Online Single-Microphone Source Separation using Non-Linear Autoregressive Models

Bart van Erp b.v.erp@tue.nl
Eindhoven University of Technology, Eindhoven, the Netherlands

Bert de Vries Bert.de.Vries@tue.nl
Eindhoven University of Technology, Eindhoven, the Netherlands
GN Hearing, Eindhoven, The Netherlands

Abstract

In this paper a modular approach to single-microphone source separation is proposed. A probabilistic model for mixtures of observations is constructed, where the independent underlying source signals are described by non-linear autoregressive models. Source separation in this model is achieved by performing online probabilistic inference through an efficient message passing procedure. For retaining tractability with the non-linear autoregressive models, three different approximation methods are described. A set of experiments shows the effectiveness of the proposed source separation approach. The source separation performance of the different approximation methods is quantified through a set of verification experiments. Our approach is validated in a speech denoising task.

Keywords: Kalman filtering; Message passing; Non-linear autoregressive models; Probabilistic inference; Source separation.

1. Introduction

Source separation is a fundamental problem with the goal of extracting the constituent sources from an observed signal. This problem underlies applications such as denoising, where the observed signal constitutes a signal of interest and a noise signal. The field of source separation is well-developed (Comon, 1994; Hong et al., 2004; Fevotte and Godsill, 2006; Erdogan, 2008; Rennie et al., 2010; Magron and Virtanen, 2018), and a wide variety of methods have been developed to solve this problem. We constrain our scope to online source separation: given current and previous observations $\boldsymbol{y}_{1:t}$ composed of a signal of interest $\boldsymbol{s}_{1:t}$ and noise signal $\boldsymbol{n}_{1:t}$, the goal is to extract the samples s_t and n_t. We follow the probabilistic school of thought, as brilliantly presented by Knuth (2013). In this approach, a generative model for the observations is constructed as a function of the constituent sources. Source separation is then phrased as a probabilistic inference problem, where we aim to track the latent constituent sources. This approach allows us to incorporate any available prior information about the constituent sources to aid the source separation process.

Despite the elegance of this approach, it is often hard to find tractable inference solutions, especially when more complicated source models are involved. Although many approximate inference solutions exist (Beal, 2003; Minka, 2001; Dauwels et al., 2005), most of them suffer from error-prone lengthy manual derivations of posterior updates. This reflects the need of flexible source models that easily submit to (approximate) inference. This paper introduces a modular and easily explainable source separation approach where the individual signals are independently modeled by non-linear autoregressive models. However, extensions to arbitrary non-linear functions, such as neural networks, are trivial. Approximate probabilistic inference then is enabled through linearization (Särkkä, 2013, Ch.5) or different versions of the unscented transform (Julier and Uhlmann, 1997; Wan and Van Der Merwe, 2000; Julier, 2002, 2003). With respect to earlier works (Wan and Nelson, 1997; Dutt et al., 2021), we modularize our approach by phrasing inference as an efficient and automatable message passing procedure that allows for the straightforward extension towards multiple sources. Furthermore, we compare the different approximate inference solutions through experiments.

In short, this paper proposes an online approach to single microphone-based source separation using non-linear autoregressive models and makes the following contributions:

- We propose a modular probabilistic model for a signal mixture in Section 2, in which the constituent sources are represented by non-linear autoregressive models.
- Online probabilistic inference in this model is realized in Section 3 through automated message passing in a factor graph, leading to the separation of the constituent sources.
- Three approximation techniques for obtaining efficient and tractable inference in this model with the non-linear state transitions are provided in Section 3.3.
- We verify the proposed methodology through a set of verification experiments and compare the performance of the different approximation techniques in Section 4. Furthermore, we apply the proposed model in a speech denoising task.

2. Model specification

Let $\boldsymbol{y} = [y_1, y_2, \dots, y_T]^\top \in \mathbb{R}^T$ denote a vector of T observations. Observation y_t at index t is modeled by the independent signal of interest s_t and noise signal n_t. We assume the joint distribution over $\boldsymbol{y}$, $\boldsymbol{s} = [s_0, s_1, \dots, s_T]^\top$ and $\boldsymbol{n} = [n_0, n_1, \dots, n_T]^\top$ to be factorized as

$$p(\boldsymbol{y}, \boldsymbol{s}, \boldsymbol{n}) = \underbrace{p(\boldsymbol{s}_0)p(\boldsymbol{n}_0)}_{\text{prior}} \prod_{t=1}^{T} \underbrace{p(y_t \mid \boldsymbol{s}_t, \boldsymbol{n}_t)}_{\text{mixing model}} \underbrace{p(\boldsymbol{s}_t \mid \boldsymbol{s}_{t-1}) p(\boldsymbol{n}_t \mid \boldsymbol{n}_{t-1})}_{\text{state transition models}}. \tag{1}$$

The mixing model $p(y_t \mid \boldsymbol{s}_t, \boldsymbol{n}_t)$ in (1) describes how the observation y_t is formed from the latent states $\boldsymbol{s}_t = [s_t, s_{t-1}, \dots, s_{t-M_s+1}]^\top \in \mathbb{R}^{M_s}$ and $\boldsymbol{n}_t = [n_t, n_{t-1}, \dots, n_{t-M_n+1}]^\top \in \mathbb{R}^{M_n}$, where M_s and M_n represent both the lengths of the state vectors and the orders of the non-linear autoregressive models. This mixing model is defined as

$$p(y_t \mid \boldsymbol{s}_t, \boldsymbol{n}_t) = \mathcal{N}(y_t \mid \boldsymbol{e}_1^\top \boldsymbol{s}_t + \boldsymbol{e}_1^\top \boldsymbol{n}_t,\ \sigma_y^2), \tag{2}$$

where $\sigma_y^2 \in \mathbb{R}_{>0}$ denotes the observation noise variance and where $\boldsymbol{e}_1 = [1, 0, \dots, 0]^\top$ represents the first Cartesian unit basis vector of appropriate length. The inner product of $\boldsymbol{s}_t$ and $\boldsymbol{n}_t$ with this vector denotes the selection of the first entry as $s_t = \boldsymbol{e}_1^\top \boldsymbol{s}_t$.

The state transition models for the underlying signals $\boldsymbol{s}_t$ and $\boldsymbol{n}_t$ are specified as

$$p(\boldsymbol{s}_t \mid \boldsymbol{s}_{t-1}) = \mathcal{N}(\boldsymbol{s}_t \mid g_s(\boldsymbol{s}_{t-1}),\ \Sigma_s), \tag{3a}$$

$$p(\boldsymbol{n}_t \mid \boldsymbol{n}_{t-1}) = \mathcal{N}(\boldsymbol{n}_t \mid g_n(\boldsymbol{n}_{t-1}),\ \Sigma_n), \tag{3b}$$

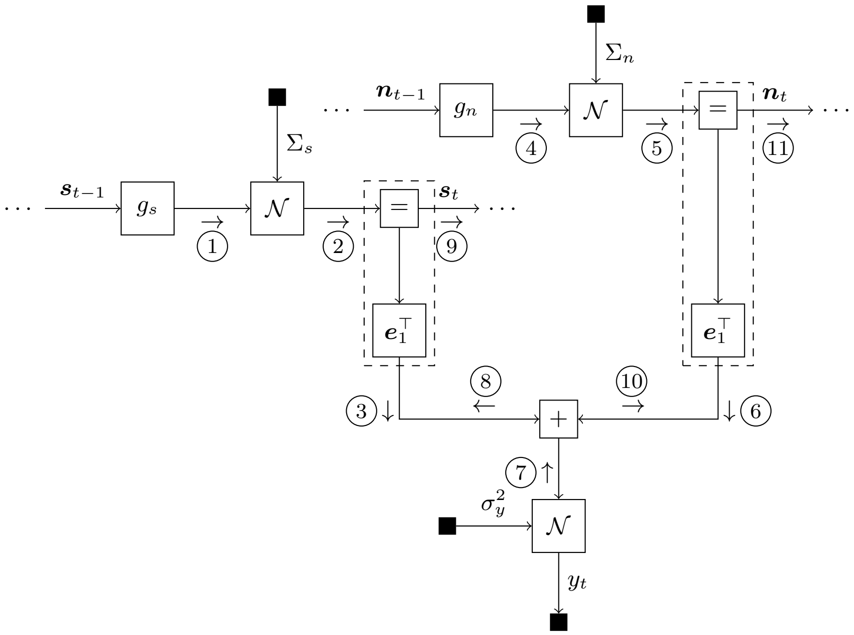


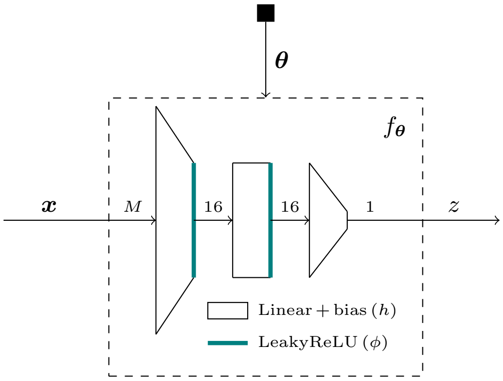
Figure 1: A Forney-style factor graph representation of a single time slice of the probabilistic model of Section 2. Probabilistic inference in this model corresponds to the source separation algorithm. For efficient probabilistic inference, some elementary factor nodes are combined in compound nodes, as denoted by the dashed boxes. The encircled numbers denote the messages corresponding to the efficient message passing schedule. The messages 9 and 11 yield the solution to (5).

where $g_s(\cdot)$ and $g_n(\cdot)$ describe the non-linear autoregressive behavior of $\boldsymbol{s}_t$ and $\boldsymbol{n}_t$, respectively. These functions are similarly defined as

$$g_s(\boldsymbol{s}_t) = [f_s(\boldsymbol{s}_t), s_t, s_{t-1}, \ldots, s_{t-M_s+2}]^\top \tag{4}$$

performing a unit delay with non-linear prediction $f_s(\cdot)$. For demonstration purposes, the non-linearities f_s and f_n in (4) are both represented by a multilayer perceptron throughout this paper, as depicted in Figure 2. However, any non-linear function will suffice. The parameters Σ_s and Σ_n describe the process noise covariance matrices. As $g(\cdot)$ predominantly performs a unit delay, the state transition only models the process noise of the first element. Therefore the covariance matrices are sparse and can be represented as $\Sigma_s = \sigma_s^2 \boldsymbol{e}_1 \boldsymbol{e}_1^\top$, with $\sigma_s^2 \in \mathbb{R}_{>0}$ denoting the variance of the non-linear prediction $f_s(\cdot)$. Finally, the underlying signals are initialized with Normal priors $p(\boldsymbol{s}_0)$ and $p(\boldsymbol{n}_0)$ as specified in Section 4.

Figure 1 shows the Forney-style factor graph (FFG) (Forney, 2001) of a single time slice of the probabilistic model specified by (1)-(3). An FFG is an undirected graphical model that visualizes the factorization of a function as a graph, where nodes and edges represent factors and variables, respectively. An edge connects to a node only if the variable associated with the edge is an argument of the node function. Throughout this paper we use FFGs with notational conventions adopted from Loeliger (2004) to visualize probabilistic models.



$$f_{\boldsymbol{\theta}} = h_3 \circ \phi_2 \circ h_2 \circ \phi_1 \circ h_1$$

where

$$h_i(\boldsymbol{x}) = W_i \boldsymbol{x} + \boldsymbol{b}_i$$
$$\phi_i(\boldsymbol{x})_j = \max(l x_j, x_j)$$

Figure 2: Overview of non-linearity $z = f_{\boldsymbol{\theta}}(\boldsymbol{x})$ in the non-linear autoregressive model g as in (4). Throughout the experiments of Section 4, f represents a multilayer perceptron. The values denote the dimensionality of the (intermediate) variables. $\boldsymbol{\theta} = \{W_1, W_2, W_3, \boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3\}$ denotes the parameters of the model. The hyperparameter of the leakyReLU function is set to $l = 0.1$.

3. Probabilistic inference

3.1 Online state tracking

Separating the sources $\boldsymbol{s}$ and $\boldsymbol{n}$ from observations $\boldsymbol{y}$ using the probabilistic model of (1)-(3) is phrased as a probabilistic inference task. Probabilistic inference concerns the computation of the posterior marginal distributions in the generative model. Specifically, the goal is to infer the posterior marginal distributions of the latent signals $\boldsymbol{s}$ and $\boldsymbol{n}$ given observations $\boldsymbol{y}$, also known as latent state tracking.

This paper focuses on online source separation, where we wish to infer the values of s_t and n_t given current and previous observations $\boldsymbol{y}_{1:t}$. Concretely, we are interested in computing the posterior distribution $p(\boldsymbol{s}_t, \boldsymbol{n}_t \mid \boldsymbol{y}_{1:t})$. From this posterior distribution the marginal posterior distributions $p(s_t \mid \boldsymbol{y}_{1:t})$ and $p(n_t \mid \boldsymbol{y}_{1:t})$ can be extracted through marginalization. The posterior distribution can be computed through a modified version of the Chapman-Kolmogorov equation (Särkkä, 2013, Ch.4) as

$$\underbrace{p(\boldsymbol{s}_t, \boldsymbol{n}_t \mid \boldsymbol{y}_{1:t})}_{\text{posterior}} \propto \underbrace{p(y_t \mid \boldsymbol{s}_t, \boldsymbol{n}_t)}_{\text{mixing model}} \int \underbrace{p(\boldsymbol{s}_{t-1}, \boldsymbol{n}_{t-1} \mid \boldsymbol{y}_{1:t-1})}_{\text{prior}} \underbrace{p(\boldsymbol{s}_t \mid \boldsymbol{s}_{t-1}) p(\boldsymbol{n}_t \mid \boldsymbol{n}_{t-1})}_{\text{state transition models}} \mathrm{d}\boldsymbol{s}_{t-1} \mathrm{d}\boldsymbol{n}_{t-1}, \quad (5)$$

where the mixing model and state transition models have already been specified in (2) and (3), respectively. The prior distribution $p(\boldsymbol{s}_{t-1}, \boldsymbol{n}_{t-1} \mid \boldsymbol{y}_{1:t-1})$ is then recursively updated by the posterior distribution $p(\boldsymbol{s}_t, \boldsymbol{n}_t \mid \boldsymbol{y}_{1:t})$.

3.2 Message passing-based inference

Because of the assumed factorization in the generative model, the global integration of (5) can be performed by smaller localized computations. The results of these computations are called messages and are propagated over the edges of the graph. This procedure is known as message passing. We choose this methodology because of its modularity, efficiency, automatability and scalability (Loeliger et al., 2007; Cox et al., 2019). The sum-product message $\overrightarrow{\mu}(z_j)$ (Kschischang et al., 2001) flowing out of some node $f(z_1, z_2, \ldots, z_K)$ with incoming messages $\overrightarrow{\mu}(z_{\setminus j})$ is given by

$$\overrightarrow{\mu}(z_j) = \int f(z_1, z_2, \ldots, z_K) \prod_{k \neq j} \overrightarrow{\mu}(z_k) \, \mathrm{d}\boldsymbol{z}_{\setminus j}. \tag{6}$$

We represent the edges by arbitrarily directed arrows in order to distinguish between forward and backward messages propagating in or against the direction of an edge z_j as $\overrightarrow{\mu}(z_j)$ and $\overleftarrow{\mu}(z_j)$, respectively. The marginal distribution of variable z_j can be computed from the colliding messages as $p(z_j) \propto \overrightarrow{\mu}(z_j)\overleftarrow{\mu}(z_j)$.

From a message passing perspective, computing the result of (5) encompasses computing the messages and products $\overrightarrow{\mu}(\boldsymbol{s}_t)\overleftarrow{\mu}(\boldsymbol{s}_t)$ and $\overrightarrow{\mu}(\boldsymbol{n}_t)\overleftarrow{\mu}(\boldsymbol{n}_t)$. Automating the message passing procedure requires deriving the message passing computation rules of (6) for common factor-message pairs. For almost all elementary factors in the probabilistic model of (1)-(3), as shown in Figure 1, the computation rules have been derived in Loeliger et al. (2007). The next subsection will elaborate on the missing message computation rules through the non-linear state transition.

A naive message passing implementation yields sub-optimal computational efficiency. Some messages represent multivariate Normal distributions and some nodes require particular parameterizations of these messages. As an example, the addition node accepts and outputs messages with a mean-covariance parameterization, whereas the equality node accepts and outputs Normal messages in the canonical form. Converting between the different parameterization is expensive as it requires matrix inversions. For improved computational efficiency only messages with a mean-covariance parameterization are passed along the graph. To allow for this, some pairs of factor nodes are combined in compound nodes as illustrated in Figure 1, whose more efficient computation rules are specified in Loeliger et al. (2007, Table 4). The message passing schedule for efficient inference is shown in Figure 1. The messages ⑨ and ⑪ yield the solution to (5).

3.3 Approximate message passing

Probabilistic inference in the model of (1)-(3) through (5) requires propagating messages with a Normal distribution through non-linear autoregressive state transition nodes. Specifically, the computation of the messages ① and ④ in Figure 1 involves computing

$$\overrightarrow{\mu}(\boldsymbol{z}) \propto \int \delta(\boldsymbol{z} - g(\boldsymbol{x}))\overrightarrow{\mu}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}, \tag{7}$$

where $\overrightarrow{\mu}(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_{\boldsymbol{x}}, \Sigma_{\boldsymbol{x}})$ represents the messages ⑨ and ⑪ from the previous time step. The outgoing message $\overrightarrow{\mu}(\boldsymbol{z})$ often does not belong to the exponential family of distributions for non-linear functions g, leading to intractable inference in consecutive parts of the model. To retain tractability, the integration of (7) needs to be approximated. Following are three different approaches for approximating this integration: 1) linearization (Särkkä, 2013, Ch.5), 2) the unscented transform (Julier and Uhlmann, 1997; Wan and Van Der Merwe, 2000) and 3) the scaled spherical simplex unscented transform (Julier, 2002, 2003).

3.3.1 Linearization

If appropriate, the function g can be linearized using a first-order Taylor series expansion around the mean of the incoming distribution according to Särkkä (2013, Ch.5) as

$$g(\boldsymbol{x}) \approx g\left(\mathrm{E}[\boldsymbol{x}]\right) + J_g\left(\mathrm{E}[\boldsymbol{x}]\right)\left(\boldsymbol{x} - \mathrm{E}[\boldsymbol{x}]\right), \tag{8}$$

where $\mathrm{E}[\cdot]$ is the expected value operator and $J_g(\cdot)$ denotes the Jacobian matrix of g. Based on this linearization procedure the outgoing message $\vec{\mu}(\boldsymbol{z})$ in (7) can be determined as

$$\vec{\mu}(\boldsymbol{z}) = \mathcal{N}\left(\boldsymbol{z} \mid g(\boldsymbol{\mu_x}), J_g(\boldsymbol{\mu_x})\Sigma_{\boldsymbol{x}} J_g(\boldsymbol{\mu_x})^\top\right). \tag{9}$$

3.3.2 Unscented transform

Downsides of the linearization approach are that it requires access to the Jacobian matrix of the non-linear function g and that the linearization assumption might not hold for highly non-linear functions g. In Julier and Uhlmann (1997); Wan and Van Der Merwe (2000) the unscented transform is proposed based on the observation that directly approximating the result of (7) is easier than approximating the non-linear function. The unscented transform is a deterministic sampling procedure with a pre-specified set of $2M+1$ weighted samples, called sigma points, which capture the first- and second-order moment of the input distribution. These sigma points are propagated through the non-linear function g and (7) is approximated by a Normal distribution, based on the weighted sample mean and covariance of the transformed sigma points. Table 1 gives an overview of the unscented transform.

3.3.3 Scaled spherical simplex unscented transform

Although the computational complexity of the unscented transform equals the computational complexity of the linearization approach (Wan and Van Der Merwe, 2000), it scales linearly with the number of sigma points. Therefore it is advantageous to use as little sigma points as possible whilst still capturing the first- and second-order moment of the input distribution. In Julier (2003) the spherical simplex unscented transform is proposed which only uses $M+2$ sigma points, located on a hypersphere with radius $\sqrt{M}$. The spread of the sigma points in the original spherical simplex unscented transform potentially also captures non-local effect of the non-linear function. In order to reduce this spread, the sigma points can be rescaled according to the scaled unscented transform (Julier, 2002). Table 1 gives an overview of the scaled spherical simplex unscented transform.

4. Experiments

All experiments[1] have been performed using the state-of-the-art probabilistic programming package `ReactiveMP.jl`[2] (Bagaev and de Vries, 2021) in `Julia` (Bezanson et al., 2017).

4.1 Verification experiments

For verification of the proposed approach, three signals were generated: a square wave, sawtooth and chirp signal. These signals overlap each other in the frequency domain, limiting

[1] All experiments are available at `https://github.com/biaslab/PGM2022-SourceSeparationNAR`.

[2] `ReactiveMP.jl` is publicly available at `https://github.com/biaslab/ReactiveMP.jl`.

Table 1: (left) An overview of the unscented transform (Julier and Uhlmann, 1997; Wan and Van Der Merwe, 2000). $(\sqrt{\cdot})_i$ denotes the i^{th} row of the matrix square root. The weights are denoted by W_i. α, β and κ denote the hyperparameters as in (Wan and Van Der Merwe, 2000). (right) An overview of the scaled spherical simplex unscented transform (Julier, 2002, 2003). The $M+2$ sigma points $\mathcal{X}_i$ are iteratively generated, based on the weights W_i. Their scaled counterparts $\mathcal{X}'_i$ and W'_i are constructed using the scaling parameter c.

Unscented transform	**Scaled spherical simplex unscented transform**
Incoming message $\vec{\mu}(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_{\boldsymbol{x}}, \Sigma_{\boldsymbol{x}})$	
Initialization $\lambda = \alpha^2(M+\kappa) - M$ $\mathcal{X}_0 = \boldsymbol{\mu}_{\boldsymbol{x}}$ $\mathcal{X}_i = \begin{cases} \boldsymbol{\mu}_{\boldsymbol{x}} + \left(\sqrt{(M+\lambda)\Sigma_{\boldsymbol{x}}}\right)_i & i = 1,\dots,M \\ \boldsymbol{\mu}_{\boldsymbol{x}} - \left(\sqrt{(M+\lambda)\Sigma_{\boldsymbol{x}}}\right)_i & i = M+1,\dots,2M \end{cases}$ $W_0^{(m)} = \lambda/(M+\lambda)$ $W_0^{(c)} = \lambda/(M+\lambda) + (1-\alpha^2+\beta)$ $W_i^{(m)} = W_i^{(c)} = 1/(2M+2\lambda) \quad i = 1,\dots,2M$	Initialization $W_0 \in [0,1]$ $W_i = (1-W_0)/(M+1) \quad i = 1,\dots,M+1$ $\mathcal{X}_0^0 = 0,\ \mathcal{X}_1^1 = -1/\sqrt{2W_1},\ \mathcal{X}_2^1 = 1/\sqrt{2W_1}$ for $j = 2,\dots,M$: $\mathcal{X}_i^j = \begin{cases} \begin{bmatrix} \mathcal{X}_0^{j-1} \\ 0 \end{bmatrix} & \text{for } i = 0 \\ \begin{bmatrix} \mathcal{X}_i^{j-1} \\ \frac{-1}{\sqrt{j(j+1)W_1}} \end{bmatrix} & \text{for } i = 1,\dots,j \\ \begin{bmatrix} \mathbf{0}_{j-1} \\ \frac{j}{\sqrt{j(j+1)W_1}} \end{bmatrix} & \text{for } i = j+1 \end{cases}$
	Scaling $W'_i = \begin{cases} W_i/c^2 + 1 - 1/c^2 & i = 0 \\ W_i/c^2 & i = 1,\dots,M+1 \end{cases}$ $\mathcal{X}'_i = \boldsymbol{\mu}_{\boldsymbol{x}} + c\sqrt{\Sigma_{\boldsymbol{x}}}\mathcal{X}_i^M$
Transformation $\mathcal{Z}_i = g(\mathcal{X}_i) \quad i = 0,\dots,2M$	Transformation $\mathcal{Z}_i = g(\mathcal{X}'_i) \quad i = 0,\dots,M+1$
Outgoing message approximation $\vec{\mu}(\boldsymbol{z}) \approx \mathcal{N}(\boldsymbol{z} \mid \boldsymbol{\mu}_{\boldsymbol{z}}, \Sigma_{\boldsymbol{z}})$ $\boldsymbol{\mu}_{\boldsymbol{z}} = \sum_{i=0}^{2M} W_i^{(m)} \mathcal{Z}_i$ $\Sigma_{\boldsymbol{z}} = \sum_{i=0}^{2M} W_i^{(c)} (\mathcal{Z}_i - \mathrm{E}[\boldsymbol{z}])(\mathcal{Z}_i - \mathrm{E}[\boldsymbol{z}])^\top$	Outgoing message approximation $\vec{\mu}(\boldsymbol{z}) \approx \mathcal{N}(\boldsymbol{z} \mid \boldsymbol{\mu}_{\boldsymbol{z}}, \Sigma_{\boldsymbol{z}})$ $\boldsymbol{\mu}_{\boldsymbol{z}} = \sum_{i=0}^{M+1} W'_i \mathcal{Z}_i$ $\Sigma_{\boldsymbol{z}} = \sum_{i=0}^{M+1} W'_i (\mathcal{Z}_i - \mathrm{E}[\boldsymbol{z}])(\mathcal{Z}_i - \mathrm{E}[\boldsymbol{z}])^\top$ $+ (1-c^2)(\mathcal{Z}_0 - \mathrm{E}[\boldsymbol{z}])(\mathcal{Z}_0 - \mathrm{E}[\boldsymbol{z}])^\top$

the use of conventional filtering solutions. 2000 instances of each signal were generated with a length of 1000 samples for varying amplitudes and phase delays. The first 1000 instances of the signals were used for training the corresponding non-linear autoregressive models of order $M_s = M_n = 16$ with the backpropagation algorithm using the Adam optimizer (Kingma and Ba, 2014) for a mean squared error loss. The latter 1000 were combined to form the observed signals under the mixing model of (2).

With the trained non-linear autoregressive models and the generated observed signals the solution to (5) was computed using message passing-based inference for the various approximation methods from Section 3.3. The unscented transform uses the default hyperparameters $\alpha = 1e-3$, $\beta = 2$ and $\kappa = 0$ as introduced in Wan and Van Der Merwe (2000). The hyperparameters of the spherical simplex unscented transform are set to $c = 1$ and $W_0 = 0.1$. The latent state priors $p(\boldsymbol{s}_0)$ and $p(\boldsymbol{n}_0)$ were initialized to be uninformative with a random mean vector and a diagonal covariance matrix with relatively large entries. The observation noise variance was set to $\sigma_y^2 = 10^{-10}$ and the process noise variance variables σ_s^2 and σ_n^2 were set to the average mean squared error loss during training.

The performance of the different approximation methods from Section 3.3 was evaluated based on the inferred signal of interest $\boldsymbol{s}$ and the actual underlying signal $\hat{\boldsymbol{s}}$. For assessing the performance we calculated the mean squared error (MSE), the average log-likelihood (ALL) and the signal-to-noise ratio improvement (ΔSNR), defined as

$$\mathrm{MSE} = \frac{1}{T}\sum_{t=1}^{T} \|\mathrm{E}[s_t] - \hat{s}_t\|^2, \tag{10a}$$

$$\mathrm{ALL} = \frac{1}{T}\sum_{t=1}^{T} \ln p(s_t = \hat{s}_t \mid \boldsymbol{y}_{1:t}), \tag{10b}$$

$$\Delta\mathrm{SNR} = 10\log_{10}\frac{\sum_{t=1}^{T}\|y_t - \hat{s}_t\|^2}{\sum_{t=1}^{T}\|\mathrm{E}[s_t] - \hat{s}_t\|^2}. \qquad \text{[dB]} \tag{10c}$$

Table 2 reports the performance of the different approximation methods, averaged over all 1000 generated observed signals. Figure 3 shows fragments of the true underlying and inferred signal components of the inference result with the most median performance in terms of MSE for the linearization approximation method.

As shown in Figure 3 the proposed source separation approach is capable of accurately separating the constituent sources. From Table 2 we observe that the linearization approach yields the best MSE for all different signals. In terms of ALL and ΔSNR there is no definitive optimal approximation method.

4.2 Validation experiments

To validate the proposed source separation framework, we apply this methodology to a speech denoising task. The experimental setup equals the one of Section 4.1. The speech model g_s is trained on 100 speech recordings of the LibriSpeech ASR corpus (Panayotov et al., 2015), lasting 23 minutes in total. This dataset contains in total 1000 hours of 16 kHz read English speech. The noise model g_n is trained on 10 air conditioner noise recordings of the Microsoft scalable noisy speech (MS-SNSD) Dataset (Reddy et al., 2019), lasting 9

Table 2: Performance evaluation of the approximation methods in Section 3.3 based on the metrics in (10). The mean performance is evaluated with respect to the signal of interest, which corresponds to first waveform in the combination. The signal-to-noise ratio is expressed in decibels. The best performing approximation method per metric and signal combination is depicted in bold.

Approximation method	Linearization			Unscented transform			Spherical simplex		
Metric	MSE	ALL	ΔSNR	MSE	ALL	ΔSNR	MSE	ALL	ΔSNR
chirp - block	**1.67e-1**	-91.7	17.0	8.06e-1	**-72.3**	14.7	1.74e-1	-94.1	**17.4**
chirp - sawtooth	**4.35e-1**	**-1.13**	**5.93**	1.60e0	-2.38	4.09	4.39e-1	-2.38	5.81
sawtooth - block	**4.45e-2**	-7.24	**20.4**	1.88e0	-7.01	14.1	5.12e-2	**-4.18**	19.2

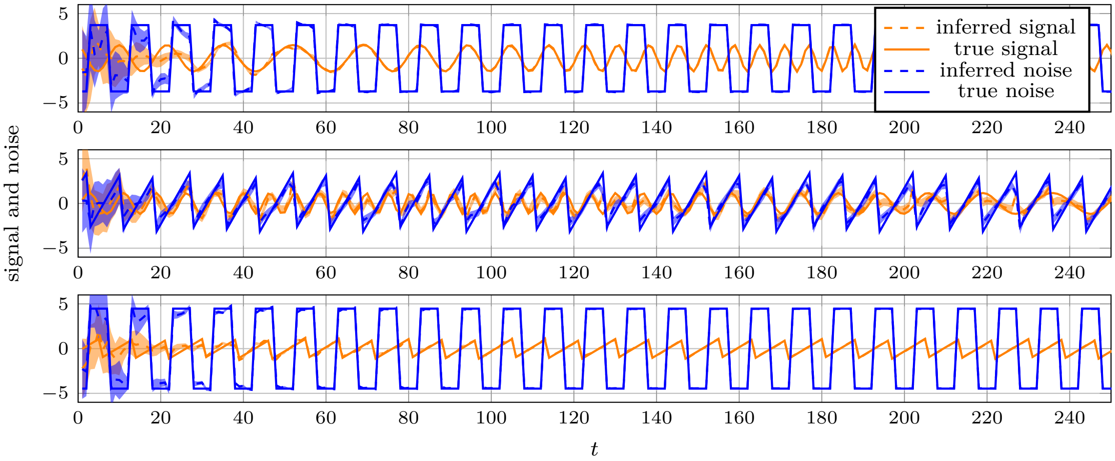


Figure 3: Fragments of the true underlying and inferred signal components for the different mixing combinations in the verification experiments of Section 4.1. The inference results achieving median performance in terms of mean squared error for the linearization approximation method have been selected here. The transparent areas denote the confidence intervals of ±σ from the mean.

minutes in total. This dataset contains a collection of 16 kHz clean speech files and a variety of environmental noise files. Mixture signals are generated from different speech recordings and alternate air conditioner noise recordings.

Probabilistic inference was performed using the scaled spherical simplex unscented transform of Section 3.3, as this yielded the highest ΔSNR on average. The mixture, inferred speech and underlying speech signals of the fragment with median ΔSNR performance is shown in Figure 4. The input SNR was -0.16 dB and the obtained SNR improvement on this fragment was 4.59 dB.

5. Discussion

The proposed source separation model of Section 2 in Figure 1 assumes an additive linear observation model. However, for some applications the signal can be first warped using some non-linear transform to a desired domain that might be better suited for the modeling of the signal. This consequently requires a non-linear mixing model for performing the source separation. Examples of non-linear mixing models for audio processing are given in Frey

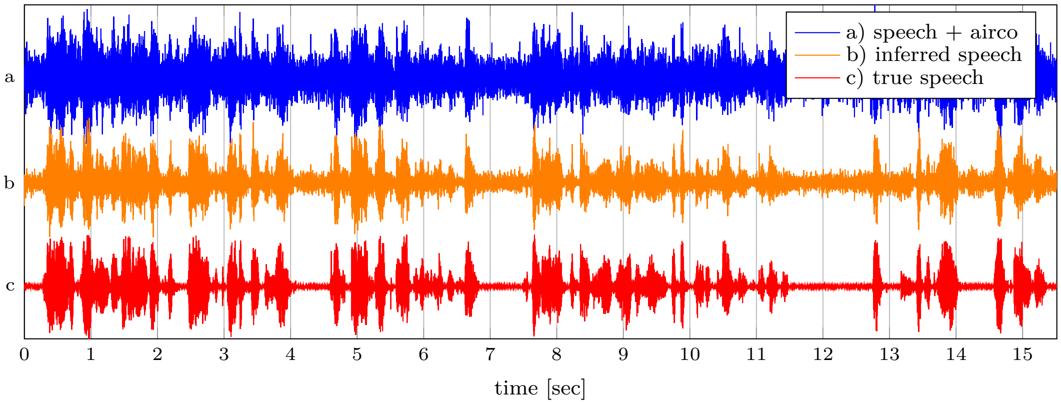


Figure 4: Experimental results of Section 4.2 with median performance in terms of ΔSNR. With an input SNR of -0.16 dB an SNR gain of 4.59 dB was obtained.

et al. (2001); van Erp et al. (2021). However, care should be taken with these non-linear mixing models as the approximations that are required for tractable inference (Hershey et al., 2010; Radfar et al., 2006) might limit source separation performance and efficiency.

The authors have purposefully chosen for the simple architecture of the multilayer perceptron in Figure 2 throughout the experiments, to highlight the source separation methodology, rather than the underlying neural network. Aware of the immense variety of alternative network architectures, we deem the exploration of this model space future research in the scope of the proposed source separation framework.

The current source separation approach focuses on online filtering, where only information from the past is available for predicting current state estimates. Depending on the application, a (fixed-lag) smoothing operation might be desired for improved performance. For common non-linear state transitions g only a forward message is defined, preventing efficient smoothing operations. However, computing the backward messages can be achieved by enforcing g to be invertible as in van Erp and de Vries (2022), based on previous work on normalizing flows (Rezende and Mohamed, 2016; Dinh et al., 2015).

6. Conclusion

This paper has introduced an explainable and modular probabilistic model architecture for mixtures of observations. These observations are formed by their constituent sources, which are independently modelled by non-linear autogressive models. The tracking of these sources through probabilistic inference, as executed through efficient message passing, yields a powerful filtering-based source separation algorithm that is suitable for real-time applications. Different approximate inference solutions are compared through a set of verification experiments and the framework is effectively employed in a speech denoising task.

References

D. Bagaev and B. de Vries. Reactive Message Passing for Scalable Bayesian Inference. *arXiv:2112.13251 [cs]*, 2021.

M. J. Beal. *Variational Algorithms for Approximate Bayesian Inference.* PhD thesis, University College London, 2003.

J. Bezanson, A. Edelman, et al. Julia: A Fresh Approach to Numerical Computing. *SIAM Review*, 59, 2017.

P. Comon. Independent Component Analysis, a new concept? *Signal Processing*, 36, 1994.

M. Cox, T. van de Laar, et al. A factor graph approach to automated design of Bayesian signal processing algorithms. *International Journal of Approximate Reasoning*, 104, 2019.

J. Dauwels, S. Korl, et al. Expectation maximization as message passing. In *Proceedings. International Symposium on Information Theory, 2005. ISIT 2005.* IEEE, 2005.

L. Dinh, D. Krueger, et al. NICE: Non-linear Independent Components Estimation. *arXiv:1410.8516 [cs]*, 2015.

R. Dutt, S. Mondal, et al. Single Channel Blind Source Separation Using Dual Extended Kalman Filter. In *IEEE International Symposium on Circuits and Systems*, 2021.

A. T. Erdogan. Adaptive algorithm for the blind separation of sources with finite support. In *16th European Signal Processing Conference*, 2008.

C. Fevotte and S. J. Godsill. A Bayesian Approach for Blind Separation of Sparse Sources. *IEEE Transactions on Audio, Speech, and Language Processing*, 14, 2006.

G. Forney. Codes on graphs: normal realizations. *IEEE Transactions on Information Theory*, 47, 2001.

B. J. Frey, L. Deng, et al. ALGONQUIN: Iterating Laplace's Method to Remove Multiple Types of Acoustic Distortion for Robust Speech Recognition. In *Proc. of the Eurospeech Conference*, 2001.

J. R. Hershey, P. Olsen, et al. Signal Interaction and the Devil Function. In *Proceedings of the Interspeech 2010*, 2010.

L. Hong, J. Rosca, et al. Bayesian single channel speech enhancement exploiting sparseness in the ICA domain. In *12th European Signal Processing Conference*, 2004.

S. Julier. The scaled unscented transformation. In *Proceedings of the 2002 American Control Conference (IEEE Cat. No.CH37301)*, volume 6, 2002.

S. Julier. The spherical simplex unscented transformation. In *Proceedings of the 2003 American Control Conference, 2003.*, volume 3, 2003.

S. J. Julier and J. K. Uhlmann. New extension of the Kalman filter to nonlinear systems. In *Signal Processing, Sensor Fusion, and Target Recognition VI*, volume 3068. International Society for Optics and Photonics, 1997.

D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*, 2014.

K. H. Knuth. Informed Source Separation: A Bayesian Tutorial. *arXiv:1311.3001 [cs, stat]*, 2013.

F. Kschischang, B. Frey, et al. Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory*, 47, 2001.

H.-A. Loeliger. An introduction to factor graphs. *IEEE Signal Processing Magazine*, 21, 2004.

H.-A. Loeliger, J. Dauwels, et al. The Factor Graph Approach to Model-Based Signal Processing. *Proceedings of the IEEE*, 95, 2007.

P. Magron and T. Virtanen. Complex ISNMF: a Phase-Aware Model for Monaural Audio Source Separation. *arXiv:1802.03156 [cs, eess]*, 2018.

T. P. Minka. Expectation Propagation for Approximate Bayesian Inference. In *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers Inc., 2001.

V. Panayotov et al. Librispeech: An ASR corpus based on public domain audio books. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015.

M. Radfar, A. Banihashemi, et al. Nonlinear minimum mean square error estimator for mixture-maximisation approximation. *Electronics Letters*, 42, 2006.

C. K. A. Reddy, E. Beyrami, et al. A scalable noisy speech dataset and online subjective test framework. *arXiv:1909.08050 [cs, eess]*, 2019.

S. Rennie, J. Hershey, et al. Single-Channel Multitalker Speech Recognition. *IEEE Signal Processing Magazine*, 27, 2010.

D. J. Rezende and S. Mohamed. Variational Inference with Normalizing Flows. *arXiv:1505.05770 [cs, stat]*, 2016.

S. Särkkä. *Bayesian Filtering and Smoothing.* Cambridge University Press, 2013.

B. van Erp and B. de Vries. Hybrid Inference with Invertible Neural Networks in Factor Graphs. In *2022 30th European Signal Processing Conference*, 2022. in press.

B. van Erp, A. Podusenko, et al. A Bayesian Modeling Approach to Situated Design of Personalized Soundscaping Algorithms. *Applied Sciences*, 11, 2021.

E. Wan and A. Nelson. Neural dual extended Kalman filtering: applications in speech enhancement and monaural blind signal separation. In *Neural Networks for Signal Processing VII. Proceedings of the 1997 IEEE Signal Processing Society Workshop*, 1997.

E. Wan and R. Van Der Merwe. The unscented Kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*. IEEE, 2000.