
Article:
Kuzin, D., Isupova, O. and Mihaylova, L. (2018) Spatio-temporal structured sparse regression with hierarchical Gaussian process priors. IEEE Transactions on Signal Processing, 66 (17). pp. 4598-4611. ISSN: 1053-587X

https://doi.org/10.1109/TSP.2018.2858207

Spatio-Temporal Structured Sparse Regression with Hierarchical Gaussian Process Priors

Danil Kuzin, Olga Isupova, and Lyudmila Mihaylova, *Senior Member, IEEE*

***Abstract*—This paper introduces a new sparse spatio-temporal structured Gaussian process regression framework for online and offline Bayesian inference. This is the first framework that gives a time-evolving representation of the interdependencies between the components of the sparse signal of interest. A hierarchical Gaussian process describes such structure and the interdependencies are represented via the covariance matrices of the prior distributions. The inference is based on the expectation propagation method and the theoretical derivation of the posterior distribution is provided in the paper. The inference framework is thoroughly evaluated over synthetic, real video and electroencephalography (EEG) data where the spatio-temporal evolving patterns need to be reconstructed with high accuracy. It is shown that it achieves 15% improvement of the F-measure compared with the alternating direction method of multipliers, spatio-temporal sparse Bayesian learning method and one-level Gaussian process model. Additionally, the required memory for the proposed algorithm is less than in the one-level Gaussian process model. This structured sparse regression framework is of broad applicability to source localisation and object detection problems with sparse signals.**

I. Introduction

SPARSE regression problems arise often in various applications, e.g., compressive sensing [1], EEG source localisation [2] and direction of arrival estimation [3]. In all these applications, a dictionary of basis functions can be constructed that allows sparse representations of the signals of interest, i.e. many of the coefficients of the basis functions are close to zero. This allows to perform sensing tasks with lower amount of observations than the signal dimensionality. However, the signal recovery problem becomes more computationally expensive when sparsity assumptions are incorporated.

The sparse signal representation can be expressed as a regression problem of finding a signal $\mathbf{x}$ given the vector $\mathbf{y}$ of observations and the design matrix $\mathbf{A}$ that satisfies the equation

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \boldsymbol{\varepsilon}, \tag{1}$$

where $\boldsymbol{\varepsilon}$ is the Gaussian noise vector, $\boldsymbol{\varepsilon} \sim \mathcal{N}(\boldsymbol{\varepsilon}; \mathbf{0}, \sigma^2\mathbf{I})$, σ^2 is the variance and $\mathbf{I}$ is the identity matrix. Therefore, the observations also have a Gaussian distribution

$$\mathbf{y} \sim \mathcal{N}(\mathbf{y}; \mathbf{A}\mathbf{x}, \sigma^2\mathbf{I}). \tag{2}$$

When the number of observations is less than the number of coefficients the problem is ill-posed in the sense that it has an infinite number of possible solutions and additional regularisation is required. This is usually achieved by imposing l_p penalty functions with $0 \leq p < 2$ [4], [5], [6].

In the compressive sensing literature, it has been shown that if a matrix $\mathbf{A}$ satisfies the restricted isometry property (RIP) [7] then a solution of a convex l_1-minimisation problem is equivalent to a solution of a sparse l_0-minimisation problem. However, the problem of identification whether a given matrix satisfies the RIP is NP-hard [8]. In contrast, Bayesian models do not impose any restrictions on the matrix $\mathbf{A}$ and regularise the problem (1) with sparsity-inducing priors [9].

Bayesian models for sparse regression can be classified into models with a *weak sparsity* prior and a *strong sparsity* prior [10]. The weak sparsity prior leads to a unimodal posterior distribution of the signal with a sharp peak at zero, thus each coefficient has a high posterior probability of being close to zero. The strong sparsity prior is a mixture of latent binary variables that explicitly capture whether coefficients are zero or non-zero. In this paper we consider one type of strong sparsity priors — spike and slab models.

In spike and slab models, sparsity is achieved by selecting each component of $\mathbf{x}$ from a mixture of a spike distribution, that is the delta function, and a slab distribution, that is some flat distribution, usually a Gaussian with a large variance [11]. Following the Bayesian approach, latent variables that are indicators of spikes are added to the model [12] and a relevant distribution is placed over them [13]. Therefore, each signal component has an independent latent variable, which controls whether this component would be a spike or a slab.

In many applications, the independence assumption is not valid [14] as non-zero elements tend to appear in groups, and an unknown structure often exists in the field of the latent variables. For example, wavelet coefficients of images are usually organised in trees [15], chromosomes have a spatial structure along a genome [16], video from single-pixel cameras has a temporal structure [17]. In these cases it is useful to introduce additional hierarchical or group penalties that promote such structures in recovered signals.

A. Contributions

This paper proposes the spike and slab model with a hierarchical Gaussian process prior on the latent variables. Such hierarchical prior allows to model spatial structural dependencies for signal components that can evolve in time.

The model has a flexible structure which is governed only by the covariance functions of the Gaussian processes. This allows to model different types of structures and does not require any

D.Kuzin, L.Mihaylova are with the Department of Automatic Control and Systems Engineering, the University of Sheffield, Sheffield, UK e-mail: dkuzin1@sheffield.ac.uk, l.s.mihaylova@sheffield.ac.uk. O.Isupova is with the Department of Engineering Science, the University of Oxford, Oxford, UK e-mail: olga.isupova@eng.ox.ac.uk

specific knowledge about the structure such as determination of particular groups of coefficients with similar behaviour. If, however, there is information about the structure, it can be easily incorporated into the covariance functions. The model is flexible as spatial and temporal dependencies are decoupled by different levels of the hierarchical Gaussian process prior. Therefore, the spatial and temporal structures are modelled independently allowing to encode different assumptions for each type of structure. It allows to reduce complexity and process streaming data.

Overall, the main contributions of this work consist in:

1) the proposed novel spike and slab model with the hierarchical Gaussian process prior for signal recovery with spatio-temporal structural dependencies;
2) the developed Bayesian inference algorithm based on expectation propagation;
3) the novel online inference algorithm for streaming data based on Bayesian filtering;
4) a thorough validation and evaluation of the proposed method over synthetic and real data including the electrical activity data for the EEG source localisation problem and video data for the compressive background subtraction problem.

The paper is organised as follows. Section II reviews the related work. Section III provides an overview of existing spike and slab models. The proposed model and the inference algorithm are presented in Section IV. Section V demonstrates the online version of the algorithm. Section VI presents the complexity evaluation and numerical experiments. Section VII concludes the paper. Appendices provide theoretical derivations of the inference algorithm.

II. Related work

Different spatial structure assumptions for sparse models have been extensively studied in the literature. The group lasso [18], [19] extends the classical lasso method for group sparsity such that coefficients form groups and all coefficients in a group are either non-zero or zero together, but groups are required to be defined in advance. In contrast to group lasso, structural dependencies in our model are defined by the parameters of covariance functions of the Gaussian processes (GPs) and the actual groups are inferred from the data.

Group constraints for weak sparse models include smooth relevance vector machines [20], spatio-temporal coupling of the parameters for the scale mixture of Gaussians representation [21], [22], row and element sparsity [23], block sparsity [24].

For spike and slab priors a spatio-temporal structure is modelled with a one-level Gaussian processes prior [25], where the prior is imposed on all locations of non-zero components together. The covariance matrix is represented as the Kronecker product of the temporal and spatial matrices.

In contrast to the one-level GP our model introduces an additional level of a GP prior for temporal dependencies. Therefore, the temporal and spatial structures are decoupled. The proposed model is thus more flexible. Broadly speaking, the top-level GP can encode the slow change of groups of spikes positions in time while the low-level GP allows to model the local changes of each group. The one-level GP prior model also requires significantly more memory to store the covariance function for modelling both spatial and temporal structural dependencies as it is built as a Kronecker product of spatial and temporal covariance matrices. The resulting size of the covariance matrix scales quadratically with spatio-temporal dimensionality, which makes it infeasible even for average size problems, whereas for our model the total size of two covariance matrices scales linearly.

More importantly, in the proposed model structural dependencies are considered at every timestamp whereas in [25] the GP prior is imposed on the whole batch of data. This consideration of every timestamp allows us to develop an incremental inference algorithm — all latent variables are inferred for the new time moment in the similar manner as for the offline inference. Meanwhile, it is unclear how to apply the one-level GP model to the incremental data without reprocessing the previous data.

GPs are widely used to model complex structures and dynamics in data not only in sparse problems. In [26] GP is used as a prior for nonlinear state transition and observation functions for state-space Bayesian filtering. Hierarchical GP models are proposed to model structures in [27].

III. Sparse models for structured data

This section presents a roadmap of models that are used in the formulation of the proposed spatio-temporal structured sparse model. It starts from the basic spike and slab model and continues with its extension for structured data.

The generative model for the spatio-temporal regression problem can be formulated in the following way:

- The data is collected for the sequence of the T discrete timestamps. Indexes are denoted by $t \in [1, \dots, T]$.
- At each timestamp t the unknown signal of size N is denoted by $\mathbf{x}_t = [x_{1t}, \dots, x_{Nt}]^\top$. Signals at all timestamps are concatenated into a matrix $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_T]$.
- The observations of size K are denoted by $\mathbf{y}_t = [y_{1t}, \dots, y_{Kt}]^\top$. They are obtained with the design matrix $\mathbf{A} \in \mathbb{R}^{K \times N}$. Observations at all timestamps are concatenated into matrix $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_T]$.
- An independent Gaussian noise with the variance σ^2 is added to the observations.

The probabilistic model can be then expressed as

$$p(\mathbf{y}_t|\mathbf{x}_t) = \mathcal{N}(\mathbf{y}_t; \mathbf{A}\mathbf{x}_t, \sigma^2\mathbf{I}) \quad \forall t. \tag{3}$$

It is assumed that the dimensionality K of observations $\mathbf{y}_t$ is less than the dimensionality N of signals $\mathbf{x}_t$, therefore the problem of recovery of signal $\mathbf{x}_t$ from observations $\mathbf{y}_t$ is underdetermined and it can have an infinite number of solutions. Sparsity-inducing priors allow to specify additional constraints that lead to a unique optimal solution.

A. Factor graphs

For Bayesian models, factor graphs are used to visualise complex distributions [28] in a form of undirected graphical

models. They are also important for the approximate inference method described in Section IV.

The joint probability density function $p(\cdot)$ of latent variables ζ_i can be factorised as a product of factors ψ_C that are functions of a corresponding set of latent variables $\boldsymbol{\zeta}_C$

$$p(\zeta_1, ..., \zeta_m) = \frac{1}{Z} \prod_C \psi_C(\boldsymbol{\zeta}_C), \tag{4}$$

where Z is a normalisation constant. This factorisation can be represented as a bipartite graph with variable vertices corresponding to ζ_i, factor vertices corresponding to ψ_C and edges connecting corresponding vertices.

The distribution of latent variables $\mathbf{x}_t$ in (3) can be represented as a factor

$$g_t(\mathbf{x}_t) = \mathcal{N}(\mathbf{y}_t; \mathbf{A}\mathbf{x}_t, \sigma^2\mathbf{I}). \tag{5}$$

The factor graphs are used in this paper to visualise different spike and slab models. In Fig. 1 – 3 circles represent variable vertices and small squares represent factor vertices.

B. Spike and slab model

Sparsity can be induced with the spike and slab model [29], where additional latent variables $\boldsymbol{\Omega} = \{\omega_{it}\}_{t=1:T,\, i=1:N}$ indicate if signal components x_{it} are zeros. This is represented as a mixture of a spike and a slab

$$p(x_{it}|\omega_{it}) = \omega_{it}\delta_0(x_{it}) + (1 - \omega_{it})\mathcal{N}(x_{it}; 0, \sigma_x^2), \tag{6}$$

where spike $\delta_0(\cdot)$ is the delta function centered at zero, and slab is the Gaussian distribution with the variance σ_x^2. The conditional distributions $p(x_{it}|\omega_{it})$ are further denoted by factors $f_{it}(\omega_{it}, x_{it})$.

In this model $\{\omega_{it}\}_{i=1:N}$ are considered conditionally independent given $\mathbf{x}_t$. The prior is imposed on the indicators

$$p(\omega_{it}) = \text{Ber}(\omega_{it}; z), \tag{7}$$

where $\text{Ber}(\cdot; z)$ denotes a Bernoulli distribution with the success probability parameter z. The prior distributions $p(\omega_{it})$ are further denoted by $h_{it}^{\text{ind}}(\omega_{it})$. The problem (5) – (7) can be solved independently for each t.

The model can be represented as a factor graph (Fig. 1) with a product of factors (5) – (7) for all t and i.

The posterior $p(\mathbf{X}, \boldsymbol{\Omega})$ of latent variables $\mathbf{X}$ and $\boldsymbol{\Omega}$ is

$$p = \prod_{t=1}^{T} \left[g_t(\mathbf{x}_t) \prod_{i=1}^{N} \left[f_{it}(\omega_{it}, x_{it}) h_{it}^{\text{ind}}(\omega_{it}) \right] \right]. \tag{8}$$

C. Spike and slab model with a spatial structure

A spatial structure can be implemented by adding interdependencies for the locations of spikes in x_{it} [25], [30], [31]. This is achieved by modelling the probabilities of spikes with the additional latent variables $\boldsymbol{\Gamma} = [\boldsymbol{\gamma}_1, \ldots, \boldsymbol{\gamma}_T] = \{\gamma_{it}\}_{t=1:T,\, i=1:N}$ that are samples from a Gaussian process. A Gaussian process is a way to specify prior on functions, it can be defined as an infinite expansion of multivariate Gaussian distribution. In GP all finite subsets of variables have a joint Gaussian distribution. The properties of the structure are defined through

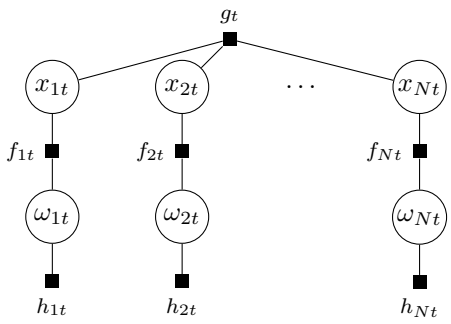


Fig. 1. Spike and slab model for one time moment (different time moments are independent). All signal components are conditionally independent given data, therefore structural assumptions cannot be modelled.

the covariance function of GP, which in this paper is assumed to be squared exponential:

$$p(\boldsymbol{\gamma}_t) = \mathcal{N}(\boldsymbol{\gamma}_t; \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_0),\ \boldsymbol{\Sigma}_0(i,j) = \alpha_\Sigma \exp\left(-\frac{(i-j)^2}{2\ell_\Sigma^2}\right), \tag{9}$$

where $\boldsymbol{\mu}_t$ is the mean vector and $\boldsymbol{\Sigma}_0$ is the covariance matrix with the hyperparameters α_Σ and ℓ_Σ^2.

The conditional independence assumption for ω_{it} from (7) is replaced by

$$p(\omega_{it}|\gamma_{it}) = \text{Ber}(\omega_{it}; \Phi(\gamma_{it})), \tag{10}$$

$$p(\boldsymbol{\gamma}_t) = \mathcal{N}(\boldsymbol{\gamma}_t; \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_0), \tag{11}$$

where $\Phi(\cdot)$ is the standard Gaussian cumulative distribution function (cdf). Scaling is required to normalise probabilities to the $[0, 1]$ interval and it is convenient to use $\Phi(\cdot)$ for this purpose in the derivations with GPs [32]. The conditional distributions $p(\omega_{it}|\gamma_{it})$ are denoted by factors $h_{it}(\omega_{it}, \gamma_{it})$. The prior distributions $p(\boldsymbol{\gamma}_t)$ are denoted by $r_t^{\text{ind}}(\gamma_t)$.

In this model $\{\boldsymbol{\gamma}_t\}_{t=1:T}$ are independent and therefore the problem can be solved separately for each timestamp. Using the introduced factors (5), (6) and (10) – (11), factor graph can be built as in Figure 2. The posterior $p(\mathbf{X}, \boldsymbol{\Omega}, \boldsymbol{\Gamma})$ of the latent variables is given by

$$p = \prod_{t=1}^{T} \left[g_t(\mathbf{x}_t) \prod_{i=1}^{N} \left[f_{it}(\omega_{it}, x_{it}) h_{it}(\omega_{it}, \gamma_{it}) \right] r_t(\boldsymbol{\gamma}_t) \right]. \tag{12}$$

IV. The Proposed Spatio-Temporal Structured Spike and Slab Model

In this paper a spatio-temporal latent structure of the positions of non-zero signal components is considered for the underdetermined recovery problem (3). The following assumptions are introduced:

1) $\mathbf{x}_t$ is sparse, i.e. it contains a lot of zeros for each timestamp t;
2) non-zero elements in $\mathbf{x}_t$ are clustered in groups for each timestamp t;
3) these groups can move and evolve in time.

This recovery problem is addressed with the hierarchical Bayesian approach. As in Section III-B, the first assumption

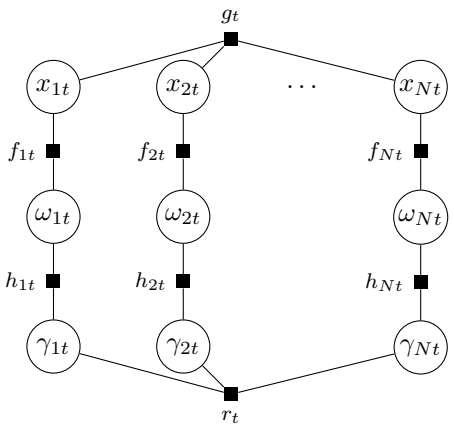


Fig. 2. Spike and slab model with a spatial structure for one time moment. The locations of spikes have a GP distribution, therefore encouraging a structure in space, but they are independent in time.

can be implemented in the model using the spike and slab prior (6).

Similarly to Section III-C, the second model assumption can be implemented by adding spatial dependencies for the positions of spikes in x_{it}. This is achieved by modelling the probabilities of spikes $\mathbf{\Omega}$ with the scaled GP on $\mathbf{\Gamma}$ (10), (11). GPs specify a prior over an unknown structure. This is particularly useful as it allows to avoid a specification of any structural patterns — the only parameter for structural modelling is the GP covariance function.

The third condition is addressed with the dynamic hierarchical GP prior. The mean $\mathbf{M} = [\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_T]$ for the spatial GP evolves over time according to the top-level temporal GP

$$\boldsymbol{\mu}_t \sim \mathcal{N}(\boldsymbol{\mu}_t; \boldsymbol{\mu}_{t-1}, \mathbf{W}),\ \mathbf{W}(i,j) = \alpha_W \exp\left(-\frac{(i-j)^2}{2\ell_W^2}\right), \tag{13}$$

where $\mathbf{W}$ is the squared exponential covariance matrix of the temporal GP with the hyperparameters α_W and ℓ_W^2.

This allows to implicitly specify the prior over the evolution function of the structure. The rate of the evolution is controlled with the top-level GP covariance function.

According to these assumptions, the model can be expressed as a factor graph (Figure 3) where the factor $r_t(\boldsymbol{\gamma}_t, \boldsymbol{\mu}_t)$ denotes $\mathcal{N}(\boldsymbol{\gamma}_t; \boldsymbol{\mu}_t, \mathbf{\Sigma}_0)$ and the factor $u_t(\boldsymbol{\mu}_t, \boldsymbol{\mu}_{t-1})$ denotes $\mathcal{N}(\boldsymbol{\mu}_t; \boldsymbol{\mu}_{t-1}, \mathbf{W})$.

The full posterior distribution $p(\mathbf{X}, \mathbf{\Omega}, \mathbf{\Gamma}, \mathbf{M})$ is then

$$p = \prod_{t=1}^{T}\left[g_t(\mathbf{x}_t)\prod_{i=1}^{N}\left[f_{it}(x_{it},\omega_{it})h_{it}(\omega_{it},\gamma_{it})\right] r_t(\boldsymbol{\gamma}_t, \boldsymbol{\mu}_t)\right] \times \prod_{t=2}^{T} u_t(\boldsymbol{\mu}_t, \boldsymbol{\mu}_{t-1}). \tag{14}$$

The exact posterior for the proposed hierarchical spike and slab model is intractable, therefore approximate inference methods should be used. In this paper expectation propagation (EP) [33] is employed. EP is shown to be the most effective Bayesian inference method for sparse modelling [34].

In this section the description of the EP method and the key components of the inference for the proposed model are presented. The details of the inference algorithm can be found in the appendices.

A. Expectation propagation

EP is a deterministic inference method that approximates the posterior distribution using the factor decomposition (4), where each factor is approximated with distributions $\tilde{\psi}_C(\cdot)$ from the exponential family:

$$\tilde{p}(\zeta_1, ..., \zeta_m) = \frac{1}{\tilde{Z}}\prod_C \tilde{\psi}_C(\zeta_C), \tag{15}$$

where $\tilde{p}$ is an approximating distribution and $\tilde{Z}$ is a normalisation constant. Approximating factorised distribution is determined by minimisation of the Kullback-Leibler (KL) divergence with the true distribution. The KL-divergence is a common measure of similarity between distributions.

Direct approximation is intractable due to intractability of the true posterior. Minimisation of the KL divergence between individual factors ψ_C and $\tilde{\psi}_C$ may not provide good approximation for the resulted product. In EP, approximation of each factor is performed in the context of other factors to improve a result for the final product. Iteratively one of the factors is chosen for refinement. The chosen factor $\tilde{\psi}_C$ is refined to minimise the KL-divergence between the product $q \propto \tilde{\psi}_C \prod_{C' \neq C} \tilde{\psi}_{C'}$ and $\psi_C \prod_{C' \neq C} \tilde{\psi}_{C'}$, where the approximating factor is replaced with a factor from the true posterior.

Factor refinement consists of five steps which are summarised below (with details given in Appendices B-E).

1) *Compute a cavity distribution* $q^{\setminus C} \propto \frac{q}{\tilde{\psi}_C}$: the joint distribution without the factor $\tilde{\psi}_C$
2) *Compute a tilted distribution* $\psi_C q^{\setminus C}$: the product of the cavity distribution and the true factor
3) *Refine the approximation* q: $q^* = \operatorname{argmin} \mathrm{KL}\left(\psi_C q^{\setminus C} || q\right)$ by minimising the KL-divergence between the tilted distribution $\psi_C q^{\setminus C}$ and the approximating distribution q. This is equivalent to matching the moments of the distributions [33].
4) *Compute an updated factor* $\tilde{\psi}_C^{\text{new}} \propto \frac{q^*}{q^{\setminus C}}$ using the refined approximation and cavity distribution.
5) *Update the current joint posterior* $q^{\text{new}} \propto \tilde{\psi}_C^{\text{new}} \prod_{C' \neq C} \tilde{\psi}_{C'}$ with the newly updated factor $\tilde{\psi}_C^{\text{new}}$.

B. Approximating factors

Here the key components of the EP inference algorithm for the proposed model are provided. The true posterior p (14) is approximated with the distribution q

$$q = \prod_t q_{g_t} q_{f_t} q_{h_t} q_{r_t} q_{u_t}, \tag{16}$$

where each factor q_a, $a \in \{g_t, f_t, h_t, r_t, u_t\}$, is from the exponential family and all latent variables are separated in the factors.

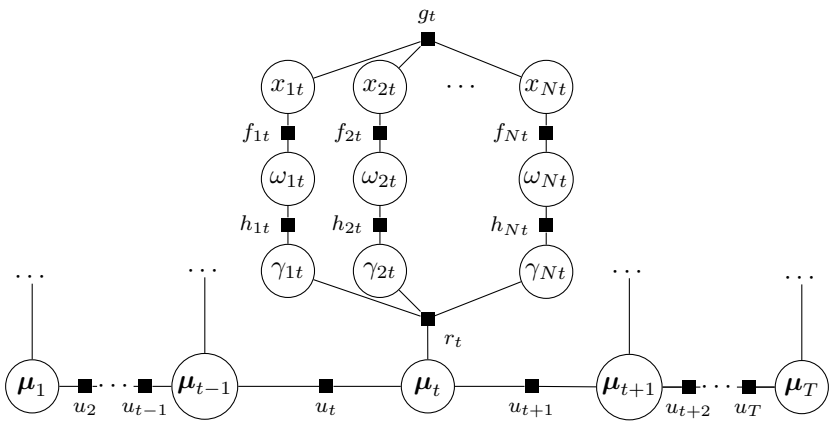


Fig. 3. Proposed spike and slab model with a spatio-temporal structure. The locations of spikes have a GP distribution in space with parameters that are controlled by a top-level GP and they evolve in time, therefore promoting temporal dependence.

Below the factors q_a of the approximating posterior q are introduced. Gaussian and Bernoulli distributions are used in the factors, which parameters are updated during the iterations of the EP algorithm.

The factors $g_t = \mathcal{N}(\mathbf{y}_t; \mathbf{A}\mathbf{x}_t, \sigma^2\mathbf{I})$ from (5) can be viewed as the distributions of $\mathbf{x}_t$ with fixed observed variables $\mathbf{y}_t$: $q_{g_t} = \mathcal{N}(\mathbf{x}_t; \mathbf{m}_{g_t}, \mathbf{V}_{g_t})$, where $\mathbf{m}_{g_t} = (\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{A}^\top\mathbf{y}_t$, $\mathbf{V}_{g_t} = \sigma^2(\mathbf{A}^\top\mathbf{A})^{-1}$.

The factors $f_t = \prod_{i=1}^N f_{it}$ and $h_t = \prod_{i=1}^N h_{it}$ from (6) and (10) are approximated with the products of Gaussian and Bernoulli distributions

$$q_{f_t} = \mathcal{N}(\mathbf{x}_t; \mathbf{m}_{f_t}, \mathbf{V}_{f_t}) \prod_{i=1}^N \mathrm{Ber}(\omega_{it}; \Phi(z_{f_{it}})), \quad (17)$$

$$q_{h_t} = \mathcal{N}(\boldsymbol{\gamma}_t; \boldsymbol{\nu}_{h_t}, \mathbf{S}_h) \prod_{i=1}^N \mathrm{Ber}(\omega_{it}; \Phi(z_{h_{it}})), \quad (18)$$

where the components of $\mathbf{x}_t$ and $\boldsymbol{\gamma}_t$ are independent. Therefore, the covariance matrices $\mathbf{V}_{f_t}$ and $\mathbf{S}_h$ are diagonal[1]. Distribution parameters $\mathbf{m}_{f_t}$, $\mathbf{V}_{f_t}$, $z_{f_{it}}$, $\boldsymbol{\nu}_{h_t}$, $\mathbf{S}_h$, and $z_{h_{it}}$ are updated during EP iterations according to Appendices B and C.

The approximation for the factors $r_t = \mathcal{N}(\boldsymbol{\gamma}_t; \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_0)$ and $u_t = \mathcal{N}(\boldsymbol{\mu}_t; \boldsymbol{\mu}_{t-1}, \mathbf{W})$ from (9) and (13) is intended to separate the latent variables and it is represented as products of Gaussian distributions

$$q_{r_t} = \mathcal{N}(\boldsymbol{\gamma}_t; \boldsymbol{\nu}_{r_t}, \mathbf{S}_r)\mathcal{N}(\boldsymbol{\mu}_t; \mathbf{e}_{r_t}, \mathbf{D}_r), \quad (19)$$

$$q_{u_t} = \mathcal{N}(\boldsymbol{\mu}_{t-1}; \mathbf{e}_{u_t\leftarrow}, \mathbf{D}_{u\leftarrow})\mathcal{N}(\boldsymbol{\mu}_t; \mathbf{e}_{u_t\rightarrow}, \mathbf{D}_{u\rightarrow}). \quad (20)$$

Distribution parameters $\mathbf{e}_{r_t}$, $\mathbf{D}_r$, $\boldsymbol{\nu}_{r_t}$, $\mathbf{S}_r$, $\mathbf{e}_{u_t\leftarrow}$, $\mathbf{D}_{u\leftarrow}$, $\mathbf{e}_{u_t\rightarrow}$, and $\mathbf{D}_{u\rightarrow}$ are updated during EP iterations according to Appendices D and E.

The posterior approximation q given by (16) thus contains the products of Gaussian and Bernoulli distributions that are equal to unnormalised Gaussian and Bernoulli distributions, respectively (Appendix A). This can be conveniently expressed in terms of the natural parameters and q can be represented in terms of distributions of the latent variables.

For $\mathbf{x}_t$ in q this product property leads to the Gaussian distribution $\mathcal{N}(\mathbf{x}_t; \mathbf{m}_t, \mathbf{V}_t)$ with natural parameters

$$\mathbf{V}_t^{-1} = \mathbf{V}_{g_t}^{-1} + \mathbf{V}_{f_t}^{-1},\ \mathbf{V}_t^{-1}\mathbf{m}_t = \mathbf{V}_{g_t}^{-1}\mathbf{m}_{g_t} + \mathbf{V}_{f_t}^{-1}\mathbf{m}_{f_t}. \quad (21)$$

Similarly, $\boldsymbol{\gamma}_t$ in q is distributed as $\mathcal{N}(\boldsymbol{\gamma}_t; \boldsymbol{\nu}_t, \mathbf{S})$, where natural parameters are

$$\mathbf{S}^{-1} = \mathbf{S}_h^{-1} + \mathbf{S}_r^{-1},\ \mathbf{S}^{-1}\boldsymbol{\nu}_t = \mathbf{S}_h^{-1}\boldsymbol{\nu}_{h_t} + \mathbf{S}_r^{-1}\boldsymbol{\nu}_{r_t}. \quad (22)$$

The top GP latent variables $\boldsymbol{\mu}_t$ have the Gaussian distributions $\mathcal{N}(\boldsymbol{\mu}_t; \mathbf{e}_t, \mathbf{D})$ with natural parameters

$$\mathbf{D}^{-1} = \mathbf{D}_r^{-1} + \mathbf{D}_{u\rightarrow}^{-1}\mathbb{1}_{t>1} + \mathbf{D}_{u\leftarrow}^{-1}\mathbb{1}_{t<T}, \quad (23a)$$

$$\mathbf{D}^{-1}\mathbf{e}_t = \mathbf{D}_r^{-1}\mathbf{e}_{r_t} + \mathbf{D}_{u\rightarrow}^{-1}\mathbf{e}_{u_t\rightarrow}\mathbb{1}_{t>1} + \mathbf{D}_{u\leftarrow}^{-1}\mathbf{e}_{u_{t+1}\leftarrow}\mathbb{1}_{t<T}, \quad (23b)$$

where $\mathbb{1}$ is the indicator function.

The distributions for $\boldsymbol{\omega}_t$ are $\prod_{i=1}^N \mathrm{Ber}(\omega_{it}; \Phi(z_{it}))$ with the parameters

$$z_{it} = \Phi^{-1}\left(\left[\frac{(1-\Phi(z_{f_{it}}))(1-\Phi(z_{h_{it}}))}{\Phi(z_{f_{it}})\Phi(z_{h_{it}})} + 1\right]^{-1}\right). \quad (24)$$

The full approximating posterior q is then

$$q = \prod_{t=1}^T \mathcal{N}(\mathbf{x}_t; \mathbf{m}_t, \mathbf{V}_t) \prod_{t=1}^T\prod_{i=1}^N \mathrm{Ber}(\omega_{it}; \Phi(z_{it})) \times \prod_{t=1}^T \mathcal{N}(\boldsymbol{\gamma}_t; \boldsymbol{\nu}_t, \mathbf{S}) \prod_{t=1}^T \mathcal{N}(\boldsymbol{\mu}_t; \mathbf{e}_t, \mathbf{D}). \quad (25)$$

In the EP inference algorithm, each of the introduced approximating factors q_{f_t}, q_{h_t}, q_{r_t}, q_{u_t} is iteratively updated according to the factor refinement procedure as in Section IV-A. Note that the factors q_{g_t} are not updated, as the corresponding factors g_t from the true posterior distribution are already from the exponential family.

[1]Note that $\mathbf{S}_h$ does not depend on time. In this paper, single covariance matrices are used for all time moments for both GP variables $\boldsymbol{\gamma}$ and $\boldsymbol{\mu}$ in the approximating factors. However, the method can be applied with individual covariance matrices for each time moment as well.

C. Implementation details

There are no theoretical guarantees of EP convergence. However, it can be achieved using *damping* [35]: during step 4 of the factor refinement procedure in Section IV-A the factor is updated as $q_a^{\text{damp}} = (q_a^{\text{new}})^{\eta}(q_a^{\text{old}})^{1-\eta}$, where q_a^{old} is the value of the factor from the previous iteration, q_a^{new} is the updated value of the factor, $\eta \in (0, 1]$ is the damping coefficient. It is exponentially decreased as $\eta = \eta^{\text{old}}\xi$ after each iteration, where $\xi \in (0, 1]$ is the parameter that governs the speed of exponential decrease and η^{old} is the value of the damping coefficient from the previous iteration.

It is also known that during the EP updates negative variances can appear [34]. In this case negative variances are replaced with a large value representing $+\infty$.

V. Online Inference with Bayesian Filtering

In this section the problem (3) is considered for streaming data, i.e. when new data becomes available at every timestamp. The conventional batch inference can be infeasible for large or streaming data. The developed online Bayesian filtering algorithm for the model presented in Section IV allows to iteratively update the approximation of $\mathbf{x}$ based on new samples of data.

Bayesian filtering consist of two steps that are iterated for each new sample of data:

- *prediction*, where an estimate of a hidden system state at the next time step is predicted based on the observations available at the current time moment;
- *update*, where this estimate is updated once an observation at the next time moment is obtained.

In the proposed model the hidden state is represented by the latent variables $\mathbf{x}_t$, $\boldsymbol{\omega}_t$, $\boldsymbol{\gamma}_t$ and $\boldsymbol{\mu}_t$ that should be inferred based on observations $\mathbf{y}_t$.

A. Prediction

At the prediction step for the timestamp $t+1$ the current estimate of the posterior distribution of the latent variables $p(\mathbf{x}_t, \boldsymbol{\omega}_t, \boldsymbol{\gamma}_t, \boldsymbol{\mu}_t|\mathbf{y}_{1:t})$ is available. It is based on all observations $\mathbf{y}_{1:t} = [\mathbf{y}_1, \ldots, \mathbf{y}_t]$ up to the timestamp t. The initial estimate of this posterior can be obtained by the offline inference algorithm applied to the initial T_{init} timestamps.

Marginalisation of the latent variables for the current timestamp t allows to obtain predictions for the latent variables for the next timestamp $t+1$

$$\begin{aligned} &p(\mathbf{x}_{t+1}, \boldsymbol{\omega}_{t+1}, \boldsymbol{\gamma}_{t+1}, \boldsymbol{\mu}_{t+1}|\mathbf{y}_{1:t}) = \\ &\quad = \int p(\mathbf{x}_{t+1}, \boldsymbol{\omega}_{t+1}, \boldsymbol{\gamma}_{t+1}, \boldsymbol{\mu}_{t+1}|\mathbf{x}_t, \boldsymbol{\omega}_t, \boldsymbol{\gamma}_t, \boldsymbol{\mu}_t) \\ &\qquad \times p(\mathbf{x}_t, \boldsymbol{\omega}_t, \boldsymbol{\gamma}_t, \boldsymbol{\mu}_t|\mathbf{y}_{1:t}) \mathrm{d}\mathbf{x}_t \mathrm{d}\boldsymbol{\omega}_t \mathrm{d}\boldsymbol{\gamma}_t \mathrm{d}\boldsymbol{\mu}_t \end{aligned} \tag{26}$$

The first term in the integral (26) is factorised according to the generative model (5),(6),(10), and (13)

$$\begin{aligned} &p(\mathbf{x}_{t+1}, \boldsymbol{\omega}_{t+1}, \boldsymbol{\gamma}_{t+1}, \boldsymbol{\mu}_{t+1}|\mathbf{x}_t, \boldsymbol{\omega}_t, \boldsymbol{\gamma}_t, \boldsymbol{\mu}_t) \\ &= p(\mathbf{x}_{t+1}|\boldsymbol{\omega}_{t+1}) p(\boldsymbol{\omega}_{t+1}|\boldsymbol{\gamma}_{t+1}) p(\boldsymbol{\gamma}_{t+1}|\boldsymbol{\mu}_{t+1}) p(\boldsymbol{\mu}_{t+1}|\boldsymbol{\mu}_t) \end{aligned} \tag{27}$$

Therefore, the terms related to variables $\mathbf{x}_{t+1}$, $\boldsymbol{\omega}_{t+1}$ and $\boldsymbol{\gamma}_{t+1}$ are independent from the integral variables in (26) and the integral can be rewritten as

$$\begin{aligned} &\int p(\mathbf{x}_{t+1}, \boldsymbol{\omega}_{t+1}, \boldsymbol{\gamma}_{t+1}, \boldsymbol{\mu}_{t+1}|\mathbf{x}_t, \boldsymbol{\omega}_t, \boldsymbol{\gamma}_t, \boldsymbol{\mu}_t) \\ &\quad \times p(\mathbf{x}_t, \boldsymbol{\omega}_t, \boldsymbol{\gamma}_t, \boldsymbol{\mu}_t|\mathbf{y}_{1:t}) \mathrm{d}\mathbf{x}_t \mathrm{d}\boldsymbol{\omega}_t \mathrm{d}\boldsymbol{\gamma}_t \mathrm{d}\boldsymbol{\mu}_t \\ &= p(\mathbf{x}_{t+1}|\boldsymbol{\omega}_{t+1}) p(\boldsymbol{\omega}_{t+1}|\boldsymbol{\gamma}_{t+1}) p(\boldsymbol{\gamma}_{t+1}|\boldsymbol{\mu}_{t+1}) \\ &\quad \times \int p(\boldsymbol{\mu}_{t+1}|\boldsymbol{\mu}_t) p(\boldsymbol{\mu}_t|\mathbf{y}_{1:t}) \mathrm{d}\boldsymbol{\mu}_t \end{aligned} \tag{28}$$

The initial estimate of the posterior $p(\boldsymbol{\mu}_{T_{\text{init}}}|\mathbf{y}_{1:T_{\text{init}}})$ obtained from the offline EP algorithm is a Gaussian distribution:

$$p(\boldsymbol{\mu}_{T_{\text{init}}}|\mathbf{y}_{1:T_{\text{init}}}) = \mathcal{N}(\boldsymbol{\mu}_{T_{\text{init}}}; \mathbf{e}_{1:T_{\text{init}}}, \mathbf{D}_{1:T_{\text{init}}}), \tag{29}$$

where $\mathbf{e}_{1:T_{\text{init}}}$ and $\mathbf{D}_{1:T_{\text{init}}}$ are the mean and the covariance matrix of the estimate of the posterior for $\boldsymbol{\mu}_{T_{\text{init}}}$ obtained based on observations $\mathbf{y}_{1:T_{\text{init}}}$.

According to the generative model (13) the first term of the integral in (28) is also Gaussian, therefore the integral is also a Gaussian distribution on $\boldsymbol{\mu}_{t+1}$ for $t = T_{\text{init}}$:

$$\begin{aligned} &\int p(\boldsymbol{\mu}_{t+1}|\boldsymbol{\mu}_t) p(\boldsymbol{\mu}_t|\mathbf{y}_{1:t}) \mathrm{d}\boldsymbol{\mu}_t \\ &= \mathcal{N}(\boldsymbol{\mu}_{t+1}; \mathbf{e}_{1:t}, \mathbf{D}_{1:t}^{\text{predict}}) \stackrel{\text{def}}{=} \hat{p}(\boldsymbol{\mu}_{t+1}), \end{aligned} \tag{30}$$

where $\mathbf{D}_{1:t}^{\text{predict}} = \mathbf{W} + \mathbf{D}_{1:t}$ is the covariance of the predicted distribution.

Substitution of (28) and (30) back into (26) provides the predicted distribution:

$$\begin{aligned} &p(\mathbf{x}_{t+1}, \boldsymbol{\omega}_{t+1}, \boldsymbol{\gamma}_{t+1}, \boldsymbol{\mu}_{t+1}|\mathbf{y}_{1:t}) \\ &= p(\mathbf{x}_{t+1}|\boldsymbol{\omega}_{t+1}) p(\boldsymbol{\omega}_{t+1}|\boldsymbol{\gamma}_{t+1}) p(\boldsymbol{\gamma}_{t+1}|\boldsymbol{\mu}_{t+1}) \hat{p}(\boldsymbol{\mu}_{t+1}) \end{aligned} \tag{31}$$

B. Update

At the update step the predicted distribution (31) of the latent variables for the next timestamp is corrected with the new data $\mathbf{y}_{t+1}$

$$\begin{aligned} &p(\mathbf{x}_{t+1}, \boldsymbol{\omega}_{t+1}, \boldsymbol{\gamma}_{t+1}, \boldsymbol{\mu}_{t+1}|\mathbf{y}_{1:t+1}) \\ &= \frac{1}{Z} p(\mathbf{y}_{t+1}|\mathbf{x}_{t+1}, \boldsymbol{\omega}_{t+1}, \boldsymbol{\gamma}_{t+1}, \boldsymbol{\mu}_{t+1}) \\ &\quad \times p(\mathbf{x}_{t+1}, \boldsymbol{\omega}_{t+1}, \boldsymbol{\gamma}_{t+1}, \boldsymbol{\mu}_{t+1}|\mathbf{y}_{1:t}) \\ &= \frac{1}{Z} p(\mathbf{y}_{t+1}|\mathbf{x}_{t+1}) p(\mathbf{x}_{t+1}|\boldsymbol{\omega}_{t+1}) p(\boldsymbol{\omega}_{t+1}|\boldsymbol{\gamma}_{t+1}) \\ &\quad \times p(\boldsymbol{\gamma}_{t+1}|\boldsymbol{\mu}_{t+1}) \hat{p}(\boldsymbol{\mu}_{t+1}), \end{aligned} \tag{32}$$

where Z is the normalisation constant.

Since components of the vectors $\mathbf{x}_{t+1}$ and $\boldsymbol{\omega}_{t+1}$ are conditionally independent, the terms $p(\mathbf{x}_{t+1}|\boldsymbol{\omega}_{t+1})$ and $p(\boldsymbol{\omega}_{t+1}|\boldsymbol{\gamma}_{t+1})$ are further factorised:

$$\begin{aligned} &p(\mathbf{x}_{t+1}, \boldsymbol{\omega}_{t+1}, \boldsymbol{\gamma}_{t+1}, \boldsymbol{\mu}_{t+1}|\mathbf{y}_{1:t+1}) \\ &= \frac{1}{Z} p(\mathbf{y}_{t+1}|\mathbf{x}_{t+1}) \left[\prod_{i=1}^{N} p(x_{it+1}|\omega_{it+1}) p(\omega_{it+1}|\gamma_{it+1}) \right] \\ &\quad \times p(\boldsymbol{\gamma}_{t+1}|\boldsymbol{\mu}_{t+1}) \hat{p}(\boldsymbol{\mu}_{t+1}), \end{aligned} \tag{33}$$

The resulting formula for update (33) is the same as the posterior distribution (14) with the only exception in the term

related to $\boldsymbol{\mu}_t$. The approximation of this posterior is proposed in Section IV. The algorithm is only required to be adjusted for the new factor $\hat{p}(\boldsymbol{\mu}_{t+1})$.

The factor $\hat{p}(\boldsymbol{\mu}_{t+1})$ is a Gaussian distribution, i.e. it is from the exponential family already and it only depends on a single latent variable, therefore this factor should not be updated in the EP iterations. The information from this factor will be passed through the general approximating distribution q to the other factors.

In the EP algorithm used for inference of the updated distribution (33) the distribution for $\boldsymbol{\mu}_t$ is approximated with the Gaussian distribution for any t. Therefore, the identity (30) is true for any t and the whole procedure can be applied for all timestamps.

C. Minibatch filtering

The developed Bayesian filtering procedure can be easily extended to the case of inferring minibatches for timestamps $[t+1 : t+M]$, where M is the size of a minibatch:

$$p(\mathbf{x}_{t+1:t+M}, \boldsymbol{\omega}_{t+1:t+M}, \boldsymbol{\gamma}_{t+1:t+M}, \boldsymbol{\mu}_{t+1:t+M} | \mathbf{y}_{1:t+M}) \quad (34)$$

rather than for the next timestamp $t+1$ only as in (33).

Indeed, due to conditional independence marginalisation (26) also comes down to integral (30) similar to (28). And the update step can also be performed by the EP algorithm with the only difference that it should be applied for M timestamps rather than one.

VI. Experiments

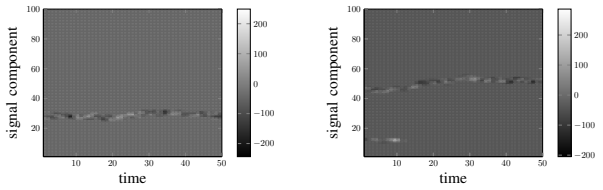
This section presents validation and evaluation results for the proposed algorithms. The performance of these two-level GP algorithms is compared with:

- the spatio-temporal spike and slab model with a one-level GP prior and its modification with common precision approximation [25];
- a popular alternating direction method of multipliers (ADMM) method [36], which is a convex optimisation method used here for the lasso problem [4];
- a spatio-temporal sparse Bayesian learning (STSBL) algorithm [37].

For quantitative comparison, the following measures are used:

- NMSE(normalised mean square error) $= \dfrac{\|\mathbf{X} - \widehat{\mathbf{X}}\|_F^2}{\|\mathbf{X}\|_F^2}$, where $\mathbf{X}$ is the true signal, $\widehat{\mathbf{X}}$ is the estimate, computed as the mean of the approximated posterior distribution, $\|\cdot\|_F$ is the Frobenius norm of a matrix;
- F-measure [13] $= 2\dfrac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$ between non-zero elements of the true signal $\mathbf{X}$ and non-zero elements of the estimate $\widehat{\mathbf{X}}$.

The NMSE shows the normalised error of signal reconstruction, with 0 corresponding to an ideal match. The F-measure shows how well slab locations are restored. An F-measure equal to 1 means that the true and estimated signals coincide, whilst 0 corresponds to lack of similarity between them. Arguably, for the sparse regression problem, the NMSE is less meaningful than the F-measure [38].



(a) Data (b) Data

Fig. 4. Examples of the true signal $\mathbf{X}$ for the synthetic data. In each example two groups of slabs generated at $t = 1$ evolve in time until $t = 50$.

Both two-level and one-level GP algorithms are iterated until convergence, which is measured by difference in the estimate of the signal $\widehat{\mathbf{X}}$ at the current and previous iterations.

A. Synthetic data

In this experiment, the algorithm performance is studied on synthetic data with known true values of signal $\mathbf{X}$ and slab locations $\boldsymbol{\Omega}$. The synthetic data represents the signals that have slowly evolving in time groups of non-zero elements. To create a spatio-temporal structure of slabs at the first timestamp $t = 1$ two groups of slab locations are generated with Poisson-distributed sizes for the signal $\mathbf{x}_t$ of dimensionality $N = 100$. Then, from $t = 2$ to $t = T = 50$, these groups randomly evolve: each border of each group can go up, down, or stay at the same location with such probabilities that in average the sparsity level remains 95%. In such way, locations of the slab groups are generated. The values of non-zero elements of the signal are then drawn from the distribution $\mathcal{N}(0, 10^4)$. This procedure is repeated 10 times to generate 10 data samples. The examples of generated $\mathbf{X}$ are shown in Fig. 4.

The elements of the design matrix $\mathbf{A}$ are generated as independent and identically distributed (iid) samples from the standard Gaussian. For each of the data samples, observations $\mathbf{Y} = \mathbf{AX}$ of different length K are generated. The value K/N is referred as an undersampling ratio. It changes from 10% to 55%.

The algorithms are evaluated in terms of average F-measure, NMSE and time[2] (Fig. 5) on this data. On the interval between 10% and 20% of the undersampling ratio both inference methods for the two-level GP model and full EP inference for the one-level GP model show competitive results in terms of the accuracy metrics while outperforming the other methods. On the interval between 20% and 30% of the undersampling ratio the inference methods for one- and two-level GP models are already able to perfectly reconstruct the sparse signal while both ADMM and STSBL show less accurate results. STSBL achieves the perfect reconstruction starting from the undersampling ratio 30% and ADMM achieves these results starting from the undersampling ratio 50%.

In the proposed EP algorithm for the two-level GP model (Section IV), the complexity of each iteration is $\mathcal{O}(N^3 T)$, as matrices of size $N \times N$ are inverted for each timestamp

[2]Time is evaluated with 4.2GHz Intel Core i7 CPU and 16GB RAM.

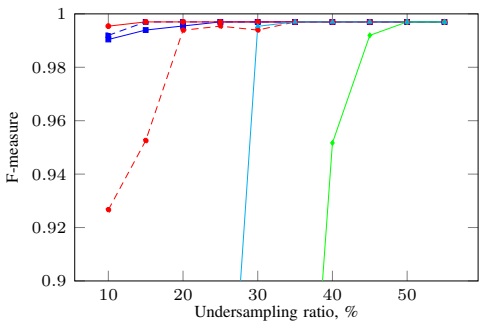
to compute cavity distributions for the factors u and r. In the proposed online inference algorithm (Section V), first the offline version is trained on size T_{init}. Then, when new data of size M is available, the previous results are used as prior and the complexity of update is $\mathcal{O}(N^3M)$, while in the offline version it is $\mathcal{O}(N^3(T_{\text{init}}+M))$.

On average, the proposed two-level GP algorithm requires similar to the full one-level GP algorithm number of iterations for convergence: approximately 30 iterations on the interval between 10% and 20% of the undersampling ratio, 15 iterations on the interval between 20% and 30%, and less than 10 iterations for the higher undersampling ratios. The approximate inference algorithm for the one-level GP model takes slightly more iterations to converge.

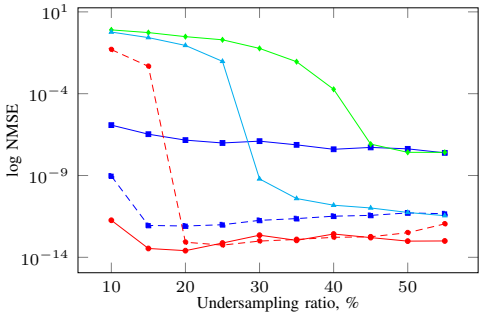
In the one-level GP algorithm [25] the complexity of one iteration is $\mathcal{O}(N^3T^3)$. This is related to inversion of full spatio-temporal covariance matrix. It is addressed with low rank and common precision approximations [25], which reduce both the computational complexity and the quality of the results. The K-rank approximation, where K is a parameter of the algorithm, reduces the computational complexity to $\mathcal{O}(N^2KT)$ and the common precision approximation reduces it to $\mathcal{O}(N^2T+T^2N)$.

In terms of the computational time the full EP inference for the one-level GP model is the slowest method. The approximated inference for the one-level GP model significantly improve its performance in terms of the computational time while also cause loss in accuracy. The ADMM method shows similar results to the approximated one-level GP model in terms of the computational time, but has even bigger loss in terms of both accuracy measures. The STSBL takes slightly more time for the lower values of the undersampling ratio, which helps it to achieve better results than the ADMM method in terms of the accuracy measures. The proposed offline and online inference methods for the two-level GP method demonstrate a satisfactory trade-off between computational time and accuracy. They obtain competitive results in terms of accuracy measures as the full EP inference for the one-level GP model while require significantly less computational time. In terms of computational time the proposed method demonstrates competitive results with the STSBL method.

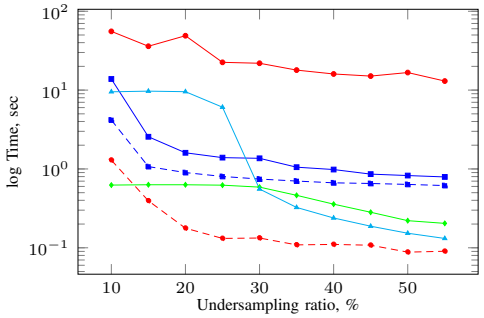
The proposed online inference method for the two-level GP model allows to save computational time while preserving the accuracy of the recovered signal. Note that the developed inference methods for the two-level GP model outperform competitors in the lowest undersampling ratio interval, i.e. they require less measurements to get the same quality as other algorithms.



(a) F-measure



(b) NMSE



(c) Time

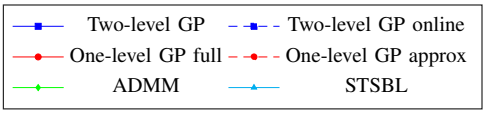
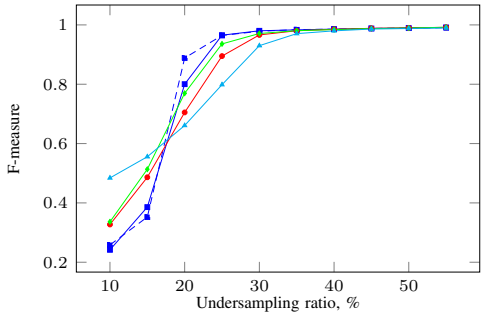


Fig. 5. Performance of the algorithms on the synthetic data. Note that the NMSE plots have logarithmic scale of y-axis. As the convergence criteria is $\frac{||\widehat{\mathbf{X}}^{\text{new}}-\widehat{\mathbf{X}}^{\text{old}}||_\infty}{||\widehat{\mathbf{X}}^{\text{old}}||_\infty} < 10^{-3}$, values below 10^{-3} are less significant. The proposed algorithms referred as two-level GP and two-level GP online outperform others in the $10-20\%$ interval, where the number of observations is the lowest.

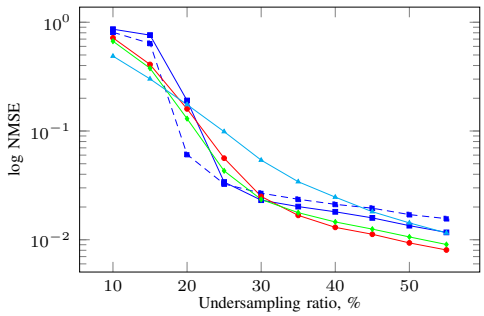
B. Real data: moving object detection in video

The considered methods for sparse regression are compared on the problem of object detection in video sequences. The Convoy dataset [39] is used where a background frame is subtracted from each video frame. As moving objects take only part of a frame the considered signal of the subtracted video frames is sparse. Moreover, objects are represented as clusters of pixels, which evolve in time. Therefore, the background subtraction application fully satisfies the proposed spatio-temporal structured model assumptions.

The frames with subtracted background are resized to 32×32 pixels and reshaped as vectors $\mathbf{x}_t \in \mathbb{R}^N$, $N = 1024$. The number of frames in the dataset is $T = 260$. The sparse observations are obtained as $\mathbf{Y} = \mathbf{AX}$, where $\mathbf{A} \in \mathbb{R}^{K\times N}$ is the matrix with iid Gaussian elements. 10 different random design matrices $\mathbf{A}$ are used to generate 10 data samples. The number of observations K is chosen such that the undersampling ratio K/N changes from 10% to 55%. This procedure corresponds



(a) F-measure



(b) NMSE

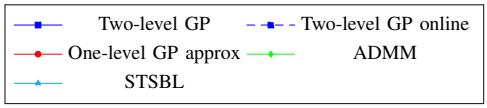


Fig. 6. Performance of the algorithms on the Convoy data. The proposed algorithms referred as two-level GP and two-level GP online outperform the others in the $20-30\%$ interval. On the interval $10-15\%$ all methods cannot reconstruct the true signal. The NMSE plot shows that the proposed algorithms underperform the competitors for the values higher than 30%, but the visual difference in performance becomes insignificant that is demonstrated in Fig. 7.

to compressive sensing observations [40].

For this problem the full EP inference for the one-level GP model is infeasible due to its memory requirements, therefore only the common precision approximated inference for the one-level GP model is considered.

The average F-measure and NMSE obtained by all the algorithms on the Convoy data are presented in Fig. 6. The proposed algorithm shows the best results for the undersampling ratio $20-30\%$. For larger values of the undersampling ratio all the algorithms provide close almost ideal results of reconstruction.

Fig. 7 presents the reconstructed sample frame from the Convoy data. For all the algorithms, the reconstruction results are provided for the undersampling ratio 10%, where the proposed algorithms slightly underperform the competitors in terms of the quality metrics, for the undersampling ratio 20%, where the proposed algorithm outperforms the competitors both in terms of NMSE and the F-measure, and for the undersampling ratio 40%, where the proposed algorithms show a little higher NMSE. It is clearly seen that for the undersampling ratio 10% the difference in the quality metrics is insignificant since none of the methods is able to reconstruct the signal. The STSBL represents an exceptional example but still the frame reconstructed by this method contains considerable amount of noise. For the undersampling ratio 20% the proposed method provides the clear reconstructed frame in contrast to the reconstructed frames by all the competitors that are more noisy. Meanwhile, for the undersampling ratio 40% the difference between reconstruction results by all four algorithms is not remarkable.

Note that similar to the synthetic data experiment the proposed algorithms obtain the best results for the lowest undersampling ratio values where the reconstruction is reasonable, i.e. they require a less number of observations.

C. Real data: EEG source localisation

The third experiment is devoted to the EEG source localisation problem.

The goal of the non-invasive EEG source localisation problem is to find 3D locations of dipoles such that their electromagnetic field coincides with the field measured by electrodes on the human head cortex. This is important, for example, for localisation of active areas in human-brain interfaces and treatment of neurological disorders [41], [42]. This problem is ill-posed in sense that there exist an infinite number of possible active areas inside the brain that could produce the same field on the head cortex. To regularise the problem, we use the idea that slab locations are distributed in space and temporally evolve, similar to [43]. Similar idea applies to the MEG source localisation [44].

Using the earlier introduced notation, the EEG source localisation problem is stated as

$$\mathbf{y}_t = \mathbf{A}\mathbf{x}_t + \boldsymbol{\varepsilon}_t, \quad \forall t \in [1, \dots, T], \tag{35}$$

where $\mathbf{y}_t \in \mathbb{R}^K$ is the vector containing observations of potential differences taken from $K = 69$ electrodes placed on a human head cortex, $\mathbf{A} \in \mathbb{R}^{K \times N}$ is the lead field matrix corresponding to $N/3 = 272$ voxels, $\mathbf{x}_t \in \mathbb{R}^N$ is the signal, that is the current density of dipole activation.

Here $\mathbf{x}_t$ represents the dipole moments corresponding to the grid locations:

$$\mathbf{x}_t = \left[x_{1x}, x_{1y}, x_{1z}, x_{2x}, x_{2y}, x_{2z}, \dots, x_{\frac{N}{3}z}\right]^\top. \tag{36}$$

For each grid voxel i inside the brain with location coordinates $loc(i) = (x_i, y_i, z_i)$ the corresponding dipole moments (x_{ix}, x_{iy}, x_{iz}) along the 3D axis are considered.

We employ the following covariance function that promotes close values for collinear dipole moments corresponding to close grid positions

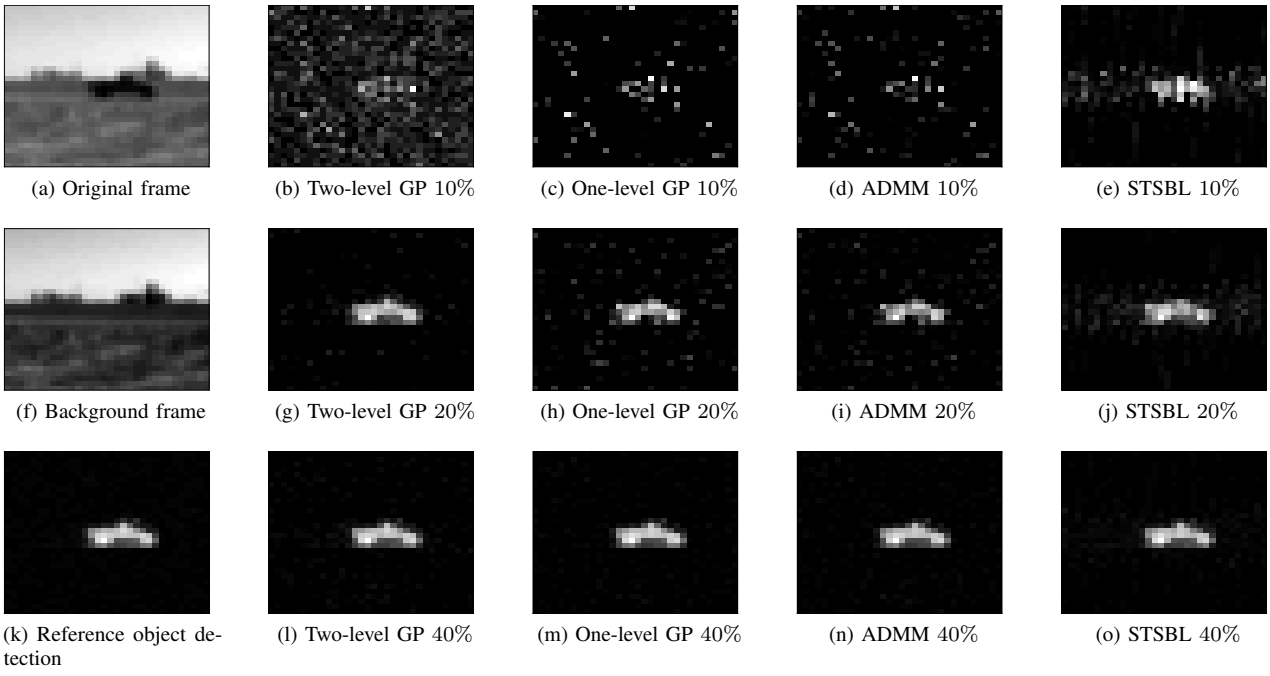
$$\mathbf{K}(i,j) = \alpha_{\mathbf{K}} \exp\left(-\frac{d(i,j)^2}{2\ell^2_{\mathbf{K}}}\right), \quad \mathbf{K} \in \{\boldsymbol{\Sigma}_0, \mathbf{W}\}, \tag{37}$$

where the distance is computed as

$$d(i,j) = \begin{cases} 0, \text{if axis for dipole moments } i,\ j \text{ are different} \\ ||loc(i) - loc(j)||_2^2, \text{otherwise.} \end{cases} \tag{38}$$

Hyperparameters are selected so that the sampled potential differences have the similar behaviour as the provided data.

The data and lead field matrix for the experiments is processed with EEGLAB [45]. We use the data provided in

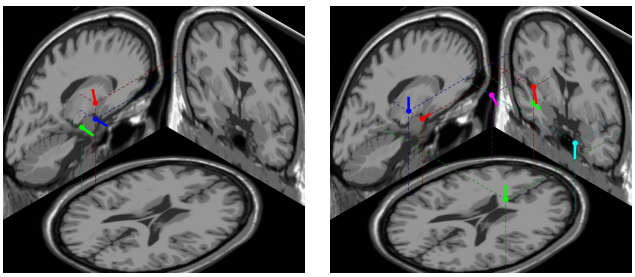


(a) Original frame (b) Two-level GP 10% (c) One-level GP 10% (d) ADMM 10% (e) STSBL 10%

(f) Background frame (g) Two-level GP 20% (h) One-level GP 20% (i) ADMM 20% (j) STSBL 20%

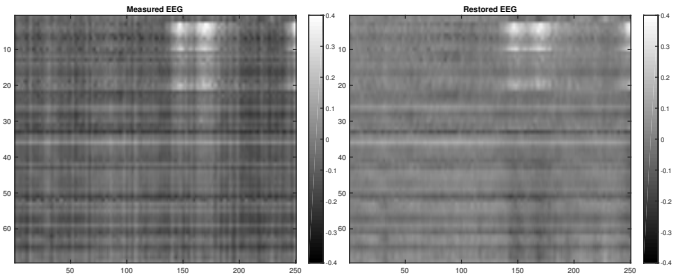
(k) Reference object detection (l) Two-level GP 40% (m) One-level GP 40% (n) ADMM 40% (o) STSBL 40%

Fig. 7. Sample frame with reconstruction results from sparse observations for the Convoy data. (a), (f): the original and static background non-compressed frames; (k): object detection results based on non-compressed frame difference (static background frame is subtracted from the original frame); (b), (g), (l): reconstruction of compressed object detection results based on the proposed online two-level GP method; (c), (h), (m): reconstruction of the compressed object detection results based on the one-level GP method; (d), (i), (n): reconstruction of the compressed object detection results based on the ADMM method; (e), (j), (o): reconstruction of the compressed object detection results based on the STSBL method. (b), (c), (d), and (e) show the results for the undersampling rate 10%, where all the algorithms fail to reconstruct the true signal. (g), (h), (i), and (j) show the reconstruction for the undersampling rate 20%, where the difference in performance between the algorithms is visible. While for the undersampling rate 40% ((l), (m), (n), and (o)) reconstruction results are indistinguishable in quality.



(a) Located dipole moments 1 ms after the event (b) Located dipole moments 170 ms after the event

Fig. 8. Located dipoles by the proposed offline two-level GP method for the EEG source localisation problem. There is no brain response immediately after the event and (a) demonstrates reconstructed brain active area that remains active during the whole period and it is not related to the event. While (b) shows the reconstructed active area when the brain response to the event is detected.



(a) Measured EEG (b) Reconstructed EEG ($\mathbf{A}\hat{\mathbf{X}}$)

Fig. 9. Reconstruction by the proposed offline two-level GP method of the EEG signal. As the true active dipole areas are not known, reconstruction quality is based on the observations $\mathbf{Y}$. Reconstructed EEG has lower magnitude, potentially because noise has been taken into account.

EEGLAB for the source localisation problem with annotated events.

Figure 8 presents located dipoles by the proposed method for the fourth event at two given time moments. The first time moment is taken right after the event happened and there is no response to it in the brain activity yet. The second time moment is chosen when the response is detected. Figure 9 shows the comparison of measured and restored potential differences by the proposed algorithm.

The true signal $\mathbf{X}$ is unknown for the EEG source localisation problem, therefore, NMSE between the observations $\mathbf{y}_t$ and reconstructed $\mathbf{A}\widehat{\mathbf{x}}_t$ is used for the quantitative comparison in this experiment. The obtained results for all the algorithms around the time of the brain response are presented in Fig. 10. The proposed two-level GP algorithms show the best results among the competitors. Both proposed offline and online inference methods demonstrate similar performance. Note that in this experiment the undersampling ratio is approximately 8%, which confirms that the proposed method is able to provide better results for lower values of the undersampling ratio.

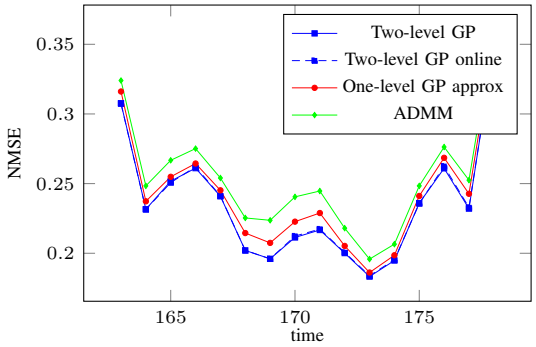


Fig. 10. Results for NMSE between $\mathbf{y}_t$ and $\mathbf{A}\widehat{\mathbf{x}}_t$ during the brain response time. The proposed algorithms referred as two-level GP and two-level GP online have the lowest NMSE among the others.

TABLE I
TWO-LEVEL GP HYPERPARAMETERS

Parameter	Synthetic	Convoy	EEG
σ_x^2	10^4	160	$4 * 10^5$
σ^2	10^{-4}	4	10^{-3}
η	0.999	0.99	0.9
ξ	0.9999	0.999	0.8
ℓ_W	15	15	22.17
ℓ_Σ	10	10	0.2217
α_W	10	10	10^{-2}
α_Σ	10	10	0.05

D. Parameters selection

For the proposed algorithm and for the one-level GP the parameters η and ξ are grid optimised to make the comparison fair. The prior shape hyperparameters ℓ_Σ, ℓ_W, α_Σ, α_W and variances σ_x^2 and σ^2 are specified so that sampled data has the same form as training data. ADMM and STSBL use the default values of parameters. The selected hyperparameter values for the proposed algorithm for all datasets are presented in Table I.

VII. CONCLUSIONS

This paper proposes a new hierarchical Gaussian process model of spatio-temporal structure representation with complex temporal evolution in sparse Bayesian inference methods. This is achieved using the flexible hierarchical GP prior for the spike and slab model, where spatial and temporal structural dependencies are encoded by different levels of the prior. Offline and online methods are developed for posterior inference for this model.

We show that the introduced model can be applied to different areas such as compressive sensing and EEG source localisation. The results show the superiority of the proposed method in comparison with the non-hierarchical GP method, the alternating direction method of multipliers and the spatio-temporal sparse Bayesian learning method. The developed algorithms demonstrate better performance both in terms of signal value reconstruction and localisation of non-zero signal components: within the low amount of measurements range it achieves around 15% improvement in terms of slab localisation quality.

Acknowledgments: The authors would like to thank the support from the EC Seventh Framework Programme [FP7 2013-2017] TRAcking in compleX sensor systems (TRAX) Grant agreement no.: 607400.

REFERENCES

[1] M. F. Duarte and Y. C. Eldar, "Structured compressed sensing: From theory to applications," *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4053–4085, 2011.
[2] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using FOCUSS: A re-weighted minimum norm algorithm," *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 600–616, 1997.
[3] J. Yin and T. Chen, "Direction-of-arrival estimation using a sparse representation of array covariance vectors," *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4489–4493, 2011.
[4] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.
[5] D. Malioutov, M. Çetin, and A. S. Willsky, "A sparse signal reconstruction perspective for source localization with sensor arrays," *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 3010–3022, 2005.
[6] A. Carmi, P. Gurfil, and D. Kanevsky, "Methods for sparse signal recovery using Kalman filtering with embedded pseudo-measurement norms and quasi-norms," *IEEE Transactions on Signal Processing*, vol. 58, no. 4, pp. 2405–2409, 2010.
[7] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203–4215, 2005.
[8] A. M. Tillmann and M. E. Pfetsch, "The computational complexity of the restricted isometry property, the nullspace property, and related concepts in compressed sensing," *IEEE Transactions on Information Theory*, vol. 60, no. 2, pp. 1248–1259, 2014.
[9] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *The Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.
[10] S. Mohamed, K. Heller, and Z. Ghahramani, "Bayesian and L1 approaches to sparse unsupervised learning," in *Proceedings of the 29th International Conference on Machine Learning*, 2012, pp. 751–758.
[11] T. J. Mitchell and J. J. Beauchamp, "Bayesian variable selection in linear regression," *Journal of the American Statistical Association*, vol. 83, no. 404, pp. 1023–1032, 1988.
[12] N. G. Polson and J. G. Scott, "Shrink globally, act locally: Sparse Bayesian regularization and prediction," *Bayesian Statistics*, vol. 9, pp. 501–538, 2010.
[13] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.
[14] F. Bach, R. Jenatton, J. Mairal, G. Obozinski *et al.*, "Structured sparsity through convex optimization," *Statistical Science*, vol. 27, no. 4, pp. 450–468, 2012.
[15] S. Mallat, *A wavelet tour of signal processing, third edition: the sparse way*, 3rd ed. Academic Press, 2008.
[16] T. Hastie, R. Tibshirani, and M. Wainwright, *Statistical learning with sparsity: the lasso and generalizations*. CRC Press, 2015.
[17] J. Yang, X. Yuan, X. Liao, P. Llull, D. Brady, G. Sapiro, and L. Carin, "Video compressive sensing using Gaussian mixture models," *IEEE Transactions on Image Processing*, vol. 23, no. 11, pp. 4863–4878, 2014.
[18] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, no. 1, pp. 49–67, 2006.
[19] P. Sprechmann, I. Ramirez, G. Sapiro, and Y. C. Eldar, "C-HiLasso: A collaborative hierarchical sparse modeling framework," *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4183–4198, 2011.
[20] A. Schmolck, "Smooth relevance vector machines," Ph.D. dissertation, University of Exeter, 2008.
[21] M. A. Van Gerven, B. Cseke, F. P. De Lange, and T. Heskes, "Efficient Bayesian multivariate fMRI analysis using a sparsifying spatio-temporal prior," *NeuroImage*, vol. 50, no. 1, pp. 150–161, 2010.
[22] A. Wu, M. Park, O. O. Koyejo, and J. W. Pillow, "Sparse Bayesian structure learning with dependent relevance determination priors," in *Advances in Neural Information Processing Systems 27*, 2014, pp. 1628–1636.

[23] W. Chen, D. Wipf, Y. Wang, Y. Liu, and I. J. Wassell, "Simultaneous Bayesian sparse approximation with structured sparse models," *IEEE Transactions on Signal Processing*, vol. 64, no. 23, pp. 6145–6159, 2016.
[24] Z. Zhang and B. D. Rao, "Sparse signal recovery with temporally correlated source vectors using sparse Bayesian learning," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 5, pp. 912–926, 2011.
[25] M. R. Andersen, A. Vehtari, O. Winther, and L. K. Hansen, "Bayesian inference for spatio-temporal spike and slab priors," *arXiv preprint arXiv:1509.04752*, 2015.
[26] M. Deisenroth and S. Mohamed, "Expectation propagation in Gaussian process dynamical systems," in *Advances in Neural Information Processing Systems*, 2012, pp. 2609–2617.
[27] N. D. Lawrence and A. J. Moore, "Hierarchical Gaussian process latent variable models," in *Proceedings of the 24th International Conference on Machine learning*, 2007, pp. 481–488.
[28] M. J. Wainwright and M. I. Jordan, "Graphical models, exponential families, and variational inference," *Foundations and Trends in Machine Learning*, vol. 1, no. 1–2, pp. 1–305, 2008.
[29] E. I. George and R. E. McCulloch, "Variable selection via Gibbs sampling," *Journal of the American Statistical Association*, vol. 88, no. 423, pp. 881–889, 1993.
[30] B. E. Engelhardt and R. P. Adams, "Bayesian structured sparsity from Gaussian fields," *ArXiv e-prints*, 2014.
[31] Q. Wu, Y. D. Zhang, M. G. Amin, and B. Himed, "High-resolution passive SAR imaging exploiting structured Bayesian compressive sensing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 8, pp. 1484–1497, 2015.
[32] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning*. The MIT Press, 2006.
[33] T. P. Minka, "Expectation propagation for approximate Bayesian inference," in *Proceedings of the 17th Conference on Uncertainty in Artificial Intelligence*, 2001, pp. 362–369.
[34] J. M. Hernandez-Lobato, D. Hernandez-Lobato, and A. Suarez, "Expectation propagation in linear regression models with spike-and-slab priors," *Machine Learning*, vol. 99, no. 3, pp. 437–487, 2015.
[35] T. Minka and J. Lafferty, "Expectation-propagation for the generative aspect model," in *Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence*, 2002, pp. 352–359.
[36] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.
[37] Z. Zhang, T.-P. Jung, S. Makeig, Z. Pi, and B. D. Rao, "Spatiotemporal sparse Bayesian learning with applications to compressed sensing of multichannel physiological signals," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, no. 6, pp. 1186–1197, 2014.
[38] B. Xin, Y. Wang, W. Gao, D. Wipf, and B. Wang, "Maximal sparsity with deep networks?" in *Advances in Neural Information Processing Systems*, 2016, pp. 4340–4348.
[39] G. Warnell, S. Bhattacharya, R. Chellappa, and T. Basar, "Adaptive-rate compressive sensing using side information," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 3846–3857, 2015.
[40] V. Cevher, A. Sankaranarayanan, M. F. Duarte, D. Reddy, and R. G. Baraniuk, "Compressive sensing for background subtraction," in *Proceedings of 10th European Conference on Computer Vision*, 2008, pp. 155–168.
[41] M. A. Jatoi, N. Kamel, A. S. Malik, I. Faye, and T. Begum, "A survey of methods used for source localization using EEG signals," *Biomedical Signal Processing and Control*, vol. 11, pp. 42–52, 2014.
[42] S. Baillet, J. C. Mosher, and R. M. Leahy, "Electromagnetic brain mapping," *IEEE Signal Processing Magazine*, vol. 18, no. 6, pp. 14–30, 2001.
[43] S. Baillet and L. Garnero, "A Bayesian approach to introducing anatomo-functional priors in the EEG/MEG inverse problem," *IEEE Transactions on Biomedical Engineering*, vol. 44, no. 5, pp. 374–385, 1997.
[44] A. Solin, P. Jylänki, J. Kauramäki, T. Heskes, M. A. van Gerven, and S. Särkkä, "Regularizing solutions to the MEG inverse problem using space-time separable covariance functions," *arXiv preprint arXiv:1604.04931*, 2016.
[45] A. Delorme and S. Makeig, "EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis," *Journal of Neuroscience Methods*, vol. 134, no. 1, pp. 9–21, 2004.

APPENDIX A
PRODUCT AND QUOTIENT RULES

EP updates are based on products and quotients of distributions. This section presents the product and quotient rules for Gaussian and Bernoulli distributions.

A. Product of Gaussians

A product of two Gaussian distributions is a unnormalised Gaussian distribution

$$\mathcal{N}(\mathbf{x};\mathbf{m}_1,\mathbf{\Sigma}_1)\mathcal{N}(\mathbf{x};\mathbf{m}_2,\mathbf{\Sigma}_2) \propto \mathcal{N}(\mathbf{x};\mathbf{m},\mathbf{\Sigma}),$$

where

$$\mathbf{\Sigma}^{-1} = \mathbf{\Sigma}_1^{-1} + \mathbf{\Sigma}_2^{-1},\ \mathbf{\Sigma}^{-1}\mathbf{m} = \mathbf{\Sigma}_1^{-1}\mathbf{m}_1 + \mathbf{\Sigma}_2^{-1}\mathbf{m}_2$$

B. Quotient of Gaussians

A quotient of two Gaussian distributions is a unnormalised Gaussian distribution[3]

$$\frac{\mathcal{N}(\mathbf{x};\mathbf{m}_1,\mathbf{\Sigma}_1)}{\mathcal{N}(\mathbf{x};\mathbf{m}_2,\mathbf{\Sigma}_2)} \propto \mathcal{N}(\mathbf{x};\mathbf{m},\mathbf{\Sigma}),$$

where

$$\mathbf{\Sigma}^{-1} = \mathbf{\Sigma}_1^{-1} - \mathbf{\Sigma}_2^{-1},\ \mathbf{\Sigma}^{-1}\mathbf{m} = \mathbf{\Sigma}_1^{-1}\mathbf{m}_1 - \mathbf{\Sigma}_2^{-1}\mathbf{m}_2$$

C. Product of Bernoulli

A product of two Bernoulli distributions is a unnormalised Bernoulli distribution

$$\text{Ber}(x;\Phi(z_1))\text{Ber}(x;\Phi(z_2)) \propto \text{Ber}(x;\Phi(t(z_1,z_2))),$$

where

$$t(z_1,z_2) = \Phi^{-1}\left(\left[\frac{(1-\Phi(z_1))(1-\Phi(z_2))}{\Phi(z_1)\Phi(z_2)}+1\right]^{-1}\right)$$

D. Quotient of Bernoulli

A quotient of two Bernoulli distributions is a unnormalised Bernoulli distribution

$$\frac{\text{Ber}(x;\Phi(z_1))}{\text{Ber}(x;\Phi(z_2))} \propto \text{Ber}(x;\Phi(d(z_1,z_2))),$$

where

$$d(z_1,z_2) = \Phi^{-1}\left(\left[\frac{(1-\Phi(z_1))\Phi(z_2)}{(1-\Phi(z_2))\Phi(z_1)}+1\right]^{-1}\right)$$

[3]Although quotient can lose positive semidefiniteness, we will still refer to it as a Gaussian distribution

APPENDIX B
EP UPDATE FOR FACTOR f_{it}

A. Cavity distribution

The unnormalised cavity distribution $q^{\backslash q_{f_{it}}}(x_{it},\omega_{it}) = \frac{q(x_{it},\omega_{it})}{q_{f_{it}}(x_{it},\omega_{it})}$ can be computed as

$$\begin{aligned} q^{\backslash q_{f_{it}}} &= \frac{\mathcal{N}(x_{it};\mathbf{m}_t(i),\mathbf{V}_t(i,i))\text{Ber}(\omega_{it};\Phi(z_{it}))}{\mathcal{N}(x_{it};\mathbf{m}_{f_t}(i),\mathbf{V}_{f_t}(i,i))\text{Ber}(\omega_{it};\Phi(z_{f_{it}}))} \\ &\propto \mathcal{N}(x_{it};m_{it}^{\backslash f},v_{it}^{\backslash f})\text{Ber}(\omega_{it};\Phi(z_{it}^{\backslash f})), \end{aligned}$$

where

$$\begin{aligned} (v_{it}^{\backslash f})^{-1} &= \mathbf{V}_t^{-1}(i,i) - \mathbf{V}_{f_t}^{-1}(i,i), \\ (v_{it}^{\backslash f})^{-1}m_{it}^{\backslash f} &= \mathbf{V}_t^{-1}(i,i)\mathbf{m}_t(i) - \mathbf{V}_{f_t}^{-1}(i,i)\mathbf{m}_{f_t}(i,i), \\ z_{it}^{\backslash f} &= z_{h_{it}} \end{aligned}$$

B. Moments matching

The moments of the tilted distribution $q^{\backslash q_{f_{it}}} f_{it}$ are

$$\begin{aligned} Z_{it} &= \Phi(z_{it}^{\backslash f})\mathcal{N}(0;m_{it}^{\backslash f},v_{it}^{\backslash f}) \\ &\quad + (1-\Phi(z_{it}^{\backslash f}))\mathcal{N}(0;m_{it}^{\backslash f},v_{it}^{\backslash f}+\sigma_x^2), \\ \mathbb{E}x_{it} &= \frac{1-\Phi(z_{it}^{\backslash f})}{Z_{it}}\mathcal{N}(0;m_{it}^{\backslash f},v_{it}^{\backslash f})\frac{m_{it}^{\backslash f}\sigma_x^2}{v_{it}^{\backslash f}+\sigma_x^2}, \\ \mathbb{E}x_{it}^2 &= \frac{1-\Phi(z_{it}^{\backslash f})}{Z_{it}}\mathcal{N}(0;m_{it}^{\backslash f},v_{it}^{\backslash f}) \\ &\quad \times\left(\frac{(m_{it}^{\backslash f})^2\sigma_x^4}{(v_{it}^{\backslash f}+\sigma_x^2)^2}+\frac{v_{it}^{\backslash f}\sigma_x^2}{v_{it}^{\backslash f}+\sigma_x^2}\right), \\ \mathbb{E}\omega_{it} &= \frac{\Phi(z_{it}^{\backslash f})}{Z_{it}}\mathcal{N}(0;m_{it}^{\backslash f},v_{it}^{\backslash f}) \end{aligned}$$

The new approximation $q^*(x_{it},\omega_{it})$ is

$$q^* = \mathcal{N}(x_{it};m_{it}^{q^*},v_{it}^{q^*})\text{Ber}(\omega_{it};\Phi(z_{it}^{q^*})),$$

where

$$m_{it}^{q^*} = \mathbb{E}x_{it},\ v_{it}^{q^*} = \mathbb{E}x_{it}^2 - (\mathbb{E}x_{it})^2,\ z_{it}^{q^*} = \Phi^{-1}(\mathbb{E}\omega_{it}).$$

C. Factor update

The new factor approximation $q_{f_{it}}^{\text{new}}(x_{it},\omega_{it}) = \frac{q^*(x_{it},\omega_{it})}{q^{\backslash q_{f_{it}}}(x_{it},\omega_{it})}$ can be computed as

$$\begin{aligned} q_{f_{it}}^{\text{new}} &= \frac{\mathcal{N}\left(x_{it};m_{it}^{q^*},v_{it}^{q^*}\right)\text{Ber}\left(\omega_{it};\Phi\left(z_{it}^{q^*}\right)\right)}{\mathcal{N}\left(x_{it};m_{it}^{\backslash f},v_{it}^{\backslash f}\right)\text{Ber}\left(\omega_{it};\Phi\left(z_{it}^{\backslash f}\right)\right)} \\ &\propto \mathcal{N}\left(x_{it};\mathbf{m}_{f_t}^{\text{new}}(i),\mathbf{V}_{f_t}^{\text{new}}(i,i)\right)\text{Ber}\left(\omega_{it};\Phi\left(z_{f_{it}}^{\text{new}}\right)\right), \end{aligned}$$

where

$$\begin{aligned} \left(\mathbf{V}_{f_t}^{\text{new}}\right)^{-1}(i,i) &= \left(v_{it}^{q^*}\right)^{-1} - \left(v_{it}^{\backslash f}\right)^{-1}, \\ \left(\mathbf{V}_{f_{it}}^{\text{new}}\right)^{-1}(i,i)\mathbf{m}_{f_t}^{\text{new}}(i) &= \left(v_{it}^{q^*}\right)^{-1}m_{it}^{q^*} - \left(v_{it}^{\backslash f}\right)^{-1}m_{f_{it}}^{\backslash f}, \\ z_{f_{it}}^{\text{new}} &= d\left(z_{it}^{q^*},z_{it}^{\backslash f}\right). \end{aligned}$$

APPENDIX C
EP UPDATE FOR FACTOR h_{it}

A. Cavity distribution

The unnormalised cavity distribution $q^{\backslash q_{h_{it}}}(\gamma_{it},\omega_{it}) = \frac{q(\gamma_{it},\omega_{it})}{q_{h_{it}}(\gamma_{it},\omega_{it})}$ can be computed as

$$\begin{aligned} q^{\backslash q_{h_{it}}} &= \frac{\mathcal{N}(\gamma_{it};\boldsymbol{\nu}_t(i),\mathbf{S}(i,i))\text{Ber}(\omega_{it};\Phi(z_{it}))}{\mathcal{N}(\gamma_{it};\boldsymbol{\nu}_{h_t}(i),\mathbf{S}_h(i,i))\text{Ber}(\omega_{it};\Phi(z_{h_{it}}))} \\ &\propto \mathcal{N}(\gamma_{it};\nu_{it}^{\backslash h},s_{it}^{\backslash h})\text{Ber}(\omega_{it};\Phi(z_{it}^{\backslash h})), \end{aligned}$$

where

$$\begin{aligned} (s_{it}^{\backslash h})^{-1} &= \mathbf{S}_t^{-1}(i,i) - \mathbf{S}_h^{-1}(i,i) \\ (s_{it}^{\backslash h})^{-1}\nu_{it}^{\backslash h} &= \mathbf{S}_t^{-1}(i,i)\boldsymbol{\mu}_t(i) - \mathbf{S}_h^{-1}(i,i)\boldsymbol{\nu}_{h_t}(i,i) \\ z_{it}^{\backslash h} &= z_{f_{it}} \end{aligned}$$

B. Moments matching

The moments of the tilted distribution $q^{\backslash q_{h_{it}}} h_{it}$ are

$$\begin{aligned} Z_{it} &= \Phi(z_{it}^{\backslash h})\Phi(a) + (1-\Phi(z_{it}^{\backslash h}))(1-\Phi(a)), \\ \mathbb{E}\gamma_{it} &= \frac{1}{Z_{it}}(\Phi(z_{it}^{\backslash h})K + (1-\Phi(z_{it}^{\backslash h}))(\nu_{it}^{\backslash h}-K)), \\ \mathbb{E}\gamma_{it}^2 &= \frac{1}{Z_{it}}\Bigg[(2\Phi(z_{it}^{\backslash h})-1)\Bigg((\nu_{it}^{\backslash h})^2\Phi(a) + s_{it}^{\backslash h}\Phi(a) \\ &\quad + \frac{2\nu_{it}^{\backslash h}s_{it}^{\backslash h}\mathcal{N}(a;0,1)}{\sqrt{1+s_{it}^{\backslash h}}} - \frac{(s_{it}^{\backslash h})^2 a\mathcal{N}(a;0,1)}{1+s_{it}^{\backslash h}}\Bigg) \\ &\quad + (1-\Phi(z_{it}^{\backslash h})(s_{it}^{\backslash h}+(\nu_{it}^{\backslash h})^2)\Bigg], \\ \mathbb{E}\omega_{it} &= \frac{\Phi(z_{it}^{\backslash h})\Phi(a)}{Z_{it}}, \end{aligned}$$

where

$$a = \frac{\nu_{it}^{\backslash h}}{\sqrt{1+s_{it}^{\backslash h}}}, \quad K = s_{it}^{\backslash h}\frac{\mathcal{N}(a;0,1)}{\sqrt{1+s_{it}^{\backslash h}}} + \nu_{it}^{\backslash h}\Phi(a)$$

The new approximation $q^*(\gamma_{it},\omega_{it})$ is

$$q^* = \mathcal{N}(\gamma_{it};\nu_{it}^{q^*},s_{it}^{q^*})\text{Ber}(\omega_{it};\Phi(z_{it}^{q^*})),$$

where

$$\nu_{it}^{q^*} = \mathbb{E}\gamma_{it},\ s_{it}^{q^*} = \mathbb{E}\gamma_{it}^2 - (\mathbb{E}\gamma_{it})^2,\ z_{it}^{q^*} = \Phi^{-1}(\mathbb{E}\omega_{it}).$$

C. Factor update

The new factor approximation $q_{h_{it}}^{\text{new}}(\gamma_{it},\omega_{it}) = \frac{q^*(\gamma_{it},\omega_{it})}{q^{\backslash q_{h_{it}}}(\gamma_{it},\omega_{it})}$ can be computed as

$$\begin{aligned} q_{h_{it}}^{\text{new}} &= \frac{\mathcal{N}\left(\gamma_{it};\nu_{it}^{q^*},s_{it}^{q^*}\right)\text{Ber}\left(\omega_{it};\Phi\left(z_{it}^{q^*}\right)\right)}{\mathcal{N}\left(\gamma_{it};\nu_{it}^{\backslash h},s_{it}^{\backslash h}\right)\text{Ber}\left(\omega_{it};\Phi\left(z_{it}^{\backslash h}\right)\right)} \\ &\propto \mathcal{N}\left(\gamma_{it};\boldsymbol{\nu}_{h_t}^{\text{new}}(i),\mathbf{S}_h^{\text{new}}(i,i)\right)\text{Ber}\left(\omega_{it};\Phi\left(z_{h_{it}}^{\text{new}}\right)\right), \end{aligned}$$

where

$$\left(\mathbf{S}_h^{\text{new}}\right)^{-1}(i,i) = \left(s_{it}^{q^*}\right)^{-1} - \left(s_{it}^{\backslash h}\right)^{-1},$$

$$(\mathbf{S}_h^{\text{new}})^{-1}(i,i)\boldsymbol{\nu}_{h_t}^{\text{new}}(i) = \left(s_{it}^{q^*}\right)^{-1}\nu_{it}^{q^*} - \left(s_{it}^{\setminus h}\right)^{-1}\nu_{it}^{\setminus h},$$
$$z_{h_{it}}^{\text{new}} = d\left(z_{it}^{q^*}, z_{it}^{\setminus h}\right).$$

APPENDIX D
EP UPDATE FOR FACTOR r_t

A. Cavity distribution

The unnormalised cavity distribution $q^{\setminus q_{r_t}}(\boldsymbol{\gamma}_t, \boldsymbol{\mu}_t) = \frac{q(\boldsymbol{\gamma}_t, \boldsymbol{\mu}_t)}{q_{r_t}(\boldsymbol{\gamma}_t, \boldsymbol{\mu}_t)}$ can be computed as

$$\begin{aligned} q^{\setminus q_{r_t}} &= \frac{\mathcal{N}(\boldsymbol{\gamma}_t; \boldsymbol{\nu}_t, \mathbf{S})\mathcal{N}(\boldsymbol{\mu}_t; \mathbf{e}_t, \mathbf{D})}{\mathcal{N}(\boldsymbol{\gamma}_t; \boldsymbol{\nu}_{r_t}, \mathbf{S}_r)\mathcal{N}(\boldsymbol{\mu}_t; \mathbf{e}_{r_t}, \mathbf{D}_r)} \\ &\propto \mathcal{N}(\boldsymbol{\gamma}_t; \boldsymbol{\nu}_t^{\setminus r}, \mathbf{S}^{\setminus r})\mathcal{N}(\boldsymbol{\mu}_t; \mathbf{e}_t^{\setminus r}, \mathbf{D}^{\setminus r}), \end{aligned}$$

where

$$\begin{aligned} (\mathbf{S}^{\setminus r})^{-1} &= (\mathbf{S})^{-1} - (\mathbf{S}_r)^{-1} \\ (\mathbf{S}^{\setminus r})^{-1}\boldsymbol{\nu}_t^{\setminus r} &= (\mathbf{S})^{-1}\boldsymbol{\nu}_t - (\mathbf{S}_r)^{-1}\boldsymbol{\nu}_{r_t} \\ (\mathbf{D}^{\setminus r})^{-1} &= (\mathbf{D})^{-1} - (\mathbf{D}_r)^{-1} \\ (\mathbf{D}^{\setminus r})^{-1}\mathbf{e}_t^{\setminus r} &= (\mathbf{D})^{-1}\mathbf{e}_t - (\mathbf{D}_r)^{-1}\mathbf{e}_{r_t} \end{aligned}$$

B. Find the update for the factor $q_{r_t}^{new}$

For the factor q_{r_t} parameters of the Gaussian distributions found during the moment matching step are cancelled out during the factor update step and the resulting formulae are

$$q_{r_t}^{\text{new}}(\boldsymbol{\gamma}_t, \boldsymbol{\mu}_t) \propto \mathcal{N}\left(\boldsymbol{\gamma}_t; \boldsymbol{\nu}_{r_t}^{\text{new}}, \mathbf{S}_r^{\text{new}}\right)\mathcal{N}\left(\boldsymbol{\mu}_t; \mathbf{e}_{r_t}^{\text{new}}, \mathbf{D}_r^{\text{new}}\right),$$

where

$$\begin{aligned} \mathbf{S}_r^{\text{new}} &= \mathbf{D}^{\setminus r} + \boldsymbol{\Sigma}_0, \qquad \boldsymbol{\nu}_{r_t}^{\text{new}} = \mathbf{e}_t^{\setminus r} \\ \mathbf{D}_r^{\text{new}} &= \mathbf{S}^{\setminus r} + \boldsymbol{\Sigma}_0, \qquad \mathbf{e}_{r_t}^{\text{new}} = \boldsymbol{\nu}_t^{\setminus r}. \end{aligned}$$

APPENDIX E
EP UPDATE FOR FACTOR u_t

A. Cavity distribution

The unnormalised cavity distribution $q^{\setminus q_{u_t}}(\boldsymbol{\mu}_{t-1}, \boldsymbol{\mu}_t) = \frac{q(\boldsymbol{\mu}_{t-1}, \boldsymbol{\mu}_t)}{q_{u_t}(\boldsymbol{\mu}_{t-1}, \boldsymbol{\mu}_t)}$ can be computed as

$$\begin{aligned} q^{\setminus q_{u_t}} &= \frac{\mathcal{N}(\boldsymbol{\mu}_{t-1}; \mathbf{e}_{t-1}, \mathbf{D})\mathcal{N}(\boldsymbol{\mu}_t; \mathbf{e}_t, \mathbf{D})}{\mathcal{N}(\boldsymbol{\mu}_{t-1}; \mathbf{e}_{u_t\leftarrow}, \mathbf{D}_{u\leftarrow})\mathcal{N}(\boldsymbol{\mu}_t; \mathbf{e}_{u_t\rightarrow}, \mathbf{D}_{u\rightarrow})} \\ &\propto \mathcal{N}(\boldsymbol{\mu}_{t-1}; \mathbf{e}_{t-1}^{\setminus u}, \mathbf{D}_{t-1}^{\setminus u})\mathcal{N}(\boldsymbol{\mu}_t; \mathbf{e}_t^{\setminus u}, \mathbf{D}_t^{\setminus u}), \end{aligned}$$

where

$$\begin{aligned} (\mathbf{D}_{t-1}^{\setminus u})^{-1} &= (\mathbf{D})^{-1} - (\mathbf{D}_{u\leftarrow})^{-1} \\ (\mathbf{D}_{t-1}^{\setminus u})^{-1}\mathbf{e}_{t-1}^{\setminus u} &= (\mathbf{D})^{-1}\mathbf{e}_{t-1} - (\mathbf{D}_{u\leftarrow})^{-1}\mathbf{e}_{u_t\leftarrow} \\ (\mathbf{D}_t^{\setminus u})^{-1} &= (\mathbf{D})^{-1} - (\mathbf{D}_{u\rightarrow})^{-1} \\ (\mathbf{D}_t^{\setminus u})^{-1}\mathbf{e}_t^{\setminus u} &= (\mathbf{D})^{-1}\mathbf{e}_t - (\mathbf{D}_{u\rightarrow})^{-1}\mathbf{e}_{u_t\rightarrow} \end{aligned}$$

B. Find the update for the factor $q_{u_t}^{new}$

For the factor q_{u_t} parameters of the Gaussian distributions found during the moment matching step are cancelled out during the factor update step and the resulting formulae are

$$q_{u_t}^{\text{new}}(\boldsymbol{\mu}_{t-1}, \boldsymbol{\mu}_t) \propto \mathcal{N}\left(\boldsymbol{\mu}_t; \mathbf{e}_{u_t\rightarrow}^{\text{new}}, \mathbf{D}_{u\rightarrow}^{\text{new}}\right)\mathcal{N}\left(\boldsymbol{\mu}_{t-1}; \mathbf{e}_{u_t\leftarrow}^{\text{new}}, \mathbf{D}_{u\leftarrow}^{\text{new}}\right),$$

where

$$\begin{aligned} \mathbf{D}_{u\rightarrow}^{\text{new}} &= \mathbf{D}_{t-1}^{\setminus u} + \mathbf{W}, \qquad \mathbf{e}_{u_t\rightarrow}^{\text{new}} = \mathbf{e}_{t-1}^{\setminus u} \\ \mathbf{D}_{u\leftarrow}^{\text{new}} &= \mathbf{D}_t^{\setminus u} + \mathbf{W}, \qquad \mathbf{e}_{u_t\leftarrow}^{\text{new}} = \mathbf{e}_t^{\setminus u}. \end{aligned}$$