nodes are the nodes that have node k as a sequential neighbour, i.e. the nodes contained in $\tilde{\Lambda}_k$. The red nodes are the nodes that are sequential neighbours of at least one of the nodes in $\tilde{\Lambda}_k$, i.e. the nodes r that satisfies $k, r \in \Lambda_s$ for some $s \in \{1, \ldots, K\}$. For instance, node $k-L+1$ is colored red because both nodes k and $k-L+1$ are sequential neighbours of node $k+1$, i.e. $k, k-L+1 \in \Lambda_{k+1}$. Similarly, node $k+L-1$ is red because $k, k+L-1 \in \Lambda_{k+L}$. Also note that some of the nodes belong to more than one of these three groups.

From Theorem 4, we have that $x_t^k \not\perp x_t^r | \boldsymbol{x}_t^{-\{k,r\}}$, if and only if at least of the three following statements are true, $r \in \Lambda_k$, $r \in \tilde{\Lambda}_k$ or $k, r \in \Lambda_s$ for some $s \in \{1, \ldots, K\}$. The nodes satisfying the first statement, $r \in \Lambda_k$, are the yellow nodes in Figure 30, while the nodes satisfying $r \in \tilde{\Lambda}_k$ are the blue nodes. The nodes r satisfying $k, r \in \Lambda_s$ for some $s \in \{1, \ldots, K\}$ are the red nodes in Figure 30. That is, the only elements x_t^r satisfying $x_t^k \not\perp x_t^r | \boldsymbol{x}_t^{-\{k,r\}}$, are the elements in $\boldsymbol{x}_t$ associated to the colored nodes in Figure 30. From (66), we have that $Q_t^{k,r} \neq 0 \iff x_t^k \not\perp x_t^r | \boldsymbol{x}_t^{-\{k,r\}}$. That is, if node r is a colored node in colored in Figure 30, then $Q_t^{k,r} \neq 0$. This entails that the bandwidth of Q_t is $m = \max |k - r|$, where r is the node number for one of the colored nodes in Figure 30. From this figure, we see that the bandwidth of Q_t is $m = |k - (k - L - 1)| = L + 1$. Since $K = L^2$, we have that $m = \sqrt{K} + 1$, which completes the proof.

□



NTNU

Norwegian University of
Science and Technology