Minimum Mean-Square Error Single-Channel Signal Estimation

Beierholm, Thomas; Winther, Ole

Publication date:
2008

Document Version
Publisher's PDF, also known as Version of record

Minimum Mean-Square Error Single-Channel Signal Estimation

Thomas Beierholm

Kongens Lyngby 2007
IMM-PHD-2007-185

Technical University of Denmark
Informatics and Mathematical Modelling
Building 321, DK-2800 Kongens Lyngby, Denmark
Phone +45 45253351, Fax +45 45882673
reception@imm.dtu.dk
www.imm.dtu.dk

IMM-PHD: ISSN 0909-3192

Summary

This topic of this thesis is MMSE signal estimation for hearing aids when only one microphone is available. The research is relevant for noise reduction systems in hearing aids. To fully benefit from the amplification provided by a hearing aid, noise reduction functionality is important as hearing-impaired persons in some noisy situations need a higher signal to noise ratio for speech to be intelligible when compared to normal-hearing persons.

In this thesis two different methods to approach the MMSE signal estimation problem is examined. The methods differ in the way that models for the signal and noise are expressed and in the way the estimator is approximated. The starting point of the first method is prior probability density functions for both signal and noise and it is assumed that their Laplace transforms (moment generating functions) are available. The corresponding posterior mean integral that defines the MMSE estimator is rewritten into an inverse Laplace transform integral over an integrand involving the moment generating functions. This integral is approximated using saddlepoint approximation. It is found that the saddlepoint approximation becomes inaccurate when two saddlepoints coalesce and a saddlepoint approximation based on two coalescing saddlepoints is derived. For practical reasons the method is limited to low dimensional problems and the results are not easily extended to the multivariate case.

In the second approach signal and noise are specified by generative models and approximate inference is performed by particle filtering. The speech model is a time-varying auto-regressive model reparameterized by formant frequencies and bandwidths. The noise is assumed non-stationary and white. Compared to the case of using the AR coefficients directly then it is found very beneficial

to perform particle filtering using the reparameterized speech model because it is relative straightforward to exploit prior information about formant features. A modified MMSE estimator is introduced and performance of the particle filtering algorithm is compared to a state of the art hearing aid noise reduction algorithm. Although performance of the two algorithms is found comparable then the particle filter algorithm is doing a much better job tracking the noise.

Resumé

Emnet for denne PhD rapport er MMSE signal estimering til brug i høreapparater som kun har én mikrofon. Emnet er relevant for støjreduktionssystemer til høreapparater. For fuldt ud at kunne drage nytte af forstærkningen i et høreapparat er det vigtigt med støjreduktionsfunktionalitet da hørehæmmede personer i nogle støjfyldte situationer har brug for et højere signal-støj forhold for at kunne forstå tale når der sammenlignes med normalthørende personer.

I denne rapport undersøges to forskellige metoder til MMSE signal estimeringsproblemet. Metoderne benytter forskellige repræsentationer af signal og støj og de benytter forskellige metoder til at approksimation af MMSE estimatoren. Udgangspunktet for den første tilgang er prior sandsynlighedstæthedsfunktioner for både signal og støj og det antages at deres Laplace transform (moment genererende funktion) er tilgængelig. Det tilhørende posterior middelværdi integral som definerer MMSE estimatoren omskrives til et invers Laplace tranform integral over en integrand som består at afledede af de moment genererende funktioner. Dette integral approksimeres ved hjælp af saddelpunktsapproksimation. Det fremgår at saddelpunktsapproksimationen bliver upræcis når to saddelpunkter er tæt på hinanden og en saddelpunktsapproksimation baseret på to saddelpunkter tæt på hinanden udledes. Af praktiske grunde er metoden begrænset til problemer af lav dimensionalitet og resultaterne kan ikke umiddelbart generaliseres til det flervariable tilfælde.

I den anden metode er signal og støj specificerede ved generative modeller og inferens approksimeres ved hjælp af partikelfiltrering. Tale modellen er en tidsvarierende autoregressiv model reparameteriseret ved formant frekvenser og båndbredder. Støjen antages ustationær og hvid. Sammenlignet med tilfældet hvor AR koefficienter benyttes direkte så er det fundet fordelagtigt at lave par-

tikelfiltrering i den reparameteriserede talemodel fordi det er relativt simpelt at benytte forhåndsviden omkring formant frekvenser og båndbredder. En modificeret MMSE estimator introduceres og performance af partikelfilteralgoritmen sammenlignes med en støjreduktionsalgoritme some bruges i høreapparater. Selvom performance af de to algoritmer er sammenlignelig så er partikelfilteralgoritmen meget bedre til at tracke støjen.

Preface

The work presented in this thesis was carried out at Informatics and Mathematical Modelling, Technical University of Denmark and at GN ReSound A/S in partial fulfillment of the requirements for acquiring the Ph.D. degree in electrical engineering.

The goal of this thesis is to provide a unifying framework of the research carried out in the Ph.D. study during the period Jun. 2004 - Oct. 2007, excluding a leave of absence in July 2007.

Contributions

Two conference papers were written just prior to the official start of the project and mentioned here as they to some degree have had an impact on the work done in the official project period.

- Thomas Beierholm, Brian Dam Pedersen and Ole Winther, *Low Complexity Bayesian Single Channel Source Separation*, In Proceedings ICASSP-04, May 2004, Vol. 5, pp. 529 – 532, Montreal, Canada.
- Thomas Beierholm and Paul M. Baggenstoss, *Speech Music Discrimination Using Class-Specific Features*, Proceedings of 17^{th} International Conference on (ICPR'04), Vol. 2, pp. 379 – 382, August 2004, Cambridge, UK.

This project has produced three papers with the following contributions

- Thomas Beierholm, Albert H. Nuttall and Lars K. Hansen, *Use and Subtleties of Saddlepoint Approximation for Minimum Mean-Square Error Estimation*, Submitted to IEEE Transactions on Information Theory June 2006.
  - Derivation of an integral representation for the minimum mean-square error estimator involving the moment-generating functions of the random variables in an observation model consisting of a linear combination of two independent random variables.

  - demonstrates that for the case of two coalescing saddlepoints, the saddlepoint approximation based on isolated saddlepoints is very inaccurate.
  - Derivation of a saddlepoint approximation based on two coalescing saddlepoints with two correction terms.
  - Demonstrates that the saddlepoint approximation based on two coalescing saddlepoints makes an excellent approximation when in fact two saddlepoints are coalescing.

- Thomas Beierholm and Ole Winther, *Particle Filter Inference in an Articulatory Based Speech Model*, Accepted for publication in IEEE Journal of Signal Processing Letters.

  - Demonstration of particle filter inference in a parallel formant synthesizer model parameterized by formant frequencies, bandwidths and gains for the application of speech enhancement.
  - Experiments show that performance of the proposed speech model provides higher SNR improvements than a speech model parameterized by auto-regressive coefficients.

- Thomas Beierholm, Ole Winther and Bert de Vries, *Non-Stationary Noise Reduction by Particle Filtering in a Cascade Formant Synthesizer Model*, Submitted to IEEE Transactions on Speech and Language Processing June 2007.

  - It is demonstrated that reducing noise in speech by particle filter inference in a time-varying auto-regressive model is more beneficial if prior knowledge of formants is exploited which is achieved by parameterizing the AR coefficients by formant frequencies and formant bandwidths
  - Adapting the gains (variances) of the speech and noise models directly instead of the log-gains produced results with less fluctuations and more importantly it eliminated a stability issue experienced when using the log-gains.
  - It is found that noise tracking capabilities of the proposed particle filter algorithm are superior to that of a hearing instrument noise reduction algorithm, especially for low SNR levels.

Acknowledgements

First of all, I would like to thank GN ReSound A/S and in particular the Algorithm R&D group for funding, financial support and commitment to setting up the project despite difficulties in doing so. I also thank the Algorithm R&D group for letting me roam around freely and members of the group for pleasant company. I would like to thank Sten Frostholm for IT support and Shawn Perman for being full of surprises.

Special thanks goes to my supervisors, Dr. Bert de Vries and Dr. Ole Winther, for guidance and support. Their contributions to the PhD project are hard to overestimate. I would also like to thank Prof. Lars Kai Hansen for always being open and available for discussing ideas and the MISP department at Aalborg university for housing me in the final stages of the project.

I would like to thank Dr. Philip Ainsleigh for bringing up the idea of pursuing a PhD project, Dr. Paul Baggenstoss for explaining basic and subtle matters on the class-specific approach, Prof. Peter Willett who made my (unfortunately too short) visit at UConn possible in the first place but also very warm and in particular Albert Nuttall whom I feel very privileged to have worked together with.

I also thank Line for understanding and support during the time of the project.

Contents

Chapter 1

Introduction

A modern digital hearing aid provides DSP solutions to several challenging problems that are of high importance in trying to help persons who suffer from a hearing loss. One of these problems consists of reducing noise in the input signal to the hearing aid. If there is noise in the input then the amplification provided by the hearing aid may not help the user because persons with a hearing loss tend to be more sensitive to noise than normal hearing persons [10]. Therefore a user of a hearing aid can experience a situation where speech is entirely audible but at the same time impossible to understand. This means that for a given noisy situation where speech is understandable for a normal-hearing person the speech may not be understandable for a hearing aid user because a higher Signal to Noise Ratio (SNR) is necessary before the noisy speech is intelligible. Conversely, in situations where the amplified noisy speech is intelligible for a hearing aid user, suppressing noise is also important in order to reduce listener fatigue and thus make it more comfortable for the user to wear the hearing aid.

There are a number of constraints a developer faces when designing a noise reduction system for a hearing aid. Since the very beginning of digital hearing aids the constraint that stands out the most is the fact the resources are very scarce and a noise reduction algorithm therefore must be of very low complexity to be a serious candidate for implementation. Furthermore, the noise reduction algorithm must introduce a small delay in the signal path and introduce virtually

no audible distortions or artifacts in the processed speech signal. Additionally, the noise reduction algorithm should preferably operate in the frequency domain and work for all types of speech and noise. These constraints makes it challenging to develop a useful noise reduction algorithm for hearing aids.

If a hearing aid has two or more microphones available then spatial information can be used to suppress noise. The principle behind such a multi-microphone noise reduction algorithm is to place a beam in the direction of the desired speaker to let speech pass undisturbed and nulls in the directions of the noise sources. Currently such algorithms tend to be more effective and useful than single-microphone noise reduction algorithms for suppressing noise [10]. However, not all hearing aids have two or more microphones but only one microphone. Hence, there is an interest in single-microphone noise reduction algorithms for hearing aids.

A single-channel noise reduction algorithm can not rely on spatial information and is thus less informed than multi-microphone approaches. In this report it is assumed that the input to the hearing aid, x, is a sum of a signal, s, and noise, v, which leads to the following observation model

$$x_n = s_n + v_n \tag{1.1}$$

where n denotes a sample/time index. The signal can for instance be thought of as speech and the noise can be thought as ambient (broad or narrow band) noise or a competing speaker. The nature of the problem in single-channel signal estimation is thus that in order to estimate the signal s_n, or for that matter the noise v_n, the n^{th} input sample x_n must be split into two numbers/samples such that (1.1) is obeyed. If only x_n is available (blind approach) there are an infinite number of ways to split x_n into s_n and v_n, such that (1.1) is obeyed. Mathematically such a problem is called under-determined meaning that more information is required in order to compute a unique solution.

Ways to approach a solution to the single-channel signal estimation problem tend to be quite dependent on the situation at hand. If for instance it is known that both signal and noise are stationary, that the signal is a sinusoid, the noise is pink and the SNR is between 10 and 15 dB then the situation is quite different from a competing speaker situation where both signal and noise represent speech. In the former situation the signal can be recovered by estimating only three parameters, the frequency, phase and amplitude of the sinusoid, whereas in the latter situation a Computational Auditory Scene Analysis (CASA) approach may be considered.

A systematic approach to the single-channel estimation problem can be performed using Bayesian methodology. Knowledge about the signal and noise is encapsulated in statistical models also called prior distributions (or just priors).

Using data available, Bayes' theorem converts the priors to posteriors. The posteriors are attractive because they encompass all available information; both the prior knowledge provided by the developer in the priors and the information conveyed in the acquired data. For instance, assuming that the signal and noise are independent then by Bayes theorem

$$p(s_n|D)p(v_n|D) = \frac{p(D|s_n, v_n)p_s(s_n)p_v(v_n)}{p(D)} \tag{1.2}$$

where D denotes all available acquired data, D=$\{..., x_{n-1}, x_n\}$. In the Bayesian methodology inference and estimation are centered around posterior distributions. The MMSE estimator of the signal is defined as the mean of the signal posterior and is computed by the following integral

$$\hat{s}_n = \int ds_n \; s_n \; p(s_n|D) \tag{1.3}$$

As mentioned in [15], the MMSE estimator is optimal for a large class of difference distortion measures and it tends to be the most used estimator for noise reduction, therefore MMSE estimation is adopted in this report.

The overall problem statement for this report is: To perform MMSE estimation of speech in a single-channel setup.

1.1 Saddlepoint Approximation

Inserting (1.2) in (1.3) and enforcing (1.1) yields

$$\hat{s}_n = \frac{1}{p(D)} \int ds_n \; s_n \; p_s(s_n) \; p_v(x_n - s_n) \tag{1.4}$$

The MMSE estimator is thus expressed directly in terms of the priors by a weighted convolution integral. It is thus clear that the choice of priors is what ultimately determines the performance of the corresponding MMSE estimator and they are parameters under control of the developer. In principle, (1.4) represents an attractive 'plug-and-play' framework in which signal and noise models can be inserted and (1.4) provides the MMSE estimator for the signal. For a developer of noise reduction algorithms such a setup could be very beneficial. Choose models (priors) for the signal and noise, choose a noisy signal and (1.4) provides the MMSE estimate of the signal for listening. In this way a developer can spend most of the time extending and enhancing models for the signal

and noise and thus increase the chance of developing an improved noise reduction algorithm. However, in reality a developer is quite limited in the choice of models because of inferential difficulties. There is no straightforward way to do the integral in (1.4) for all but the simplest models. The integral in (1.4) thus represents a bottleneck for a developer who will spend most of the time dealing with the integration instead of concentrating on improving the signal and noise models.

The topic of chapter 2 is approximation of the integral in (1.4) by the method of Saddlepoint Approximation (SPA). The idea has been motivated by work done by Albert H. Nuttall in obtaining accurate approximations to tail probabilities for a number of useful statistics (see e.g. [29, 30]) which has been an important method in developing the class-specific approach [3, 4]. A number of observations made SPA an interesting topic for further investigation; the SPA provided excellent tail probability approximations, the SPA was developed for statistics consisting of a linear combination of independent random variables (note that the observation model in (1.1) is a linear combination of two independent random variables) and the SPA was computed by searching for a unique real saddlepoint. Thus the prospect was an accurate approximation of the integral in (1.4) where the problem of evaluating the integral in (1.4) could be replaced by the simpler problem of searching for a unique real saddlepoint, hence closing in on a 'plug-and-play' setup.

Problem statement 1*: Approximate the integral in (1.4) by means of saddlepoint approximation for general priors.*

1.2 Particle Filtering

As mentioned in section 1.1 the choice of priors for the signal and noise is crucial for the resulting performance of the corresponding MMSE estimator. The starting point of the approach described in section 1.1 is the availability of the models in terms of the prior PDFs, $p_s(s_n)$ and $p_v(v_n)$ or their Laplace transforms as it will appear in chapter 2. Alternatively, priors can be specified by generative models which explicitly states how the data (observations) are assumed to have been generated. For instance, speech is often modeled as a time-varying auto-regressive process driven by white Gaussian noise where AR coefficients and innovation noise variance can change from sample to sample. Using such a model for speech it may be cumbersome and difficult to try and derive an expression for $p_s(s_n)$ because parameters are unknown and must be integrated

out. If instead the noisy speech is formulated as a state-space model where clean speech enters as a hidden variable a number of methods for approximate inference can be used. One such method is particle filtering.

The idea of investigating particle filter inference in a time-varying auto-regressive model for speech was initially inspired by work done by Hermansen et al. in [22, 20] where the model based speech spectrum (the speech spectrum envelope) was parameterized by formant features; frequencies, bandwidths and gains. The prospect of using such features instead of AR coefficients directly is to facilitate incorporation of prior knowledge of the formant features in the MMSE estimation. Inference by particle filtering in a time-varying auto-regressive speech model was performed by Vermaak et al. in [39] and has been adopted in this report and modified to work with formant features.

Problem statement 2*: Perform particle filter inference in a time-varying auto-regressive generative model for speech parameterized by formant features.*

1.3 Reading Guideline

The main body of this report consists of two self-contained independent chapters, one chapter on MMSE estimation using saddlepoint approximation and the other chapter on MMSE estimation using particle filtering. The purpose of these two chapters is primarily to introduce the basic ideas behind the methods of saddlepoint approximation and particle filtering but the intent is also that they can serve as background reading for the three papers found in appendices E-G. Although the two main chapters in this report are self-contained the papers should be read together with the chapters as it has been an aim to minimize the overlap between the chapters and the papers. As such the papers constitute an important part of this report. Also the results in the papers will be referred to in chapter 4.

Chapter 2

Saddlepoint Approximation for MMSE Estimation

This chapter describes work done on Minimum Mean-Square Error (MMSE) estimator approximation using SaddlePoint Approximation (SPA). Deriving the MMSE estimator for a signal using observations from a single channel when the mixing coefficients are known can be intractable for non-standard distributions. The derivation involves a weighted convolution between the Probability Density Functions (PDFs) of the involved Random Variables (RVs). The idea presented in this chapter is to make use of integral transforms of the involved random variables instead of the PDFs and reformulate the original integral into an inverse transform integral. The applied integral transform is the Laplace transform. The inverse transform integral will in general also be intractable, however, by transforming the path of integration to go through a SaddlePoint (SP) of the integrand, the inverse integral transform can be approximated by using information only at the SP.

The work described in this chapter was motivated by a desire to extend the work in [5] where separation of two speakers was performed in a two-step procedure: Maximum Likelihood (ML) estimation of mixing coefficients and MMSE estimation of the sources conditional on the mixing coefficients. As a result of this work the question arose as to what kind of source PDFs would result in excellent separation performance given that the mixing coefficients are known and that MMSE estimation is performed. One obstacle in extending the work in [5] is

the derivation of the MMSE estimator. Considering for example Class-specific PDFs or multivariate sparse distributions the derivation of an exact analytic expression for the MMSE estimator looks like a daunting task.

The work described in this chapter is inspired by work done by Albert H. Nuttall in obtaining accurate tail probability evaluations using SPA for a number of useful statistics see e.g. [29][30]. The author is indebted to Albert H. Nuttall without whom this work would not have been possible. He has acted as both a mentor and a collaborator.

The purpose of this chapter is to explain or describe background theory that may be useful when reading the paper "Use and Subtleties of Saddlepoint Approximation for Minimum Mean-Square Error Estimation" in appendix. Especially section 2.2.4 may be helpful. Perhaps "theory" is not the correct word to use because the chapter emphasizes intuitive explanations/descriptions over mathematical rigor. The contributions of the work done on MMSE estimator approximation are the derivation and use of the inverse transform integral formulation based on Moment Generating Functions (MGFs) and the derivation of the SPA for two close or coalesced SPs. The formulation based on MGFs seems new. When two SPs are close or coalesce the normal SPA based on a single SP does not work which motivated the derivation of the SPA for two close or coalesced SPs. This chapter also extends some of the ideas to the M-dimensional case and finally points out some problems in applying SPA for MMSE signal estimation.

2.1 Introduction to the model and the estimator

The mixing model under consideration is

$$\mathbf{y} = a\mathbf{x}_1 + b\mathbf{x}_2 \ , \ a > 0, \ b > 0 \tag{2.1}$$

where the scalars a and b are referred to as mixing coefficients. The observations are contained in $\mathbf{y}$ and $\mathbf{x}_1$ and $\mathbf{x}_2$ denote the two sources which are mixed together. The model is a linear underdetermined noise-free two-in-one model. In the following it is assumed that the mixing coefficients are known. It's the estimation of the sources that is under consideration. To be specific, it is the MMSE estimator of the sources that is under consideration. Let the M-variate RV $\mathbf{x}_k = [x_{k,1}, \ldots, x_{k,M}]^T$ have PDF $p_k(\mathbf{x})$ then the MMSE estimator for $\mathbf{x}_1$ is given by the following weighted convolution integral

$$E\{\mathbf{x}_1|\mathbf{y}\} = \frac{1}{p(\mathbf{y})}\frac{1}{b}\int d\mathbf{x} \ \mathbf{x} \ p_1(\mathbf{x})\, p_2\left(\frac{\mathbf{y} - a\mathbf{x}}{b}\right) \tag{2.2}$$

or alternatively by

$$E\{\mathbf{x}_1|\mathbf{y}\} = \left[\frac{\partial}{\partial \mathbf{h}} \ln \int d\mathbf{x}\; p_1(\mathbf{x})\, p_2\left(\frac{\mathbf{y}-a\mathbf{x}}{b}\right) \exp(\mathbf{h}^T\mathbf{x}) \frac{1}{b}\right]\Bigg|_{\mathbf{h}=0} \tag{2.3}$$

These integrals are in general intractable except for a few special cases. For univariate RVs the integrals can be computed for various sparse distributions such as the Laplace and Cauchy distributions. For multivariate distributions the integrals can be computed analytically to obtain closed form expressions for e.g. Gaussian RVs or RVs having Gaussian mixture distributions. The integral in (2.2) can be reformulated into a complex contour integral where the integrand is derived from the Laplace transforms of $p_1(\mathbf{x})$ and $p_2(\mathbf{x})$. Let multivariate RV $x_k = [x_{k,1}, \ldots, x_{k,M}]^T$ have MGF $\mu_k(\lambda)$ where the Region Of Analyticity (ROA) of the MGF is R_k in the multidimensional complex $\boldsymbol{\lambda}$-space and $k = 1, 2$. Let contour C_k lie in ROA R_k for $k = 1, 2$. The estimator is then given by

$$E\{\mathbf{x}_1|\mathbf{y}\} = \frac{1}{(i\,2\pi)^M p(\mathbf{y})} \int_C d\mathbf{z}\; \exp(-\mathbf{y}^T\mathbf{z})\; \mu_1'(a\mathbf{z})\, \mu_2\,(b\mathbf{z}) \tag{2.4}$$

The derivation is found in appendix A. One benefit of rewriting (2.2) into (2.4) is connected to properties of integral transforms of RVs. As argued in [37], then algebraic operations on RVs are many times most conveniently handled by the use of integral transforms. For instance, if the RV $\mathbf{x}_k$ can be written as a linear combination of independent RVs then the Laplace transform of the PDF of $\mathbf{x}_k$ is the product of the Laplace transforms of the PDFs of the RVs in the linear combination. Another potential extension is for the case that the sources have models on the form $\mathbf{x}_k = g_k\mathbf{s}_k$, where g_k is a univariate positive RV independent of $\mathbf{s}_k$. For models of this form the mixing model in (2.1) turns into a sum of products of RVs. Products between RVs are conveniently handled using Mellin transforms if they are available, however two issues limit the usefulness of using Mellin transforms. The Mellin transform is defined for non-negative RVs. It is possible to work with RVs not everywhere positive but that gets quite involved rather quickly (example of computing the Mellin transform of a bivariate RV not everywhere positive is given in [37] page 153). For use in (2.4) the resultant Mellin transforms will have to be transformed to Laplace transforms which in the univariate case involves a complex contour integral (details are found in Appendix C in [37]). The relation between the Mellin transform and the bilaterial Laplace transform for the multidimensional case is given in [8] page 197.

The situation is thus that two different integrals for the MMSE estimator are at hand, one via PDFs and the other via MGFs. SPAs can be developed in either domain. As will be explained in sections 2.2.3 and 2.2.4 then close SPs have implications when applying the SPA. Maybe the PDF integral won't have close SPs, or maybe close SPs will happen at different values of the parameters [31].

Although the PDF integral starts out along the real axis, that contour can be modified and extended into another complex plane, if found fruitful. There is a chance that one would get two different SPAs, in general. This idea is not pursued in this chapter.

2.2 SPA

The history of SPA goes back almost two centuries. The SPA was used for the first time in statistics in a 1954 paper by Daniels [9] but [9] ascribes the physicist Peter Debye for first having used the method of steepest descent systematically in 1909 for studying Bessel functions. However, the authors in [33] point out that Debye himself borrowed the idea of the method from a 1863 unpublished paper of Bernhard Riemann. The paper further shows that elements of the method was used even earlier in 1827 by Cauchy and that a Russian Mathematician P. A. Nekrasov had given a very general and detailed exposition of the method 25 years before the work of Debye.

In [9] it is shown how the SPA represents the leading term in a proper asymptotic expansion by using the method of steepest descent in which the path of integration is the curve of steepest descent through an appropriate saddlepoint. The method of steepest descent, also known as the saddle-point method, is a natural development of Laplace's method applied to the asymptotic estimate of integrals of analytic functions [33]. As such, this section begins by briefly reviewing the principles in Laplace's method (Laplace approximation). Thereafter some important fundamental properties of analytic functions are enumerated and briefly explained. Next, the concepts of SPs and Monkey saddles will be explained and finally in section 2.2.4 an explanation of the SPA is given.

2.2.1 Laplace approximation

Let's say that the following (real) integral poses a problem

$$F = \int f(x)dx \tag{2.5}$$

where $f(x)$ is a positive function, $f(x) > 0$. One approach would be to expand the integrand in a Taylor series truncated to the first few terms and integrate. A slightly different approach will be used here. The integral is written in the following way

$$F = \int e^{\phi(x)}dx \tag{2.6}$$

where $\phi(x) = \ln f(x)$. The point is to expand the exponent, $\phi(x)$, in a truncated Taylor series. The Taylor expansion of $\phi(x)$ around a point x_0 is

$$\phi(x) \approx \phi(x_0) + \phi'(x_0)(x - x_0) + \frac{1}{2}\phi''(x_0)(x - x_0)^2 \tag{2.7}$$

A key point now is to choose x_0 wisely. The point x_0 is chosen such that $\phi'(x_0) = 0$ (it is assumed that $\phi''(x_0) \neq 0$). This means that (2.7) reduces to

$$\phi(x) \approx \phi(x_0) + \frac{1}{2}\phi''(x_0)(x - x_0)^2 \tag{2.8}$$

Inserting (2.8) into (2.6) it is clear that the integral is over a gaussian kernel with mean x_0 and variance $-1/\phi''(x_0)$ provided that x_0 is a maximum of the function so that the second derivative evaluated in x_0 is negative. Thus,

$$\begin{aligned} F &= \int e^{\phi(x)} dx && (2.9)\\ &\simeq \int e^{\phi(x_0)+\frac{1}{2}\phi''(x_0)(x-x_0)^2} dx && (2.10)\\ &= e^{\phi(x_0)} \left(-\frac{2\pi}{\phi''(x_0)}\right)^{1/2} && (2.11) \end{aligned}$$

This approximation is for instance useful for normalizing a PDF [25]. There is a two-page chapter about this approximation in [25] called 'Laplace's Method'. There it is written "Physicists also call this widely-used approximation the saddle-point approximation.". In the view of this chapter it may be misleading to call Laplace's Method for saddlepoint approximation. In [25] the generalization of (2.11) to a function of many variables is shown. As an aside, [33] notes that originally Laplace estimated an integral on the form

$$F_n = \int_a^b f^n(x)g(x)dx = \int_a^b e^{nu(x)}g(x)dx, \quad f(x) > 0, \tag{2.12}$$

as $n \to \infty$. Even though this integrand is different from the integrand in (2.5) then the approximation principle is the same.

The same type of approximation can also be used for approximating a function that is written as an integral [19]

$$f(x) = \int m(x,t)dt = \int e^{k(x,t)}dt \tag{2.13}$$

where $k(x,t) = \ln m(x,t)$. Expanding $k(x,t)$ around a stationary point, $\hat{t}(x)$,

gives

$$f(x) \simeq \int e^{k(x,\hat{t}(x))+\frac{(t-\hat{t}(x))^2}{2}\left.\frac{\partial^2 k(x,t)}{\partial t^2}\right|_{\hat{t}(x)}} dt \tag{2.14}$$

$$= e^{k(x,\hat{t}(x))}\left(-\frac{2\pi}{\left.\frac{\partial^2 k(x,t)}{\partial t^2}\right|_{\hat{t}(x)}}\right)^{1/2} \tag{2.15}$$

The key thing to note about this approximation is that the stationary point $\hat{t}(x)$ depends on x so that for each x there is a new integrated Taylor series expansion. The stationary point $\hat{t}(x)$ is a maximum and the second-order partial derivative satisfies $\partial^2 k(x,t)/\partial t^2 < 0$. As also noted in [19] then because the integral is being centralized there is hope that the approximation is 'accurate' and the price to pay is that for each x the stationary point $\hat{t}(x)$ must be found and the functions $k(x,t)$ and $\partial^2 k(x,t)/\partial t^2$ must be evaluated at that point.

2.2.2 Results from Calculus of Complex Variables

The concept of an analytic function is important for the understanding and application of the SPA. This section briefly summarizes a few of the most central properties of analytic functions and explains the role of the properties in the context of SPA. It will appear that analyticity of a function imposes some quite strong restrictions on the function. Two quick definitions taken from [7] are "An entire function is a function that is analytic at each point in the entire finite plane. Since the derivative of a polynomial exists everywhere, it follows that every polynomial is an analytic function." and "A Singular point of a function f is a point where f fails to be analytic."

- An analytic function obeys the Cauchy-Riemann equations

Let f be a function of a complex variable $z = x + iy$ such that $f(x + iy) = u(x,y) + iv(x,y)$. Also let u_x and u_y denote the first-order derivatives with respect to x and y of the function u. Then the Cauchy-Riemann equations can be written as

$$u_x = v_y, \quad u_y = -v_x \tag{2.16}$$

- If a function $f(z)$ is analytic in a Domain D, then its component functions u and v are harmonic in D.

That the component functions are harmonic means that they obey Laplace's equation, that is,

$$u_{xx} + u_{yy} = 0 \quad \text{and} \quad v_{xx} + v_{yy} = 0 \tag{2.17}$$

This property of analytic functions can be derived directly from the Cauchy-Riemann equations and implies that stationary points (points where $f(z)/dz = 0$) where the second-order partial derivatives are non-zero are SPs of u and v because the second-order partial derivatives have opposite signs.

- For an analytic function the level curves of u and v are orthogonal

If $f(z)$ is analytic then the families of level curves $u(x, y) = c_1$ and $v(x, y) = c_2$, where c_1 and c_2 are arbitrary real constants, are orthogonal. This result follows directly from the Cauchy-Riemann equations. This property implies that when following a Path Of Steepest Descent (POSD) of the real component u then on the same path the imaginary component, v, is constant. The orthogonality of the level curves of u and v is illustrated in figures 2.1 to 2.3 for the case of $f(z) = z^2$. Level curves for the component functions through a stationary point of $f(z)$ are not orthogonal. For instance, the curves $u(x, y) = 0$ and $v(x, y) = 0$ for the z^2 example intersect at the origin but are not orthogonal to each other.

- The path of integration can be changed for an analytic function

This property is also referred to as the principle of deformation of path. It says that the original contour of integration, C, can be deformed into (an equivalent) contour C_1 as long as only points where the integrand is analytic are passed and the value of the contour integral remains the same. A physical interpretation of an equivalent contour can be given [31]: At every singularity (poles, essential stationary points, branch points) in the (horizontal) complex plane, erect a tall stick. Consider C to be an infinitely flexible string lying on the complex plane. If the string is moved around arbitrarily in the plane, but never lifting it over a stick , the new contour C_1 is equivalent to C. That is, one will get the same value for the integral, using C or C_1.

In approximating a contour integral it is often of interest to deform the contour of integration to go through a SP of the integrand. It is for instance often desired to deform the original contour into a new equivalent contour that goes through a useful SP of the integrand because the neighborhood around the SP will provide a major contribution to the value of the integral. If numerical evaluation of a contour integral is performed then it is very desirable to deform the original contour such that integration along the POSDs out of a SP is performed because

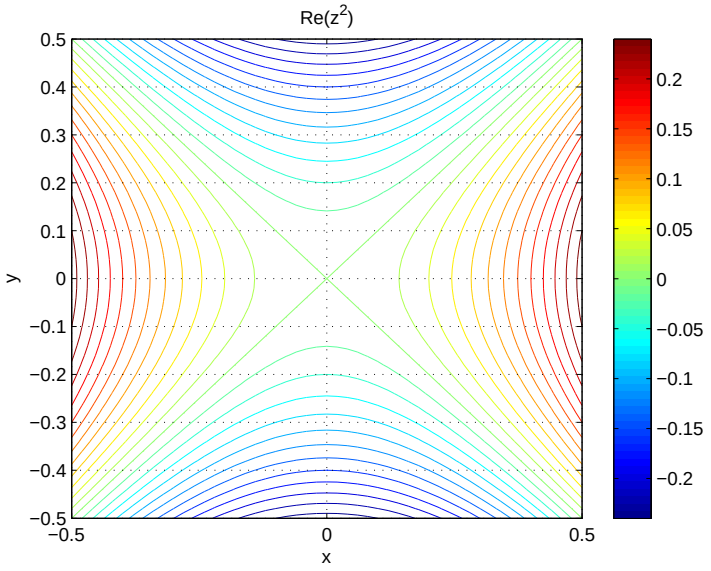

Figure 2.1: Level curves of real part of $f(z) = z^2$ ($x^2 - y^2 = c_1$).

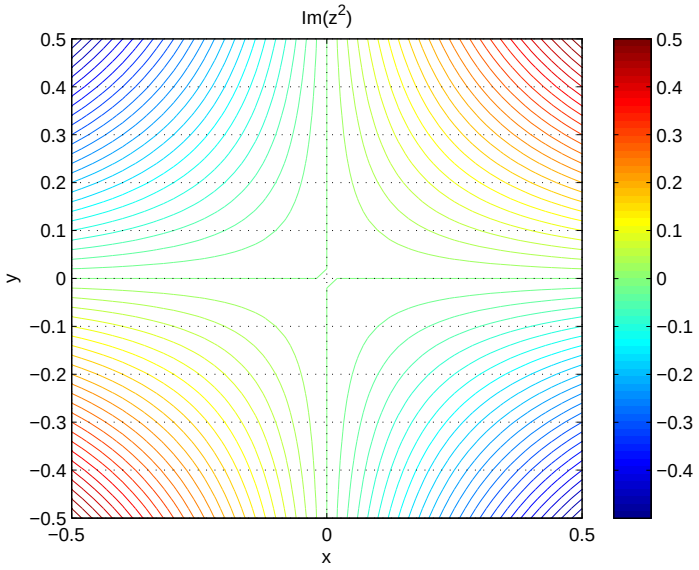

Figure 2.2: Level curves of imaginary part of $f(z) = z^2$ ($xy = c_2$).

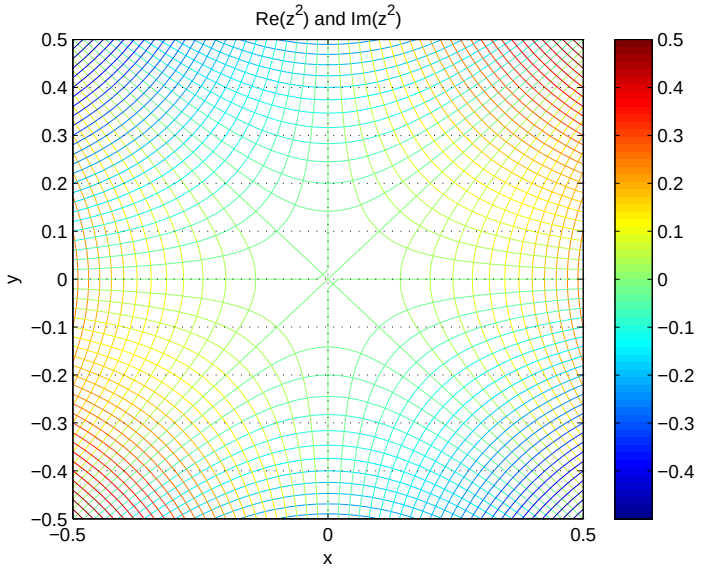

Figure 2.3: Level curves of both real and imaginary part of $f(z) = z^2$.

the magnitude of the integrand rapidly goes to zero along this path (there are no oscillations) and will give the most accurate answer for the integral.

- An analytic function may be subjected to analytic continuation thereby extending the domain of definition of the analytic function

Generally, even if an integrand does not have a SP in a desired region, it may be possible to use analytic continuation to extend the integrand outside its ROA, where there could very well be a SP. Complexities can arise with multivalued functions and the presence of zeros of the function $f(z)$, which, unfortunately, act like "black holes" to paths of steepest descent [31].

2.2.3 Saddlepoints

Consider a general contour integral

$$I = \int_C f(z)\, dz = \int_C \exp\left(\, \phi(z) \,\right)\, dz \tag{2.18}$$

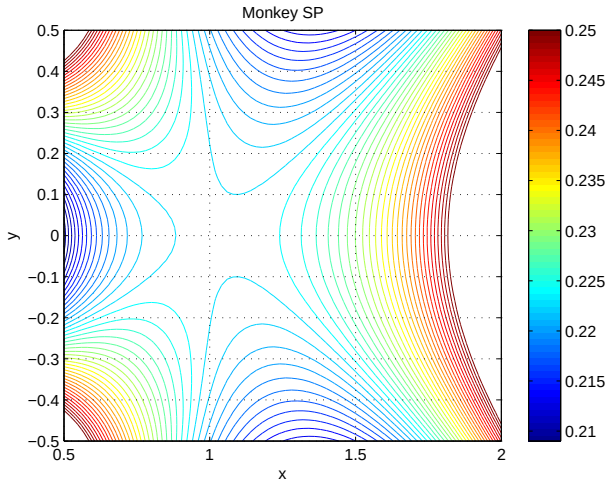

Figure 2.4: Illustration of Monkey SP for the integrand $f(z) = z\exp(-2\,q\,z + z^2/2)$ when $q = 1$.

where $\phi(z) = \ln f(z)$. The real and imaginary parts of the exponent, $\phi(z)$ determines the magnitude and phase of the integrand, respectively. This means that the location of SPs of $\phi(z)$ specify the location of SPs of the magnitude of the integrand, $f(z)$. The location of SPs of the magnitude of the integrand are thus found by solving $\phi'(z) = f'(z)/f(z) = 0$. As simple examples it can easily be verified that $f(z) = z^2$ does not have any SPs whereas $f(z) = \exp\left(z^2\right)$ has a SP at the origin.

If at a location not only the first-order derivative of $\phi(z)$ vanishes but also the second-order derivative vanishes this indicates the location of a monkey SP. When two SPs coalesce the second-order derivative of $\phi(z)$ vanishes. The case when two SPs are close or coalesce have implications in performing SPA, this will be explained further in section 2.2.4. More information on SPs can be found in [6] section 7.1. Illustrations of a monkey SP and two close SPs are shown in figures 2.4 and 2.5, respectively.

2.2.4 Integration Through a SP

This section is based entirely on emails from Albert H. Nuttall [31] (only slightly edited).

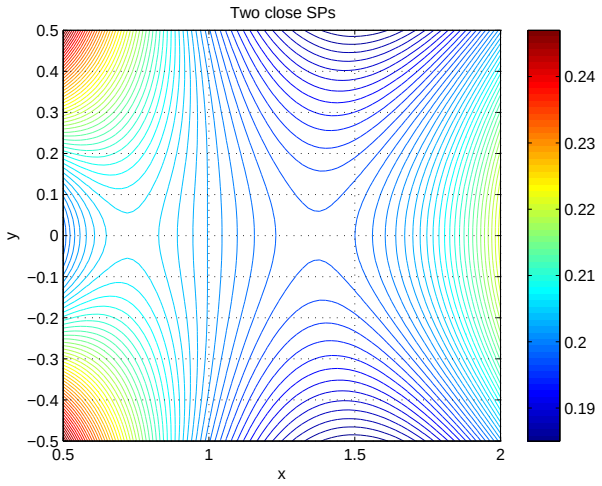

Figure 2.5: Illustration of two close SPs for the integrand $f(z) = z\exp(-2\,q\,z + z^2/2)$ when $q = 1.05$.

That a SP makes a major contribution to a contour integral can be illustrated using the following physical explanation. Think of the vertical Bromwich contour as being a road trip through the mountains. You see a mountain range ahead and you want to minimize the strain on your car. Scan the horizon and head for the lowest point P on the mountain range. As you cross over this point P, the road drops off, both behind you and in front of you. On the other hand, the mountain height is larger on your right and left, as you pass through P. This is similar to passing over the saddle of a horse, perpendicular to the direction that the horse is facing. Integration is a summation process. If you can find a contour on which the maximum magnitude of the integrand is as small as possible, say the POSD through a SP, you will not have to deal with oscillations in the integrand (remember that the level curves of the real and imaginary parts of an analytic function are orthogonal meaning that on the POSD the phase is constant and therefore contributions along the POSD are in-phase). This contour yields the most accurate numerical answer for the integral. But if you pass through a point higher in magnitude, then the integrand must develop oscillations somewhere along this alternative path because the value of the integral is independent of the path taken (within the limits of the ROA). So, large positive values at the passover point must be cancelled by large negative values somewhere else. A nice property of the POSD through a SP, in addition to the most rapid decay of the magnitude of the integrand, is that the angle of the integrand remains absolutely constant on the POSD. That is, there are no

oscillations in the complex values of the integrand. It is therefore of interest to move the contour of integration to go through a SP of the integrand.

Moving the original contour, so as to pass through a SP of the integrand, does not in itself constitute an approximation. The new integral will give exactly the same result as the original integral. It is only when we decide to approximate the integrand by a Gaussian function, as we go through the mountain pass, that the SPA is utilized. In fact, the shape of $|f(z)|$ in the neighborhood of a SP is not Gaussian, but is rather like a saddle. It is only the behavior of $|f(z)|$ on the contour of steepest descent that is Gaussian-like. A Gaussian mountain is placed at the peak of the magnitude of the integrand, and the curvatures at the SP are made to match. For complex SPs, all the same comments apply. We may have to use two SPs to connect the two valleys at $\pm i\infty$ and then we have to evaluate two SPAs as we pass over both mountain passes. Alternatively, the integral passing through the two SPs is still exact until we make the Gaussian approximations to the integrands as they go through the SPs themselves.

If one wants a very accurate numerical evaluation of a contour integral (which cannot be done analytically), the POSD is very attractive because the magnitude of the integrand, $|f(z)|$, decays the fastest on such a path, thereby allowing for a quick termination of the contour when $|f(z)|$ has gone below some desired tolerance level, relative to its value at the SP. And one can get a very accurate numerical value for the integral. Of course, one doesn't have to use the POSD. According to Cauchy's theorem, any equivalent path will give the same numerical result for the integral, if carried out to $\pm i\infty$. But the POSD allows for the quickest termination of the contour, because all numerical integrations must stop at a finite limit. On the other hand, if one is willing to settle for a simple approximate result, one can resort to a SPA, which just mimics $f(z)$ near the SP. To get this SPA, the POSD, as it passes through the SP, is the initial pertinent path of integration. However, the precise detailed POSD is not of interest or used; only the initial direction of the POSD near the SP is relevant, as far as the SPA is concerned. To be more precise, $|f(z)|$ is not necessarily Gaussian-like as we depart from the SP on the POSD. However, we compute $\phi(z)$, $\phi'(z)$, $\phi''(z)$ at the SP and pretend that $|f(z)|$ is Gaussian all the way out to infinity, so that we can do the resulting integral analytically. Then, sometimes, we also compute ϕ''' and ϕ'''' and put in a first-order correction term. In both approaches, the extension is made all the way out to infinity, to get a closed-form result.

For two close SP locations, z_1 and z_2, the simple Gaussian approximations will not suffice, and a cubic approximation may then be used. Again, until the approximation near the peak(s) is made, the integral on the modified contour is still exact, no matter how close the SPs are or how many there are. Along the POSD leading from the SP, the shape of $|f(z)|$ behaves like the Gaussian

function $\exp(-b\,d^2)$, where d is the distance away from the SP. With that in mind, and with a contour that passes through 2 SPs, if the Gaussian function approximation about SP_1 has decayed significantly by the distance that SP_2 is reached, then a decent approximation can be expected from the two SPAs. But if the Gaussian function approximation about SP1 has not decayed significantly by the distance to SP_2, then one have to switch to a cubic approximation for the argument of the exponential approximation near the two close SPs. On the POSD out of one SP at z_1, a Gaussian approximation based on the behavior right at the peak location z_1 is made. Then, when we start integrating along that POSD out of z_1, before we can get to a region of small magnitude of the integrand, $|f(z)|$, we run into another POSD coming from another close SP at z_2, and we then have to ascend up that path toward the other SP at z_2. A Gaussian mountain is placed at the peak of the magnitude of the integrand, and the curvatures at the SP are made to match. These two mountains intersect and interfere with each other, and a different function of z must be used near these two close peaks, to approximate their interacting behavior. A quadratic approximation to $\phi'(z)$ does just that, by being zero at the two SPs; that behavior is perfect near the two close SPs. The integral of the quadratic function gives a cubic function of z that does a good job of approximating the true behavior of $\phi(z)$ in the neighborhood of the two close SPs, which are the major points and region of interest. We will then integrate on this approximate function to get our SPA.

When the two close SPs z_1, z_2 are complex conjugates, they are both locations of peaks of the magnitude of integrand $f(z)$. The contour C could be moved to pass through both of these points. But when the two close SPs are both real (and still close), C can pass through only one of these SPs, namely, the one which has a minimum of $f(z)$ as z is varied along the real axis (The other close real SP has a local maximum along the real axis). In the direction perpendicular to the real axis (i.e., along the Bromwich contour), the function $f(z)$ then has a maximum at the SP used. Once the replacement of $\phi(z)$ by the approximating cubic function of z is made, the exact crossing point of the real axis in the complex plane is irrelevant because the new function of z is entire and has no singularities anywhere. This allows for considerable freedom of path movement, the real important thing is which valleys do the ends of the contours end up in as the new path must end up in the same valley(s) as beforehand.

2.3 Laplace-Gauss Example

The intent of including this example is to give an impression of the difficulties in applying SPA for a multivariate example where both variables are not Gaussian.

The example could represent a speech enhancement scenario (see [17][18]) where speech is modeled by a multivariate Laplace distribution and noise is modeled by multivariate Gaussian distribution. In the end of this section a number of points will be mentioned that have to be understood before this example can be completed.

The zero-mean multivariate Laplace distribution, as computed from a multivariate scale mixture of Gaussians [14], is

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2}} \frac{2}{\lambda} \frac{K_{(d/2)-1}\left(\sqrt{q(\mathbf{x})\ 2/\lambda}\right)}{\left(\sqrt{q(\mathbf{x})\ 2/\lambda}\right)^{(d/2)-1}} \tag{2.19}$$

where d denotes the dimensionality of the RV $\mathbf{x}$, $K_m(x)$ denotes the modified Bessel function of the second kind and order m, λ denotes a positive scalar and $q(\mathbf{x}) = \mathbf{x}^T\ \mathbf{\Gamma}\ \mathbf{x}$, where $\mathbf{\Gamma}$ is a positive definite matrix with $\det \mathbf{\Gamma} = 1$. The covariance matrix of RV $\mathbf{x}$ is $\Sigma_{\mathbf{x}} = \lambda \mathbf{\Gamma}$. In (2.1), let RV $\mathbf{x}_1$ be Laplacian and RV $\mathbf{x}_2$ be Gaussian. The MGF of the multivariate Laplace distribution and its first-order derivative (see appendix C), is

$$\mu_{\mathbf{x}}(\boldsymbol{\omega}) = \frac{1}{1 - \frac{\lambda}{2}\,\boldsymbol{\omega}^T \mathbf{\Gamma} \boldsymbol{\omega}} \quad , \quad \mu'_{\mathbf{x}}(\boldsymbol{\omega}) = \frac{\lambda \mathbf{\Gamma} \boldsymbol{\omega}}{\left(1 - \frac{\lambda}{2}\,\boldsymbol{\omega}^T \mathbf{\Gamma} \boldsymbol{\omega}\right)^2} \tag{2.20}$$

Inserting (2.20) into (2.4) then

$$E\{\mathbf{x}_1|\mathbf{y}\} = \frac{1}{(i\,2\pi)^M p(\mathbf{y})} \int_C d\mathbf{z}\ e^{-\mathbf{y}^T \mathbf{z}}\ \frac{a\lambda_1 \mathbf{\Gamma}_1 \mathbf{z}}{\left(1 - \frac{a^2 \lambda_1}{2}\,\mathbf{z}^T \mathbf{\Gamma}_1 \mathbf{z}\right)^2}\ e^{\frac{1}{2} b^2 \mathbf{z}^T \mathbf{\Sigma} \mathbf{z}} \tag{2.21}$$

Substituting $\tilde{\mathbf{z}} = b\mathbf{\Sigma}^{1/2}\mathbf{z}$ in (2.21) yields

$$E\{\mathbf{x}_1|\mathbf{y}\} = \frac{a\lambda_1 \mathbf{\Gamma}_1}{b^2\,(i\,2\pi)^M p(\mathbf{y})} \int_C d\tilde{\mathbf{z}}\ e^{-\mathbf{p}^T \tilde{\mathbf{z}} + \frac{1}{2} \tilde{\mathbf{z}}^T \tilde{\mathbf{z}}}\ \frac{\tilde{\mathbf{z}}}{\left(1 - \tilde{\mathbf{z}}^T \mathbf{A} \tilde{\mathbf{z}}\right)^2} \tag{2.22}$$

where

$$\mathbf{p}^T = \mathbf{y}^T \mathbf{\Sigma}^{-1/2} \frac{1}{b} \quad , \quad \mathbf{A} = \frac{a^2 \lambda_1}{b^2\,2} \left(\mathbf{\Sigma}^{-1/2}\right)^T \mathbf{\Gamma}_1 \mathbf{\Sigma}^{-1/2} \tag{2.23}$$

The fundamental integral is then

$$I = \frac{1}{(i\,2\pi)^M} \int_C d\mathbf{z}\ e^{-\mathbf{p}^T \mathbf{z} + \frac{1}{2} \mathbf{z}^T \mathbf{z}}\ \frac{\mathbf{z}}{\left(1 - \mathbf{z}^T \mathbf{A} \mathbf{z}\right)^2} \tag{2.24}$$

It may not be possible to evaluate the integral for $M > 2$. For the particular case $M = 2$, we have

$$I_1 = \frac{1}{(i\,2\pi)^2} \int_{C_1} \int_{C_2} dz_1 dz_2\ z_1\ e^{-p_1 z_1 - p_2 z_2 + \frac{1}{2} z_1^2 + \frac{1}{2} z_2^2}\ \frac{1}{\left(1 - a_{11} z_1^2 - a_{22} z_2^2 - 2a_{12} z_1 z_2\right)^2} \tag{2.25}$$

It may be possible to evaluate this integral analytically (see e.g. section 18 in [1]) using a multidimensional residue theorem. Solving for the SPs the Matlab function `solve` returns 8 SPs. In order to complete this example, the boundary on allowed values of the $\mathbf{z}$ vector in the 2-dimensional $\mathbf{z}$-plane need to be established. It also needs to be established which of the 8 SPs is the useful SP (possibly 2 complex conjugate SPs). It also needs to be determined if there are close SPs, and if so, how a SPA for two close SPs for the multivariate case can be developed. It is not straight forward to extend the results in the paper in appendix E to this example and it seems much harder to apply SPA for MMSE estimator approximation for multivariate problems.

2.4 Discussion

For the univariate case it has been demonstrated in two examples in the paper found in appendix E how SPA represents a viable method for approximating the MMSE estimator when using MGFs. Excellent approximations were obtained. It was also demonstrated that when two SPs are close the normal SPA based on a single SP becomes very inaccurate and how a SPA based on two close SPAs should be used instead to obtain accurate approximations for this case. All in all, the paper in appendix E provides a quite complete account of applying SPA for MMSE estimator approximation for the univariate case and as the examples in the paper illustrate it is relative straight forward to apply the results.

In extending the ideas to the M-variate case the situation gets rather involved rather quickly. Finding a useful SP may be difficult. In general, one has to solve a set of coupled non-linear equations, numerically. When SPA is used for obtaining tail probabilities the search can be restricted to finding a real unique SP in the ROA, however, such a restriction is, in general, not valid when SPA is used for MMSE approximation. A SPA for coalescing or close SPs in the multidimensional case may also be needed. Therefore, SPA does not look useful for approximating a MMSE estimator in the multidimensional case, especially not for higher dimensions.

Chapter 3

Particle Filtering for MMSE Estimation

The purpose of this chapter is to demystify particle filtering (PF) and demonstrate in a couple of examples how PF can be used. This chapter also serves as an introduction to the two papers applying PF found in appendices F and G. The intent is to describe the fundamental ideas behind PF and not focus on theoretical aspects.

PF is a technique used for approximate inference that is suitable for online applications. In the literature PF is also referred to as Sequential Monte Carlo. From a Bayesian point of view a fundamental idea of PF is that of representing a posterior PDF of interest by a number of samples where we can think of a sample as a number that is returned by a Random Number Generator (RNG) with a distribution equal to that of the posterior PDF. Herein lies an approximation but the more samples that are used to represent the posterior PDF the better the approximation becomes and in the limit the representation becomes exact (see e.g. [12] ch. 2). Given a number of samples from a posterior PDF of interest it is possible to estimate properties of the posterior PDF such as for instance the location of a maximum or the mean value. The mean value is the main quantity of interest in Minimum Mean-Square Error (MMSE) estimation which will be considered later in the examples. The idea is thus to replace the functional representation of the posterior PDF by samples. In [36], it is written "..we note the essential duality between a sample and the density (distribution)

from which it is generated. Clearly, the density generates the sample; conversely, given a sample we can approximately recreate the density (as a histogram, a kernel density estimate, an empirical cdf, or whatever)."

Another key idea in PF is to obtain samples from a posterior PDF that are "important" such that useful approximations can be made by a manageable number of samples. This is done by attaching a weight to each sample which tells how important a particular sample is. Samples with low weight can then be discarded and replaced by samples with higher weight in a process referred to as *resampling*. A *particle* consists of a sample and its weight. It is described later how the weights are computed. The weights are needed because samples will be produced by a RNG which is considerably simpler than the desired target RNG. By attaching weights to the samples generated by the simpler RNG these weighted samples constitutes a representation of the posterior PDF that is just as valid as samples drawn from the complex target RNG. The combination of using a simple RNG and weighted samples is the concept behind Importance Sampling (IS) which plays a fundamental role in PF.

A crucial idea behind PF is that of propagating the particles sequentially in time making PF suitable for online applications. PF processes a sequence one sample at a time by propagating particles (samples and weights) recursively. The process of propagating particles recursively in time is called Sequential Importance Sampling (SIS). The combination of SIS and the above mentioned resampling constitutes the main body of a particle filter.

In section 3.1 a class of useful models will be introduced and it will appear that PF in this class of models is very convenient. The following sections go into details of PF by first briefly explaining the principle behind the Monte Carlo method in section 3.2, then describing the idea of IS in section 3.3 and outlining the procedure of SIS in section 3.4. Next, the purpose of resampling is described in section 3.5, followed by three examples in sections 3.6-3.8 that all have some relevance to signal processing for hearing aids. Finally this chapter is concluded with a discussion in section 3.9.

3.1 Introduction to the models

The starting point of a class of useful and convenient discrete system models is the assumption that the z-transform representation of the model, the system function $H(z)$, is a rational function of polynomials in z;

$$H(z) = \frac{Y(z)}{X(z)}, \tag{3.1}$$

where $Y(z)$ and $X(z)$ are polynomials in z. Most often, however, $H(z)$ is expressed as a ratio of polynomials in z^{-1}, see e.g. [32] ch. 6; i.e.,

$$H(z) = \frac{\sum_{k=0}^{M} b_k \, z^{-k}}{1 - \sum_{k=1}^{N} a_k \, z^{-k}} \ , \tag{3.2}$$

with the corresponding difference equation

$$y_n = \sum_{k=1}^{N} a_k \, y_{n-k} + \sum_{k=0}^{M} b_k \, x_{n-k} \ . \tag{3.3}$$

If zero-mean white Gaussian noise with variance σ^2 is added to the right hand side of (3.3) the difference equation can be written in a direct form II matrix formulation with the following state and observation equations

$$\mathbf{w}_n = \mathbf{A}\,\mathbf{w}_{n-1} + \mathbf{B}\,\mathbf{x}_n \tag{3.4}$$

$$y_n = \mathbf{C}\,\mathbf{w}_n + \mathrm{D}\,d_n \quad , \quad d_n \overset{iid}{\sim} \mathcal{N}(0,1) \ , \tag{3.5}$$

where $\mathbf{w}_n = [\, w_n, \, \cdots, \, w_{n-N+1} \,]^T$ is a state vector, $\mathbf{x}_n = [\, x_n, \, \cdots, \, x_{n-N+1} \,]^T$ the input vector and

$$\mathbf{A} = \begin{bmatrix} & \mathbf{a}^T \\ \mathbf{I}_{(N-1)} & \mathbf{0}_{(N-1)\times 1} \end{bmatrix} \tag{3.6}$$

$$\mathbf{B} = \begin{bmatrix} 1 \\ \mathbf{0}_{(N-1)\times 1} \end{bmatrix} \tag{3.7}$$

$$\mathbf{C} = \begin{bmatrix} \mathbf{b}^T \end{bmatrix} \tag{3.8}$$

$$\mathrm{D} = \begin{bmatrix} \sigma \end{bmatrix} \tag{3.9}$$

where $\mathbf{a} = [\, a_1, \, \cdots, \, a_N \,]^T$ and $\mathbf{b} = [\, b_0, \, \cdots, \, b_M \,]^T$ are coefficient vectors. The matrix $\mathbf{I}_N$ denotes the eye matrix of dimensions $N \times N$ and $\mathbf{0}_{N\times M}$ denotes a zero matrix of dimensions $N \times M$. If the objective is to estimate the coefficient vectors $\mathbf{a}$ and $\mathbf{b}$ and prior knowledge of the system to model is available then it may be difficult to relate that knowledge to behavior of the coefficients in $\mathbf{a}$ and $\mathbf{b}$. That is, it may be difficult to make use of the prior knowledge when estimating $\mathbf{a}$ and $\mathbf{b}$. If, for instance, it is known that the system to model has a spectral peak in the frequency interval from say 1000 Hz to 2000 Hz then it is not obvious how to constrain the coefficients in $\mathbf{a}$ and $\mathbf{b}$ to take this information into account. However, as will be explained next then it is possible to find other, perhaps more useful, parameterizations of the system described by (3.4)-(3.9).

If information is available about the magnitude spectrum of the system then the roots of the polynomials $Y(z)$ and $X(z)$, referred to as zeros and poles respectively, play an important role. Given a set of zeros and poles it is relative

straightforward to visualize the shape of the magnitude spectrum, see e.g. [32] section 5.3 and [28]. It is also possible to roughly determine locations of zeros and poles from a sketch of a magnitude response. This will be illustrated with an example using babble noise ($f_s = 16$ kHz). A babble noise sound is analyzed in frames using a Hamming window of 25 ms duration and shifted in time in steps of 1.5 ms. For each frame a power spectrum is computed. The power spectra are averaged in intervals of 500 ms and a 10^{th} order LPC analysis is performed on each of the averaged power spectra. Finally, roots (poles) are computed from the denominator polynomial in the system function. The poles computed from one of the averaged power spectra are seen in the plot in figure 3.1. The magnitude and phase response of the corresponding all-pole system function are shown in figure 3.2. As can be seen in figure 3.1 then the poles occur in complex

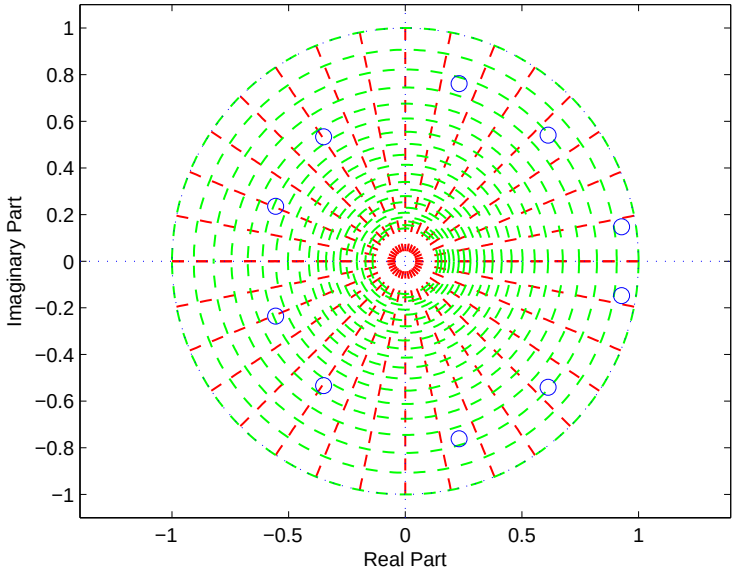

Figure 3.1: Figure shows unit circle in the complex z-plane and the location of poles obtained by a 10^{th} order LPC analysis of a smoothed babble power spectrum.

conjugate pairs, 5 in total, all inside the unit circle. Each complex conjugate pole pair forms a resonance whose frequency is determined by the angles of the poles and whose sharpness (bandwidth) is determined by the distance of the poles to the unit circle. There is 500 Hz between each of the concentric green circles and 500 Hz between each of the red lines coming radially out of the origin. Each peak in the magnitude spectrum in the upper plot in figure 3.2

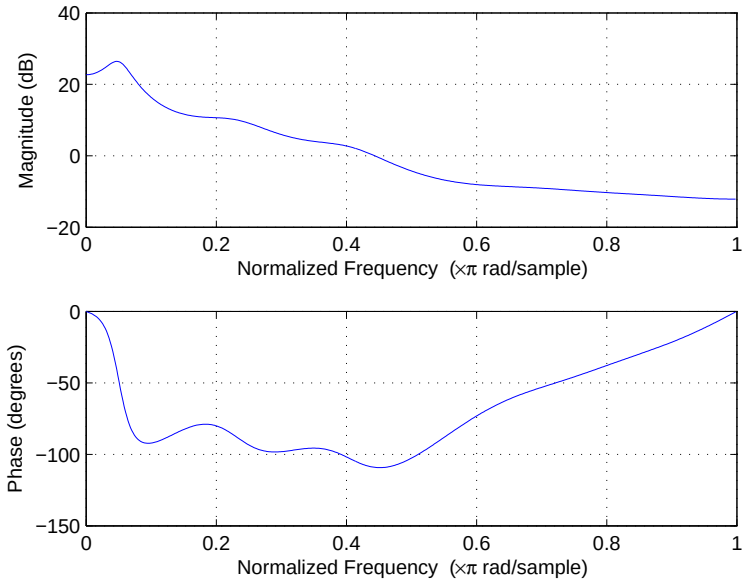

Figure 3.2: Magnitude and phase spectrum corresponding to a 10^{th}order all-pole model with pole locations as shown in figure 3.1.

can be ascribed to a complex conjugate pole pair in figure 3.1. In figure 3.3 the same poles are shown in a plot where the pole regions for speech are also shown. The pole regions for speech have been established by minimum and maximum frequencies and bandwidths for the first four formants. It is apparent that by using poles, represented by their real and imaginary parts or by radii and angles, instead of the corresponding AR coefficients then the relation to the magnitude spectrum is more intuitive and easier to visualize. A complex conjugate pole pair can also be represented uniquely by a center frequency and a bandwidth parameter which may be the most convenient and intuitive representation to make use of if information on the magnitude spectrum involves location and sharpness of peaks and valleys. There are several additional parameterizations that may be useful for a particular problem e.g. reflection coefficients, log-area ratio coefficients or line-spectral frequency coefficients.

From the above description it is apparent that complex conjugate roots (zeros or poles) play an important role in re-parameterizing the coefficient vectors **a** and **b**. When the coefficients in **a** and **b** are real, complex-valued roots come in conjugate pairs. By factorizing the numerator and denominator polynomials in (3.2) the system function can be expressed on the form, see e.g. [32] p. 303,

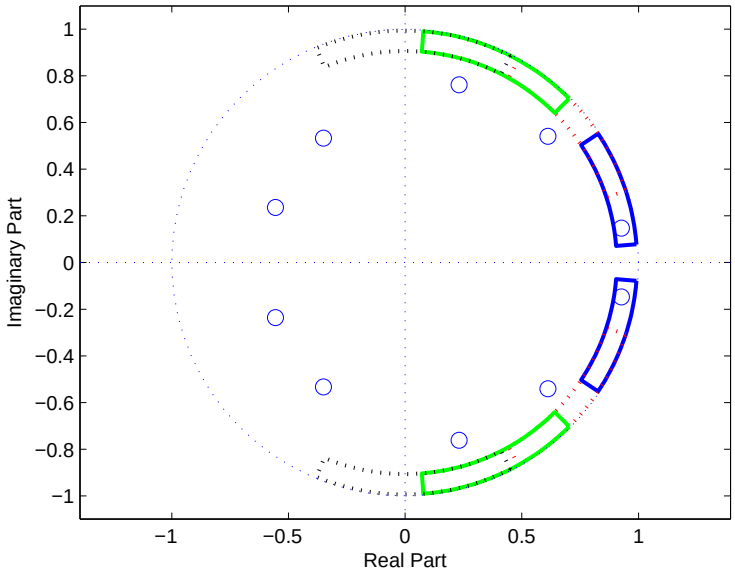

Figure 3.3: Figure shows the unit circle and the poles from figure 3.1. The figure also shows the pole regions for speech corresponding to the first four formants.

(by a theorem known as the fundamental theorem of algebra [28] p. 24)

$$H(z) = A\frac{\prod_{k=1}^{M_1}(1-g_k\,z^{-1})\prod_{k=1}^{M_2}(1-h_k\,z^{-1})(1-h_k^*\,z^{-1})}{\prod_{k=1}^{N_1}(1-c_k\,z^{-1})\prod_{k=1}^{N_2}(1-d_k\,z^{-1})(1-d_k^*\,z^{-1})} \tag{3.10}$$

where $M = M_1 + 2M_2$ and $N = N_1 + 2N_2$. In this factorization there are M_2 pairs of complex conjugate zeros and N_2 pairs of complex conjugate poles. A pair of complex conjugate roots corresponds to a Second-Order Section (SOS) on the form

$$H_{SOS}(z) = 1 - a_1 z^{-1} - a_2 z^{-2} \tag{3.11}$$

where a_1 and a_2 are real coefficients. As an example, consider a pair of complex conjugate roots, z_1 and z_1^*. They can be related directly to the coefficients in (3.11) in the following way

$$H_{SOS}(z) = (1 - z_1 z^{-1})(1 - z_1^* z^{-1}) = 1 - (z_1 + z_1^*)z^{-1} + z_1 z_1^* z^{-2} \tag{3.12}$$

Thus,

$$a_1 = z_1 + z_1^* \quad , \quad a_2 = -z_1 z_1^*. \tag{3.13}$$

Given the zeros and poles of a system function it is now possible to compute the corresponding coefficients in **a** and **b** by first mapping the complex conjugate

roots to coefficients in a SOS on the form given by (3.11). Note that real roots correspond directly to the coefficient in a first-order section. Finally, the coefficient vectors corresponding to first and second-order sections are convolved. If the complex conjugate roots are represented in polar form as $z_1 = Re^{j\omega}$ and $z_1^* = Re^{-j\omega}$ and inserted in equation (3.12) one obtains

$$\begin{aligned} H_{SOS}(z) &= 1 - (Re^{j\omega} + Re^{-j\omega})z^{-1} + Re^{j\omega}Re^{-j\omega}z^{-2} && (3.14) \\ &= 1 - 2R\cos(\omega)z^{-1} + R^2z^{-2} \ . && (3.15) \end{aligned}$$

The SOS coefficients expressed in terms of the radius and angle of a complex conjugate root pair are thus

$$a_1 = 2R\cos(\omega) \quad , \quad a_2 = -R^2 \ . \tag{3.16}$$

Given a formant frequency, f, and formant bandwidth, b, (both in Hz) then the corresponding complex root is given by

$$z_1 = e^{-\pi\frac{b}{f_s}+j2\pi\frac{f}{f_s}} \ . \tag{3.17}$$

This relation is explained in [26] and [35] and is based on transforming from s-plane to z-plane. The mapping from f and b to the coefficients in (3.11) are thus given by

$$a_1 = 2e^{-\pi\frac{b}{f_s}}\cos\left(2\pi\frac{f}{f_s}\right) \quad , \quad a_2 = -e^{-2\pi\frac{b}{f_s}} \tag{3.18}$$

The mappings in (3.18) are non-linear. Reasonable approximations can be found for the mappings in (3.18) when modeling speech because the bandwidth of the resonances created in the vocal tract typically will be less than say 500 Hz. This means that $e^{-\pi\frac{b}{f_s}}$ and $e^{-2\pi\frac{b}{f_s}}$ may be approximated by first-order Taylor series expansions. That is

$$e^{-\pi\frac{b}{f_s}} \approx 1 - \pi\frac{b}{f_s} \quad , \quad e^{-2\pi\frac{b}{f_s}} \approx 1 - 2\pi\frac{b}{f_s} \ . \tag{3.19}$$

In figure 3.4 the accuracy of the approximations are shown for $f_s = 16$ kHz

To summarize, motivated by the usefulness of discrete linear time-invariant systems a linear state-space model has been introduced in (3.4)-(3.9). The model is parameterized by the coefficient vectors **a** and **b** and an observation noise variance. When estimating the coefficients in the **a** and **b** vectors it has been argued that different parameterizations may be more advantageous. The basis of these representations is the factorization of the corresponding numerator and denominator polynomials in terms of first and second-order sections. The coefficients in the second-order sections can be represented by a pair of complex conjugate roots. The roots can be specified by their real and imaginary parts

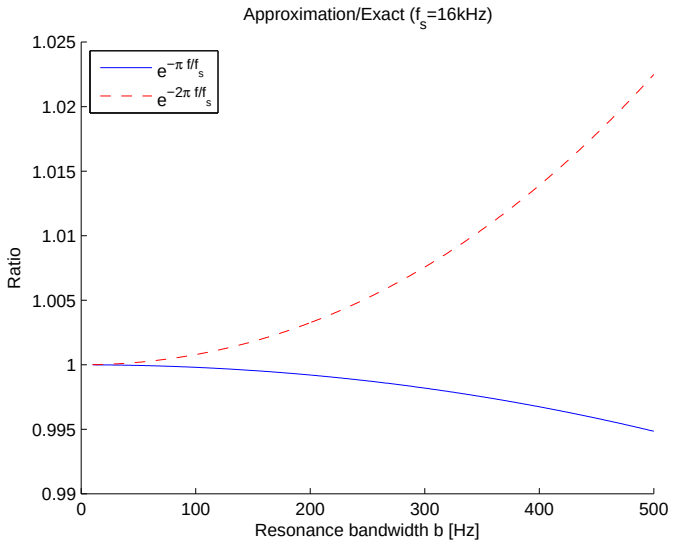

Figure 3.4: Figure shows that accuracy of the approximations in (3.19). The accuracy is presented by computing the ratio between the approximation and the expression to be approximated.

as in (3.13), by their radii and angles as in (3.16) or by center frequency and bandwidth parameters as in (3.18).

Many systems of interest are time-variant and to take this into account the parameters in the state-space model given by (3.4)-(3.9) have to be adapted over time. As such the parameters should have a time index. For instance, the vocal tract changes characteristics when producing speech. An often used model for the vocal tract is an all-pole model and speech is often modeled as the being the output of an all-pole model excited by white Gaussian noise. The state-space model in (3.4)-(3.9) can be modified slightly to form a time-varying all-pole model for speech to obtain

$$\mathbf{w}_n = \mathbf{A}_n\,\mathbf{w}_{n-1} + \mathbf{B}_n\,x_n \quad , \quad x_n \overset{iid}{\sim} \mathcal{N}(0,1) \tag{3.20}$$

$$y_n = \mathbf{C}\,\mathbf{w}_n + D_n\,d_n \quad , \quad d_n \overset{iid}{\sim} \mathcal{N}(0,1), \tag{3.21}$$

where

$$\mathbf{A}_n = \begin{bmatrix} & \mathbf{a}_n^T \\ \mathbf{I}_{(N-1)} & \mathbf{0}_{(N-1)\times 1} \end{bmatrix} \tag{3.22}$$

$$\mathbf{B}_n = \begin{bmatrix} \sigma_{x_n} \\ \mathbf{0}_{(N-1)\times 1} \end{bmatrix} \tag{3.23}$$

$$\mathbf{C} = \begin{bmatrix} 1 & \mathbf{0}_{1\times(N-1)} \end{bmatrix} \tag{3.24}$$

$$D_n = \begin{bmatrix} \sigma_{d_n} \end{bmatrix} \tag{3.25}$$

Noise reduction can be performed by MMSE estimation of the state-vector $\mathbf{w}_n$. MMSE estimates of the model parameters are also of interest. In the following sections, a method is described that can be used for approximate inference in the models specified by (3.4)-(3.9) and (3.20)-(3.25).

3.2 Monte Carlo

The basic idea behind the Monte Carlo method is to repeat an experiment a number of times and use the outcomes of the experiments to estimate a quantity of interest. For instance, the task of computing the probability that a given solitaire game will succeed may be very difficult. The Monte Carlo approach to estimate this probability is to play (simulate) a number of games (say $1,000,000$) and take the ratio between the number of successful games and the total number of games as the probability estimate. The same way probabilities of interest in poker or blackjack may be approximated; simulate many games and compute an estimate of the probability of winning given a particular setup.

The history of the Monte Carlo method is quite fascinating. It began in 1945 when physicists at Los Alamos were studying the feasibility of a thermonuclear weapon. It's beginning is also closely connected to the development of the first electrical computers. In [27] it is written "The spirit of Monte Carlo is best conveyed by an example discussed in von Neumann's letter to Richtmeyer. Consider a spherical core of fissionable material surrounded by a shell of tamper material. Assume some initial distribution of neutrons in space and in velocity but ignore radiative and hydrodynamic effects. The idea is to now follow the development of a large number of individual neutron chains as a consequence of scattering, absorption, fission, and escape....Thus, a genealogical history of an individual neutron is developed. The process is repeated for other neutrons until a statistically valid picture is generated."

For the case of performing MMSE estimation the Monte Carlo approach is to generate samples from the posterior PDF and average the value of the samples. However, although it sounds simple it is typically not realistic to draw samples directly from the posterior PDF. It is not as easy as simulating a game of poker. There are several ways to generate samples from a given distribution see e.g. [23] pp. $23-31$. Some of these methods are not very practical for high-dimensional problems or suitable for on-line applications. IS, however, is a promising method and is explained in the next section.

3.3 Importance Sampling

Importance sampling is a method for generating samples from a distribution. It works by generating samples from a simpler *proposal* or *importance distribution* and correcting the bias by attaching *importance weights* to the drawn samples. To motivate the use of IS we will introduce an example adapted from [23] p. $31 - 34$. The goal is to evaluate the integral

$$I = \int_{-1}^{1} \int_{-1}^{1} dx\, dy\ f(x, y)\ , \tag{3.26}$$

where

$$f(x, y) = 0.5e^{-90(x-0.5)^2-45(y+0.1)^4} + e^{-45(x+0.4)^2-60(y-0.5)^2}. \tag{3.27}$$

The function $f(x, y)$ is shown to the left in figure 3.5. The integral in (3.26)

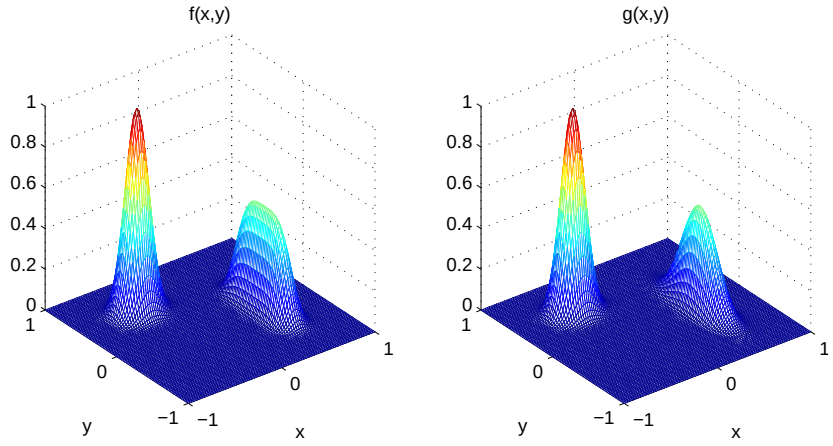

Figure 3.5: Plot to the left shows the function $f(x, y)$ and the plot to the right shows the importance distribution $g(x, y)$.

can be approximated using a Riemann sum by discretizing the domain $[-1, 1] \times [-1, 1]$ on a regular grid. Thus, the integral can be approximated as

$$\hat{I}_{Riemann} = \Delta x \Delta y \sum_i \sum_j f(x^i, y^j)\ , \tag{3.28}$$

where Δx and Δy are the spacings in the x and y directions, respectively. If a regular grid with a spacing of 0.02 is used in both the x and y directions then approximately 16 percent of the points are above the threshold 0.01. If only points above this threshold are used then it produces a 1% relative difference

in the result. A regular grid thus uses many computations on points where $f(x, y)$ is negligible. Another disadvantage of the regular grid approach is that the complexity of evaluating an integral scales exponentially with the number of dimensions. For instance, in the regular grid example 101 points is used on the interval $[-1, 1]$ on both the x and y-axis giving a total of $101^2 = 10201$ points. However, if the example is extended with a third z-axis (using same interval and grid spacing) we would have to use $101^3 = 1030301$ points. It is thus obvious that the regular grid approach becomes rather involved rather quickly as the number of dimensions increase.

A different approach is to uniformly sample the domain $[-1, 1] \times [-1, 1]$. The integral in (3.26) can be written

$$\begin{aligned} I &= \int_{-1}^{1}\int_{-1}^{1} dx\, dy\ f(x, y)\frac{p(x, y)}{p(x, y)} && (3.29)\\ &= \frac{1}{P}\int_{-1}^{1}\int_{-1}^{1} dx\, dy\ f(x, y)p(x, y) && (3.30)\\ &= \frac{1}{P}E_{p(x,y)}\{f(x, y)\}\ , && (3.31) \end{aligned}$$

where $P = p(x, y) = 1/4$ and it is understood that $f(x, y)$ is zero outside the domain $[-1, 1] \times [-1, 1]$. Because $p(x, y)$ is a PDF, an approximation to the expectation in (3.31) and thus the integral in (3.26) can be obtained by drawing N samples from $p(x, y)$ and approximate $p(x, y)$ by a sum of delta functions (with equal weighting $1/N$). The approximation based on uniform sampling thus becomes

$$\hat{I}_{Uniform} = \frac{1}{N\,P}\sum_{n} f(x^n, y^n)\ . \tag{3.32}$$

Figure 3.6 shows drawn samples for the case $N = 1000$. Because the approximation is based on a sample of random samples then a new value of $\hat{I}_{uniform}$ is obtained for each new experiment. That is, accuracy of the approximation has to be measured statistically. One way to measure the accuracy of the approximation is to perform K experiments and compute the mean and standard deviation of the K results. As is evident from figure 3.6 then uniform sampling is also very inefficient because many samples are placed in regions where $f(x, y)$ is negligible. The idea in IS is that instead of sampling the domain uniformly we could find a different distribution to sample from that is more likely to produce samples in the areas where $f(x, y)$ is non-negligible and thus improve efficiency of the samples significantly. Samples could be generated from a distribution on the form

$$g(x, y) \propto 0.5e^{-90(x-0.5)^2-45(y+0.1)^2} + e^{-45(x+0.4)^2-60(y-0.5)^2}\ , \tag{3.33}$$

with $(x, y) \in [-1, 1] \times [-1, 1]$. This corresponds to a truncated mixture of

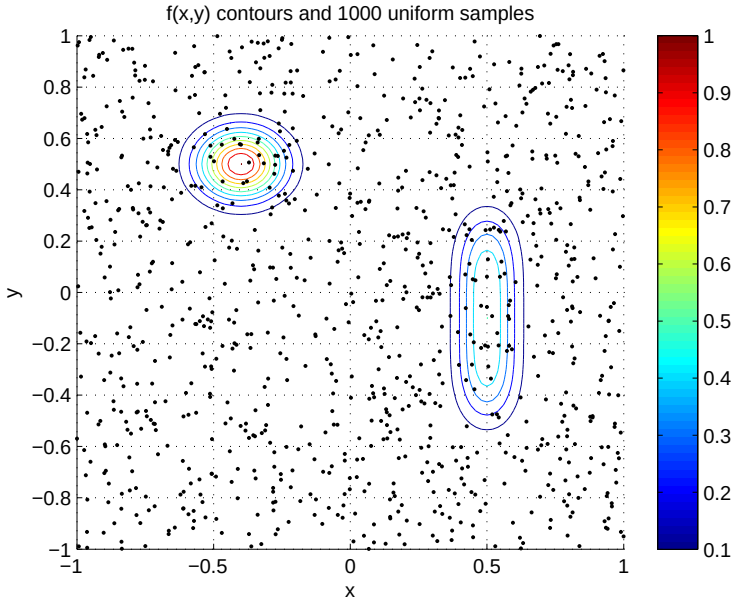

Figure 3.6: Figure shows contours of the integrand $f(x, y)$ and 1000 samples obtained by uniformly sampling the domain $[-1, 1] \times [-1, 1]$.

Gaussian distribution from which samples can be drawn, that is,

$$g(x,y) = 0.46\mathcal{N}\left(\begin{bmatrix} 0.5 \\ -0.1 \end{bmatrix}, \begin{bmatrix} \frac{1}{180} & 0 \\ 0 & \frac{1}{20} \end{bmatrix}\right) + 0.54\mathcal{N}\left(\begin{bmatrix} -0.4 \\ 0.5 \end{bmatrix}, \begin{bmatrix} \frac{1}{90} & 0 \\ 0 & \frac{1}{120} \end{bmatrix}\right) \tag{3.34}$$

This distribution is shown to the right in figure 3.5. When sampling from $g(x, y)$, samples will be located as shown in figure 3.7 where 1000 samples have been generated. Now, consider the integral in (3.26) one more time.

$$\begin{aligned} I &= \int_{-1}^{1}\int_{-1}^{1} dx\, dy\ g(x,y)\frac{f(x,y)}{g(x,y)} && (3.35)\\ &= \int_{-1}^{1}\int_{-1}^{1} dx\, dy\ g(x,y)w(x,y) && (3.36)\\ &= E_{g(x,y)}\{w(x,y)\}\ . && (3.37) \end{aligned}$$

The original integral has be reformulated into a statistical expectation of a *weight-function*, $w(x, y)$, with respect to the distribution $g(x, y)$. If samples are drawn from $g(x, y)$ the integral in (3.26) or equivalently the expectation in (3.36)

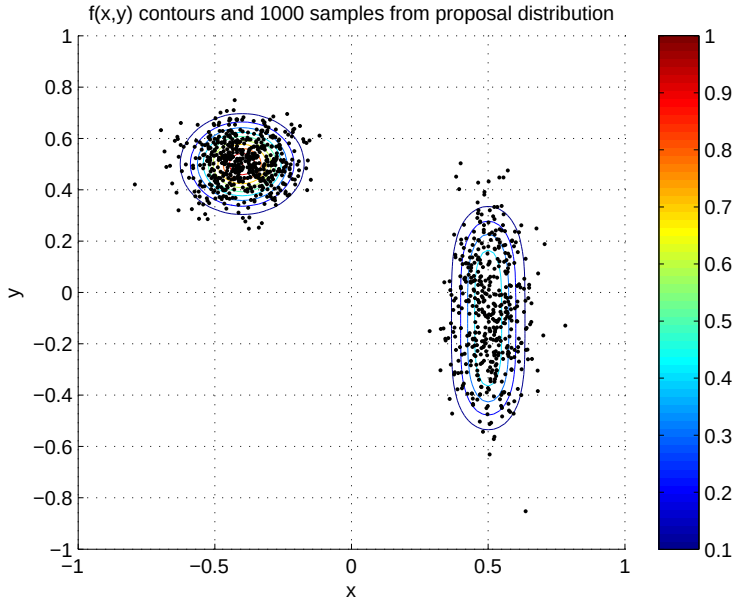

Figure 3.7: Figure shows contours of the integrand $f(x, y)$ and 1000 samples obtained by sampling from the importance distribution $g(x, y)$ in (3.34).

can be approximated using the following expression

$$\hat{I}_{Importance} = \frac{1}{N} \sum_{n} w(x^n, y^n) \ . \tag{3.38}$$

This approximation reduces to the approximation in (3.32) when $g(x, y)$ is a uniform distribution. The expression in (3.38) is the IS approach to approximate the integral in (3.26). It is clear that if a good proposal distribution $g(x, y)$ is used less resources are wasted on regions with low probability that do not contribute significantly to the value of the integral. In [23] p. 33 it is written "...the idea of biasing toward 'importance' regions of the sample space becomes essential for Monte Carlo computation with high-dimensional models... In high-dimensional problems, the region in which the target function is meaningfully nonzero compared with the whole space χ is just like a needle compared with a haystack." and in [11] they write "Choice of importance function is of course crucial and one obtains poor performance when the importance function is not well chosen."

In summary, as the example has illustrated, then IS represents an attractive approach to evaluate an integral as it makes efficient use of particles which is

important for practical use. It has also been indicated that IS is a method that can work for high-dimensional problems. The example also demonstrated how IS works by drawing samples from a simpler importance distribution and how a weight for each sample is computed.

As this chapter is about MMSE estimation the IS procedure is now repeated for a general MMSE-type integral. The pertinent integral is

$$I_{MMSE} = \int d\mathbf{x}\ \mathbf{x}\ p(\mathbf{x})\ . \tag{3.39}$$

If N independent samples are drawn from $p(\mathbf{x})$ then the integral in (3.39) can be approximated by

$$\hat{I}_{MMSE} = \frac{1}{N}\sum_i \mathbf{x}^i\ . \tag{3.40}$$

Generally, it is impossible to draw independent samples directly from the posterior distribution of interest. However, the integral in (3.39) can be written

$$I_{MMSE} = \int d\mathbf{x}\ \mathbf{x}\ \frac{p(\mathbf{x})}{\pi(\mathbf{x})}\pi(\mathbf{x})\ , \tag{3.41}$$

and if N independent samples are generated from the importance distribution, $\pi(\mathbf{x})$, then the integral can be approximated by

$$\hat{I}_{MMSE} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}^i\,\frac{p(\mathbf{x}^i)}{\pi(\mathbf{x}^i)} = \frac{1}{N}\sum_i \mathbf{x}^i\,w(\mathbf{x}^i)\ . \tag{3.42}$$

The approximation in (3.42) is dependent on the choice of importance distribution $\pi(\mathbf{x})$. If the vector $\mathbf{x}$ can be divided into two parts $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2]^T$ such that integration over $\mathbf{x}_1$ conditional on $\mathbf{x}_2$ can be performed analytically the integral in (3.41) can be subjected to the principle of *Rao-Blackwellization*. Rewriting the integral in (3.41) into

$$I_{MMSE} = \int d\mathbf{x}_1 \int d\mathbf{x}_2\ \ [\mathbf{x}_1, \mathbf{x}_2]\ \ p(\mathbf{x}_1|\mathbf{x}_2)\frac{p(\mathbf{x}_2)}{\pi(\mathbf{x}_2)}\pi(\mathbf{x}_2) \tag{3.43}$$

$$= \int d\mathbf{x}_2\ E_{p(\mathbf{x}_1|\mathbf{x}_2)}\left\{[\mathbf{x}_1, \mathbf{x}_2]\right\}\ \ w(\mathbf{x}_2)\pi(\mathbf{x}_2) \tag{3.44}$$

then if N independent samples are drawn from the importance distribution $\pi(\mathbf{x}_2)$, the integral in (3.39) can be approximated by

$$\hat{I}_{Rao-MMSE} = \frac{1}{N}\sum_{i=1}^{N} E_{p(\mathbf{x}_1|\mathbf{x}_2)}\left\{\left[\mathbf{x}_1, \mathbf{x}_2^i\right]\right\}\ \ w(\mathbf{x}_2^i)\ . \tag{3.45}$$

In [23] p.27 it says "Rao-Blackwellization. This method reflects a basic principle (or rule of thumb) in Monte Carlo computation: One should carry out analytical computation as much as possible." This principle will be used in section 3.4.

Finally, another important issue in IS is weight normalization. In MMSE estimation the density $p(\mathbf{x})$ in (3.39) represents a posterior PDF which may be known up to a constant, that is, $p(\mathbf{x}) = \dot{p}(\mathbf{x})/c$ where $c = \int d\mathbf{x}\, \dot{p}(\mathbf{x})$. The integral in (3.41) can thus be written

$$I_{MMSE} = \int d\mathbf{x}\ \mathbf{x}\ \frac{\dot{p}(\mathbf{x})}{\pi(\mathbf{x})}\pi(\mathbf{x}) \ \Big/ \int d\mathbf{x}\ \frac{\dot{p}(\mathbf{x})}{\pi(\mathbf{x})}\pi(\mathbf{x})\ . \tag{3.46}$$

If N independent samples are generated from the importance distribution, $\pi(\mathbf{x})$, then the integral can be approximated by

$$\hat{I}_{Norm-MMSE} = \frac{\sum_i \mathbf{x}^i\, \dot{w}(\mathbf{x}^i)}{\sum_i\ \dot{w}(\mathbf{x}^i)} = \sum_i \mathbf{x}^i\, \bar{w}(\mathbf{x}^i) \quad , \quad \bar{w}(\mathbf{x}^i) = \frac{\dot{w}(\mathbf{x}^i)}{\sum_i\ \dot{w}(\mathbf{x}^i)}\ . \tag{3.47}$$

Weight normalization is used in all subsequent examples.

3.4 Sequential Importance Sampling

Sequential Importance Sampling (SIS) refers to the task of sequentially generating samples from the importance distribution and recursively computing the accompanying weights. This idea is e.g. important for practical use of IS to state estimation in *dynamical systems* as can be understood from the following excerpt from [2] "For many problems an estimate is required every time that a measurement is received. In this case a recursive filter is a convenient solution. A recursive filtering approach means that the received data can be processed sequentially rather than as a batch, so that it is not necessary to store the complete data set nor to reprocess existing data if a new measurement becomes available."

A common way to introduce SIS is to the problem of state estimation in a nonlinear state-space model with known parameters. Because the model is nonlinear direct Kalman filtering is not possible. However, in this chapter a slightly different approach is taken because SIS is introduced for inference in the models (3.4)-(3.9) and (3.20)-(3.25) which are both linear given the parameters in the models and parameters are assumed unknown. For the model (3.4)-(3.9) parameter estimation is of interest and for the model (3.4)-(3.9) joint estimation of the state and the parameters are of interest. Because the model in (3.20)-(3.25) is linear given the parameters the principle of Rao-Blackwellization (see section 3.3) can be used whereby the problem of MMSE estimator approximation is reduced to the problem of sampling from the posterior PDF of the parameters.

Now a general derivation of the recursive weight-update equation is given. The derivation largely follows the derivation provided in [11]. The starting point of

the derivation is a sequence of PDFs $\{p(\mathbf{x}_{1:n})\}$ where n represents a time index. The goal is to derive a method for sampling sequentially from $\{p(\mathbf{x}_{1:n})\}$; that is, first obtain samples and their weights from $p(\mathbf{x}_1)$, then $p(\mathbf{x}_{1:2})$ and so on. The importance weights at time n are given by

$$w_n = \frac{p(\mathbf{x}_{1:n})}{\pi(\mathbf{x}_{1:n})} \ , \tag{3.48}$$

where $\pi(\mathbf{x}_{1:n})$ denotes the importance distribution. The importance distribution is decomposed into $\pi(\mathbf{x}_{1:n}) = \pi(\mathbf{x}_n|\mathbf{x}_{1:n-1})\pi(\mathbf{x}_{1:n-1})$. This decomposition is crucial because it means that if a path $\mathbf{x}_{0:n-1}^{(i)}$ is available (superscript i denotes an index into a set of paths) at time n this path can be augmented with a sample, $\mathbf{x}_n^{(i)}$, drawn from $\pi(\mathbf{x}_n|\mathbf{x}_{1:n-1}^{(i)})$ to obtain the path $\mathbf{x}_{0:n}^{(i)}$, thus the path $\mathbf{x}_{0:n-1}^{(i)}$ is not modified. Inserting the decomposed importance distribution into (3.48), yields

$$\begin{aligned} w_n &= \frac{p(\mathbf{x}_{1:n-1})}{\pi(\mathbf{x}_{1:n-1})} \cdot \frac{p(\mathbf{x}_{1:n})}{p(\mathbf{x}_{1:n-1})\pi(\mathbf{x}_n|\mathbf{x}_{1:n-1})} && (3.49) \\ &= w_{n-1} \ \cdot \ \frac{p(\mathbf{x}_{1:n})}{p(\mathbf{x}_{1:n-1})\pi(\mathbf{x}_n|\mathbf{x}_{1:n-1})} && (3.50) \\ &= w_{n-1} \ \cdot \ \underbrace{\frac{p(\mathbf{x}_n|\mathbf{x}_{1:n-1})}{\pi(\mathbf{x}_n|\mathbf{x}_{1:n-1})}}_{w_{update}} \ . && (3.51) \end{aligned}$$

In going from (3.50) to (3.51) an implicit assumption has been made that PDF $p(\mathbf{x}_{1:n-1})$ at times $n-1$ and n are equal. As noted in [11], the densities $\{\pi(\mathbf{x}_n|\mathbf{x}_{1:n-1})\}$ are parameters to be selected by the user. At each time-step the SIS procedure consists of propagating the particles at the previous time-step by a sampling step and a weight-update step performed for each particle. That is, at time n the SIS procedure completes, for each particle indexed by a superscript i, the steps

1. Sample $\mathbf{x}_n^{(i)} \sim \pi(\mathbf{x}_n|\mathbf{x}_{1:n-1}^{(i)})$
2. Update weight using (3.51) to get $w_n^{(i)}$.
3. Normalize weights.

To summarize SIS, when the objective is to apply IS for state-estimation in a dynamical system or for parameter estimation in a model whose parameters change over time we have to draw samples from a sequence of posterior PDFs denoted by e.g. $\{p(\mathbf{x}_{1:n})\}$ and compute weights given by the ratio $p(\mathbf{x}_{1:n})/\pi(\mathbf{x}_{1:n})$. The

problem is that the raw IS procedure requires us to draw paths whose length increase over time which means that the number of computations and memory consumption increase with time. The key idea in arriving at the SIS procedure was that of decomposing the importance distribution in such a way that samples from the sequence of PDFs of interest, $\{p(\mathbf{x}_{1:n})\}$, could be generated recursively. The decomposition could also be used to derive an expression for the weights such that they can be recursively propagated in time. Thus, SIS eliminates the increase in the number of computations and memory consumption over time.

Considering the speech model in (3.20)-(3.25) the sequence of PDFs of interest is the sequence of posterior PDFs $\{p(\mathbf{w}_{0:n}, \mathbf{a}_{0:n}, \sigma_{x_{0:n}}, \sigma_{d_{0:n}}|y_{1:n})\}$. Now a notational context switch is made in order to comply with the notation common to SIS literature. The state vector is renamed from $\mathbf{w}$ to $\mathbf{x}$ and all parameters are stacked in the parameter vector $\boldsymbol{\theta}$. That is, the sequence of posterior PDFs is denoted $\{p(\mathbf{x}_{0:n}, \boldsymbol{\theta}_{0:n}|y_{1:n})\}$. The n^{th} element in the sequence can be decomposed into $p(\mathbf{x}_{0:n}, \boldsymbol{\theta}_{0:n}|y_{1:n}) = p(\mathbf{x}_{0:n}|\boldsymbol{\theta}_{0:n}, y_{1:n})p(\boldsymbol{\theta}_{0:n}|y_{1:n})$. The point is to realize that $p(\mathbf{x}_{0:n}|\boldsymbol{\theta}_{0:n}, y_{1:n})$ is Gaussian because the model in (3.20)-(3.25) reduces to a linear Gaussian state-space model (see also [39]). Integration over the state vector can thus be performed analytically. This process is referred to as Rao-Blackwellization as explained in section 3.3. The MMSE estimator for the state vector and parameter vector is given by the following expression

$$< \mathbf{x}_n, \boldsymbol{\theta}_n >= \int d\mathbf{x}_{0:n} \int d\boldsymbol{\theta}_{0:n} \; [\mathbf{x}_n, \boldsymbol{\theta}_n] \; p(\mathbf{x}_{0:n}, \boldsymbol{\theta}_{0:n}|y_{1:n}) \; . \tag{3.52}$$

As a consequence of the Rao-Blackwellization, approximating this MMSE estimator is depending on generating samples from the sequence $\{p(\boldsymbol{\theta}_{0:n}|y_{1:n})\}$. If N independent samples are drawn from the importance distribution $\pi(\boldsymbol{\theta}_{0:n})$ then (3.52) can be approximated by

$$< \widehat{\mathbf{x}_n, \boldsymbol{\theta}_n} > = \frac{1}{N} \sum_{i=1}^{N} E_{p(\mathbf{x}_{0:n}|\boldsymbol{\theta}^i_{0:n}, y_{1:n})}\{[\mathbf{x}_n, \boldsymbol{\theta}^i_n]\} \; w(\boldsymbol{\theta}^i_{0:n}|y_{1:n}) \tag{3.53}$$

The factor w_{update} in (3.51) is given by

$$w_{update} = \frac{p(\boldsymbol{\theta}_n|\boldsymbol{\theta}_{0:n-1}, y_{1:n})}{\pi(\boldsymbol{\theta}_n|\boldsymbol{\theta}_{0:n-1}, y_{1:n})} \tag{3.54}$$

$$\propto \frac{p(y_n|\boldsymbol{\theta}_{0:n}, y_{1:n-1})p(\boldsymbol{\theta}_n|\boldsymbol{\theta}_{1:n-1})}{\pi(\boldsymbol{\theta}_n|\boldsymbol{\theta}_{0:n-1}, y_{1:n})} \tag{3.55}$$

At time n the SIS procedure for MMSE estimation of state and parameter vector completes, for each particle indexed by a superscript i, the steps

1. Sample $\boldsymbol{\theta}^{(i)}_n \sim \pi(\boldsymbol{\theta}_n|\boldsymbol{\theta}^{(i)}_{1:n-1})$.

2. Perform Rao-Blackwellization to get $\mathbf{x}_n^{(i)}$.
3. Update weight using (3.55) to get $w_n^{(i)}$.
4. Normalize weights to get $\bar{w}_n^{(i)}$.
5. Estimate state and parameter vector by $\sum_i [\mathbf{x}_n^{(i)}, \boldsymbol{\theta}_n^{(i)}] \bar{w}_n^{(i)}$.

For the model in (3.4)-(3.9) the SIS procedure is very similar. However, when the input is not assumed stochastic which is the case for the model (3.4)-(3.9) used in the examples in sections 3.6-3.8, only the parameters need to be estimated and the Rao-Blackwellization step disappears.

In the examples in sections 3.6-3.8 it is assumed that the parameters evolve according to first-order Markov processes such that $p(\boldsymbol{\theta}_n|\boldsymbol{\theta}_{1:n-1}) = p(\boldsymbol{\theta}_n|\boldsymbol{\theta}_{n-1})$. Furthermore the importance distribution is set equal to $p(\boldsymbol{\theta}_n|\boldsymbol{\theta}_{n-1})$ which means that drawing samples is simple and that the weight update factor in (3.55) reduces to $w_{update} \propto p(y_n|\boldsymbol{\theta}_{0:n}, y_{1:n-1})$. This choice of importance distribution is discussed in the next section.

3.5 Importance Distribution and Resampling

Now the SIS procedure is applied to the example of estimating the mean of the un-normalized PDF given by (3.27). That is, the mean value of the PDF $p(x, y) = f(x, y)/c$ where c is a constant will be approximated by applying the SIS procedure outlined in section 3.4. This example is a little different in that $p(x, y)$ is 'static' and does not necessarily represent a posterior PDF. Particles are adapted by choosing independent and equal first-order Markov processes, also called Random Walks (RWs), as the importance distributions for x and y. The RW variance (reminiscent of a step-size parameter) is set to 0.01 and initially particles are distributed uniformly in the region $[-1, 1] \times [-1, 1]$. That is, if $\boldsymbol{\theta}_n = [x_n, y_n]^T$ then

$$p(\boldsymbol{\theta}_0) = \mathcal{U}([-1, -1],\ [1, 1]) \tag{3.56}$$

$$p(\boldsymbol{\theta}_n|\boldsymbol{\theta}_{n-1}) = \tilde{\mathcal{N}}(\mathbf{0}_{2\times 1},\ 0.01\mathbf{I}_{2\times 2}) \tag{3.57}$$

where $\mathcal{U}(\mathbf{a}, \mathbf{b})$ denotes the uniform distribution on the interval given by $[\mathbf{a}, \mathbf{b}]$ and $\tilde{\mathcal{N}}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes a Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ that in this case is truncated to the region $[-1, 1] \times [-1, 1]$.

Figure 3.8 and 3.9 show particle paths and weights, respectively, after one iteration. For this example thirty particles are used. Figure 3.10 and 3.11 show

particle paths and weights after six iterations. From this example it is clear that only few particles contribute to the mean estimation because many particle weights are insignificant. The whole idea of IS is to make efficient use of particles. For this example it is clear that inefficient use of the particles is the result of a non-optimal importance distribution which places particles in regions where $f(x, y)$ is negligible. This leads to the question of what the optimal importance distribution for SIS is?

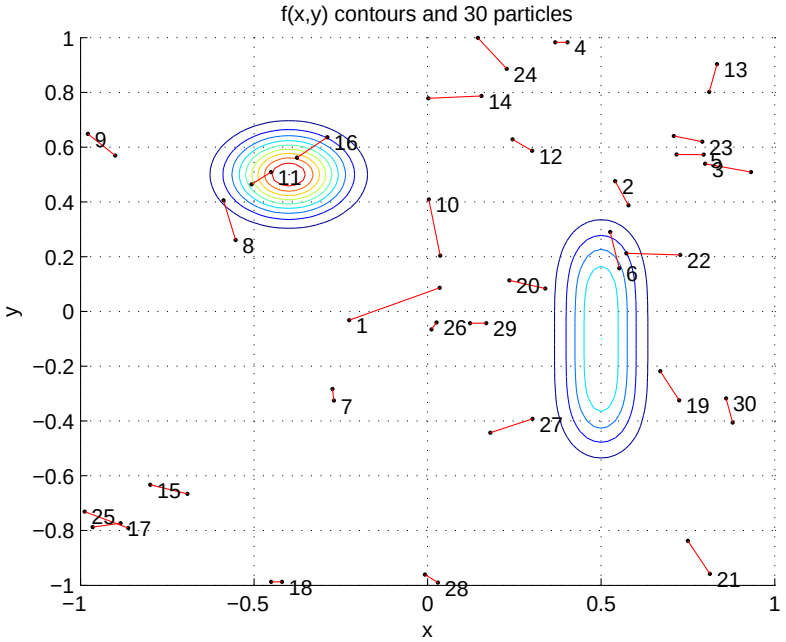

Figure 3.8: Figure shows contours of $f(x, y)$ and the 30 particle paths after one iteration of SIS.

In [13] the optimal importance function is found under the criterion of minimum variance of the importance weights. However, use of the optimal importance function is made difficult by two issues 1) sampling from it and 2) evaluation of the importance weights. As mentioned in both [13] and [11] then the SIS procedure is not guaranteed to be efficient even though the optimal importance function can be used. Furthermore, it can even be shown that the variance of the importance weights increases over time even though the optimal importance function is used. In [13] it says "Thus, it is impossible to avoid a degeneracy phenomenon. In practice, after a few iterations of the algorithm, all but one of the normalized importance weights are very close to zero and a large computa-

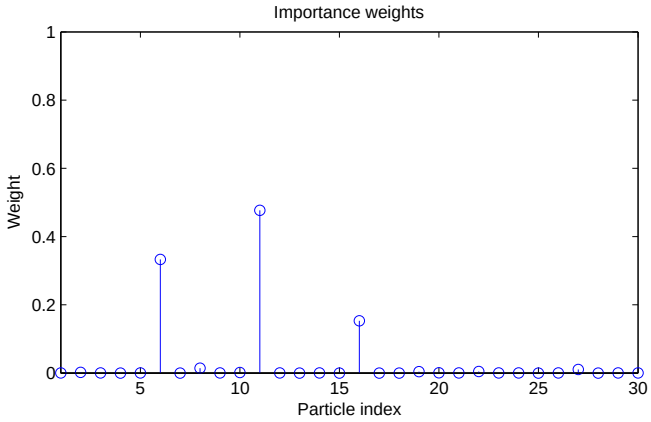

Figure 3.9: Figure shows particle weights after one iteration of SIS.

tional effort is devoted to updating trajectories whose distribution to the final estimate is almost zero." This phenomenon is referred to as degeneration of the particles.

In order to reduce the effect of the degeneracy, particle filters adds a resampling step (also called selection step in some literature) to the SIS procedure. The fundamental idea behind the resampling is to eliminate particles with low importance weights and move them to regions with higher importance weights. The way this is done is to duplicate some particles a number of times. In [34] it is written "A selection scheme associates to each particle $\mathbf{x}_{0:t}^{(i)}$ a number of 'children', say $N_i \in \mathbb{N}$, such that $\sum_{i=1}^{N} N_i = N$" where N denotes the number of particles. After resampling all particles (original and duplicated) have equal weights $1/N$. The example is now continued to demonstrate how this works.

The example is repeated but now with a resampling step in each iteration. In figure 3.12 those particles after first iteration (see figure 3.8) to be deleted are marked by a red cross. In figure 3.13 those particles marked by a cross in figure 3.12 have been removed and the four particles left after the deletion have been duplicated a number of times and adapted with one step of a Gaussian RW. In figure 3.14 this procedure has been repeated six times and we see that all thirty particles are more or less placed in 'important' regions and thereby increasing efficiency of the particles. However, the example also do illustrate that in this particular case the particle filter is not 'capturing' the presence of the weaker second mode of the PDF which makes the estimate of the mean value quite inaccurate. Eventually, even when using thousands of particles then after enough iterations the particles will all be placed around the 'stronger' mode.

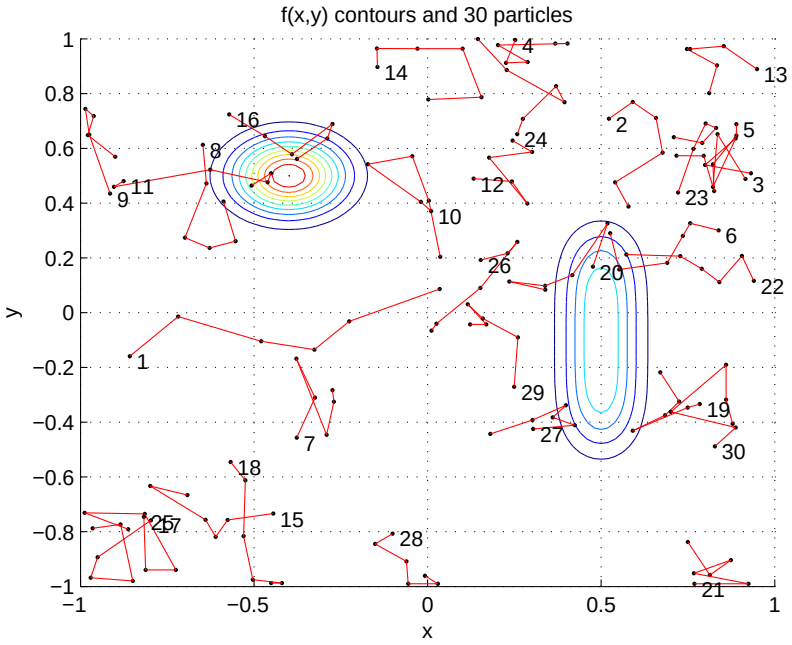

Figure 3.10: Figure shows contours of $f(x, y)$ and particle paths after six iterations of SIS.

The resampling procedure used in this example is called *residual resampling*. This resampling procedure is used in the ReBEL (Recursive Bayesian Estimation Library) toolbox and is described in [34].

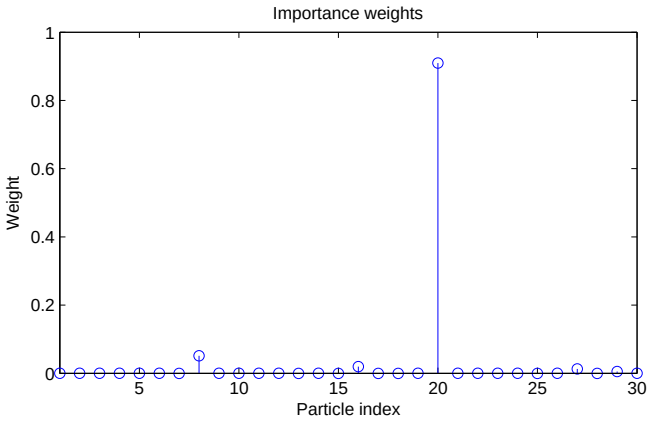

Figure 3.11: Figure shows particle weights after six iterations of SIS.

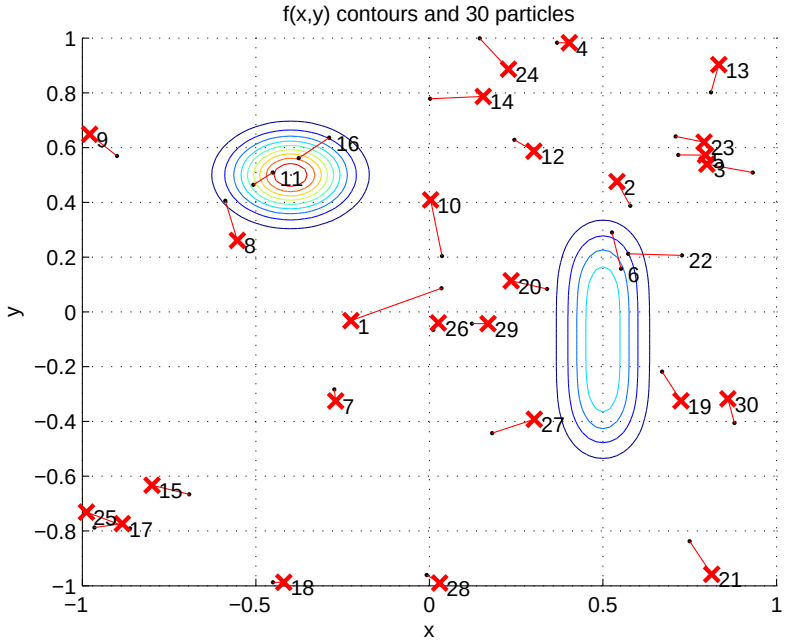

Figure 3.12: The figure shows which particles are deleted after a resampling step in first iteration. Se figure 3.8 for comparison. After resampling four particles are retained (number 6, 11, 16, 22)

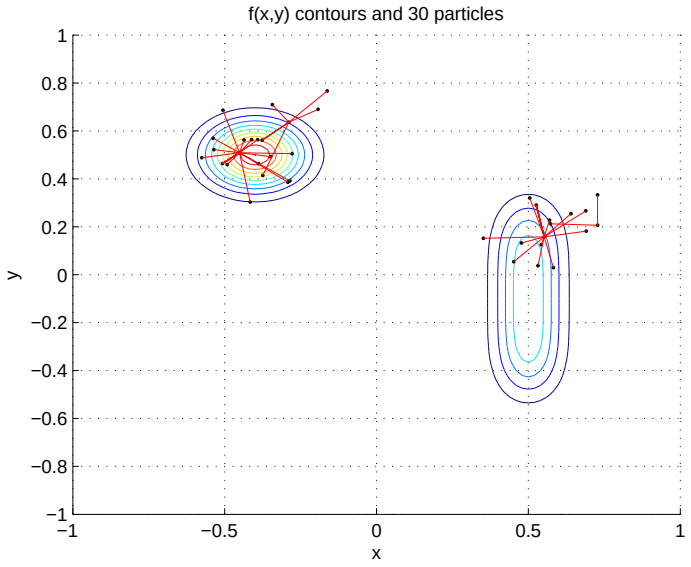

Figure 3.13: The four particles left after the deletion have been duplicated a number of times and adapted with one step of a Gaussian RW.

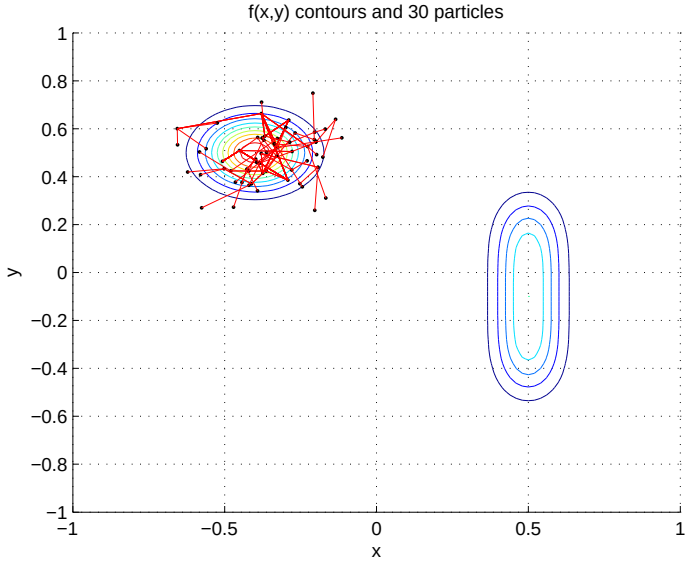

Figure 3.14: This figure shows particles paths after six iterations of SIS with resampling.

3.6 Bayesian Adaptive FIR Filtering

In this section a PF approach to the adaptive FIR filtering problem is presented. Generally, the FIR model is

$$y_n = \sum_{k=0}^{M} b_k\, x_{n-k} + \sigma_{u_n}\, u_n = \mathbf{b}_n^T\, \mathbf{x}_n + \sigma_{u_n}\, u_n \quad , \quad u_n \sim \mathcal{N}(0,1) \tag{3.58}$$

where M denotes the order of the filter. The model in (3.58) is a special case of the model in (3.4)-(3.9). In adaptive filtering the goal is to estimate the coefficient vector $\mathbf{b}_n$ given input and output data. In addition to the coefficient vector the noise variance is also assumed unknown. The parameters are assumed to evolve according to the following first-order Markov processes

$$\begin{array}{llll} \mathbf{b}_0 = \boldsymbol{\nu} & , & \boldsymbol{\nu} & \sim \mathcal{N}\left(\mathbf{0}_{(M+1)\times 1}, \delta_{\mathbf{b}_0}^2 \mathbf{I}_{(M+1)}\right) \\ \mathbf{b}_n = \mathbf{b}_{n-1} + \boldsymbol{\varsigma}_n & , & \boldsymbol{\varsigma}_n & \sim \mathcal{N}\left(\mathbf{0}_{(M+1)\times 1}, \delta_{\mathbf{b}}^2 \mathbf{I}_{(M+1)}\right) \\ \sigma_{u_0}^2 = \gamma & , & \gamma & \sim \mathcal{U}\left(0, \delta_{u_0}^2\right) \\ \sigma_{u_n}^2 = \sigma_{u_{n-1}}^2 + \psi_n & , & \psi_n & \sim \mathcal{N}\left(0, \delta_u^2\right) \end{array} \tag{3.59}$$

where the hyper-parameters $\delta_{\mathbf{b}_0}^2, \delta_{\mathbf{b}}^2, \delta_{u_0}^2, \delta_u^2$ need to be tuned for the particular problem at hand. Because samples are generated by the prior the weights are given by

$$\omega_n \propto p(y_n|\mathbf{x}_n, \mathbf{b}_n, \sigma^2) \quad , \quad p(y_n|\mathbf{x}_n, \mathbf{b}_n, \sigma^2) = \mathcal{N}\left(y_n - \mathbf{b}_n^T\, \mathbf{x}_n, \sigma^2\right) \tag{3.60}$$

As a demonstration one thousand samples are generated using (3.58) where $M = 3$, $\mathbf{b} = [0.7, -0.4, 0.2, -0.9]^T$ and $\sigma_{u_n}^2 = 0.1$. In figure 3.15 is shown the result of using the NLMS algorithm to estimate the coefficients for two different step-sizes. Figure 3.15 indicates that the step-size parameter can be varied to adjust the magnitude of the fluctuations and the rate of convergence. In the PF method the RW variances have a similar role. Furthermore, the number of particles to use also needs to be determined. Figure 3.16 shows the results of applying the PF method to estimate the coefficients and the noise variance. The setting of the hyperparameters is $\delta_{\mathbf{b}_0}^2$=1, $\delta_{u_0}^2$=1, δ_u^2=5e−5 and the number of particles used is 30. In figure 3.16a $\delta_{\mathbf{b}}^2$=1e−4 and in figure 3.16b $\delta_{\mathbf{b}}^2$=5e−4. The PF method is also used to estimate the noise variance which is also shown in figure 3.16. Computational-wise the PF method is more expensive than the NLMS algorithm, approximately the number of particles times more expensive. Maybe the most interesting aspect of using a PF based adaptive filter is the opportunity to incorporate knowledge of the system into the adaptation. It is for instance straightforward to reparameterize the model in (3.58) using one of the parameterizations discussed in section 3.1.

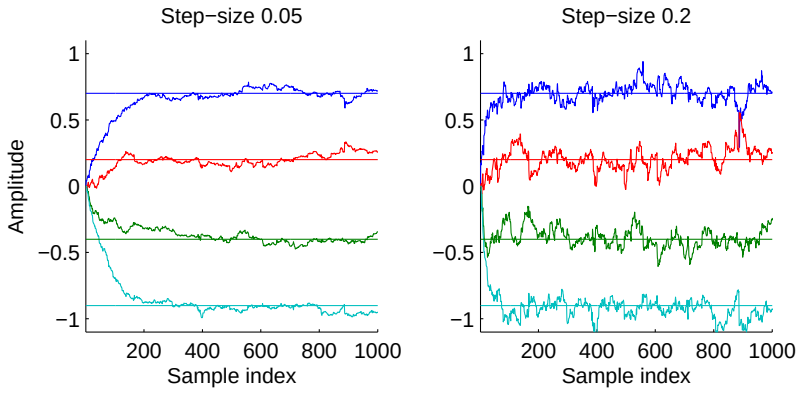

Figure 3.15: Plot to the left shows the true values of the coefficients and the estimated values by the NLMS algorithm using step-size 0.05. Plot to the right shows the same with step-size 0.2.

3.7 Bayesian Adaptive IIR Filtering

The PF method can also be used in the adaptive IIR filtering scenario. The IIR model is

$$z_n = \sum_{k=1}^{N} a_k \, z_{n-k} + x_n \tag{3.61}$$

$$y_n = \sum_{k=0}^{M} b_k \, z_{n-k} + \sigma_u \, u_n \quad , \quad u_n \sim \mathcal{N}(0,1) \tag{3.62}$$

which is equivalent to the model in (3.4)-(3.9). Given input and output the goal is to estimate $\mathbf{a}$ and $\mathbf{b}$, where $\mathbf{a} = [a_1, \ldots, a_N]^T$ and $\mathbf{b} = [b_0, \ldots, b_M]^T$. When adapting the a_k coefficients care must be taken to make sure that the resulting filter remains stable. That is, the poles of the IIR system must remain within the unit circle. One way to ensure stability in the PF approach is to model each of the a_k coefficients with a Gaussian RW and every time a set of coefficients is generated, the roots of the corresponding polynomial is calculated. If the roots are not all within the unit circle the coefficients are discarded and a new set of coefficients is generated and so on. However, this approach is not very efficient because of the need to compute the roots of a polynomial for every set of coefficients generated. A better approach is to model the roots directly because it is then much simpler to check for stability. As a demonstration a complex conjugate pole pair is added to the example in section 3.6. The complex conjugate pole pair is chosen such that a bump is introduced in the frequency

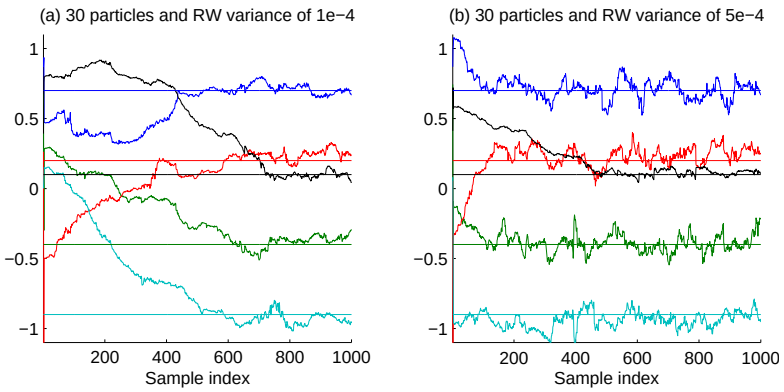

Figure 3.16: Plot to the left shows the true values of the coefficients and the estimated values by the PF algorithm using a RW variance of $1e{-}4$ for the coefficient. Plot to the right shows the same with a RW variance of $5e{-}4$.

response of the filter at a digital frequency of 0.125π and with a bandwidth of 0.0625π which results in $\mathbf{a} = [1.675, -0.8217]^T$. If a sample rate of 16 kHz is assumed then the center frequency of the resonance occurs at 1000 Hz and the bandwidth corresponds to 500 Hz. It is assumed that the center frequency and bandwidth of the resonance evolve according to first-order Markov processes specified by

$$\begin{array}{llll} f_0 = \epsilon & , & \epsilon & \sim \mathcal{U}\left(f_{low}, f_{high}\right) \\ f_n = f_{n-1} + \lambda_n & , & \lambda_n & \sim \mathcal{N}\left(0, \delta_f^2\right) \\ b_0 = \beta & , & \beta & \sim \mathcal{U}\left(b_{low}, b_{high}\right) \\ b_n = b_{n-1} + \alpha_n & , & \alpha_n & \sim \mathcal{N}\left(0, \delta_b^2\right) \end{array} \tag{3.63}$$

with $f_{low} = 200$ Hz, $f_{high} = 4000$ Hz, $b_{low} = 20$ Hz, $b_{high} = 1000$ Hz, $\delta_f^2 = \delta_b^2 = 20$. The setting for the b_k coefficients and noise variance is the same as in section 3.6 and the RW variance for the b_k coefficients is $\delta_{\mathbf{b}}^2$=$1e{-}4$. The result of one simulation is shown in figure 3.17.

3.8 Bayesian Periodic Signal Modeling

A time-series, f_n, is periodic with period N if $f_{n+N} = f_n$. An N-periodic time-series can be represented by a Fourier series

$$f_n = a_0 + \sum_{k=1}^{N-1} \left(a_k \cos(k\, 2\pi n/N) + b_k \sin(k\, 2\pi n/N) \right) \tag{3.64}$$

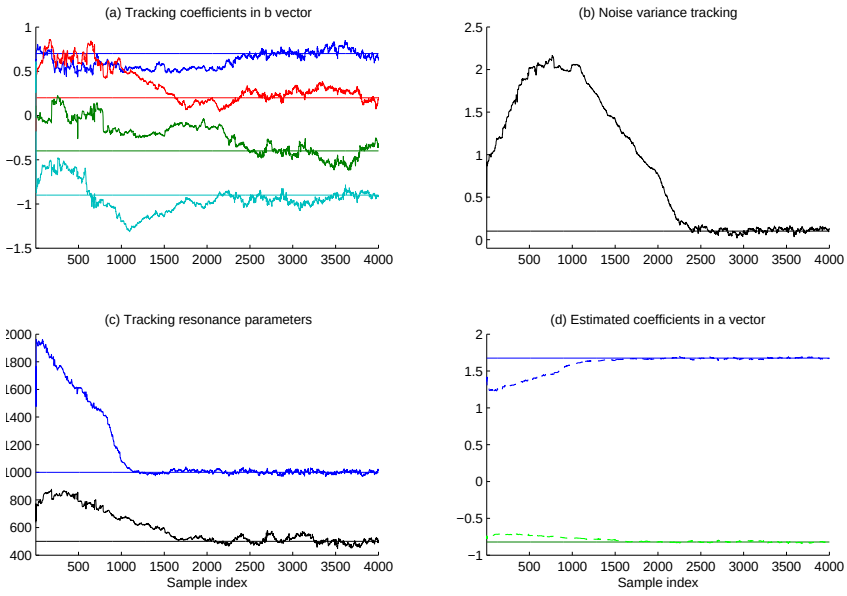

Figure 3.17: (a) True and estimated b_k coefficients (b) true and estimated noise variance (c) true and estimated resonance frequency and bandwidth (d) paths from (c) converted to a_k coefficients.

For the example in this section a zero-mean signal will be modeled which means that a_0=0 and the Fourier series is truncated to L components (the fundamental and $L-1$ harmonics) such that the following model is used

$$\hat{f}_n = \sum_{k=1}^{L} \left(\, a_k \cos(k\, 2\pi n/N) + b_k \sin(k\, 2\pi n/N) \, \right) \tag{3.65}$$

A model for a noisy periodic signal is thus

$$y_n = \mathbf{c}_n^T \mathbf{x}_n + \sigma_u \, u_n \quad , \quad u_n \sim \mathcal{N}(0,1) \tag{3.66}$$

where (omitting time index) $\mathbf{c}^T = [a_1, b_1, \ldots, a_L, b_L]^T$ and

$$\mathbf{x}_n = [\cos(2\pi n/N), \sin(2\pi n/N), \ldots, \cos(L\, 2\pi n/N), \sin(L\, 2\pi n/N)]^T \tag{3.67}$$

Here, (3.65) is used to model a synthetic periodic signal. The periodic signal is generated using the Liljencrants-Fant (LF) model which models the harmonic component present in the signal exciting the vocal tract when producing voiced speech. The waveform was generated using the settings t_0=$99t_s$, t_e = $0.7t_0$, t_p=$0.6t_0$, t_a=$0.05t_0$, E_e=3, where t_s denotes the sampling period. See

Appendix D for an introduction to the LF model and its parameters. Figure 3.18 shows one period of the generated excitation signal. The figure also shows the Fourier series coefficients of the waveform and the waveform which results when retaining only six components in the Fourier series. The waveform shown

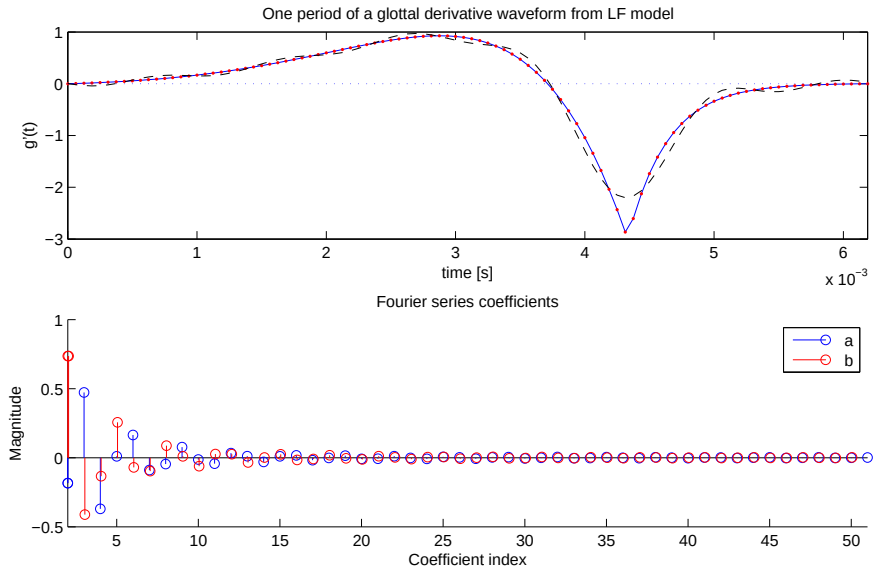

Figure 3.18: Upper plot shows one period of the derivative glottal flow waveform generated using the LF model. The period is 100 samples and $f_s = 16$ kHz. The plot also shows the approximation to this periodic signal (black dashed line) by retaining the first L components in the Fourier series of the waveform, that is, $L = 6$ in (3.65). Lower plot shows the Fourier series coefficients of the periodic waveform (blue: a_k coefficients, red: b_k coefficients).

in the upper plot of figure 3.18 was repeated 80 times such that an 8000-samples sequence was obtained. White Gaussian noise with variance $3.7e{-}2$ was added to the periodic signal which resulted in a SNR of approximately 12 dB. Each Fourier coefficient was adapted using a Gaussian RW with variance $1e{-}3$. The pitch period is adapted using a discrete uniform RW. If N denotes the pitch period then,

$$\begin{aligned} &N_0 = \delta \quad , \quad \delta \sim \mathcal{UD}\,(70, 130) \\ &N_n = \begin{cases} N_{n-1} & , \qquad\qquad\qquad , \quad if\ mod(n, N_{n-1}) == 0 \\ N_{n-1} + \phi_n & , \quad \phi_n \sim \mathcal{UD}\,(-4, 4) \quad , \quad else \end{cases} \end{aligned} \tag{3.68}$$

where $\mathcal{UD}(a,b)$ denotes the discrete uniform distribution on the interval from a to b which is the set $\{a, a+1, \ldots, b\}$. If c denotes a Fourier coefficient then

$$\begin{aligned} c_0 &= \delta \quad , \quad \delta \sim \mathcal{N}(0, 0.2) \\ c_n &= \begin{cases} c_{n-1} & , \\ c_{n-1} + \eta_n & , \quad \eta_n \sim \mathcal{N}(0, 1e{-}3) \end{cases} \begin{array}{l} , \\ , \end{array} \begin{array}{l} if\ mod(n, N_{n-1}) == 0 \\ else \end{array} \end{aligned} \tag{3.69}$$

The noise variance was adapted using a Gaussian RW with variance $1e{-}5$ limited to positive values. The particle filter used 200 particles. One simulation was made and figure 3.19 shows the pitch period estimates. Figure 3.20 shows the last 1000 samples of the clean period signal and the estimated periodic signal.

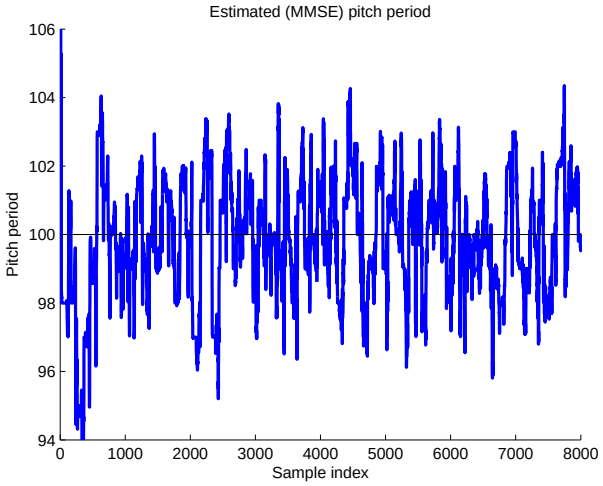

Figure 3.19: Figure shows the estimated pitch period. The true pitch period is 100 samples.

3.9 Discussion

Particle filtering was introduced by first explaining properties of IS before moving on to SIS and the necessity of resampling. Particle filtering inference is based on random numbers. This has the side-effect that the output of the particle filter will be different every time a new experiment is performed unless the sequence of random numbers used are exactly the same. This aspect of particle filtering may represent a significant disadvantage because it complicates quantification of the performance of a given particle filter. However, on the other hand particle

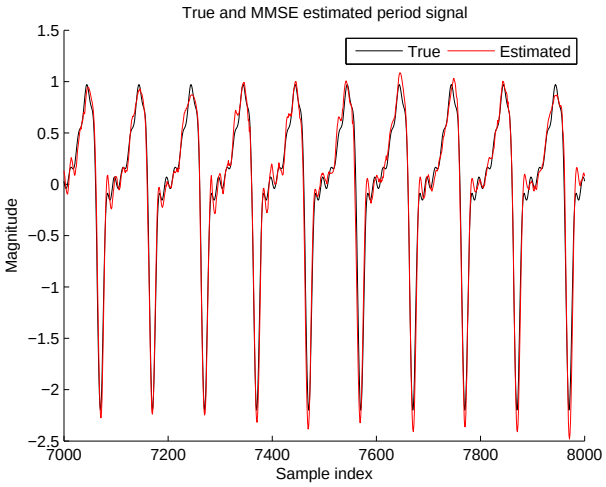

Figure 3.20: Figure shows a number of periods of the clean period signal and the estimated signal for an SNR of approximately 12 dB.

filtering represents a quite flexible method for approximate Bayesian inference as it can be modified to work with reparameterizations or changes in a model in a relative straightforward manner as demonstrated by the examples.

Particle filtering was demonstrated in sections 3.6-3.8 by applying the method to three problems of relevance to hearing aids. In appendices F and G particle filtering was applied to the problem of reducing noise in speech. Especially the results in the paper in appendix G seem promising. The paper examines the usefulness of particle filter inference in three different speech models for the application of non-stationary noise reduction. The most promising candidate speech model consists of a cascade-coupling of second-order IIR resonance filters parameterized by formant frequencies and formant bandwidths. Next, performance of a commercial-grade hearing aid noise reduction algorithm is compared to the particle filter algorithm which uses the chosen candidate speech model. Performance is compared for both stationary and non-stationary white noise and the particle filter algorithm compares favorably to the hearing aid noise reduction algorithm in low-SNR situations.

Chapter 4

Results & Outlook

This report has considered MMSE signal estimation in a single-channel setup where the signal is contaminated by additive noise. This problem has been approached in two different ways. One way in which priors for the signal and noise have been expressed by their PDFs and the MMSE estimator is approximated using saddlepoint approximation and another way in which the priors have been provided in terms of generative models and the MMSE estimator is approximated by particle filtering. Conclusions and recommendations for further research will be given for both approaches.

4.1 Saddlepoint Approximation

The integral defining the MMSE estimator when expressed directly in terms of prior PDFs consists of a weighted convolution integral between the prior PDFs, see e.g. (1.4) or (2.2). However as has been shown in this report then for the univariate case considered in the paper in appendix E and for the multivariate case considered in chapter 2 then this integral can be rewritten into an inverse Laplace transform integral with an integrand that is the product between the Moment Generating Function (MGF) of the noise and the first-order derivative of the MGF of the signal. The presence of the derivative of the MGF changes

everything compared to the situation of using SPA for approximating tail probabilities where there is no derivative of MGFs involved. The consequence of the derivative is that the search for a useful saddlepoint becomes significantly more complex as the search problem is not confined to locating a unique and real saddlepoint as in the case of approximating a tail probability. Furthermore, there is the added issue of nearby or coalescing saddlepoints as has been demonstrated in the paper in appendix E. This means that applying SPA to approximate the MMSE integral is far from 'plug-n-play'. It is not straightforward to extend and apply the results of the paper in appendix E to the multivariate case. Even if the results in appendix E is (or can be) extended to the multivariate case then applying the results gets rather involved and will be limited to low dimensional problems. As such,

MMSE estimation by SPA appears not as promising as we thought before we started and we do not consider it a priority to further investigate this method.

4.2 Particle Filtering

The use of particle filtering for MMSE estimation of speech has been examined in detail. A reparameterized time-varying auto-regressive model for speech was used. It has been found very beneficial to reparameterize the speech model in terms of formant feature parameters to exploit prior knowledge of these features in the estimation task. Even though the proposed reparameterization introduced non-linearities into the model then particle filtering turned out to be a quite flexible and easy-to-use method for such reparameterizations. The versatility of particle filtering for parameter estimation has also been demonstrated in the three examples in sections 3.6-3.8, where especially the example in section 3.8 could be used to improve the speech model.

The performance of the particle filtering algorithm was compared to a state of the art hearing aid noise reduction algorithm in the paper in appendix G. Both stationary and non-stationary white noise were used. In order to make a meaningful comparison a modified MMSE estimator was introduced. It allowed trading off SNR improvement for increased speech intelligibility by adjusting one parameter. Performance of the two algorithms were measured by two complementary objective measures, segmental SNR improvement and a signal distortion measure. The measurements indicated that the particle filtering algorithm performs better than the hearing aid algorithm for SNR levels below 10 dB. However, informal listening tests indicated that performance of the two algorithms was very similar. It is believed that short-comings of the used speech model is the main reason that listening tests did not favor the particle filtering

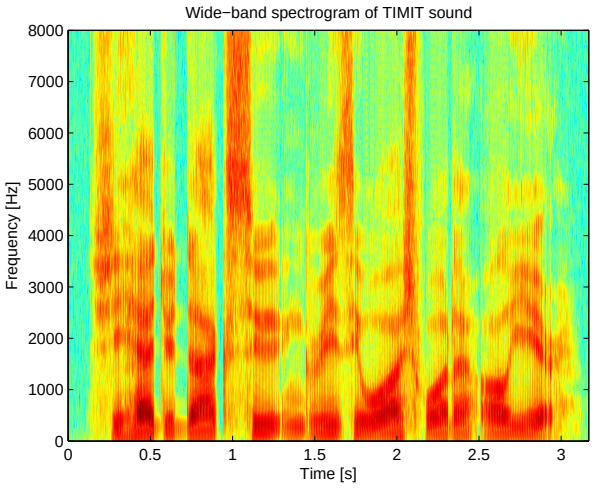

Figure 4.1: Wide-band spectrogram of TIMIT sound SA1 downsampled to 16 kHz.

algorithm over the hearing aid algorithm.

It was also found that the particle filtering algorithm was able to track the noise variance better than the hearing aid algorithm for SNR levels below 10 dB. However, for SNR levels above 10 dB the noise tracking of the particle filter algorithm became inaccurate during intervals of speech with most of the energy concentrated at high frequencies. The used speech model does not model such high frequency energy. To illustrate this point, figure 4.1 shows a wide-band spectrogram of the TIMIT sound SA1 ('She had your dark suit in greasy wash-water all year'). It is observed that at the time points 1.0 s, 1.7 s and 2.1 s bursts with significant energy at high frequencies are present in the speech signal. If this spectrogram is correlated with figure 11b in the paper in appendix G it is seen that these intervals cause inaccurate estimates of the noise variance.

It is believed that performance of the particle filter algorithm can be improved significantly by extending the speech model. Both noise tracking as well as noise reduction capabilities and also for SNR levels above 10 dB. One shortcoming of the speech model is that it does not model speech energy above 5000 Hz because formants do not appear above approximately this frequency and for a sampling frequency of 16 kHz it gives a high-frequency interval where present speech energy is not modeled. The speech model could for instance be extended such that the output is given as a mixing of the non-extended speech

model and high-pass filtered noise. Another short-coming of the speech model is a missing harmonic component in the excitation signal which occurs during voiced speech. Such a harmonic component can for instance be modeled by a Fourier series which was also used in the example in section 3.8 but even better it could be modeled by for instance the Rosenberg-Klatt++ model described in appendix D.

The noise tracking capabilities of the particle filtering algorithm looks interesting from a hearing aid point of view. Excellent noise tracking capabilities is important for the success of a non-stationary noise reduction algorithm and traditional approaches to obtain a noise estimate have limited noise tracking capabilities during periods of speech. Therefore it may be worth investigating how the noise estimate from a particle filter algorithm can be integrated into a hearing aid.

The approximate inference performed by the particle filtering algorithm can also be improved by introducing more sophisticated proposal distributions such as for instance the method developed in [11]. Such approaches will increase efficiency of particles at the expense of increased computational complexity.

There are two primary issues that will have to be investigated further before noise reduction by particle filtering can be considered a serious candidate for use in hearing aids. Firstly, performance must be improved by extending the speech model such that high-frequency speech components are modeled and such that a harmonic excitation signal is also modeled. This should make the particle filtering algorithm more robust in the sense that performance is not varying too much dependent of what speech sounds are being processed and it should also improve performance for higher SNRs. Secondly, validation and quantification of performance is relative tedious because many experiments have to be conducted before a solid statistical picture of the workings of the particle filtering algorithm can been obtained. The particle filtering algorithm is reliant on random numbers and hence performance should be measured statistically. It must be determined whether the increased overhead of doing many experiments is not a practical nightmare. There is also the possibility lurking that for a particular situation the particle filter algorithm will be highly sensitive to some sets of random numbers such that on average in one out of say thousand experiments the particle filtering algorithm performs poorly due to for instance local maxima.

Computational complexity is not considered a primary issue. It is considered possible to implement small particle filters with say ten to twenty particles on next generation digital hearing aid platforms. Another point is that many hearing aid companies are now focusing on wireless links between hearing aids in binaural settings and at some point it is considered likely that communication between a hearing aid and for instance a cell phones will be possible hence the

hearing aid will have access to much more powerful processing units. Another issue that should be investigated further on is that of extending the noise model. In this report emphasis has been focused on developing a useful speech model but the usefulness of the particle filtering algorithm will also be dependent on how accurate the noise model is.

As such,

it is recommended to consider particle inference in the proposed generative model for speech as an alternative approach to the single-channel speech estimation problem and further work should focus on extending the speech model and on issues related to validation and quantification as mentioned above.

Appendix A

Conditional Mean of Signal in Single-Channel Additive Noise Observation Model

Let M-variate RV $\mathbf{x}_k = [x_{k,1}, \ldots, x_{k,M}]^T$ have joint PDF $p_k(\mathbf{x})$ and joint MGF $\mu_k(\boldsymbol{\lambda})$, where the ROA of the MGF is R_k in the M-dimensional complex $\boldsymbol{\lambda}$-plane, and $k = 1, 2$. Let contours $C_k = \{C_{k,m}\},\ m = 1 \ldots M$ lie in the ROA R_k for $k = 1, 2$. Form the sum variable

$$\mathbf{y} = a\,\mathbf{x}_1 + b\,\mathbf{x}_2 \quad , \quad a > 0,\ b > 0. \tag{A.1}$$

The conditional PDF of $\mathbf{x}_1$ given $\mathbf{y}$ is

$$p_c(\mathbf{x}_1|\mathbf{y}) = \frac{p_1(\mathbf{x}_1)p(\mathbf{y}|\mathbf{x}_1)}{p(\mathbf{y})} = \frac{p_1(\mathbf{x}_1)}{p(\mathbf{y})} p_2\left(\frac{\mathbf{y} - a\mathbf{x}_1}{b}\right)\frac{1}{b}. \tag{A.2}$$

Then, the conditional mean of interest is

$$\begin{aligned}
E\{\mathbf{x}_1|\mathbf{y}\} &= \int d\mathbf{x}_1\, \mathbf{x}_1\, p_c(\mathbf{x}_1|\mathbf{y}) \\
&= \frac{1}{p(\mathbf{y})} \int d\mathbf{x}\ \mathbf{x}\ p_1(\mathbf{x})\ p_2\left(\frac{\mathbf{y}-a\mathbf{x}}{b}\right)\ \frac{1}{b} \qquad \text{(A.3)} \\
&= \frac{1}{p(\mathbf{y})\, b} \int d\mathbf{x}\ \mathbf{x}\ p_1(\mathbf{x})\ \frac{1}{(i\,2\pi)^M} \int_{C_2} d\boldsymbol{\lambda}\ \exp\left(-\frac{1}{b}(\mathbf{y}-a\mathbf{x})^T\,\boldsymbol{\lambda}\right) \mu_2(\boldsymbol{\lambda}) \\
&= \frac{1}{(i\,2\pi)^M}\ \frac{1}{p(\mathbf{y})\, b} \int_{C_2} d\boldsymbol{\lambda}\ \mu_2(\boldsymbol{\lambda})\ \exp\left(-\frac{1}{b}\mathbf{y}^T\,\boldsymbol{\lambda}\right) \int d\mathbf{x}\ \mathbf{x}\ p_1(\mathbf{x})\ \exp\left(\frac{a}{b}\mathbf{x}^T\boldsymbol{\lambda}\right) \qquad \text{(A.4)}
\end{aligned}$$

where the convenient condensed notation is made use of: $\int d\mathbf{x} = \int dx_1 \ldots \int dx_M$ and $\int_{C_k} d\boldsymbol{\lambda} = \int_{C_{k,1}} d\lambda_1 \ldots \int_{C_{k,M}} d\lambda_M$. But MGF

$$\mu_1(\boldsymbol{\lambda}) = \int d\mathbf{x}\ p_1(\mathbf{x}) \exp\left(\mathbf{x}^T\,\boldsymbol{\lambda}\right)\ ,\ \mu_1'(\boldsymbol{\lambda}) = \int d\mathbf{x}\ \mathbf{x}\ p_1(\mathbf{x}) \exp\left(\mathbf{x}^T\,\boldsymbol{\lambda}\right) \qquad \text{(A.5)}$$

for $\boldsymbol{\lambda}\,\epsilon\, R_1$, meaning that

$$\mathbf{x}\ p_1(\mathbf{x}) = \frac{1}{(i\,2\pi)^M} \int_{C_1} d\boldsymbol{\lambda}\ \exp\left(-\mathbf{x}^T\,\boldsymbol{\lambda}\right) \mu_1'(\boldsymbol{\lambda}). \qquad \text{(A.6)}$$

The use of (A.5) in (A.4) yields

$$E\{\mathbf{x}_1|\mathbf{y}\} = \frac{1}{(i\,2\pi)^M} \frac{1}{p(\mathbf{y})\, b} \int_C d\boldsymbol{\lambda} \exp\left(-\frac{1}{b}\mathbf{y}^T\,\boldsymbol{\lambda}\right) \mu_2(\boldsymbol{\lambda}) \mu_1'\left(\frac{a}{b}\boldsymbol{\lambda}\right),$$

provided that contours $C = \{C_m\}$, $m = 1 \ldots M$ lies in the intersection of ROAs R_1 and R_2. The substitution $\boldsymbol{\lambda} = b\mathbf{z}$ yields perhaps the neatest form, namely,

$$E\{\mathbf{x}_1|\mathbf{y}\} = \frac{1}{(i\,2\pi)^M\, p(\mathbf{y})} \int_C d\mathbf{z} \exp\left(-\mathbf{y}^T\,\mathbf{z}\right) \mu_1'(a\mathbf{z}) \mu_2(b\mathbf{z}). \qquad \text{(A.7)}$$

The vector differentiation in (A.5) has the following meaning

$$\begin{aligned}
\frac{\partial}{\partial \boldsymbol{\lambda}} \exp\left(\mathbf{x}^T\,\boldsymbol{\lambda}\right) &= \begin{bmatrix} \frac{\partial}{\partial \lambda_1} \exp\left(x_1\lambda_1 + \cdots x_M\lambda_M\right) \\ \vdots \\ \frac{\partial}{\partial \lambda_1} \exp\left(x_1\lambda_1 + \cdots x_M\lambda_M\right) \end{bmatrix} \qquad \text{(A.8)} \\
&= \begin{bmatrix} x_1\ \exp\left(x_1\lambda_1 + \cdots x_M\lambda_M\right) \\ \vdots \\ x_M\ \exp\left(x_1\lambda_1 + \cdots x_M\lambda_M\right) \end{bmatrix} \qquad \text{(A.9)} \\
&= \mathbf{x} \exp\left(\mathbf{x}^T\boldsymbol{\lambda}\right) \qquad \text{(A.10)}
\end{aligned}$$

Appendix B

SPA for Single Isolated Saddlepoint

Consider the integral

$$I = \frac{1}{(i\,2\pi)^M} \int_C d\mathbf{z}\, \exp\left[\phi(\mathbf{z})\right], \tag{B.1}$$

where $C = \{C_m\}$, $m = 1 \ldots M$ are Bromwich contours. Suppose that

$$\left.\frac{\partial \phi(\mathbf{z})}{\partial z_m}\right|_{\mathbf{z}_s} = 0 \quad \text{for } m = 1 \ldots M, \tag{B.2}$$

meaning that $\mathbf{z}_s$ is the location of a saddlepoint of the integrand. Then,

$$\phi(\mathbf{z}) \cong \phi(\mathbf{z}_s) + \frac{1}{2}(\mathbf{z} - \mathbf{z}_s)^T \mathbf{\Lambda}(\mathbf{z}_s)(\mathbf{z} - \mathbf{z}_s) \tag{B.3}$$

for $\mathbf{z}$ near $\mathbf{z}_s$. Substitution in (B.1) yields

$$I \cong \frac{\exp\left[\phi(\mathbf{z}_s)\right]}{(i\,2\pi)^M} \int_{C_1} d\mathbf{z}\, \exp\left(\frac{1}{2}(\mathbf{z} - \mathbf{z}_s)^T \mathbf{\Lambda}(\mathbf{z}_s)(\mathbf{z} - \mathbf{z}_s)\right) \tag{B.4}$$

where it is presumed that contours C_1 are equivalent Bromwich contours to C that passes through the point $\mathbf{z}_s$. Substituting $\mathbf{z} = \mathbf{z}_s + i\mathbf{t}$ in (B.4) then

$$I \cong \frac{\exp\left[\phi(\mathbf{z}_s)\right]}{(2\pi)^M} \int_{-\infty}^{\infty} d\mathbf{t}\, \exp\left(-\frac{1}{2}\,\mathbf{t}^T \mathbf{\Lambda}(\mathbf{z}_s)\,\mathbf{t}\right) = \frac{\exp\left[\phi(\mathbf{z}_s)\right]}{(2\pi)^{M/2} |\mathbf{\Lambda}(\mathbf{z}_s)|^{1/2}} \tag{B.5}$$

where the Jacobian of the variable substitution is $J = i^M$. This is the SPA for an isolated saddlepoint, at which $\mathbf{\Lambda}(\mathbf{z}_s) \neq 0$. Correction terms can be derived by extending the expansion in (B.3) to higher-order terms.

Appendix C

MGF M-variate Laplace Distribution

Let x be a zero-mean M-variate Laplace distributed RV with dimensionality M, then the corresponding MGF is computed from

$$\mu_x(\mathbf{z}) = \int d\mathbf{x}\ \exp\left(\mathbf{z}^T\,\mathbf{x}\right) p_{\mathbf{x}}(\mathbf{x}) \tag{C.1}$$

$$= \int d\mathbf{x}\ \exp\left(\mathbf{z}^T\,\mathbf{x}\right) \int_0^\infty ds\ p_{\mathbf{X}|S}(\mathbf{x}|S=s) p_S(s) \tag{C.2}$$

where $p_{\mathbf{X}|S}(\mathbf{x}|S=s)$ is Gaussian with covariance matrix $s\mathbf{\Gamma}$ ($\det \mathbf{\Gamma} = 1$), thus

$$p_{\mathbf{X}|S}(\mathbf{x}|S=s) = \frac{1}{(2\pi\, s)^{M/2}} \exp\left(-\frac{1}{2s}\mathbf{x}^T\mathbf{\Gamma}^{-1}\mathbf{x}\right) \tag{C.3}$$

and s is exponentially distributed,

$$p_S(s) = \frac{1}{\lambda}\ \exp\left(-\frac{s}{\lambda}\right)\ ,\ s > 0,\ \lambda > 0. \tag{C.4}$$

Interchanging the order of integration and inserting (C.4) into (C.2) yields,

$$\mu_x(\mathbf{z}) = \frac{1}{\lambda}\int_0^\infty ds\ \exp\left(-\frac{s}{\lambda}\right) \int d\mathbf{x}\ \exp\left(\mathbf{z}^T\,\mathbf{x}\right) p_{\mathbf{X}|S}(\mathbf{x}|S=s) \tag{C.5}$$

The MD integral over $\mathbf{x}$ is recognized as the MGF of a Gaussian RV with PDF as in (C.3). Hence,

$$\mu_x(\mathbf{z}) = \frac{1}{\lambda}\int_0^\infty ds\ \exp\left(-s\left(\frac{1}{\lambda} - \frac{1}{2}\mathbf{z}^T\mathbf{\Gamma}\mathbf{z}\right)\right) \tag{C.6}$$

$$= \frac{1}{\lambda}\int_0^\infty ds\ \exp\left(s\left(\frac{\lambda\mathbf{z}^T\mathbf{\Gamma}\mathbf{z} - 2}{2\lambda}\right)\right) \tag{C.7}$$

$$= -\frac{1}{\lambda}\frac{2\lambda}{\lambda\mathbf{z}^T\mathbf{\Gamma}\mathbf{z} - 2} \tag{C.8}$$

$$= \frac{1}{1 - \frac{\lambda}{2}\mathbf{z}^T\mathbf{\Gamma}\mathbf{z}} \tag{C.9}$$

The integral in (C.7) converges when $\mathbf{z}^T\mathbf{\Gamma}\mathbf{z} < \frac{2}{\lambda}$.

Appendix D

Brief Introduction to Excitation Models

D.1 The Rosenberg-Klatt Model

The continuous-time Rosenberg-Klatt model (RK model) is given by (see e.g. [24][21])

$$g(t) = \begin{cases} 2at - 3bt^2, & 0 \leq t \leq OQ \cdot T_0 \\ 0 \quad , & OQ \cdot T_0 < t \leq T_0 \end{cases} \tag{D.1}$$

where

$$a = \frac{27 \cdot AV}{4 \cdot OQ^2 \cdot T_0} \quad , \quad b = \frac{27 \cdot AV}{4 \cdot OQ^3 \cdot T_0^2} \quad , \quad a = OQ \cdot T_0 \cdot b. \tag{D.2}$$

Here, $g(t)$ models one period of the harmonic excitation. The upper equation in (D.1) models the open phase of the period and the lower equation models the closed phase of the period. T_0 is the period in seconds, AV is an amplitude parameter and OQ is called the open quotient. OQ is 0 if the there is no open phase (always closed) and it is 1 if there is no closed phase (always open). A plot of two periods of the model waveform is plotted in figure D.1 for the setting $F_s = 16$ kHz, $OQ = 0.6$, $AV = 2.4$ and $T_0 = 0.01$ s (pitch 100 Hz). This model is also referred to as the KLGLOTT88 model.

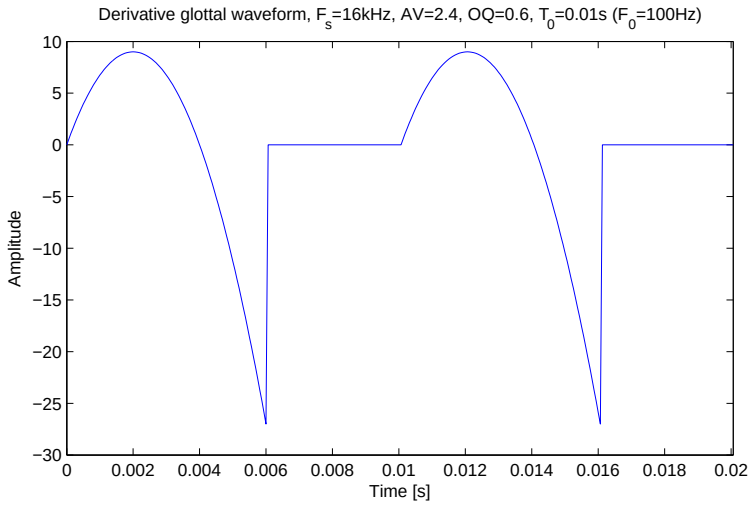

Figure D.1: Two periods of the derivative glottal waveform.

D.2 The LF model

Whereas the well known all-pole model for the vocal tract (the filter) originally was derived based on an acoustic model then the Liljencrants-Fant model (LF model) for the residual signal is derived based on signal processing theory. The LF model was introduced in [16] and the model can make excellent fits to commonly encountered derivative glottal pulse shapes with only 5 parameters. The model is also capable of matching extreme phonations. In [16] it is given by

$$E(t) = \begin{cases} E_0 e^{\alpha t} \sin(\omega_g t) & , \quad 0 \leq t \leq T_e \\ -\frac{E_0}{\epsilon T_a} \left[e^{-\epsilon(t-T_e)} - e^{-\epsilon(T_c-T_e)} \right], & T_e \leq t \leq T_c \end{cases} \tag{D.3}$$

where the first part models the derivative glottal flow from start of the open-phase to the GCI (Glottal Closure Instant). The second part models the return-phase as being exponential. Instead of using the original formulation in [16] the formulation and notation in [38] will be used.

In [38] a framework for a general description of a model of a derivative glottal waveform with an exponential decay modeling the return-phase is presented. The general expression for the derivative glottal waveform is given as

$$g'(t) = \begin{cases} f(t) & , \quad 0 \leq t < t_e \\ f(t_e) \frac{\exp(-(t-t_e)/t_a) - \exp(-(t_0-t_e)/t_a)}{1-\exp(-(t_0-t_e)/t_a)}, & t_e \leq t < t_0 \end{cases} \tag{D.4}$$

In the LF model

$$f(t) = B \sin\left(\pi \frac{t}{t_p}\right) \exp(\alpha t) \ , \tag{D.5}$$

and by direct integration

$$g(t) = \begin{cases} \int_0^t f(\tau)d\tau \quad , \quad 0 \leq t < t_e \\ \int_0^{t_e} f(\tau)d\tau + t_a f(t_e)\frac{1-\exp(-(t-t_e)/t_a)-((t-t_e)/t_a)\exp(-(t_0-t_e)/t_a)}{1-\exp(-(t_0-t_e)/t_a)} \\ , \quad t_e \leq t < t_0 \end{cases}$$

$$= \begin{cases} B\ t_p \left[\frac{\pi+\exp(\alpha t)\left(\alpha t_p \sin(\pi\frac{t}{t_p})-\pi\cos(\pi\frac{t}{t_p})\right)}{(\alpha t_p)^2+\pi^2}\right] \quad , \quad 0 \leq t < t_e \\ \int_0^{t_e} f(\tau)d\tau + t_a f(t_e)\frac{1-\exp(-(t-t_e)/t_a)-((t-t_e)/t_a)\exp(-(t_0-t_e)/t_a)}{1-\exp(-(t_0-t_e)/t_a)} \\ , \quad t_e \leq t < t_0 \end{cases} \tag{D.6}$$

There are 5 parameters in the model, 4 timing parameters and one amplitude parameter. The timing parameters are $\{t_p,t_e,t_a,t_0\}$ and the amplitude parameter is denoted E_e. The parameters are illustrated in figure D.2. The timing parameter t_p denotes the instant of the peak of the glottal flow which also corresponds to when the derivative glottal flow, $g'(t)$, crosses zero. Timing parameter t_e denotes the instant when the derivative glottal flow reaches its negative peak, E_e. This instant is also referred to as the Glottal Closure Instant (GCI). Timing parameter t_0 designates the time it takes to complete one period of the glottal pulse, which is also the pitch period. The timing parameter t_a is the time constant of the exponential decay but it also designates the duration of the return phase. After $t_e + t_a$ the vocal folds have reached maximum closure and the airflow has reduced to its minimum [38]. In order to generate or synthesize a period of the glottal-pulse the synthesis parameters $\{B, \alpha\}$ must be determined. Especially α is not straightforward to determine. In order to determine α a continuity condition must be fulfilled. The continuity condition must be satisfied in order to have zero leakage. To obey the continuity condition the following must be fulfilled

$$\int_0^{t_e} f(\tau)d\tau + t_a f(t_e) D(t_0, t_e, t_a) = 0 \tag{D.7}$$

where

$$D(t_0, t_e, t_a) = 1 - \frac{(t_0 - t_e)/t_a}{\exp((t_0 - t_e)/t_a) - 1} \tag{D.8}$$

which is found by inserting $t = t_0$ into (D.6). If all 4 timing parameters are known then α is determined by solving the continuity condition in (D.7). This can be done for instance by using Matlab's `fzero` function. In [38] an iterative procedure for computing α more suitable for implementation on a DSP is described. After α has been determined the amplitude parameter B can be found by inserting the point (t_e, E_e) into (D.5).

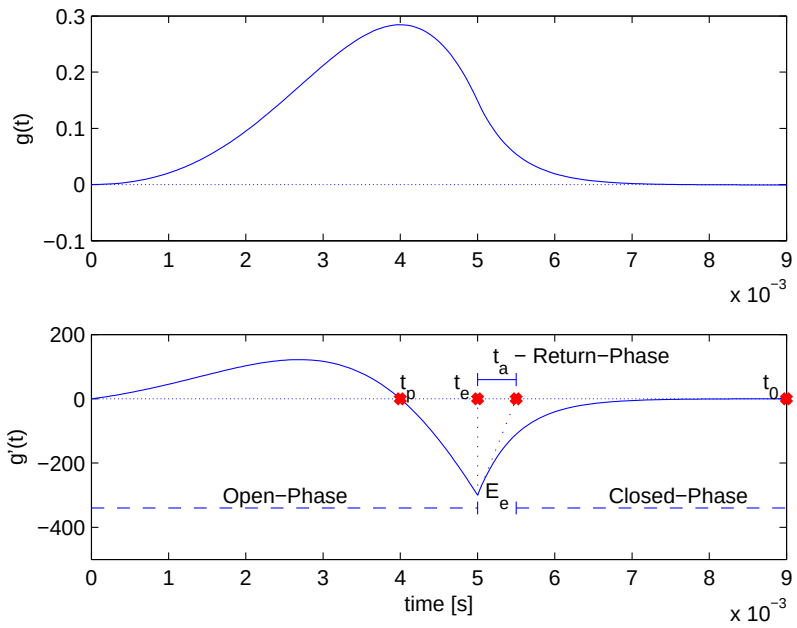

Figure D.2: Glottal pulse (top) and its time derivative (bottom).

D.3 The Rosenberg++ model

The motivation for using the LF model over the Rosenberg-Klatt model is the enhanced flexibility of the LF model and because it also models the return-phase. In particle filtering samples will be drawn from priors of the timing parameters and these samples will then be used to generate one period of the derivative glottal pulse period. This can get quite complex computationally because for every particle the continuity condition has be to solved for α every time new values of the timing parameters are obtained. The Rosenberg++ model (R++ model) is an upgraded Rosenberg-Klatt model which makes a good approximation to the LF model and for which the complexity of solving the continuity condition is significantly decreased. For the R++ model

$$f(t) = 4At(t_p - t)(t_x - t) \tag{D.9}$$

The continuity condition can be solved for t_x analytically

$$t_x = t_e \left(1 - \frac{\frac{1}{2}t_e^2 - t_e t_p}{2t_e^2 - 3t_e t_p + 6t_a(t_e - t_p)D(t_0, t_e, t_a)}\right) \tag{D.10}$$

Appendix E

Use and Subtleties of Saddlepoint Approximation for Minimum Mean-Square Error Estimation

Submitted to IEEE Transactions on Information Theory June 2006.

Use and Subtleties of Saddlepoint Approximation for Minimum Mean-Square Error Estimation

Thomas Beierholm, *Student Member, IEEE,* Albert H. Nuttall, and Lars K. Hansen

***Abstract*— An integral representation for the minimum mean-square error estimator for a random variable in an observation model consisting of a linear combination of two random variables is derived. The derivation is based on the moment-generating functions for the random variables in the observation model. The method generalizes so that integral representations for higher-order moments of the posterior of interest can be easily obtained. Two examples are presented that demonstrate how saddlepoint approximation can be used to obtain accurate approximations for a minimum mean-square error estimator using the derived integral representation. However, the examples also demonstrate that when two saddlepoints are close or coalesce, then saddlepoint approximation based on isolated saddlepoints is not valid. A saddlepoint approximation based on two close or coalesced saddlepoints is derived and in the examples, the validity and accuracy of the derivation is demonstrated.**

***Index Terms*— Minimum mean-square error estimation (MMSE), saddlepoint approximation, coalescing saddlepoints, monkey saddlepoint, moment-generating functions.**

I. Introduction

THE possibility of using SaddlePoint Approximation (SPA) to evaluate the conditional mean for a Random Variable (RV) in a linear combination with constant coefficients of two RVs is examined. The integral defining the conditional mean contains a weighted convolution integral of the corresponding Probability Density Functions (PDFs). This integral is in general intractable, ruling out analytic evaluation of the conditional mean and therefore necessitating the use of approximations. The conditional mean is also referred to as the Minimum Mean-Square Error (MMSE) estimator. The idea described in this paper is based on two observations 1) that the weighted convolution integral defining the desired estimator can be rewritten as a complex inversion integral involving the Moment-Generating Functions (MGFs) of the RVs; 2) that the complex inversion integral can be approximated by SPA.

The SPA was introduced into statistics in [2] where it was used for approximating the mean of n independent and identically distributed RVs. More recently, in [3] and [4] the SPA has proven very useful in obtaining tail probabilities for a number of statistics. Using the SPA to obtain accurate evaluations of a PDF, a unique real SaddlePoint (SP) in the Region Of Analyticity (ROA) is used. In contrast, in order for the SPA to be useful for the approximation of an MMSE estimator, two SPs, as a complex conjugate pair, will in general have to be made use of. This brings on the subtlety that when the two SPs are close, a SPA based on an isolated SP becomes very inaccurate or even worse, when the two SPs coalesce, the usual SPA becomes invalid. In two examples, it has been found that two SPs moved towards each other on the real line as the domain of the conditional RV in the MMSE estimator was traversed. The two SPs finally coalesced into a monkey SP and then moved vertically in opposite directions into the complex plane. A SPA based on close or coalesced SPs is derived, and the accuracy and validity of this SPA, when the two SPs are close or coalesce, is demonstrated in two examples.

T. Beierholm is with GN ReSound A/S, 2630 Taastrup, Denmark.
A. H. Nuttall, P. O. Box 401, Old Lyme, CT 06371, USA.
L. K. Hansen is with Informatics and Mathematcal Modeling, Technical University of Denmark, DK-2800 Kgs. Lyngby, Denmark.

The main contributions of the paper are the derivation of the integral representation for the MMSE estimator and the derivation of a SPA based on two close or coalesced SPs.

In section II-A the derivation of an integral representation for the MMSE estimator is presented. Section II-B reviews the SPA based on an isolated SP, while section II-C derives the SPA based on two close or coalesced SPs. Two examples are presented in section III that demonstrate the subtlety of coalescing SPs and the accuracy of the SPA based on two close or coalesced SPs. Finally, in section IV a summary is presented.

II. Theory

A. Conditional Mean

Let RV x_k have PDF $p_k(x)$ and MGF $\mu_k(\lambda)$, where the ROA of the MGF is R_k in the complex λ-plane, and $k = 1, 2$. Let contour C_k lie in ROA R_k for $k = 1, 2$. Form the sum variable

$$y = ax_1 + bx_2 \ , \ a > 0 \ , \ b > 0$$

The conditional PDF of x_1 given y is

$$p_c(x_1|y) = \frac{p_1(x_1)p(y|x_1)}{p(y)} = \frac{p_1(x_1)}{p(y)} p_2\left(\frac{y - ax_1}{b}\right)\frac{1}{b} \quad (1)$$

Then, the conditional mean of interest is

$$\begin{aligned} E\{x_1|y\} &= \int dx_1 \, x_1 p_c(x_1|y) \\ &= \frac{1}{p(y)} \int dx \, x p_1(x) p_2\left(\frac{y - ax}{b}\right)\frac{1}{b} \quad (2) \\ &= \frac{1}{p(y)b} \int dx \, x p_1(x) \frac{1}{i2\pi} \int_{C_2} d\lambda \, e^{\left(-\frac{y-ax}{b}\lambda\right)} \mu_2(\lambda) \\ &= \frac{1}{i2\pi p(y) b} \int_{C_2} d\lambda \, \mu_2(\lambda) e^{-\frac{y}{b}\lambda} \int dx \, x p_1(x) e^{\frac{a\lambda}{b}x} \quad (3) \end{aligned}$$

But MGF

$$\mu_1(\lambda) = \int dx \ p_1(x) e^{\lambda x} \ , \ \mu_1'(\lambda) = \int dx \ x p_1(x) e^{\lambda x} \quad (4)$$

for $\lambda \epsilon R_1$, meaning that

$$xp_1(x) = \frac{1}{i2\pi}\int_{C_1} d\lambda\; e^{-x\lambda}\mu_1'(\lambda) \tag{5}$$

The use of (4) in (3) yields

$$E\{x_1|y\} = \frac{1}{i2\pi p(y) b}\int_C d\lambda\, e^{-\frac{y}{b}\lambda}\mu_2(\lambda)\mu_1'\left(\frac{a}{b}\lambda\right)$$

provided that contour C lies in the intersection of ROAs R_1 and R_2. The substitution $\lambda = bz$ yields perhaps the neatest form, which is also the form used in the examples, namely,

$$E\{x_1|y\} = \frac{1}{i2\pi p(y)}\int_C dz\, e^{-yz}\mu_1'(az)\mu_2(bz) \tag{6}$$

Alternatively, if $p(y)$ is not given in closed form

$$E\{x_1|y\} = \frac{1}{a}\frac{\int_C dz\, e^{-yz}\mu_1'(az)\mu_2(bz)}{\int_C dz\, e^{-yz}\mu_1(az)\mu_2(bz)}$$

An alternative route uses (5) in (2) to get

$$\begin{aligned} E\{x_1|y\} &= \frac{1}{p(y)b}\int dx\, xp_2\left(\frac{y-ax}{b}\right)\frac{1}{i2\pi}\int_{C_1} d\lambda\; e^{-x\lambda}\mu_1'(\lambda) \\ &= \frac{1}{i2\pi p(y)b}\int_{C_1} d\lambda\; \mu_1'(\lambda)\int dx\, p_2\left(\frac{y-ax}{b}\right)e^{-x\lambda} \\ &= \frac{1}{i2\pi p(y)a}\int_C d\lambda\; \mu_1'(\lambda)\mu_2\left(\frac{b}{a}\lambda\right)e^{-\frac{y}{a}\lambda} \end{aligned}$$

where the same condition as above is required of contour C. The substitution $\lambda = az$ leads to (6). Finally, it is straight forward to consider higher-order moments of the conditional PDF in (1) e.g.

$$E\{x_1^2|y\} = \frac{1}{a}\frac{\int_C dz\, e^{-yz}\mu_1''(az)\mu_2(bz)}{\int_C dz\, e^{-yz}\mu_1(az)\mu_2(bz)}$$

B. SPA for Single Isolated Saddlepoint

Consider the integral

$$I = \frac{1}{i2\pi}\int_C dz\, e^{\phi(z)} \tag{7}$$

where C is a Bromwich contour. Suppose that $\phi'(z_s) = 0$, meaning that z_s is the location of a SP of the integrand. Then,

$$\phi(z) \cong \phi(z_s) + \frac{1}{2}\phi''(z_s)(z - z_s)^2 \tag{8}$$

for z near z_s. Substitution in (7) yields

$$I \cong \frac{e^{\phi(z_s)}}{i2\pi}\int_{C_1} dz\, e^{\frac{1}{2}\phi''(z_s)(z-z_s)^2} \tag{9}$$

where it is presumed that contour C_1 is an equivalent Bromwich contour to C that passes through the point z_s. Let

$$z = z_s + r\,i/\sqrt{\phi''(z_s)}\;,\; dz = dr\,i/\sqrt{\phi''(z_s)}$$

Then, (9) becomes

$$I \cong \frac{e^{\phi(z_s)}}{2\pi\sqrt{\phi''(z_s)}}\int_{-\infty}^{\infty} dr\, e^{-\frac{1}{2}r^2} = \frac{e^{\phi(z_s)}}{\sqrt{2\pi\phi''(z_s)}} \tag{10}$$

This is the SPA for an isolated SP, at which $\phi''(z_s) \neq 0$. Correction terms can be derived by extending the expansion in (8) to higher-order terms. In particular, the first-order correction to the SPA is

$$I \cong \frac{e^{\phi(z_s)}}{\sqrt{2\pi\phi''(z_s)}}\left(1 + \frac{1}{8}\frac{\phi''''(z_s)}{[\phi''(z_s)]^2} - \frac{5}{24}\frac{[\phi'''(z_s)]^2}{[\phi''(z_s)]^3}\right) \tag{11}$$

C. SPA for Two Close Saddlepoints

Consider the integral

$$I = \frac{1}{i2\pi}\int_C dz\, e^{\phi(z)} \tag{12}$$

where C is a Bromwich contour in the original strip of analyticity of the integrand, and

$$\phi'(z_1) = 0\;,\; \phi'(z_2) = 0$$

where z_1 is near (or equal to) z_2. That is, there are two close (or coalesced) saddlepoints of the integrand of (12). Then, it follows that

$$\phi'(z) \cong g\,(z - z_1)(z - z_2) = g\,(z^2 - 2z_m z + z_1 z_2) \tag{13}$$

for z near z_1 and z_2, where g is a constant to be determined, and a midpoint is defined as

$$z_m = \frac{z_1 + z_2}{2}$$

From (13),

$$\phi''(z) \cong 2g(z - z_m)\;,\; \phi'''(z) \cong 2g \tag{14}$$

for z near z_m. Substituting $z = z_m$ in the first relation yields $\phi''(z_m) \cong 0$ which is useless. However, the same substitution in the second relation leads to the selection

$$g \equiv \frac{1}{2}\phi'''(z_m)$$

which is well behaved, even as the two SPs approach and coalesce with each other. It is presumed that $\phi'''(z_m) \neq 0$. Integration of (13) now yields

$$\phi(z) \cong g\left(\frac{1}{3}z^3 - z_m z^2 + z_1 z_2 z\right) + h \tag{15}$$

for z near z_1 and z_2. The constant h is determined by forcing equality in (15) at the midpoint z_m, with the result that

$$h \equiv \phi(z_m) + \frac{1}{6}\,g\,z_m\,(z_1^2 - 4z_1 z_2 + z_2^2)$$

Integral (12) now takes the form

$$I \cong \frac{e^h}{i2\pi}\int_{C_1} dz\, e^{\frac{1}{3}gz^3 - gz_m z^2 + gz_1 z_2 z} \tag{16}$$

where it is presumed that C_1 is an equivalent Bromwich contour to C. Let

$$\begin{aligned} A = g^{1/3} = [\phi'''(z_m)/2]^{1/3}\;,\; z = -t/A \\ \alpha = z_m A\;,\; \beta = z_1 z_2 A^2 \end{aligned} \tag{17}$$

to get

$$I \cong \frac{e^h}{i2\pi A}\int_{C_2} dt\, e^{-\frac{1}{3}t^3 - \alpha t^2 + \beta t} \tag{18}$$

The minus sign in $dz = -dt/A$ has been absorbed into this expression, by switching the direction of integration along

contour C_2. C_2 must take account of the possible complexity of $\phi'''(z_m)$, the three possible roots for A in (17), and the $z = -t/A$ change of variables.

Now let $t = u - \alpha$ in (18) in order to eliminate the quadratic term and get

$$I \cong \frac{e^{h+\alpha(\beta-\frac{2}{3}\alpha^2)}}{i2\pi A}\int_{C_3} du\, e^{-\frac{1}{3}u^3+(\alpha^2-\beta)u}$$

Contour C_3 is simply a shift of contour C_2. Finally, using all the definitions above, along with some algebra, there follows

$$I \cong \frac{e^{\phi(z_m)}}{i2\pi A}\int_{C_3} du\, e^{-\frac{1}{3}u^3+\rho u} \tag{19}$$

where

$$A = \left(\frac{1}{2}\phi'''(z_m)\right)^{1/3}\ ,\ \rho = \frac{1}{4}A^2(z_1 - z_2)^2 \tag{20}$$

As z_1 approaches z_2, ρ tends to zero; however, the integral in (19) is well-defined for all values of ρ, real or complex. Finally, if contour C_3 is arranged to start at infinity anywhere in the angular sector of width $\pi/3$ centered about angle $-2\pi/3$ and end at infinity anywhere in the angular sector of width $\pi/3$ centered about angle $2\pi/3$, then (19) takes the special form

$$I \cong \frac{e^{\phi(z_m)}}{A}Ai(\rho)$$

where $Ai(z)$ is an Airy function. This is the SPA for two close SPs under the conditions cited above.

In order to get correction terms to the SPA in (19), it is necessary to consider additional terms in expansion (14), namely,

$$\phi'''(z) \cong 2g + \phi''''(z_m)(z - z_m) + \frac{1}{2}\phi'''''(z_m)(z - z_m)^2$$

for z near z_m. Then, (15) is modified to read

$$\phi(z) \cong g\left(\frac{1}{3}z^3 - z_m z^2 + z_1 z_2 z\right) + h + \frac{1}{24}\phi''''(z_m)(z - z_m)^4 + \frac{1}{120}\phi'''''(z_m)(z - z_m)^5$$

for z near z_m. This adds a multiplicative factor of

$$e^{\frac{1}{24}\phi''''(z_m)(z-z_m)^4} \cong 1 + \frac{1}{24}\phi''''(z_m)(z - z_m)^4 + \frac{1}{120}\phi'''''(z_m)(z - z_m)^5$$

to (16). This results in a multiplicative factor

$$1 + \frac{1}{24}\frac{\phi''''(z_m)}{A^4}(t + \alpha)^4 - \frac{1}{120}\frac{\phi'''''(z_m)}{A^5}(t + \alpha)^5$$

in (18) and the factor

$$1 + \frac{1}{24}\frac{\phi''''(z_m)}{A^4}u^4 - \frac{1}{120}\frac{\phi'''''(z_m)}{A^5}u^5$$

in (19), namely,

$$I \cong \frac{e^{\phi(z_m)}}{i2\pi A}\int_{C_3} du\, e^{-\frac{1}{3}u^3+\rho u}\left(1 + \frac{\phi''''(z_m)}{24\,A^4}u^4 - \frac{\phi'''''(z_m)}{120\,A^5}u^5\right) \tag{21}$$

Again, if contour C_3 is arranged to start at infinity anywhere in the angular sector of width $\pi/3$ centered about angle $-2\pi/3$ and end at infinity anywhere in the angular sector of width $\pi/3$ centered about angle $2\pi/3$, then (21) takes the special form

$$I \cong \frac{e^{\phi(z_m)}}{A}\left(Ai(\rho) + \frac{\phi''''(z_m)}{24\,A^4}Ai''''(\rho) - \frac{\phi'''''(z_m)}{120\,A^5}Ai'''''(\rho)\right) \tag{22}$$

This is the SPA with correction terms for two close SPs. Contour C_3 is exactly the contour that describes the Airy function $Ai(z)$, see [5, p.52]. The Airy functions are named after the Royal Astronomer, Sir George Biddell Airy. The airy functions are obtained as solutions to Airy's differential equation [5, p.50]

$$\frac{d^2f}{dz^2} - zf = 0$$

They are

$$f_n(z) = \frac{1}{i2\pi}\int_{L_n} dt\, \exp\left(-t^3/3 + zt\right)\ ,\ n = 1, 2, 3$$

When contour L_1 is arranged to start at infinity anywhere in the angular sector of width $\pi/3$ centered about angle $-2\pi/3$ and end at infinity anywhere in the angular sector of width $\pi/3$ centered about angle $2\pi/3$, the solution is called the Airy function of the first kind and is denoted by $Ai(z)$.

III. Examples

Two examples are described. The examples demonstrate how the SPA based on isolated SPs becomes very inaccurate in the region where two SPs coalesce. The examples also demonstrate how the SPA based on nearby or coalesced SPs improves accuracy significantly in this region. It should be noted that for both examples, the conditional mean estimator of interest can be derived analytically in closed form.

A. Example 1: Gaussian RVs

The MGF of a zero-mean gaussian RV with variance σ_x^2 and its first-order derivative with respect to λ is

$$\mu(\lambda) = \exp\left(\sigma_x^2\lambda^2/2\right)\ ,\ \mu'(\lambda) = \sigma_x^2\lambda\exp\left(\sigma_x^2\lambda^2/2\right)$$

Using (6) the conditional mean for Gaussian RVs is given by

$$m_x = \frac{a\sigma_1^2}{i2\pi p(y)}\int_{C_x} dz\, z\, \exp\left(-yz + \sigma^2 z^2/2\right) \tag{23}$$

where

$$\sigma^2 = a^2\sigma_1^2 + b^2\sigma_2^2\ ,\ p(y) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{1}{2}\frac{y^2}{\sigma^2}\right)\ ,\ \text{for all } y$$

The Bromwich contour C_x can be anywhere in the z-plane because the integrand is entire. After substituting $\tilde{z} = \sigma z$, the integral in (23) becomes

$$I = \frac{1}{\sigma}\frac{1}{i2\pi}\int_{C_{\tilde{z}}} d\tilde{z}\,\frac{\tilde{z}}{\sigma}\exp(-y\frac{\tilde{z}}{\sigma} + \frac{\tilde{z}^2}{2})\ ,\ \text{with } m_x = \frac{a\sigma_1^2}{p(y)}I$$

Define $q = y/(2\sigma)$ and let $z = \tilde{z}$; then

$$I = \frac{1}{\sigma}\frac{1}{i2\pi}\int_{C_{\tilde{z}}} dz\, z \exp(-2qz + \frac{z^2}{2})$$

The fundamental integral is then

$$\tilde{I} = \frac{1}{i2\pi}\int_{C} dz\, z\, \exp(-2qz + \frac{z^2}{2}) \tag{24}$$

which depends on only one parameter, namely, $q = y/(2\sigma)$. The integral in (24) can be calculated analytically. Making the substitutions $z = ir$ and $dz = idr$, then

$$\begin{aligned}\tilde{I} =& -\frac{1}{i2\pi}\int_{-\infty}^{\infty} dr\, r\, \exp(-2qir - \frac{r^2}{2}) \\ =& \sqrt{\frac{2}{\pi}}\, q \exp(-2q^2)\end{aligned} \tag{25}$$

Continuing with the SPA to the integral in (24), the integrand, $f(z)$, is

$$f(z) = \exp\left(\phi(z)\right) = \exp\left(-2qz + \frac{z^2}{2} + \log z\right)$$

The following functions are needed in the SPA, the correction term or in finding the SPs

$$\phi(z) = -2qz + \log(z) + z^2/2 \tag{26}$$

$$\phi'(z) = -2q + \frac{1}{z} + z\ ,\ \phi''(z) = 1 - \frac{1}{z^2} \tag{27}$$

$$\phi'''(z) = \frac{2}{z^3}\ ,\ \phi''''(z) = -\frac{6}{z^4}\ ,\ \phi'''''(z) = \frac{24}{z^5} \tag{28}$$

The SPs are found by solving $\phi'(z) = 0$. The integrand in (24) will go to $-\infty$ for negative z and will go to $+\infty$ for z positive. As the integrand goes through zero at $z = 0$ this means that the integrand may not have real SPs. It is known that there must be two SPs and they are found as solutions to the equation

$$z^2 - 2qz + 1 = 0$$

namely,

$$z_s = q \pm \sqrt{q^2 - 1} \tag{29}$$

This means that for $q^2 < 1$, the pair of SPs are complex conjugate; otherwise, there are two real SPs. For $q^2 < 1$, the two complex conjugate SPs are

$$z_1 = q + i\sqrt{1 - q^2}\ ,\ z_2 = q - i\sqrt{1 - q^2} \tag{30}$$

For $q^2 \geq 1$, the two real SPs are

$$z_1 = q + \sqrt{q^2 - 1}\ ,\ z_2 = q - \sqrt{q^2 - 1} \tag{31}$$

Figure 1 shows a trace of the SPs for q varying from -1.2 to 1.2. Initially the SPs start out on the negative real axis and move towards each other. When $q = -1$ the two SPs have coalesced at $z = -1$ into a monkey SP, an SP of order 2, where $\phi''(z_s) = 0$. The SPs then move vertically in opposite directions into the complex z-plane because the SPs are a complex conjugate pair. When $q = 0$ the SPs are pure imaginary. For q becoming more positive, the SPs start to approach each other and become nearly real. When $q = 1$ the SPs are real and have coalesced at $z = 1$; then they move in opposite directions on the positive real axis.

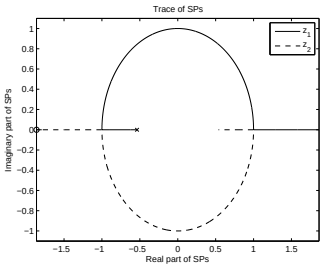

Fig. 1. Plot showing trace of SPs for q going from -1.2 to 1.2. The initial locations of the SPs when $q = -1.2$ are also indicated in the figure by a small circle and cross.

For the case $q^2 < 1$, the SPA can be computed by summing up the contributions to the integral for each of the two SPs. However, because the two SPs are complex conjugates, the contributions will be complex conjugates as well. This means that the SPA can be computed by taking the real part of the contribution for one of the two SPs and multiply this value by two. So in essence, the functions $\phi(z)$ and $\phi''(z)$ need to be evaluated only once. Using the SP, z_1, the total SPA to the integral in (24) is

$$\tilde{I} \cong 2 \cdot \mathrm{Re}\left\{\frac{\exp(\phi(z_1)}{\sqrt{2\pi\phi''(z_1)}}\right\} \tag{32}$$

Inserting the expression for z_1 in (30) into (26) and (27)

$$\begin{aligned}\phi(z_1) =& -\frac{1}{2} - q^2 - iq\sqrt{1 - q^2} + \log\left(q + i\sqrt{1 - q^2}\right) \\ \phi''(z_1) =& \frac{2(q^2 + iq\sqrt{1 - q^2} - 1)}{(q + i\sqrt{1 - q^2})^2}\end{aligned}$$

It is not possible to obtain a simple analytic expression for the SPA to the integral in (24). The ratio between the SPA in (32) and the exact analytic expression for the integral in (24) doesn't lead to a simple expression either.

For the case $q^2 \geq 1$, the real SP, where $f(z)$ is minimum on the real z-axis, is used. The useful SP in (31) is z_1. Inserting the expression for z_1 into (26) and (27)

$$\begin{aligned}\phi(z_1) =& -\frac{1}{2} - q^2 - q\sqrt{q^2 - 1} + \log\left(q + \sqrt{q^2 - 1}\right) \\ \phi''(z_1) =& \frac{2q^2\sqrt{q^2 - 1} + q^4 - 1}{q^2(1 + \sqrt{q^2 - 1})^2}\end{aligned}$$

Using these expressions in the SPA based on isolated SPs and in the ratio between SPA and the exact value of the integral, simple analytical expressions are not obtained. The same is true for the SPA with a correction term.

In figure 2, the accuraces of the SPA and the SPA with a correction term are illustrated. This is done by computing the ratios between the SPAs using (10) and (11) and the exact

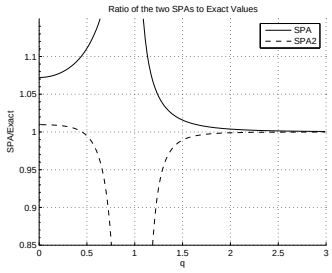

Fig. 2. Ratios of the SPA based on isolated SPs to the exact values of estimator. SPA2 denotes the SPA with a correction term.

values of the integral. The figure reveals that the approximations are particularly inaccurate near the two points where the SPs go from being real to complex; also note that the ratios in figure 2 are symmetric around $q = 0$. The two originally-distinct SPs have coalesced into a SP of order two (a monkey SP), or are close to it. The results for the SPA based on isolated SPs don't apply when two SPs are very close or coalesce. The reason is that on the Path of Steepest Descent (POSD) out of z_1, a Gaussian approximation based on the behavior right at the peak location of z_1, is made. Then, when integrating along that POSD out of z_1, before getting to a region of small magnitude of the integrand, another POSD coming from the other close SP at z_2 is met, and to get to the other SP, z_2, that path of ascent has to be traversed. When the two SPs are close the Gaussian contours from the two SPs are overlapping with the result that the shape of the magnitude of the integrand in the neighborhood of one of the two SPs is not Gaussian, but is rather like a saddle. It is only the behavior of the magnitude of the integrand on the contour of steepest descent that is Gaussian-like.

The SPA based on two close SPs is now applied. The midpoint is $z_m = q$ and inserting into (26)-(28), there follows

$$\phi(z_m) = \log(q) - \frac{3}{2}q^2 \ , \ \phi''(z_m) = 1 - \frac{1}{q^2} \qquad (33)$$
$$\phi'''(z_m) = \frac{2}{q^3} \, , \, \phi''''(z_m) = -\frac{6}{q^4} \, , \, \phi'''''(z_m) = \frac{24}{q^5}$$

By reference to (20), take $A = 1/q$ which is real, but can be positive or negative, depending on q. Then, using (20), (29), there follows

$$\rho = 1 - \frac{1}{q^2}$$

If $q > 0$, then $A > 0$. Bend both ends of contour C in (24) into the right-half z-plane so that they asymptotically approach infinity with angles $\pm\pi/3$. This is contour C_1 described in (16). The substitution $z = -t/A$, $dz = -dt/A$ in (17) then results in t-plane contour C_2 in (18) asymptotically starting from infinity with angle $-2\pi/3$ and ending at infinity with angle $2\pi/3$, which is the desired contour for the $Ai(z)$ function representation.

If $q < 0$, then $A < 0$. Bend both ends of contour C in (24) into the left-half z-plane so that they asymptotically approach infinity with angles $\pm 2\pi/3$. This is contour C_1 described in (16). The substitution $z = -t/A = t/|A|$, $dz = -dt/A = dt/|A|$, in (17) then results in t-plane contour C_2 in (18) also asymptotically starting from infinity with angle $-2\pi/3$ and ending at infinity with angle $2\pi/3$, which is the desired contour for the $Ai(z)$ function representation. However, the factor A in the denominator in (18) is now replaced by $|A|$.

Both polarities of q and A can be accommodated in this example by replacing the A in the denominator of (18) by $A\,\mathrm{sgn}(q)$. When the results in (29), (33) are substituted into (22), the end result for the SPA is

$$I \sim \qquad (34)$$
$$\mathrm{sgn}(q) q^2 \exp\left(-\frac{3}{2}q^2\right)\left[Ai(\rho) - \frac{Ai''''(\rho)}{4} - \frac{Ai'''''(\rho)}{5}\right]$$

The ratio between the expression in (34) and the exact value of the integral in (25), gives, for all q,

$$R(q) = \sqrt{\frac{\pi}{2}}\,|q|\,\exp\left(\frac{1}{2}q^2\right)\left[Ai(\rho) - \frac{Ai''''(\rho)}{4} - \frac{Ai'''''(\rho)}{5}\right]$$

The region where this ratio $R(q)$ is of interest is for $q \sim 1$ according to (29). A plot of ratio $R(q)$ for $0.8 < q < 1.2$ reveals a maximum variation from 0.992 to 1.07 from its ideal value of 1 for $R(q)$, see figure 3. In fact, at $q = 1$ where the two SPs coalesce, $R(q) = 1.001$. Thus, SPA (34) affords an excellent estimate of the contour integral (24) for the situation where the two SPs are close or have coalesced.

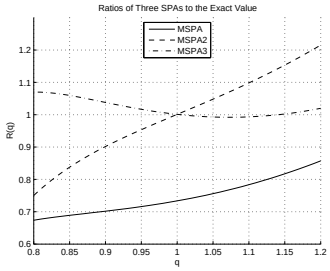

Fig. 3. Ratio of the exact value to the SPAs based on close or coalesced SPs. MSPA denotes the SPA based on two close SPs. MSPA2 denotes the SPA based on two close SPs with a correction term. MSPA3 denotes the SPA based on two close SPs with two correction terms.

B. Example 2: Laplace RVs

A univariate laplace distributed zero-mean RV has a PDF given by

$$p(x) = \frac{1}{2\beta_x}\exp\left(-\frac{|x|}{\beta_x}\right) \ , \ \beta_x > 0 \qquad (35)$$

For this Laplace-distributed RV, the MGF is

$$\mu_x(\lambda) = E\{\exp(\lambda x)\} = \frac{1}{1-\beta_x^2\lambda^2}\ ,\ |Re(\lambda)| < 1/\beta_x$$

and the first-order derivative with respect to λ is

$$\mu_x'(\lambda) = \frac{2\beta_x^2\lambda}{(1-\beta_x^2\lambda^2)^2}$$

Inserting into (6), the integrand is

$$f(z) = \exp\left(\phi(z)\right) = \frac{2a\beta_1^2 z\exp(-yz)}{(1-b^2\beta_2^2z^2)(1-a^2\beta_1^2z^2)^2}$$

The conditional mean is then given by

$$m_x = \frac{2a\beta_1^2}{i2\pi p(y)}\int_{C_x} dz\ \exp(-yz)\frac{z}{(1-\frac{z^2}{c^2})(1-\frac{z^2}{d^2})^2} \quad (36)$$

where y, c and d are real positive constants, $c = 1/(b\beta_2)$ and $d = 1/(a\beta_1)$ and

$$p(y) = \frac{\beta_1\beta_2\left(e^{\frac{\beta_1 y}{a}}\beta_1 b - e^{\frac{\beta_2 y}{b}}\beta_2 a\right)e^{-\frac{y(\beta_1 b+\beta_2 a)}{ab}}}{2(\beta_1^2b^2-\beta_2^2a^2)}\ ,\ y>0$$

After substituting $z = c\tilde{z}$, the integral in (36) becomes

$$I = \frac{1}{i2\pi}\int_{C_{\tilde{x}}} d\tilde{z}\ \exp(-yc\tilde{z})\frac{c^2\tilde{z}}{(1-\tilde{z}^2)(1-\frac{c^2\tilde{z}^2}{d^2})^2} \quad (37)$$

After making the substitutions $A = yc$ and $B = c/d$ in (37), the fundamental integral to approximate is

$$\tilde{I} = \frac{1}{i2\pi}\int_C dz\ \exp(-Az)\frac{z}{(1-z^2)(1-B^2z^2)^2} \quad (38)$$

The exact value of this integral will depend on the two fundamental parameters A and B. Therefore, the ratio of the SPA to the exact value will depend only on A and B: $R(A,B)$. The integral in (38) can be evaluated analytically using residues. For the pole $z = 1$ the residue is

$$res_1 = -\frac{\exp(-A)}{2(B^2-1)^2}$$

and for the double pole at $z = 1/B$ the residue is

$$res_2 = \frac{\exp(-A/B)\left(A+2B-AB^2\right)}{4B(B^2-1)^2}$$

and the exact value for the integral is then

$$\tilde{I} = -res_1 - res_2 \quad (39)$$

Bromwich contour C in (38) crosses the real axis of the z-plane in the original strip of analyticity between the points $\pm x_0$, $x_0 = \min(1, 1/B)$. This integrand has simple poles at ± 1 and double poles at $\pm 1/B$. The integrand in (38) is then extended to the rest of the z-plane by analytic continuation (AC). Since parameter A is positive, the two ends of contour C can be bent to the right in the z-plane, so that the new contour, C_1, starts at infinity with angle $-\pi/3$ and ends up at infinity with angle $\pi/3$, still passing between the points $\pm x_0$. Contour C_1 is equivalent to C.

The $\exp(-Az)$ function dominates the behavior of the integrand for large z. For $B < 1$ the original contour C is in the interior of $(-1, 1)$ which is the original ROA. For z real and positive the integrand starts at a value 0 for $z = 0$ and increases to $+\infty$ as $z \to 1$ from the left. For z real and negative, the integrand starts at a value 0 and decreases to $-\infty$ for $z \to -1$ from the right. This means that there is no real SP inside the ROA unless A is large enough for the function $\exp(-Az)$ to dominate inside the ROA so that a real and positive SP is created inside the ROA. One way to deal with a missing SP inside the ROA is to extend the integrand to all z by means of AC and move the contour of integration outside the original ROA towards $+\infty$ in the complex z-plane because the function $\exp(-Az)$ decays rapidly to zero there. However, this contour is not equivalent to the original contour because a pole has been crossed. Another possibility is to make use of complex SPs in the complex z-plane to obtain a SPA.

The SPA makes use of the following functions

$$\begin{aligned}
\phi(z) &= -Az + \log(z) - \log(1-z^2) - 2\log(1-B^2z^2)\\
\phi'(z) &= -A + \frac{1}{z} + \frac{2z}{1-z^2} + \frac{4B^2z}{1-B^2z^2}\\
\phi''(z) &= -\frac{1}{z^2} + \frac{2(1+z^2)}{(z^2-1)^2} + \frac{4B^2(B^2z^2+1)}{(B^2z^2-1)^2}\\
\phi'''(z) &= \frac{2}{z^3} - \frac{4z(3+z^2)}{(z^2-1)^3} - \frac{8B^4z(3+B^2z^2)}{(B^2z^2-1)^3}\\
\phi''''(z) &= -\frac{6}{z^4} + \frac{12(z^4+6z^2+1)}{(z^2-1)^4}\\
&\quad + \frac{24B^4(B^4z^4+6B^2z^2+1)}{(B^2z^2-1)^4}\\
\phi'''''(z) &= \frac{24}{z^5} - \frac{48z(z^4+10z^2+5)}{(z^2-1)^5}\\
&\quad - \frac{96B^6z(B^4z^4+10B^2z^2+5)}{(B^2z^2-1)^5}
\end{aligned}$$

The SPs are found by solving for z in the equation

$$AB^2z^5 + 5B^2z^4 - A(B^2+1)z^3 - (3B^2+1)z^2 + Az - 1 = 0$$

This polynomial of degree 5 in z cannot be factored (by the Matlab symbolic toolbox). So, analytic expressions for the locations of the SPs in the complex z-plane cannot be found, although we know there must be 5 of them. The SPs can be found numerically by assigning numerical values to the parameters A and B and using Matlab's roots routine. For $B < 1$ and a large value of A in $\exp(-Az)$, there is 1 SP in $(-\infty, -1/B)$, 1 SP in $(-1/B, -1)$, 2 SPs in $(0, 1)$, and 1 SP in $(1, 1/B)$. These are all on the real z-axis. As A decreases, the two real SPs in $(0, 1)$ coalesce and then separate vertically into the complex z-plane. It is not known if their real parts remain in $(-1, 1)$. For $B > 1$ and large A, there is 1 SP in $(-\infty, -1)$, 1 SP in $(-1, -1/B)$, 2 SPs in $(0, 1/B)$, and 1 SP in $(1/B, 1)$. All are real. Again, as A decreases, they coalesce and separate vertically and it is not known if they remain in the original ROA. However, this analysis suggests that a SPA can be applied by using SPs in the complex z-plane when a real SP in the original ROA doesn't exist. We must make sure that the corresponding POSD through the two complex conjugate SPs is equivalent to the Bromwich contour confined to the original ROA.

Figure 4 shows the POSD for the example with two complex conjugate SPs in the ROA (see [6] for how to find POSDs). The two crosses are the locations of the two complex conjugate SPs in the ROA. The circles show the POSDs out of the SPs away from the origin and the stars show the POSDs out of the SPs towards the origin. Notice how both POSDs out of the SPs in the first and second quadrants head directly into the zero at the origin. This two-pieced path is an equivalent contour to the original Bromwich contour, and SPA or numerical integration can be used.

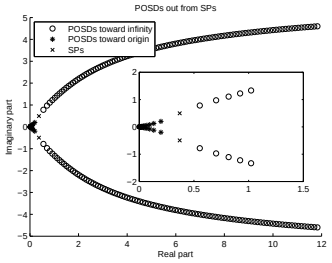

Fig. 4. POSDs out of the SPs in the complex z-plane.

The SPA is computed as in example I where two complex conjugate SPs are used when there is no real SP in the ROA. Plots illustrating the ratio $R(A, B)$ for the SPA and the SPA with a correction term (SPA2) are seen in figure 5. It is seen that throughout the AB-plane, the ratios are close to 1 as expected, except for a ridge that identifies the values of A and B where two close or coalesced SPs make the SPA based on isolated SPs inaccurate. In figure 6, the ratio is illustrated for the case $B = 0.38$.

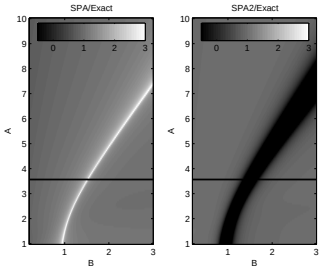

Fig. 5. Ratio $R(A, B)$ for the SPA and the SPA with a correction term (SPA2.

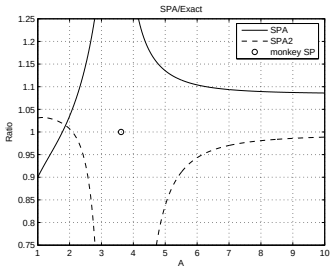

Fig. 6. Comparison of ratios for the SPA and the SPA with a correction term (SPA2) for the case $B = 0.38$.

With the same setting, $B = 0.38$, the SPA based on two close or coalesced SPs is computed in the interval of A where the SPA based on isolated SPs is particular inaccurate. The results are seen in figure 7. From the figure it is clear that the SPA based on two close or coalesced SPs with two correction terms results in an excellent approximation in the region where the SPA based on isolated SPs is particularly inaccurate.

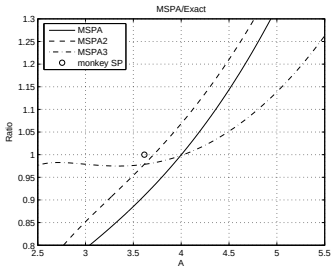

Fig. 7. Ratio of the exact values of the estimator to the SPAs based on close or coalesced SPs. MSPA denotes the SPA based on two close SPs. MSPA2 denotes the SPA based on two close SPs with a correction term. MSPA3 denotes the SPA based on two close SPs with two correction terms..

IV. Summary

Given a linear combination of two random variables, the possibility of using a saddlepoint approximation for minimum mean-square error estimation of either of the two random variables in the linear combination, has been demonstrated. The approach was based on a derived integral representation for the minimum mean-square error estimator involving the

moment-generating functions of the random variables. Furthermore, it was demonstrated that for the case of two coalescing saddlepoints, the saddlepoint approximation based on isolated saddlepoints became very inaccurate and if a saddlepoint approximation based on two coalescing saddlepoints with two correction terms was used instead, this saddlepoint approximation makes an excellent approximation.

ACKNOWLEDGMENTS

The first author would like to thank Professor Peter Willett for providing the author the opportunity to make a stay at the University of Connecticut (UConn) and thereby make this work possible. The first author would also like to thank Tod E. Luginbuhl, Phillip L. Ainsleigh, Paul M. Baggenstoss and Peter Willett for making the stay at UConn a warm and insightful experience.

REFERENCES

[1] L. Benaroya, F. Bimbot, and R. Gribonval, "Audio source separation with a single sensor," *IEEE Trans. Speech Audio Processing*, vol. 14, pp. 191–199, Jan. 2006.

[2] H. E. Daniels, "Saddlepoint approximations in statistics," *The Annals of Mathematical Statistics*, vol. 25, no. 4, pp. 631–650, Dec 1954.

[3] A. H. Nuttall, "Saddlepoint approximation and first-order correction term to the joint probability density function of m quadratic and linear forms in k gaussian random variables with arbitrary means and covariances," NUWC, Technical Report 11262, December 2000.

[4] ——, "Saddlepoint approximations for the combined probability and joint probability density function of selected order statistics and the sum of the remainder," NUWC, Technical Report 11509, February 2004.

[5] N. Bleistein and R. A. Handelsman, *Asymptotic Expansions of Integrals*. Dover Publications, 1986.

[6] A. H. Nuttall, "Evaluation of the generalized qm function for complex arguments," NUWC, Technical Report 11708, August 2005.

Thomas Beierholm was born in Aalborg, Denmark, in September 1976. He received the M.Sc. degree from the Department of Communication Technology, Aalborg University, Aalborg, Denmark, in 2001. From 2001 to 2004 he worked as a DSP software developer in the company GN ReSound A/S.

He is currently employed in GN ReSound A/S as an Industrial PhD student. His research interests include digital signal processing theory and methods with application to speech and audio, in particular signal separation and noise reduction.

Albert H. Nuttall received the B.Sc., M.Sc., and Ph.D. degrees in electrical engineering from the Massachusetts Institute of Technology (MIT), Cambridge, in 1954, 1955, and 1958, respectively.

He was with MIT as an Assistant Professor until 1959. From 1957 to 1960, he was with Melpar, and from 1960 to 1968, he was with Litton Industries. He was with the Naval Underwater Systems Center (NUSC), New London, CT, where his interest was in statistical communication theory. He is now with the Naval Undersea Warfare Center, Newport, RI.

Dr. Nuttall received the NUSC Distinguished Chair in Signal Processing from the Navy in April 1987.

Lars K. Hansen received the M.Sc. and Ph.D. degrees in physics from the University of Copenhagen, Copenhagen, Denmark, in 1984 and 1986, respectively.

He was with Andrex Radiation Products A/S, from 1987 to 1990. Since 1990, he has been with the Technical University of Denmark, Lyngby, and since 2000 as a Full Professor of signal processing. He is the Director of the THOR Center for Neuroinformatics and Head of the Copenhagen Image and Signal Processing Graduate School (CISP). His current research interests include modeling of adaptive systems, meta-analysis, and applications in neuroimaging.

Appendix F

Particle Filter Inference in an Articulatory Based Speech Model

Accepted for publication in IEEE Journal of Signal Processing Letters.

JOURNAL OF SIGNAL PROCESSING LETTERS, VOL. X, NO. XX, NOVEMBER 2006 1

Particle Filter Inference in an Articulatory Based Speech Model

Thomas Beierholm (*), *Student Member, IEEE*, and Ole Winther

***Abstract*— A speech model parameterized by formant frequencies, formant bandwidths and formant gains is proposed. Inference in the model is made by particle filtering for the application of speech enhancement. The advantage of the proposed parameterization over existing parameterizations based on AR coefficients or reflection coefficients is the smooth time-varying behavior of the parameters and their loose coupling. Experiments confirm this advantage both in terms of parameter estimation and SNR improvement.**

***Index Terms*— Particle filtering, time-varying auto-regressive speech model, formant frequency.**

I. Introduction

In the application of speech enhancement, the speech signal is commonly modeled as a time-varying Auto-Regressive (AR) Gaussian process. In block-processing systems the speech signal is assumed quasi-stationary meaning that the parameters of the AR process describing the speech signal are assumed fixed in the duration of the block. As described in Ref. [1] the articulators of speech, such as the vocal tract, are continually moving, hence the assumption of quasi-stationarity of speech can be improved upon. The Time-Varying Auto-Regressive (TVAR) model used in Refs. [1], [2] lets the parameters of the AR process describing the speech signal vary from sample to sample and thus avoids the assumption of quasi-stationarity of the speech signal.

The TVAR model facilitates a state-space formulation of the observed noisy signal in which the problem of joint estimation of the unknown parameters of the model and the state sequence becomes a challenge. One approach is to perform ML estimation using the EM algorithm. A different approach was used in Refs. [1], [2], where sequential Bayesian estimation of the unknown parameters and state sequence was performed by particle filtering. Instead of using the AR coefficients directly, then, in Ref. [2] the TVAR model was reparameterized in terms of reflections coefficients as this lead to a stronger physical interpretation of the model and stability of the model could easily be verified.

In this paper a model similar to the TVAR model but with an even stronger physical interpretation is used [3]. The model is parameterized in terms of formant frequencies, formant bandwidths and formant gains, called the fbg parameters in the following. It is intended that this new parameterization can lead to improved particle filtering by way of exploiting known properties of the fbg parameters and thereby eventually improve quality of the estimated speech signal. As stressed in Ref. [3] the new parameters have a slow time variation due to the inertia of the speech producing system in contrast to the reflection coefficients which can have a rapid time variation. The new parameters are also loosely coupled, exhibit smooth trajectories and the stability of the model is thus easily ensured.

A common feature of the TVAR model [1] and the fbg parameterized model introduced in this work is that conditional on the unknown parameters of the model, the model reduces to a linear Gaussian state-space system. In Ref. [1] this feature was made use of in a variance reduction (Rao-Blackwellization) step whereby the problem of sampling from the joint posterior distribution of the states and the unknown parameters of the model is reduced to that of sampling from the posterior distribution of the unknown parameters only.

In short, the contributon of this paper is to introduce a speech model with an even stronger physical interpretation than the reflection coefficients [2] and obtain filtered estimates of the clean speech signal with Rao-Blackwellized particle filtering [1].

Manuscript received July xx, 2006; revised December xx, 2006.

T. Beierholm is with GN ReSound A/S, Lautrupbjerg 7, P.O. Box 99, 2750 Ballerup, Denmark, email tbe@imm.dtu.dk.

O. Winther is with IMM, Danish Technical University, 2800 Lyngby, Denmark.

II. Time-Varying Auto-Regressive Model

In the Time-Varying Auto-Regressive (TVAR) model a speech signal is modeled as a non-stationary AR(p) process, where p denotes the order of the AR process which is assumed fixed in the following. The coefficients of the AR(p) process and the variance of the process noise are allowed to change from sample to sample, i.e.

$$x[n] = \sum_{i=1}^{p} a_i[n]x[n-i] + \sigma_e[n]e[n] \quad , \quad e[n] \sim \mathcal{N}(0,1) \ ,$$

where $\sigma_e^2[n]$ is the variance of the innovation sequence and $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes a Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. It is assumed that the speech signal is contaminated by non-stationary Gaussian noise

$$y[n] = x[n] + \sigma_d[n]d[n] \quad , \quad d[n] \sim \mathcal{N}(0,1) \ ,$$

where $\sigma_d^2[n]$ denotes the variance of the observation noise. The TVAR model is conveniently formulated as a state-space model with the following state and observation equations

$$\mathbf{x}[n] = \mathbf{A}[n]\mathbf{x}[n-1] + \mathbf{B}[n]\mathbf{v}[n] \ , \ \mathbf{v}[n] \sim \mathcal{N}(\mathbf{0}_{p\times 1}, \mathbf{I}_p) \quad (1)$$

$$y[n] = \mathbf{C}\mathbf{x}[n] + D[n]w[n] \ , \ w[n] \sim \mathcal{N}(0,1) \ , \quad (2)$$

where $\mathbf{a}[n] = (\, a_1[n], \ \cdots \ , \ a_p[n] \,)^T$ is the coefficient vector, $\mathbf{x}[n] = (\, x[n], \ \cdots \ , \ x[n-p+1] \,)^T$ the state vector and

$$\mathbf{A}[n] = \begin{pmatrix} \mathbf{a}^T[n] \\ \mathbf{I}_{(p-1)} \quad \mathbf{0}_{(p-1)\times 1} \end{pmatrix} \ , \ \mathbf{B}[n] = \begin{pmatrix} \sigma_e[n] \\ \mathbf{0}_{(p-1)\times 1} \end{pmatrix} \quad (3)$$

$$\mathbf{C} = \begin{pmatrix} 1\,0\cdots 0 \end{pmatrix} \ , \ D[n] = \begin{pmatrix} \sigma_d[n] \end{pmatrix} \ . \tag{4}$$

The state-space formulation of the TVAR model in eqs. (1) and (2), with the parameterization eqs. (3) and (4), is used in Ref. [1]. The unknown parameters of the model are the p AR coefficients in $\mathbf{a}[n]$ and the innovation and observation noise variances. The AR coefficients and the two noise variance parameters represented by their logarithms were assumed independent and taken as evolving according to first-order Markov random walk processes. The variance of the random walk processes for each of the AR coefficients and the variances of the random walk processes for the logarithms to the variance of the innovation sequence and observation noise are denoted δ_a, δ_e, δ_d, respectively.

III. Articulatory-Based Speech Model

The speech model based on the fbg parameters (which are close to the articulators of speech [3]), is referred to as a *Parallel Formant Synthesizer* (PFS) in speech synthesis. A PFS synthesizes speech by summing the outputs of a number of parallel connected resonance circuits. The structure of a PFS is illustrated in Fig. 1. The resonators are driven by a common excitation signal which is taken to be white standard normal distributed noise. Each resonance circuit models a formant in the spectrum of the speech signal, in the sense that the spectrum of the excitation signal is shaped to have a peak at the resonance frequency and the bandwidth and gain of the 'bump' is determined by the resonance circuit as well. The resonators are taken as second-order IIR filters with z-transforms

$$H_k(z) = \frac{g_k}{1 - a_{k,1}(f_k, b_k)z^{-1} - a_{k,2}(b_k)z^{-2}} \ , \tag{5}$$

where f_k, b_k and g_k denotes the formant frequency, formant bandwidth and formant gain, respectively, of the k^{th} formant. The mapping from these parameters to the coefficients of the resonators is given by [4], [5]

$$a_{k,1}(f_k, b_k) = 2\exp(-\pi b_k/f_s)\cos(2\pi f_k/f_s) \tag{6}$$

$$a_{k,2}(b_k) = -\exp(-2\pi b_k/f_s) \ , \tag{7}$$

where f_s is the sampling frequency in Hz. By letting

$$\mathbf{A}_k[n] = \begin{pmatrix} a_{k,1}(f_k, b_k) & a_{k,2}(b_k) \\ 1 & 0 \end{pmatrix} \tag{8}$$

then in state-space form, the PFS model is described by the TVAR model in eqs. (1) and (2) with parameterization

$$\mathbf{A}[n] = \text{diag}\left(\mathbf{A}_1, \cdots, \mathbf{A}_K\right) \tag{9}$$

$$\mathbf{B}[n] = \begin{pmatrix} \mathbf{0}_{1\times 2K} \\ g_1[n]\,0 \quad \cdots \quad g_K[n]\,0 \end{pmatrix}^T \tag{10}$$

$$\mathbf{C} = \begin{pmatrix} 1\,0\,1\,0\cdots 1\,0 \end{pmatrix} \ , \ D[n] = \begin{pmatrix} \sigma_d[n] \end{pmatrix} \ , \tag{11}$$

where $\mathbf{x}[n] = (x_1[n],\ x_1[n-1],\ \cdots,\ x_K[n],\ x_K[n-1])^T$ is the state vector, K is the number of formants, $x_k[n]$ denotes the output from the k^{th} resonator (see figure 1) and $\mathbf{v}[n] \sim \mathcal{N}(\mathbf{0}_{2\times 1}, \mathbf{I}_2)$. A formulation based on a cascade structure of second-order sections can also be used. In fact the TVAR model is equivalent to such a cascade structure hence the two main differences between the TVAR and the PFS models are the different structures, cascade and parallel respectively, and the parameterization. As opposed to the cascade structure the parallel structure also contains a zero which means that the PFS model can improve modeling of speech sounds containing anti-resonances. A description of the pros and cons of the cascade and parallel structures in speech synthesis can be found in Ref. [5]. The PFS model has $3K+1$ parameters. The

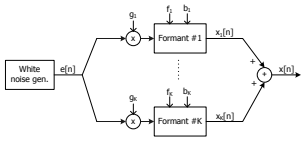

Fig. 1. Block diagram of Parallel Formant Synthesizer.

f_k and b_k parameters and the logarithm to the g_k parameters are assumed independent and taken as evolving according to first-order Markov random walk processes with variances δ_f, δ_b and δ_g, respectively.

IV. Particle Filter Inference

We provide a brief summary of the particle filter method used for inference in the PFS model as it is described in detail elsewhere [1]. Generally, filtering refers to the task of drawing samples from the *filtering distribution* $p(\mathbf{x}_n, \boldsymbol{\theta}_n \,|\, \mathbf{y}_{1:n})$ in order to estimate the mean (the MMSE estimate) of the state vector and the parameter vector. In the context of the PFS model

$$\boldsymbol{\theta} = (f_1,\, b_1,\, g_1,\, \cdots,\, f_K,\, b_K,\, g_K,\, \sigma_d).$$

The joint distribution is decomposed using Bayes rule:

$$p(\mathbf{x}_n, \boldsymbol{\theta}_{0:n} \,|\, \mathbf{y}_{1:n}) = p(\mathbf{x}_n \,|\, \boldsymbol{\theta}_{0:n}, \mathbf{y}_{1:n})\, p(\boldsymbol{\theta}_{0:n} \,|\, \mathbf{y}_{1:n}) \ . \tag{12}$$

For both models (TVAR and PFS) the state vector can be integrated out analytically because $p(\mathbf{x}_n \,|\, \boldsymbol{\theta}_{0:n}, \mathbf{y}_{1:n})$ is Gaussian. This so-called Rao-Blackwellization has the effect of reducing the variance of the MMSE estimate of the state and parameter vectors. The problem is then reduced to sampling from the lower dimensional distribution $p(\boldsymbol{\theta}_{0:n} \,|\, \mathbf{y}_{1:n})$ instead of sampling from $p(\mathbf{x}_{0:n}, \boldsymbol{\theta}_{0:n} \,|\, \mathbf{y}_{1:n})$. In particle filtering this distribution is approximated by a weighted sum of δ-functions (the particles). The importance weight of particle with history $\boldsymbol{\theta}_{0:n}$ is given by

$$w(\boldsymbol{\theta}_{0:n}) \propto \frac{p(\boldsymbol{\theta}_{0:n} \,|\, y_{1:n})}{\pi(\boldsymbol{\theta}_{0:n} \,|\, y_{1:n})} \ , \tag{13}$$

where $\pi(.)$ denotes the importance distribution where samples are drawn from. Sequential importance sampling can be performed if the importance distribution is restricted to be of the general form

$$\pi(\boldsymbol{\theta}_{0:n} \,|\, y_{1:n}) = \pi(\boldsymbol{\theta}_{0:n-1} \,|\, y_{1:n-1})\pi(\boldsymbol{\theta}_n \,|\, \boldsymbol{\theta}_{0:n-1}, y_{1:n}) \tag{14}$$

which facilitates recursive propagation of the importance weights in time. The crucial restriction is that the time dependence only goes to $n-1$ in the first term. Inserting eq. (14) in

eq. (13) and expanding the numerator using Bayes' rule and using the assumption that the parameters evolve according to a first-order Markov process, i.e. $p(\boldsymbol{\theta}_n|\boldsymbol{\theta}_{0:n-1})$=$p(\boldsymbol{\theta}_n|\boldsymbol{\theta}_{n-1})$, then the weights obey $w(\boldsymbol{\theta}_{0:n}) \propto w(\boldsymbol{\theta}_{0:n-1})w_n$ with

$$w_n \propto \frac{p(y_n|\boldsymbol{\theta}_{0:n}, y_{1:n-1})p(\boldsymbol{\theta}_n|\boldsymbol{\theta}_{n-1})}{\pi(\boldsymbol{\theta}_n \,|\, \boldsymbol{\theta}_{0:n-1}, y_{1:n})} \quad . \tag{15}$$

This way sequential importance sampling avoids the need for storing the paths $\boldsymbol{\theta}_{0:n-1}$ of the particles. Using (15) directly, then for each time step the procedure of the particle filter is to draw N samples from the chosen importance distribution (the denominator in (15)) and then compute the weights by direct evaluation of (15). However, the complexity of computing the weights can be simplified if the importance distribution at time n is set equal to the prior distribution, i.e. $\pi(\boldsymbol{\theta}_n \,|\, \boldsymbol{\theta}_{0:n-1}, y_{1:n}) = p(\boldsymbol{\theta}_n|\boldsymbol{\theta}_{n-1})$ so that (15) reduces to $w_n \propto p(y_n|\boldsymbol{\theta}_{0:n}, y_{1:n-1})$. This way the procedure of the particle filter is to draw N samples from the prior distribution (second factor in numerator in (15)) which is relative simple because of the first-order Markov process assumption.

Setting the importance distribution equal to the prior distribution contributes to a degeneracy whereby all weights except one after a few time steps are very close to zero. This happens because the importance distribution is different from the true posterior distribution. As a remedy, a resampling step is introduced. The resampling step duplicates particles in proportion to their importance weights in such a way that all particles have approximately the same weight after the resampling step.

V. Experiments

The performance of the PFS model is examined and compared to the TVAR model examined by Vermaak et al. in [1]. The starting point is the TIMIT speech sentences "In simpler terms, it amounts to pointing the platform in the proper direction." (si1466) and "His sudden departure shocked the cast." (sx111) both downsampled to 16 kHz. From each sentence a 0.38s (6000 samples) speech sound is extracted. The waveforms of the extracted sounds are seen in Fig. 2. From the wide-band spectrogram of the waveform shown in the upper plot of Fig. 2 it was evident that by low-pass filtering $F1$ (the first formant) could be separated from the other formants. A modified speech sound was created where frequencies above 1100 Hz were suppressed and only $F1$ existed in the modified speech sound. In the same manner a speech sound with only $F1$ and $F2$ was created. Those two modified sounds were contaminated by zero-mean stationary white Gaussian noise at 0 dB SNR and subsequently used to manually tune the random walk parameters for both the PFS and the TVAR model so that the particle filtering gave as high SNR improvements as possible. The particle filtering used by Ref. [1] was modified in two respects; 1) so that it exploited that the variance of the observation noise was known and constant and 2) it was initialized using f and b parameters that were then mapped to AR coefficients using eqs. (6) and (7). In this way the initializations in the TVAR model and the PFS model were alike. The first 800 samples were not used in the computation of the SNR improvements

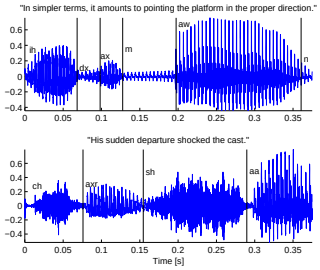

Fig. 2. Plot of the waveforms for extracted sounds from two TIMIT sentences. Phonetic information is also shown in the plots.

in order to minimize initialization effects. The manual tuning of the random walk parameters lead to the following setting δ_a=2.5×10^{-5} and δ_e=5×10^{-3} in the TVAR model and the setting δ_f=20, δ_b=7 and δ_g=5×10^{-3} in the PFS model. Both the performance of the TVAR model and the PFS model was found to be relative insensitive to the setting of the random walk parameters.

With these settings and using 100 particles the particle filter was run on the sound where only $F1$ exists. As also observed in Ref. [1] using more than 100 particles produced only small performance improvements. The experiment was repeated 7 times. The TVAR model was specified to use 2 AR coefficients and in the PFS model K=1. The mean SNR improvement measured for the TVAR model was 6.27 dB and the SNR improvement measured for the PFS model was 7.21 dB. The PFS model provided slightly higher but consistent SNR improvements for this setup. Halving the value of the random walk parameters δ_f and δ_b produced a mean SNR improvement of 7.12 dB and doubling them produced 7.16 dB. The value of these parameters could be changed at least an order of magnitude and still produce higher SNR improvements than that of the TVAR model which favors the PFS model as being a better speech model than the TVAR model.

In the particle filter the unknown parameters are augmented to the state vector. In this way the particle filter provides estimates of the unknown parameters. Using *Praat* [6] the formant frequency tracks were extracted from the clean speech. The 'true' $F1$ formant frequency is illustrated in the upper plot of Fig. 3 together with the estimated $F1$ formant frequency tracks using the TVAR model and the PFS model. The estimated formant frequency tracks were obtained by averaging the estimates from 7 repeated experiments. By using the inverse mapping in eqs. (6) and (7) the formant frequency track for the TVAR model was computed from the estimated AR coefficients. As is evident from Fig. 3a the PFS model provides a much better estimate of the formant frequency than the TVAR model. It is also seen from Fig. 3b that the PFS model provides a more smooth and accurate estimate of the AR coefficients. Next, performance was measured for the sound with $F1$ and

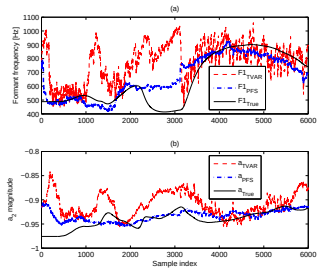

Fig. 3. (a) Estimated $F1$ formant frequency track using the TVAR and PFS models and the 'true' formant frequency extracted from clean speech. (b) Estimated tracks of the a_2 AR coefficient together with the 'true' values.

$F2$ using the same conditions as for the sound with $F1$ only. The PFS model and the TVAR model gave 6.24 dB and 5.41 dB mean SNR improvements, respectively. The PFS model provided slightly higher and consistent SNR improvements for this setup also. The estimated formant frequency tracks using the models are seen in Fig. 4. This experiment illustrated even more the convenience of the PFS model over the TVAR model in that it's much more straight forward to use the properties of the PFS model to ensure reasonable behavior of particle paths. It is for instance more cumbersome to initialize the TVAR model so that the formant frequencies of the particle paths get in range with the formants of the sound. It is also significantly more cumbersome to ensure that the particle paths of the TVAR model remain within the limits of the range of $F1$ and $F2$. If this is not ensured the estimated spectrum of the sound using the TVAR model can have a low-pass characteristic or a single peak instead of two peaks. Performance was then

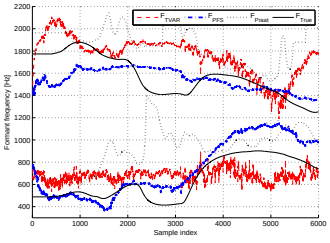

Fig. 4. Estimated $F1$ and $F2$ formant frequency tracks using the TVAR and PFS models and the 'true' formant frequencies extracted from clean speech. On the figure are also shown the formant frequency tracks extracted using Praat on the noisy signal.

measured for the two fullband waveforms shown in Fig. 2.

TABLE I

MEASURED DB SNR IMPROVEMENTS FOR THE TVAR AND PFS MODELS FOR 4 DIFFERENT SNRS USING THE SI1466 AND SX111 TIMIT SOUNDS.

	si1466				sx111			
Model	0	5	10	20	0	5	10	20
TVAR	5.20	3.06	1.03	0.24	3.80	1.59	0.92	−0.02
PFS	5.69	4.54	3.07	1.43	4.82	2.92	1.62	0.42

SNR improvements were measured for 4 different SNRs using both models and same conditions as in previous experiments. The results are seen in Table I. The PFS model produced higher SNR improvements for both sounds and all 4 SNRs. It's also seen that there is a negative correlation between measured dB SNR improvement and SNR. As a last experiment particle filtering was performed on the full length TIMIT waveforms in order to test the quality of the enhanced speech signals. The signals were degraded by non-stationary noise at 0 dB SNR. Only one run on each sound was made and the variance of the noise was made time-varying and the particle filtering changed accordingly. The listening tests revealed a significant reduction of the noise with only minor noticeable artifacts introduced to the speech.

VI. CONCLUSION

We have proposed a new parameterization of a time-varying auto-regressive speech model and used particle filtering for inference in a noise reduction set-up. The performance of the proposed speech model was compared to that of a speech model parameterized by auto-regressive coefficients for the application of speech enhancement [1]. The results from a number of experiments showed that the proposed model provided higher SNR estimates of the speech over a large interval of the random walk parameters of the particle filter and more accurate and smooth estimates of the model parameters were obtained as well which favors the proposed model as a better model for speech.

ACKNOWLEDGMENT

The authors would like to thank Jaco Vermaak for making the code used in [1] available to us.

REFERENCES

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

[2] A. Doucet, S. Godsill, and M. West, "Monte carlo filtering and smoothing with application to time-varying spectral estimation," in *In Proc. ICASSP*, Istanbul, Turkey, June 2000, pp. 701–704.

[3] K. Hermansen and U. Hartmann, "Fbg model based low rate coding of speech," in *Proceedings of NORSIG 2002 : The 5th IEEE Nordic Signal Processing Symposium*, Norway, Oct. 2002.

[4] J. D. Markel and A. H. Gray, *Linear Prediction of Speech.* New York, USA: Springer-Verlag, New York, 1976.

[5] D. H. Klatt, "Software for a cascade/parallel formant synthesizer," *J. Acoust. Soc. Am.*, vol. 67, pp. 971–995, 1980.

[6] P. Boersma and D. Weening. (2006) Praat: doing phonetics by computer (version 4.4.24). Compter program. [Online]. Available: http://www.praat.org/

Appendix G

Non-Stationary Noise Reduction by Particle Filtering in a Cascade Formant Synthesizer Model

---

Submitted to IEEE Transactions on Speech and Language Processing June 2007.

IEEE TRANSACTIONS ON AUDIO, SPEECH AND LANGUAGE PROCESSING, VOL. X, NO. XX, NOVEMBER 2007 1

Non-Stationary Noise Reduction by Particle Filtering in a Cascade Formant Synthesizer Model

Thomas Beierholm (*), *Student Member, IEEE,* Ole Winther and Bert de Vries

***Abstract*— This paper examines the usefulness of particle filter inference in three different speech models for the application of non-stationary noise reduction. The most promising candidate speech model consists of a cascade-coupling of second-order IIR resonance filters parameterized by formant frequencies and formant bandwidths. Next, performance of a commercial-grade hearing aid noise reduction algorithm is compared to the particle filter algorithm which uses the chosen candidate speech model. Performance is compared for both stationary and non-stationary white noise and the particle filter algorithm compares favorably to the hearing aid noise reduction algorithm in low-SNR situations.**

***Index Terms*— Bayesian signal processing, particle filtering, time-varying auto-regressive speech model, formant frequencies, noise reduction, MMSE estimation, formant synthesis.**

EDICS Category: SPE-ENHA

I. Introduction

THE field of single-channel noise reduction (also referred to as speech enhancement) has received much attention and, still today, remains an active field of research. One application of single-channel Noise Reduction (NR) is for Hearing Instruments (HI) where potentially such an algorithm can make a tremendous difference for the user. Typically, NR in a HI is performed by time-varying linear filtering either in the frequency domain or directly in the time domain based on the concept of spectral subtraction. No matter what domain is used for processing the noisy signal, the most common approach is to start the analysis of the noisy signal in the frequency domain. This is most natural as a frequency domain representation is already available for performing the actual amplification in an HI. The focus of the frequency analysis is centered around obtaining a noise power estimate which is then used for computing gains using some objective function derived from e.g. Wiener filter theory or statistical estimation theory. A basic limitation of this approach is to obtain an accurate noise power estimate. During speech activity it is difficult to obtain an accurate estimate of the noise, especially if the noise is non-stationary. Therefore it is often assumed that the noise is stationary or its spectrum changes slowly during speech intervals. The dilemma is typically that if the noise is tracked slowly, then inaccurate noise estimates result when the noise is non-stationary, whereas if fast tracking of the noise is performed there is a high risk of canceling speech.

Manuscript received June xx, 2007; revised Month xx, 2007.

T. Beierholm is with GN ReSound A/S, Lautrupbjerg 7, P.O. Box 99, 2750 Ballerup, Denmark, email tbe@imm.dtu.dk.

O. Winther is with IMM, Danish Technical University, 2800 Lyngby, Denmark.

B. de Vries is with GN ReSound A/S, Algorithm R&D, Eindhoven, The Netherlands (email: bdevries@gnresound.com).

In this work a rather different single-channel NR is proposed. The approach is based on a generative model for the noisy speech and Bayesian inference of the clean speech and the parameters in the generative model. An important feature of the used generative model is that speech features and the noise variance are allowed to change from sample to sample and hence fluctuating or non-stationary noise is built into the model. The degree of non-stationarity of the noise is controlled by a single scalar hyperparameter that determines how fast the variance of the noise can change.

Three different candidate speech models are examined. One speech model is parameterized by AR coefficients and is called a Time-Varying Auto-Regressive (TVAR) model [1]. The other two speech models are parameterized by formant frequencies, formant bandwidths and formant gains. These models are called Cascade Formant Synthesizer (CFS) and Parallel Formant Synthesizer (PFS) models in speech synthesis [2]. All three speech models facilitate a state-space formulation of the observed noisy signal in which the problem of joint estimation of the unknown parameters of the model and the state sequence becomes a challenge. One approach is to perform ML estimation using the EM algorithm. A different approach was used in [3], [4], where sequential Bayesian estimation of the unknown parameters and state sequence was performed by particle filtering. Particle filtering is suitable for online processing and will be used for inference in this work.

Representing speech using AR coefficients was used in [3]. However, we demonstrate that by representing speech using formant features instead will appear to be very beneficial if known properties of the formant feature parameters are exploited. By parameterizing the AR coefficients in terms of formant features instead of parameterizing directly in terms of the AR coefficients, more realistic dynamics of the AR coefficients are ensured.

A common feature of the generative models introduced in our work is that, conditional on the unknown parameters of the models, the state space model becomes Gaussian. In [3], this feature is made use of in a variance reduction step (Rao-Blackwellization) whereby the problem of sampling from the joint posterior distribution of the states and the unknown parameters of the model is reduced to that of sampling from the posterior distribution of the unknown parameters only.

Based on initial tests the CFS model is chosen for further evaluation and comparison with a HI NR algorithm which is based on the principle of spectral subtraction. The objective of the HI NR algorithm is to suppress noise as much as possible without introducing any audible speech distortion. To perform a meaningful comparison between the particle filtering algorithm and the HI NR algorithm, a modified Minimum Mean

Square-Error (MMSE) cost function is introduced. This leads to an estimator that can trade-off between the amount of noise reduced and speech distortion by adjusting a single scalar parameter. To evaluate the performance of the algorithms two objective performance measures are used, one to measure the amount of noise reduced and one to measure the amount of speech distortion. The algorithms are compared using both stationary and non-stationary white noise.

The paper is organized as follows. Sections II and III can be considered the theory part of the paper. In section II, three speech models are described and mathematically formulated and probabilistic models for the parameters in the models are specified. In section III, the basics of particle filtering are introduced and pseudo-code for the algorithms is presented. Section IV compares performance of the three models by examining accuracy of parameter and speech estimates on synthetic and real speech data. The purpose of section IV is to choose one of the three candidate models for further evaluation. The performance of the chosen model is compared to the performance of a HI NR algorithm. Sections V-VII concern the performance comparison. Section V introduces the modified MMSE cost function and section VI introduces the objective performance measures used. Section VII presents the results of the performance comparison and finally section VIII has a conclusion.

II. Speech Models

In section II-A the TVAR model is presented. The CFS and PFS models are described in sections II-B and II-C, respectively. Details on the state-space formulations of the models can be found in appendix I.

A. Time-Varying Auto-Regressive Model

In the Time-Varying Auto-Regressive (TVAR) model a speech signal is modeled as a non-stationary AR(p) process, where p denotes the order of the AR process which is assumed fixed in the following. The coefficients of the AR(p) process and the variance of the process noise are allowed to change from sample to sample, i.e. if x_n denotes a random variable associated with a TVAR process at time n, then

$$x_n = \sum_{i=1}^{p} a_{n,i}\; x_{n-i} + \sigma_{e_n}\, e_n \quad , \quad e_n \overset{iid}{\sim} \mathcal{N}(0,1) \qquad (1)$$

where $a_{n,i}$ denotes the i^{th} AR coefficient at time n, $\sigma^2_{e_n}$ is the variance of the innovation sequence and $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes a Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. As will be clear later, the parameters in (1) may not be the most convenient to use for the purpose of particle filtering. Instead, parameters will be used that can be mapped to the parameters in (1). It is assumed that the TVAR process is contaminated by non-stationary white Gaussian noise

$$y_n = x_n + \sigma_{d_n}\, d_n \quad , \quad d_n \overset{iid}{\sim} \mathcal{N}(0,1),$$

where $\sigma^2_{d_n}$ denotes the variance of the observation noise. The unknown parameter vector at time n is $\boldsymbol{\theta}_n^{TVAR}$=[$\mathbf{a}_n$, ϕ_{e_n}, ϕ_{d_n}]^T, where ϕ_{e_n}=$\log \sigma^2_{e_n}$, ϕ_{d_n}=$\log \sigma^2_{d_n}$ and $\mathbf{a}_n$=[$a_{n,1}$, $\cdots$, $a_{n,p}$]^T is the coefficient vector.

B. Cascade Formant Synthesizer Model

The Cascade Formant Synthesizer (CFS) model is a TVAR model reparameterized by formant frequency and formant bandwidth parameters. The CFS model synthesizes speech by exciting K cascade-coupled resonance filters with white noise as shown in figure 1. The resonators are taken as second-order IIR filters with z-transforms

$$H_{n,k}(z) = \frac{1}{1 - a_{n,2k-1}\, z^{-1} - a_{n,2k}\, z^{-2}} \qquad (2)$$

for k=1, . . . , K. The coefficients in (2) can be computed from formant frequencies and formant bandwidths by the following mapping [2], [5]

$$a_{n,2k-1} = 2e^{-\pi b_{n,k}/f_s} \cos(2\pi f_{n,k}/f_s) \qquad (3)$$

$$a_{n,2k} = -e^{-2\pi b_{n,k}/f_s}\;, \qquad (4)$$

where $f_{n,k}$ and $b_{n,k}$ denote the formant frequency and formant bandwidth, respectively, of the k^{th} formant at time n and f_s is the sampling frequency in Hz. By constraining the formant frequencies and bandwidths the CFS model can be viewed as a constrained TVAR model. As opposed to the TVAR model in section II-A, the poles of the CFS model are constrained to occur in complex conjugate pairs confined to certain areas within the unit circle. If $\mathbf{f}_n$=[$f_{n,1}$, . . . , $f_{n,K}$]^T and $\mathbf{b}_n$=[$b_{n,1}$, . . . , $b_{n,K}$]^T then the unknown parameter vector in the CFS model is $\boldsymbol{\theta}_n^{CFS}$ = [$\mathbf{f}_n$, $\mathbf{b}_n$, ϕ_{e_n}, ϕ_{d_n}]^T.

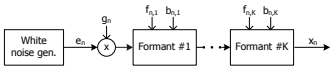

Fig. 1. Block diagram of Cascade Formant Synthesizer.

C. Parallel Formant Synthesizer Model

A PFS model synthesizes speech by summing the outputs of a number of resonance circuits connected in parallel. The structure of a PFS is illustrated in Fig. 2. The resonators are driven by a common excitation signal which is taken to be white noise. The PFS model is not a TVAR model because the transfer function for the PFS model contains $p-2$ zeros. A description of the advantages and disadvantages of the cascade and parallel structures in speech synthesis can be found in [2]. Even though the PFS model is not a TVAR model, particle filtering inference for the PFS and CFS models are quite similar because the models share most of their parameters and because the PFS model can also be formulated as a state-space model obeying (23) and (24). The resonators for the PFS model are taken as second-order IIR filters with z-transforms

$$H_{n,k}(z) = \frac{g_{n,k}}{1 - a_{n,2k-1}\, z^{-1} - a_{n,2k}\, z^{-2}}$$

where each resonance filter has a separate time-varying gain $g_{n,k}$. If $\mathbf{g}_n$=[$\phi_{g_{n,1}}$, . . . , $\phi_{g_{n,K}}$]^T then the unknown parameter vector in the PFS model is $\boldsymbol{\theta}_n^{PFS}$ = [$\mathbf{f}_n$, $\mathbf{b}_n$, $\mathbf{g}_n$, ϕ_{d_n}]^T where $\phi_{g_{n,k}} = \log g^2_{n,k}$.

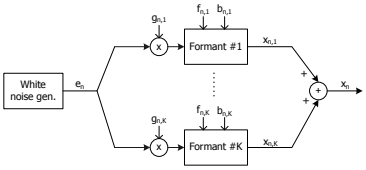

Fig. 2. Block diagram of Parallel Formant Synthesizer.

D. Probabilistic Models for Parameters

As will be evident in section III, particle filtering depends on statistical models for the parameters contained in the parameter vector $\boldsymbol{\theta}$. It is important that samples can be drawn from the parameter models, specifically because, as section III will demonstrate particle filtering relies on samples being drawn from the initial state distribution $p(\boldsymbol{\theta}_0)$ and the state-transition distribution $p(\boldsymbol{\theta}_n \,|\, \boldsymbol{\theta}_{n-1})$. A specification of the parameter models can be found in appendix II.

III. Particle Filter Inference

Generally, filtering refers to the task of computing the *filtering distribution* $p(\mathbf{x}_n, \boldsymbol{\theta}_n \,|\, \mathbf{y}_{1:n})$ and the MMSE estimate of the state vector and the parameter vector at time n using observations $\mathbf{y}_{1:n} = [y_1, \cdots, y_n]^T$ up to and including time n. In the ideal case it is possible to compute a closed-form analytic expression for the MMSE estimator of the parameter vector and the state vector. For non-ideal cases this is not possible, making approximations necessary. Given N independent samples from the filtering distribution at time n, an approximation to the MMSE estimator can be computed as

$$[\hat{\mathbf{x}}_n \,,\, \hat{\boldsymbol{\theta}}_n] = \frac{1}{N} \sum_{i=1}^{N} [\mathbf{x}_n^i \,,\, \boldsymbol{\theta}_n^i] \tag{5}$$

where the superscript denotes the sample index. It is generally not possible to draw samples directly from the filtering distribution but samples can be drawn using *importance sampling*. In this case samples are drawn from an importance distribution, $\pi(\mathbf{x}_n, \boldsymbol{\theta}_n \,|\, \mathbf{y}_{1:n})$, (also called a proposal distribution) and importance weights are formed

$$w(\mathbf{x}_n \,,\, \boldsymbol{\theta}_n) = \frac{p(\mathbf{x}_n, \boldsymbol{\theta}_n \,|\, \mathbf{y}_{1:n})}{\pi(\mathbf{x}_n, \boldsymbol{\theta}_n \,|\, \mathbf{y}_{1:n})} \tag{6}$$

Using the importance distribution the MMSE estimator can be written

$$< \mathbf{x}_n \,,\, \boldsymbol{\theta}_n >= E_{\pi(\mathbf{x}_n, \boldsymbol{\theta}_n \,|\, \mathbf{y}_{1:n})} \{[\mathbf{x}_n \,,\, \boldsymbol{\theta}_n] \; w(\mathbf{x}_n \,,\, \boldsymbol{\theta}_n)\} \tag{7}$$

However, the expression in (7) is just as intractable as the original integral defining the MMSE estimator. If N independent samples are drawn from the importance distribution and the weights given by (6) are evaluated for each drawn sample then the expression for the MMSE estimator in (7) can be approximated as

$$[\hat{\mathbf{x}}_n \,,\, \hat{\boldsymbol{\theta}}_n] = \sum_{i=1}^{N} [\mathbf{x}_n^i \,,\, \boldsymbol{\theta}_n^i] \; \tilde{w}(\mathbf{x}_n^i \,,\, \boldsymbol{\theta}_n^i) \tag{8}$$

where

$$\tilde{w}(\mathbf{x}_n^i \,,\, \boldsymbol{\theta}_n^i) = \frac{w(\mathbf{x}_n^i \,,\, \boldsymbol{\theta}_n^i)}{\sum_{i=1}^{N} w(\mathbf{x}_n^i \,,\, \boldsymbol{\theta}_n^i)}$$

are the normalized importance weights. The above reasoning can be generalized to consider full trajectories $p(\mathbf{x}_{0:n}, \boldsymbol{\theta}_{0:n} \,|\, \mathbf{y}_{1:n})$.

The posterior distribution over the full trajectories can be decomposed into

$$p(\mathbf{x}_{0:n}, \boldsymbol{\theta}_{0:n} \,|\, \mathbf{y}_{1:n}) = p(\mathbf{x}_{0:n} \,|\, \boldsymbol{\theta}_{0:n}, \mathbf{y}_{1:n}) \, p(\boldsymbol{\theta}_{0:n} \,|\, \mathbf{y}_{1:n}) \tag{9}$$

Because the state variables given the parameters are Gaussian, the state variables can be marginalized out with Kalman (Rao-Blackwellization) leaving us with a filtering model for the parameters. So, $p(\mathbf{x}_{0:n} \,|\, \boldsymbol{\theta}_{0:n}, \mathbf{y}_{1:n})$ can be marginalized and the marginalized posterior $p(\mathbf{x}_n \,|\, \boldsymbol{\theta}_{0:n}, \mathbf{y}_{1:n})$ admits a Gaussian distribution whose mean and covariance can be computed by Kalman filtering. Introducing a new importance distribution, $\pi(\boldsymbol{\theta}_{0:n}|\mathbf{y}_{1:n})$, the MMSE estimator approximation can be computed as

$$[\hat{\mathbf{x}}_n, \hat{\boldsymbol{\theta}}_n] = \sum_{i=1}^{N} E_{p(\mathbf{x}_n \,|\, \boldsymbol{\theta}_{0:n}, \mathbf{y}_{1:n})} \left\{[\mathbf{x}_n, \, \boldsymbol{\theta}_{0:n}^i]\right\} \tilde{w}(\boldsymbol{\theta}_{0:n}^i) \tag{10}$$

where

$$w(\boldsymbol{\theta}_{0:n}) = \frac{p(\boldsymbol{\theta}_{0:n} \,|\, \mathbf{y}_{1:n})}{\pi(\boldsymbol{\theta}_{0:n} \,|\, \mathbf{y}_{1:n})} \;,\; \tilde{w}(\boldsymbol{\theta}_{0:n}^i) = \frac{w(\boldsymbol{\theta}_{0:n}^i)}{\sum_{i=1}^{N} w(\boldsymbol{\theta}_{0:n}^i)} \tag{11}$$

A very undesirable feature of the importance sampling method in the general case is the need to store the trajectories (the history) of the samples drawn from the importance distribution and the observations. This means that storage requirements and the computational load will grow over time. This makes the importance sampling method impractical for real-time applications. Fortunately, there is a way to formulate a recursive version of the importance sampling method which alleviates the undesirable features. This method is known as *Sequential Importance Sampling* (SIS). The important step to develop a recursive version of the approximation in (10) is to obtain a recursion formula for the weights in (11) which can then be used in (10). This can be done if the importance distribution is restricted to the general form

$$\pi(\boldsymbol{\theta}_{0:n} \,|\, \mathbf{y}_{1:n}) = \pi(\boldsymbol{\theta}_{0:n-1} \,|\, \mathbf{y}_{1:n-1}) \cdot \pi(\boldsymbol{\theta}_n \,|\, \boldsymbol{\theta}_{0:n-1}, \mathbf{y}_{1:n}) \tag{12}$$

The crucial restriction is that the time dependence only goes to $n-1$ in the first term. It can then be shown (see appendix III) that the weights obey the recursion

$$w(\boldsymbol{\theta}_{0:n}) \propto w(\boldsymbol{\theta}_{0:n-1}) \cdot w(\boldsymbol{\theta}_n) \tag{13}$$

with

$$w(\boldsymbol{\theta}_n) \propto \frac{p(y_n|\, \mathbf{y}_{1:n-1}, \boldsymbol{\theta}_{0:n}) \; p(\boldsymbol{\theta}_n|\boldsymbol{\theta}_{n-1})}{\pi(\boldsymbol{\theta}_n|\, \boldsymbol{\theta}_{0:n-1}, \mathbf{y}_{1:n})} \;. \tag{14}$$

In this way sequential importance sampling avoids the need to store the paths $\boldsymbol{\theta}_{0:n-1}$ of the particles. The complexity of computing the weights can be simplified if the importance distribution at time n is set equal to the prior distribution, i.e. $\pi(\boldsymbol{\theta}_n \,|\, \boldsymbol{\theta}_{0:n-1}, y_{1:n}) = p(\boldsymbol{\theta}_n|\boldsymbol{\theta}_{n-1})$ so that (14) reduces to

$$w_n \propto p(y_n|\boldsymbol{\theta}_{0:n}, \mathbf{y}_{1:n-1}). \tag{15}$$

However, setting the importance distribution equal to the prior distribution contributes to a degeneracy whereby all weights except one after a few time steps are very close to zero. This happens because the importance distribution is different from the true posterior distribution. As a remedy, a resampling step is introduced. The resampling step duplicates particles in proportion to their importance weights in such a way that all particles have approximately the same weight after the resampling step. Incorporating the selection step, the weights $w(\boldsymbol{\theta}_{0:n-1})$ will all be equal to 1 and $w(\boldsymbol{\theta}_{0:n}) \propto w(\boldsymbol{\theta}_n)$.

Putting it all together, the procedure of the particle filter is illustrated with pseudo-code in Algorithm 1. The main references for this summary of particle filtering are [3], [6].

Algorithm 1 Particle filter pseudo-code

```
Draw samples from initial distributions (35), (37) or (39)
for t = 1 to T do
    for i = 1 to N do
        Draw sample from state-transition distributions (36), (38) or (40)
        Form system matrices (25)-(28) or (30)-(33)
        Perform a Kalman filter step according to state-space eqns (23)-(24) (Rao-Blackwellization)
        Compute weight using (15)
    end for
    Resampling
    Compute estimates using (5)
end for
```

IV. Experiments & Discussion of Speech Models

In section IV-A the data used to examine and compare performance of the three models is described. Also in section IV-A, the setting of the hyper-parameters for the models is specified and the measured dB SNR improvements are presented. In sections IV-B to IV-D separate discussions of the workings of each of the models are presented. Finally, in section IV-E a summary of the experiments is given and one of the three models is chosen for further evaluation.

A. Data Generation & Measured SNR Improvements

To investigate the behavior of the three models, both synthetic and real speech data are used. The synthetic data is generated from the TVAR model but the parameter values are extracted from real speech. Formant frequencies, formant bandwidths and formant gains are extracted from the TIMIT sentence, $sa2$, containing a female speaker saying "Don't ask me to carry an oily rag like that." down-sampled to 16 kHz. Formant frequencies are extracted by finding the location of peaks in a 14^{th} order model-based spectrum, the formant bandwidths are extracted by computing the 3 dB down points from the location of the peaks and the formant gains are taken as the values of the peaks in the spectrum. The parameters are extracted from a 25ms window and the window is shifted in steps of 1.25ms duration. After the raw extraction of the parameters, the formant frequency tracks are manually prepared by first sorting out the first four formants and then cubic interpolation is performed on the parameter tracks such that the rate of the parameter values is equal to the sampling rate. The upper plot in figure 3 shows a wide-band

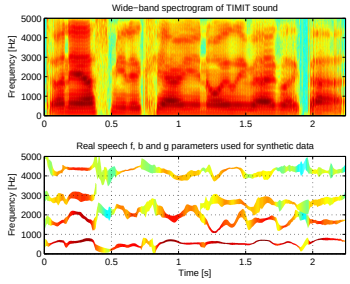

Fig. 3. Upper plot shows a wide band spectrogram of the TIMIT sentence, $sa2$, and the lower plot illustrates the extracted fbg parameters. The thickness of the tracks in the lower plot indicates the estimated formant bandwidths and the strength of the formants are indicated so that dark intervals represent strong formants.

spectrogram of the original TIMIT speech sentence and the lower plot illustrates the formant-bandwidth-gain parameters extracted from the TIMIT sentence. The synthetic data used for experiments in this section is generated using the parameter values illustrated in the lower plot by first mapping formant frequencies and bandwidths to AR coefficients using (3) and (4). The noise variance was made known and constant at unity, that is $D_n(\boldsymbol{\theta}) = 1$ for all n. The standard deviation of the innovation noise was taken to be proportional to the gain of the first formant $\sigma_{e_n} \propto g_{n,1}$. The standard deviation of the innovation noise was scaled such that the SNR of the generated noisy data was approximately 5 dB. Finally, data was generated using (23) and (24). The real speech data consists of the TIMIT sentence with white Gaussian noise added at an SNR of 5 dB.

The number of particles used in all experiments is 100. Using more particles provides only minor changes in SNR improvement. All experiments are repeated seven times and the measured SNR improvements and the parameter estimates represent an average over the seven repeated experiments. All hyper parameters were manually tuned to provide highest SNR improvements. This resulted in the settings shown in table I (see appendix II for a specification of the hyper parameters). The measured dB SNR improvements are seen in table II.

B. TVAR

To better understand how the TVAR particle filter works, model-based spectrograms were made using the true and estimated AR coefficients and gains. The plot in figure 4a shows the model-based spectrogram using the true AR coefficients and gains and the plot in figure 4b shows the corresponding

TABLE I
MANUALLY TUNED SETTINGS OF HYPER PARAMETERS.

Parameter	Setting
$\delta^2_{\mathbf{a}}$	$1{\cdot}10^{-4}$
$\delta^2_{\mathbf{c}}$	$5{\cdot}10^{-3}$
$\delta^2_{\mathbf{a}_0}$	0.5
$\delta^2_{\mathbf{c}_0}$	0.5
$\delta^2_{\mathbf{f}}$	20
$\delta^2_{\mathbf{b}}$	7

TABLE II
MEASURED DB SNR IMPROVEMENTS FOR THE TVAR, CFS AND PFS MODELS FOR SYNTHETIC AND REAL SPEECH DATA AT AN SNR OF 5 DB.

	TVAR	CFS	PFS
Synthetic	4.84	5.17	4.77
Real Speech	4.19	4.86	4.56

model-based spectrogram computed from AR coefficients and gains averaged over the seven runs. Figure 5 shows four plots comparing directly the estimated and true spectrums at four time points. It is seen that the TVAR particle filter models the strongest formant or the two strongest formants but that it also uses some of its modeling capability/capacity at higher frequencies where there are no formants. The TVAR particle filter is not constrained to form realistic speech spectra and as such nothing prevents the particle filter from generating particles with spectra having formants above 5000Hz. Not only is the TVAR particle filter inefficient because of inefficient particles but the TVAR particle filter is also computationally inefficient because every time a particle is generated the roots of the denominator polynomial in the system function must be computed, as is also noted in appendix II-A.

C. CFS

The estimated model-based spectrogram created from the averaged estimated parameters can be seen in figure 4c. Comparing with the estimated model-based spectrograms in figure 4b and figure 4c it is evident that the CFS particle filter provides parameter estimates that are much more accurate and smooth than the TVAR particle filter estimates. Figure 5 shows true spectra and estimated spectra at four different time points and can be compared with those obtained using the TVAR model. Basically, the CFS particle filter improves on all the deficiencies of the TVAR model.

Two subtleties of the CFS particle filter are demonstrated in figure 6 which shows the true gain and the estimated gain. It is seen that the estimated gain curve exhibits relative large fluctuations around the true gain curve. It is also seen that in the interval from approximately 0.5s to 0.8s where the speech signal is relative weak, the estimated gain is virtually zero. Lowering the value of $\delta^2_{\mathbf{c}}$ a more smooth estimated gain curve is obtained but the measured SNR improvement is decreased slightly. Figure 7 shows true formant tracks and formant tracks estimated by the CFS particle filter. It is seen that the particle filter provides a relatively accurate frequency estimation of the strong first formant whereas less accurate estimates of the frequencies of the weaker second and third formants are obtained. By comparing figures 6 and 7 it is seen that when the estimated gain is zero or very close to zero the particle

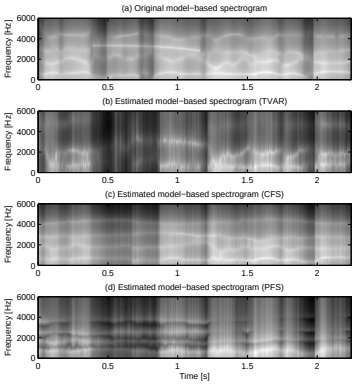

Fig. 4. (a) Original model-based spectrogram of the synthetic data. (b), (c) and (d) model-based spectrograms computed from the estimated parameters by the TVAR, CFS and PFS particle filters, respectively.

filter effectively freezes adaptation of the formant frequencies which is particularly clear for the interval containing the weak speech sound from approximately $0.5 - 0.8$ s. Even though the SNR of the noisy signal is approximately 5 dB, locally in the interval $0.5 - 0.75$ s the SNR is approximately -11 dB which is why the estimated gain is zero for this interval. By increasing the SNR the speech gain is estimated accurately during this interval as well.

D. PFS

The idea behind the PFS model is the same as that of the CFS model, to constrain the spectrum of the model to being speech-like, and this is ensured by constraining the poles to occur in complex conjugate pairs and by constraining the paths of the formant frequencies to certain regions and not allowing them to cross. Another feature of the PFS model is that fading or emerging formants are conveniently handled by appropriately setting the formant gain parameters; at least in principle. However, the PFS model is essentially a bank of 2^{nd} order IIR filters with non-linear phase. Due to the introduced phase distortion the gains of the IIR filters do not relate directly to the peaks in the composite frequency response in the same way as if the filters had linear phase. This issue is illustrated in figure 8 where the estimated gains for the real speech data are shown. It is seen that only the gain of the first formant follows the energy in the signal. This basically means that the strength of the formants cannot be controlled using the gains only.

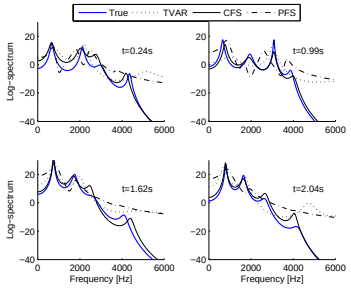

Fig. 5. The figure shows the true and estimated spectra at 4 time points of the synthetic data.

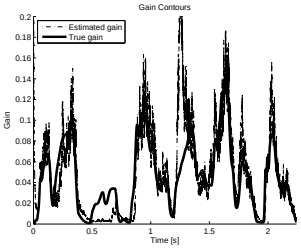

Fig. 6. Plot shows the true gain of the synthetic data and gain estimated by the CFS particle filter. The estimated gain is close to zero in the interval 0.5s to 0.8s because locally the SNR is -11 dB in this interval.

E. Summary

Some deficiencies of the TVAR model have been described. It has also been described how the CFS model is a reparameterized and constrained TVAR model that improves upon the deficiencies of the TVAR model. The CFS model performs significantly better than the TVAR model both in terms of measured SNR improvement on synthetic and real speech data and on accuracy of the parameter estimates. The PFS model was also proposed as an alternative speech model but phase distortion and performance issues makes the PFS model less attractive for particle filtering than the CFS model. Hence the CFS particle filter is chosen for further evaluation and comparison with a HI NR algorithm in section VII.

V. Perceptual Estimator

A very important criterion for a single-channel NR algorithm to be appropriate for use in an HI is to introduce very little noticeable distortion to the speech signal. The HI NR algorithm used for comparison does not have maximal SNR

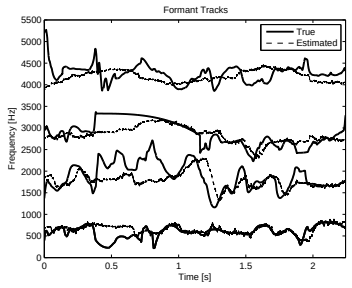

Fig. 7. Plot shows true formant frequencies of the synthetic data and the formant frequencies estimated by the CFS particle filter. In the original speech data the third formant faded out in the interval from approximately 0.4s to 1.2s and an artificial frequency path was inserted in this interval.

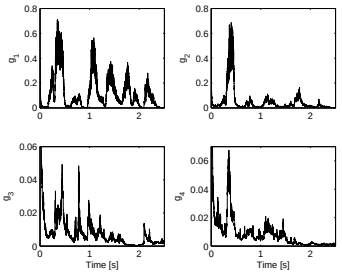

Fig. 8. The figure shows the estimated gain values by the PFS particle filter processing the real speech data. Due to the introduced phase distortion in the PFS model there is no synchrony (common onset/offset) between the gains of the IIR filters and the gains do not relate directly to the peaks in the composite frequency response.

improvement as its objective function. In order to take this knowledge into account in the particle filtering approach, the new simple modified MMSE cost function that was proposed in [7] is used. The modified cost function introduces a parameter that trades-off SNR improvement for reduced signal distortion. The modified cost function is

$$C(\mathbf{x}, \hat{\mathbf{x}}) = ((\mathbf{x} + \alpha\mathbf{n}) - \hat{\mathbf{x}})^2 \quad , \quad 0 \leq \alpha \leq 1 \qquad (16)$$

where $\mathbf{n}$ denotes the noise and $\hat{\mathbf{x}}$ denotes an estimate of the signal. The α parameter is a convenient parameter that can trade-off between SNR improvement and signal distortion. Minimizing the risk yields the modified MMSE estimator

$$\hat{\mathbf{x}} = (1 - \alpha) E\{\mathbf{x}|\mathbf{y}\} + \alpha\mathbf{y} \qquad (17)$$

where the following observation model $\mathbf{y} = \mathbf{x} + \mathbf{n}$ has been assumed. The modified estimator is thus expressed as a linear interpolation between the MMSE estimator and the observed noisy speech signal and therefore can be used directly in the particle filtering framework.

VI. Objective Performance Measures

There is not a clear winner when it comes to objectively evaluating speech enhancement algorithms but several studies, see [8]–[11], indicate that separate measures for residual noise suppression and speech distortion provide a reasonable indication of performance. We chose to evaluate noise suppression by the segmental-SNR metric and speech distortion by the frequency-weighted log-spectral distance, see [8]–[11].

The HI NR algorithm processes the noisy speech block-wise by convolving the noisy speech with a slowly time-varying linear filter whereas the CFS particle filter processes the noisy speech sample-wise by evaluation of a statistical estimator function. The group delay introduced by the HI NR algorithm must be taken into account when computing the performance measures. Otherwise a simple delay of the input could give substantial SNR improvement measurements even though no noise has been removed. The same issue does not exist with the CFS particle filter. For the CFS particle filter the segmental SNR improvement is computed from

$$segSNRI = \frac{1}{M}\sum_{m=1}^{M}\left[10\log_{10}\frac{\sum_t n_{in,m}^2(t)}{\sum_t n_{out,m}^2(t)}\right]_{-20}^{+35} \quad (18)$$

where m is the block index and the SNR is restricted to values between -25 and $+35$ dB to avoid bias. For the CFS particle filter, the output noise is given by $\mathbf{n}_{out} = \mathbf{y} - \widehat{\mathbf{x}}$. For the HI NR algorithm, the segmental SNR improvement is computed from

$$segSNRI = segSNR_{out} - segSNR_{in} \quad (19)$$

where

$$segSNR_{out} = \frac{1}{M}\sum_{m=1}^{M}\left[10\log_{10}\frac{\sum_t x_{out,m}^2(t)}{\sum_t n_{out,m}^2(t)}\right]_{-20}^{+35} \quad (20)$$

$$segSNR_{in} = \frac{1}{M}\sum_{m=1}^{M}\left[10\log_{10}\frac{\sum_t x_{in,m}^2(t)}{\sum_t n_{in,m}^2(t)}\right]_{-20}^{+35} \quad (21)$$

In order to compute the segmental SNR at the output, the HI NR algorithm is modified such that the clean speech and the noise signals are separately filtered in the same way as the noisy speech is filtered. The clean speech and noise filtered in this way are used in (20).

The frequency-weighted log-spectral signal distortion measure indicates an Euclidian distance between short-time spectra (in dB) of the clean speech component and the processed speech component and is computed from

$$SD = \frac{1}{M}\sum_{m=1}^{M}\sqrt{\frac{1}{K}\sum_{k=1}^{K}\left(w_{aud}(k)10\log_{10}\frac{\widehat{X}(k,m)}{X(k,m)}\right)^2} \quad (22)$$

where $\widehat{X}(k,m)$ denotes the estimated speech-frequency power-density spectrum in block m, k denotes frequency bin index and $w_{aud}(k)$ denotes a frequency weighting factor giving approximately equal weight to each auditory critical band.

VII. Performance

The performance of the HI NR and the CFS algorithms are tested using stationary and non-stationary white noise created by performing a Gaussian Random Walk (RW) on the variance of the noise using an RW variance of $1e{-}6$ truncated to positive values. Performance is measured for four different SNRs, 0, 5, 10 and 20 dB, and six different speakers taken from the TIMIT database, three male (SA1, SX333, SX451) and three female (SX206, SX312, SX402) speakers.

We compared two approaches to the state transition for the gain: Gaussian distributed in the gain and the log gain, respectively. The latter has the advantage that the gain is guaranteed to stay non-negative. However, we observed stability problems with it because it takes unrealistically large steps for large gain and excruciatingly small steps for small gain. We therefore applied a Gaussian RW truncated to positive values to the gain directly, guaranteeing a fixed step size, independent of the value of the gain. Hence, instead of adapting the log-variances it was chosen to adapt the standard deviations of the noise and speech. The CFS parameter vector used is thus $\boldsymbol{\theta}_n^{CFS} = [\ \mathbf{f}_n,\ \mathbf{b}_n,\ \sigma_{e_n},\ \sigma_{d_n}\ ]^T$ and $\delta_{d'}^2{=}5e{-}7$, $\delta_{e'}^2{=}1e{-}6$, where $\delta_{d'}$ and $\delta_{e'}$ denotes the RW variances for σ_{d_n} and σ_{e_n}, respectively.

We compare the performance of our algorithm to a commercial-grade algorithm from the hearing aids industry. GN ReSound supplied us with a version of their spectral subtraction based noise reduction algorithm. In this algorithm, the noise spectrum is tracked through time-varying low-pass filtering of the signal+noise spectrum. Exact details of the algorithm are not disclosed, but we find it important to compare our method to commercial standards rather than choosing one of the many proposed methods in the literature. We are particularly interested in evaluating the performance of our system for estimating the signal under low-SNR conditions, because this is the situation that is notoriously difficult for single-microphone systems, including our reference algorithm that was supplied by GN ReSound.

Figures 9 and 10 show objective performance measures obtained for stationary noise and non-stationary noise, respectively. The figures show how performance vary depending of the value of the α parameter in (17) for four different SNR levels. Each curve represents an average performance computed from the average performance for each of the six speakers. As expected the $segSNRI$ measure decreases with increasing α, however, at an SNR level of 20 dB the $segSNRI$ measure increases with α because the raw MMSE estimate gives a negative $segSNRI$ score. It is also seen that the SD measure increases with increasing α except when α is small. At an SNR level of 20 dB the SD curve is relative flat though. The corresponding objective measures for the HI NR are also indicated. The grey horizontal lines in the figure show the

corresponding segSNRI and SD values for the HI NR. Based on figures 9 and 10 and informal listening tests a suitable setting of the α parameter was found to be 0.25. The objective performance measurements indicate that the CFS algorithm performs better compared to the HI NR algorithm for low SNR levels. When the SNR level is high enough the measurements indicate that the HI NR algorithm performs better than the CFS algorithm. Informal listening tests indicate that there is a slight difference in quality of the enhanced speech for the two algorithms where the enhanced speech produced by the HI NR algorithm is preferable. It was found that the distortion introduced by the CFS algorithm is more noticeable than for the HI NR algorithm. This observation can explain why the CFS algorithm is not performing much better than the HI NR algorithm for non-stationary noise relative to the stationary noise case.

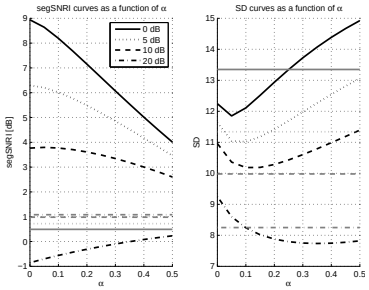

Fig. 9. Plots illustrating the variation in segSNRI and SD measurements as a function of α at four different SNR levels for stationary white noise. The measurements have been made by averaging performance for the six speakers. The grey horizontal lines show the corresponding segSNRI and SD values for the HI NR. Error bars are not shown because they were found to be visible on the 20 dB level curves only.

By converting the sub-band noise power estimates in the HI NR algorithm to a noise variance estimate, it is possible to compare the noise variance estimates of the CFS and HI NR algorithms. Figure 11 shows three examples for the case of speaker SA1 and non-stationary noise. In figure 11(a) and (b) the non-stationary noise with RW variance is used at 0 dB SNR and 5 dB SNR, respectively and in (c) noise where the variance changes abruptly every 500 ms and otherwise kept constant is used. In figure 11a the CFS algorithm tracks the noise variance much better than the HI NR algorithm which seems to not only react slowly but also seems to underestimate the noise variance. In figure 11b the HI NR algorithm is better at tracking the noise, mostly because the increased SNR makes it easier for the HI NR to track the noise. Figure 11b makes visible some intervals where the noise variance estimated by the CFS algorithm deviates significantly from the true noise variance. This is caused by a short-coming of the CFS model. The deviations occur during intervals corresponding to

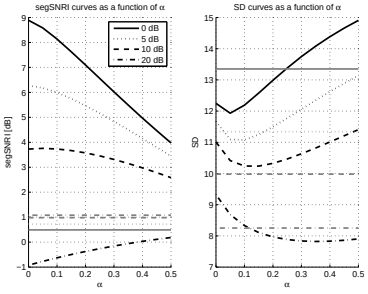

Fig. 10. Figure legend as in figure 9 but now with non-stationary white noise.

unvoiced speech sounds where most of the speech energy is concentrated at higher frequencies. Figure 11c demonstrates the difference between the two algorithms reacting to sudden changes in the noise variance. The CFS algorithms is tracking the noise variance much faster than the HI NR algorithm for this case.

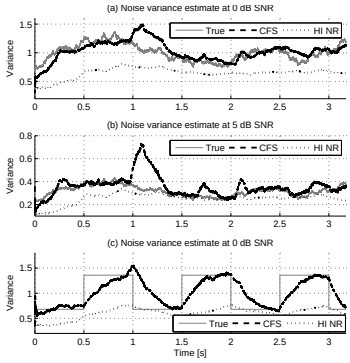

Fig. 11. (a) shows the true variance of the non-stationary noise with RW variance and the noise variance estimated by the CFS and HI NR algorithms at an SNR level of 0 dB (using speaker SA1). (b) show the noise variance estimates for the same setup but at an SNR level of 5 dB. (c) illustrates how the two algorithms react to sudden changes in the noise variance.

VIII. Conclusion

It has been found that reducing noise in speech by particle filter inference in a TVAR model is more beneficial if prior knowledge of the formants is exploited. This was achieved by parameterizing the AR coefficients by formant frequencies

and formant bandwidths instead of using the AR coefficients directly. It was also found that conditional on the resonance filter used to model the formants, the cascade formant synthesizer model was found more useful than the parallel formant synthesizer model. The performance of a hearing instrument noise reduction algorithm was compared to the particle filter algorithm for stationary and non-stationary white noise. It was found that noise tracking capabilities of the particle filter algorithm were superior to that of the hearing instrument noise reduction algorithm, especially for low SNR levels. We plan to further develop the particle filtering approach for high-SNR situations and consider boosting low-SNR performance for our hearing instrument set-up through integration of the particle filter algorithm with the hearing instrument noise reduction algorithm. We will develop the particle filtering approach by extending the speech model to take into account high-frequency speech components and to include a harmonic excitation term.

Appendix I
State-Space Formulation

A. TVAR

The TVAR model is conveniently formulated as a state-space model with the following state and observation equations

$$\mathbf{x}_n = \mathbf{A}_n\,\mathbf{x}_{n-1} + \mathbf{B}_n\,e_n \tag{23}$$

$$y_n = \mathbf{C}\,\mathbf{x}_n + D_n\,d_n, \tag{24}$$

where $\mathbf{x}_n = [\,x_n,\ \cdots,\ x_{n-p+1}\,]^T$ is the state vector and

$$\mathbf{A}_n = \begin{bmatrix} \mathbf{a}_n^T \\ \mathbf{I}_{(p-1)} \quad \mathbf{0}_{(p-1)\times 1} \end{bmatrix} \tag{25}$$

$$\mathbf{B}_n = \begin{bmatrix} \sigma_{e_n} \\ \mathbf{0}_{(p-1)\times 1} \end{bmatrix} \tag{26}$$

$$\mathbf{C} = [\ 1 \quad \mathbf{0}_{1\times(p-1)}\ ] \tag{27}$$

$$D_n = [\ \sigma_{d_n}\ ]\ , \tag{28}$$

where $\mathbf{a}_n = [\,a_{n,1},\ \cdots,\ a_{n,p}\,]^T$ is the coefficient vector. The state-space formulation of the TVAR model in eqs. (23) and (24), with the parameterization in (25), (26), (27) and (28), is used in [3].

B. PFS

By letting

$$\mathbf{A}_{n,k} = \begin{bmatrix} a_{n,2k-1} & a_{n,2k} \\ 1 & 0 \end{bmatrix} \tag{29}$$

then in state-space form, the PFS model is described by (23) and (24) with system matrices

$$\mathbf{A}_n = \mathrm{diag}\,(\mathbf{A}_{n,1}, \cdots, \mathbf{A}_{n,K}) \tag{30}$$

$$\mathbf{B}_n = \begin{bmatrix} \mathbf{0}_{1\times 2K} \\ g_{n,1}\ 0\ \cdots\ g_{n,K}\ 0 \end{bmatrix}^T \tag{31}$$

$$\mathbf{C} = [\,1\ 0\ 1\ 0\ \cdots\ 1\ 0\,] \tag{32}$$

$$D_n = [\ \sigma_{d_n}\ ]\ , \tag{33}$$

where the state vector changes form to

$$\mathbf{x}_n = [x_{n,1},\ x_{n-1,1},\ \cdots,\ x_{n,K},\ x_{n-1,K}]^T \tag{34}$$

and the process noise becomes bivariate $\mathbf{e}_n \sim \mathcal{N}(\mathbf{0}_{2\times 1}, \mathbf{I}_2)$. In (34) $x_{n,k}$ denotes the output from the k^{th} resonator (see figure 2) .

Appendix II
Probabilistic Models for Parameters

The initial state and state-transition distributions for the TVAR model are factorized as in [3], that is

$$p(\boldsymbol{\theta}_0^{TVAR}) = p(\mathbf{a}_0)p(\phi_{e_0})p(\phi_{d_0}) \tag{35}$$

$$\begin{aligned} p(\boldsymbol{\theta}_n^{TVAR}|\boldsymbol{\theta}_{n-1}^{TVAR}) = {} & p(\mathbf{a}_n|\mathbf{a}_{n-1})p(\phi_{e_n}|\phi_{e_{n-1}}) \\ & \times p(\phi_{d_n}|\phi_{d_{n-1}})\ ,\ n>1\ . \end{aligned} \tag{36}$$

The initial state and state-transition distributions for the parameters in the CFS model are factorized as

$$p(\boldsymbol{\theta}_0^{CFS}) = p(\mathbf{f}_0)p(\mathbf{b}_0)p(\phi_{e_0})p(\phi_{d_0}) \tag{37}$$

$$\begin{aligned} p(\boldsymbol{\theta}_n^{CFS}|\boldsymbol{\theta}_{n-1}^{CFS}) = {} & p(\mathbf{f}_n|\mathbf{f}_{n-1})p(\mathbf{b}_n|\mathbf{b}_{n-1})p(\phi_{e_n}|\phi_{e_{n-1}}) \\ & \times p(\phi_{d_n}|\phi_{d_{n-1}})\ ,\ n>1\ . \end{aligned} \tag{38}$$

The initial state and state-transition distributions for the parameters in the PFS model are factorized as

$$p(\boldsymbol{\theta}_0^{PFS}) = p(\mathbf{f}_0)p(\mathbf{b}_0)p(\mathbf{g}_0)p(\phi_{d_0}) \tag{39}$$

$$\begin{aligned} p(\boldsymbol{\theta}_n^{PFS}|\boldsymbol{\theta}_{n-1}^{PFS}) = {} & p(\mathbf{f}_n|\mathbf{f}_{n-1})p(\mathbf{b}_n|\mathbf{b}_{n-1})p(\mathbf{g}_n|\mathbf{g}_{n-1}) \\ & \times p(\phi_{d_n}|\phi_{d_{n-1}})\ ,\ n>1\ . \end{aligned} \tag{40}$$

A. TVAR

The initial state distributions are taken to be Gaussian and the parameters are assumed to evolve according to first-order Gaussian Markov processes. As such, samples from the initial state and state-transition distributions are generated using the following generative expressions

$$\begin{array}{lll} \mathbf{a}_0 = \mathbf{v} & , & \mathbf{v} \ \sim \mathcal{N}\left(\mathbf{0}_{p\times 1}, \delta_{\mathbf{a}_0}^2\mathbf{I}_p\right) \\ \mathbf{a}_n = \mathbf{a}_{n-1} + \mathbf{w}_n & , & \mathbf{w}_n \sim \mathcal{N}\left(\mathbf{0}_{p\times 1}, \delta_{\mathbf{a}}^2\mathbf{I}_p\right) \\ \phi_{e_0} = \alpha & , & \alpha \ \sim \mathcal{N}\left(-2, 0.25\right) \\ \phi_{e_n} = \phi_{e_{n-1}} + \beta_n\ , & & \beta_n \sim \mathcal{N}\left(0, \delta_e^2\right) \\ \phi_{d_0} = \gamma & , & \gamma \ \sim \mathcal{N}\left(0, \delta_{d_0}^2\right) \\ \phi_{d_n} = \phi_{d_{n-1}} + \epsilon_n\ , & & \epsilon_n \ \sim \mathcal{N}\left(0, \delta_d^2\right) \end{array} \tag{41}$$

Samples drawn from the chosen initial state and state-transition distributions for the AR coefficients in (41) that produce poles of the system function that reside on or outside the unit circle are discarded. This step is a sufficient condition to ensure stability of the TVAR process.

B. CFS

For the CFS model, samples are drawn from the initial and state-transition distribution in the following way

$$\begin{array}{lll} \mathbf{f}_0 = \boldsymbol{\omega} & , & \boldsymbol{\omega} \ \sim \mathcal{U}\left(\mathbf{f}_l, \mathbf{f}_h\right) \\ \mathbf{f}_n = \mathbf{f}_{n-1} + \boldsymbol{\psi}_n\ , & & \boldsymbol{\psi}_n \sim \mathcal{N}\left(\mathbf{0}_{K\times 1}, \delta_{\mathbf{f}}^2\mathbf{I}_K\right) \\ \mathbf{b}_0 = \boldsymbol{\nu} & , & \boldsymbol{\nu} \ \sim \mathcal{U}\left(\mathbf{b}_l, \mathbf{b}_h\right) \\ \mathbf{b}_n = \mathbf{b}_{n-1} + \boldsymbol{\varsigma}_n\ , & & \boldsymbol{\varsigma}_n \ \sim \mathcal{N}\left(\mathbf{0}_{K\times 1}, \delta_{\mathbf{b}}^2\mathbf{I}_K\right) \end{array} \tag{42}$$

The models for the process noise and the observation noise variances are equivalent to those in (41). In order to exploit known properties of the formant frequencies in the particle

filtering the formant frequencies are constrained to the frequencies ranges $f_{n,1} \in [200, 1500]Hz$, $f_{n,2} \in [800, 2700]$Hz, $f_{n,3} \in [2000, 3800]$Hz and $f_{n,4} \in [2800, 5600]$Hz. Furthermore, the formant frequencies are constrained to be at least 200Hz apart, that is, $f_{n,k} - f_{n,k-1} > 200$Hz for $1 < k \leq K$. These constraints are similar to what was used in [12]. The formant bandwidths are constrained to the frequency interval $b_{n,k} \in [30, 500]$Hz. In (42), $\mathcal{U}(\mathbf{a}, \mathbf{b})$ denotes a multivariate uniform distribution on the domain $[\mathbf{a}, \mathbf{b}]$. The setting used in the following is $\mathbf{f}_l = [300, 1000, 2000, 3000]^T$, $\mathbf{f}_h = [1000, 2000, 3000, 4000]^T$, $\mathbf{b}_l = [30, 30, 30, 30]^T$ and $\mathbf{b}_h = [500, 500, 500, 500]^T$.

C. PFS

The parameter models for the formant frequencies and formant bandwidths in the PFS model are equivalent to those in (42) and the observation noise model equivalent to that in (41). The model for the formant gains are

$$\begin{array}{ll} \mathbf{g}_0 = \boldsymbol{\kappa} & , \quad \boldsymbol{\kappa} \sim \mathcal{N}(-2\mathbf{I}_{K\times 1}, 0.25\mathbf{I}_K) \\ \mathbf{g}_n = \mathbf{g}_{n-1} + \boldsymbol{\xi}_n\,, & \boldsymbol{\xi}_n \sim \mathcal{N}\left(\mathbf{0}_{K\times 1}, \delta_{\mathbf{g}}^2 \mathbf{I}_K\right) \end{array} \tag{43}$$

APPENDIX III
DERIVATION OF (13) AND (14)

The numerator in (11) can be expanded as follows using Bayes' rule

$$\begin{aligned} & p(\boldsymbol{\theta}_{0:n} \,|\, \mathbf{y}_{1:n}) = \frac{p(\mathbf{y}_{1:n} \,|\, \boldsymbol{\theta}_{0:n}) p(\boldsymbol{\theta}_{0:n})}{p(\mathbf{y}_{1:n})} \\ &= \frac{p(y_n \,|\, \mathbf{y}_{1:n-1}, \boldsymbol{\theta}_{0:n}) p(\mathbf{y}_{1:n-1} \,|\, \boldsymbol{\theta}_{0:n}) p(\boldsymbol{\theta}_n | \boldsymbol{\theta}_{0:n-1}) p(\boldsymbol{\theta}_{0:n-1})}{p(y_n \,|\, \mathbf{y}_{1:n-1}) p(\mathbf{y}_{1:n-1})} \\ &= \frac{p(y_n \,|\, \mathbf{y}_{1:n-1}, \boldsymbol{\theta}_{0:n}) p(\mathbf{y}_{1:n-1} \,|\, \boldsymbol{\theta}_{0:n-1})}{p(y_n \,|\, \mathbf{y}_{1:n-1}) p(\mathbf{y}_{1:n-1})} \\ &\quad \times p(\boldsymbol{\theta}_n | \boldsymbol{\theta}_{n-1}) p(\boldsymbol{\theta}_{0:n-1}) \end{aligned} \tag{44}$$

where it has been assumed that the parameters evolve according to a first-order Markov process, i.e. $p(\boldsymbol{\theta}_n \,|\, \boldsymbol{\theta}_{0:n-1}) = p(\boldsymbol{\theta}_n \,|\, \boldsymbol{\theta}_{n-1})$ and where the conditional independence $p(\mathbf{y}_{1:n-1} | \boldsymbol{\theta}_{0:n}) = p(\mathbf{y}_{1:n-1} | \boldsymbol{\theta}_{0:n-1})$ has been used. Inserting (12) and (44) into (11) then the weights obey

$$\begin{aligned} & w(\boldsymbol{\theta}_{0:n}) \\ &\propto \frac{p(y_n | \mathbf{y}_{1:n-1}, \boldsymbol{\theta}_{0:n}) p(\mathbf{y}_{1:n-1} | \boldsymbol{\theta}_{0:n-1}) p(\boldsymbol{\theta}_n | \boldsymbol{\theta}_{n-1}) p(\boldsymbol{\theta}_{0:n-1})}{p(y_n | \mathbf{y}_{1:n-1}) p(\mathbf{y}_{1:n-1}) \pi(\boldsymbol{\theta}_{0:n-1} | \mathbf{y}_{1:n-1}) \pi(\boldsymbol{\theta}_n | \boldsymbol{\theta}_{0:n-1}, \mathbf{y}_{1:n})} \\ &\propto \frac{p(\mathbf{y}_{1:n-1} | \boldsymbol{\theta}_{0:n-1}) p(\boldsymbol{\theta}_{0:n-1})}{p(\mathbf{y}_{1:n-1}) \pi(\boldsymbol{\theta}_{0:n-1} | \mathbf{y}_{1:n-1})} \\ &\quad \times \frac{p(y_n | \mathbf{y}_{1:n-1}, \boldsymbol{\theta}_{0:n}) p(\boldsymbol{\theta}_n | \boldsymbol{\theta}_{n-1})}{p(y_n | \mathbf{y}_{1:n-1}) \pi(\boldsymbol{\theta}_n | \boldsymbol{\theta}_{0:n-1}, \mathbf{y}_{1:n})} \\ &\propto w(\boldsymbol{\theta}_{0:n-1}) \cdot \frac{p(y_n | \mathbf{y}_{1:n-1}, \boldsymbol{\theta}_{0:n})\ p(\boldsymbol{\theta}_n | \boldsymbol{\theta}_{n-1})}{p(y_n | \mathbf{y}_{1:n-1})\ \pi(\boldsymbol{\theta}_n | \boldsymbol{\theta}_{0:n-1}, \mathbf{y}_{1:n})} \\ &\propto w(\boldsymbol{\theta}_{0:n-1}) \cdot \frac{p(y_n | \mathbf{y}_{1:n-1}, \boldsymbol{\theta}_{0:n})\ p(\boldsymbol{\theta}_n | \boldsymbol{\theta}_{n-1})}{\pi(\boldsymbol{\theta}_n | \boldsymbol{\theta}_{0:n-1}, \mathbf{y}_{1:n})} \\ &\propto w(\boldsymbol{\theta}_{0:n-1}) \cdot w(\boldsymbol{\theta}_n) \end{aligned}$$

with

$$w(\boldsymbol{\theta}_n) \propto \frac{p(y_n | \mathbf{y}_{1:n-1}, \boldsymbol{\theta}_{0:n})\ p(\boldsymbol{\theta}_n | \boldsymbol{\theta}_{n-1})}{\pi(\boldsymbol{\theta}_n | \boldsymbol{\theta}_{0:n-1}, \mathbf{y}_{1:n})}\ .$$

REFERENCES

[1] Y. Grenier, "Time-dependent ARMA modeling of nonstationary signals," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 31, pp. 899–911, Aug 1983.

[2] D. H. Klatt, "Software for a cascade/parallel formant synthesizer," *J. Acoust. Soc. Am.*, vol. 67, pp. 971–995, 1980.

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

[4] A. Doucet, S. Godsill, and M. West, "Monte carlo filtering and smoothing with application to time-varying spectral estimation," in *In Proc. ICASSP*, Istanbul, Turkey, June 2000, pp. 701–704.

[5] J. D. Markel and A. H. Gray, *Linear Prediction of Speech*. New York, USA: Springer-Verlag, New York, 1976.

[6] A. Doucet, N. de Freitas, and N. Gordon, Eds., *Sequential Monte Carlo in Practice*. Springer Verlag, 2001, ch. 1.

[7] D. Zhao and W. B. Kleijn, "HMM-based speech enhancement using explicit gain modeling," in *Proc. ICASSP 2006*, Toulouse, France, 2006.

[8] J. Hansen and B. Pellom, "An effective quality evaluation protocol for speech enhancement algorithms," 1998.

[9] Y. Hu and P. Loizou, "Evaluation of objective measures for speech enhancement," in *Proceedings of INTERSPEECH-2006*, Philadelphia, PA, 2006.

[10] R. Huber, "Objective assessment of audio quality using an auditory processing model," Dissertation, Oldenburg, University.

[11] T. Rohdenburg, V. Hohmann, and B. Kollmeier, "Objective perceptual quality measures for the evaluation of noise reduction schemes," in *Proc. International Workshop on Acoustic Echo and Noise Control*, Eindhoven, The Netherlands, September 2005.

[12] Y. Zheng and M. Hasegawa-Johnson, "Formant tracking by mixture state particle filter," in *IEEE ICASSP*, vol. 5, Quebec, Canada, May 2004.

Bibliography

[1] I. A. Aĭzenberg and A. P. Yuzhakov. *Integral representations and residues in multidimensional complex analysis*, volume 58 of *Translations of Mathematical Monographs*. American Mathematical Society, Providence, RI, 1983. Translated from the Russian by H. H. McFaden, Translation edited by Lev J. Leifman.

[2] M. Sanjeev Arulampalam, Simon Maskell, Neil Gordon, and Tim Clapp. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Trans. Signal Processing*, 50(2):174–188, February 2002.

[3] Paul M. Baggenstoss. The pdf projection theorem and the class-specific method. *IEEE Trans. Signal Processing*, 51(3):672–685, March 2003.

[4] Thomas Beierholm and Paul M. Baggenstoss. Speech music discrimination using class-specific features. International Conference on Pattern Recognition (ICPR), August 2004.

[5] Thomas Beierholm, Brian Dam Pedersen, and Ole Winther. Low complexity bayesian single channel source separation. Montreal, Canada, May 2004. Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing.

[6] Norman Bleistein and Richard A. Handelsman. *Asymptotic Expansions of Integrals*. Dover Publications, Inc., New York, 1986.

[7] James Ward Brown and Ruel V. Churchill. *Complex Variables and Applications*. McGraw-Hill, 7 edition, 2004.

[8] Yu A. Brychkov, H.-J. Glaeske, A. P. Prudnikov, and Vu Kim Tuan. *Multidimensional Integral Transformations*. Gordon and Breach Science Publishers, 1992.

[9] H. E. Daniels. Saddlepoint approximations in statistics. *The Annals of Mathematical Statistics*, 25(4):631–650, December 1954.

[10] Bert de Vries and Rob A.J. de Vries. An integrated approach to hearing aid algorithm design or enhancement of audibility, intelligibility and comfort. IEEE Benelux Signal Processing Symposium, April 2004.

[11] Arnaud Doucet, Mark Briers, and Stéphane Sénécal. Efficient block sampling strategies for sequential monte carlo. *J. Comp. Graph. Statist.*, 15(3):693–711, 2006.

[12] Arnaud Doucet, Nando de Freitas, and Neil Gordon. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, 3 edition, 2001.

[13] Arnaud Doucet, Simon Godsill, and Christophe Andrieu. On sequential monte carlo sampling methods for bayesian filtering. *Statistics and Computing*, 10:197–208, 2000.

[14] T. Eltoft, Kim Tasesu, and Te-Won Lee. On the multivariate laplace distribution. *IEEE Signal Processing Lett.*, 13:300–303, May 2006.

[15] Yariv Ephraim. A bayesian estimation approach for speech enhancement using hidden markov models. *IEEE Trans. Signal Processing*, 40(4):725–735, April 1992.

[16] G. Fant, J. Liljencrants, and Q. Lin. A four-parameter model of glottal flow. *STL-QPSR*, 26(4):1–13, 1985.

[17] Saeed Gazor and Wei Zhang. Speech probability distribution. *IEEE Signal Processing Lett.*, 10:204–207, July 2003.

[18] Saeed Gazor and Wei Zhang. Speech enhancement employing laplacian-gaussian mixture. *IEEE Trans. Speech Audio Processing*, 13:896–904, September 2005.

[19] Constantino Goutis and George Casella. Explaining the saddlepoint approximation. *The American Statistician*, 53(3):216–224, August 1999.

[20] U. Hartmann and K. Hermansen. Model based spectral subtraction used for noise suppression in speech with low SNR. In *Nordic Signal Processing Symposium (NORSIG)*, June 2000.

[21] Pamornpol Jinachitra and Julius O. Smith III. Joint estimation of glottal source and vocal tract for vocal synthesis using kalman smoothing and em algorithm. In *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, New York, 2005.

[22] U. Hartmann K. Hermansen, F. Fink and V. Moss Hansen. Hearing aids for profoundly deaf people based on a new parametric concept. In *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, New York, June 1993.

[23] Jun S. Liu. *Monte Carlo Strategies in Scientific Computing*. Springer-Verlag, 2001.

[24] Hui-Ling Lu and Julius O. Smith III. Glottal source modeling for singing voice. In *Proceedings of the International Computer Music Conference (ICMC-00, Berlin)*, pages 90–97, Berlin, 2000.

[25] David J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003. available from http://www.inference.phy.cam.ac.uk/mackay/itila/.

[26] J. D. Markel and Jr. A. H. Gray. *Linear Prediction of Speech*. Springer-Verlag, 3 edition, 1982.

[27] N. Metropolis. The beginning of the monte carlo method. *Los Alamos Science, Special Issue dedicated to Stanislaw Ulam*, pages 125–130, 1987.

[28] Tristan Needham. *Visual Complex Analysis*. Oxford University Press, 1997.

[29] Albert H. Nuttall. Saddlepoint approximation and first-order correction term to the joint probability density function of M quadratic and linear forms in K gaussian random variables with arbitrary means and covariances. NUWC Technical Report 11262, December 2000.

[30] Albert H. Nuttall. Saddlepoint approximations for the combined probability and joint probability density function of selected order statistics and the sum of the remainder. NUWC Technical Report 11509, February 2004.

[31] Albert H. Nuttall. private communication, 2005.

[32] Alan V. Oppenheim and Ronald W. Schafer. *Discrete-Time Signal Processing*. Prentice-Hall, 1989.

[33] Svetlana S. Petrova and Alexander D. Solov'ev. The origin of the method of steepest descent. *Historia Mathematica*, 24:361–375, 1997.

[34] A. Doucet R. van der Merwe, N. de Freitas and E. A. Wan. The unscented particle filter. Technical Report CUED/F-INFENG/TR 380, Cambridge University Engineering Department, Cambridge, England, August 2000.

[35] Lawrence R. Rabiner and Ronald W. Schafer. *Digital Processing of Speech Signals*. Prentice-Hall, 1978.

[36] A. F. M. Smith and A. E. Gelfand. Bayesian statistics without tears: A sampling-resampling perspective. *The American Statistician*, 46(2):84–88, May 1992.

[37] Melvin D. Springer. *The Algebra of Random Variables*. John Wiley & Sons, 1979.

[38] Raymond Veldhuis. A computationally efficient alternative for the liljencrants-fant model and its perceptual evaluation. *Journal of the Acoustical Society of America*, 103:556–571, January 1998.

[39] Jaco Vermaak, Christophe Andrieu, Arnaud Doucet, and Simon John Godsill. Particle methods for bayesian modeling and enhancement of speech signals. *IEEE Trans. Speech Audio Processing*, 10:173–185, March 2002.