A State-Space Approach to Dynamic Nonnegative Matrix Factorization

Nasser Mohammadiha, *Member, IEEE*, Paris Smaragdis, *Fellow, IEEE*, Ghazaleh Panahandeh, *Student Member, IEEE*, and Simon Doclo, *Senior Member, IEEE*

***Abstract*—Nonnegative matrix factorization (NMF) has been actively investigated and used in a wide range of problems in the past decade. A significant amount of attention has been given to develop NMF algorithms that are suitable to model time series with strong temporal dependencies. In this paper, we propose a novel state-space approach to perform dynamic NMF (D-NMF). In the proposed probabilistic framework, the NMF coefficients act as the state variables and their dynamics are modeled using a multi-lag nonnegative vector autoregressive (N-VAR) model within the process equation. We use expectation maximization and propose a maximum-likelihood estimation framework to estimate the basis matrix and the N-VAR model parameters. Interestingly, the N-VAR model parameters are obtained by simply applying NMF. Moreover, we derive a maximum a posteriori estimate of the state variables (i.e., the NMF coefficients) that is based on a prediction step and an update step, similarly to the Kalman filter. We illustrate the benefits of the proposed approach using different numerical simulations where D-NMF significantly outperforms its static counterpart. Experimental results for three different applications show that the proposed approach outperforms two state-of-the-art NMF approaches that exploit temporal dependencies, namely a nonnegative hidden Markov model and a frame stacking approach, while it requires less memory and computational power.**

***Index Terms*—Constrained Kalman filtering, nonnegative dynamical system (NDS), nonnegative matrix factorization (NMF), prediction, probabilistic latent component analysis (PLCA).**

I. Introduction

NONNEGATIVE matrix factorization (NMF) [1] is an approach to obtain a low-rank representation of nonnegative data. In NMF, a nonnegative data matrix $\mathbf{X}$ is factorized into a product of two nonnegative matrices $\mathbf{W}$ and $\mathbf{H}$ such that $\mathbf{WH}$ provides a good approximation of $\mathbf{X}$ with respect to (w.r.t.) a predefined criterion. In our notation, each column of $\mathbf{X}$ corresponds to a multivariate observation in time. $\mathbf{W}$ and $\mathbf{H}$ are referred to as the basis matrix and NMF coefficient matrix, respectively, where each row of $\mathbf{H}$ represents the activity of its corresponding basis vector over time.

In many signal processing applications, e.g., audio processing and analysis of time series, the consecutive columns of $\mathbf{X}$ exhibit a strong temporal correlation. In the basic NMF approach, however, each observation is treated individually. A simple and useful approach to alleviate this problem is to stack the consecutive columns of the data matrix into high-dimensional super-vectors, and to apply NMF to learn high-dimensional basis vectors. This frame stacking approach is one of the key ingredients in so-called exemplar-based representations [2]. More rigorously, to model the temporal dependencies in NMF, three main approaches have been developed in the past: 1) Convolutive NMF [3], [4], in which the dependencies are usually imposed on the basis matrix $\mathbf{W}$. 2) Smooth NMF [5], [6, and references therein] where each row of $\mathbf{H}$ is individually constrained to evolve smoothly over time. 3) Approaches that combine NMF and hidden Markov model (HMM) paradigms [7]–[10]. In the so-called nonnegative hidden Markov model (N-HMM) [7], [9], the NMF coefficient vectors are assumed to be sparse and a first-order Markovian chain is considered over the index of the active coefficients. These approaches are explained and compared in [11] in a unified framework.

Kalman filtering and its nonlinear extensions [12], [13] have been extensively studied within the signal processing community to exploit temporal correlations in an optimal way. The basic Kalman filter is based on a linear state-space model in which both the process noise and observation noise are assumed to be Gaussian-distributed. The goal of the Kalman filter is then to find a minimum-mean-square-error (MMSE) estimator of the state variables given the current and past observations. This estimator is obtained by minimizing the Bayesian mean square error (MSE) where no additional constraints are imposed on the state variables. If the noise distributions are not Gaussian, the Kalman filter still provides the optimal linear MMSE estimator [12].

Recently, there has been some research on developing Kalman filters subject to state constraints. In addition to the model reparameterization, the projection and pseudo-observation approaches are two usual solutions to handle constraints [14]–[16]. In the projection approach, the unconstrained estimate (after the observation update) is projected to satisfy the constraints. In the pseudo-observation approach, however, a fictitious observation is considered using which the

Manuscript received January 30, 2014; revised August 21, 2014; accepted December 08, 2014. Date of publication December 23, 2014; date of current version January 22, 2015. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Raviv Raich. This work was supported by the Cluster of Excellence 1077 "Hearing4all," funded by the German Research Foundation (DFG).

N. Mohammadiha and S. Doclo are with the Department of Medical Physics and Acoustics, University of Oldenburg, Oldenburg 26111, Germany (e-mail: nasser.mohammadiha@uni-oldenburg.de; simon.doclo@uni-oldenburg.de).

P. Smaragdis is with the Department of Computer Science and the Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA (e-mail: paris@illinois.edu).

G. Panahandeh is with the Signal Processing Group, School of Electrical Engineering, KTH Royal Institute of Technology, Stockholm, Sweden (e-mail: ghpa@kth.se).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2014.2385655

unconstrained estimate is further updated similarly to a real observation update. For example, in [16] Kalman filtering with sparseness constraints on the state variables is addressed where a pseudo-observation approach is developed. In this paper, we focus on dynamic filtering in a state-space representation where both the state variables and the observations are considered to be nonnegative. In this case, the mentioned approaches can not be used to optimally apply the constraints. This is not only because the state variables are nonnegative, but also the distribution of the nonnegative observations may be far from a Gaussian distribution.

Very recently, nonnegative dynamical systems [17], [18] have been proposed in which the NMF coefficients act as the state variables. Compared to N-HMM, the continuous state-spaces utilized in these approaches provide a richer mechanism to benefit from the temporal dependencies. To use the temporal dynamics in the estimation of the NMF coefficients, we proposed a two-step algorithm in [17] based on a prediction and an update step. However, none of the estimators for the NMF coefficients and the dynamic model parameters were optimally derived for the specified assumptions. In the current work, we further refine the theoretical foundations of our previous study in [17] and derive new estimators and present new examples and results.

In this paper, we formulate the dynamic NMF using a novel state-space representation and use the expectation-maximization (EM) algorithm to derive optimal update rules for the NMF coefficients and the model parameters. We consider a probabilistic formulation of NMF [19] and develop a state-space representation to model the temporal dependencies in NMF. The process equation, which describes the evolution of the NMF coefficients, is based on an exponential distribution whose parameter is given by a multi-lag nonnegative vector autoregressive (N-VAR) model. The observation equation is similar to static NMF where the observations are assumed to be drawn from a multinomial distribution. The choice of these distributions is based on both their appropriateness to model nonnegative data and the possibility to derive closed-form solutions. We propose a maximum a posteriori (MAP) approach to estimate the state variables $\mathbf{H}$. The obtained MAP estimate that consists of a prediction step and an update step is a filtering solution since it is only conditioned on the current and past observations. Additionally, we derive maximum likelihood (ML) estimates of the basis vectors $\mathbf{W}$ and the N-VAR model parameters. We show that the ML estimate of the N-VAR model parameters is obtained by simply applying NMF, which is well suited to our nonnegative framework. We provide numerical simulations for three examples, i.e., tracking the frequency of a single sinusoid in noise, separation of two sources with similar basis matrices, and speaker-dependent and -independent speech denoising examples. We compare the performance of the proposed D-NMF approach to the performance of the static NMF approach, the N-HMM in [7], and the frame stacking approach in [2]. Our simulations show that the D-NMF approach outperforms these competing algorithms, while it is less complex and hence it is a better choice for real-time applications.

The remainder of the paper is organized as follows: Section II provides a short overview of NMF. The proposed dynamic NMF using a state-space model is presented in Section III. Numerical simulations for several problems are presented in Section IV.

II. Nonnegative Matrix Factorization

Nonnegative Matrix Factorization is a method using which a $K \times T$-dimensional nonnegative matrix $\mathbf{X} = \{x_{kt}\}$ is approximated as $\mathbf{WH}$, where the $K \times I$-dimensional matrix $\mathbf{W} = \{w_{ki}\}$ and the $I \times T$-dimensional matrix $\mathbf{H} = \{h_{it}\}$ are both constrained to be nonnegative. The model order I, i.e., the number of columns in $\mathbf{W}$, is usually less than K, i.e., the number of rows in $\mathbf{X}$, such that a dimension reduction is also achieved using NMF. The t-th columns of $\mathbf{X}$ and $\mathbf{H}$ are denoted by $\mathbf{x}_t$ and $\mathbf{h}_t$, respectively. The nonnegativeness property is usually helpful to interpret the factorization using the underlying physical phenomena. In the deterministic NMF approaches [1], a cost function measuring the approximation error is minimized under the given nonnegativity constraints. Popular choices for the cost function include Euclidean distance (in EUC-NMF), Kullback-Leibler divergence (in KL-NMF), and the Itakura-Saito divergence (in IS-NMF) [1], [5].

In the following, we briefly describe the IS-NMF since we will use it in our algorithm. Letting $\hat{\mathbf{X}} = \mathbf{WH}$, the IS divergence for the NMF problem is defined as:

$$d_{\mathrm{IS}}\left(\mathbf{X}\|\hat{\mathbf{X}}\right) = \sum_{k=1}^{K}\sum_{t=1}^{T}\left(\frac{x_{kt}}{\hat{x}_{kt}} - \log\frac{x_{kt}}{\hat{x}_{kt}} - 1\right). \tag{1}$$

A widely used approach to minimize NMF cost functions is using the multiplicative update rules, which minimize the cost function in an iterative manner. For the IS-NMF, these update rules are given as (see, e.g., [5]):

$$\mathbf{H} \leftarrow \mathbf{H} \odot \frac{\mathbf{W}^{\top}\left((\mathbf{WH})^{-2} \odot \mathbf{X}\right)}{\mathbf{W}^{\top}(\mathbf{WH})^{-1}}, \tag{2}$$

$$\mathbf{W} \leftarrow \mathbf{W} \odot \frac{\left((\mathbf{WH})^{-2} \odot \mathbf{X}\right)\mathbf{H}^{\top}}{(\mathbf{WH})^{-1}\mathbf{H}^{\top}}, \tag{3}$$

where $\top$ denotes matrix transpose, $\odot$ represents element-wise multiplication, and the division and powers are performed element-wise. These updates are iteratively performed until a local minimum of the cost function is found.

In contrast to the deterministic NMF approaches, probabilistic formulations facilitate deriving the desired estimates in a statically optimal manner. In the next section, we present our D-NMF algorithm that is based on probabilistic latent component analysis (PLCA)[19].

III. Proposed Dynamic NMF

A. Statistical Model Description

We propose a state-space approach to perform dynamic nonnegative factorization. In this approach, the NMF coefficients

$\mathbf{h}_t$ are assumed to evolve over time according to the following nonnegative vector autoregressive (N-VAR) model:

$$\begin{cases} f\left(\mathbf{h}_t \mid \mathbf{A}, \mathbf{h}_{t-1}, \dots \mathbf{h}_{t-J}\right) = \text{v} - \exp\left(\mathbf{h}_t; \sum_{j=1}^{J} \mathbf{A}_j \mathbf{h}_{t-j}\right), \\ f\left(\mathbf{x}_t \mid \mathbf{W}, \mathbf{h}_t\right) = \text{mult}\left(\mathbf{x}_t; \gamma_t, \mathbf{W}\mathbf{h}_t\right), \end{cases} \tag{4}$$

where $f(\cdot)$ denotes a probability density function, $\mathbf{x}_t$ denotes the nonnegative observation vector at time t with $\gamma_t = \sum_k x_{kt}$, J is the order of the N-VAR model, $\mathbf{A}_j$ denotes the $I \times I$-dimensional N-VAR model parameters corresponding to the j-th lag (and $\mathbf{A}$ denotes the union of $\mathbf{A}_j$, $\forall j$), v-exp $(\mathbf{h}_t; \boldsymbol{\eta}_t)$ is the exponential probability density function over the vector $\mathbf{h}_t$ with independent elements h_{it} as:

$$\text{v-exp}\left(\mathbf{h}_t; \boldsymbol{\eta}_t\right) = \prod_{i=1}^{I} \eta_{it}^{-1} e^{-h_{it}/\eta_{it}}, \tag{5}$$

and mult $(\mathbf{x}_t; \gamma_t, \mathbf{p}_t)$ represents a multinomial distribution as:

$$\text{mult}\left(\mathbf{x}_t; \gamma_t, \mathbf{p}_t\right) = \gamma_t! \prod_{k=1}^{K} \frac{p_{kt}^{x_{kt}}}{x_{kt}!}, \tag{6}$$

where ! denotes the factorial and $\sum_k p_{kt} = 1$. The conditional expected values of $\mathbf{h}_t$ and $\mathbf{x}_t$ under the model (4) are given by:

$$\begin{cases} E\left(\mathbf{h}_t \mid \mathbf{A}, \mathbf{h}_{t-1}, \dots \mathbf{h}_{t-J}\right) = \sum_{j=1}^{J} \mathbf{A}_j \mathbf{h}_{t-j}, \\ E\left(\mathbf{x}_t \mid \mathbf{W}, \mathbf{h}_t\right) = \left(\sum_k x_{kt}\right) \mathbf{W}\mathbf{h}_t, \end{cases} \tag{7}$$

which is used to obtain an NMF approximation of the input data as $\mathbf{x}_t \approx (\sum_k x_{kt})\mathbf{W}\mathbf{h}_t$.

The distributions in (4) are chosen to be appropriate for nonnegative data. For example, it is well known that the conjugate prior for the multinomial likelihood is the Dirichlet distribution. However, it can be shown that the obtained state estimates in this case are no longer guaranteed to be nonnegative. Therefore, we propose to use the exponential distribution in (4) for which, as will be shown in Section III-C, the obtained state estimates are always nonnegative. In addition, a closed-form solution can be derived under the given statistical assumptions, see Section III-C.

If we discard the first equation in (4), we recover the basic PLCA algorithm [19]. This special case (corresponding to $J = 0$) is referred to as the static NMF as it does not model temporal dependencies. Here, the observations $\mathbf{x}_t$ are assumed to be count data over K possible categories. Using the PLCA notation, each vector $\mathbf{h}_t$ is a probability vector that represents the contribution of each basis vector in explaining the observation, i.e., $h_{it} = f\left(z_t = i\right)$ where z_t is a latent variable used to index the basis vectors at time t. Moreover, each column of $\mathbf{W}$ is a probability vector that contains the underlying structure of the observations given the latent variable z and is referred to as a basis vector. More precisely, w_{ki} is the probability that the k-th element of $\mathbf{x}_t$ will be chosen in a single draw from the multinomial distribution in (4), i.e., $w_{ki} = f\left(\mathbf{x}_t = \mathbf{e}_k \mid z_t = i\right)$ with $\mathbf{e}_k$ being a K-dimensional indicator vector whose k-th element is equal to one (see [17] for more explanation). Note that (by definition) w_{ki} is time-invariant. In the following, this notation is abbreviated to $w_{ki} = f\left(k \mid z_t = i\right)$.

It is worthwhile to compare (4) to the state-space model utilized in the Kalman filter and to highlight the main differences between the two. First, all the variables are constrained to be nonnegative in (4). Second, the process and observation noises are embedded into the specified distributions, which is different from the additive Gaussian noise utilized in the Kalman filtering. Finally, in the process equation, we have used a multi-lag N-VAR model. In our proposed algorithm, different lags can have different importance weights, which will be discussed in Section III-C. It is also important to note that we aim to estimate both state-space model parameters ($\mathbf{W}$ and $\mathbf{A}$) and state variables $\mathbf{H}$, where Kalman filter only estimates $\mathbf{H}$, given a priori determined $\mathbf{W}$ and $\mathbf{A}$.

In Section III-C, we derive an expectation-maximization (EM) algorithm to compute maximum likelihood (ML) estimates of $\mathbf{A}$ and $\mathbf{W}$ and to compute a MAP estimate of the state variables $\mathbf{H}$. In the latter case, the estimation consists of prediction and update steps, similarly to the classical Kalman filter. However, we no longer update the prediction with an additive term but we have a nonlinear update function.

B. Relation to Other Works

The proposed state-space representation in (4) provides a framework to exploit the correlation between the consecutive columns of the nonnegative data matrix in NMF. Several approaches have been proposed in the literature to exploit the temporal dynamics in NMF, such as frame stacking [2], convolutive NMF [3], [4], smooth NMF [5], [6], [20], [21], and state-space representations [7]–[10], [17], [18]. The state-space representations (including our proposed approach) model the interdependencies between different rows of the NMF coefficient matrix $\mathbf{H}$, unlike the smooth NMF approaches that assume these rows to be independent. Most of these approaches can be explained in a unified framework [11]. Our proposed approach is most related to the N-HMM approach in [7] and the nonnegative dynamical system (NDS) in [18].

Both our proposed D-NMF approach and the N-HMM approach in [7] use the PLCA framework and provide a state-space representation to benefit from the temporal dynamics. However, unlike the N-HMM approach that uses a discrete state-space representation, our approach is based on a continuous state-space representation. The principal difference between both approaches is hence the same as the difference between HMM and Kalman filter. A continuous dynamical system is superior if the underlying source signal smoothly transits between many (possibly infinite) states, whereas a discrete dynamical system can be more suitable if the source signal switches between a limited number of states. Hence, N-HMM can for example be a good model for speech if we assume that a speech signal exhibits a switching behavior between limited number of phonemes. On the other hand, a continuous state-space representation is more appropriate for multitalker babble noise, since it is generated as the sum of a number of speech signals, and there are in principle many states obtained by the combination of the states of the underlying speech signals. A thorough discussion on this

example can be found in [9]. Another important difference between our proposed D-NMF method and the N-HMM methods in [7], [9] is computational complexity. To analyze a mixture of two (or more) sources where each source is individually modeled using an N-HMM, a factorial N-HMM has to be used. This leads to exponential complexity in the number of sources for N-HMM based systems, and approximate inference approaches, e.g., [22], have to be used to keep the complexity tractable. In contrast, the complexity of the D-NMF approach is linear in the number of sources and no approximation is needed.

Similar to our D-NMF approach, the NDS approach in [18] uses a continuous state-space representation that is written as:

$$\begin{cases} f\left(\mathbf{h}_t \mid \mathbf{A}_1, \mathbf{h}_{t-1}, \boldsymbol{\alpha}\right) = \text{v-gamma}\left(\mathbf{h}_t; \boldsymbol{\alpha}, \mathbf{A}_1\mathbf{h}_{t-1}/\boldsymbol{\alpha}\right), \\ f\left(\mathbf{x}_t \mid \mathbf{W}, \mathbf{h}_t, \delta\right) = \text{v-gamma}\left(\mathbf{x}_t; \delta\mathbf{1}, \mathbf{W}\mathbf{h}_t/\delta\right), \end{cases} \tag{8}$$

where I-dimensional vector $\boldsymbol{\alpha}$ and scalar δ are model parameters, $\mathbf{1}$ is a K-dimensional all-ones-vector, division of vectors is performed element by element, and v-gamma $(\mathbf{h}_t \mid \boldsymbol{\alpha}, \boldsymbol{\beta})$ corresponds to a gamma distribution over the vector $\mathbf{h}_t$ with independent elements h_{it} as

$$\text{v-gamma}\left(\mathbf{h}_t \mid \boldsymbol{\alpha}, \boldsymbol{\beta}\right) = \prod_{i=1}^{I} \frac{1}{\beta_i^{\alpha_i}\Gamma\left(\alpha_i\right)} h_{it}^{\alpha_i - 1} e^{-h_{it}/\beta_i}, \tag{9}$$

where $\Gamma\left(\cdot\right)$ is the gamma function. Using (9), the conditional expected values of the state variables and data are given as $E(\mathbf{h}_t \mid \mathbf{A}_1, \mathbf{h}_{t-1}, \boldsymbol{\alpha}) = \mathbf{A}_1\mathbf{h}_{t-1}$, and $E(\mathbf{x}_t \mid \mathbf{W}, \mathbf{h}_t, \delta) = \mathbf{W}\mathbf{h}_t$, respectively.

There are three main differences between our D-NMF and the NDS approaches. Firstly, the NDS approach assumes that each element of the observation vector (x_{kt}) is a gamma-distributed random variable, while in our approach the observation vectors $(\mathbf{x}_t)$ are multinomial-distributed. These assumptions lead to two different NMF cost functions, where one of them may be preferred for a specific application [11]. The NDS method minimizes the IS divergence and hence is a dynamical IS-NMF, while our method minimizes a weighted KL divergence and hence is the dynamical counterpart for PLCA. Additionally, the assumed distribution for the NMF coefficients corresponds to an exponential and a gamma distribution for the NDS approach and our D-NMF approach, respectively. Secondly, our proposed D-NMF approach provides a more general multi-lag predictor for the state variables, while the NDS approach (as well as N-HMM approaches) use a one-lag predictor, i.e., $J = 1$. Thirdly, our proposed estimation approach (Section III-C) has appealing properties regarding the estimation of the state variables and the N-VAR model parameters. Our estimation of the state variables consists of two steps corresponding to a prediction and an update step, similar to a Kalman filter, which leads to an easy and intuitive explanation of the update rules. Moreover, we show that the N-VAR model parameters can be estimated by applying a separate NMF, which is more suitable in the nonnegative framework. Neither of these properties are provided in the NDS approach.

C. Estimation Algorithm

In this section, we derive an EM algorithm to estimate the nonnegative parameters in (4), which are denoted by $\boldsymbol{\lambda} = \{\mathbf{A}, \mathbf{H}, \mathbf{W}\}$, given a nonnegative data matrix $\mathbf{X}$. We aim to maximize the MAP objective function for the model given in (4), (5), and (6), i.e., as[1]:

$$\begin{aligned} Q^{\text{MAP}} &= \log f\left(\mathbf{X}, \mathbf{H} \mid \mathbf{W}, \mathbf{A}\right) \\ &= \log f\left(\mathbf{X} \mid \mathbf{W}, \mathbf{H}\right) + \log f\left(\mathbf{H} \mid \mathbf{A}\right). \end{aligned} \tag{10}$$

Maximizing Q^{MAP} w.r.t. $\mathbf{W}$, $\mathbf{A}$ and $\mathbf{H}$ results in a MAP estimate of $\mathbf{H}$ and ML estimates of $\mathbf{W}$ and $\mathbf{A}$. For this optimization, we derive an EM algorithm [23], which is a commonly used approach to estimate the unknown parameters in the presence of latent variables. The EM algorithm maximizes a lower bound on Q^{MAP} and iterates between an expectation (E) step and a maximization (M) step until convergence. We denote the EM latent variables by z_t, an indicator variable to index the basis vectors. In the E step, the posterior probabilities of these variables are obtained as:

$$\begin{aligned} f\left(z_t = i \mid k, \boldsymbol{\lambda}\right) &= \frac{f\left(k \mid z_t = i\right) f\left(z_t = i\right)}{\sum_{i=1}^{I} f\left(k \mid z_t = i\right) f\left(z_t = i\right)} \\ &= \frac{w_{ki} h_{it}}{\sum_i w_{ki} h_{it}}, \end{aligned} \tag{11}$$

where $\boldsymbol{\lambda}$ denotes the estimated parameters from the previous iteration of the EM algorithm. In the M step, the expected log-likelihood of the complete data [23, Chapter 9][2]:

$$\begin{aligned} Q\left(\hat{\boldsymbol{\lambda}}, \boldsymbol{\lambda}\right) = \sum_{t,i} f\left(z_t = i \mid k, \boldsymbol{\lambda}\right) \log f\left(\mathbf{x}_t, z_t \mid \hat{\boldsymbol{\lambda}}\right) \\ + \sum_t \log f\left(\hat{\mathbf{h}}_t \mid \hat{\mathbf{A}}, \hat{\mathbf{h}}_{t-1}, \ldots \hat{\mathbf{h}}_{t-J}\right), \end{aligned} \tag{12}$$

is maximized w.r.t. $\hat{\boldsymbol{\lambda}}$ to obtain a new set of estimates. Note that using Jensen's inequality, it can be easily proved that $Q\left(\hat{\boldsymbol{\lambda}}, \boldsymbol{\lambda}\right)$ is a lower bound for Q^{MAP}. Using (5) and (6), $Q\left(\hat{\boldsymbol{\lambda}}, \boldsymbol{\lambda}\right)$ can be equivalently (up to a constant) written as (also see [24]):

$$\begin{aligned} Q\left(\hat{\boldsymbol{\lambda}}, \boldsymbol{\lambda}\right) = \sum_{k,t,i} x_{kt} f\left(z_t = i \mid k, \boldsymbol{\lambda}\right) \left(\log \hat{w}_{ki} + \log \hat{h}_{it}\right) \\ - \sum_{i,t} \left(\log \hat{\eta}_{it} + \frac{\hat{h}_{it}}{\hat{\eta}_{it}}\right), \end{aligned} \tag{13}$$

where $\hat{\boldsymbol{\eta}}_t = \sum_{j=1}^{J} \hat{\mathbf{A}}_j \hat{\mathbf{h}}_{t-j}$. As mentioned in Section III-A, $\mathbf{w}_i$ and $\mathbf{h}_t$ are probability vectors, and hence, to make sure that they sum to one, we need to impose two constraints $\sum_i \hat{h}_{it} = 1$ and $\sum_k \hat{w}_{ki} = 1$. To solve the constrained optimization problem, we form the Lagrangian function $\mathcal{L}$ and maximize it:

$$\mathcal{L} = Q\left(\hat{\boldsymbol{\lambda}}, \boldsymbol{\lambda}\right) + \sum_i \alpha_i \left(1 - \sum_k \hat{w}_{ki}\right) + \sum_t \beta_t \left(1 - \sum_i \hat{h}_{it}\right), \tag{14}$$

[1]Note that $f\left(\mathbf{h}_t\right)$ (as part of $f\left(\mathbf{H}\right)$) in (10) is not only conditioned on $\mathbf{A}$ but also on $\mathbf{h}_{t-1}, \ldots \mathbf{h}_{t-J}$. The latter conditioning is omitted in this equation to keep the notations uncluttered.

[2]For $t \leq J$, $\hat{\mathbf{h}}_{t-j}$ is set to a vector consisting of ones to prevent accessing undefined variables.

where α_i, $i = 1 \dots I$ and β_t, $t = 1 \dots T$ are Lagrange multipliers. In the following, we describe the maximization w.r.t. $\mathbf{W}$, $\mathbf{H}$, and $\mathbf{A}$, respectively.

Equation (14) can be easily maximized w.r.t. $\hat{w}_{ki}$ to obtain:

$$\hat{w}_{ki} = \frac{\sum_t x_{kt} f\left(z_t = i \mid k, \boldsymbol{\lambda}\right)}{\alpha_i}, \tag{15}$$

where the Lagrange multiplier $\alpha_i = \sum_{t,k} x_{kt} f\left(z_t = i \mid k, \boldsymbol{\lambda}\right)$ to ensure that $\hat{\mathbf{w}}_i$ sums to one. For the estimation of $\mathbf{H}$, we propose a recursive algorithm, i.e., we estimate $\hat{\mathbf{h}}_1, \hat{\mathbf{h}}_2, \dots$ sequentially. Therefore, we first predict the state variables as

$$\hat{\mathbf{h}}_{t|t-1} = \hat{\boldsymbol{\eta}}_t = \sum_{j=1}^{J} \hat{\mathbf{A}}_j \hat{\mathbf{h}}_{t-j}, \tag{16}$$

where $\hat{\mathbf{h}}_{t|t-1}$ is the prediction result given all the past observations $\mathbf{x}_1, \dots \mathbf{x}_{t-1}$. In the update step, the current observation $\mathbf{x}_t$ is used to update the state estimate. This is done by maximizing (14) w.r.t. $\hat{\mathbf{h}}_t$. Setting the derivative of $\mathcal{L}$ w.r.t. $\hat{h}_{it}$ to zero, we obtain:

$$\hat{h}_{it|t} = \hat{h}_{it} = \frac{\sum_k x_{kt} f\left(z_t = i \mid k, \boldsymbol{\lambda}\right)}{\beta_t + 1/\hat{\eta}_{it}}. \tag{17}$$

The Lagrange multiplier β_t has to be computed such that $\hat{\mathbf{h}}_t$ sums to one, for which we have used an iterative Newton's method.

Finally, the estimation of the N-VAR parameters $\hat{\mathbf{A}}$ is presented in the following. Note that there are many approaches to estimate the VAR model parameters in the literature [25], [26]. However, since most of these approaches are based on least-squares estimation, they are not suitable for our nonnegative framework. Moreover, they tend to be very time-consuming for high-dimensional data. First, let us define the $I \times IJ$-dimensional matrix $\hat{\mathbf{A}}$ as: $\hat{\mathbf{A}} = \left[\hat{\mathbf{A}}_1 \ \hat{\mathbf{A}}_2 \dots \hat{\mathbf{A}}_J\right]$. Accordingly, let IJ-dimensional vector $\hat{\mathbf{v}}_t$ represent the stacked state variables as: $\hat{\mathbf{v}}_t^\top = \left[\hat{\mathbf{h}}_{t-1}^\top \ \hat{\mathbf{h}}_{t-2}^\top \dots \hat{\mathbf{h}}_{t-J}^\top\right]$. The parts of (14) that depend on $\hat{\mathbf{A}}$ are equivalently written as:

$$\begin{aligned}\mathcal{L}^{(A)} &= -\sum_{i,t} \left(\log \left[\hat{\mathbf{A}}\hat{\mathbf{v}}_t\right]_i + \frac{\hat{h}_{it}}{\left[\hat{\mathbf{A}}\hat{\mathbf{v}}_t\right]_i} \right) \\ &= -d_{\mathrm{IS}}\left(\hat{\mathbf{H}} \| \hat{\mathbf{A}}\hat{\mathbf{V}}\right) - \sum_{i,t} \left(\log \hat{h}_{it} + 1\right), \end{aligned} \tag{18}$$

where $\hat{\mathbf{V}} = [\hat{\mathbf{v}}_1 \dots \hat{\mathbf{v}}_T]$, $[\cdot]_i$ denotes the i-th entry of its argument, and $d_{\mathrm{IS}}(\cdot\|\cdot)$ is the IS divergence as defined in (1). The second term in (18) is constant and can be ignored for the purpose of optimization w.r.t. $\hat{\mathbf{A}}$. Hence, the ML estimate of $\hat{\mathbf{A}}$ can be obtained by performing IS-NMF in which the NMF coefficient matrix $\hat{\mathbf{V}}$ is held fixed and only the basis matrix $\hat{\mathbf{A}}$ is optimized. This is done by executing (3) iteratively until convergence. Alternatively, we can repeat (3) only once resulting in a generalized EM algorithm. We used the latter alternative in our simulations.

The proposed estimation approach for the N-VAR parameters is able to automatically capture the importance weight for each lag, i.e., $\mathbf{A}_j$, $j = 1 \dots J$ are not required to, e.g., be normalized to have the same l_1 norm. Hence, different lags may contribute differently, proportional to their norm, in computing $\sum \mathbf{A}_j \mathbf{h}_{t-j}$. This is achieved because the NMF coefficients $\hat{\mathbf{V}}$ are held fixed in the IS-NMF, and we no longer have a scale ambiguity in the NMF representation.

Algorithm 1 Proposed dynamic NMF: algorithm to learn the model parameters.

1) Set the predefined variables I (number of NMF basis vectors), J (N-VAR model order), and M, q (see the text).
2) Initialize $\hat{\mathbf{W}}, \hat{\mathbf{H}}$ and $\hat{\mathbf{A}}_j$ for $j = 1 \dots J$ with positive random numbers. Set $r = 1$.
3) Repeat until convergence:
   a) Compute $\hat{\mathbf{W}}$ using **(15)**
   b) Compute the state variables $\hat{\mathbf{H}}$
      for $t = 1 : T$ **do**
         % Predict
         if $r > M$ **then**
            Compute $\hat{\boldsymbol{\eta}}_t$ using **(16)**.
            Anneal the prediction as $\hat{\eta}_{it} = \hat{\eta}_{it}^q$.
         else
            Set $\hat{\boldsymbol{\eta}}_t$ to all-ones-vector.
         end if
         % Update
         Update the state estimate $\hat{\mathbf{h}}_t$ using **(17)**.
   c) Compute the N-VAR parameters
      if $r \geq M$ **then**
         Compute $\hat{\mathbf{A}}_j$ for $j = 1 \dots J$ using **(18)** and **(3)**.
      end if
   d) $r = r + 1$

Algorithm 1 summarizes our proposed D-NMF approach to estimate all the model parameters simultaneously, which is usually applied on the training data (cf. Section IV) to learn the model parameters $\mathbf{W}$ and $\mathbf{A}$. As convergence criterion, the stationarity of Q^{MAP} or EM lower bound can be checked, or a fixed (sufficient) number of iterations can be simply used. In our simulations, we have used 100 iterations. This algorithm includes two practical additions. First, since the EM algorithm converges to a local optimum of the objective function, a good initialization can improve the performance. Therefore, we have introduced a parameter M that is used to postpone the estimation of $\hat{\mathbf{A}}$ until a relatively good ML estimate of the state variables $\hat{\mathbf{H}}$ has been found. We intuitively set M to half of the maximum number of iterations ($M = 50$). Additionally, we have defined a parameter q that is used to anneal (or weight) the predictions, and it was experimentally set to 0.15 in our experiments. Intuitively, this heuristic trick takes into account the uncertainties (as the covariances in the Kalman filtering), and it was found to be beneficial in our simulations.

Algorithm 2 summarizes our filtering algorithm where the model parameters (including $\mathbf{W}$ and $\mathbf{A}$) are learned a priori and held fixed during the process, as it is done in classical Kalman filtering. Here, motivated by the simulated annealing, we use an adaptive annealing of the predictions. Intuitively, the predictions are effectively used in the first iterations to prevent the EM algorithm to get stuck in a local maximum. Then, the predictions

are smoothed over the iterations causing the NMF approximation to be a better fit to the current observation. Moreover, for practical problems where the dynamics of unseen data can never be learned accurately, this adaptive annealing makes the algorithm more robust.

Algorithm 2 Proposed dynamic NMF: filtering algorithm applied at time t.

1) Set the predefined variable $q < 1$
2) Initialize $\hat{\mathbf{h}}_t$ with positive random numbers. Load the model parameters $\mathbf{W}$ and $\mathbf{A}$ learned using Algorithm 1. Set $r = 1$.
   % Predict
3) Compute predictions:
   a) Compute $\hat{\boldsymbol{\eta}}_t$ using **(16)**.
   b) Backup the prediction $\mathbf{b}_t = \hat{\boldsymbol{\eta}}_t$.
   %Update
4) Repeat until convergence:
   a) Anneal the prediction as $\hat{\eta}_{it} = b_{it}^{q/r}$.
   b) Update the state estimate $\hat{\mathbf{h}}_t$ according to **(17)**.
   c) $r = r + 1$.

IV. Numerical Simulations

In this section, we present our experimental results using the proposed D-NMF algorithm. We have performed simulations for three examples, namely tracking the frequency of a single sinusoid in noise (Section IV-A), separation of two signals with similar basis (Section IV-B), and speech denoising (Section IV-C). Since the original time-domain signals in the described examples can take negative values, we need to transform them to a nonnegative domain. For this purpose, we apply a discrete Fourier transformation (DFT) to Hann-windowed (overlapping) short-time frames to obtain a complex-valued time-frequency representation of the input signals. We then use the magnitudes of the DFT coefficients to construct the nonnegative observation matrix to be used with NMF. We compare the performance of the proposed D-NMF approach using objective measures with the performance of the static NMF approach [19] and two other NMF approaches that exploit the temporal dynamics, namely the N-HMM approach in [7] and a frame stacking approach [2]. The signal-to-noise ratio (SNR) is used to quantify the noise level in the observations. Denoting the clean (not known to the algorithms) and noisy time-domain signals as x and y, the input SNR is defined as:

$$\text{Input SNR} = 10\log_{10}\frac{\sum_n \mathsf{x}_n^2}{\sum_n \left(\mathsf{y}_n - \mathsf{x}_n\right)^2}, \tag{19}$$

where n is the sample index.

A. Tracking the Frequency of a Single Sinusoid in Noise

In this section, the performance of the proposed D-NMF approach is demonstrated using a tracking example. Estimation of the frequency and phase of sinusoids in noise is still an active area of research [27]. In this experiment, we aim to estimate the frequency of a single sinusoid in the presence of noise with high levels. The target signal is sampled at a sampling frequency of 8 kHz. The frequency of the sinusoid is time-varying, and increases from 0.24 radians/sample (300 Hz) to 2.9 radians/sample (3700 Hz) and then reduces to 0.24 radians/sample again. The DFT with a frame length of 128 samples and a (non-overlapping) Hann window was applied and the obtained magnitude spectrogram was used as the nonnegative observation matrix[3]. Fig. 1 depicts the noise-free observation matrix. Here, the k-th element of $\mathbf{x}_t$ is proportional to the signal's energy at a specific frequency given by $2\pi(k-1)/128$ radians/sample.

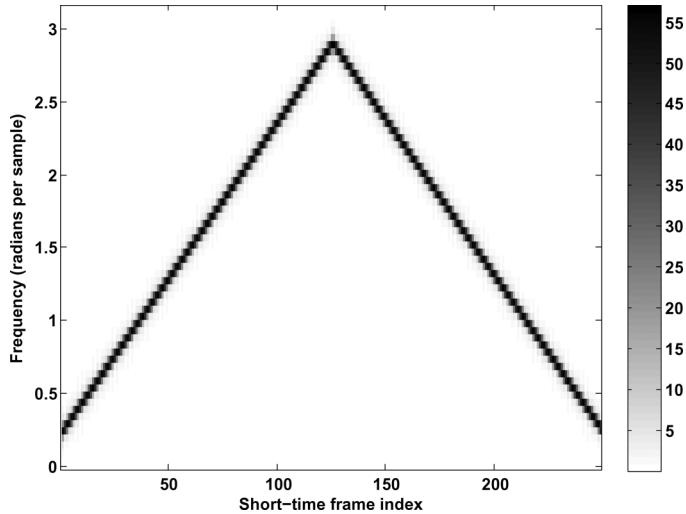

Fig. 1. Time-frequency representation of the DFT magnitudes of a single sinusoid with time-varying frequency. First the frequency increases, and before reaching the Nyquist frequency (at the 127-th frame) it gradually reduces.

For the simulations, white Gaussian noise was added to the target signal at various input SNRs. In the NMF approaches, the basis matrix $\mathbf{W}$ was predefined (and was held fixed) as the identity matrix of size 65×65. We set $J = 1$, and since we do not expect any large jump of the frequency, we predefined $\mathbf{A}_1$ such that the diagonal elements and their adjacent neighbors have a value of 1/3 while the rest of the elements are set to zero. This assumption means that the frequency will either stay constant or will smoothly increase or decrease to a higher or a lower value, respectively.

To estimate the frequency in each short-time frame, NMF or D-NMF ($J = 1$, $q = 0.25$ in Algorithm 2) was first applied and then the frequency was computed as $\hat{\Omega}_t = 2\pi(k_{\max}-1)/128$ radians/sample in which $k_{\max}$ is the index of the maximum entry of $\mathbf{h}_t$. For comparison purposes, the frequency was also estimated using an N-HMM approach. The N-HMM consisted of 65 states with one spectral vector per state. The same basis matrix $\mathbf{W}$ and $\mathbf{A}_1$ were used to predefine the N-HMM state spectral vectors and transition matrix. This N-HMM is effectively an HMM where the state-conditional likelihoods are computed using a multinomial distribution. For the N-HMM approach, $k_{\max}$ is the index of the state with the

[3]Although the DFT results in a $128 \times T$-dimensional matrix, because of the symmetry property of the DFT for real-valued signals, we only use the first $K = 65$ rows as the observation matrix.

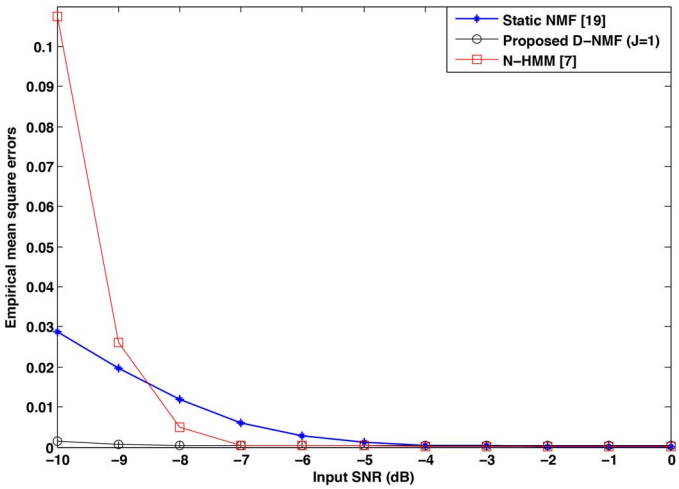

Fig. 2. Empirical mean square errors in tracking the frequency of a single sinusoid as a function of the input SNR.

highest posterior probability, which is determined by applying the forward algorithm [28].

The tracking performance is evaluated using the empirical mean square error:

$$\text{MSE} = \frac{1}{LT}\sum_{l=1}^{L}\sum_{t=1}^{T}\left(\hat{\Omega}_t - \Omega_t\right)^2, \tag{20}$$

where Ω_t is the ground-truth frequency, and L is the number of Monte Carlo runs that is set to 50 in our simulations. Fig. 2 shows the MSE as a function of the input SNR. As can be seen, the D-NMF approach provides a significantly smaller error compared to the static NMF and N-HMM approaches, especially at low input SNRs. The performance of the N-HMM approach degrades quickly at low input SNRs, which indicates that the approach is not as robust as D-NMF to high noise levels. The difference arises from the fact that in the N-HMM approach, the state-conditional likelihoods are used during the forward algorithm, which are sensitive to high noise levels and exhibit a large dynamic range. For the D-NMF approach, however, the posterior probabilities $f\left(z_t = i \mid k, \boldsymbol{\lambda}\right)$ that are used to compute $\hat{\mathbf{h}}_t$ (in (17)) have a smaller dynamic range and the noise effect can be more effectively compensated by using the temporal continuity. For higher input SNRs, the input data matrix exhibits a clearer energy distribution (closer to the noise-free case) such that applying the NMF and the N-HMM approaches will also lead to good results. The simulation results shows that at an input SNR of about -3 dB, applying D-NMF leads to slightly larger error than static NMF. This error is due to the additional latency that is imposed by using the previous observations to predict the current state variables.

B. Separation of Two Signals With Similar Basis

In the second experiment, we applied our proposed D-NMF approach as a supervised separation approach for separating two sources that share a similar basis matrix $\mathbf{W}$. In this experiment, two sources (each consisting of two sinusoids with time-varying frequencies) were added at an input SNR of 0 dB to obtain the

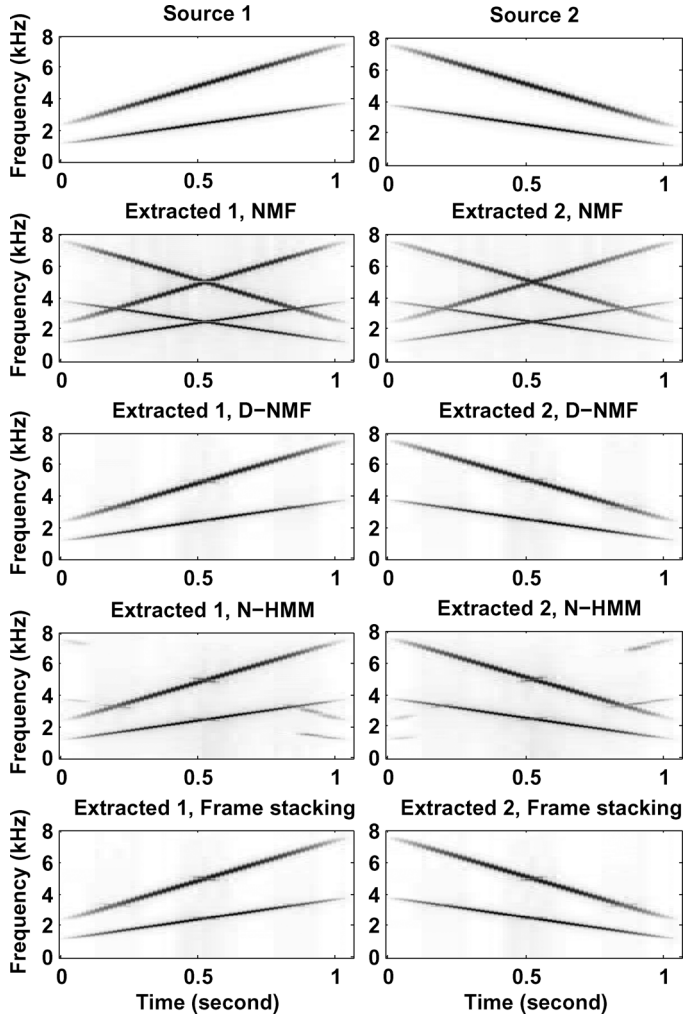

Fig. 3. Original and separated sources using NMF, D-NMF ($J = 2$), N-HMM, and frame stacking approaches.

time-domain mixture. The DFT was applied using overlapping Hann windows with a frame length of 1024 samples and an update length of 256 samples. The sampling frequency was 16 kHz. The magnitude spectrogram of the two sources are separately shown in the top panel of Fig. 3. Although these sources share a similar basis matrix (because they are just time-reversed versions of each other) they have a very different dynamic behavior. The frequencies of source 1 are increasing, while the frequencies of source 2 are decreasing.

To learn the model parameters, the static NMF and D-NMF approaches were applied on the observations of each source separately. The number of basis vectors was set to $I = 50$ for each source both for NMF and for D-NMF. For the D-NMF approach, in addition to the basis matrix $\mathbf{W}$, the N-VAR model parameters $\mathbf{A}_j, j = 1 \ldots J$ were learned for $J \in \{1, 2, 3, 4, 5\}$. The annealing parameter q was set to 0.1 in Algorithm 2. In addition to the static NMF and the proposed D-NMF approaches, an N-HMM approach [7] and a frame stacking approach [2] were implemented as alternative methods that exploit the temporal dynamics in NMF. For the N-HMM approach, 50 states with one spectral vector per state were learned for each source. For the frame stacking approach, 8 consecutive frames were

stacked to obtain 4096-dimensional vectors and a tall basis matrix with $I = 50$ columns were learned to represent each source. In this experiment and also in Section IV-C, the high-resolution DFT-domain magnitude spectral vectors are stacked rather than the low-resolution mel-domain spectral vectors which is proposed in [2] because the DFT version outperformed the mel counterpart. The number of N-HMM states, the number of basis vectors, and the number of consecutive frames to be stacked were experimentally set to get the best performance.

To model the mixture, we assume that the DFT magnitude of the mixture is (approximately) equal to the sum of the magnitudes of the DFT coefficients of the two sources [3], [6], [7], i.e., $\mathbf{x}_t \approx \mathbf{x}_t^{(1)} + \mathbf{x}_t^{(2)}$, where the superscripts represent the source numbers. For the NMF, D-NMF and frame stacking approaches, the basis matrix of the mixture is constructed by concatenating the (learned) individual basis matrices, i.e, $\mathbf{W} = [\,\mathbf{W}^{(1)} \quad \mathbf{W}^{(2)}\,]$. Similarly, for the D-NMF approach, the N-VAR parameters of the mixture are constructed by concatenating the (learned) individual N-VAR parameters, i.e., $\mathbf{A}_1 = \left[\begin{array}{cc} \mathbf{A}_1^{(1)} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_1^{(2)} \end{array}\right]$ where $\mathbf{0}$ is a 50×50 zero matrix. For the N-HMM approach, a factorial N-HMM [7] is constructed to model the mixture.

For the separation, the basis matrix $\mathbf{W}$ and N-VAR parameters $\mathbf{A}_j$ are held fixed and only the NMF coefficients $\mathbf{H} = [\,\mathbf{H}^{(1),\top} \quad \mathbf{H}^{(2),\top}\,]^\top$ are estimated. For all the approaches, after convergence of the estimation algorithm, the magnitude of the DFT coefficients of the individual sources are estimated using a Wiener reconstruction [17]:

$$\hat{\mathbf{x}}_t^{(1)} = \frac{\mathbf{W}^{(1)}\mathbf{h}_t^{(1)}}{\mathbf{W}^{(1)}\mathbf{h}_t^{(1)} + \mathbf{W}^{(2)}\mathbf{h}_t^{(2)}} \odot \mathbf{x}_t, \tag{21}$$

where $\odot$ represents the element-wise multiplication, and division is performed element by element. The separated signals using the NMF, D-NMF ($J = 2$), N-HMM, and frame stacking approaches are shown in Fig. 3. As can be seen, the static NMF approach is not able to separate the sources because of the ambiguity that is caused by the similarity of the individual basis matrices. On the other hand, the three other approaches, lead to a satisfactory separation of the sources by benefiting from the temporal dependencies, where the D-NMF and frame stacking approaches have clearly led to a better separation compared to the N-HMM approach.

To quantify the separation performance, the output SNR was computed as:

$$\text{Output SNR} = 10\log_{10}\frac{\sum\limits_n \mathsf{x}_n^2}{\sum\limits_n \left(\hat{\mathsf{x}}_n - \mathsf{x}_n\right)^2}, \tag{22}$$

where x is the time-domain signal corresponding to one of the sources, and $\hat{\mathsf{x}}$ is the separated time-domain signal, obtained by applying the overlap-add procedure to the separated magnitude spectrogram, where the phase of the mixture signal was used to compute the inverse DFT.

Fig. 4 shows the output SNR as a function of the N-VAR model order (J). Here, $J = 0$ corresponds to the static NMF approach with no temporal modeling. As can be seen, including temporal dynamics in NMF has improved the output SNR by more than 11 dB. By increasing J, the performance slightly improves, reaching its maximum at $J = 3$ for this experiment. Moreover, as also shown in Fig. 3, the D-NMF and frame stacking approaches have produced higher output SNRs compared to the N-HMM approach, where D-NMF has led to the best separation performance.

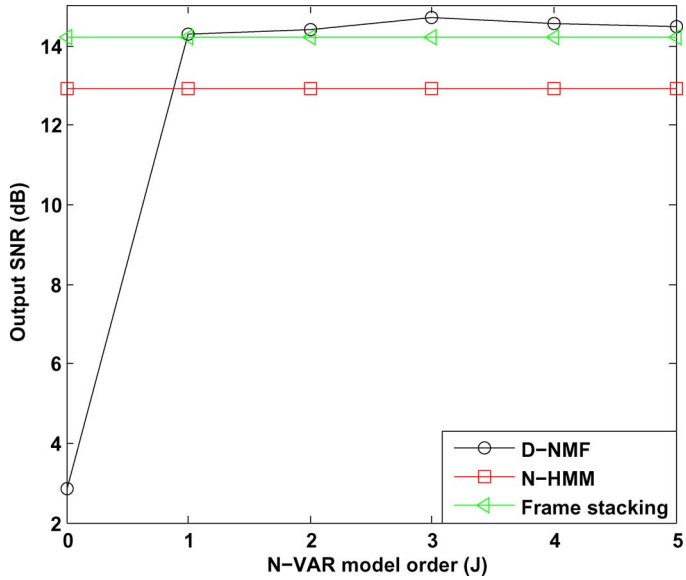

Fig. 4. Output SNR as a function of the N-VAR model order J. $J = 0$ corresponds to the static NMF approach with no temporal modeling.

C. Denoising

As the last experiment, we applied our proposed D-NMF approach to a speech denoising problem. In this experiment, speech signals are degraded by additive noise and the goal is to suppress the noise and estimate the speech component given the noisy observations. The speech signals were degraded with multitalker babble noise or factory noise at input SNRs in the range -5 dB to 5 dB. The speech and noise signals were taken from the TIMIT [29] and NOISEX-92 [30] databases, respectively. The signals were sampled at a sampling frequency of 16 kHz. The DFT analysis was performed with the same parameters as in Section IV-B.

For each noise type, an NMF model was learned using the first 75% of the noise signals and the last 25% was used to test the algorithms. The noise type is assumed to be known to choose a suitable noise-dependent NMF model for denoising. This assumption is practical for some applications and the required information can be provided by state-of-the-art environment classification techniques (see [6] for a discussion on this topic). The denoising was performed under two conditions, depending on the available information about the speaker identity. In the matched condition, the speaker identity is assumed to be known and speaker-dependent (SD) speech models were used in all the approaches. These models were learned using 9 speech sentences from each speaker, and another sentence from the same speaker was used to test the algorithms. Alternatively, in the mismatched case, a universal speaker-independent (SI) speech model was learned using 200 speech sentences from different speakers. The denoising experiments were repeated for 20

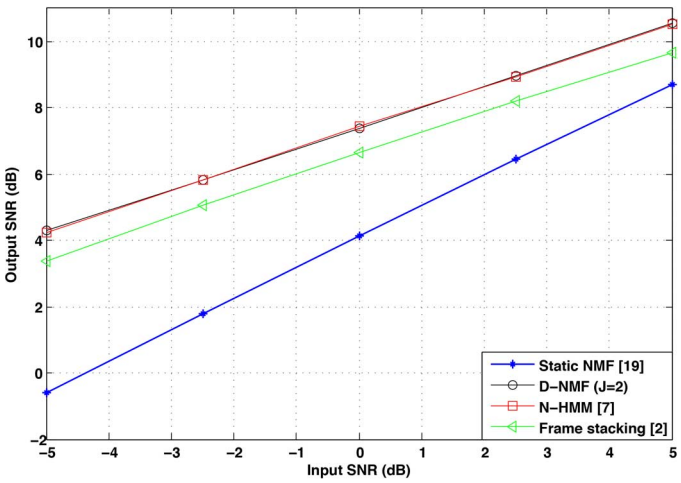

Fig. 5. Averaged output SNR over the factory and babble noise types under the matched speaker-dependent condition. Noise type and speaker identity are assumed to be known a priori and they are used to select noise- and speaker-dependent models for denoising.

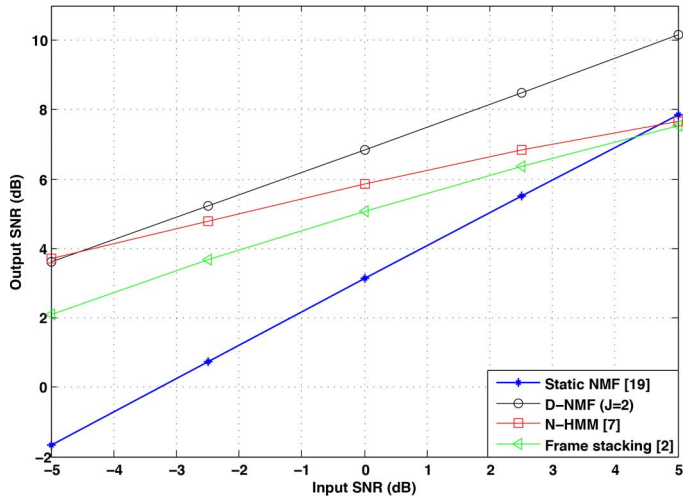

Fig. 6. Averaged output SNR over the factory and babble noise types, where a universal speaker-independent speech model is used for denoising. Noise type is assumed to be known a priori and is used to select a noise-specific NMF (or N-HMM) model for denoising.

different speakers, where the training and test data were disjoint in all the simulations. For all the methods, the speech DFT magnitudes were estimated using the Wiener reconstruction (21).

The number of basis vectors for speech and noise were experimentally found for each approach to obtain the best results. For the NMF, D-NMF, and frame stacking approaches, $I = 60$ speech basis vectors were learned for both SD and SI models, where for the N-HMM approach, 40 and 60 states each consisting of 10 spectral vectors were respectively learned for the SD and SI models. For the NMF and D-NMF approaches, 20 basis vectors were learned for each noise type, where for the frame stacking approach, 100 and 150 basis vectors were learned for babble and factory noise, respectively. For the N-HMM approach a single-state model was learned for each noise type, where the number of spectral vectors was set to 20 (for both noise types in the SD condition) and to 20 and 100 (in SI condition) for babble and factory noise types, respectively. For the D-NMF approach, the N-VAR model parameters were learned for $J \in \{1, 2, 3, 4, 5\}$, where the annealing parameter q was experimentally set to 0.3 (for speech) and 0.1 (for both noise types) in Algorithm 2.

Fig. 5 shows the results (averaged over both noise types) of our denoising experiment for the matched SD condition (with $J = 2$ for the D-NMF approach). The figure shows the output SNR, defined in (22), as a function of the input SNR in the range of -5 dB to 5 dB. The simulation results show that the D-NMF and N-HMM approaches have a similar denoising performance, while they significantly outperform the static NMF approach for all considered input SNRs. The difference is maximum at the lowest input SNR (-5 dB), where the D-NMF approach results in around 4.5 dB higher output SNR. Moreover, the frame stacking approach has a considerably improved performance compared to the NMF approach, but is worse than the D-NMF and N-HMM approaches.

The results of the denoising approaches (averaged over both noise types) for the mismatched SI condition are shown in Fig. 6. The results show that the D-NMF ($J = 2$) approach outperforms both the N-HMM and frame stacking approaches, where the difference is more than 2 dB at an input SNR equal to 5 dB. Comparing Figs. 5 and 6, we see that a higher input SNR is obtained under the SD condition.

Fig. 7 shows the output SNR as a function of the N-VAR model order J, for the SI condition and at an input SNR equal to 0 dB. The results for factory noise and babble noise types are plotted in the top and bottom panels, respectively. The static NMF is shown as a special case of D-NMF with $J = 0$. The results show that a significant improvement is obtained by incorporating the temporal dynamics into the denoising process. For the factory noise, a small improvement is obtained by increasing J to 3, while for the babble noise the best performance is obtained at $J = 1$. In both cases, it can be seen that a single-lag predictor with $J = 1$ can be used to achieve a good denoising performance.

Finally, it is interesting to compare the computational complexity and the memory requirement of the proposed D-NMF approach to the N-HMM and frame stacking approaches. To have a better understanding, we simply provide an estimate of the required time to process one second of speech in the SD denoising example in our implementation in a PC with 3.4 GHz Intel CPU and 8 GB RAM. It should be mentioned that this time can be significantly reduced for all the approaches by using an optimized implementation. Our D-NMF approach requires around 1.5 seconds to process 1 second of input signal, while the N-HMM and frame stacking approaches require 40 and 0.75 seconds, respectively. Considering the memory requirements (to store the learned model parameters), D-NMF requires less than 25% and 10% of the memory required by the N-HMM and frame stacking approaches, respectively. As a result, the proposed approach is more suitable for real-time applications with power or memory restrictions.

V. Conclusions

In this paper, we presented a state-space representation for nonnegative observations and nonnegative state variables,

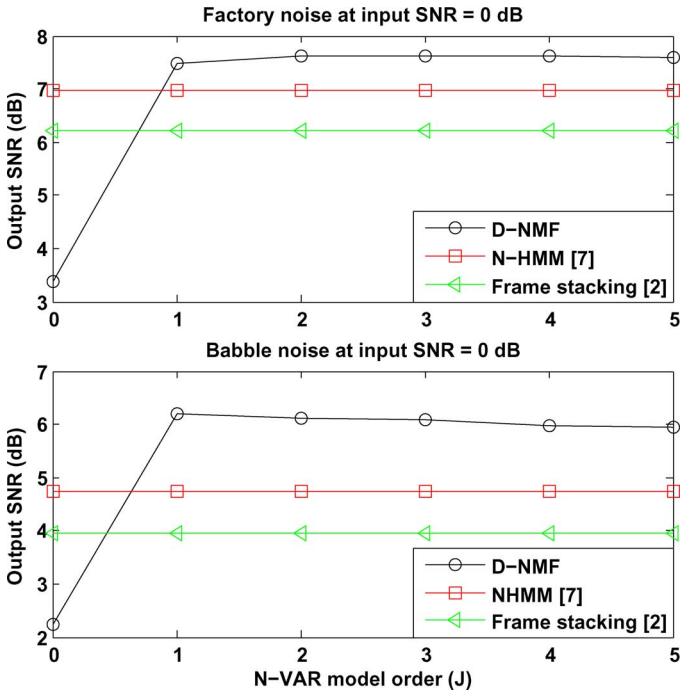

Fig. 7. Output SNR corresponding to the factory noise (top panel) and babble noise (bottom panel) as a function of the N-VAR model order J at input $\mathrm{SNR} = 0$ dB. Universal speaker-independent speech model is used for denoising.

which is able to efficiently model the temporal dependencies. Since the classical Kalman filtering is not appropriate for this setup, we derived a novel algorithm, referred to as D-NMF, to learn the model parameters, and we developed a novel filtering approach for the state variables. Using an iterative EM-based estimation algorithm, an ML estimate of the basis matrix and the N-VAR model parameters is computed. We showed that computing the ML estimate of the N-VAR parameters is equivalent to applying IS-NMF in which the observations and the NMF coefficients are the estimates of the state variables and their shifted versions, respectively. As for the state variables, the algorithm provides a MAP solution that, similar to the Kalman filtering, consists of a prediction step and an update step. We demonstrated the algorithm using three examples targeting tracking, separation, and denoising applications. The results show that exploiting the temporal dynamics in NMF can improve the performance significantly, especially at low input SNRs. Moreover, our experimental results show that the proposed approach outperforms an N-HMM and a frame stacking approach where it also requires substantially less computational power and memory, and hence, it is a better alternative for real-time applications. Finally, our approach to model the temporal dependencies is causal, i.e., it only uses the past observations to process the current observation. Therefore, unlike the frame stacking approach that has an inherent delay of several time steps, our approach does not impose any delay on the processed signals.

REFERENCES

[1] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Adv. Neural Inf. Process. Systems (NIPS)*. Cambridge, MA, USA: MIT Press, 2000, pp. 556–562.

[2] J. F. Gemmeke, T. Virtanen, and A. Hurmalainen, "Exemplar-based sparse representations for noise robust automatic speech recognition," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 7, pp. 2067–2080, Sep. 2011.

[3] P. Smaragdis, "Convolutive speech bases and their application to supervised speech separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 1, pp. 1–12, Jan. 2007.

[4] W. Wang, A. Cichocki, and J. A. Chambers, "A multiplicative algorithm for convolutive non-negative matrix factorization based on squared Euclidean distance," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2858–2864, Jul. 2009.

[5] C. Févotte, N. Bertin, and J. L. Durrieu, "Nonnegative matrix factorization with the Itakura-Saito divergence: With application to music analysis," *Neural Comput.*, vol. 21, pp. 793–830, 2009.

[6] N. Mohammadiha, P. Smaragdis, and A. Leijon, "Supervised and unsupervised speech enhancement using NMF," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 10, pp. 2140–2151, Oct. 2013.

[7] G. J. Mysore, P. Smaragdis, and B. Raj, "Non-negative hidden Markov modeling of audio with application to source separation," in *Proc. Int. Conf. Latent Variable Anal. Signal Separation*, 2010, pp. 140–148.

[8] M. Nakano, J. L. Roux, H. Kameoka, Y. Kitano, N. Ono, and S. Sagayama, "Nonnegative matrix factorization with Markov-chained bases for modeling time-varying patterns in music spectrograms," in *Proc. Int. Conf. Latent Variable Anal. Signal Separation*, 2010, pp. 149–156.

[9] N. Mohammadiha and A. Leijon, "Nonnegative HMM for babble noise derived from speech HMM: Application to speech enhancement," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 5, pp. 998–1011, May 2013.

[10] N. Mohammadiha, W. B. Kleijn, and A. Leijon, "Gamma hidden Markov model as a probabilistic nonnegative matrix factorization," presented at the Eur. Signal Process. Conf. (EUSIPCO), Sep. 2013.

[11] P. Smaragdis, C. Févotte, G. J. Mysore, N. Mohammadiha, and M. Hoffman, "Static and dynamic source separation using nonnegative factorizations: A unified view," *IEEE Signal Process. Mag.*, vol. 31, no. 3, pp. 66–75, May 2014.

[12] T. Kailath, A. H. Sayed, and B. Hassibi, *Linear Estimation*. Englewood Cliffs, NJ, USA: Prentice Hall, 2000.

[13] D. Simon, *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. New York, NY, USA: Wiley, 2006.

[14] S. J. Julier and J. J. LaViola, "On Kalman filtering with nonlinear equality constraints," *IEEE Trans. Signal Process.*, vol. 55, no. 6, pp. 2774–2784, Jun. 2007.

[15] D. Simon, "Kalman filtering with state constraints: A survey of linear and nonlinear algorithms," *IET Contr. Theory Appl.*, vol. 4, no. 8, pp. 1303–1318, Aug. 2010.

[16] A. Carmi, P. Gurfil, and D. Kanevsky, "Methods for sparse signal recovery using Kalman filtering with embedded pseudo-measurement norms and quasi-norms," *IEEE Trans. Signal Process.*, vol. 58, no. 4, pp. 2405–2409, Apr. 2010.

[17] N. Mohammadiha, P. Smaragdis, and A. Leijon, "Prediction based filtering and smoothing to exploit temporal dependencies in NMF," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Process. (ICASSP)*, May 2013, pp. 873–877.

[18] C. Févotte, J. L. Roux, and J. R. Hershey, "Non-negative dynamical system with application to speech and audio," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, May 2013, pp. 3158–3162.

[19] P. Smaragdis, B. Raj, and M. V. Shashanka, "A probabilistic latent variable model for acoustic modeling," presented at the Adv. Models Acoust. Process. Workshop, NIPS, 2006.

[20] N. Mohammadiha, T. Gerkmann, and A. Leijon, "A new linear MMSE filter for single channel speech enhancement based on nonnegative matrix factorization," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust. (WASPAA)*, Oct. 2011, pp. 45–48.

[21] N. Mohammadiha, J. Taghia, and A. Leijon, "Single channel speech enhancement using Bayesian NMF with recursive temporal updates of prior distributions," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Mar. 2012, pp. 4561–4564.

[22] G. J. Mysore and M. Sahani, "Variational inference in non-negative factorial hidden Markov models for efficient audio source separation," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2012.

[23] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York: Springer-Verlag, 2006.

[24] M. Shashanka, "Latent Variable Framework for Modeling and Separating Single Channel Acoustic Sources," Ph.D. dissertation, Dept. Cogn. Neural Syst., Boston Univ., Boston, MA, USA, 2007.

[25] J. D. Hamilton, *Time Series Analysis*. Princeton, NJ: Princeton Univ. Press, 1994.
[26] H. Lütkepohl, *New Introduction to Multiple Time Series Analysis*. Berlin, Germany: Springer-Verlag, 2005.
[27] H. Fu and P.-Y. Kam, "MAP/ML estimation of the frequency and phase of a single sinusoid in noise," *IEEE Trans. Signal Process.*, vol. 55, no. 3, pp. 834–845, Mar. 2007.
[28] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, Feb. 1989.
[29] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, *TIMIT Acoustic-Phonetic Continuous Speech Corpus*. Philadelphia, PA, USA: Linguistic Data Consortium, 1993.
[30] A. Varga and H. J. M. Steeneken, "Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Commun.*, vol. 12, no. 3, pp. 247–251, Jul. 1993.

Nasser Mohammadiha (S'11–M'13) received the M.Sc. degree in Electrical Engineering from Sharif University of Technology, Tehran, Iran, in 2006. He worked on digital hardware and software design until 2008. He completed his Ph.D. studies at the Communication Theory Laboratory, KTH Royal Institute of Technology, Stockholm, Sweden, in 2013. Nasser received the Best Student Paper Award at the European Signal Processing Conference, 2013.

He is currently a postdoctoral fellow at the University of Oldenburg, Germany. His research interests include machine learning for signal processing, statistical signal modeling, speech processing, and image processing.

Paris Smaragdis (M'03–F'14) is faculty in the Computer Science and the Electrical and Computer Science departments at the University of Illinois at Urbana-Champaign. He is also a senior research scientist at Adobe. He completed his graduate and postdoctoral studies at MIT, where he conducted research on computational perception and audio processing. Prior to the University of Illinois he was a senior research scientist at Adobe Systems and a research scientist at Mitsubishi Electric Research Labs, during which time he was selected by the MIT Technology Review as one of the top 35 young innovators of 2006. Paris' research interests lie in the intersection of machine learning and signal processing, especially as they apply to audio problems.

Ghazaleh Panahandeh (S'11) received the M.Sc. degree in Electrical Engineering from Sharif University of Technology, Tehran, Iran, in 2008. She completed her Ph.D. studies at the Signal Processing Department, KTH Royal Institute of Technology, Stockholm, Sweden, in 2014. Her research interests include inertial navigation and positioning, vision-aided inertial navigation, estimation theory, and applied machine learning. She is currently working on active safety at Volvo cars, Gothenburg, Sweden. She is also a part-time postdoctoral fellow at the Signal Processing Department, KTH Royal Institute of Technology, Stockholm.

Simon Doclo (S'95–M'03-SM'13) received the M.Sc. degree in electrical engineering and the Ph.D. degree in applied sciences from the Katholieke Universiteit Leuven, Belgium, in 1997 and 2003. From 2003 to 2007 he was a Postdoctoral Fellow with the Research Foundation – Flanders at the Electrical Engineering Department (Katholieke Universiteit Leuven) and the Adaptive Systems Laboratory (McMaster University, Canada). From 2007 to 2009 he was a Principal Scientist with NXP Semiconductors at the Sound and Acoustics Group in Leuven, Belgium. Since 2009 he is a full professor at the University of Oldenburg, Germany, and scientific advisor for the project group Hearing, Speech and Audio Technology of the Fraunhofer Institute for Digital Media Technology. His research activities center around signal processing for acoustical and biomedical applications, more specifically microphone array processing, active noise control, acoustic sensor networks and hearing aid processing. Prof. Doclo received the Master Thesis Award of the Royal Flemish Society of Engineers in 1997 (with Erik De Clippel), the Best Student Paper Award at the International Workshop on Acoustic Echo and Noise Control in 2001, the EURASIP Signal Processing Best Paper Award in 2003 (with Marc Moonen) and the IEEE Signal Processing Society 2008 Best Paper Award (with Jingdong Chen, Jacob Benesty, Arden Huang). He was secretary of the IEEE Benelux Signal Processing Chapter (1998–2002), member of the IEEE Signal Processing Society Technical Committee on Audio and Acoustic Signal Processing (2008–2013), and Technical Program Chair for the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA) in 2013. Prof. Doclo has served as guest editor for several special issues (IEEE Signal Processing Magazine, Elsevier Signal Processing) and is associate editor for the EURASIP Journal on Advances in Signal Processing.