Rao-Blackwellized Particle Filtering for Sequential Speech Enhancement

Sunho Park and Seungjin Choi

***Abstract*— In this paper we present a method of sequential speech enhancement, where we infer clean speech signal using a Rao-Blackwellized particle filter (RBPF), given a noise-contaminated observed signal. In contrast to Kalman filtering-based methods, we consider a non-Gaussian speech generative model that is based on the generalized auto-regressive (GAR) model. Model parameters are learned by sequential expectation maximization, incorporating the RBPF. Empirical comparison to Kalman filter, confirms the high performance of the proposed method.**

I. Introduction

Speech enhancement is a fundamental problem, which aims at estimating clean speech, given noise-contaminated signals. Various speech enhancement methods have been developed. The spectral subtraction method [8] is a widely-used speech enhancement method, but suffers from audible distortion called "musical noise". The H_∞ filter-based method involves the infinite-norm minimization, where the prior knowledge of noise distribution is not required. Thus, it works in a robust manner for arbitrary noise [11], however, it does not operate sequentially. The Kalman filter is widely used for speech enhancement [10], since it can be easily implemented and gives the optimal solution in the mean-squared sense. However, Kalman filter assumes a Gaussian distribution, hence it has a limitation for modelling speech which follows a non-Gaussian distribution. Recently particle filters were applied to the problem of speech enhancement [13], where the time-varying auto-regressive model (Gaussian model) for the clean speech was used and associated parameters were sequentially updated by an approximated Bayesian method.

In this paper, we consider the generalized auto-regressive (GAR) model for clean speech, in order to accommodate the non-Gaussian characteristics of speech. With the GAR model, we formulate the speech enhancement problem as a Rao-Blackwellized particle filtering. Associated model parameters are learned by a sequential expectation maximization (SEM) method. Empirical comparison to the Kalman filter, confirms that the proposed method based on the Rao-Blackwellized particle fitler, is superior to Kalman filter, in the task of sequential speech enhancement.

II. Generalized Auto-Regressive Model

The auto-regressive (AR) model is a widely-used linear modelling method, where the current value of a time series, s_t, is expressed as a linear sum of its past values, $\{s_{t-d}\}$, and an innovation v_t:

$$s_t = \sum_{d=1}^{p} \alpha_d s_{t-d} + v_t. \tag{1}$$

The AR modelling involves determining coefficients $\{\alpha_d\}$ that provide a linear optimal fitting to given time series $\{s_t\}$, assuming that the innovation v_t is Gaussian. The AR model captures the dependence of the current value of a time series on its past values, through a linear model. The innovation contains a truly new information that is not found in past values of time series.

The generalized auto-regressive (GAR) model is a non-Gaussian extension of the AR model, which adopts the same linear model (1) but assumes the innovation v_t is drawn from the generalized exponential (GE) distribution (a.k.a. generalized Gaussian) with mean zero [2] that is of the form

$$p(v; R, \beta) = \frac{R\beta^{1/R}}{2\Gamma(1/R)} \exp\left\{-\beta \left|v\right|^R\right\}, \tag{2}$$

where $1/\beta$ determine width of the density and R is a parameter which determines a shape of distribution.

The GE distribution accommodates a wide class of unimodal probability distribution. For example, $p(v; R, \beta)$ becomes Gaussian distribution for $R = 2$ and Laplacian distribution for $R = 1$. The GAR model reflects the non-Gaussian characteristics of speech signals. However, in such a model, the probabilistic inference is intractable, in contrast to Kalman filters. This leads us to consider particle filters which are described in section IV.

III. State-Space Models

The noise-contaminated observed signal y_t is modelled as a linear sum of clean speech s_t and noise n_t:

$$y_t = s_t + n_t, \tag{3}$$

where the clean speech and noise follow GAR and AR models, respectively, i.e.,

$$s_t = \sum_{d=1}^{p} \alpha_d s_{t-d} + v_t, \tag{4}$$

$$n_t = \sum_{d=1}^{q} \gamma_d n_{t-d} + u_t, \tag{5}$$

where v_t obeys the generalized exponential distribution and u_t is drawn from Gaussian distribution. We assume that s_t and n_t are statistically independent.

Sunho Park is with the Department of Computer Science, Pohang University of Science and Technology, Korea (email: titan@postech.ac.kr).

Seungjin Choi is with the Department of Computer Science, Pohang University of Science and Technology, Korea (email: seungjin@postech.ac.kr).

We define $\boldsymbol{s}_t \in \mathbb{R}^p$ and $\boldsymbol{n}_t \in \mathbb{R}^q$ as

$$\boldsymbol{s}_t = [s_t, s_{t-1}, \ldots, s_{t-p+1}]^\top,$$
$$\boldsymbol{n}_t = [n_t, n_{t-1}, \ldots, n_{t-q+1}]^\top.$$

Concatenating these two vectors, we define a state vector $\boldsymbol{x}_t = [\boldsymbol{s}_t^\top, \boldsymbol{n}_t^\top]^\top \in \mathbb{R}^{p+q}$. Accommodating generative models (4) and (5) for speech and noise, the state-space model that we consider, is of the form:

$$\boldsymbol{x}_t = \boldsymbol{A}\boldsymbol{x}_{t-1} + \boldsymbol{B}\boldsymbol{r}_t, \quad (6)$$
$$y_t = \boldsymbol{b}^\top \boldsymbol{x}_t, \quad (7)$$

where

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}_s & 0 \\ 0 & \boldsymbol{A}_n \end{bmatrix},$$
$$\boldsymbol{B} = \begin{bmatrix} \boldsymbol{b}_s & 0 \\ 0 & \boldsymbol{b}_n \end{bmatrix},$$
$$\boldsymbol{r}_t = [v_t, u_t]^\top,$$
$$\boldsymbol{b}^\top = [\boldsymbol{b}_s^\top, \boldsymbol{b}_n^\top],$$

and

$$\boldsymbol{b}_s = [1, 0, \ldots, 0]^\top \in \mathbb{R}^p,$$
$$\boldsymbol{b}_n = [1, 0, \ldots, 0]^\top \in \mathbb{R}^q.$$

The state transition matrix $\boldsymbol{A} \in \mathbb{R}^{(p+q)\times(p+q)}$ is a block diagonal matrix where $\boldsymbol{A}_s$ is given by

$$\boldsymbol{A}_s = \begin{bmatrix} \alpha_1 & \alpha_2 & \cdots & \cdots & \alpha_p \\ 1 & 0 & \cdots & \cdots & 0 \\ 0 & 1 & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & 0 \end{bmatrix},$$

and $\boldsymbol{A}_n$ is constructed in a similar way.

IV. Sequential Monte Carlo Method

A. Generic Particle Filter

Particle filter uses a set of particles to solve nonlinear and non-Gaussian probabilistic inference problems, approximating the true posterior distribution of a hidden state by a discrete distribution determined by the evaluation of importance weight at each particle. Estimating posterior density sequentially requires two-step procedure. First, we generate new particles from a proposal density $\pi(\boldsymbol{x}_t|\boldsymbol{x}_{0:t-1}, y_{1:t})$,

$$\boldsymbol{x}_t^{(i)} \sim \pi(\boldsymbol{x}_t|\boldsymbol{x}_{0:t-1}^{(i)}, y_{1:t}), \quad (8)$$

where $\boldsymbol{x}_{0:t}^{(i)} \triangleq \{\boldsymbol{x}_0^{(i)}, \boldsymbol{x}_1^{(i)}, \ldots, \boldsymbol{x}_t^{(i)}\}$ and $y_{1:t} = \{y_1, \ldots, y_t\}$. Second, we update the weight $w_t^{(i)}$ of each particle:

$$w_t^{(i)} \propto w_{t-1}^{(i)} \frac{p(y_t|\boldsymbol{x}_t^{(i)})p(\boldsymbol{x}_t^{(i)}|\boldsymbol{x}_{t-1}^{(i)})}{\pi(\boldsymbol{x}_t^{(i)}|\boldsymbol{x}_{0:t-1}^{(i)}, y_{1:t})}. \quad (9)$$

These sequential sampling (the generation of new particles) and weight-updating are the main part of particle filter.

In our inference problem, the likelihood $p(y_t|\boldsymbol{x}_t)$ is not well defined. Moreover, the posterior density of the noise has to be approximated in the framework of the generic particle filter, although it can be exactly calculated due to the Gaussian assumption. In order to overcome these two limitations, we re-formulate the inference problem using a Rao-Blackwellised particle fitler (RBPF).

B. Rao-Blackwellised Particle Filter

The posterior density of the hidden state can be decomposed as

$$p(\boldsymbol{x}_{0:t}|y_{1:t}) = p(\boldsymbol{n}_{0:t}|\boldsymbol{s}_{0:t}, y_{1:t})p(\boldsymbol{s}_{0:t}|y_{1:t}), \quad (10)$$

which leads us to estimate the speech and noise separately. Taking the GE generative model (4) into account, the distribution $p(\boldsymbol{s}_{0:t}|y_{1:t})$ does not admit a closed-form expression. On the other hand, the Gaussian generative model (5) enables us to determine $p(\boldsymbol{n}_{0:t}|\boldsymbol{s}_{0:t}, y_{1:t})$ in a tractable manner, which admits a closed-form expression.

The decomposition (10) leads us to develop more efficient inference algorithm, compared to the generic particle filter. The posterior density of the noise, $p(\boldsymbol{n}_{0:t}|\boldsymbol{s}_{0:t}, y_{1:t})$, can be analytically computed using the Kalman filter, if we know the marginal posterior density $p(\boldsymbol{s}_{0:t}|y_{1:t})$. Only the posterior density of speech, $p(\boldsymbol{s}_{0:t}|y_{1:t})$, is approximately calculated through a sampling method. This method, motivated by the decomposition (10), is known as RBPF.

The estimation of hidden states associated with speech and noise using a RBPF, is as follows. First, the marginal posterior density of speech is approximated by importance weights of particles [4],

$$\widehat{p}(\boldsymbol{s}_{0:t}|y_{1:t}) = \sum_{i=1}^{N} \tilde{w}_t^{(i)} \delta_{\boldsymbol{s}_{0:t}^{(i)}}(\boldsymbol{s}_{0:t}),$$

where, $\{\tilde{w}_t^{(i)}\}_{1,\ldots,N}$ is normalized importance weights and N is the number of the particles. Next, the marginal posterior density of noise [3] is given by

$$\widehat{p}(\boldsymbol{n}_{0:t}|y_{1:t}) = \sum_{i=1}^{N} \tilde{w}_t^{(i)} p(\boldsymbol{n}_{0:t}|\boldsymbol{s}_{0:t}, y_{1:t}). \quad (11)$$

The marginal posterior density of noise (11) is a mixture of Gaussians where mixing parameters correspond to the importance weights of particles. Thus, we attach Kalman filter to each particle in the RBPF for the inference of noise. Next section illustrates the details on the inference.

V. Inference

A. Inference for the state of noise

Let $\boldsymbol{s}_t^{(i)}$ $(i = 1, 2, \cdots, N)$ be particles of clean speech and σ^2 be the variance of u_t in the noise AR model. We sample $\boldsymbol{s}_t^{(i)}$ by the method described in Sec. V-B and then propagate the mean $\boldsymbol{\mu}_t^{(i)}$ and covariance $\boldsymbol{\Sigma}_t^{(i)}$ of $\boldsymbol{n}_t$ with a Kalman filter

as follows:

$$\begin{aligned}
\boldsymbol{\mu}_{t|t-1}^{(i)} &= \boldsymbol{A}_n \boldsymbol{\mu}_{t-1|t-1}^{(i)}, \\
\boldsymbol{\Sigma}_{t|t-1}^{(i)} &= \boldsymbol{A}_n \boldsymbol{\Sigma}_{t-1|t-1}^{(i)} \boldsymbol{A}_n^\top + \sigma^2 \boldsymbol{b}_n \boldsymbol{b}_n^\top, \\
\boldsymbol{\Gamma}_t^{(i)} &= \boldsymbol{b}_n^\top \boldsymbol{\Sigma}_{t|t-1}^{(i)} \boldsymbol{b}_n, \\
y_{t|t-1}^{(i)} &= \boldsymbol{b}_n^\top \boldsymbol{\mu}_{t|t-1}^{(i)} + \boldsymbol{b}_s^\top \boldsymbol{s}_t^{(i)}, \\
\boldsymbol{\mu}_{t|t}^{(i)} &= \boldsymbol{\mu}_{t|t-1}^{(i)} - \boldsymbol{\Sigma}_{t|t-1}^{(i)} \boldsymbol{b}_n \left[\boldsymbol{\Gamma}_t^{(i)}\right]^{-1} (y_t - y_{t|t-1}^{(i)}), \\
\boldsymbol{\Sigma}_{t|t}^{(i)} &= \boldsymbol{\Sigma}_{t|t-1}^{(i)} - \boldsymbol{\Sigma}_{t|t-1}^{(i)} \boldsymbol{b}_n \left[\boldsymbol{\Gamma}_t^{(i)}\right]^{-1} \boldsymbol{b}_n^\top \boldsymbol{\Sigma}_{t|t-1}^{(i)},
\end{aligned}$$

where

$$\begin{aligned}
\boldsymbol{\mu}_{t|t-1} &\triangleq \mathbb{E}\{\boldsymbol{n}_t | y_{1:t-1}\}, \\
\boldsymbol{\mu}_{t|t} &\triangleq \mathbb{E}\{\boldsymbol{n}_t | y_{1:t}\}, \\
y_{t|t-1} &\triangleq \mathbb{E}\{y_t | y_{1:t-1}\}, \\
\boldsymbol{\Sigma}_{t|t-1} &\triangleq \mathrm{cov}(\boldsymbol{n}_t | y_{1:t-1}), \\
\boldsymbol{\Sigma}_{t|t} &\triangleq \mathrm{cov}(\boldsymbol{n}_t | y_{1:t}), \\
\boldsymbol{\Gamma}_t &\triangleq \mathrm{cov}(y_t | y_{1:t-1}).
\end{aligned}$$

The predictive density [3] is given by

$$p(y_t | y_{1:t-1}, \boldsymbol{s}_{0:t}) = \mathcal{N}(y_t; y_{t|t-1}, \boldsymbol{\Gamma}_t). \tag{12}$$

Finally the marginal filtering density of noise is estimated by a minimum mean square estimation (MMSE) method [7],

$$\widehat{p}(\boldsymbol{n}_t | y_{0:t}) = \mathcal{N}(\widehat{\boldsymbol{\mu}}_{t|t}, \widehat{\boldsymbol{\Sigma}}_{t|t}), \tag{13}$$

where

$$\begin{aligned}
\widehat{\boldsymbol{\mu}}_{t|t} &= \sum_{i=1}^{N} \tilde{w}_t^{(i)} \boldsymbol{\mu}_{t|t}^{(i)}, \\
\widehat{\boldsymbol{\Sigma}}_{t|t} &= \sum_{i=1}^{N} \tilde{w}_t^{(i)} \{\boldsymbol{\Sigma}_{t|t}^{(i)} + (\boldsymbol{\mu}_{t|t}^{(i)} - \widehat{\boldsymbol{\mu}}_{t|t})(\boldsymbol{\mu}_{t|t}^{(i)} - \widehat{\boldsymbol{\mu}}_{t|t})^\top\}.
\end{aligned}$$

B. Inference for the state of clean speech

In the RBPF, updating importance weights in (9) (that is slightly different from the one in [3]) is done by

$$w_t^{(i)} \propto w_{t-1}^{(i)} \frac{p(y_t | \boldsymbol{s}_{0:t}^{(i)}, y_{1:t-1}) p(\boldsymbol{s}_t^{(i)} | \boldsymbol{s}_{t-1}^{(i)})}{\pi(\boldsymbol{s}_t^{(i)} | \boldsymbol{s}_{0:t-1}^{(i)}, y_{1:t})}, \tag{14}$$

where $p(y_t | \boldsymbol{s}_{0:t}, y_{t-1})$ is the predictive density given in (12), $p(\boldsymbol{s}_t | \boldsymbol{s}_{t-1})$ is the prior distribution determined by the GAR model for clean speech, and $\pi(\boldsymbol{s}_t | \boldsymbol{s}_{0:t-1}, y_{1:t})$ is the proposal density. Depending on the choice of the proposal density, the updating rule for importance weights is different. We consider the following two proposal densities:

1) Prior Importance Distribution: If we use the prior distribution for the proposal density, i.e., $\pi(\boldsymbol{s}_t | \boldsymbol{s}_{0:t-1}, y_{1:t}) = p(\boldsymbol{s}_t | \boldsymbol{s}_{t-1})$, then importance weights are simplified as

$$w_t^{(i)} \propto w_{t-1}^{(i)} p(y_t | \boldsymbol{s}_{0:t}^{(i)}, y_{1:t-1}). \tag{15}$$

The distribution $p(y_t | \boldsymbol{s}_{0:t}^{(i)}, y_{1:t-1})$ in (15) is easily evaluated through one-step time-update in the Kalman filter on each particle. However, in such a case, we should generate new particles from GE density, which is a time-consuming task [14].

2) Optimal Importance Distribution: The better choice for the proposal density is the optimal importance distribution that minimizes the variance of the importance weights [4]. Using Bayes rule, the optimal importance distribution is expressed by

$$p(\boldsymbol{s}_t | \boldsymbol{s}_{0:t-1}, y_{1:t}) = \frac{p(y_t | \boldsymbol{s}_{0:t}, y_{1:t-1}) p(\boldsymbol{s}_t | \boldsymbol{s}_{t-1})}{p(y_t | \boldsymbol{s}_{0:t-1}, y_{1:t-1})}, \tag{16}$$

where $p(y_t | \boldsymbol{s}_{0:t-1}, y_{1:t-1})$ is the normalizing constant given by

$$\begin{aligned}
&p(y_t | \boldsymbol{s}_{0:t-1}, y_{1:t-1}) \\
&\quad = \int p(y_t | \boldsymbol{s}_{0:t}, y_{1:t-1}) p(\boldsymbol{s}_t | \boldsymbol{s}_{t-1}) d\boldsymbol{s}_t.
\end{aligned}$$

Note that the variable s_t is the only stochastic component, in $\boldsymbol{s}_t = [s_t, \ldots, s_{t-p+1}]^\top$. Thus, Eq. (16) is simplified as

$$p(s_t | \boldsymbol{s}_{t-1}, y_{1:t}) \propto p(y_t | \boldsymbol{s}_{0:t}, y_{1:t-1}) p(s_t | \boldsymbol{s}_{t-1}). \tag{17}$$

The distribution $p(s_t | \boldsymbol{s}_{t-1})$ in (17) follows the GE density, hence the sampling might be a time-consuming job. Here we consider the Gaussian-approximation of the GE density, i.e.,

$$\begin{aligned}
\widehat{p}(s_t | \boldsymbol{s}_{t-1}^{(i)}) &= \mathcal{N}(s_t; \widehat{s}_{t|t-1}^{(i)}, \sigma_s^2), \\
\widehat{s}_{t|t-1}^{(i)} &= \boldsymbol{\alpha}^\top \boldsymbol{s}_{t-1}^{(i)}, \\
\sigma_s^2 &= \frac{c}{2\beta},
\end{aligned} \tag{18}$$

where $\widehat{s}_{t|t-1}$ is the prediction determined by the GAR model. The positive constant c is a value determined by a user such that the shape of Gaussian is similar to that of GE density. In our experiment, we set $c = 0.85$ for $R = 1.25$.

Using the Gaussian-approximated optimal distribution, we generate new samples by

$$\begin{aligned}
s_t^{(i)} &\sim \widehat{p}(s_t | \boldsymbol{s}_{t-1}) p(y_t | \boldsymbol{s}_{0:t}, y_{1:t-1}), \\
&= \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0),
\end{aligned} \tag{19}$$

where

$$\begin{aligned}
\boldsymbol{\mu}_0 &= \widehat{s}_{t|t-1}^{(i)} - \boldsymbol{K}_t (y_t - y_{t|t-1}^{(i)}), \\
\boldsymbol{\Sigma}_0 &= \sigma_s^2 - \boldsymbol{K}_t \sigma_s^2, \\
\boldsymbol{K}_t &= \sigma_s^2 (\sigma_s^2 + \boldsymbol{\Gamma}_t)^{-1}.
\end{aligned}$$

Thus, it follows from (16) and (19), the updating rule for importance weights in (14), is simplified as

$$w_t^{(i)} \propto w_{t-1}^{(i)} \frac{p(s_t^{(i)} | \boldsymbol{s}_{t-1}^{(i)})}{\widehat{p}(s_t^{(i)} | \boldsymbol{s}_{t-1}^{(i)})} \widehat{p}(y_t | \boldsymbol{s}_{0:t-1}^{(i)}, y_{1:t-1}), \tag{20}$$

where

$$\begin{aligned}
&\widehat{p}(y_t | \boldsymbol{s}_{0:t-1}, y_{1:t-1}) \\
&\quad = \int p(y_t | \boldsymbol{s}_{0:t}, y_{1:t-1}) \widehat{p}(s_t | \boldsymbol{s}_{t-1}) ds_t.
\end{aligned}$$

The ratio of distributions, $\frac{p(s_t^{(i)} | \boldsymbol{s}_{t-1}^{(i)})}{\widehat{p}(s_t^{(i)} | \boldsymbol{s}_{t-1}^{(i)})}$ in (20), involves the difference between GE density and its associated Gaussian approximation. It can be viewed as a compensation term

which makes up for the Gaussian approximation-based sampling (19). After updating the importance weights, we normalize them to sum to one,

$$\widetilde{w}_t^{(i)} = \frac{w_t^{(i)}}{\sum_{i=1}^N w_t^{(i)}}.$$

In addition, the resampling scheme is also applied in order to avoid a degeneracy problem, so that only particles with high weights survive. In this way, we obtain a set of particles which approximates $p(\boldsymbol{s}_{0:t}|y_{1:t})$.

VI. Parameter Learning

So far, parameters involving the speech model as well as the noise model, are assumed to be known in the inference. However, these parameters are not available in advance. Parameters to be learned, are $\theta = \{\boldsymbol{\alpha}, \beta, \boldsymbol{\gamma}, \sigma^2\}$, where $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_p]^\top$ is the set of speech GAR model coefficients in (4), β is the parameter determining the width of the generalized exponential density in (2), $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_q]^\top$ is the set of noise AR model coefficients in (5), and σ_n^2 is the variance of e_t in the noise AR model (5). In this section, we present a sequential Newton expectation maximization (SNEM) algorithm which learns model parameters recursively using the approximated posterior distribution of hidden variables determined by the RBPF.

A. Sequential EM

The EM algorithm is an iterative method which finds local maxima of the log-likelihood function. E-step involves calculating the expected complete-data log-likelihood and M-step updates parameters that maximize the expected complete-data log-likelihood. In our model (6), the observation y_t is related to the hidden state $\boldsymbol{x}_t$ by a deterministic mapping. Thus, the parameterized log-distribution of hidden states can be treated as the complete-data log-likelihood. The quick summary of EM algorithm is as follows:

E-step

$$Q(\theta, \widehat{\theta}_k) = \mathbb{E}\{\log p(\boldsymbol{x}_{0:T}; \theta)|y_{1:T}, \widehat{\theta}_k\}, \qquad (21)$$

where T is a total number of observations and $p(\boldsymbol{x}; \theta)$ represents the probability density of $\boldsymbol{x}$ parameterized by θ.

M-step

$$\theta_{k+1} = \arg\max_{\theta} Q(\theta, \widehat{\theta}_k).$$

The EM algorithm described above is a batch algorithm which cannot be directly applied to our learning problem. In order to develop a sequential EM algorithm, we first consider $p(\boldsymbol{x}_{0:T}; \theta)$ given by

$$\begin{aligned}
&\log p(\boldsymbol{x}_{0:T}; \theta) \\
&= C + \frac{T}{R}\log\beta - \frac{T}{2}\log\sigma^2 - \beta\sum_{t=1}^{T}|s_t - \boldsymbol{\alpha}^\top \boldsymbol{s}_{t-1}|^R \\
&\quad -\frac{1}{2\sigma^2}\sum_{t=1}^{T}(n_t - \boldsymbol{\gamma}^\top \boldsymbol{n}_{t-1})^2, \\
&= C + \sum_{t=1}^{T}\log p(\boldsymbol{x}_t; \theta),
\end{aligned}$$

where C is a constant which is independent of parameters and $\log p(\boldsymbol{x}_t; \theta_t)$ is a single factor of the complete-data log-likelihood at time t, given by

$$\begin{aligned}
&\log p(\boldsymbol{x}_t; \theta) \\
&\triangleq \frac{1}{R}\log\beta - \frac{1}{2}\log\sigma^2 - \beta|s_t - \boldsymbol{\alpha}^\top \boldsymbol{s}_{t-1}|^R \\
&\quad -\frac{1}{2\sigma^2}(n_t - \boldsymbol{\gamma}^\top \boldsymbol{n}_{t-1})^2.
\end{aligned}$$

In order to accommodate the sequential learning, we modify the E-step in such a way that the expected complete-data log-likelihood is evaluated through the expectation given observations up to t (instead of the whole observations). In other words, the sequential EM updating has the form

E-step

$$\begin{aligned}
&Q(\theta_{t+1}, \widehat{\theta}_t) \\
&= \mathcal{L}_{t+1}(\theta_{t+1}) \\
&= \sum_{\tau=1}^{t+1}\lambda^{t+1-\tau}\mathbb{E}\{\log p(\boldsymbol{x}_\tau; \theta_\tau)|y_{1:\tau}, \widehat{\theta}_{\tau-1}\} \\
&= \lambda L_t(\theta_t) + \mathbb{E}\{\log p(\boldsymbol{x}_{t+1}; \theta_{t+1})|y_{1:t+1}, \widehat{\theta}_t\},
\end{aligned}$$

where λ is the forgetting factor $0 \leq \lambda \leq 1$.

M-step

$$\widehat{\theta}_{t+1} = \arg\max_{\theta_{t+1}} Q(\theta_{t+1}, \widehat{\theta}_t).$$

The E-step involves computing

$$\mathbb{E}\{\log p(\boldsymbol{x}_t; \theta_t)|y_{1:t}, \widehat{\theta}_{t-1}\},$$

which can be carried out using the particle filter,

$$\mathbb{E}\{\log p(\boldsymbol{x}_t; \theta_t)|y_{1:t}, \widehat{\theta}_{t-1}\} = \sum_{i=1}^{N}\widetilde{w}_t^{(i)}\log p(\boldsymbol{x}_t^{(i)}). \qquad (22)$$

The 2nd-order Taylor approximation of $Q(\theta_{t+1}, \widehat{\theta}_t)$, leads to the M-step [5], [12] that has the form

$$\begin{aligned}
\widehat{\theta}_{t+1} &= \widehat{\theta}_t + \widehat{\boldsymbol{H}}_{t+1}^{-1}\boldsymbol{\varphi}(\widehat{\theta}_t), &(23)\\
\widehat{\boldsymbol{H}}_{t+1} &= \lambda\widehat{\boldsymbol{H}}_t + \boldsymbol{H}(\widehat{\theta}_t), &(24)
\end{aligned}$$

where

$$\begin{aligned}
\boldsymbol{\varphi}(\theta_t) &= \mathbb{E}\{\nabla_{\theta_t}\log p(\boldsymbol{x}_t; \theta_t)|y_{1:t}, \widehat{\theta}_{t-1}\}, \\
\boldsymbol{H}(\theta_t) &= -\mathbb{E}\{\nabla_{\theta_t}^2\log p(\boldsymbol{x}_t; \theta_t)|y_{1:t}, \widehat{\theta}_{t-1}\}.
\end{aligned}$$

These updating rules (23) and (24) are referred to as *sequential Newton-Raphson EM* (SNEM).

We also consider a simpler updating rule that is of the form

$$\widehat{\theta}_{t+1} \;=\; \widehat{\theta}_t + \lambda_0 \boldsymbol{H}(\widehat{\theta}_t)^{-1}\boldsymbol{\varphi}(\widehat{\theta}_t), \tag{25}$$

where λ_0 is a sufficiently-small positive constant. The updating rule (25) use only current Hessian $\boldsymbol{H}(\widehat{\theta}_t)$. In such a case, we employ the quasi-Newton method where the inverse of the Hessian is approximated. This simpler algorithm is referred to as *sequential quasi-Newton EM* (SQEM).

B. Sequential Updating Rules

It follows from (22) and (22) that we have

$$\begin{aligned}\mathbb{E}\{\log p(\boldsymbol{x}_t;\theta_t)|y_{1:t},\widehat{\theta}_{t-1}\}&\\ =\;& \mathbb{E}\{\log p(\boldsymbol{s}_t;\boldsymbol{\alpha}_t,\beta_t)|y_{1:t},\widehat{\theta}_{t-1}\}\\ +\;& \mathbb{E}\{\log p(\boldsymbol{n}_t;\boldsymbol{\gamma}_t,\sigma_t^2)|y_{1:t},\widehat{\theta}_{t-1}\},\end{aligned} \tag{26}$$

which implies that updating parameters for speech and noise can be done separately. In addition, the decomposition (26) allows us to evaluate the statistical expectation using RBPF. The first term in Eq. (26) is approximated by

$$\mathbb{E}\{\log p(\boldsymbol{s}_t;\boldsymbol{\alpha}_t,\beta_t)|y_{1:t},\widehat{\theta}_{t-1}\} \approx \sum_{i=1}^{N} \widetilde{w}_t^{(i)} \log p(\boldsymbol{s}_t^{(i)};\boldsymbol{\alpha}_t,\beta_t),$$

where $\boldsymbol{s}_t^{(i)}$ is particles of the RBPF. The second term in Eq. (26) can be easily calculated, because the expectation is carried out with respect to Gaussian distribution given in Eq. (13). Therefore, the sequential parameter updating is carried out in the framework of RBPF.

Updating rules for parameters $\{\boldsymbol{\alpha}_t,\beta_t\}$ in the speech model, are given by

$$\widehat{\boldsymbol{\alpha}}_{t+1} \;=\; \widehat{\boldsymbol{\alpha}}_t + \widehat{\boldsymbol{H}}_{t+1}^{-1(s)}\boldsymbol{\varphi}^{(s)}(\widehat{\theta}_t), \tag{27}$$

$$\widehat{\beta}_{t+1} \;=\; \frac{1-\lambda_s}{1-\lambda_s^{t+1}}\{\epsilon(t+1)\}, \tag{28}$$

$$\widehat{\boldsymbol{H}}_{t+1}^{(s)} \;=\; \lambda\widehat{\boldsymbol{H}}_t^{(s)} + \boldsymbol{H}^{(s)}(\widehat{\theta}_t),$$

where

$$\begin{aligned}\boldsymbol{\varphi}^{(s)}(\widehat{\theta}_t) &= \mathbb{E}\{\nabla_{\boldsymbol{\alpha}}\log p(\boldsymbol{s}_{t+1};\boldsymbol{\alpha})|_{\boldsymbol{\alpha}=\widehat{\boldsymbol{\alpha}}_t}|y_{1:t+1},\widehat{\theta}_t\}\\ &= \widehat{\beta}_t R\sum_{i=1}^{N}\widetilde{w}_{t+1}^{(i)}\mathrm{sign}(e_{t+1}^{(i)})|e_{t+1}^{(i)}|^{R-1}\boldsymbol{s}_t^{(i)},\\ \boldsymbol{H}^{(s)}(\widehat{\theta}_t) &= -\mathbb{E}\{\nabla_{\boldsymbol{\alpha}}^2\log p(\boldsymbol{s}_{t+1};\boldsymbol{\alpha})|_{\boldsymbol{\alpha}=\widehat{\boldsymbol{\alpha}}_t}|y_{1:t+1},\widehat{\theta}_t\}\\ &= \widehat{\beta}_t R(R-1)\sum_{i=1}^{N}\widetilde{w}_{t+1}^{(i)}|e_{t+1}^{(i)}|^{R-2}\boldsymbol{s}_t^{(i)}(\boldsymbol{s}_t^{(i)})^\top,\\ \epsilon(t+1) &\triangleq \sum_{\tau=1}^{t+1}\lambda_s^{t-\tau+1}R\mathbb{E}\{|e_\tau|^R|y_{1:\tau},\widehat{\theta}_{\tau-1}\},\\ &= \lambda_s\epsilon(t) + R\mathbb{E}\{|e_{t+1}|^R|y_{1:t+1},\widehat{\theta}_t\},\\ e_{t+1} &= s_{t+1}-\widehat{\boldsymbol{\alpha}}_t^\top\boldsymbol{s}_t,\quad e_{t+1}^{(i)} = s_{t+1}^{(i)}-\widehat{\boldsymbol{\alpha}}_t^\top\boldsymbol{s}_t^{(i)},\end{aligned}$$

where e_t is a prediction error in the speech GAR model and $e_t^{(i)}$ is a prediction error of each particle and $0 \le \lambda_s \le 1$ is the forgetting factor for β. Note that β is restricted to be a positive value. Thus, the sequential EM can not be directly applied. Instead, we use a gradient method, following a suggestion in [6].

Updating rules for parameters in the noise model are simple, since they involve Gaussian distribution. Let us define the matrix $\boldsymbol{V}_t$ as

$$\begin{aligned}\boldsymbol{V}_t &\triangleq \mathbb{E}\{\widetilde{\boldsymbol{n}}\widetilde{\boldsymbol{n}}^\top|y_{1:t},\widehat{\theta}_{t-1})\},\\ &= \begin{bmatrix}\boldsymbol{V}_t^{11} & \boldsymbol{V}_t^{12}\\ \boldsymbol{V}_t^{21} & \boldsymbol{V}_t^{22}\end{bmatrix},\end{aligned}$$

where $\widetilde{\boldsymbol{n}}_t = [n_t \quad \boldsymbol{n}_{t-1}]^\top$, $\boldsymbol{V}_t^{11}$ is a scalar, $\boldsymbol{V}_t^{22}$ is a $q\times q$ matrix, and $\boldsymbol{V}_t^{12} = (\boldsymbol{V}_t^{21})^\top$is a q-dimensional vector [6]. With these definitions, updating rules are given by

$$\widehat{\boldsymbol{\gamma}}_{t+1} \;=\; \widehat{\boldsymbol{\gamma}}_t + \widehat{\boldsymbol{H}}_{t+1}^{-1(n)}\boldsymbol{\varphi}^{(n)}(\widehat{\theta}_t), \tag{29}$$

$$\widehat{\sigma}_{t+1}^2 \;=\; \frac{1-\lambda_n}{1-\lambda_n^{t+1}}\sum_{\tau=1}^{t+1}\lambda_n^{t-\tau+1}\{\boldsymbol{V}_\tau^{11}-\widehat{\boldsymbol{\gamma}}_{\tau-1}^\top\boldsymbol{V}_\tau^{12}\} \tag{30}$$

$$\widehat{\boldsymbol{H}}_{t+1}^{(n)} \;=\; \lambda\widehat{\boldsymbol{H}}_t^{(n)} + \boldsymbol{H}^{(n)}(\widehat{\theta}_t),$$

where

$$\begin{aligned}\boldsymbol{\varphi}^{(n)}(\widehat{\theta}_t) &= \mathbb{E}\{\nabla_{\boldsymbol{\gamma}}\log p(\boldsymbol{n}_{t+1};\boldsymbol{\gamma})|_{\boldsymbol{\gamma}=\widehat{\boldsymbol{\gamma}}_t}|y_{1:t+1},\widehat{\theta}_t\}\\ &= \frac{1}{\widehat{\sigma}_t^2}\{\boldsymbol{V}_{t+1}^{12}-\boldsymbol{V}_{t+1}^{22}\widehat{\boldsymbol{\gamma}}_t\},\\ \boldsymbol{H}^{(n)}(\widehat{\theta}_t) &= -\mathbb{E}\{\nabla_{\boldsymbol{\gamma}}^2\log p(\boldsymbol{n}_{t+1};\boldsymbol{\gamma})|_{\boldsymbol{\gamma}=\widehat{\boldsymbol{\gamma}}_t}|y_{1:t+1},\widehat{\theta}_t\}\\ &= \frac{1}{\widehat{\sigma}_t^2}\{\boldsymbol{V}_{t+1}^{22}\},\end{aligned}$$

and $0 \le \lambda_n \le 1$ is the forgetting factor for σ^2.

As mentioned in Sec. VI-A, we also apply the sequential Quasi-Newton method. Our two proposed algorithms are referred to as: (1) RBPF+SNEM (Rao-Blackwellized particle filtering+ sequential Newton-Raphson EM); (2) RBPF+SQEM (Rao-Blackwellized particle filtering+ sequential quasi-Newton EM). The outline of our sequential speech enhancement method is summarized in Table I.

TABLE I

ALGORITHM OUTLINE: OUR SEQUENTIAL SPEECH ENHANCEMENT.

Generation step: Generate particles according to (19) and update $\boldsymbol{\mu}_{t|t-1}^{(i)}$ and $\boldsymbol{\Sigma}_{t|t-1}^{(i)}$ for each Kalman filter.

Weight updating step: Update importance weights by (20) and normalize them to sum to one.

Resampling step: Resample particles, $\{\boldsymbol{\mu}_{t|t-1}^{(i)},\boldsymbol{\Sigma}_{t|t-1}^{(i)},y_{t|t-1}^{(i)}\}$, according to importance weights, using the method in [9].

MCMC diversity step: Apply Markov Chain Monte Carlo (MCMC) to the invariant distribution $p(\boldsymbol{s}_{0:t}|y_{1:t})$, leading some particles to move a more probable region within preserving the invariant distribution [1].

Kalman update step: Update $\{\boldsymbol{\mu}_{t|t}^{(i)},\boldsymbol{\Sigma}_{t|t}^{(i)}\}$ through Kalman using the update rule for Kalman filter in section (V-A).

Parameter estimation step: Update parameters of both speech and noise model using sequential Newton or Quasi-Newton EM algorithm in section VI

VII. Experimental Results

For experiments, we used a speech signal that is publicly available [1]. The speech signal was resampled at 8 kHz and first 5000 data points ($T = 5000$) were used in our experiments.

The order of the speech GAR model was set as $p = 12$ and the order of the noise AR model was set as $q = 5$. The value of R in the generalized exponential density was set as $R = 1.25$. The number of particles were $N = 200$.

As a performance measure, we evaluate output signal-noise-ratio (SNR), with respect to various input SNR. The output SNR is defined by

$$\begin{aligned} \mathrm{SNR}_{out} &= 10\log_{10}\frac{\sum_{t=1}^{T} s_t^2}{\sum_{t=1}^{T}[s_t - \widehat{s}_t]^2}, \\ \widehat{s}_t &= \sum_{i=1}^{N} \widetilde{w}_t^{(i)} \boldsymbol{b}_s^\top \boldsymbol{s}_t^{(i)}, \end{aligned}$$

where s_t is a clean speech signal and $\widehat{s}_t$ is its estimate. In the calculation of the input SNR, $\widehat{s}_t$ is replaced by y_t in Eq. (31). The output SNR was evaluated by averaging 30 independent runs for each input SNR.

We compare our method to a Kalman filter-based sequential speech enhancement method. Kalman-gradient-descent-sequential (KGDS) algorithm [6], employs Gaussian density for speech model and updates parameters sequentially using a gradient method. For the case of high input SNR, KGDS is sometimes unstable [6]. Thus, we consider the SNEM in the framework of Kalman filter, leading to KF+SNEM where inference and parameter learning is carried out using Kalman filter. Experimental results are shown in Fig. 1, where output SNRs for 4 different algorithms are plotted, with respect to various input SNRs (varying from 0 dB to 10 dB).

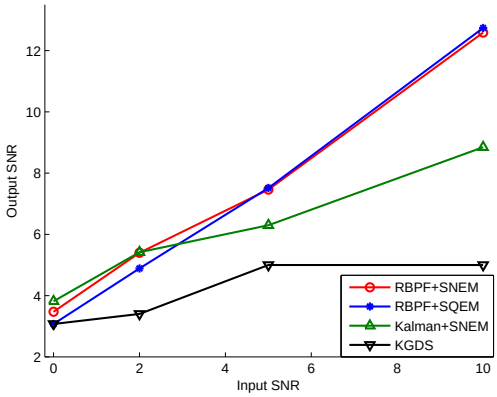

Fig. 1. Performance of sequential speech enhancement methods are shown in terms of output SNR with respect to input SNR.

Fig. 1 shows that our methods (RBPF+SNEM and RBPF+SQEM) outperform Kalman filter-based methods (KGDS and KF+SNEM). In case of high input SNR (input SNR $\geq$ 5dB), the observed signal is closer to actual speech signal, implying that it is far from Gaussian assumption. In such a case, we observe that our methods are much more appropriate, compared to Kalman filter-based methods. RBPF+SQEM is quite comparable to RBPF+SNEM, although the former takes a simpler updating rule than the latter.

[1] http://www.ece.mcmaster.ca/ reilly/html/projects/dereverb/speechRHINTE.wav

VIII. Conclusions

We have presented new sequential speech enhancement algorithm which employ the GAR speech model in order to accommodate the non-Gaussian characteristics of speech. Two new algorithms, including RBPF+SNEM and RBPF+SQEM, employ Rao-Blackwellized particle filter for inference and update model parameters using the sequential EM method. Experimental results confirmed the high performance of our proposed methods, compared to Kalman filter-based methods.

Acknowledgments: This work was supported by ITEP Brain Neuroinformatics Program and Korea MIC under ITRC support program supervised by the IITA (IITA-2005-C1090-0501-0018).

References

[1] C. Andrieu, N. de Freitas, A. Doucet, and M. I. Jordan, "An introduction to MCMC for machine learning," *Machine Learning*, vol. 50, pp. 5–43, 2003.

[2] S. Choi, A. Cichocki, and S. Amari, "Flexible independent component analysis," *Journal of VLSI Signal Processing*, vol. 26, no. 1/2, pp. 25–38, Aug. 2000.

[3] N. de Freitas, "Rao-Blackwellised particle filtering for fault diagnosis," in *IEEE Aerospace Conf.*, 2001, pp. 1767–1772.

[4] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, pp. 197–208, 2000.

[5] L. Frenkel and M. Feder, "Recursive expectation-maximization (EM) algorithms for time-varying parameters with applications to multiple target tracking," *IEEE Trans. Signal Processing*, vol. 47, pp. 306–320, 1999.

[6] S. Gannot, D. Burshtein, and E. Weinstein, "Iterative and sequential Kalman filter-based speech enhancement algorithms," *IEEE Trans. Speech and Audio Processing*, vol. 6, pp. 373–385, 1998.

[7] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P. J. Nordlund, "Particle filters for positioning, navigation, and tracking," *IEEE Trans. Signal Processing*, vol. 50, no. 2, pp. 425–437, 2002.

[8] G. S. Kang and L. J. Fransen, "Quality improvement of LPC-processed noisy speech by using spectral subtraction," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 37, pp. 939–943, 1989.

[9] G. Kitagawa, "Monte Carlo filter and smoother for non-Gaussian nonlinear state space models," *Journal of Computational and Graphical Statistics*, vol. 5, pp. 1–25, 1996.

[10] K. K. Paliwal and A. Basu, "A speech enhancement method based on Kalman filtering," in *Proc. ICASSP*, 1987, pp. 177–180.

[11] X. Shen and L. Deng, "A dynamic system approach to speech enhancement using the H_∞ filtering algorithm," *IEEE Trans. Speech and Audio Processing*, vol. 7, pp. 391–399, 1999.

[12] D. M. Titterington, "Recursive parameter estimation using incomplete data," *Journal of the Royal Statistical Society*, vol. 46, pp. 257–267, 1984.

[13] J. Vermaak, C. Andrieu, A. Doucet, and S. J. Godsill, "Particle methods for Bayesian modeling and enhancement of speech signals," *IEEE Trans. Speech and Audio Processing*, vol. 10, pp. 173–185, 2002.

[14] Y. Yang, "Generating generalized exponentially distributed random variates with transformed density rejection and ratio-of-uniform methods," Master's thesis, Virginia Polytechnic Institute and State University, 2004.