Applied Mathematics and Computation 241 (2014) 222–232

Contents lists available at ScienceDirect

Applied Mathematics and Computation

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

A numerically efficient implementation of the expectation maximization algorithm for state space models

Wolfgang Mader [a,b,*], Yannick Linke [a,b], Malenka Mader [a,b,c], Linda Sommerlade [d], Jens Timmer [a,b], Björn Schelter [d]

[a] Institute for Physics, University of Freiburg, Germany
[b] Freiburg Centre for Data Analysis and Modeling (FDM), University of Freiburg, Germany
[c] Department of Neuropediatrics and Muscular Disease, University Medical Center of Freiburg, Germany
[d] Institute for Complex Systems and Mathematical Biology, University of Aberdeen, UK

ARTICLE INFO

Keywords:
Kalman filter
Expectation–maximization algorithm
Parameter estimation
State-space model

ABSTRACT

Empirical time series are subject to observational noise. Naïve approaches that estimate parameters in stochastic models for such time series are likely to fail due to the error-in-variables challenge. State space models (SSM) explicitly include observational noise. Applying the expectation maximization (EM) algorithm together with the Kalman filter constitute a robust iterative procedure to estimate model parameters in the SSM as well as an approach to denoise the signal. The EM algorithm provides maximum likelihood parameter estimates at convergence. The drawback of this approach is its high computational demand. Here, we present an optimized implementation and demonstrate its superior performance to naïve algorithms or implementations.

1. Introduction

Empirical signals are often obscured by a significant amount of observational noise. Observational noise is assumed to be white and Gaussian distributed, which is a justified assumption given the central limit theorem; it is further often assumed to enter the measurements as an additive effect. In contrast to dynamic noise, which adds to the dynamics of the process, observational noise is not part of the dynamics. In stochastic models both types of noise can be accounted for using the state space model (SSM). The SSM consists of an equation describing the dynamics of a process, and an observation equation, modeling the observation function and observational noise.

For linear stochastic models, vector autoregressive (VAR[p]) processes are often used as a model for the dynamics. A VAR[p] is a versatile model which turns out to be powerful in estimating spectral characteristics, interaction structures, or network topologies. For example, the partial directed coherence [2,13] in the frequency domain and the directed partial correlation [4,7] in time domain, both measures for Granger causality, are based on parameters of vector autoregressive processes. Accurate parameter estimates are vital to get reliable results using such measures.

Naïve estimators for parameters in the autoregressive model neglect observational noise. This leads to strongly biased estimates. The expectation maximization (EM) algorithm [3] provides an iterative maximum likelihood estimator [1] for

[*] Corresponding author at: Freiburg Center for Data Analysis and Modeling (FDM), Eckerstr. 1, 79104 Freiburg, Germany.
E-mail addresses: Wolfgang.Mader@fdm.uni-freiburg.de (W. Mader), Yannick.Linke@fdm.uni-freiburg.de (Y. Linke), Malenka.Mader@fdm.uni-freiburg.de (M. Mader), l.sommerlade@abdn.ac.uk (L. Sommerlade), jeti@fdm.uni-freiburg.de (J. Timmer), b.schelter@abdn.ac.uk (B. Schelter).

http://dx.doi.org/10.1016/j.amc.2014.05.021
0096-3003/© 2014 Elsevier Inc. All rights reserved.

the parameters in the SSM [16]. This maximum likelihood approach explicitly accounts for observational noise by relying on the state space model and therefore provides unbiased estimators. The EM algorithm for state space models is based on the Kalman filter [5]. This filter is used to get estimates of the hidden states. The state estimates are then used to improve the estimates of the process parameters. Once the EM algorithm has converged, the parameter estimates are asymptotically unbiased and have smallest possible variance. Additionally, an estimate of the hidden state is obtained.

The EM algorithm is typically used in this setting as it turns out to be more robust than direct application of the Newton–Raphson algorithm, quasi Newton or conjugate gradient approaches [11]. Robustness here refers to more stable convergence, as the EM algorithm ensures, for instance, stability of the VAR[p] at all times. For the afore mentioned approaches this cannot be guaranteed such that they typically converge only if initialized close to the optimal solution.

The usefulness of state space modeling together with the EM algorithm has already been demonstrated [8,9]. However, no comprehensive overview including a numerically efficient implementation seems to exist. Given the relevance for many fields, especially in the neuroscience, we here discuss such a robust and numerically efficient implementation and provide a toolbox written in C++. Moreover, we prove that the update strategy of the expectation maximization algorithm is optimal even for the general case in which autoregressive processes of order $p > 1$ are used in the state space model.

The manuscript is organized as follows. In Section 2.1, the state space model is introduced. Sections 2.2 and 2.3 deal with the Kalman filter. The expectation maximization (EM) algorithm is described in Section 2.4. Approaches to overcome its high computational demand are discussed in Section 2.5. Results are presented in Section 3. A first simulation study demonstrates the run-time improvement of the proposed optimization, Section 3.1. In a second simulation study shown in Section 3.2, the optimized EM algorithm is shown to properly estimate spectral properties. In Section 3.2.2, the estimation of absolute values of process parameters and their uncertainties are concerned. Finally, a conclusion in drawn in Section 4.

2. Methods

This section introduces the expectation maximization (EM) algorithm [3], c.f. Fig. 1. The iterative EM algorithm (dotted box) consists of two steps. In the expectation step conditional expected values of the hidden states $\boldsymbol{x}_t$ and its covariance $\boldsymbol{P}_t$ are obtained using the Kalman filter and smoother. Based on those values, in the maximization step, the expected value of the likelihood is maximized with respect to the parameters, yielding a new set of parameters, which is used in the next iteration of the EM algorithm. In the first iteration of the EM algorithm the parameters $\boldsymbol{P}(1)$ needs to be initialized. Therefore, e.g. least squares parameter estimates can be used. The algorithm iterates until a convergence criterion is reached.

The observed data $\boldsymbol{y}_t$ is a function of the hidden process $\boldsymbol{x}_t$ and observational noise $\boldsymbol{\eta}_t$. The SSM links the observations to the hidden states. Hence, the EM algorithm returns an estimate of the hidden states and an estimate of the parameters after convergence.

This section provides details of all components of the EM-Kalman framework, beginning with the SSM (Section 2.1) followed by the Kalman filter (Section 2.2) and Kalman smoother (Section 2.3). Based on those components, the EM algorithm is used for parameter estimation in the SSM (Section 2.4). For a numerically efficient implementation, optimization of the EM algorithm is essential. In Section 2.5 we propose and test two decisive optimization possibilities.

2.1. State space model

The state space model (SSM) is used in the Kalman filter to model the data. It consists of two equations. The first equation models the dynamics of the process (Section 2.1.1), comprising its parameters. The second equation models its observation

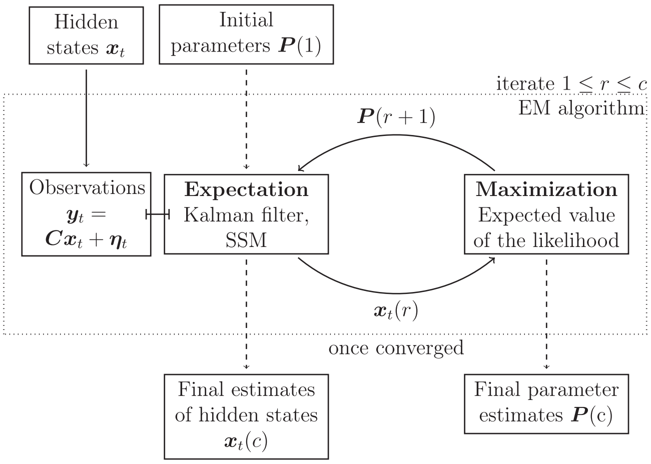

Hidden states $\boldsymbol{x}_t$
Initial parameters $\boldsymbol{P}(1)$
iterate $1 \le r \le c$
EM algorithm
$\boldsymbol{P}(r+1)$
Observations $\boldsymbol{y}_t = \boldsymbol{C}\boldsymbol{x}_t + \boldsymbol{\eta}_t$
Expectation Kalman filter, SSM
Maximization Expected value of the likelihood
$\boldsymbol{x}_t(r)$
once converged
Final estimates of hidden states $\boldsymbol{x}_t(c)$
Final parameter estimates $\boldsymbol{P}(c)$

Fig. 1. Kalman filter in the expectation maximization algorithm. The Kalman filter is deployed to obtain conditional means using parameters $\boldsymbol{P}(r)$ in every iteration r. Maximization of the expected value of the likelihood function leads to new parameters $\boldsymbol{P}(r+1)$.

(Section 2.1.2). Therefore, it is possible to account for driving noise, which is part of the process, as well as additive observational noise, which obfuscates the measurement. Moreover, every channel of the observation can be a combination of different components of the process, some of which might be unobserved. Since the process states can not be observed directly, they are called hidden states, and the process itself the hidden process.

2.1.1. VAR[p] processes

Here, the dynamics of the underlying process is modeled by a linear stochastic equation, which is the vector autoregressive (VAR[p]) process of order p

$$\boldsymbol{x}_t = \sum_{\tau=1}^{p} \boldsymbol{A}_t(\tau)\boldsymbol{x}_{t-\tau} + \boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}). \tag{1}$$

It assembles its current state vector $\boldsymbol{x}_t$ from its past p state vectors and additive Gaussian driving noise $\boldsymbol{\epsilon}_t$, with zero mean and covariance matrix $\boldsymbol{Q}$. The dynamics of the process is determined by the transition matrices $\boldsymbol{A}_t(\tau)$, which, in general, can vary over time. The state and noise vectors, $\boldsymbol{x}_t$ and $\boldsymbol{\epsilon}_t$, are of dimension d while $\boldsymbol{A}$ and $\boldsymbol{Q}$ are $d \times d$-matrices.

Throughout this manuscript, only processes with time constant $\boldsymbol{A}(\tau)$ are considered. This implies, that the dynamics of the process does not change on the time scale of the length of the measurement.

For the application in the EM algorithm it is useful to formulate a general VAR[p] as a VAR[1] process. By embedding, ${}_{\mathrm{e}}(\cdot)$, transition matrix, state, and noise vector, Eq. (1) translates to [16]

$${}_{\mathrm{e}}\boldsymbol{x}_t = \begin{pmatrix} x_t^1 \\ \vdots \\ x_t^d \\ \vdots \\ x_{t-p+1}^1 \\ \vdots \\ x_{t-p+1}^d \end{pmatrix} = \underbrace{\begin{pmatrix} A(1)^{1,1} & \dots & A(1)^{1,d} & \dots & A(p)^{1,1} & \dots & A(p)^{1,d} \\ \vdots & \vdots & \vdots & \vdots & \vdots & & \\ A(1)^{d,1} & \dots & A(1)^{d,d} & \dots & A(p)^{d,1} & \dots & A(p)^{d,d} \\ 1 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & \ddots & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & \ddots & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & 0 & \dots & 0 \end{pmatrix}}_{{}_{\mathrm{e}}\boldsymbol{A}} \underbrace{\begin{pmatrix} x_{t-1}^1 \\ \vdots \\ x_{t-1}^d \\ \vdots \\ x_{t-p}^1 \\ \vdots \\ x_{t-p}^d \end{pmatrix}}_{{}_{\mathrm{e}}\boldsymbol{x}_{t-1}} + \underbrace{\begin{pmatrix} \epsilon_t^1 \\ \vdots \\ \epsilon_t^d \\ 0 \\ \vdots \\ \vdots \\ 0 \end{pmatrix}}_{{}_{\mathrm{e}}\boldsymbol{\epsilon}_t}. \tag{2}$$

The sum of the past p states in Eq. (1) has been rewritten as the product of the $pd \times pd$ matrix ${}_{\mathrm{e}}\boldsymbol{A}$ and the $pd \times 1$ vector ${}_{\mathrm{e}}\boldsymbol{x}_t$. The $pd \times pd$ covariance matrix of the driving noise reads

$${}_{\mathrm{e}}\boldsymbol{Q} = \begin{pmatrix} \mathrm{var}(\epsilon^1, \epsilon^1) & \dots & \mathrm{cov}(\epsilon^1, \epsilon^d) & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathrm{cov}(\epsilon^d, \epsilon^1) & \dots & \mathrm{var}(\epsilon^d, \epsilon^d) & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & 0 & 0 & \dots & 0 \end{pmatrix}. \tag{3}$$

Since the Kalman filter is build around a VAR[1] process, the embedded form is used in the following.

2.1.2. Observation of the process

The observation

$$\boldsymbol{y}_t = \boldsymbol{C}\boldsymbol{x}_t + \boldsymbol{\eta}_t, \quad \boldsymbol{\eta}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R}) \tag{4}$$

is modeled by the $b \times pd$ observation matrix $\boldsymbol{C}$ and additive observational noise $\boldsymbol{\eta}_t$. The dimension of $\boldsymbol{y}_t$ and $\boldsymbol{\eta}_t$ is $b \times 1$. The observational noise is assumed to be Gaussian distributed with zero mean and $b \times b$ covariance matrix $\boldsymbol{R}$. In general, $b \neq pd$ since not all components of the underlying process are observed. This is especially the case when reformulating a VAR[p] as a VAR[1] process, where only the first d hidden states are observed. By setting off-diagonal entries of $\boldsymbol{C}$ to non-zero values, linear combinations of components of the state vector are observed.

The linear state space model in its general form can be written as

$$\begin{aligned} \boldsymbol{x}_t &= \boldsymbol{A}_t\boldsymbol{x}_{t-1} + \boldsymbol{\epsilon}_t, \\ \boldsymbol{y}_t &= \boldsymbol{C}\boldsymbol{x}_t + \boldsymbol{\eta}_t. \end{aligned} \tag{5}$$

2.1.3. Uniqueness of parameters

Identical observations $\boldsymbol{y}_t$ can be obtained from an infinite number of combinations of matrices $\boldsymbol{A}$, $\boldsymbol{C}$, and $\boldsymbol{Q}$, since the state space model is invariant under a transformation with an invertible matrix. Due to this invariance, the components of the process, e.g. the positions of entries in the state vector $\boldsymbol{x}_t$, can be reordered. Therefore, the entries of the parameter matrix $\boldsymbol{A}$ are not uniquely defined. Reordering $\boldsymbol{A}$ leads to a new set of parameters without changing the likelihood. Hence, contour lines exist in the likelihood landscape which prevent the optimization from finishing. By fixing the observation matrix $\boldsymbol{C}$, e.g. to the identity $\mathbf{1}$, the order of components of the process is fixed, such that parameters can be estimated. In consequence, the observation of linear combinations of dimensions of the hidden state $\boldsymbol{x}_t$ is excluded from the model. The important case of instantaneous interaction is still covered however, since the covariance of the driving noise $\boldsymbol{Q}$ is estimated without constraints, and instantaneous interaction can be realized by correlated driving noise.

2.2. Kalman filter

In the following, a measurement time series containing n observations is assumed. Time $t = 1, \ldots, n$ is used to reference these observations. For conditional expectations [16]

$$\boldsymbol{x}_t^s = \mathrm{E}[\boldsymbol{x}_t | \boldsymbol{y}_1, \ldots, \boldsymbol{y}_s], \tag{6}$$

$$\boldsymbol{P}_{t_1,t_2}^s = \mathrm{E}\left\{(\boldsymbol{x}_{t_1} - \boldsymbol{x}_{t_1}^s)(\boldsymbol{x}_{t_2} - \boldsymbol{x}_{t_2}^s)^{\mathrm{T}}\right\} \overset{a}{=} \mathrm{E}\left\{(\boldsymbol{x}_{t_1} - \boldsymbol{x}_{t_1}^s)(\boldsymbol{x}_{t_2} - \boldsymbol{x}_{t_2}^s)^{\mathrm{T}} | \boldsymbol{y}_1, \ldots, \boldsymbol{y}_s\right\}, \tag{7}$$

the subscript denotes the estimation time point, the superscript up to which measurement it is conditioned on. The expected value is denoted by $\mathrm{E}[\cdot]$. The relation $\overset{a}{=}$ in Eq. (7) only holds if the process underlying $\boldsymbol{x}_t$ is Gaussian as assumed here.

The Kalman filter equations are [15]

$$\boldsymbol{x}_t^{t-1} = \boldsymbol{A}\boldsymbol{x}_{t-1}^{t-1} \tag{8}$$

$$\boldsymbol{x}_t^t = \boldsymbol{x}_t^{t-1} + \boldsymbol{K}_t\left(\boldsymbol{y}_t - \boldsymbol{C}\boldsymbol{x}_t^{t-1}\right) \tag{9}$$

$$\boldsymbol{P}_t^{t-1} = \boldsymbol{A}\boldsymbol{P}_{t-1}^{t-1}\boldsymbol{A}^{\mathrm{T}} + \boldsymbol{Q} \tag{10}$$

$$\boldsymbol{P}_t^t = \boldsymbol{P}_t^{t-1} - \boldsymbol{K}_t\boldsymbol{C}\boldsymbol{P}_t^{t-1} \tag{11}$$

$$\boldsymbol{K}_t = \boldsymbol{P}_t^{t-1}\boldsymbol{C}^{\mathrm{T}}\left(\boldsymbol{C}\boldsymbol{P}_t^{t-1}\boldsymbol{C}^{\mathrm{T}} + \boldsymbol{R}\right)^{-1}, \tag{12}$$

with initial values $\boldsymbol{x}_0^0 = \mathrm{E}[\boldsymbol{x}_t] = \boldsymbol{\mu}$, and $\boldsymbol{P}_0^0 = \mathrm{E}[\boldsymbol{x}_t \cdot \boldsymbol{x}_t^{\mathrm{T}}] = \boldsymbol{\Sigma}$.

The filter involves a time update and a measurement update step [17]. Time update, Eqs. (8) and (10), is a model driven step advancing from time $t-1$ to t. It results in the prior estimate $\boldsymbol{x}_t^{t-1}$ and its covariance $\boldsymbol{P}_t^{t-1}$. The measurement update, Eqs. (9) and (11), corrects the prior estimates by taking into account the current prediction $\boldsymbol{x}_t^{t-1}$, measurement $\boldsymbol{y}_t$, and the Kalman gain, Eq. (12), leading to the posterior estimates.

Eqs. (8)–(12) implement a forward recursion since only observations and estimates from the past and the present are used. Hence, those equations are causal and allow for online application. Moreover, Eqs. (10)–(12) do not depend on the observations $\boldsymbol{y}_t$ and can be calculated offline if the parameters of the model are known.

2.3. Kalman smoother

In contrast to the Kalman filter, the smoother is a backward recursion [12]. Therefore, one can take advantage of the smoother once the entire time series has been recorded. Smoothed estimates are more accurate than filtered ones. The Kalman smoother equations for $t = n,\ n-1, \ldots, 1$ are [15]

$$\boldsymbol{J}_{t-1} = \boldsymbol{P}_{t-1}^{t-1}\boldsymbol{A}^{\mathrm{T}}\left(\boldsymbol{P}_t^{t-1}\right)^{-1}, \tag{13}$$

$$\boldsymbol{x}_{t-1}^n = \boldsymbol{x}_{t-1}^{t-1} + \boldsymbol{J}_{t-1}\left(\boldsymbol{x}_t^n - \boldsymbol{A}\boldsymbol{x}_{t-1}^{t-1}\right), \tag{14}$$

$$\boldsymbol{P}_{t-1}^n = \boldsymbol{P}_{t-1}^{t-1} + \boldsymbol{J}_{t-1}\left(\boldsymbol{P}_t^n - \boldsymbol{P}_t^{t-1}\right)\boldsymbol{J}_{t-1}^{\mathrm{T}}. \tag{15}$$

The matrix $\boldsymbol{J}$ is the smoothing gain, analogously to the Kalman gain $\boldsymbol{K}$. The initial values for the smoother are the final estimates of the filter, $\boldsymbol{x}_n^n$ and $\boldsymbol{P}_n^n$. The recursion [15]

$$\boldsymbol{P}_{t-1,t-2}^n = \boldsymbol{P}_{t-1}^{t-1}\boldsymbol{J}_{t-2}^{\mathrm{T}} + \boldsymbol{J}_{t-1}\left(\boldsymbol{P}_{t,t-1}^n - \boldsymbol{A}\boldsymbol{P}_{t-1}^{t-1}\right)\boldsymbol{J}_{t-2}^{\mathrm{T}}, \quad t = n, \quad n-1, \ldots, 2 \tag{16}$$

with initial value

$$\boldsymbol{P}_{n,n-1}^n = (\boldsymbol{I} - \boldsymbol{K}_n\boldsymbol{C})\boldsymbol{A}_n\boldsymbol{P}_{n-1}^{n-1} \tag{17}$$

is the lag one covariance smoother. The lag one covariance $\boldsymbol{P}_{t,t-1}^n$ is required in the expectation maximization algorithm, see Eq. (20).

2.4. Expectation maximization algorithm

Different approaches exist to fit model parameters to data, one of which is maximum likelihood estimation (MLE). The likelihood is a function describing the probability of the data recorded given the model parameters. Maximizing the likelihood leads to the parameters of the model for which the observed time series is most likely.

Following [15], an iterative maximum likelihood estimator of the parameters of the state space model is derived. For the complete data log-likelihood

$$\log\ L = -\frac{1}{2}\log|\boldsymbol{\Sigma}| - \frac{1}{2}(\boldsymbol{x}_0-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}_0-\boldsymbol{\mu}) - \frac{n}{2}\log|\boldsymbol{Q}| - \frac{1}{2}\sum_{t=1}^{n}(\boldsymbol{x}_t-\boldsymbol{A}\boldsymbol{x}_{t-1})^{\mathrm{T}}\boldsymbol{Q}^{-1}(\boldsymbol{x}_t-\boldsymbol{A}\boldsymbol{x}_{t-1}) - \frac{n}{2}\log|\boldsymbol{R}| \\ - \frac{1}{2}\sum_{t=1}^{n}(\boldsymbol{y}_t-\boldsymbol{C}\boldsymbol{x}_t)^{\mathrm{T}}\boldsymbol{R}^{-1}(\boldsymbol{y}_t-\boldsymbol{C}\boldsymbol{x}_t). \tag{18}$$

both $\boldsymbol{x}_t$ and $\boldsymbol{y}_t$ are taking into account. However, since the hidden states $\boldsymbol{x}_t$ are unknown, only the expected value

$$\begin{aligned} G(\boldsymbol{\Theta}) = \mathrm{E}(\log\ L|\boldsymbol{y}_1,\ldots,\boldsymbol{y}_n) = &-\frac{1}{2}\log|\boldsymbol{\Sigma}| - \frac{1}{2}\mathrm{tr}\left\{\boldsymbol{\Sigma}^{-1}\left(\boldsymbol{P}_0^n + (\boldsymbol{x}_0^n-\boldsymbol{\mu})(\boldsymbol{x}_0^n-\boldsymbol{\mu})^{\mathrm{T}}\right)\right\} \\ &-\frac{n}{2}\log|\boldsymbol{Q}| - \frac{1}{2}\mathrm{tr}\left\{\boldsymbol{Q}^{-1}\left(\boldsymbol{F}-\boldsymbol{E}\boldsymbol{A}^{\mathrm{T}}-\boldsymbol{A}\boldsymbol{E}^{\mathrm{T}}+\boldsymbol{A}\boldsymbol{D}\boldsymbol{A}^{\mathrm{T}}\right)\right\} - \frac{n}{2}\log|\boldsymbol{R}| \\ &-\frac{1}{2}\mathrm{tr}\left\{\boldsymbol{R}^{-1}\sum_{t=1}^{n}\left[(\boldsymbol{y}_t-\boldsymbol{C}\boldsymbol{x}_t^n)(\boldsymbol{y}_t-\boldsymbol{C}\boldsymbol{x}_t^n)^{\mathrm{T}} + \boldsymbol{C}\boldsymbol{P}_t^n\boldsymbol{C}^{\mathrm{T}}\right]\right\}, \end{aligned} \tag{19}$$

of the log-likelihood conditioned on $\boldsymbol{y}_1,\ldots,\boldsymbol{y}_n$ is accessible. The abbreviations

$$\begin{aligned} &\boldsymbol{D} = \sum_{t=1}^{n}\left(\boldsymbol{P}_{t-1}^n + \boldsymbol{x}_{t-1}^n\boldsymbol{x}_{t-1}^{n\ \mathrm{T}}\right), \quad \boldsymbol{E} = \sum_{t=1}^{n}\left(\boldsymbol{P}_{t,t-1}^n + \boldsymbol{x}_t^n\boldsymbol{x}_{t-1}^{n\ \mathrm{T}}\right), \\ &\boldsymbol{F} = \sum_{t=1}^{n}\left(\boldsymbol{P}_t^n + \boldsymbol{x}_t^n\boldsymbol{x}_t^{n\mathrm{T}}\right) \end{aligned} \tag{20}$$

have been used in Eq. (19). The quantities required in Eq. (20) are the result of the Kalman smoother of the rth EM iteration, see Eqs. (14) and (15).

To maximize $G(\boldsymbol{\Theta})$, its derivative is set to zero, leading to the update rules

$$\boldsymbol{A}^{(r+1)} = \boldsymbol{E}\boldsymbol{D}^{-1}, \tag{21}$$

$$\boldsymbol{Q}^{(r+1)} = \frac{1}{n}\left(\boldsymbol{F}-\boldsymbol{E}\boldsymbol{D}^{-1}\boldsymbol{E}^{\mathrm{T}}\right), \tag{22}$$

$$\boldsymbol{R}^{(r+1)} = \frac{1}{n}\sum_{t=1}^{n}\left[(\boldsymbol{y}_t-\boldsymbol{C}\boldsymbol{x}_t^n)(\boldsymbol{y}_t-\boldsymbol{C}\boldsymbol{x}_t^n)^{\mathrm{T}} + \boldsymbol{C}\boldsymbol{P}_t^n\boldsymbol{C}^{\mathrm{T}}\right]. \tag{23}$$

The update of $\boldsymbol{\mu}$ is $\boldsymbol{x}_0^n$ of the last EM iteration. If the measurement is corrected for the mean, the initial value for the first EM iteration of $\boldsymbol{\mu}$ is set to $\boldsymbol{0}$. The initial value of the covariance of the process $\boldsymbol{\Sigma}$ can either be estimated or set to a reasonable baseline value [15].

The update rules of the EM algorithm are guaranteed to always yield parameters of a stationary process. Moreover, the likelihood never decreases. Therefore, no adjustment of step size is needed [3]. However, the EM algorithm converges slower than quadratic methods, but has more robust convergence properties [15].

2.4.1. VAR parameter constraints

A decisive step for parameter estimation in the EM-framework is that the expected value of the log-likelihood (Eq. (19)) consists of variables, i.e. $\boldsymbol{x}_t^n$, $\boldsymbol{x}_{t-1}^n$, and $\boldsymbol{P}_{t,t-1}^n$, which can be derived from the Kalman smoother, Section 2.3. The Kalman smoother is initialized by the output of the Kalman filter, Section 2.2, which is designed for VAR[1] processes. Hence, to use the general VAR[p] model, it must be reformulated as a VAR[1], which is always possible as described in Section 2.1.1. As also noted in Section 2.1.1, $\boldsymbol{A}$ and $\boldsymbol{Q}$ gain structure by this reformulation (Eqs. (2) and (3)) which are maintained by the update rules of the EM algorithm, Eqs. (21) and (22). This is proofed in the following, with more details given in Appendix A.

By means of Lagrangian multipliers, the optimization underlying the update rules can be carried out with constraints. The constraints for $\boldsymbol{A}$ are formulated by ϕ_{ij}, the ones for $\boldsymbol{Q}$ by ψ_{ij}. The Lagrange multipliers λ_{ij} and ω_{ij} are set to zero for the unconstrained parts of $\boldsymbol{A}$ and $\boldsymbol{Q}$ respectively, c.f. Eqs. (2) and (3). For the constrained part of $\boldsymbol{A}$ and $\boldsymbol{Q}$, the parameters ϕ_{ij} and ψ_{ij} are set to 0 or 1 corresponding to the target value. This adds two additional term to the likelihood equation Eq. (19) yielding

$$G^{\mathrm{c}} = G(\boldsymbol{\mu},\boldsymbol{\Sigma},\boldsymbol{A},\boldsymbol{Q},\boldsymbol{R}) + \sum_{i,j}\lambda_{ij}(A_{ij}-\phi_{ij}) + \sum_{i,j}\omega_{ij}(Q_{ij}-\psi_{ij}). \tag{24}$$

Maximization with respect to $\boldsymbol{Q}$ leads to

$$\boldsymbol{Q} = \check{\boldsymbol{Q}} + \frac{2}{n}\boldsymbol{Q}\boldsymbol{\Omega}\boldsymbol{Q}, \tag{25}$$

where $\check{\boldsymbol{Q}}$ denotes the unconstrained covariance from Eq. (22). The matrix

$$\boldsymbol{\Omega} = \begin{pmatrix} \boldsymbol{0} & \\ & \check{\boldsymbol{\Omega}} \end{pmatrix} \tag{26}$$

is composed of a $d \times d$ zero matrix located at the upper left, and the Lagrange multipliers, $\check{\Omega}_{ij} = \omega_{ij}$. The second term in Eq. (25) is zero due to the shape of the constrained $\boldsymbol{Q}$ and $\boldsymbol{\Omega}$, and $\boldsymbol{Q} = \check{\boldsymbol{Q}}$ follows. A similar argument applies to $\boldsymbol{A}$. Nevertheless, due to numerical inaccuracy, it is advisable to set the constrained elements of $\boldsymbol{A}$ and $\boldsymbol{Q}$ to zero or one, respectively.

2.4.2. Convergence criterion

The EM algorithm is stopped when a convergence criterion is reached. Two approaches are often used. The first criterion is based on the convergence of the incomplete data likelihood [14]

$$\log L = -\frac{1}{2}\sum_{t=1}^{n} \log\left|\boldsymbol{C}\boldsymbol{P}_t^{t-1}\boldsymbol{C}^{\mathrm{T}} + \boldsymbol{R}\right| - \frac{1}{2}\sum_{t=1}^{n}\left(\boldsymbol{y}_t - \boldsymbol{C}\boldsymbol{x}_t^{t-1}\right)^{\mathrm{T}}\left(\boldsymbol{C}\boldsymbol{P}_t^{t-1}\boldsymbol{C}^{\mathrm{T}} + \boldsymbol{R}\right)^{-1}\left(\boldsymbol{y}_t - \boldsymbol{C}\boldsymbol{x}_t^{t-1}\right). \tag{27}$$

The second approach uses convergence of the parameters $\boldsymbol{A}$, $\boldsymbol{Q}$, and $\boldsymbol{R}$. For instance for the parameter matrix $\boldsymbol{A}$, the relative change

$$\delta A = \max_{i,j}\left\{\frac{(\boldsymbol{A}_t^t)_{ij} - (\boldsymbol{A}_{t-1}^{t-1})_{ij}}{(\boldsymbol{A}_t^t)_{ij}}\right\}$$

can be used as converge criterion. Both approaches suffer from the drawback that vanishing changes of either the likelihood or the parameters do not necessarily imply global optimality of the parameters. However, the primary aim of the EM algorithm is to obtain parameter estimates. Therefore, it remains highly advisable to define convergence by means of parameter changes.

2.5. Run-time optimization

In this section we describe two run-time optimization approaches and their results. We exploited the possibility of the EM algorithm for parallelization as well as fixed-point equations for the filter and smoothing gains.

2.5.1. Parallelization

The EM algorithm allows for parallelization in case of multiple panels. Panels refer to different time series which are recorded from the same process, e.g. repeated measurements. A single recording might also be split up into shorter parts, leading to panels. For each panel j the EM algorithm can be run independently, obtaining $\boldsymbol{D}_j$, $\boldsymbol{E}_j$, and $\boldsymbol{F}_j$ in parallel. To update $\boldsymbol{A}$, $\boldsymbol{Q}$, and $\boldsymbol{R}$, Eqs. (21)–(23), the sums $\boldsymbol{D} = \sum_j \boldsymbol{D}_j$, $\boldsymbol{E} = \sum_j \boldsymbol{E}_j$ and $\boldsymbol{F} = \sum_j = \boldsymbol{F}_j$ have to be used. This can been shown by introducing the sum over panels in the likelihood (Eq. (18)) and carrying out the maximization. On a multi-core computer system, the runtime can be reduced by a factor of the minimum of number of panels or number of cores.

2.5.2. Fixed point equations for the gains

The Kalman filter and smoother gains

$$\boldsymbol{K}_t = \boldsymbol{P}_t^{t-1}\boldsymbol{C}^{\mathrm{T}}\left(\boldsymbol{C}\boldsymbol{P}_t^{t-1}\boldsymbol{C}^{\mathrm{T}} + \boldsymbol{R}\right)^{-1}, \tag{28}$$

$$\boldsymbol{J}_{t-1} = \boldsymbol{P}_{t-1}^{t-1}\boldsymbol{A}^{\mathrm{T}}\left(\boldsymbol{P}_t^{t-1}\right)^{-1} \tag{29}$$

are computationally expensive due to matrix inversion, which scales with approx. $\mathcal{O}(d^3)$, where d is the dimension of the matrix. The constituents of the gains are $\boldsymbol{C}$, $\boldsymbol{R}$, $\boldsymbol{P}_t^{t-1}$, $\boldsymbol{P}_t^t$, and $\boldsymbol{A}$. Only the covariances of states, $\boldsymbol{P}_t^{t-1}$ and $\boldsymbol{P}_t^t$, vary with time t, but since only stationary processes are considered $\boldsymbol{P}_t^{t-1}$ and $\boldsymbol{P}_t^t$ are both constant for the whole time series. This implies constant values for $\boldsymbol{K}_t$, $\boldsymbol{J}_t$, $\boldsymbol{P}_t^n$ and $\boldsymbol{P}_{t,t-1}^n$ also. Therefore, the Eqs. (10), (11), (15) and (16) are fixed point equations. Attempts to solve the equation analytically lead to a quadratic equation in matrices, known as discrete algebraic Riccati equation [6]. When the fixed points are reached, all remaining matrices, $\boldsymbol{J}_t$, $\boldsymbol{P}_t^n$, and $\boldsymbol{P}_{t,t-1}^n$, can be calculated analytically, leading to the following procedure.

- Iterate

$$\begin{aligned} \boldsymbol{P}_t^{t-1} &= \boldsymbol{A}\boldsymbol{P}_{t-1}^{t-1}\boldsymbol{A}^{\mathrm{T}} + \boldsymbol{Q}, \\ \boldsymbol{P}_t^t &= \boldsymbol{P}_t^{t-1} - \boldsymbol{K}_t\boldsymbol{C}\boldsymbol{P}_t^{t-1}, \\ \boldsymbol{K}_t &= \boldsymbol{P}_t^{t-1}\boldsymbol{C}^{\mathrm{T}}\left(\boldsymbol{C}\boldsymbol{P}_t^{t-1}\boldsymbol{C}^{\mathrm{T}} + \boldsymbol{R}\right)^{-1}. \end{aligned} \tag{30}$$

up to certain accuracy. Start with $\boldsymbol{P}_0^0 = \boldsymbol{P}_0^n$ from the last iterations.

- Calculate

$$\begin{aligned}
&\boldsymbol{J} = \boldsymbol{P}_t^t \boldsymbol{A}^T \left(\boldsymbol{P}_t^{t-1}\right)^{-1},\\
&\text{vec}\boldsymbol{P}_t^n = (\boldsymbol{I} - \boldsymbol{J} \otimes \boldsymbol{J})^{-1} \text{vec}(\boldsymbol{P}_t^t - \boldsymbol{J}\boldsymbol{P}_t^{t-1}\boldsymbol{J}^{\mathrm{T}}),\\
&\text{vec}\boldsymbol{P}_{t,t-1}^n = (\boldsymbol{I} - \boldsymbol{J} \otimes \boldsymbol{J})^{-1} \text{vec}\left((\boldsymbol{I} - \boldsymbol{J}\boldsymbol{A})\boldsymbol{P}_t^t \boldsymbol{J}^T\right).
\end{aligned} \tag{31}$$

Here, vec denotes the vec-operation and $\otimes$ the Kronecker product [10]. Since Eq. (30) and (31) do not depend on observations, these quantities can be used for all panels. Note that the covariance matrices $\boldsymbol{P}_t^t$, $\boldsymbol{P}_t^{t-1}$, $\boldsymbol{P}_t^n$ and $\boldsymbol{P}_{t,t-1}^n$ do not depend on time t anymore. Still, the superscripts remain to distinguish between them.

- Filter the observations, possibly in parallel for all panels. Use $\boldsymbol{x}_0^n$ from last iteration as initial value for $\boldsymbol{x}_0^0$.

$$\boldsymbol{x}_t^t = \boldsymbol{K}\boldsymbol{y}_t + (\boldsymbol{I} - \boldsymbol{K}\boldsymbol{C})\boldsymbol{A}\boldsymbol{x}_{t-1}^{t-1}, \quad t = 1\ldots n, \tag{32}$$

$$\boldsymbol{x}_{t-1}^n = (\boldsymbol{I} - \boldsymbol{J}\boldsymbol{A})\boldsymbol{x}_{t-1}^{t-1} + \boldsymbol{J}\boldsymbol{x}_t^n, \quad t = n\ldots 1. \tag{33}$$

- Calculate

$$\boldsymbol{D} = n\boldsymbol{P}_{t-1}^n + \sum_{t=1}^{n}\left(\boldsymbol{x}_{t-1}^n {\boldsymbol{x}_{t-1}^n}^{\mathrm{T}}\right), \tag{34}$$

$$\boldsymbol{E} = n\boldsymbol{P}_{t,t-1}^n + \sum_{t=1}^{n}\left(\boldsymbol{x}_t^n {\boldsymbol{x}_{t-1}^n}^{\mathrm{T}}\right), \tag{35}$$

$$\boldsymbol{F} = n\boldsymbol{P}_t^n + \sum_{t=1}^{n}\left(\boldsymbol{x}_t^n {\boldsymbol{x}_t^n}^{\mathrm{T}}\right), \tag{36}$$

for each panel, and take the sum over panels to update $\boldsymbol{A}$, $\boldsymbol{Q}$, and $\boldsymbol{R}$, Eqs. (21)–(23).

3. Results and discussion

In this section the results of the simulation studies are presented. The first demonstrates the gain in run-time which is achieved with the proposed optimization, Section 3.1. The second shows the ability of the EM algorithm to estimate spectral properties of the process in the presence of strong observational noise, Section 3.2. In the third simulation, Section 3.2.2 absolute values of the process parameters together with their uncertainties are estimated from noisy data. The simulation studies demonstrate that the implementation discussed here is fast and correct.

3.1. Effect of run-time optimization

We compare three different implementations of the EM algorithm. The first and the second are naïve implementations in MATLAB® and C++, respectively. The third one is also implemented in C++, but with run-time optimization.

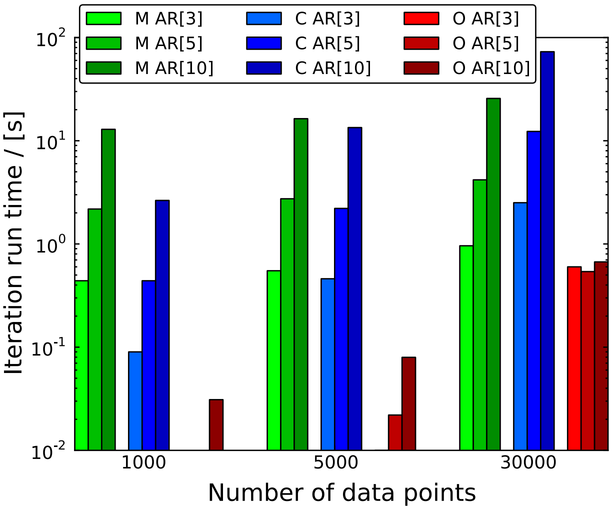


Fig. 2. Comparison of the mean run-time of one EM iteration. Naive implementation MATLAB® (M, green), Naive implementation C++ (C, blue), optimized implementation C++ (O, red). The shade of the color indicates the process order used for fitting. Lower is better. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The run-time test of different EM algorithm implementations is performed by fitting a state space model of process order 3, 5, and 10 to data generated from a three dimensional VAR[2]. All calculations were carried out on two 2.93 GHz Quad-Core Intel Xeon CPUs with 32 GB of RAM in total. The results are shown in Fig. 2. The number of data points included in the fit is shown on the x-axes. To rate the algorithms, the run-time of single EM iterations were measured. On the y-axes, the mean of 100 EM iterations is given in seconds. The order of the fitted process, 3, 5, 10, is indicated by the color shade of the bars from light to dark.

As described in Fig. 2, the naïve MATLAB® (green) and naïve C++ (blue) implementation perform comparable. The MATLAB® implementation scales better with the length of the time series. In contrast, the naïve C++ implementation is faster for shorter time series. The result of the optimized C++ implementation is shown in red. Taking the VAR[10] and 30,000 data points as example, the optimized implementation is faster by a factor of 100 compared to both naïve implementations. The exact run-times are given in Table 1.

3.2. Application to simulated data

Here we discuss two simulation studies which are concerned with the quality of the parameter estimates of the implementation discussed in this article. For the first study, spectral properties represented by the power spectrum were of concern. In the second study, the performance of the algorithm with respect to the absolute values of parameter estimates and their uncertainties were investigated. Since the true spectrum and parameters are known, the estimated properties can be compared to the true ones.

3.2.1. Estimating spectral properties

The process underlying the data is a VAR[3] with $d = 2$ and transition matrices

$$\boldsymbol{A}(1) = \begin{pmatrix} 0.9 & 0 \\ 0.35 & 0.7 \end{pmatrix}, \quad \boldsymbol{A}(2) = \begin{pmatrix} -0.5 & 0.1 \\ 0.2 & -0.3 \end{pmatrix}, \tag{37}$$

$$\boldsymbol{A}(3) = \begin{pmatrix} 0 & 0.15 \\ -0.25 & -0.4 \end{pmatrix}, \quad \boldsymbol{Q} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \tag{38}$$

From this process, a realization with 30,000 data points was obtained. Observational noise was added, such that the resulting signal-to-noise ratio is 1:8. The signal-to-noise ratio is defined as the variance of the process divided by the variance of the noise.

The EM algorithm was applied with three different thresholds for the relative changes of $\boldsymbol{A}$, 10^{-3}, 10^{-4}, 10^{-7} (Eq. (2.4.2)). By lowering the threshold, the fit gets more accurate. The run-time was 23 sec for 10^{-3}, 4.5 min for 10^{-4}, and about 22.5 h for 10^{-7}. The actual parameter values are only given for the lowest threshold of 10^{-7},

$$\hat{\boldsymbol{A}}(1) = \begin{pmatrix} 0.8727 & 0.0904 \\ 0.1258 & 0.5724 \end{pmatrix}, \quad \hat{\boldsymbol{A}}(2) = \begin{pmatrix} -0.5151 & -0.0168 \\ 0.4910 & -0.1966 \end{pmatrix}, \tag{39}$$

$$\hat{\boldsymbol{A}}(3) = \begin{pmatrix} 0.0039 & 0.2501 \\ -0.3371 & -0.4041 \end{pmatrix}, \quad \hat{\boldsymbol{Q}} = \begin{pmatrix} 1.1090 & 0.1222 \\ 0.1222 & 1.6585 \end{pmatrix}. \tag{40}$$

The initial values for this fit were set to the true value. The deviation can therefore be seen as the limit of accuracy of the algorithm.

Another approach to compare the estimated and true parameter values is the power spectrum $s(\omega)$, which is the power per frequency of a signal. This spectrum can be estimated by smoothing the periodogram. It can also be derived from VAR[p] parameters

Table 1
Results of the performance test: mean run-times of one EM iteration in seconds.

	Mean run-time in seconds			Data points
	VAR[2]	VAR[5]	VAR[10]	
naive MATLAB®	0.44	0.55	0.96	1000
	2.18	2.74	4.18	5000
	12.90	16.36	25.70	30,000
naive C++	0.09	.46	2.51	1000
	0.44	2.21	12.32	5000
	2.64	13.41	72.87	30,000
optimized C++	0.002	0.014	0.602	1000
	0.006	0.022	0.54	5000
	0.031	0.080	0.67	30,000

$$s(\omega)_{cc} = \frac{\sigma^2}{2\pi}\frac{1}{\|1-\sum_{\tau=1}^{p} a_{cc}(\tau)e^{-i\omega\tau}\|^2}, \tag{41}$$

with $a_{cc}(\tau)$ a diagonal entry of $\boldsymbol{A}(\tau)$. Since observational noise is assumed to be white, it manifests as a constant offset over all frequencies in the power spectrum. Removing it, the resulting power spectrum is expected to be of the same shape as the one with observational noise, but lower in amplitude. If the signal-to-noise ratio is too low however, only a flat spectrum characterizing the noise can be obtained.

In Fig. 3, power spectra from simulated data estimated using different approaches are compared. The results of the first dimension of the process are shown on the left, those for the second dimension on the right. The blue line is the smoothed periodogram of the signal. The smoothing window spanned over 0.01 phase/π. It does not resemble the hidden process' frequency content, since it is dominated by observational noise. The green line is the true spectrum calculated from the true parameter values, and is termed "Target". The red line is the spectrum which has been calculated from least-squares fitted parameters. It follows mainly the smoothed periodogram and no useful spectral information of the true process can be inferred from this estimate.

The turquoise, purple, and yellow lines are the power spectra which are based on the parameter estimates of the EM algorithm for the three convergence thresholds. Even for the lowest (10^{-3}) which is associated with the least accuracy, the peak frequency, in shape and position, was recovered. The area around the peak is the most important part of the spectrum, since it defines the main characteristics of the process. When the convergence criterion became more strict, the estimated power spectrum approached the true one. Even at a signal-to-noise ratio of 1:8, the power spectrum could be reliably estimated using the EM algorithm and Kalman filter with the proposed optimization.

3.2.2. Estimating parameters

To investigate the accuracy by means of absolute values of parameters, we used a $d = p = 2$ autoregressive process with parameter matrices

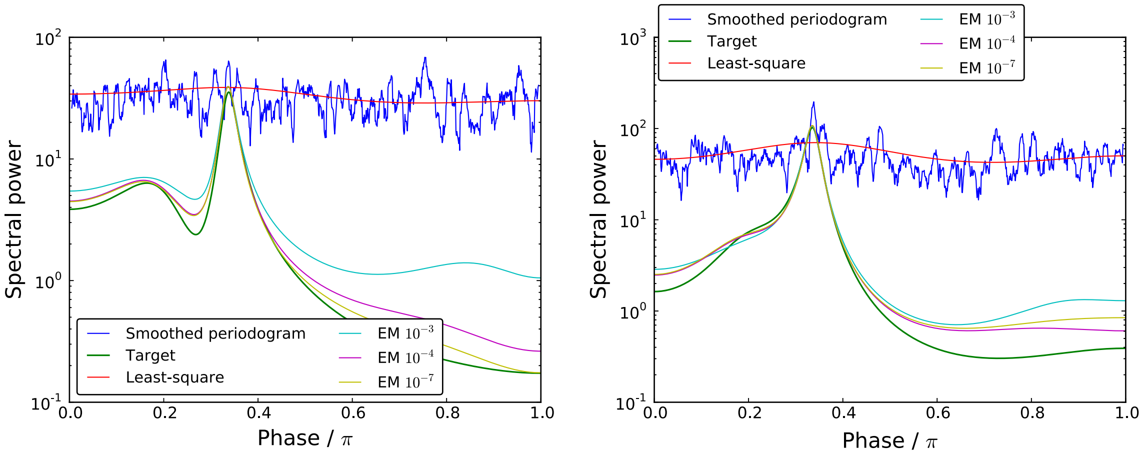
$$\boldsymbol{A}(1) = \begin{pmatrix} 1.3 & c \\ 0 & 1.7 \end{pmatrix}, \quad \boldsymbol{A}(2) = \begin{pmatrix} -0.8 & 0 \\ 0 & -0.8 \end{pmatrix}. \tag{42}$$

The coupling of the two dimensions of the process can be adjusted by the parameter c. This coupling parameter was varied from 0 to 0.5 in steps of 0.005. For every value of c, a time series of 5000 data points was obtained, and observational noise was added to get a signal to noise ration of 2:1.

Initial parameters

$$\boldsymbol{A}(1) = \begin{pmatrix} 0.8 & 0 \\ 0 & 0.8 \end{pmatrix}, \quad \boldsymbol{A}(2) = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}. \tag{43}$$

constituting a stationary AR process were chosen. Since the parameter c is changed for every realization obtained from the AR process, the initial values also vary with respect to each realization. Therefore, this simulation investigates, if it is possible



(a) AR3, first dimension

(b) AR3, second dimension

Fig. 3. Comparison of estimated power spectra. In blue, the smoothed periodogram and in red the spectrum calculated from least squares fitted VAR model is shown. Both are dominated by observational noise only. Calculated from the parameters of the fit with the EM algorithm are the spectra in turquoise, violet, and yellow. All of them show the shape of the true spectrum. Decreasing the convergence threshold leads to a better reconstruction of the true spectrum shown in green. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

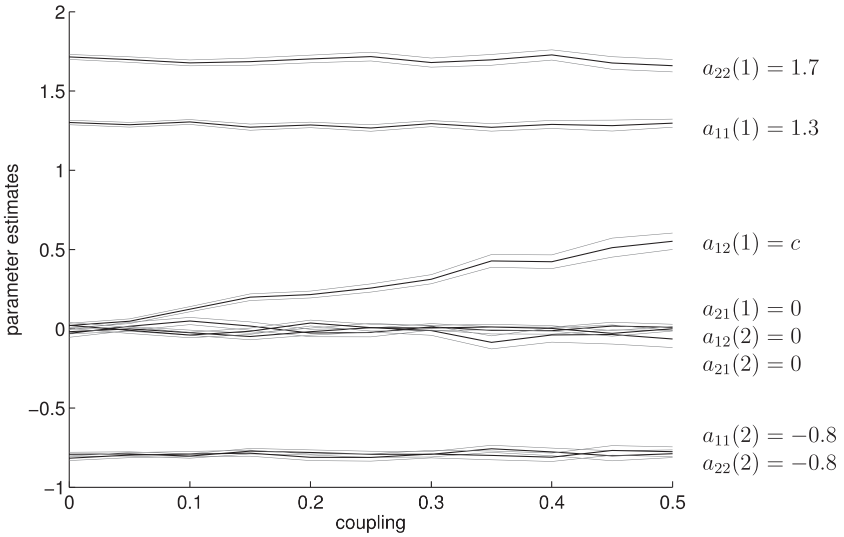


Fig. 4. Parameter estimates of the AR process obtained using the EM algorithm plotted in black. The coupling of the two dimensions of the process c is increased from 0 to 0.5 in steps of 0.05. The 1σ confidence interval is denoted by the gray lines. The parameters and their true values are denoted on the right.

to track parameter changes in the underlying system. It also explores if the estimated parameter values depend on initial values.

The results are presented in Fig. 4. The black lines are the parameter values as estimated for the corresponding value of c. In gray the 1σ confidence intervals of the estimates are given. The confidence interval was calculated from the Hessian of the log-likelihood. In Fig. 4, the entries of the transition matrices are labeled $a_{ij}(\tau)$, where τ denotes the time lag. As can be seen, the true parameters are within the error bounds of the estimated parameters almost all the times. Hence, we showed, that parameters are estimated correctly within their error bounds, that changes in parameters can be revealed, and that the parameter values estimated for VAR[p] systems do not depend on initial values.

4. Conclusion

This manuscript draws attention to the expectation maximization (EM) algorithm. The major drawback of this approach for everyday use is its high computational demand. As shown in this manuscript, the run-time can be optimized in the range of magnitudes. Tests fitting a VAR[10] to 3 dimensional time series with 30,000 data points show a gain in run-time of the factor 100 compared to naïve implementations.

Using simulated data we showed that spectral properties of a process as well as absolute values of process parameters and their errors can be estimated reliably with the implementation of the EM algorithm discussed here.

Whenever measurements are covered in observational noise, or different components of the process of interest can only be observed as combinations of each other, the state space model is worth considering. The implementation discussed in this article uses the identity matrix for observations, disallowing the observation of linear combinations of the hidden states. Still, instantaneous interactions are covered by the model by correlated driving noise corresponding to non-zero off-diagonal entries in the driving noise covariance matrix $\boldsymbol{Q}$. The EM algorithm using the Kalman filter offers a maximum likelihood estimator for the parameters of the state space model. Therefore, EM algorithm, Kalman filter, and state space model are highly effective tools in data analysis. A fast and easy to use implementation of this tool is available upon request form the authors.

Appendix A. Constraints of parameter matrices within EM-framework

Here, we show that the block-structure of $\boldsymbol{A}$ and $\boldsymbol{Q}$ of a VAR[p], which is formulated as a VAR[1], is maintained in the update rules of the EM-framework.

In order to update the $\boldsymbol{A}$ and $\boldsymbol{Q}$, the expected value of the likelihood G Eq. (19), has to be maximized. To proof that the update maintains the structure, Lagrange multipliers λ_{ij} and ω_{ij} were introduced into the likelihood, yielding Eq. (24). Only the derivative with respect to $\boldsymbol{Q}$ is shown. For $\boldsymbol{A}$, the same argument applies. The constrained update step is found by setting the derivative with respect to $\boldsymbol{Q}$ to zero.

$$\frac{\partial G^{c}}{\partial \boldsymbol{Q}} = \frac{\partial}{\partial \boldsymbol{Q}}\left(-\frac{n}{2}\log \boldsymbol{Q} - \frac{1}{2}\mathrm{tr}\left\{\boldsymbol{Q}^{-1}\left(\boldsymbol{F} - \boldsymbol{E}\boldsymbol{A}^{\mathrm{T}} - \boldsymbol{A}\boldsymbol{E}^{\mathrm{T}} + \boldsymbol{A}\boldsymbol{D}\boldsymbol{A}^{\mathrm{T}}\right)\right\} + \sum_{i,j}\omega_{ij}(Q_{ij} - \psi_{ij})\right) \tag{44}$$

$$= -\frac{n}{2}\boldsymbol{Q}^{-1}\frac{\partial \boldsymbol{Q}}{\partial \boldsymbol{Q}} - \frac{1}{2}\frac{\partial}{\partial \boldsymbol{Q}}\mathrm{tr}\left\{\boldsymbol{Q}^{-1}\boldsymbol{F} - \boldsymbol{Q}^{-1}\boldsymbol{E}\boldsymbol{A}^{\mathrm{T}} - \boldsymbol{Q}^{-1}\boldsymbol{A}\boldsymbol{E}^{\mathrm{T}} + \boldsymbol{Q}^{-1}\boldsymbol{A}\boldsymbol{D}\boldsymbol{A}^{\mathrm{T}}\right\} + \frac{\partial}{\partial \boldsymbol{Q}}\sum_{i,j}\omega_{ij}(Q_{ij} - \psi_{ij}) \tag{45}$$

$$= -\frac{n}{2}\boldsymbol{Q}^{-1}\frac{\partial \boldsymbol{Q}}{\partial \boldsymbol{Q}} - \frac{1}{2}\frac{\partial}{\partial \boldsymbol{Q}}\mathrm{tr}\left\{\boldsymbol{Q}^{-1}\boldsymbol{F} - \boldsymbol{Q}^{-1}\boldsymbol{E}\boldsymbol{A}^{\mathrm{T}} - \boldsymbol{Q}^{-1}\boldsymbol{A}\boldsymbol{E}^{\mathrm{T}} + \boldsymbol{Q}^{-1}\boldsymbol{A}\boldsymbol{D}\boldsymbol{A}^{\mathrm{T}}\right\} + \underbrace{\textstyle\sum_{i,j}\omega_{ij}\hat{\boldsymbol{e}}_{ij}}_{\boldsymbol{\Omega}} \tag{46}$$

$$\overset{!}{=} 0. \tag{47}$$

Multiplication by a factor 2/n as well as by $\boldsymbol{Q}$ from the left and the right yields

$$0 = \boldsymbol{Q} + \frac{1}{n}\left\{-\boldsymbol{F}^{\mathrm{T}} + \boldsymbol{A}\boldsymbol{E}^{\mathrm{T}} + \boldsymbol{E}\boldsymbol{A}^{\mathrm{T}} - \boldsymbol{A}\boldsymbol{D}^{\mathrm{T}}\boldsymbol{A}^{\mathrm{T}}\right\} - \frac{2}{n}\boldsymbol{Q}\boldsymbol{\Omega}\boldsymbol{Q} \tag{48}$$

The term in curly brackets is the parameter matrix obtained in the unconstrained case, i.e. without Lagrangian multipliers. Hence, the update $\boldsymbol{Q}$ with constraints is the same as the unconstrained update, if

$$\frac{2}{n}\boldsymbol{Q}\boldsymbol{\Omega}\boldsymbol{Q} = \boldsymbol{0}. \tag{49}$$

This is ensured, however, by the structure of $\boldsymbol{Q}$ and $\boldsymbol{\Omega}$

$$\boldsymbol{Q} = \begin{pmatrix} \boxed{(d \times d) = P} & 0 \\ 0 & 0 \end{pmatrix}, \quad \boldsymbol{\Omega} = \begin{pmatrix} \boxed{(d \times d) = 0} & P \\ P & P \end{pmatrix}. \tag{50}$$

Here, $\boldsymbol{P}$ denotes parameter entries in the respective matrix, while 0 denotes zero-entries. □

References

[1] J. Aldrich, RA Fisher and the making of maximum likelihood 1912–1922, Stat. Sci. 12 (3) (1997) 162–176.

[2] L. Baccala, K. Sameshima, Partial directed coherence: a new concept in neural structure determination, Biol. Cybernet. 84 (6) (2001) 463–474.

[3] A. Dempster, N. Laird, D. Rubin, Maximum likelihood from incomplete data via the EM algorithm, J. R. Stat. Soc. Ser. B (Methodological) (1977) 1–38.

[4] M. Eichler, A graphical approach for evaluating effective connectivity in neural systems, Philos. Trans. R. Soc. B: Biol. Sci. 360 (1457) (2005) 953.

[5] R. Kalman, A new approach to linear filtering and prediction problems, Trans. ASME-J. Basic Eng. 82 (1960) 35–45. Series D.

[6] P. Lancaster, L. Rodman, Algebraic Riccati Equations, Oxford University Press, USA, 1995.

[7] W. Mader, D. Feess, R. Lange, D. Saur, V. Glauche, C. Weiller, J. Timmer, B. Schelter, On the detection of direct directed information flow in fMRI, IEEE J. Sel. Top. Signal Process. 2 (6) (2008) 965–974.

[8] H. Nalatore, M. Ding, G. Rangarajan, Mitigating the effects of measurement noise on Granger causality, Phys. Rev. E 75 (3) (2007) 031123.

[9] H. Nalatore, M. Ding, G. Rangarajan, Denoising neural data with state-space smoothing: method and application, J. Neurosci. Methods 179 (1) (2009) 131–141.

[10] K. Petersen, M. Pedersen, The Matrix Cookbook, Technical University of Denmark, 2008.

[11] W. Press, S. Teukolsky, W. Vetterling, B. Flannery, Numerical recipes, third ed., The Art of Scientific Computing, Cambridge University Press, 2007.

[12] H. Rauch, F. Tung, C. Striebel, Maximum likelihood estimates of linear dynamic systems, AIAA 3 (8) (1965) 1445–1450.

[13] B. Schelter, J. Timmer, M. Eichler, Assessing the strength of directed influences among neural signals using renormalized partial directed coherence, J. Neurosci. Methods 179 (1) (2009) 121–130.

[14] F. Schweppe, Evaluation of likelihood functions for Gaussian signals, IEEE Trans. Inf. Theory 11 (1) (1965) 61–70.

[15] R. Shumway, D. Stoffer, An approach to time series smoothing and forecasting using the EM algorithm, J. Time Ser. Anal. 3 (4) (1982) 253–264.

[16] R. Shumway, D. Stoffer, Time Series Analysis and its Applications, Springer Verlag, 2000.

[17] G. Welch, G. Bishop, An Introduction to the Kalman Filter, Technical report, 1997.