Bayesian inference in multivariate nonlinear state-space models

Citation for published version (APA):

Shapovalova, Y. (2019). *Bayesian inference in multivariate nonlinear state-space models*. [Doctoral Thesis, Maastricht University]. ProefschriftMaken Maastricht. <https://doi.org/10.26481/dis.20190919ys>

Document status and date:

Published: 01/01/2019

DOI:

[10.26481/dis.20190919ys](https://doi.org/10.26481/dis.20190919ys)

Document Version:

Publisher's PDF, also known as Version of record

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.umlib.nl/taverne-license

Take down policy

If you believe that this document breaches copyright please contact us at:

repository@maastrichtuniversity.nl

providing details and we will investigate your claim.

Bayesian Inference in Multivariate Nonlinear State-Space Models

Yuliya Shapovalova

© Yuliya Shapovalova, Maastricht 2019

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form, or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission in writing from the author.

This book was typeset by the author using L^AT_EX.

Published by Universitaire Pers Maastricht

ISBN: 978-94-6380-440-0

Printed in The Netherlands by ProefschriftMaken Maastricht

Bayesian Inference in Multivariate Nonlinear State-Space Models

Dissertation

to obtain the degree of Doctor at Maastricht University,
on the authority of the Rector Magnificus
Prof. dr. Rianne M. Letschert
in accordance with the decision of the Board of Deans,
to be defended in public
on Friday 20th September 2019, at 10.00 hours

by

Yuliya Shapovalova

Supervisors:

Dr. M. Eichler

Prof. dr. A. Hecq

Assessment Committee:

Prof. dr. F.C. Palm (Chair)

Dr. N. Bastürk

Prof. dr. M. Lubrano (Aix-Marseille Université, France)

Prof. dr. S.J. Koopman (Vrije Universiteit Amsterdam)

In the dear memory of
my promoter Jean-Pierre Urbain
and my grandparents Nelly and Rudolf

Acknowledgements

This work is the result of a few years I spent in Maastricht as a Ph.D. student. There were a lot of people who directly or indirectly contributed to this work, academically and through other parts of my life. Once I started writing acknowledgments, I realized how many people I am grateful to and that it is inevitable only to mention some and not all here. For that, I apologize.

First, I would like to thank my supervisors Michael Eichler and Jean-Pierre Urbain. They gave me an incredible opportunity to pursue my academic interests. Jean-Pierre was a supportive promoter and a wonderful mentor. He always asked me core and essential questions which helped to shape my own views: about the role of education, universities, statistical schools of thought, life in general. Although I am sad he is not here to see the result, it is not too much to say that without Jean-Pierre the path of my career would have been different. I am thankful to Michael for the time and effort he put in our research discussions. He was the person who inspired me to pursue research and a Ph.D. Initially, I was planning to specialize in game theory during my Master's program. However, after the course 'Stochastic Processes' and after reading the description of a project on Michael's web-page on "Causal inference for multivariate time series with latent variables" and diving into the literature, there was no way back. I settled my mind on this topic. My Ph.D. project turned out to be not exactly what we originally planned, but we tackled the problems

step by step, and I am happy Michael was on board with me.

I want to thank the reading committee for their time and effort to read my thesis. Michel Lubrano, Nalan Bastrüik, Franz Palm and Siem-Jan Koopman, thanks to all of you for invaluable comments. All of you assessed this manuscript from different perspectives and asked relevant questions. It was illuminating to read your comments and look at my work from your points of view. Thank you very much.

I want to thank my friends and colleagues from the QE department. I had a wonderful time in Maastricht, and it would not have been so without all of you. Karin and Yolanda, I feel lucky to have had your support with numerous administrative matters. Your friendliness and willingness to help made the years during my Master's and Ph.D. brighter and more relaxed. When one comes to a new system from abroad, this kind of support is precious. I want to thank my paronyms Hanno and Yicong. You have been there for me during the best and the worst moments. Up until this day, I miss our afternoon coffee breaks and walks in the park next to the library. You filled my academic days in Maastricht with humor, friendship, and lots of cups of coffee. I want to thank my office mates Andrey and Jan. Thank you, Andrey, for being a fantastic office mate and a friend. You always helped me in the times of need, and I feel fortunate to have had a lot of exciting discussions with you. Thank you Jan, it was a pleasure to see you every week and talk about wild nature, bush crafting and ice-climbing in the breaks. I have been fortunate enough to have other friends and colleagues who made my years in Maastricht special. Thank you, Marina, for being such an open-minded person, I have learned a lot from you. Sean, thank you for the great time in Paris which we had during our accidentally overlapping research visits. It was comforting to know someone in an unbearably large city. Thank you, Rasmus, for the coffee breaks and brainstorming different research ideas, your enthusiasm is always inspiring. I want to thank all of my other colleagues and fellow students in QE department: you made going to work every day more interesting than I thought it would be (and I already had high expectations!).

During my Ph.D., I went on two research visits. I want to thank Vera

who, during her research visit in Maastricht, lightened up the idea of visiting Aix-Marseille University. After talking with Jean-Pierre, we decided that visiting Professor Michel Lubrano, an expert in Bayesian statistics, would be valuable. Thank you, Michel, for hosting me at Aix-Marseille University. My stay has broadened my knowledge (in particular on mixture models — thanks to the reading group) and allowed me to connect with wonderful academics in your research group. Up until this day, I keep telling stories about the famous lunch meetings the group had together. My second research visit was to University Paris-Dauphine. Professor Christian Robert was the host who also connected me with Professor Nicolas Chopin. I appreciate the valuable discussions we have had, and I definitely have learned a lot. You introduced me to the new topics such as Approximate Bayesian Computation (thanks to you I went to a related workshop later on), and you also enriched my knowledge of the literature on Sequential Monte Carlo methods and Particle Markov Chain Monte Carlo. Thanks to both of you for being kind and inspiring academics.

I have been fortunate to have had friends who inspired and supported me. Thank you Artem, Alla, and Alexandr for helping me to pursue a Master's degree in Maastricht. Thank you Artem for your support and enthusiasm, because of you I picked this Master's program in Maastricht, and I am happy where it took me. A big thanks go to my slackline and climbing friends: all these years you made my non-work time unforgettable. Thank you, Yon and Jeroen for being great climbing partners, travel companions and inspiring me for challenging projects. Thank you, Bram, for being a good friend and introducing me to bushcrafting: you made me look at nature from a different perspective as well as at other numerous things we have discussed during our forest walks. Thank you climbing friends and members of the MaasSAC, too many to mention by name. Thank you, Tyas, Britt, Job, Gabi, Johanna and Jeroen for pursuing together the art of balancing on a slackline. We might not meet so often anymore, but I have warm memories of our relaxed days in the park. Nalan and Rui, thank you for your invaluable advise when I needed it and occasional mountain bike rides.

ACKNOWLEDGEMENTS

After the end of my Ph.D. contract, I worked as a Research Associate at Manchester University. I want to thank Professor Magnus Rattray for guiding me in the new field and being supportive even after I left the group. It was a pleasure to work with you in a vibrant scientific environment at Manchester University, and I hope our collaboration will continue in the future. I want to thank my current research group at Radboud University in Nijmegen, which inspires me every day to grow and be a better researcher and teacher. In particular, I would like to thank Professor Tom Heskes for giving me this opportunity and being an excellent mentor. You motivate by being a good example and always ask relevant questions. Thank you for your trust and encouragement.

Last but not least, I would like to thank my family. I am forever grateful to my grandparents, Rudolf and Nelly, who raised me as a kid and made me the person I am. They have put a lot of time and effort into my education, both academic and non-academic. I want to thank my mother and sister, Elena and Ksenia, for their constant support. We live far away from each other, but you are always in my heart. Thank you, Michael, for being there for me every day. We have climbed a lot of mountains together, literally and figuratively, and you always hold the other end of the rope tight and have my back. Thank you also for helping to proofread some parts of this manuscript.

Yuliya Shapovalova

Maastricht, September 2019

Contents

Acknowledgements	vii
1 Introduction and Outline	1
2 Advances in Bayesian Computation for Estimation of the Stochastic Volatility Models: a comparative review	9
2.1 Introduction	10
2.2 Stochastic Volatility Model	14
2.2.1 Univariate Case	14
2.2.2 Multivariate Case	18
2.3 QML and Kalman Filter	20
2.4 Bayesian Inference	21
2.5 SMC and MCMC	23
2.5.1 Introduction to Markov Chain Monte Carlo Methods	24
2.5.2 Sequential Monte Carlo	28
2.5.3 Approximate of marginal likelihood with Auxiliary Sequential Importance Resampling (ASIR)	33
2.5.4 Particle Metropolis-Hastings	34
2.5.5 Auxiliary particle filter for SV model	35
2.5.6 Variance of the estimated likelihood	44
2.5.7 Iterated Auxiliary Particle Filter	50
2.5.8 Particle Gibbs Sampler	54

CONTENTS

2.5.9	Overall comments on SMC and PMCMC	59
2.6	Riemann Manifold Langevin and Hamiltonian Monte Carlo	60
2.6.1	Hamiltonian Monte Carlo	61
2.6.2	Computational Implementation: discretizing Hamiltonian's equations	65
2.6.3	Constructing MCMC with Hamiltonian Dynamics .	67
2.6.4	Metropolis Adjusted Langevin Algorithm	69
2.6.5	Riemann Manifold Metropolis Adjusted Langevin Algorithm	70
2.6.6	Riemann Manifold Hamiltonian Monte Carlo	71
2.6.7	RM-HMC and mMALA for SV model	71
2.6.8	Illustration of RM-HMC with simulated data	75
2.6.9	Overall comments on HMC	80
2.7	Approximate Methods	80
2.7.1	Integrated Nested Laplace Approximation (INLA) .	81
2.7.2	Fixed-Form Variational Bayes	85
2.7.3	Overall comments on Approximate Methods	92
2.8	Discussion	92
2.A	Appendix: Simulated Data	95
3	Volatility Spillovers with Application to Stochastic Volatility Models	97
3.1	Introduction	98
3.2	Multivariate Stochastic Volatility Model	101
3.3	Measuring spillover effects	103
3.3.1	Granger-causality in volatility	103
3.3.2	Graphical representation of the causal structure . . .	104
3.3.3	Forecast Error Variance Decomposition	107
3.3.4	Variance Shares	108
3.3.5	Total Spillover Effect	109
3.3.6	Directional Spillover Effect	109
3.4	Estimation	110
3.4.1	Outline of the estimation	110

3.4.2	Simulated Likelihood with ASIR	114
3.4.3	Sampling from the posterior	119
3.4.4	Implementation details	121
3.5	Simulation Experiments	121
3.5.1	Prior distributions	122
3.5.2	Bivariate simulated example	122
3.5.3	Trivariate example	124
3.5.4	Five-dimensional example	140
3.5.5	Model Selection with Spike and Slab Prior	143
3.6	Empirical applications	146
3.6.1	Spillovers between Australia and New Zealand . . .	146
3.6.2	Five-dimensional Application	154
3.7	Discussion	161
3.A	Appendix: Five-dimensional empirical example	164
3.B	Appendix: Resampling	165
3.C	Appendix: Posterior Draws	165
4	Multivariate Models for Count Data	167
4.1	Introduction	168
4.2	Count Time Series	171
4.2.1	Typical examples of count data	171
4.2.2	Poisson distribution	172
4.3	Observation driven models	174
4.3.1	MACI (INGARCH)	174
4.3.2	Quasi-maximum likelihood for MACI models	176
4.3.3	Log-linear autoregressive model	178
4.3.4	Quasi-maximum likelihood for log-linear models . .	178
4.3.5	Multivariate Poisson Distribution	179
4.4	Parameter driven model: nonlinear state-space model . . .	181
4.4.1	Multivariate SSM	182
4.4.2	Bayesian Inference	183
4.4.3	Illustrative example with bivariate simulated data .	190
4.4.4	Forecasting with SSM	194

CONTENTS

4.5	Model comparison and prediction assessment	195
4.6	Simulation Examples	198
4.6.1	Prediction assessment of the simulation study	198
4.7	Empirical Applications	200
4.7.1	Road accidents: Utrecht, Dordrecht and Haarlem- mermeer	200
4.7.2	Bank Failures	201
4.7.3	Transactions	203
4.8	A note on copula approach	205
4.9	Discussion	209
5	Conclusion	211
6	Valorisation	217
	Bibliography	221
	Nederlandse Samenvatting	233
	Curriculum Vitae	237

Chapter 1

Introduction and Outline

“The theory of probabilities is at bottom nothing but common sense reduced to calculus; it enables us to appreciate with exactness that which accurate minds feel with a sort of instinct for which oftentimes they are unable to account.”

-Introduction to Théorie Analytique des Probabilités,
Pierre-Simon Laplace

When applying statistical modeling and inference to the sampled data, we try to understand, explain and predict real-world phenomena. By studying hidden patterns in the data we can learn about the world and either enrich our knowledge or improve different aspects of our life, and sometimes both. In order to learn from the data, we need some mathematical structure and foundations. In applied statistics and econometrics researchers usually try to develop plausible models that are representations of some specific real-world problems. One of the modeling issues that many scientific fields face are latent (unobserved) processes. This issue is shared in biology, psychology, medicine, physics, and economics among others. It is appealing to include latent processes into a model from multiple points of view. First, they can represent things that belong to the physical world, but for some reason were not recorded. Second, they can account for abstract concepts such as behavioral state in psychology or volatility process in financial econometrics. Third, latent processes can also help us express

dependency that is otherwise hard to model. In this case the dependency in observations is modeled through the dependency in latent variables. State-space models, in some fields of applications referred to as Hidden-Markov models, are equipped to deal with the issue of modeling latent/unobserved processes. The idea of this modeling approach is that every observation depends on an unobserved hidden state in some functional form. This means that we have two equations: one is called *observation equation* and another *state equation* which specifies latent variables in the model. When the functional form between observations and states is linear, we say that we work with a linear state-space model, and correspondingly with a non-linear state-space model when the functional dependence is nonlinear. Part of the inferential problem in state-space models usually lies in filtering out the latent process from the noisy observations. When the model is linear, variations of Kalman filter (Kalman, 1960) are often successfully applied. In nonlinear state-space models usually other filtering techniques have to be applied, such as Sequential Monte Carlo methods (Halton, 1962; Doucet et al., 2001). Intuitively, when the dependence is more complex, it becomes harder to extract the latent process from the observed noisy data. Although state-space models can generally be applied to different types of data, in this thesis we will focus on time series data – in our case, the data comes with fixed indexes and is equally spaced over time. For more general introduction of state-space modeling for time-series data see Durbin and Koopman (2012).

Various approaches have been proposed for the estimation of the non-linear state-space models, however, the problem is not trivial. Both fast approximate techniques and more exact methods have their own problems when it comes to the estimation. Moreover, the specification we are considering, where latent processes are defined through VAR(p) framework with non-diagonal autoregressive matrix, has not been estimated in the literature in high-dimensional cases which makes the problem interesting from methodological point of view. The reward for a good solution however, can be a powerful tool that can be used in many interesting applied problems. To get an idea of the complexity of the model specification with which

we will be dealing in this thesis, imagine that the observation equation is specified not just for one process, but for multiple time series processes and they are driven by multiple latent processes. Moreover, let us allow for dependence between the latent processes as Granger-causal relationships and also for contemporaneous correlation in the observations as it is specified in $\text{VAR}(p)$ models. It is clear that the problem of filtering and estimation of such a model becomes more difficult with the increased dimensionality. Already univariately, the estimation of the likelihood of the nonlinear state-space model is challenging since the likelihood function does not have an analytical expression. The multivariate extension will bring more ‘stepping stones’ that one has to discover on the way. However, when one does find these stepping stones, one might have a very flexible model that could accommodate various data types and applied problems.

The first part of this thesis focuses on a specific type of a nonlinear state-space model that is used in financial econometrics – Stochastic Volatility (SV) models (Shephard and Torben, 2009). Note, that when we talk about stochastic volatility models, we mean stochastic *latent* volatility models, i.e., the observed log-returns, in this case, depend solely on an independent latent volatility process. Thus, the observation equation, in this case, is specified in a nonlinear form where log-returns depend exponentially on the volatility. We make this emphasis because Generalized Autoregressive Conditional Heteroskedasticity (GARCH) models introduced in Bollerslev (1986) generally also fall into the class of stochastic volatility models (Shephard and Torben, 2009). GARCH models gained popularity due to the simplicity of estimation, however with the availability of computational power and rapidly developing computational methods, flexibility and interpretability of the SV models become more and more appealing.

The presence of a latent process in stochastic volatility models makes them very flexible and it allows us to model complex dependencies. However, this in turn leads to the intractability of the likelihood and hence the standard inferential procedures cannot be used. A natural framework into which inferential problems fit is the Bayesian framework, and even though we will discuss how the problem is often tackled with classical econometric

methods and numerical techniques, in this thesis, we will be concerned with designing a fully Bayesian inferential approach for this class of models.

Univariate stochastic volatility models have been studied extensively for many years. With the availability of computational power, developments in Bayesian statistics and machine learning, new methods are being proposed to tackle the inferential problem. One of the first methods, namely quasi-maximum likelihood inference with Kalman filter (Ruiz, 1994), relies on the linearization of the observation equation. Once the logarithmic transformation is applied, one is likely to lose some information from the model, and hence this method is not favored, especially when it comes to multivariate extensions with complex dependencies in the latent process. Another filtering method that is appropriate for nonlinear states-space models is called particle filtering, also known as Sequential Monte Carlo methods (see Halton, 1962). For the original publication and an introduction that is relevant to our problem (see Doucet et al., 2001), which allows estimating the likelihood of nonlinear state-space models. The intuitive idea behind this is that we generate N particles that represent our best 'guess' about the state of the latent process. These particles then are weighted according to the likelihood of the data given the states and the parameters. The particles that have higher weights are more likely to be propagated to the next step. It has been shown, that the likelihood estimate is the byproduct of this filtering procedure (see, for example, Doucet and Johansen, 2009). It has also been shown that this estimate of the likelihood can be incorporated into standard Markov Chain Monte Carlo (MCMC) schemes, and in particular, can be used with Metropolis-Hastings algorithm. MCMC methods originate in Metropolis et al. (1953) and provide a tool to sample from the distributions from which direct sampling is hard or impossible, which is the case with the nonlinear state-space models we are considering in this thesis. MCMC methods combined with particle filtering methods for estimation of the likelihood can provide a coherent framework for the estimation problem we are facing in Bayesian framework. Both methods, MCMC and particle filters, however, can exhibit several problems in multivariate extensions. Multiple problems associated with particle filtering were documented even

in the univariate case. One is that the variance of the estimated likelihood exponentially grows with the dimension (Chopin et al., 2004). A number of approaches have been proposed to solve this problem which we discuss in Chapter 2. The advantage of particle filtering and Markov Chain Monte Carlo methods introduced in Andrieu et al. (2010) is simplicity in implementation and the 'exact' inferential solution, the disadvantage is the computational burden these methods imply, which clearly increases with the dimension. In the attempts to solve computational issues, researchers in statistics and machine learning have developed approximate techniques following the idea: 'it is far better to have an approximate answer to the right question, which is often vague, instead of the exact answer to the wrong question, which can always be made precise'. This translates into the idea that it is better to give an approximate answer to a complex model than a precise answer to something that is overly simplified. Chapter 2 of this thesis therefore considers different methodologies for estimation of the univariate stochastic volatility models with the intention of comparing them and identifying which are good candidates for multivariate extensions. We compare the methods from multiple angles. We look into the computational speed of the methods, how they will change with increasing dimensionality and whether there are means to improve them by using more advanced programming implementations. Another important point we are considering in the comparison is flexibility. Frequently, researchers and practitioners in financial econometrics use different distributions for their models depending on the problem at hand. Thus the method should be easily adjustable to such changes. Extendibility to the high-dimensional case is one of the main concerns too. In practice, when one looks at the dependence structure, one wants to look at high-dimensional data. In particular, if we want to look at, say, Granger-causality and we include in the model only two financial time series we are more likely to get spurious effects in our results. To sum up, we will compare methods in terms of computational complexity, flexibility and possible high-dimensional extensions. We will compare the the following methods: particle Markov Chain Monte Carlo (Andrieu et al., 2010), Hamiltonian Monte Carlo (also known as Hy-

brid Monte Carlo) (Duane et al., 1987; Girolami and Calderhead, 2011a), Integrated Nested Laplace Approximation (Rue et al., 2009), Variational Bayes (Salimans et al., 2013).

In Chapter 3 we build a fully Bayesian framework for detecting and measuring volatility spillover effects on financial markets with a multivariate stochastic volatility model. The estimation framework is built with the insights gained from Chapter 2. The advantage of using stochastic volatility approach instead of the computationally more straightforward GARCH framework, is flexibility and interpretability. Intuitive interpretation is possible because the latent process is modeled through a $\text{VAR}(p)$ model. Thereby, it incorporates both contemporaneous correlation and Granger-causal structure, where the latter is encoded into autoregressive matrices of the $\text{VAR}(p)$ process. By using variance shares, decomposition volatility spillover measures can be constructed in a similar way as in the $\text{VAR}(p)$ framework. Furthermore, we estimate these measures in a Bayesian framework and provide moments of the posterior distributions of these measures to make further conclusions about possible presence of the spillover effects. We also consider model selection approaches with spike and slab priors, and propose an algorithm for exploring not only the parameter space, but also the model space.

Nonlinear state-space models provide a flexible framework for financial time series models, but a similar modeling strategy can be applied in other fields, since latent or unobservable data is common in science. Moreover, the strategy for detecting and measuring dependencies using these models can be conveniently extended to count data models and we are considering this problem in Chapter 4 of this thesis. The distribution of counts usually is modeled through a Poisson distribution. The difficulty with this arises however with multivariate modeling, as probability mass function of multivariate Poisson distribution is complicated, which make inference cumbersome. In addition, a multivariate Poisson distribution does not allow modeling negative dependency in the data. A standard approach in the literature is the use of Autoregressive Conditional Intensity models, also referred to as Integer-valued GARCH models. The disadvantage of

this approach is that stationarity condition requires model parameters to be positive which is quite restrictive. Moreover, it imposes problems for the case when parameters lie close or on the boundary of the parameter space. According to our knowledge no theoretical properties have been established for this case, and hence confidence intervals and standard errors are not easily available. An alternative approach that appeared in the literature more recently are log-linear models (Fokianos and Tjøstheim, 2011). Through appropriate log-transformations they bypass the problem of the parameters that lie close to/on the boundary of the parameter space and allow them to be negative. Modeling of the joint distribution of the count, however, remains a problem. Nonlinear state-space models seem like a good alternative that provides us with flexibility and allows for modeling positive and negative Granger-causal feedback as well as positive and negative contemporaneous correlations. The aim of Chapter 4 is to gain insights into whether flexibility of state-space approach is beneficial enough to pay the computational costs it carries with it. We discuss fully Bayesian inference for this model and compare various count-modeling approaches in terms of model fit and forecasting performance.

Summary and concluding remarks

Nonlinear state-space models have applications in many research fields. They provide a researcher with a flexible tool for modeling latent processes which can be the result of missing data, the data that has not been recorded, but is relevant for the model, or the data we are not able to observe. Because of this appealing feature, these models have been studied extensively. However, many proposed methods dealing with univariate models are already challenging in terms of estimation: there is a trade-off between quality of the estimation and computational burden. However, real-world applications often require multivariate modeling. We believe that nonlinear state-space models can provide a powerful tool for modeling various types of dependencies in multivariate- and in the future in high-dimensional-frameworks. Since there is no off-the-shelf solution for

multivariate estimation of these models, in Chapter 2 we discuss possibilities that one might consider. We look into methods that were developed in Bayesian statistics and machine learning and compare them with the goal of developing an efficient method for estimation that can deal with complex dependencies in the latent processes. In Chapter 3 we propose a novel method for measuring volatility spillovers in the framework of the multivariate (latent) stochastic volatility model and describe a fully Bayesian approach for doing such inference. In Chapter 4 we review the application of nonlinear state-space models to count data. There are observation and parameter-driven models, and multivariate comparison of these two classes of models is currently limited in the literature. We compare the multivariate nonlinear state-space model with multivariate conditional intensity and multivariate log-linear models, where the latter is more flexible and thereby is a more natural competitor to the state-space approach. Since the research with application to multivariate nonlinear state-space models is active and there are many ways in which one can proceed with improving the methodologies and extending the models, in Chapter 5 we propose suggestions for future research.

Chapter 2

Advances in Bayesian Computation for Estimation of the Stochastic Volatility Models: a comparative review

Stochastic volatility models (SVM) belong to the class of state-space models. Volatility in this framework is an independent latent process as opposed to GARCH-type models. Latent processes in the model make the estimation difficult. Standard techniques such as Quasi-Maximum Likelihood and General Method of Moments do not perform well which has been documented in the literature. Particle filters have been considered as the most promising solution for quite long time, intuitively explicit implementation and good performance of simulated likelihood makes this procedure appealing. However, it is computationally very demanding. For practitioners speed of estimation might be crucial and hence, methods that might work faster are of interest. Our primary goal in this review is to discuss advantages and disadvantages of the methods for multivariate extensions. Com-

putational intensiveness of MCMC methods suggests that for more efficient estimation combination with approximate techniques should be considered. We review various Sequential Monte Carlo approaches in combination with particle Markov Chain Monte Carlo, Hamiltonian Monte Carlo, Fixed-Form Variational Bayes, Integrated Nested Laplace Approximation. We illustrate the implementation of the methods that are promising for the multivariate extensions with some simulation examples and compare some of them. We conclude with various possible strategies for multivariate estimation of the stochastic volatility models.

2.1 Introduction

The field of Bayesian statistics and machine learning has advanced in the last years quite rapidly. The methods that have been developed lately do not often find fast assimilation across different fields. In this review, we aim to provide the reader with new methodologies that try to solve the estimation problem in models with latent variables and intractable likelihoods. We are in particular interested in nonlinear state-space models and applications of these methods to stochastic (latent) volatility models. There exist multiple studies that conducted review and comparison of the methods of estimation of the stochastic volatility models (Shephard and Torben, 2009; Platanioti et al., 2005; Asai et al., 2006). We briefly mention some of the methodologies that have been reviewed, however, most of the methods considered in this paper have not entered those reviews. The other point that distinguishes our review from existing ones is the focus. In particular, we aim to identify methods promising for the estimation of multivariate and high-dimensional stochastic volatility models.

In financial econometrics literature, GARCH-type models prevail since they are much simpler to estimate. Stochastic (latent) volatility models, however, can be a more natural framework for modeling asset returns. They can provide a flexible and intuitive tool for applications in financial econometrics as well as some other disciplines. In particular, multivariate stochastic volatility models offer an attractive framework for detection and

measuring *volatility spillover effects*. Volatility spillovers in this framework can be defined through causal links in the latent (unobservable) volatility process which is modeled with a Vector Autoregressive model (VAR(p)). Insights about the causal structure can help to identify a type of the relationship (Granger-causality or/and contemporaneous correlation) between the financial markets. Obtaining such information can be relevant for portfolio managers and policymakers, in particular, distinguishing correlation and Granger-causality can be helpful in decision making. For example, for policymakers, it is relevant to know which economies affect the economy of their country or region so that they can create sufficient bailout funds for supporting their financial markets in the time of crisis. This kind of dependency, however, can be assumed in other applications. Thus, solving the estimation problem of these models and being able to scale estimation to higher dimensions can be beneficial in various fields.

The stepping stone of estimation of the nonlinear state-space models (and stochastic volatility models in particular) lies in the intractability of the likelihood which is the result of the presence of unobservable process in the model and nonlinear dependence between this process and the observed data. The likelihood, however, can be estimated with particle filter methods, also known as Sequential Monte Carlo. It is a computationally intensive procedure, however, depending on the problem and the data it can provide excellent results. Moreover, when we are in the Bayesian framework, we are interested in estimating the posterior distribution of the parameters of the model by sampling from an appropriate distribution. However, when the posterior is not available in closed form, or it is hard to get it, other sampling strategies have to be considered. A good starting point for this would be a Metropolis-Hastings algorithm which is an off-shelf solution nowadays for sampling from the posterior distribution when sampling from the posterior directly is not possible. It works quite well when the number of parameters in the model is small. However, the textbook implementation might not work well in multivariate state-space models. In this paper, we discuss implementation strategies related to multivariate models. Note, that Metropolis-Hastings algorithm required an evaluation

of the likelihood which is not available in the closed form for the class of models we are considering. We can overcome this issue by combining particle filtering together with Markov Chain Monte Carlo methods; the idea initially appeared in Andrieu et al. (2010). The authors elegantly combine SMC methods and MCMC (PMCMC) and show that this combination makes inference possible in a wide range of models. This method can be considered 'exact' in a sense that Markov Chain Monte Carlo methods converge under some assumptions, and it also has been shown that under some weak assumptions particle Markov Chain Monte Carlo methods remain the same properties as if the likelihood was available analytically. It is a very strong result, and PMCMC provide a framework for inference in complex models, including nonlinear state-space models.

Bayesian statistics and machine learning offer 'exact' solutions such as PMCMC, but clearly, this kind of inference imposes some computational burden. Thereby, the researchers have been developing approximate approaches to this problem. Approximate solutions, however, would not always provide sufficiently good results, especially when it comes to the applications we are considering one would rather know a more clear answer to how much a particular economy depends on the others. But depending on how much time we are willing to spend on finding a good solution, one might have to consider a trade-off between computational burden and estimation accuracy. As a result, we are reviewing the methods relevant for estimation of nonlinear state-space models, and stochastic volatility models in particular, from different points of view. First, computational complexity and possibility to improve computational speed by some means (depending on the method, it can be for instance exploiting parallelization or utilizing sparse matrix computations). Second, estimation accuracy. In univariate models, in particular, the variance of the latent process seems to present a challenge. In multivariate case, all parameters that encode (causal) relationships would be challenging to estimate. Third, we are also interested in the flexibility of the method considered. Different financial times series exhibit different features (for example, when one deals with heavy tails one would be interested in relaxing Gaussian assumption and

use t-Student distribution instead), and thus it is of interest how easy it is to apply the methods to different specifications of the model and whether the algorithms can adopt this features easily.

We believe that the optimal solution might lie somewhere in the middle ground, meaning that sufficient accuracy of estimation in feasible computational time can be achieved by combining approximate and exact techniques. Moreover, as computationally expensive methods sometimes might require some degree of manual calibration. Thus, approximations can also provide a good starting point for obtaining results with more accurate methods and hence reduce the time spend for the calibration of the algorithms. Understanding of pros and cons of 'exact' and approximate methods can help us to develop more efficient algorithms for a feasible methodological procedure in multivariate and high-dimensional nonlinear state-space models.

In this review paper, we focus our attention on the following methodologies and provide a comparison for some of the methods via a simulation study: various Sequential Monte Carlo (SMC) methods in combination with particle Markov Chain Monte Carlo (PMCMC) methods proposed in Andrieu et al. (2010). From the methods reviewed it is the most intensive computationally methods. However, it also provides good estimation results when implemented efficiently and calibrated sufficiently well. Calibration can be difficult depending on a particular model and data. We further review Hamiltonian Monte Carlo methods that were originally developed in physics literature and thereby have original interpretation through differential equations and Hamiltonian dynamics with which movement of an object is described. This approach found a beautiful interpretation in statistical literature (Duane et al., 1987) and sometimes is also referred to as Hybrid Monte Carlo, where we consider our parameter vector as a moving object in the parameter space. Further, by exploiting Hamiltonian dynamics, we can make proposals that are more likely to be accepted than the once being generated with random walk proposals in Metropolis-Hastings algorithm. Calibration of Hamiltonian Monte Carlo is not necessarily an easy task. However, when calibrated properly it provides very good mixing of

the samples from the posterior distribution. We further also consider multiple approximate approaches, such as Fixed-form Variational Bayes (VB) algorithm (Salimans et al., 2013) and Integrated Nested Laplace Approximation (INLA) (Rue et al., 2009) for which we perform some simulations studies.

2.2 Stochastic Volatility Model

2.2.1 Univariate Case

Stochastic volatility (SV) models are concerned with modeling asset prices or asset returns depending on how the model is formulated. These models are built in such a way that they can mimic *stylized facts* about financial markets. Stylized facts are empirically observed statistical properties of asset prices and asset returns. Typical examples of stylized facts are

- *Volatility clustering and persistence*: the big changes in asset returns tend to be followed by big changes and small changes in asset returns tend to be followed by small changes, in other words, there are periods of large fluctuations and small fluctuations. (Mandelbrot, 1997)
- *Leverage Effect*: the changes in stock prices may be negatively related to the changes in volatility. (Black, 1976)
- *Co-movements*: different stocks tend to exhibit co-movements which means that if the volatility of one stock changes in a specific direction, volatilities of the other stocks tend to change in the same direction. (Black, 1976)

One of the earlier works that received a lot of attention in financial literature and proposed a mathematical model that tried to explain the dynamics of financial markets is Black and Scholes (1973). Numerous continuous time stochastic volatility models have been proposed since then, among the first ones multiple variants should be mentioned Hull and White (1987), Johnson and Shanno (1987), Wiggins (1987), Heston (1993). The model we will be considering in this chapter can be viewed as a discrete version of the

model in Wiggins (1987) derived by using Euler–Maruyama approximation. Stochastic volatility model in continuous time can be written as

$$ds(t) = \sigma(t)dB_1(t), \quad (2.1)$$

$$\ln \sigma^2(t) = \alpha + \beta \ln \sigma^2(t)dt + \sigma_\eta dB_2(t), \quad (2.2)$$

where $s(t)$ is log of asset price, $\sigma^2(t)$ is the volatility, $B_1(t)$ and $B_2(t)$ are Brownian motions that satisfy $\text{corr}(B_1(t), B_2(t)) = \rho$. In case where $\rho < 0$ there is leverage effect present. The discrete model then follows by using Euler–Maruyama approximation

$$y_t = \sigma_t \epsilon_t, \quad (2.3)$$

$$\ln \sigma_{t+1}^2 = \alpha + \phi \ln \sigma_t^2 + \sigma_\eta \eta_{t+1}, \quad (2.4)$$

where y_t is logarithmic return, $\epsilon_t = B_1(t+1) - B_1(t)$, $\eta_{t+1} = B_2(t+1) - B_2(t)$, $\phi = 1 + \beta$. Further, $\epsilon_t \sim N(0, 1)$ and $\eta_t \sim N(0, 1)$, $\text{corr}(\epsilon_t, \eta_{t+1}) = \rho$.

Further, by defining $h_t = \ln \sigma_t^2$ and $\sigma_t^2 = \exp(h_t)$ we get state-space representation of the model that is commonly used nowadays

$$y_t = \exp(h_t/2)\epsilon_t, \quad (2.5)$$

$$h_{t+1} = \mu + \phi h_t + \eta_{t+1}, \quad (2.6)$$

where y_t are log-returns that are observed and volatility h_t is latent and drives the dynamics of y_t . Further, the parameter ϕ is the persistence parameter of the process and reflects one of the stylized facts of financial returns, namely volatility persistence. Further, if $\phi > 0$ and $\exp(h_{t-1}/2)$ is large then $\exp(h_t/2)$ will tend to be large too. Hence, the model can account for volatility clustering. Finally, one can also incorporate leverage effects by defining negative correlation between noise terms ϵ_t and η_{t+1} . Intuitive interpretation of this effect usually goes as follows: bad news tend to decrease the price, which means that financial leverage increases, the firm becomes riskier and thus expected volatility also increases. The leverage effects in this model have been studied in Harvey and Shephard (1996). The

stochastic volatility model can be parametrized in multiple ways; often the following alternatives are considered (Platanioti et al., 2005):

Other ways to parametrize this model are presented in Equations (2.7) and (2.8). The left-hand side version of the model corresponds to that of Jacquier et al. (2004). The right-hand side version is a different way to define the scaling parameter, in this case, it is β . For identifiability reasons only β or μ as in Equation (2.6) should be included in the model.

$$y_t = \sqrt{h_t} \epsilon_t \qquad y_t = \beta \exp(h_t/2) \epsilon_t \qquad (2.7)$$

$$\log h_t = \mu + \phi \log h_{t-1} + \eta_t \qquad h_t = \phi h_{t-1} + \eta_t. \qquad (2.8)$$

Note, that Jacquier et al. (2004) define the leverage effect as correlation between ϵ_t and η_t , so the correlation between noise terms is contemporaneous while Harvey and Shephard (1996) model correlation between ϵ_t and η_{t+1} which corresponds to correlation of the returns with one-step ahead volatility. Yu (2005) shows that approach of Harvey and Shephard (1996) is preferable. In particular, while in case of Harvey and Shephard (1996) the model is a martingale difference sequence, i.e., the past does not help to predict the future of the time series, in case of Jacquier et al. (2004) it is not. Hence, in the latter case, the efficient market hypothesis is violated.

In the remainder of this manuscript we will work with either specification of the model defined in Equations (2.5) and (2.6) or in right-hand side of Equations (2.7) and (2.8). Note that these models are equivalent and we interchange the representation either for the convenience of using some of the methods or for comparison with other work. In the literature, both specifications are frequently used and in some papers (for example, Kim et al., 1998) the transition from one specification to another is conducted by observing that $\beta = \exp(\mu/2)$.

We ensure that y_t is stationary by assuming that h_t is stationary. Thus, the absolute value of the persistence parameter should be smaller than 1, i.e. $|\phi| < 1$. The unconditional first and second moments of the latent

process h_t are

$$\mathbb{E}(h_t) = \frac{\mu}{1 - \phi}, \quad \text{Var}(h_t) = \frac{\sigma_\eta^2}{1 - \phi^2}. \quad (2.9)$$

Since we are going to focus on the methods that are usually applied in the Bayesian paradigm, we find it useful to write the model through distributional assumptions

$$y_t \sim p(y_t \mid h_{0:t}, \theta, y_{1:t-1}) = N(0, \exp(h_t)) \quad (2.10)$$

$$h_t \sim p(h_t, \theta \mid h_{0:t-1}, \theta, y_{1:t-1}) = N(\mu + \phi h_{t-1}, \sigma_\eta^2) \quad (2.11)$$

$$h_0 \sim \pi(h_0 \mid \theta) \quad (2.12)$$

$$\theta \sim \pi(\theta) \quad (2.13)$$

where (2.10) is the observation (measurement) equation, (2.11) is the state dynamics equation and (2.12) and (2.13) are prior distributions. Thus the model is specified by a latent state that drives the dynamics of the observations. Figure 2.1 shows the graphical representation of the model in consideration.

We further can write down the complete-data likelihood as follows

$$L(y|\theta) = \prod_{t=1}^T p(y_t|y_{1:t-1}, \theta), \quad (2.14)$$

where the terms in the product can be computed recursively

$$p(y_t|y_{1:t-1}, \theta) = \int p(y_t|h_t, \theta)p(h_t|y_{1:t-1}, \theta)dh_t. \quad (2.15)$$

Note that the likelihood is a high-dimensional integral, and the analytical solution to this integral is not available. It is one of the issues in the estimation of these models we will be considering in this paper which different methods tackle differently.

2.2.2 Multivariate Case

The main objective of this paper is to give a review of existing methods in Bayesian statistics with application to stochastic volatility models and chose the direction for estimation and model selection in the multivariate and potentially high-dimensional framework. Most of the methods considered in this paper theoretically are extendable to the multivariate case. However, there are a few things that have to be considered when implementing these methods in practice. Firstly, it is of course the quality of estimation. When it comes to multivariate models approximative techniques often either fail to produce good approximations or lose their computational attractiveness.¹ Secondly, the difficulties in calibration of the used algorithms have to be considered. When it comes to multivariate models and exploration of high-dimensional parameter space calibration can be

¹For example, see the section on Integrated Nested Laplace Approximation. The method exploits sparseness of Gaussian random fields. However, in the multivariate case of the stochastic volatility model the sparseness is lost, and hence fast computational methods for sparse matrices can no longer be exploited.

Figure 2.1: **Graphical representation of stochastic volatility model**

Note: In the figure above h_t is latent volatility process and y_t is observable log-returns. At every time point we observe y_t that is dependent on the value of the latent process h_t at the same time instant.

come a very challenging and time-consuming task. Thirdly, it is important to see how adaptable these methods are to different specifications of the model. SVMs should be able to reflect different stylized facts of financial markets, depending on which kinds of financial data is in consideration the specification of the model can slightly vary (for instance, one might be interested in incorporation of the leverage effects into the model or heavy-tails and hence using t-Student distribution instead of standard assumption of Gaussian model). The ease of adaptation of these methods to a particular specification of the model is also of interest. Finally, one can consider combination of the exact and approximate techniques to develop feasible algorithms for estimation of this class of models in the multivariate case. We discuss these possibilities in the conclusion.

Let us further discuss the specification of the model of interest. A general stochastic volatility model of order p reads as follows

$$y_t = \Omega_t \epsilon_t, \quad (2.16)$$

$$h_t = \mu + \sum_{i=1}^p \Phi_i(h_{t-i} - \mu) + \eta_t, \quad (2.17)$$

where $\epsilon_t \stackrel{iid}{\sim} N(0, \Sigma_\epsilon)$ and $\eta \stackrel{iid}{\sim} N(0, \Sigma_\eta)$. With Equation (2.16) being the observation equation and Equation (2.17) being the state equation, i.e. $y_t = (y_{1t}, y_{2t}, \dots, y_{nt})'$ is vector of observable data, $h_t = (h_{1t}, h_{2t}, \dots, h_{nt})'$ is vector of log-volatilities, $\mu = (\mu_1, \mu_2, \dots, \mu_n)'$ is vector containing means of volatility processes, $\eta_t = (\eta_{1t}, \eta_{2t}, \dots, \eta_{nt})'$, Φ_p is matrix that contains vector autoregressive coefficients of volatility process. The matrix Ω in the observation equation is diagonal, $\Omega_t = \text{diag}(\exp(h_{1t}/2), \dots, \exp(h_{nt}/2))$.

Useful consequence of the specification of the model is that we can define volatility spillover effects on the financial markets through the concept of Granger-causality. Let us consider the case when $p = 1$. We say that there is no spillover effect between market i and j if there is non-causality established between volatility process i and j which translates into $\phi_{ij} = 0$.

For now we will keep in mind the model of order $p = 1$ as this case in high-dimensional scenario is already challenging. However, with an efficient estimation technique for the high-dimensional SVM one would be able to extend the methodology by, for example, incorporating sparse prior distribution for Φ . In particular, spike and slab priors (Ishwaran and Rao, 2005) or horseshoe prior are of interest (Carvalho et al., 2009). The former provides a tool for model selection in Bayesian framework where we explore not only the parameter space, but also the model space. In case one wishes to do model selection, this might be a plausible framework. In this specification, the spike regime corresponds to a coefficient to be zero and slab regime corresponds to a non-zero case (say, the parameter is then assumed to come from the normal distribution), by moving through these regimes and simultaneously in the parameter space, one can design an algorithm and a decision rule based on the posterior inclusion probability, to make a decision regarding whether there is non-causality between two processes (i.e., say whether $\phi_{ij} = 0$). The horseshoe prior provides a tool for modeling sparse relationships. In particular, when one is considering a high-dimensional model this might be of interest as we would rather expect a sparse dependence structure in the autoregressive matrix. We further proceed to the review of the methods that can be considered for the estimation of the model discussed in this section.

2.3 QML and Kalman Filter

Even though Quasi-Maximum Likelihood has been shown to be troublesome in the estimation of stochastic volatility models and it is not a method of our interest by itself, it is important for some other methodologies considered in this paper. It is also a standard technique that researchers are still applying in different variations of it to the problem at hand. In particular Salimans et al. (2013) use Kalman filter for estimation of the latent state in their Variational Bayes algorithm. The idea of applying QML and Kalman filter relies on linearization of the observation equation. Note, that once we linearize the model the observation equation does not have normal

distribution anymore. This is crucial for estimation and different solutions have been proposed to tackle the problem. Linearized observation equation reads

$$\log y_t^2 = h_t + \log \epsilon_t^2, \quad (2.18)$$

Note, that the error term is no longer standard normal, and hence it has to be approximated. It has been proposed to approximate the mean and the variance of the error term as $\mathbb{E}(\log \epsilon_t^2) = -1.2704$ and $\text{Var}(\log \epsilon_t^2) = 4.93$, see for more details Ruiz (1994). Another approach that appeared to be popular is to approximate the distribution of the error term with a mixture distribution, see Kim et al. (1998). Finally, the state equation for the stochastic volatility model reads

$$h_t = \phi h_{t-1} + \eta_t. \quad (2.19)$$

The volatility process is estimated by using recursively for $t = 1, \dots, T$ the Kalman filter and smoother (for more details see Durbin and Koopman, 2012; Kim et al., 1998). Then quasi-likelihood for parameters of interest θ is

$$\log L_Q(y \mid \theta) = -\frac{n}{2} \log 2\pi - \frac{1}{2} \sum_{t=1}^T \log F_t - \frac{1}{2} \sum_{t=1}^T \nu_t^2 / F_t, \quad (2.20)$$

where ν^2 is the one-step-ahead prediction error for the best linear estimator of $\log y_t^2$ and F_t mean square error. The quasi-log-likelihood is then maximized by using numerical optimization techniques.

2.4 Bayesian Inference

In this paper we are going to work in Bayesian paradigm for the estimation of nonlinear state-space models and the stochastic volatility models in particular. We will not go into discussion of advantages and disadvantages of Bayesian and frequentist methods in general. It is not our goal

to compare frequentist and Bayesian frameworks with the problem of our interest in mind. However, we believe it is important to note that some of the methods discussed in this paper can be also applied with classical statistical methodology in mind. In particular, Sequential Monte Carlo methods that allow to estimate the likelihood can be used in optimization algorithms to find frequentist estimates of the parameters of the model. In Bayesian framework, however, we will be mainly interested in obtaining the posterior distribution of the parameters. We do not go into details on historical perspective and development of Bayesian statistics. The famous Bayes' rule can be written in multiple ways, we however are interested in the version that formulates relation between posterior distribution of the *parameters* of a *model* given *the data*. The Bayes' rule in this case reads

$$p(\theta|y) = \frac{p(\theta)p(y|\theta)}{p(y)}, \quad (2.21)$$

where $p(\theta)$ is the prior, $p(y|\theta)$ is the likelihood and $p(y)$ can be viewed as normalizing constant. Note, that θ is the vector of parameters of interest, and y is the data. By removing the denominator we get Bayes' rule in terms of proportionality

$$p(\theta|y) \propto p(\theta)p(y|\theta). \quad (2.22)$$

Note, that Equation (2.22) summarizes the general idea about Bayesian paradigm. We initialize our prior beliefs about the parameters of the model, and then once we obtain more data, we update our beliefs. Further, recall that we have a latent process in our model. Since we never observe it, this is also an unknown we have to estimate. The Bayes' rule then becomes

$$p(\theta, h|y) \propto p(y|\theta, h)p(h|\theta)p(\theta), \quad (2.23)$$

The object of interest in Bayesian inference is $p(\theta, h|y)$ – the posterior distribution of the latent states and the parameters of the model given the data – which factorizes in our case according to the Bayes' rule as in Equa-

tion (2.23). Note, that this is not possible to do directly in our case, since neither the likelihood nor the posterior are available in closed form. Thus, there are two ways to proceed with this problem: either apply Markov Chain Monte Carlo methods for sampling from the posterior distribution which we discuss in the next section or use approximate techniques that we discuss further in this paper. Note that we will further discuss various methods that are applicable to our problem, but we will in particular look how in the issues of scalability, computational complexity, estimation accuracy and flexibility of the method considered.

2.5 SMC and MCMC

Sequential Monte Carlo (SMC) method, also known in the literature as *particle filtering*, is considered state of the art method for estimation of the intractable likelihood in nonlinear state-space models. The general idea behind this method lies in the estimation of the latent states by drawing multiple samples of samples (particles) and then propagating them in time according to corresponding weights. The weights are obtained using the likelihood of the data. By combining the weights over all time steps, one can obtain marginal likelihood. Standard and well known schemes are *Bootstrap particle filter* (BPF) Gordon et al. (1993), *Sequential Importance Sampling* (SIS) and *Sequential Importance Resampling* (SIR) Doucet et al. (2001). Sequential Monte Carlo methods were elegantly combined with Markov Chain Monte Carlo in Andrieu et al. (2010) and the method was named particle Markov Chain Monte Carlo (PMCMC). This method provides a powerful and coherent approach for Bayesian inference in a wide range of complex models. It also naturally fits the estimation problem of stochastic volatility models. The underlying idea is to estimate intractable likelihood by Sequential Monte Carlo methods and use that estimate of the likelihood in the MCMC algorithm. The main challenges of this approach are coming from both of the combined procedures. First, the variance of the estimated likelihood is of concern, especially when it comes to higher dimensions. Standard SMC techniques such as SIS are prone to have high

variance of the estimated likelihood once the dimensionality of the problem increases (Chopin et al., 2004). A number of studies have tried to address this problem. The common choice of proposal for sample of particles in standard schemes is $p(h_t|h_{t-1})$. Pitt and Shephard (1999) propose an *auxiliary particle filter* as a solution that is using proposal for particles which takes into account the current observation $p(h_t|h_{t-1}, y_t)$ and not only the dynamics of the latent process itself. Scharth and Kohn (2016) suggest to use efficient importance sampling (Richard and Zhang, 2007) inside PMCMC procedure. Guarniero et al. (2017) use twisted representation of the model and use the look-ahead type of particle filtering to address the issue of high variance of the estimated likelihood. Johansen and Doucet (2008) compare sequential importance resampling (SIR) with auxiliary particle filter and find that APF does not always outperform SIR. Often the variance of the estimated likelihood is analyzed in the true value of the parameters, such as in Scharth and Kohn (2016). However, when using particle Markov Chain Monte Carlo, it is also of interest whether the same conclusions hold in different points of the parameter space. In particular, we never start running the algorithm at the point of the true parameter values. This means, that if the variance of the estimated likelihood is much larger in some areas of the parameter space, the convergence of the algorithm can be affected. Having insights into how the variance of the estimated likelihood is different in the parameter space can help to make a more efficient choice of the starting point for the algorithm.

2.5.1 Introduction to Markov Chain Monte Carlo Methods

Markov Chain Monte Carlo (MCMC) is a popular statistical method and in Bayesian statistics it is often used for sampling from posterior distributions of the parameters. The idea first has appeared in the famous work by Metropolis et al. (1953) and the method proposed is known as Metropolis algorithm. Without going into details of the original work, we will say that the method was designed to simulate the dynamics of a physical system in its equilibrium point, and hence the idea was to simulate the system and

run the algorithm until it converges to the equilibrium (see Geyer, 2011). Then this method was generalized in Hastings (1970), the result is known as Metropolis-Hastings algorithm and it further became widespread in the statistical literature and its applications were found to be useful in many scientific fields. This is also the algorithm which will be a core of some of the methods we will be discussing in this paper. Further, a special case of this algorithm is known as Gibbs sampler which was introduced in Geman and Geman (1984). Even though Gibbs sampler is of less interest to us, there is still a method that builds upon it that we will discuss with application to stochastic volatility models Lindsten et al. (2014).

Since MCMC methods are quite important for the estimation of the models of our interest, we will introduce the idea behind them and then build up on this idea in the next sections. We further define a Markov chain as in Geyer (2011).

Definition 2.1. A sequence $X_1, X_2, \dots$ of random elements of some set is a *Markov chain* if the conditional distribution of X_{n+1} given $X_1, \dots, X_n$ depends on X_n only. The set in which the X_i take values is called the *state space* of the Markov chain.

Further, in the construction of MCMC methods we are in particular interested in Markov chains with *stationary transition probabilities*. This is satisfied, when $p(X_{n+1} | X_n)$, the conditional distribution, does not depend on n (Geyer, 2011). Thus, X_1 is initial distribution and $p(X_{n+1} | X_n)$ is the transition probability distribution.

We further also provide the reader with properties that Markov chains need to satisfy when one constructs MCMC methods. Understanding these properties and some of the terminology could be useful for understanding of how MCMC methods are designed and in which situations they might fail, but also one could find it helpful when discussing more advanced methods for sampling from the posterior distribution such as *Hamiltonian Monte Carlo*.

First, a Markov chain is said to be *stationary* if it is a stationary stochas-

tic process (Geyer, 2011).² If the stationarity of the Markov chain is satisfied, then one also gets the stationarity of the transition probabilities. Second, a transition probability distribution should be *reversible*. We give the definition as in Geyer (2011).

Definition 2.2. A transition probability is *reversible* with respect to an initial distribution if, for the Markov chain $X_1, X_2, \dots$ they specify, the distribution pairs (X_i, X_{i+1}) is exchangeable.

Definition 2.3. A Markov chain is reversible if its transition probability is reversible with respect to its initial distribution.

Reversibility plays important role in the construction of MCMC algorithms. It can be useful from both applied perspective where reversibility is exploiting to construct more efficient algorithms, as well as from theoretical perspective where the Central Limit Theorem (CLT) for the Markov Chains simplifies when reversibility is satisfied. A typical example of a Markov chain that is also useful for construction and understanding of the Metropolis-Hastings algorithm is *random walk* process. It is defined in the following way by Robert and Casella (2005).

Definition 2.4. A sequence of random variables (X_n) is a *random walk* if it satisfies

$$X_{n+1} = X_n + \epsilon_n,$$

where ϵ_n is generated independently of $X_n, X_{n-1}, \dots$. If the distribution of the ϵ_n is symmetric about zero, the sequence is called a *symmetric random walk*.

The theory behind Markov chains is quite extensive. Here we mainly provide basic properties and definitions to create the general set-up for the methods we are going to discuss. For more details we refer the reader to an excellent book Robert and Casella (2005) that discusses various Monte

²A stochastic process said to be (weakly) stationary if for every $k > 0$ the distribution of the $(X_{n+1}, \dots, X_{n+k})$ does not depend on n (Geyer, 2011).

Carlo and Markov Chain Monte Carlo methods and the following sources on introduction to MCMC methods van Ravenzwaaij et al. (2016), Geyer (2011). Further, the definition of Markov Chain Monte Carlo method is given in Robert and Casella (2005) as follows

Definition 2.5. A *Markov chain Monte Carlo method (MCMC)* for the simulation of a distribution $p(\cdot)$ is any method producing an ergodic Markov chain $(X^{(t)})$ whose stationary distribution is $p(\cdot)$.

The Metropolis-Hastings algorithm is one of the off-shelf MCMC methods in the statistical community. It is quite general and can be applied to various problems. To implement Metropolis-Hastings algorithm to sample from a target density $p(\cdot)$ multiple expressions need to be specified. We need to define a conditional density $q(y|x)$ that is a proposal distribution, generally $q(y|x)$ should be such that we can easily simulate from it. In many applications, including ours it is reasonable to take the Gaussian distribution as proposal distribution. In this case it is also symmetric, meaning $q(y|x) = q(x|y)$. The Metropolis-Hastings iteration is then represented in the Algorithm 1. In this algorithm $\rho(\theta, \theta^c)$ is the Metropolis-Hastings acceptance probability, where θ is the current state of the chain and θ^c is the candidate state of the parameter vector. Generally, in the simulations it is desired to have around 25% of proposed candidate values accepted (Roberts

Algorithm 1 Metropolis-Hastings Algorithm

- 1: Given $\theta^{(t)}$,
- 2: Generate $\theta_t^c \sim q(y \mid \theta^{(t)})$,
- 3: Take

$$\theta^{(t+1)} = \begin{cases} \theta_t^c, & \text{with probability } \rho(\theta^{(t)}, \theta_t^c) \\ \theta^{(t)} & \text{with probability } 1 - \rho(\theta^{(t)}, \theta_t^c), \end{cases}$$

where

$$\rho(\theta, \theta^c) = \min \left(\frac{f(\theta^c) q(\theta \mid \theta^c)}{f(\theta) q(\theta^c \mid \theta)}, 1 \right)$$

et al., 1997). The idea is that when the proposal steps are too large (we make a proposal that is far away from the current state, θ , in the Markov chain), we do not explore local regions sufficiently well, moreover many of the candidates are then very likely to be rejected. When the proposal steps are very small, the acceptance rate will be very high, however, then we are not likely to leave regions of the local maximum or the convergence will happen very slowly.

The performance of the Metropolis-Hastings algorithm to a big extent can depend on the choice of the proposal distribution. When $q(x, y)$ is chosen to be $q(y)$ it is called *independent* Metropolis-Hastings, i.e. the proposal is independent of its first argument Marin and Robert (2007).

The independent Metropolis-Hastings is usually not a good proposal (meaning that it is hard to achieve optimal acceptance rate) in high-dimensional problems when there is not much information available about the target distribution. In this case a better choice would be *random walk* Metropolis-Hastings. In random walk proposal the proposal for the next step depends on the current step, i.e. $\theta_{t+1}^c = \theta_t + \epsilon_t$.

Moreover, note that in the Algorithm 1 we can exploit the idea of random-walk proposal (see the definition of the random walk earlier in this section): the proposal for the next step depends on the current step, i.e. $\theta_{t+1}^c = \theta_t + \epsilon_t$. This is a kind of the proposal that we will be using most of the time in Metropolis-Hastings algorithm. For more on theoretical properties of this algorithm and convergence results we refer the reader to Robert and Casella (2005).

2.5.2 Sequential Monte Carlo

Assume that we are in the framework with an observed time series process y_t and a latent Markovian process h_t . Since we never observe the latent process, we need to infer it. The objective that can be achieved with *Sequential Monte Carlo* (SMC) also known as *particle filtering* is estimation of the states h_t with arriving observations y_t , in other words we are interested in sequential estimation of the posterior distribution. Bayes theorem

implies

$$p(h_{0:t}|y_{1:t}) = \frac{p(y_{1:t}|h_{0:t})p(h_{0:t})}{\int p(y_{1:t}|h_{0:t})p(h_{0:t})dh_{0:t}}. \quad (2.24)$$

The above equation involves computation of high-dimensional integrals, as well as the evaluation of the likelihood in Equation (2.25). Sequential Monte Carlo (or particle filters) is the method developed to tackle such kind of problems and estimate high-dimensional integrals.

$$\begin{aligned} L(y_{1:T}) &= \int p(y_{1:T}, h_{1:T})dh_{1:T} = \int p(y_{1:T}|h_{1:T})p(h_{1:T})dh_{1:T} \\ &= \int p(y_1|h_1)p(h_1) \prod_{t=2}^T p(y_t|h_t)p(h_t|h_{t-1})dh_1 \dots h_T. \end{aligned} \quad (2.25)$$

The basic procedure of particle filtering in this setting can be summarized by three crucial steps: prediction $p(h_t|y_{1:t-1})$, updating $p(h_t|y_{1:t})$ and re-sampling $p(y_t|h_t)$. The outline of a basic particle filter can be summarized in the following way

- Initialization: given the prior distribution $\pi(\theta_0)$ we draw N independent random samples $\{h_i^0\}_{i=1}^N$ from it, these samples we call *particles*.
- Prediction: for every particle we make a prediction for the latent process

$$p(h_t|y_{1:t-1}) = \int p(h_t|h_{t-1})p(h_{t-1}|y_{1:t-1})dh_{t-1} \quad (2.26)$$

- Updating: During updating we assign a weight to every particle according to the likelihood function of the observation process. Every weight can be interpreted as our "confidence" about this particle. The posterior distribution of the latent volatility state h_t can be computed using Bayes' rule.

$$p(h_t|y_{1:t}) = \frac{p(y_t|h_t)p(h_t|y_{1:t-1})}{\int p(y_t|h_t)p(h_t|y_{1:t-1})dh_t} \quad (2.27)$$

The well documented problem that rises in implementation of basic particle filters is the trade-off between particle *degeneracy* and particle *impoverishment*. The former happens when the resampling step is ignored or is not performed frequently enough. In this case one ends up with a particle set that has zero weights. The latter problem happens when the particle set is resampled too frequently, then eventually one gets one particle with a large weight and hence the particle set lacks the diversity. The way to find the balance between these two problems is resampling when the efficient number of particles is smaller than a certain threshold. This approach is sometimes referred to as adaptive resampling and is reflected in Equation (2.28). The simplest version of a particle filter is represented in Algorithm 2. The resampling step is described in the Algorithm 3.

The bootstrap filter is a variation of a more general approach - sequential importance sampling (resampling). The distinction of the bootstrap

Figure 2.2: **Illustration of a simple particle filter.**

Note: This figure illustrates basic steps of a standard particle filter. First, particles are initialised, usually according to the prior distribution. Then we do a prediction step for every particle $p(h_t|h_{t-1})$. During the update state every predicted particle is weighted according to the likelihood of the data, $p(y_t|h_t, \theta)$. Further, the particles are resampled according to the obtained weights. The particles with large weights are multiplied while particles with low weights are discarded.

Figure 2.3: Illustration of two problems of particle filtering: degeneracy (left) and sample impoverishment (right).

Note: This figure illustrates two basic problems that particle filter algorithms can exhibit. First (picture on the left) is called *particle degeneracy*. It happens when we do not perform resampling step frequently enough. In this case we end up with a particle set in which most particles have zero weight (which is not very informative and clearly does not help the inference). To diversify the particle set one needs to perform resampling frequently enough. However, if we resample too often we will face *particle impoverishment*. Which means that we likely will end up with a single particle that has biggest weight multiplied many times. Clearly, such a particle set is not very representative either.

filter is the proposal from the dynamic model $h_t^{(i)} \sim p(h_t|h_{t-1}^{(i)})$, while in the importance sampling it is an importance density $q(h_t|h_{t-1}^{(i)}, |y_{1:t})$. Equation (2.28) gives a formula for calculation of the effective number of particles. Adaptive resampling is performed when the estimated effective number of particles is below prespecified threshold.

$$N_{eff} = \frac{1}{\sum_{i=1}^N (\omega_t^{(i)})^2}, \quad (2.28)$$

with $\omega_t^{(i)}$ normalized weight of particle i at the time step k . Resampling is then performed when the effective number of particles is less than the total number, for example $N_{eff} < N/10$.

Bootstrap filter (which is a version of Sequential Importance Resampling scheme) will not perform well in the situations where the data exhibits peaky observations. Particular disadvantage of this algorithm is that we

Algorithm 2 Bootstrap Particle Filter

- 1: Draw a new point $h_t^{(i)}$ for each point in the sample set $\{h_{t-1}^{(i)} : i = 1, \dots, N\}$ from the dynamic model:

$$h_t^{(i)} \sim p(h_t|h_{t-1}^{(i)}), \quad i = 1, \dots, N.$$

- 2: Calculate the weights

$$\omega_t^{(i)} \sim p(y_t|h_t^{(i)}), \quad i = 1, \dots, N,$$

and normalize them to sum to unity.

- 3: Do resampling.
-

Algorithm 3 Resampling

- 1: Interpret each weight $\omega_t^{(i)}$ as the probability of obtaining the sample index i in the set $\{h_t^{(i)} : i = 1, \dots, N\}$.
 - 2: Draw N samples from that discrete distribution and replace the old samples set with this new one.
 - 3: Set all weights to the constant values $\omega_t^{(i)} = 1/N$.
-

make proposals $p(h_t|h_{t-1})$ ignoring the fact that we know the data y . As a solution to this problem the auxiliary particle filter has been proposed which we consider in the next subsection.

2.5.3 Approximate of marginal likelihood with Auxiliary Sequential Importance Resampling (ASIR)

As we mentioned before, the marginal likelihood can be approximated sequentially through particle filtering. The marginal likelihood approximation of the parameters θ reads

$$p(y_{1:T}|\theta) \approx \prod_t \hat{p}(y_t|y_{1:t-1}, \theta). \quad (2.29)$$

In practice, usually the log-likelihood

$$\log p_\theta(y_{1:T}) = \log p_\theta(y_1) + \sum_{t=2}^T \log p_\theta(y_t|y_{1:t-1}) \quad (2.30)$$

is estimated for the purpose of numerical stability (as the product of small weights would lead to unstable results). The estimate of the log-likelihood is the by-product of the particle filtering, as it is the average over log-weights that are assigned to the particles at every time step. The main distinction of auxiliary particle filter presented in the Algorithm 4 (Särkkä, 2013) from the bootstrap filter is visible in the equations (2.32) and (2.33). If in the bootstrap particle filter the proposals for the particles are made on the basis of the dynamics of the model $p(h_t|h_{t-1})$, in the auxiliary particle filter one also takes into account the current observation y_t . This allows to make better proposals for the particles, and hence reduces the variance of the estimated likelihood which grows with the dimensionality of the model. However, usually one cannot get an analytical expression for the proposal density (unless we work with the linear Gaussian state-space model in which case the auxiliary particle filter is referred to as fully adapted particle filter). Thereby, in a nonlinear non-Gaussian state-space models the term $p(h_t|h_{t-1}, y_t)$ has to be approximated. In the next section we

will review how the proposal can be made in the auxiliary particle filter proposed by Pitt and Shephard (1999) in the case of stochastic volatility models.

2.5.4 Particle Metropolis-Hastings

Particle Markov Chain Monte Carlo (PMCMC) methods were introduced in Andrieu et al. (2010). The basic idea is that MCMC methods, and in particular, Metropolis-Hastings algorithm which is of interest to us can be combined with Sequential Monte Carlo to make draws from the posterior distributions of the parameters. Algorithm 5 presents particle Metropolis-

Algorithm 4 Approximation of marginal likelihood with ASIR algorithm

- 1: Draw N samples $h_0^{(i)}$ from the prior

$$h_0^{(i)} \sim p(h_0 \mid \theta), \quad i = 1, \dots, N \quad (2.31)$$

and set $w_0^{(i)} = 1/N$, for all $i = 1, \dots, N$.

- 2: For each $t = 1, \dots, T$ do the following
- 3: Draw samples $h_t^{(i)}$ from the importance distributions

$$h_t^{(i)} \sim \pi(h_t \mid h_{t-1}^{(i)}, y_{1:t}, \theta), \quad i = 1, \dots, N. \quad (2.32)$$

- 4: Compute the following weights

$$v_t^{(i)} = \frac{p(y_t \mid h_t^{(i)}, \theta) p(h_t^{(i)} \mid h_{t-1}^{(i)}, \theta)}{q(h_t^{(i)} \mid h_{t-1}^{(i)}, y_{1:t}, \theta)} \quad (2.33)$$

and compute the estimate of $p(y_t \mid y_{1:t-1}, \theta)$ as

$$\hat{p}(y_t \mid y_{1:t-1}, \theta) = \sum_i w_{t-1}^{(i)} v_t^{(i)}. \quad (2.34)$$

- 5: Compute normalized weights as

$$w_t^{(i)} \propto w_{t-1}^{(i)} v_t^{(i)}. \quad (2.35)$$

- 6: If the effective number of particles is too low, perform resampling.
-

Hastings algorithm. The difference with the standard Metropolis-Hastings is in the quantity $\hat{p}_{\theta^*}(y_{1:T})$ which is the estimate of the likelihood obtained with a particle filter conditioning on the parameters vector θ . In this algorithm $q(\theta(i-1)|\theta^*)$ is the proposal distribution (which cancels out when it is symmetric), $\pi(\cdot)$ is prior distribution.

2.5.5 Auxiliary particle filter for SV model

Incorporating knowledge of y_t into proposals for particles $p(h_t|h_{t-1}, y_t)$ helps to reduce the variance of the estimated likelihood with particles filters and improves the approximation of the filtering distribution $p(h_t|y_{1:t})$. In particular, we observe a drastic improvement when the variance of the latent volatility process is small. This is illustrated in Figures 2.4, 2.5, 2.6. As we mentioned, only in the case of linear Gaussian state-space models the proposal density from Equation (2.32) has an analytical expression. Hence, for the stochastic volatility models this term must be approximated. Pitt and Shephard (1999) propose to use non-blind proposals for the next generation of particles by first expanding $\log p(y_{t+1}|h_{t+1})$ to a second-order term

Algorithm 5 Particle Metropolis-Hastings

- 1: Initialize algorithm, $i = 0$
- 2: Initialize the parameters at $\theta(0)$
- 3: run an SMC algorithm targeting $p_{\theta(0)}(x_{1:T} | y_{1:T})$, sample $X_{1:T}(0) \sim \hat{p}_{\theta(0)}(\cdot | y_{1:T})$ and let $\hat{p}(y_{1:T})$ denote the marginal likelihood estimate, and
- 4: with probability

$$1 \wedge \frac{\hat{p}_{\theta^*}(y_{1:T})}{\hat{p}_{\theta^{(i-1)}}(y_{1:T})} \frac{\pi(\theta^*)}{\pi(\theta^{(i-1)})} \frac{q(\theta^{(i-1)}|\theta^*)}{q(\theta^*|\theta^{(i-1)})} \quad (2.36)$$

set $\theta^{(i)} = \theta^*$, $X_{1:T}^{(i)} = X_{1:T}^*$ and $\hat{p}_{\theta^{(i)}}(y_{1:T}) = \hat{p}_{\theta^*}(y_{1:T})$; otherwise set $\theta^{(i)} = \theta^{(i-1)}$, $X_{1:T}^{(i)} = X_{1:T}^{(i-1)}$ and $\hat{p}_{\theta^{(i)}}(y_{1:T}) = \hat{p}_{\theta^{(i-1)}}(y_{1:T})$.

around μ_{t+1}^k via Taylor expansion

$$\begin{aligned} \log g(y_{t+1}|h_{t+1}, \mu_{t+1}^k) &= \log p(y_{t+1}|\mu_{t+1}^k)' \times \frac{\partial \log p(y_{t+1}|\mu_{t+1}^k)}{\partial h_{t+1}} + \\ &+ \frac{1}{2} \times (h_{t+1} - \mu_{t+1}^k)' \times \frac{\partial^2 \log p(y_{t+1}|\mu_{t+1}^k)}{\partial h_{t+1} h_{t+1}'} \times (h_{t+1} - \mu_{t+1}^k) \end{aligned} \quad (2.37)$$

For deriving the expression for $\log g(y_{t+1}|h_{t+1}, \mu_{t+1}^k)$ recall that $y_t \sim N(0, \exp(h_t))$ and hence

$$\begin{aligned} p(y_t|h_t) &= \frac{1}{\sqrt{2\pi \exp(h_t)}} \exp \left\{ -\frac{y_t^2}{2 \exp(h_t)} \right\} = \\ &= \frac{1}{\sqrt{2\pi}} \exp \left\{ -\frac{y_t^2}{\exp(-h_t)} - \frac{h_t}{2} \right\} \end{aligned} \quad (2.38)$$

and further note that $p(h_t|h_{t-1}) = N(\mu + \phi(h_{t-1} - \mu), \sigma_\eta^2)$, thus

$$p(h_t|h_{t-1}) = \frac{1}{\sqrt{2\pi\sigma_\eta^2}} \exp \left\{ -\frac{(h_t - \mu - \phi(h_{t-1} - \mu))^2}{2\sigma_\eta^2} \right\}. \quad (2.39)$$

It follows that the proposal for particles at time $t + 1$ when taking into account the observation of the same period is

$$\begin{aligned} &g(h_{t+1} | h_t^{(k)}, y_{t+1}; \mu_{t+1}^{(k)}) = \\ &N \left(\mu_{t+1}^{(k)} + \frac{\sigma^2}{2} \left\{ \frac{y_t^2}{\beta^2} \exp(-\mu_{t+1}^{(k)}) - 1 \right\}, \sigma^2 \right). \end{aligned} \quad (2.40)$$

Improvements in the filtering distribution of the latent process extend to the multivariate case as well. It is, however, important to note that these improvements are problem specific (Johansen and Doucet, 2008). It would be of interest to conduct a simulation study for multivariate stochastic volatility models and see how the improvements regarding the reduction of the variance of the estimated likelihood work over the parameter space. By that, we mean that usually particle filters are compared in the point of the ground truth that is known to a researcher in the simulated data, however, when one uses particle Markov Chain Monte Carlo for the poste-

Figure 2.4: **Estimate of the log-volatility with blind and non-blind proposals.**

Note: This figure shows estimates of the latent log-volatility process with blind (left) and non-blind (right) proposals for the particles. The solid black line indicates the volatility process generated with SV model and the red solid line indicates the estimate of the volatility process with a particle filter. The number of particles used for the estimation is $N=100$ and the variance of the latent log-volatility is $\sigma_\eta^2 = 0.5$.

Figure 2.5: **Estimate of the log-volatility with blind and non-blind proposals.**

Note: This figure shows estimates of the latent log-volatility process with blind (left) and non-blind (right) proposals for the particles. The solid black line indicates the volatility process generated with SV model and the red solid line indicates the estimate of the volatility process with a particle filter. The number of particles used for the estimation is $N=100$ and the variance of the latent log-volatility is $\sigma_\eta^2 = 0.1$.

rrior inference, it is very unlikely that one starts at the true values. Hence, comparing the variance in different points of the parameter space would allow getting more insights into the improvements that auxiliary (or any other particle filter) can bring and into the speed of the convergence.

Further, Figures 2.7, 2.8, 2.9, 2.10, 2.11 show results of particle Markov Chain Monte Carlo for time series generated with the stochastic volatility model. We used auxiliary particle filter for the estimation of the marginal likelihood.

Figure 2.6: Estimate of the log-volatility with blind and non-blind proposals.

Note: This figure shows estimates of the latent log-volatility process with blind (left) and non-blind (right) proposals for the particles. The solid black line indicates the volatility process generated with SV model and the red solid line indicates the estimate of the volatility process with a particle filter. The number of particles used for the estimation is $N=100$ and the variance of the latent log-volatility is $\sigma_\eta^2 = 0.05$.

Table 2.1: **Posterior results for estimation of the SV model with PMCMC.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	95% <i>HPD_l</i>	95% <i>HPD_u</i>	<i>True</i>
μ	1.1985	1.2028	1.2140	1.0771	1.3034	1.189
ϕ	0.9713	0.9747	0.9732	0.9392	0.9987	0.959
σ	0.0537	0.0524	0.0514	0.0260	0.0828	0.089

Note: This table shows posterior results of the parameters as well as the ground truth. We observe that mostly the values are captured by the highest posterior density intervals. The true variance, however, is slightly higher than the upper bound indicates.

Figure 2.7: **Estimation of SV model with particle Markov Chain Monte methods: trace plots, histograms and autocorrelation-function.**

Note: This figure shows trace plots, histograms and autocorrelation functions. Trace plots and autocorrelation function indicate good mixing of the Markov chains.

Table 2.2: **Posterior results for estimation of the SV model with PMCMC.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	<i>95% HPD_l</i>	<i>95% HPD_u</i>	<i>True</i>
μ	0.8629	0.8647	0.8780	0.7870	0.9385	0.903
ϕ	0.9644	0.9732	0.9843	0.9077	0.9987	0.967
σ	0.0534	0.0500	0.0504	0.0233	0.0912	0.076

Note: This table shows posterior results of the parameters as well as the ground truth. We observe that all parameters are captured well by the highest posterior density intervals.

Figure 2.8: **Estimation of SV model with particle Markov Chain Monte methods: trace plots, histograms and autocorrelation-function.**

Note: This figure shows trace plots, histograms and autocorrelation functions. Trace plots and autocorrelation function indicate good mixing of the Markov chains.

Table 2.3: **Posterior results for estimation of the SV model with PMCMC.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	<i>95% HPD_l</i>	<i>95% HPD_u</i>	<i>True</i>
μ	0.9579	0.9575	0.9623	0.9090	1.0043	0.938
ϕ	0.9079	0.9199	0.9583	0.7974	0.9885	0.764
σ	0.0429	0.0405	0.0317	0.0184	0.0725	0.08

Note: This table shows posterior results of the parameters as well as the ground truth. We observe that the scale (μ) and persistence (ϕ) parameters are captured quite well. The variance, however, is underestimated.

Figure 2.9: **Estimation of SV model with particle Markov Chain Monte methods: trace plots, histograms and autocorrelation-function.**

Note: This figure shows trace plots, histograms and autocorrelation functions. Trace plots and autocorrelation function indicate good mixing of the Markov chains.

Table 2.4: **Posterior results for estimation of the SV model with PMCMC.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	<i>95% HPD_l</i>	<i>95% HPD_u</i>	<i>True</i>
μ	0.8865	0.8849	0.8932	0.7297	1.0485	0.942
ϕ	0.9826	0.9834	0.9892	0.9672	0.9961	0.981
σ	0.1100	0.1085	0.1011	0.0702	0.1503	0.1

Note: This table shows posterior results of the parameters as well as the ground truth. We observe that all parameters are captured well by the highest posterior density intervals.

Figure 2.10: **Estimation of SV model with particle Markov Chain Monte methods: trace plots, histograms and autocorrelation-function.**

Note: This figure shows trace plots, histograms and autocorrelation functions. Trace plots and autocorrelation function indicate good mixing of the Markov chains.

Table 2.5: **Posterior results for estimation of the SV model with PMCMC.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	<i>95% HPD_l</i>	<i>95% HPD_u</i>	<i>True</i>
μ	0.5836	0.5828	0.5893	0.5359	0.6293	0.553
ϕ	0.9497	0.9565	0.9970	0.8890	0.9970	0.840
σ	0.0756	0.0712	0.0525	0.0294	0.1331	0.215

Note: This table shows posterior results of the parameters as well as the ground truth. We observe that the scale (μ) and persistence (ϕ) parameters are captured quite well. The variance, however, is underestimated.

Figure 2.11: **Estimation of SV model with particle Markov Chain Monte methods: trace plots, histograms and autocorrelation-function.**

Note: This figure shows trace plots, histograms and autocorrelation functions. Trace plots and autocorrelation function indicate good mixing of the Markov chains.

2.5.6 Variance of the estimated likelihood

Here we present the variance of the estimated likelihood for the bootstrap and auxiliary particle filters. The variance is estimated as follows

$$Bias = 5000^{-1} \sum_{i=1}^{50} \sum_{j=1}^{100} (\log \hat{L}^j - \log \bar{L}(y^i)), \quad (2.41)$$

$$Variance = 5000^{-1} \sum_{i=1}^{50} \sum_{j=1}^{100} (\log \hat{L}^j - \log \bar{L}(y^i))^2, \quad (2.42)$$

where y_i is the i -th time series, and $\log L$ is the "true" log-likelihood value. We use $N=100$, $N=1000$ and $N=10000$ number of particles for this study. In the appendix we report the variance for individual time series. The next step would be to analyze the variance in different parameter values while in this Monte Carlo study we stick to the true values following the approach in Scharth and Kohn (2016). Johansen and Doucet (2008) showed that the asymptotic variance is not always smaller for the APF in comparison to the BPF. It is also clear from the results in the paper that the variance is different with respect to the different points in the parameter space. Since we are interested in using the estimated likelihood in the Markov Chain Monte Carlo setting, it is relevant how the variance behaves over the parameter space, and this would be the next step to be undertaken in the search for the efficient algorithm for estimation of the multivariate stochastic volatility models.

Table 2.6: Variance of the estimated likelihood with bootstrap particle filter (BPF) and auxiliary particle filter (APF) averaged over 50 time series.

	BPF	APF
$N=100$	2534.944	29.4490437
$N=1000$	2631.785	1.3051421
$N=10000$	2639.952	0.1352639

Note: This table shows variance of the estimated likelihood for different values of number of particles N and different particle filters: bootstrap and auxiliary particle filter. The results are averaged over 50 time series. For more details regarding synthetic data see the Appendix and the data set named Data Set 1. From this table it is clear that the increased number of particles can improve the variance of the estimated likelihood with auxiliary particle filter significantly while for bootstrap particle filter that is not the case.

Table 2.7: **Variance of the estimated likelihood for different data generating processes.**

Bootstrap particle filter				Auxiliary particle filter			
TS	$N=100$	$N=1000$	$N=10000$	TS	$N=100$	$N=1000$	$N=10000$
1	511.26	556.01	559.85	1	2.436	0.251	0.017
2	128.50	138.05	139.19	2	0.688	0.040	0.005
3	18143.	18883.	18945.	3	259.3	9.669	0.993
4	79.085	84.045	83.510	4	0.517	0.045	0.005
5	51.753	52.112	52.078	5	0.066	0.005	0.001
6	4.2531	4.2704	4.2961	6	0.026	0.003	0.000
7	2684.2	2780.2	2790.5	7	5.076	0.339	0.032
8	137.22	140.65	140.48	8	0.210	0.018	0.002
9	8.5151	13.541	13.419	9	0.808	0.067	0.008
10	3601.2	3665.6	3670.4	10	25.32	2.614	0.291

Note: This table shows the variance of the estimated likelihood with bootstrap and auxiliary particle filters for 10 different time series generated from the stochastic volatility model with different parameters. The data generating processes are described in the Appendix under the name Data set 2. Monte Carlo studies (see Table 2.6) average over different time series. Thus, single effects are not transparent. This table in particular shows that depending on the data variance of the estimated likelihood is not always smaller in the case of the auxiliary particle filter. This is in particular the case when one is using the number of particles that is too small. It is clear from the table that with increased number of particles performance of the auxiliary particle filter improves. This also is visible from Table 2.6.

Figure 2.12: Variance of the estimated likelihood in different points of the parameter space.

Note: This figure shows variance of the estimated likelihood in different points of the parameter space. The red dot corresponds to the value of the parameter vector with which the data were generated. The left (a) picture corresponds to the bootstrap particle filter and the right (b) to the auxiliary particle filter. We observe that for the both filters the variance of the estimated likelihood is larger in the areas where the variance of the latent process σ^2 is larger than the ground truth. The same holds for the scale parameter β .

2 ADVANCES IN BAYESIAN COMPUTATION WITH APPLICATION TO STOCHASTIC VOLATILITY MODELS

Figure 2.13: Variance of the estimated likelihood in different points of the parameter space.

Note: This figure shows variance of the estimated likelihood in different points of the parameter space. The red dot corresponds to the value of the parameter vector with which the data were generated. The left (a) picture corresponds to the bootstrap particle filter and the right (b) to the auxiliary particle filter. We observe similar pictures as before, for both filters the variance of the estimated likelihood is larger in the areas where the variance of the latent process σ^2 is larger than the ground truth. The same holds for the scale parameter β . However, in this case this difference seems to be more pronounced for the bootstrap particle filter.

Figure 2.14: Variance of the estimated likelihood in different points of the parameter space.

Note: This figure shows variance of the estimated likelihood in different points of the parameter space. The red dot corresponds to the value of the parameter vector with which the data were generated. The left (a) picture corresponds to the bootstrap particle filter and the right (b) to the auxiliary particle filter. We observe on this figure that once the variance of the latent process is further away from the ground truth, the variance of the estimated likelihood tends to increase.

Figure 2.15: Variance of the estimated likelihood in different points of the parameter space.

Note: This figure shows variance of the estimated likelihood in different points of the parameter space. The red dot corresponds to the value of the parameter vector with which the data were generated. The left (a) picture corresponds to the bootstrap particle filter and the right (b) to the auxiliary particle filter. In this example, the picture seems to be similar to the previous cases, for the estimated likelihood for both filters is sensitive to the variance of the latent process.

Figure 2.16: **Variance of the estimated likelihood in different points of the parameter space.**

Note: This figure shows variance of the estimated likelihood in different points of the parameter space. The red dot corresponds to the value of the parameter vector with which the data were generated. The left (a) picture corresponds to the bootstrap particle filter and the right (b) to the auxiliary particle filter. As in some of the previous cases, we see that the variance of the estimates likelihood is sensitive to the variance of the latent process σ^2 and the scale parameter β .

2.5.7 Iterated Auxiliary Particle Filter

Guarniero et al. (2017) propose a new approach in particle filtering that is called iterated auxiliary particle filter and belongs to the class of so called "look-ahead" methods. It is an offline approach as opposed to online methods that are processing data in real-time. The core idea of the method is defining *twisted* models and using this representation to obtain an unbiased estimate of the likelihood $L(y_{1:T})$. The twisted model is defined using ψ positive functions, a sequence ψ^* can be identified as optimal in a sense that the likelihood estimate is unbiased and strongly consistent. As before, let us denote the latent log-volatility process by $h_{1:t}$ and the observed log-returns by $y_{1:t}$. Using the notation as in Guarniero et al. (2017), which is useful for specifying the twisted representation later on, the model reads

$$\begin{aligned} \mu(\cdot) &= N\left(\cdot, 0, \frac{\sigma^2}{(1-\phi)^2}\right), & f(h, \cdot) &= N(\cdot; \phi h, \sigma^2), \\ g(h, \cdot) &= N(\cdot; \beta^2 \exp(h)), \end{aligned} \tag{2.43}$$

with $\phi \in (0, 1)$, $\beta > 0$ and $\sigma^2 > 0$. Further, we need to define twisted representation of the model. Assume that $\psi = (\psi_1, \psi_2, \dots, \psi_T)$ is a sequence of real-valued, bounded, continuous and positive functions. Let $f(h, \psi) = \int_H f(h, h')\psi(h')dh'$, and define a sequence of normalizing functions $(\tilde{\psi}_1, \tilde{\psi}_2, \dots, \tilde{\psi}_T)$ on $H = R^d$ in the following way $\tilde{\psi}_t(h_t) = f(h_t, \psi_{t+1})$ for $t \in \{1, \dots, T-1\}$. Further, let $\tilde{\psi}_T \equiv 1$, and define normalizing constant as $\tilde{\psi}_0 = \int_H \mu(h_1)\psi_1(h_1)dh_1$. Then the twisted model is defined as follows

$$\mu_1^\psi(h_1) = \frac{\mu(h_1)\psi_1(h_1)}{\tilde{\psi}_0}, \quad f_t^\psi(h_{t-1}, h_t) = \frac{f(h_{t-1}, h_t)\psi_t(h_t)}{\tilde{\psi}_t(h_{t-1})}, \quad (2.44)$$

with the sequence of positive functions

$$g_1^\psi(h_1) = g(h_1, y_1) \frac{\tilde{\psi}_1(h_1)}{\psi_1(h_1)} \tilde{\psi}_0, \quad g_t^\psi(h_t) = g(h_t, y_t) \frac{\tilde{\psi}_t(h_t)}{\psi_t(h_t)}. \quad (2.45)$$

Assume that $\xi_t^1, \dots, \xi_t^N$ are the particles for time period t . Then the ψ -APF algorithm is summarized in Algorithm 6. Note, if we chose $\mu_1 = \mu^\psi$,

Algorithm 6 ψ -Auxiliary Particle Filter

- 1: Sample $\xi_1^i \sim \mu^\psi$ independently for $i \in \{1, \dots, N\}$.
- 2: For $t = 2, \dots, T$, sample independently

$$\xi_t^i \sim \frac{\sum_{j=1}^N g_{t-1}^\psi(\xi_{t-1}^j) f_t^\psi(\xi_{t-1}^j, \cdot)}{\sum_{j=1}^N g_{t-1}^\psi(\xi_{t-1}^j)}, \quad i \in \{1, \dots, N\}.$$

$f_t = f_t^\psi$, $g_t = g_t^\psi$ Algorithm 6 turns out to be the bootstrap particle filter (BPF). In accordance with Guarniero et al. (2017) let us denote by Z_ψ^N approximation of L (marginal likelihood) obtained by ψ -APF. In practice optimal functions ψ^* are approximated recursively using Algorithm 7 where $(\xi_1^{1:N}, \dots, \xi_T^{1:N})$ are particles obtained by running particle filter and assuming $\psi_{T+1} \equiv 1$. Algorithm 7 is written in a general form and can be implemented in different ways depending on the computational complexity of the model. An inexpensive way to implement it is to solve parametrically minimization problem in Equation (2.46) and then set ψ_t according

Algorithm 7 Recursive function approximations

```

1: for  $t=T, \dots, 1$  do
2:   Set  $\psi_t^i \leftarrow g(\xi_t^i, y_t)f(\xi_t^i, \psi_{t+1})$  for  $i \in \{1, \dots, N\}$ .
3:   Choose  $\psi_t$  as a member of  $\Psi$  on the basis of  $\xi_t^{1:N}$  and  $\psi_t^{1:N}$ .
4: end for

```

to Equation (2.47). For each $t \in \{1, \dots, T\}$ compute numerically

$$(m_t^*, \Sigma_t^*, \lambda_t^*) = \arg \min_{m, \Sigma, \lambda} \sum_{i=1}^N [N(\xi_t^i; m, \Sigma) - \lambda \psi_{t+1}^i]^2, \quad (2.46)$$

and set

$$\psi_t(x_t) = N(x_t; m_t^*, \Sigma_t^*) + c(N, m_t^*, \Sigma_t^*), \quad (2.47)$$

with $c(\cdot)$ a positive real-valued function that warrants $f_t^\psi(x, \cdot)$ being a mixture of densities with some non-zero weight associated with the mixture component $f(x, \cdot)$. Guarniero et al. (2017) choose $k = 3$ as the stopping rule in stochastic volatility application. $\tau = 0.5$ is chosen for univariate model and $\tau = 1$ is chosen for multivariate model to improve the speed. Σ is assumed to be diagonal in the multivariate case. Finally, the iterated auxiliary particle filter is defined as follows (note, that standard resampling step that is often performed together with BPF or APF can be introduced to ψ -APF in a similar manner).

Even though iterated auxiliary particle filter provides good results in univariate case and in multivariate case with diagonal matrix Φ (i.e. it is assumed there is no Granger-causal feedback) which is shown in Guarniero et al. (2017), multivariate implementation is quite expensive. Note, that when Φ is assumed to be non-diagonal the parameter space of the model increases significantly.

We further give more details on the implementation of the optimization. Note that we can take the derivative of the function in Equation (2.46) with

respect to λ

$$\begin{aligned} \frac{\partial}{\partial \lambda} = \sum_{i=1}^N 2 \left\{ \prod_{j=1}^k \frac{1}{\sqrt{2\pi\sigma_j^2}} \exp \left(-\frac{(\xi_i^{ij} - \mu)^2}{2\sigma_j^2} \right) - \lambda \psi_{t+1}^i \right\} \\ \times (-\psi_{t+1}^i) = 0 \end{aligned} \quad (2.48)$$

and by putting this derivative to zero we get an expression for λ

$$\lambda = \frac{\sum_{i=1}^N \left\{ \prod_{j=1}^k \frac{1}{\sqrt{2\pi\sigma_j^2}} \exp \left(-\frac{(\xi_i^{ij} - \mu)^2}{2\sigma_j^2} \right) \right\}}{\sum_{i=1}^N \psi_{t+1}^i}. \quad (2.49)$$

Note, that this optimization problem has to be solved sequentially for every $t = T, \dots, 1$. The optimization problem is solved numerically and even if we do not restrict ourselves to convergence in the optimization, the solution is likely to come in high computational costs. Moreover, since we are interested in inferring the dependence structure in the latent process, the assumption that Σ matrix in the optimization problem is diagonal is too simplistic. It is more sensible to have non-diagonal structure in Σ in our application which would preserve dependence between particles. Relaxing this assumption, however, would only increase computational complexity of already expensive problem. Thus, practical implementation of this algorithm in multivariate case remain limited. One of the possible direction to undertake is to use approximation methods for the optimization problem.

Algorithm 8 An iterated auxiliary particle filter with parameters (N_0, k, τ)

- 1: Initialize: set ψ^0 to be a sequence of constant functions, $l \leftarrow 0$.
 - 2: Repeat
 - 3: Run a ψ^l -APF with N_l particles, and set $\hat{Z}_l \leftarrow Z_{\psi^l}^{N_l}$.
 - 4: If $l > k$ and $\text{sd}\hat{Z}_{l-k:l} < \tau$, go to
 - 5: Compute ψ^{l+1} using a version of Algorithm with the particles produced.
 - 6: If $N_{l-k} = N_l$ and the sequence $\hat{Z}_{l-k:l}$ is not monotonically increasing, set $N_{l+1} \leftarrow 2N_l$. Otherwise, set $N_{l+1} \leftarrow N_l$.
 - 7: Set $l \leftarrow l + 1$ and go back to
 - 8: Run a ψ^l -APF and return $\hat{Z} = Z_{\phi}^{N_l}$.
-

2.5.8 Particle Gibbs Sampler

So far we have been discussing how SMC and Metropolis-Hastings algorithms can be combined for the estimation of the stochastic volatility models. As we mentioned the combination of SMC and MCMC is referred in the literature as particle Markov Chain Monte Carlo (PMCMC). Gibbs sampler is another MCMC method that can be combined with SMC in the case when estimation of the latent states is challenging. We first briefly introduce the idea of the Gibbs sampler and then we proceed with the approach that allows to combine Gibbs sampling with particle filtering.

The original Gibbs sampling method goes back to Geman and Geman (1984) and is motivated by the fact that if the following equations hold (see for more details Marin and Robert, 2007).

$$p(x_1) = \int p_1(x_1|x_2)\tilde{p}(x_2)dx_2, \quad (2.50)$$

we can jointly simulate (x_1, x_2) from the distribution

$$p_1(x_1|x_2)\tilde{p}(x_2). \quad (2.51)$$

Frequently it happens that $\tilde{p}(x_2)$ is intractable and it is not possible to simulate from it. But when both $p_1(x_1|x_2)$ and $p_2(x_2|x_1)$ can be simulated from, one also can simulate from the joint density $p_1(x_1|x_2)\tilde{p}(x_2)$ (Marin

and Robert, 2007). The idea for this kind of sampling is summarized in the Algorithm 9.

Gibbs samplers for stochastic volatility models have appeared in the literature many times, in particular we refer the reader to Kim et al. (1998). Note, that Gibbs sampler for parameter inference in the state-space models iterated between two steps: sampling from the conditional distribution of the parameters given the data and the latent process $p(\theta|y_{1:T}, h_{1:T})$ and sampling the latent process $p(h_{0:T}|\theta, y_{1:T})$. It is commonly recognized that the latter step is challenging. The idea introduced in Lindsten et al. (2014) is based on combining particle filtering and Gibbs sampling with what the authors call *the ancestor sampling* procedure. This elegant way of sampling allows to estimate PGAS kernel with only a few number of particles while still producing a fast mixing of the kernel.

Specification of the priors and conditionals

Implementation of particle Gibbs sampler with ancestor resampling requires specification of the prior and conditional distributions. The following priors for the parameters of stochastic volatility model would be a standard choice

$$\phi \sim N(a, b^2)\mathbb{I}_{(-1,1)}(\phi) \quad \text{or} \quad \phi^* \sim \text{Beta}(\phi^{(1)}, \phi^{(2)}), \quad (2.52)$$

where $\mathbb{I}_{(-1,1)}$ is the indicator function on the interval $(-1, 1)$ and $\phi = 2\phi^* - 1$,

$$\mu \sim N(c, d^2), \quad \sigma_\eta^2 \sim IG(\alpha, \beta), \quad h_0 \sim N(0, \sigma_\eta^2). \quad (2.53)$$

Algorithm 9 Two-stage Gibbs sampler

- 1: Initialization: Start with an arbitrary value $x_2^{(0)}$
 - 2: For $t = 1, 2, \dots$: given $x_2^{(t-1)}$, generate
 - 3: $x_1^{(t)}$ according to $p_1(x_1|x_2^{(t-1)})$
 - 4: $x_2^{(t)}$ according to $p_2(x_2|x_1^{(t)})$
-

As a matter of convenience we specify the prior for μ and after sampling we record $\beta = \exp(\mu/2)$ following Kim et al. (1998). The priors are assumed to be independent, the Bayes' theorem states

$$p(\theta, h|y) \propto p(y|\theta, h)p(h|\theta)p(\theta) \quad (2.54)$$

and thus it follows that

$$\begin{aligned} p(\phi|y, h, \mu, \sigma_\eta^2) &\propto p(h|\mu, \phi, \sigma_\eta^2)p(\phi), \\ p(\mu|y, h, \phi, \sigma_\eta^2) &\propto p(h|\mu, \phi, \sigma_\eta^2)p(\mu), \\ p(\sigma_\eta^2|y, h, \mu, \phi) &\propto p(h|\mu, \phi, \sigma_\eta^2) \end{aligned} \quad (2.55)$$

Expressions in the equation (2.55) are used for deriving the posterior distributions from which one samples in the Gibbs sampling method. In the next subsection we derive sampling distributions for the prior specifications assumed earlier.

Sampling the parameters

Sampling σ_η . Assuming the prior for σ_η being Inverse-Gamma, $IG(\alpha/2, \beta/2)$ and given μ and ϕ , σ_η is sampled from

$$\begin{aligned} \sigma_\eta \mid y, h, \mu, \phi &\sim IG\left(\frac{T + \alpha}{2}, \right. \\ &\left. \frac{\beta + (h_1 - \mu)^2(1 - \phi^2) + \sum_{t=1}^{T-1} ((h_{t+1} - \mu) - \phi(h_t - \mu))^2}{2} \right) \end{aligned} \quad (2.56)$$

Sampling ϕ . Define $\phi = 2\phi^* - 1$, and assume $\phi^* \sim \text{Beta}(\phi^{(1)}, \phi^{(2)})$, then the prior for the parameter ϕ has the form

$$p(\phi) \propto \left\{ \frac{(1 + \phi)}{2} \right\}^{\phi^{(1)} - 1} \left\{ \frac{(1 - \phi)}{2} \right\}^{\phi^{(2)} - 1}, \quad \phi^{(1)}, \phi^{(2)} > \frac{1}{2} \quad (2.57)$$

Sampling μ . Parameter $\mu \mid h, \phi, \sigma_\eta$ is sampled from $N(c, d^2)$, where

$$c = d^2 \left\{ \frac{(1 - \phi^2)}{\sigma_\eta^2} h_1 + \frac{(1 - \phi)}{\sigma_\eta^2} \sum_{t=1}^{T-1} (h_{t+1} - \phi h_t) \right\} \quad (2.58)$$

$$d^2 = \sigma_\eta^2 \{(T - 1)(1 - \phi)^2 + (1 - \phi^2)\}^{-1} \quad (2.59)$$

PGAS algorithm

As we mentioned, in state-space models Gibbs sampler iterated between sampling the parameters of the model and sampling latent states. In the case of stochastic volatility models sampling latent states is challenging due to nonlinearity of the observation equation. However, as we discussed previously SMC methods can be applied to this problem. This is the idea that stands behind particle Gibbs with ancestor sampling (PGAS). PGAS, see Algorithm 10, returns a particle trajectory that is an estimate of the filtering process which further can be used in Gibbs sampling algorithm for state-space models, see Algorithm 11.

Definition 2.6. (PGAS kernels) For any $N \geq 1$ and any $\theta \in \Theta$, Algorithm 10 maps $x'_{1:T}$ stochastically into $x^*_{1:T}$, thus implicitly defining a Markov kernel P_θ^N on $(X^T, \mathcal{X}^T)$. The class of Markov kernels $\{P_\theta^N : \theta \in \Theta\}$, indexed by $N \geq 1$ is referred to as the PGAS class of kernels (Lindsten et al., 2014).

Algorithm 10 Particle Gibbs with Ancestor Sampling

- 1: Input: Reference trajectory $x'_{1:T} \in X^T$ and parameter $\theta \in \Theta$.
- 2: Output: Sample $x_{1:T}^* \sim P_\theta^N(x'_{1:T}, \cdot)$ from the PGAS Markov kernel.
- 3: Draw $x_1^i \sim r_{\theta,1}(x_1)$ for $i = 1, \dots, N-1$.
- 4: Set $x_1^N = x'_1$.
- 5: Set $w_1^i = W_{\theta,1}(x_1^i)$ for $i = 1, \dots, N$.
- 6: **for** $t=2, \dots, T$ **do**
- 7: Draw $\{a_t^i, x_t^i\} \sim M_{\theta,t}(a_t, x_t)$ for $i = 1, \dots, N-1$.
- 8: Set $x_t^N = x'_t$.
- 9: Compute $\{\tilde{w}_{t-1|T}^i\}_{i=1}^N$ according to

$$\tilde{w}_{t-1|T}^i = w_{t-1}^i \frac{\gamma_{\theta,T}((x_{1:t-1}^i, x'_{t:T}))}{\gamma_{\theta,t-1}(x_{1:t-1}^i)}, \text{ for } i = 1, \dots, N.$$

- 10: Draw a_t^N with $\mathbb{P}(a_t^N = i) \propto \tilde{w}_{t-1|T}^i$.
 - 11: Set $x_{1:t}^i = (x_{1:t-1}^{a_t^i, x_t^i})$ for $i = 1, \dots, N$.
 - 12: Set $w_t^i = W_{\theta,t}(x_{1:t}^i)$ for $i = 1, \dots, N$.
 - 13: **end for**
 - 14: Draw k with $\mathbb{P}(k = i) \propto w_T^i$. **return** $x_{1:T}^* = x_{1:T}^k$.
-

Algorithm 11 Particle Gibbs Sampler for SSM

- 1: Initialize: $h_{0:T}^1$ and θ^1 .
 - 2: **for** $m=1, \dots, M$ **do**
 - 3: Draw $\theta^{(m)} \sim p(\theta|h_{0:T}^{(m-1)} y_{1:T})$
 - 4: Draw $h_{0:T}^{(m)} \sim \kappa_{N,\theta^{(m)}}(h_{0:T}^{(m-1)}, h_{0:T}^*)$
 - 5: **end for**
-

Table 2.8: **Example of posterior estimation of the parameters of stochastic volatility model with PGAS algorithm.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	95% <i>HPD_l</i>	95% <i>HPD_u</i>
β	1.0002	1.0000	0.9980	0.9779	1.0214
ϕ	0.8621	0.8893	0.7255	0.6510	0.9998
σ_η	0.0012	0.0006	0.0004	0.0001	0.0034

Note: Here is given example of PGAS estimation of the posterior distribution for the stochastic volatility model with the number of particles $N=500$ and $M=10000$ samples from the posterior. The data were generated with the parameters $\beta = 1.189$, $\phi = 0.965$, $\sigma^2 = 0.134$.

2.5.9 Overall comments on SMC and PMCMC

In this section we reviewed SMC and MCMC methods with application to stochastic volatility models. In particular, we compared bootstrap and auxiliary particle filter and showed that for the stochastic volatility model the auxiliary particle filter does not always outperform the bootstrap filter. The simulation study was done for the true parameter values as well as for other points of the parameter space. In particular, this gave us insights that the variance of the estimated likelihood is most sensitive to the variance of the latent process. Other simulation studies (see section on particle Markov Chain Monte Carlo), as well as simulation studies methods in the further sections, show that this is indeed the parameter that is hardest to estimate for univariate stochastic volatility model.

We further reviewed iterated auxiliary particle filter (iAPF) that can be used for the estimation of the likelihood and reduce variance of the estimated likelihood by taking into account the whole data set available when propagating the particles. An alternative method that we do not discuss in detail in this paper is efficient importance sampling and in particular particle efficient importance sampling. The idea of the method is similar in a sense that the authors offer an approach to make better proposals in the particle propagation stage. This is done through finding the importance parameters a_t that depend on the observations $y_{1:T}$ and are found through an optimization problem. For more details we refer the reader to the original paper Scharth and Kohn (2016). In particular in multivariate case finding importance parameters can be challenging. Even though both algorithms allow to decrease the variance of the estimated likelihood and generally allow to use a smaller number of particles in the estimation, in the proposed implementation they are going to be a) very expensive in multivariate framework; b) ignore the dependence structure in the Φ matrix in which we are interested. Thus, in particular for the iAPF it is of interest to develop an approach for the Algorithm 7 that would be suitable for the multivariate framework. Some approximate methods, such as Expectation Propagation might be a solution. Thus, the variance of the estimated like-

likelihood is the problem that comes from the SMC part of the estimation and needs to be tackled carefully once one considers the problem in high dimensions.

Metropolis-Hastings as a general MCMC method is easy to implement and works well for the univariate model. The estimation results are satisfying when it is properly calibrated and good mixing of the chains is achieved. One, however, can foresee that even though it might work in low-dimensional multivariate models, it is not practical in high-dimensional models. The reason is that we are going to move in a high-dimensional parameter space and Metropolis-Hastings algorithm is not equipped with a tool to make good proposals in high-dimensional space which will result in low acceptance rates and poor mixing. Up until certain dimension, one might use proposal from a mixing distribution with small and large proposal steps. However, in truly high-dimensional problem it is very unlikely that would work sufficiently well either. In the next section we review and illustrate methods that are called Hamiltonian Monte Carlo. Originating from physics and taking advantage of differential geometry, these methods allow to make proposals that are accepted with high probability. Thus, combination of the approaches can be beneficial.

2.6 Riemann Manifold Langevin and Hamiltonian Monte Carlo

In this section, we review Riemann Manifold Langevin Hamiltonian Monte Carlo methods that are introduced in Girolami and Calderhead (2011b) and can be applied for estimation of the stochastic volatility models. Hamiltonian Monte Carlo (HMC) methods originate in the statistical physics literature. Recall, that MCMC methods can be used to sample from the posterior distribution of the parameters. However, when the problem is high-dimensional standard methods such as Metropolis-Hastings usually take a long time to converge. Assume, that we are at a parameter state θ_i , where i denotes current iteration. In Metropolis-Hastings algorithm we make a random walk proposal and get a candidate value θ_{i+1}^c . With a cer-

tain probability, see for more details on its specification in Algorithm 1, we either accept this candidate and set $\theta_{i+1} = \theta_{i+1}^c$ or reject the candidate and set $\theta_{i+1} = \theta_i$. For the algorithm to converge, we need to calibrate the acceptance rate to an optimal value. However, even when it is calibrated well, the random walk proposal functions in such a way that with equal probability the next value of the candidate can be in any direction. The result is that Metropolis-Hastings and similar algorithms are not capable of making large transitions with high acceptance probability: when we want to make a proposal θ_{i+1}^c that is far away from the current state of the parameter vector θ_i only a few out of many proposals are likely to be accepted while all of them are equally likely to be proposed. HMC solves this issue and allows to make big transitions with high acceptance probability. The idea of HMC is based on the relationship between differential geometry and statistical theory (MCMC in particular). We will discuss the general idea first. Further, we will discuss Metropolis adjusted Langevin algorithm defined on the Riemann manifold which was proposed in Girolami and Calderhead (2011b) and illustrate its performance in the estimation of stochastic volatility models. It is in particularly useful when sampling from high-dimensional densities that may show a strong correlation. We further provide the general background and summary of the algorithms together with the necessary quantities for their implementation in the case of stochastic volatility models.

2.6.1 Hamiltonian Monte Carlo

In standard MCMC setting one uses a probability distribution to propose the next state of the Markov chain. Hamiltonian Monte Carlo methods exploit physical system dynamics to make proposals for the next state. It can improve the mixing drastically and result in a more efficient algorithm. Especially, since we are interested in multivariate modeling, more efficient exploration of the posterior distribution is of interest. Once the dimension of the model grows with standard random walk it is very hard to make proposals that would be accepted frequently enough and result in a good

mixing Markov chain. We first introduce some basic ideas on which Hamiltonian Monte Carlo method is built, for an extensive introduction we refer to Betancourt (2017). Here we give a basic notation and definitions for Hamiltonian dynamics. More details, as well as various examples from a dynamical systems point of view, can be found in Meyer et al. (2008), for more details about the relation to statistics we keep referring to Betancourt (2017).

For understanding the intuition behind Hamiltonian dynamics, it might be useful to recall the second Newton's law states that the acceleration of the object's centre of mass is equal to the sum of the forces acting on it divided by the object's mass (Crowell, 2003). The idea brings up the second-order differential equations in $\mathbb{R}_n$, and the equations of motion can be written as a Hamiltonian system. In other words, we are describing how an object is moving through the system in time. We further define Hamiltonian system through the equations

$$\dot{\theta} = H_p, \quad \dot{p} = -H_\theta, \quad (2.60)$$

$$\dot{\theta}_i = \frac{\partial H}{\partial p_i}(t, \theta, p), \quad \dot{p}_i = -\frac{\partial H}{\partial \theta_i}(t, \theta, p), \quad i = 1, \dots, n, \quad (2.61)$$

with $H = H(t, \theta, p)$ called the Hamiltonian which is assumed to be a smooth real-valued function for $(t, \theta, p) \in \mathcal{O}$, an open set in $\mathbb{R}^1 \times \mathbb{R}^n \times \mathbb{R}^n$. Further, $\theta = (\theta_1, \dots, \theta_n)^3$ and $p = (p_1, \dots, p_n)$ are called the position and the momentum vectors respectively and where t denotes time (Meyer et al., 2008). The variables p and θ are called conjugate. The momentum parameters $\mathbf{p}$ complement the position parameters that are correspond to our target parameter space and allow to expand the D -dimensional parameter space $2D$ -dimensional phase space (Betancourt, 2017). Further, the

³In physics literature the momentum vectors often have notation q instead of θ , however for consistency in terminology we will use θ .

canonical density $\pi(q, p)$ can be defined through the Hamiltonian function

$$\pi(p, \theta) = \exp^{-H(\theta, p)}. \quad (2.62)$$

This is the function from which we would like to sample in our MCMC setup. Adopting the physics terminology the energy point is defined as a value of the function $H(\cdot)$ at any point in phase space (where phase space represents all possible states of a system). It follows

$$H(\theta, p) = -\log p(\theta, p), \quad (2.63)$$

and it can be decomposed into $K(p, \theta)$ – kinetic energy – and $V(\theta)$ potential energy

$$H(\theta, p) = -\log p(p \mid \theta) - \log p(\theta) = K(p, \theta) + V(\theta). \quad (2.64)$$

Then the Hamiltonian equations can be rewritten in terms of kinetic energy and potential energy as follows

$$\frac{d\theta}{dt} = +\frac{\partial H}{\partial p} = \frac{\partial K}{\partial p} \quad (2.65)$$

$$\frac{dp}{dt} = -\frac{\partial H}{\partial \theta} = -\frac{\partial K}{\partial \theta} - \frac{\partial V}{\partial \theta}. \quad (2.66)$$

Note that $\frac{\partial V}{\partial \theta}$ can be interpreted as the gradient of the logarithm of the target density. Hamiltonian Monte Carlo methods usually have two important steps: updates of the momentum parameters and Metropolis updates. The latter are performed with the Hamiltonian dynamics in mind which defines a trajectory for the new state. This kind of 'efficient' proposal ensures that we would accept new states quite frequently even if they are far away from the current state. When one constructs MCMC with Hamiltonian dynamics it is important to remember that some of its properties have to be satisfied. The Hamiltonian dynamics must be reversible, the Hamiltonian must

be invariant, Hamiltonian dynamics must preserve volumes and it must be symplectic. Even though we do not go into details of these properties here, we refer an interested reader to Neal et al. (2011).

In order to construct MCMC with Hamiltonian dynamics, some important properties of the latter has to be taken into account. Hamiltonian dynamics is *reversible* and has one-to-one mapping from the state at time t to the state at time $t + s$. Further, the Hamiltonian is *invariant*

$$\frac{dH}{dt} = \sum_{i=1}^d \left[\frac{dq_i}{dt} \frac{\partial H}{\partial q_i} + \frac{dp_i}{dt} \frac{\partial H}{\partial p_i} \right] = \sum_{i=1}^d \left[\frac{\partial H}{\partial p_i} \frac{\partial H}{\partial q_i} - \frac{\partial H}{\partial q_i} \frac{\partial H}{\partial p_i} \right] = 0. \quad (2.67)$$

It follows, that theoretically when we construct Metropolis with Hamiltonian proposal dynamics the acceptance probability should be 1 when H is invariant. Practically this is hard to achieve and usually one can make H approximately invariant Neal et al. (2011).

The next property in which we are interested is called *volume preservation*.

$$\begin{aligned} \sum_{i=1}^d \left[\frac{\partial}{\partial q_i} \frac{dq_i}{dt} + \frac{\partial}{\partial p_i} \frac{dp_i}{dt} \right] &= \sum_{i=1}^d \left[\frac{\partial}{\partial q_i} \frac{\partial H}{\partial p_i} - \frac{\partial}{\partial p_i} \frac{\partial H}{\partial q_i} \right] = \\ &= \sum_{i=1}^d \left[\frac{\partial^2 H}{\partial q_i \partial p_i} - \frac{\partial^2 H}{\partial p_i \partial q_i} \right] = 0. \end{aligned} \quad (2.68)$$

When it comes to constructing MCMC the volume preservation property plays the following role: it is not necessary to account for the change in volume in the acceptance probabilities for the updates in Metropolis algorithm. For more examples and illustration of this property we refer the reader to Neal et al. (2011).

Another property that lies in the core of construction of the Hamiltonian dynamics is its *symplecticness*. From mathematical point of view this means that the Jacobian matrix (B_s), of the mapping T_s satisfies (Neal et al., 2011)

$$B_s^T J^{-1} B_s = J^{-1}. \quad (2.69)$$

The volume conservation also follows from the Equation (2.69)

$$\det(B_s^T)\det(J^{-1})\det(B_s) = \det(J^{-1}) \Rightarrow \det(B_s)^2 = 1. \quad (2.70)$$

It is important to note that these properties – reversibility, preservation of volume, symplecticness – hold even when the Hamiltonian dynamics itself is approximated which is often necessary in practice Neal et al. (2011).

2.6.2 Computational Implementation: discretizing Hamiltonian's equations

Hamiltonian equations are given in continuous time, for practical implementation however one must discretize them by taking some small step size, let us call it ϵ . Thereby, starting with time 0 we further discretize the time into steps of the size ϵ , i.e. we get the grid $\epsilon, 2\epsilon, 3\epsilon, \dots$. Assuming that the Hamiltonian is represented by the form $H(q, p) = U(q) + K(p)$ and let for simplicity also assume that $K(p) = p^T M^{-1} p / 2$ with M being diagonal matrix which implies

$$K(p) = \sum_{i=1}^d \frac{p_i^2}{2m_i}, \quad (2.71)$$

where $m_i, i = 1, \dots, d$ are the diagonal elements of M . There are multiple ways in which one can discretize differential equations. Following Neal et al. (2011), we will discuss multiple methods as they might give the reader intuition about the problem of discretization. First, let us consider *Euler's method* which is perhaps most known. The Hamiltonian equations then can be expressed as

$$p_i(t + \epsilon) = p_i(t) + \epsilon \frac{dp_i}{dt}(t) = p_i(t) - \epsilon \frac{\partial U}{\partial q_i}(q(t)), \quad (2.72)$$

$$q_i(t + \epsilon) = q_i(t) + \epsilon \frac{dq_i}{dt}(t) = q_i(t) + \epsilon \frac{p_i(t)}{m_i}, \quad (2.73)$$

with the index $i = 1, \dots, d$. Thus, we start at time $t=0$ with initial values for $q_i(0)$ and $p_i(0)$, and then the steps will be iterated to get the position and momentum values, in particular the approximate values for $q(\tau)$ and $p(\tau)$ after τ/ϵ steps, where τ/ϵ is an integer value. The performance of the algorithm depends on the value of ϵ and number of steps that the algorithm takes, say N_{HMC} , where $\tau = \epsilon \times N_{HMC}$. As shown in Neal et al. (2011) with bigger values of ϵ and smaller number of steps the produced trajectory tends to diverge to infinity with the true trajectory being a circle. With smaller values of ϵ and bigger number of steps N_{HMC} the result is more plausible, yet the divergence to infinity is not completely eliminated, but rather happens slower. Thus, one would want a better way of the discretization for this problem. The *modified Euler's method* can produce better results. The discretized Hamiltonian equations in this case are

$$p_i(t + \epsilon) = p_i(t) - \epsilon \frac{\partial U}{\partial q_i}(q(t)), \quad (2.74)$$

$$q_i(t + \epsilon) = q_i + \epsilon \frac{p_i(t + \epsilon)}{m_i}. \quad (2.75)$$

Here the properties of the Hamiltonian dynamics we discussed earlier come into play. In particular, the performance of the modified Euler's method is improved because it preserves volume even though the time is discrete in this case. As Neal et al. (2011) note, one could see that the volume is preserved by following this logic: both transformations from $(q(t), p(t))$ to $(q(t), p(t + \epsilon))$ and from $(q(t), p(t + \epsilon))$ to $(q(t + \epsilon), p(t + \epsilon))$ imply that either p_i or q_i change by the quantity which is dependent only on the values of the variables that do not change. In this sense, it is a "shear" transformation, and such a transformation always preserves volume (Neal et al., 2011).

The modified Euler's methods perform much better than standard Euler's method, however, it is still possible to improve it. A more satisfactory alternative is known as *the leapfrog method*. The corresponding system of

equations reads

$$p_i(t + \epsilon/2) = p_i(t) - (\epsilon/2) \frac{\partial U}{\partial q_i}(q(t)), \quad (2.76)$$

$$q_i(t + \epsilon) = q_i(t) + \epsilon \frac{p_i(t + \epsilon/2)}{m_i}, \quad (2.77)$$

$$p_i(t + \epsilon) = p_i(t + \epsilon/2) - (\epsilon/2) \frac{\partial U}{\partial q_i}(q(t + \epsilon)). \quad (2.78)$$

The idea is that we iterate between a half-step for the moment variables and a full step of the position variables: make a half-step for the momentum, use these values for updating the position, make another half-step for the momentum. This method fulfils the properties of volume preservation and reversibility. Neal et al. (2011) also show that for a simple example (target being a circle) even for big step size ϵ trajectory is stable.

2.6.3 Constructing MCMC with Hamiltonian Dynamics

Eventually, we are interested in how to exploit Hamiltonian dynamics in constructing Markov Chain Monte Carlo algorithms. As Neal et al. (2011) mention in their introduction to these methods, one has to translate the density function from which one wants to sample to a potential energy function and introduce *momentum variables* into the algorithm. The position variables in this case correspond to the original variables of interest. In this case the algorithm builds upon resampling the momentum and proceeding to a Metropolis update that uses Hamiltonian dynamics.

Remember, that we would like to sample from a distribution. In order to do that in the framework with Hamiltonian dynamics, we need to relate it to a potential energy function. This can be done through a concept known as *canonical distribution*. Let $\mathbb{E}(x)$ be some energy function for the state x . Then the canonical distribution has the density function that reads

$$P(x) = \frac{1}{Z} \exp\left(\frac{-\mathbb{E}(x)}{T}\right). \quad (2.79)$$

Note, that here the notation changes from the one we use through out the paper. T in Equation (2.79) is the temperature of the system, Z is the normalizing constant. The joint distribution for the position (p) and momentum (q) is expressed then through the following expression (Neal et al., 2011)

$$P(p, q) = \frac{1}{Z} \exp\left(\frac{-H(q, p)}{T}\right). \quad (2.80)$$

Further, assuming that $H(q, p) = U(q) + K(p)$ Equation (2.80) becomes

$$P(q, p) = \frac{1}{Z} \exp\left(\frac{-U(q)}{T}\right) \exp\left(\frac{-K(p)}{T}\right). \quad (2.81)$$

From this point we come back to the idea that we operate in Bayesian framework. This means that we are interested in the posterior distribution of the parameters of interest (as mentioned before, this parameters correspond to the variables of position q). Then by taking $T=1$ and a potential energy expressed as

$$U(q) = -\log[\pi(q)L(q|D)], \quad (2.82)$$

we can get the posterior in canonical form. In Equation (2.82) $\pi(q)$ is the prior distribution and $L(q|D)$ is the likelihood given the data.

Finally, the idea behind construction and implementation of HMC lies in iteration between two steps we mentioned: update of the momentum and position. First, the momentum is updated y drawing variables from the Gaussian distribution randomly and independently of the current state of the variables of interest (position variables). Then both momentum and position are updated: the Hamiltonian dynamics is used for the proposal of the new state. L steps are generated with the leapfrog method discussed before with a step-size ϵ . In the end of step L the momentum variables

are updated. And hence we get a candidate joint state q^c, p^c . Then the acceptance probability for this state is

$$\min \{1, \exp(-H(q^c, p^c)) + H(q, p)\} = \min [1, \exp(-U(q^c) + U(q) - K(p^c + K(p)))] . \quad (2.83)$$

L and ϵ in this algorithm have to be calibrated to achieve optimal performance. For more details on that we refer to Neal et al. (2011).

Thus, HMC algorithm bypasses random walk proposals which are used in random-walk Metropolis-Hastings algorithm. However, just applying HMC algorithm to the problem of our interest – stochastic volatility models – is not straightforward. In the next sections we will discuss algorithms that build up on the methods described and can be applied to the problem we are interested in. We illustrate the methods discussed below by simulation examples.

2.6.4 Metropolis Adjusted Langevin Algorithm

In this section we describe Metropolis Adjusted Langevin Algorithm (MALA) algorithm on which further Riemann Manifold Metropolis Adjusted Langevin Algorithm (RM-MALA) is built, see for more details Girolami and Calderhead (2011b). It is defined through stochastic differential equations for the random vector $\theta \in \mathbb{R}^n$ that has density $p(\theta)$.

$$d\theta(t) = \frac{1}{2} \nabla_{\theta} \log p(\theta) dt + db(t) \quad (2.84)$$

with b being a Brownian motion process with dimension n . The proposal mechanism for MCMC scheme is defined through the first-order Euler discretisation of the SDE

$$\theta^{n+1} = \theta^n + \frac{\epsilon^2}{2} \nabla_{\theta} \log p(\theta^n) + \epsilon z^n, \quad (2.85)$$

where ϵ denotes the integration step size and $z \sim N(z|0, I)$. The proposal density then is $q(\theta^*|\theta^n) = N(\theta^*|\mu(\theta^n, \epsilon), \epsilon^2 I)$, where $\mu(\theta^n, \epsilon) = \theta^n + \frac{\epsilon^2}{2} \nabla_{\theta} \log p(\theta)$. Then the acceptance probability in the Metropolis-Hastings al-

gorithm takes standard form with the proposal $q(\theta^*|\theta^n)$. Thereby, MALA iterates between two general steps. First, Langevin dynamics is used for the proposals and it exploits the gradients of the target. Second, the proposals are accepted or rejecting within Metropolis-Hastings setup. Originally the method was proposed in Roberts and Rosenthal (1998) and in the next section we will review the extension which was proposed in Girolami and Calderhead (2011b).

2.6.5 Riemann Manifold Metropolis Adjusted Langevin Algorithm

We further define Riemann Manifold Metropolis Adjusted Langevin Algorithm (RM-MALA) through the Langevin diffusion on the manifold that is expressed as stochastic differential equation of the form

$$d\theta(t) = \frac{1}{2} \tilde{\nabla}_{\theta} \log p(\theta(t)) dt + d\tilde{b}(t), \quad (2.86)$$

with the natural gradient (Girolami and Calderhead, 2011b; Amari and Nagaoka, 2007) $\tilde{\nabla}_{\theta} \log p(\theta(t)) = G^{-1}(\theta(t)) \nabla_{\phi} \theta \log p(\theta(t))$ where $G(\theta)$ is the metric tensor. Further, the Brownian motion on the Riemannian manifold is defined by

$$db_i(\tilde{t}) = |G(\theta(t))|^{-1/2} \sum_{j=1}^n \frac{\partial}{\partial \theta_j} \left(G^{-1}(\theta(t))_{ij} |G(\theta(t))|^{1/2} \right) dt + (\sqrt{G^{-1}(\theta(t))} db(t))_i. \quad (2.87)$$

The discretisation of the SDE above implies the first order Euler integrator as follows (Girolami and Calderhead, 2011b)

$$\theta_i^{n+1} = \mu(\theta^n, \epsilon)_i + (\epsilon \sqrt{G^{-1}(\theta^n)} z^n)_i. \quad (2.88)$$

Further, note that for the flat manifold that has constant curvature simplifies to

$$\theta^{n+1} = \theta^n + \frac{\epsilon^2}{2} G^{-1}(\theta^n) \nabla \log p(\theta^n) + \epsilon \sqrt{G^{-1}(\theta^n)} z^n, \quad (2.89)$$

where ϵ is the optimal scaling. The corresponding algorithm mMALA is summarized in the Algorithm 12.

2.6.6 Riemann Manifold Hamiltonian Monte Carlo

Girolami and Calderhead (2011b) define HMC methods in the form of Riemann manifold, and this can be seen as generalization of HMC. The Hamiltonian on the Riemann manifold is defined as follows

$$H(\theta, \mathbf{p}) = -\log p(\theta) + \frac{1}{2} \log((2\pi)^n |G(\theta)|) + \frac{1}{2} \mathbf{p}^T G(\theta) \mathbf{p} \quad (2.90)$$

with $\exp(-H(\theta, \mathbf{p})) = p(\theta, \mathbf{p}) = p(\theta)p(\mathbf{p} | \theta)$ and the marginal density reads

$$\begin{aligned} p(\theta) &\propto \int \exp(-H(\theta, \mathbf{p})) d\mathbf{p} = \frac{\exp\{\log p(\theta)\}}{\sqrt{2\pi^n |G(\theta)|}} \times \\ &\int \exp\left\{-\frac{1}{2} \mathbf{p}^T G(\theta)^{-1} \mathbf{p}\right\} d\mathbf{p} = \exp(\log p(\theta)). \end{aligned} \quad (2.91)$$

In the next section we discuss implementation for the stochastic volatility model.

2.6.7 RM-HMC and mMALA for SV model

Recall the stochastic volatility model with the scale parameter β in the observation equation, ϕ persistence parameter in volatility process and noise variance σ . In this specification the model reads

$$y_t = \beta \exp(h_t/2) \epsilon_t, \quad h_{t+1} = \phi h_t + \eta_{t+1}, \quad (2.92)$$

$\epsilon_t \sim N(0, 1)$, $\eta_t \sim N(0, \sigma^2)$, with $h_1 \sim N(0, \sigma^2/(1 - \phi^2))$.

We can write down the joint likelihood of the model as follows

$$p(y, h, \beta, \phi, \sigma) = \prod_{t=1}^T p(y_t | h_t, \beta) \prod_{t=2}^T p(h_t | h_{t-1}, \phi, \sigma) \pi(\beta) \pi(\phi) \pi(\sigma) \quad (2.93)$$

For simulation example in the next subsection we specify prior distributions as in Girolami and Calderhead (2011b)

$$\begin{aligned} \beta &\propto \exp(\beta), \\ \sigma^2 &\sim \text{Inv} - \chi^2(10, 0.05), \\ (\phi + 1)/2 &\sim \text{Beta}(20, 1.5). \end{aligned} \quad (2.94)$$

Further, following Girolami and Calderhead (2011b) we write the partial derivatives of $L = p(y, h | \beta, \phi, \sigma)$ with respect to β , ϕ and σ . We get the following expressions

$$\frac{\partial L}{\partial \beta} = -\frac{T}{\beta} + \sum_{t=1}^T \frac{y^2}{\beta^3 \exp(h_t)}, \quad (2.95)$$

$$\frac{\partial L}{\partial \phi} = -\frac{\phi}{(1 - \phi^2)} + \frac{\phi h_1^2}{\sigma^2} + \sum_{t=2}^T \frac{h_{t-1}(h_t - \phi h_{t-1})}{\sigma^2}, \quad (2.96)$$

$$\frac{\partial L}{\partial \sigma} = -\frac{T}{\sigma} + \frac{h_1^2(1 - \phi^2)}{\sigma^3} + \sum_{t=2}^T \frac{(h_t - \phi h_{t-1})^2}{\sigma^3}. \quad (2.97)$$

Implementation of the algorithms requires the expressions for the individual components of the metric tensor for the likelihood. We get the following

expressions (Girolami and Calderhead, 2011b)

$$\begin{aligned} \mathbb{E} \left\{ \frac{\partial L}{\partial \beta} \frac{\partial L}{\partial \beta} \right\} &= \frac{2T}{\beta^2}, & \mathbb{E} \left\{ \frac{\partial L}{\partial \sigma} \frac{\partial L}{\partial \sigma} \right\} &= \frac{2T}{\sigma^2}, \\ \mathbb{E} \left\{ \frac{\partial L}{\partial \beta} \frac{\partial L}{\partial \sigma} \right\} &= \mathbb{E} \left\{ \frac{\partial L}{\partial \beta} \frac{\partial L}{\partial \phi} \right\} = 0, & (2.98) \\ \mathbb{E} \left\{ \frac{\partial L}{\partial \sigma} \frac{\partial L}{\partial \phi} \right\} &= \frac{2\phi}{\sigma^3(1-\phi^2)}, & \mathbb{E} \left\{ \frac{\partial L}{\partial \phi} \frac{\partial L}{\partial \phi} \right\} &= \frac{2\phi^2}{(1-\phi^2)^2} + \frac{T-1}{1-\phi^2} \end{aligned}$$

Also the expressions for the metric tensor for the likelihood and its partial derivatives follow

$$G(\phi, \sigma, \beta) = \begin{bmatrix} \frac{2T}{\beta^2} & 0 & 0 \\ 0 & \frac{2T}{\sigma^2} & \frac{2\phi}{\sigma^3(1-\phi^2)} \\ 0 & \frac{2\phi}{\sigma^3(1-\phi^2)} & \frac{2\phi^2}{(1-\phi^2)^2} + \frac{T-1}{1-\phi^2} \end{bmatrix}, \quad (2.99)$$

$$\frac{\partial G}{\partial \beta} = \begin{bmatrix} -\frac{4T}{\beta^3} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad (2.100)$$

$$\frac{\partial G}{\partial \sigma} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\frac{4T}{\sigma^3} & -\frac{6\phi}{\sigma^4(1-\phi^2)} \\ 0 & -\frac{6\phi}{\sigma^4(1-\phi^2)} & 0 \end{bmatrix}, \quad (2.101)$$

$$\frac{\partial G}{\partial \phi} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \frac{2}{\sigma^3(1-\phi^2)} + \frac{4\phi^2}{\sigma^3(1-\phi^2)^2} \\ 0 & \frac{2}{\sigma^3(1-\phi^2)} + \frac{4\phi^2}{\sigma^3(1-\phi^2)^2} & \frac{2\phi(1+T)}{(1-\phi^2)^2} + \frac{6\phi^3}{(1-\phi^2)^3} \end{bmatrix} \quad (2.102)$$

Finally, Algorithms 12 and 13 correspondingly summarize Metropolis Adjusted Langevin Lagorithm and Riemann Manifold Hamiltonian Monte Carlo. Note, in the latter algorithm the crucial expression is $H(\theta, p)$, Riemann manifold, is defined in Equation (2.90).

Algorithm 12 Manifold Metropolis Adjusted Langevin Algorithm

```

1: Initialize current  $\theta$ 
2: for  $i = 1$  to NumSamples do
3:   Sample  $\theta^*$  based on current  $\theta$  according to the first order discretisation
4:   Calculate current log-likelihood  $L(\theta)$  and proposed log-likelihood  $L(\theta^*)$ 
5:   Calculate  $\log p(\theta^* | \theta)$ ,  $\log p(\theta | \theta^*)$ ,  $\log \pi(\theta)$ ,  $\log \pi(\theta^*)$ 

       
$$a = L(\theta^*) + \log \pi(\theta^* + \log p(\theta | \theta^*) - L(\theta) - \log \pi(\theta) - \log p(\theta^* | \theta))$$


6:   if  $a > 0$  OR  $a > \log(rand)$  then return
7:     Current  $\theta = \theta^*$ 
8:   end if
9: end for

```

Algorithm 13 RMHMC with Generalized Leapfrog

```

1: Initialize current  $\theta$ 
2: for IterationNum=1 to NumSamples do
3:   Sample new momentum  $\mathbf{p}^1$ 
4:   Calculate current  $H(\theta, \mathbf{p}^1)$ 
5:   Randomize  $N$  (leapfrog steps)
6:    $\theta^1 = \text{Current}\theta$ 
7:   for  $n = 1$  to  $N$  (leapfrog steps) do
8:     %Update the momentum with fixed point iterations
9:      $\hat{\mathbf{p}}^0 = \mathbf{p}^n$ 
10:    for  $i = 1$  to NumOfFixedPointSteps do
11:       $\hat{\mathbf{p}}^i = \mathbf{p}^n - \frac{\epsilon}{2} \nabla_{\theta} H(\theta^n, \hat{\mathbf{p}}^{i-1})$ 
12:    end for
13:     $\mathbf{p}^{n+\frac{1}{2}} = \hat{\mathbf{p}}^i$ 
14:    % Update the parameters with fixed point iterations
15:     $\hat{\theta}^i = \theta^n$ 
16:    for  $i = 1$  to NumOfFixedPointSteps do
17:       $\hat{\theta}^i = \theta^n + \frac{\epsilon}{2} \nabla_{\mathbf{p}} H(\theta^n, \mathbf{p}^{n+\frac{1}{2}}) + \frac{\epsilon}{2} \nabla_{\mathbf{p}} H(\hat{\theta}^i, \mathbf{p}^{n+\frac{1}{2}})$ 
18:    end for
19:     $\theta^{n+1} = \hat{\theta}^i$ 
20:    % Update the momentum exactly  $\mathbf{p}^{n+1} = \mathbf{p}^{n+1} - \frac{\epsilon}{2} \nabla_{\theta} H(\theta^{n+1}, \mathbf{p}^{n+\frac{1}{2}})$ 
21:  end for
22:  Calculate proposed  $H(\theta^N, \mathbf{p}^N)$ 
23:   $Ratio = -\log(ProposedH) + \log(CurrentH)$ 
24:  % Accept or reject according to Metropolis ratio
25:
26:  if  $Ratio > 0$  OR  $Ratio > \log(rand)$  then return
27:    Current  $\theta = \theta^N$ 
28:  end if
29: end for

```

2.6.8 Illustration of RM-HMC with simulated data

In this section we illustrate performance of the RM-HMC on multiple time series generated from the stochastic volatility model. We show the figures with trace plots of the parameters, histograms and autocorrelation functions. In the tables we report moments of the posterior distributions, highest posterior density intervals and true values with which the data were generated.

Figure 2.17: Trace plots, histograms and autocorrelation functions for β , ϕ , σ_η in the estimation of the SV model with RM-HMC.

Note: This figure shows trace plots, histograms and autocorrelation functions for the estimation of the SV model with data generating process described in Table 2.9.

Table 2.9: **Posterior results for estimation of the SV model with RM-HMC.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	<i>95% HPD_l</i>	<i>95% HPD_u</i>	<i>True</i>
β	1.18628	1.18582	1.16608	1.10609	1.26526	1.18870
ϕ	0.83824	0.84920	0.78390	0.71272	0.93643	0.95932
σ	0.21889	0.21316	0.17203	0.14846	0.30072	0.08852

Note: This table shows posterior results for the parameters β , ϕ , σ . The scale parameter, β , it captures quite well and lies inside the highest posterior density interval with the mean being almost equal to the true value. Persistence parameter, ϕ , and variance of the latent volatility process, σ , are not that well captured. The former is underestimated and the latter overestimated.

Figure 2.18: **Trace plots, histograms and autocorrelation functions for β , ϕ , σ_η in the estimation of the SV model with RM-HMC.**

Note: This figure shows trace plots, histograms and autocorrelation functions for the estimation of the SV model with data generating process described in Table 2.10.

Table 2.10: **Posterior results for estimation of the SV model with RM-HMC.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	<i>95% HPD_l</i>	<i>95% HPD_u</i>	<i>True</i>
β	0.85071	0.85064	0.82888	0.79927	0.90418	0.90253
ϕ	0.79293	0.80722	0.73992	0.62454	0.92638	0.96662
σ	0.22291	0.21837	0.23163	0.14751	0.30825	0.07558

Note: This table shows posterior results for the parameters β , ϕ , σ . The scale parameter, β , it captures quite well and lies inside the highest posterior density interval. Persistence parameter, ϕ , is underestimated and variance of the latent volatility process, σ , is overestimated.

 Figure 2.19: **Trace plots, histograms and autocorrelation functions for β , ϕ , σ_η in the estimation of the SV model with RM-HMC.**

Note: This figure shows trace plots, histograms and autocorrelation functions for the estimation of the SV model with data generating process described in Table 2.11.

Table 2.11: **Posterior results for estimation of the SV model with RM-HMC.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	<i>95% HPD_l</i>	<i>95% HPD_u</i>	<i>True</i>
β	0.94395	0.94366	0.90574	0.89572	0.99246	0.93837
ϕ	0.67333	0.68986	0.43245	0.42649	0.88097	0.76405
σ	0.19759	0.19327	0.18142	0.13385	0.27098	0.08030

Note: This table shows posterior results for the parameters β , ϕ , σ . The scale parameter, β , it captures quite well and lies inside the highest posterior density interval. Persistence parameter, ϕ , is captures inside the highest posterior density interval, the variance of the latent process, σ , is overestimated.

Figure 2.20: **Trace plots, histograms and autocorrelation functions for β , ϕ , σ_η in the estimation of the SV model with RM-HMC.**

Note: This figure shows trace plots, histograms and autocorrelation functions for the estimation of the SV model with data generating process described in Table 2.12.

Table 2.12: **Posterior results for estimation of the SV model with RM-HMC.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	<i>95% HPD_l</i>	<i>95% HPD_u</i>	<i>True</i>
β	0.90560	0.90262	0.80476	0.74840	1.07499	0.94224
ϕ	0.96391	0.96513	0.96149	0.93865	0.98672	0.98073
σ	0.17588	0.17391	0.15766	0.13452	0.22005	0.10011

Note: This table shows posterior results for the parameters β , ϕ , σ . The scale parameter, β , it captures quite well and lies inside the highest posterior density interval. Persistence parameter, ϕ , is captures inside the highest posterior density interval, the variance of the latent process, σ , is overestimated.

 Figure 2.21: **Trace plots, histograms and autocorrelation functions for β , ϕ , σ_η in the estimation of the SV model with RM-HMC.**

Note: This figure shows trace plots, histograms and autocorrelation functions for the estimation of the SV model with data generating process described in Table 2.13.

Table 2.13: **Posterior results for estimation of the SV model with RM-HMC.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	95% <i>HPD_l</i>	95% <i>HPD_u</i>	<i>True</i>
β	0.56805	0.56784	0.57215	0.53372	0.60258	0.55326
ϕ	0.77895	0.79364	0.56565	0.60134	0.92411	0.84040
σ	0.22905	0.22424	0.21974	0.14799	0.31755	0.21475

Note: This table shows posterior results for the parameters β , ϕ , σ . The scale parameter, β , it captures quite well and lies inside the highest posterior density interval. Persistence parameter, ϕ , is captures inside the highest posterior density interval, the variance of the latent process, σ , is also captures inside the highest posterior density interval.

2.6.9 Overall comments on HMC

Hamiltonian Monte Carlo is a very attractive method for high-dimensional problems as it allows to explore the parameter space efficiently. In particular, the gain in the efficiency comes from avoiding random walk behaviour in the proposals. However, HMC in its different variations still requires careful calibration. The choice of the leapfrog stepsize and the number of leapfrog steps is essential for the performance of the algorithm. In our simulation examples we noticed that usually the scale parameter β is captured well. However, that is not always the case for the persistence parameter ϕ and variance of the latent process σ . When the true variance is relatively small, the method tends to overestimate it. Overall, it is clearly a promising alternative to off-shelf Markov Chain Monte Carlo methods.

2.7 Approximate Methods

Nowadays researches often deal with complex problems, in Bayesian paradigm it often means that the posterior distribution of interest has a complex form and associated quantities (such as expectations with respect to this distribution) in statistical inference are analytically intractable. Many standard methods in Bayesian inference become either extremely costly or even infeasible. In this case, one can resort to approximate inference. A general

overview on approximate inference, and in particular on variational inference, is given in Bishop (2006). Further in this section we will discuss two approximate methods with application to stochastic volatility models: fixed-form variational Bayes and integrated nested Laplace approximation.

2.7.1 Integrated Nested Laplace Approximation (INLA)

Integrated Nested Laplace Approximation was introduced in Rue et al. (2009). The method is based on the nested version of the classical Laplace approximation and was introduced for latent Gaussian models (LGMs). It became a popular approach in Bayesian inference due to its good performance in the variety of models that belong to the class of LGMs and its computational advantages over other methods in Bayesian literature. Computational appeal of this method comes from the possibility of exploiting sparse matrix computations when evaluating certain approximations. INLA has found its applications in many fields which tend to have high-dimensional problems. Stochastic volatility models have been analyzed using INLA in Martino et al. (2011) and Ehlers and Zevallos (2015). Bivariate stochastic volatility model has been considered in Martino (2007) where the authors presented and tried to solve some of the issues that arise when INLA is used in multivariate case of the model. We further will discuss the details of the method and the shortcomings one must solve in multivariate implementation. We also present the reader with a small simulation study that illustrates the performance of discussed approach. Stochastic volatility model written in the form of LGMs reads

$$y|h, \theta_1 \sim \prod_{i \in \mathcal{J}} \pi(y_i|h_i, \theta_1), \quad (2.103)$$

$$h|\theta_2 \sim N(\mu(\theta_2), Q^{-1}(\theta_2)). \quad (2.104)$$

As before, y_t is the data that we observe and h_t is the latent volatility process, and we are interested in the posterior distribution of the parameters of the model θ and the latent process given the data

$$p(h, \theta | y) \propto p(\theta) p(h | \theta) \prod_{t=1}^T p(y | h_t, \theta). \quad (2.105)$$

The outline of the INLA approach can be summarized in the following steps (see for more details Rue et al., 2009; Martino et al., 2011)

1. Build an approximation $p(\theta | y)$
2. Build an approximation to $p(h_t | \theta, y)$
3. Compute an approximation to $p(h_t | y)$ using the approximations from steps 1 and 2.

The first approximation $p(\theta | y)$ relies on the Gaussian approximation of the form

$$p(x | y, \theta) \propto \exp \left\{ -\frac{1}{2} x^T Q x + \sum g_t(h_t) \right\}, \quad (2.106)$$

where $x = (\mu, h)$, $g_t(h_t) = \log p(y_t | h_t, \theta)$. By matching the mode and curvature in the mode we obtain the Gaussian approximation for our model

$$\tilde{p}_G(x | y, \theta) = K_1 \exp \left\{ -\frac{1}{2} (x - \mu)^T (Q + \text{diag}(c)) (x - m) \right\}, \quad (2.107)$$

where K_1 is normalizing constant, m is the modal value of $p(x | y, \theta)$ and the vector c is the second order terms in the Taylor expansion of $\sum g_t(h_t)$ at the modal value m . Q is the precision matrix that has the form

$$Q = \begin{bmatrix} 1 & -\phi & & & & \\ -\phi & 1 + \phi^2 & -\phi & & & \\ & \ddots & \ddots & \ddots & & \\ & & -\phi & 1 + \phi^2 & -\phi & \\ & & & -\phi & 1 & \end{bmatrix} \quad (2.108)$$

The sparsity of the precision matrix above allows us to exploit efficient sparse matrix computational methods and thus gain computational speed. Note, that in multivariate case this advantage disappears since the matrix Q is not sparse any more.

When it comes to the estimation of stochastic volatility models, approximation of the marginals $p(h_t|\theta, y)$ is always the most challenging task. The solution which is proposed in Rue et al. (2009) is (simplified) Laplace approximation of the form

$$\log \tilde{p}_{SLA}(x_t|\theta, y) = \text{const} - \frac{1}{2}x_t^2 + \gamma_t^{(1)}(\theta)x_t + \frac{1}{6}x_t^3\gamma_t^{(3)}(\theta) + \dots, \quad (2.109)$$

where $\gamma_t^{(1)}$ and $\gamma_t^{(3)}$ are the terms in the Taylor expansion. The final step of the method is to approximate $p(x_t|y)$ with the numerical integration scheme

$$\tilde{p}(x_t|y) = \sum_k \tilde{p}(x_t|\theta^k, y) \tilde{p}(\theta^k|y) \Delta_k, \quad (2.110)$$

for some θ^k of θ , where θ^k is selected by creating a grid of points that covers the area of high density for $\tilde{p}(\theta|y)$. For more details on implementation of the simplified Laplace approximation and the selection of grid of points for θ^k see Rue et al. (2009) and Martino et al. (2011).

We replicate a brief Monte Carlo study for estimation of the stochastic volatility model with the following parametrization

$$y_t = \exp(h_t/2)\epsilon_t \quad (2.111)$$

$$h_t = \mu + \phi h_{t-1} + \eta_t. \quad (2.112)$$

We extend simulation study provided in Ehlers and Zevallos (2015) by analyzing data generating processes with some different values of μ and σ_η , in particular it is of interest when the values of σ_η are relatively small since that is one of the cases where some of the estimation methods fail and this case has not been studied with respect to INLA. Table 2.14 presents the results of this simulation study.

Table 2.14: **Results for the simulation study with INLA when the model is parametrized with scale parameter μ .**

μ	ϕ	σ_η	bias (μ)	smse(μ)	bias(ϕ)	smse(ϕ)	bias(σ_η)	smse(σ_η)
0.136	0.90	0.018	-0.067	0.018	-0.613	0.541	0.291	0.091
-0.214	0.90	0.036	-0.056	0.009	-0.581	0.495	0.273	0.081
-0.068	0.90	0.063	-0.066	0.015	-0.582	0.517	0.249	0.071
-0.028	0.95	0.018	-0.067	0.015	-0.635	0.555	0.288	0.090
0.021	0.95	0.036	-0.070	0.021	-0.579	0.484	0.267	0.079
-0.063	0.95	0.063	-0.093	0.033	-0.414	0.325	0.212	0.059
-0.017	0.98	0.018	-0.076	0.022	-0.617	0.545	0.278	0.087
0.133	0.98	0.036	-0.153	0.079	-0.433	0.365	0.222	0.065
0.058	0.98	0.063	-0.259	0.137	-0.209	0.171	0.128	0.034

Note: This table shows results for one of the simulation studies for the estimation of stochastic volatility models with INLA. The scale parameter μ is estimated quite well, while persistence parameter ϕ is underestimated and the variance of the latent process σ_η is overestimated.

Our findings are comparable to Ehlers and Zevallos (2015), the mean of the volatility process and the persistence parameter are estimated quite accurately, while for the variance of the latent volatility process INLA produces quite some bias, usually it is overestimated. Further, Table 2.15 presents a similar simulation study for the model that is parametrized with the scale parameter β in the observation equation instead of μ in the latent volatility process. The results are similar in nature. We observe quite a lot of bias in the estimation of the variance of the latent volatility process.

Table 2.15: **Results for the simulation study with INLA when the model is parametrized with scale parameter β .**

β	ϕ	σ_η	bias(β)	smse(β)	bias(ϕ)	smse(ϕ)	bias(σ_η)	smse(σ_η)
0.367	0.965	0.134	-0.229	0.052	-0.008	0.000	0.472	0.223
1.188	0.959	0.088	0.200	0.040	-0.191	0.036	0.346	0.120
1.937	0.897	0.433	1.616	2.612	-0.002	0.000	0.592	0.351
0.902	0.966	0.075	-0.188	0.035	-0.305	0.093	0.315	0.099
0.938	0.764	0.080	-0.051	0.002	-0.492	0.242	0.146	0.021
1.435	0.793	0.048	0.601	0.361	-0.387	0.149	0.215	0.046
1.588	0.919	0.275	0.253	0.064	0.020	0.001	0.478	0.229
0.092	0.857	0.109	-0.084	0.007	-0.749	0.561	0.134	0.018
0.942	0.981	0.100	-0.324	0.105	0.019	0.000	95.838	9185
0.553	0.841	0.214	-0.238	0.056	-0.390	0.152	0.192	0.037

Note: This table shows results for one of the simulation studies for the estimation of stochastic volatility models with INLA. The results for the scale parameter β are ambiguous, persistence parameter ϕ is sometimes underestimated, and the variance of the latent process σ_η is consistently overestimated.

2.7.2 Fixed-Form Variational Bayes

As before, assume we observe a process $\{y_t\}_{t=1}^T$ which is driven by an unobservable or latent process $\{h_t\}_{t=1}^T$. Recall the Bayes' rule gives us the posterior distribution of the parameters of the system

$$p(h|y) \propto p(y|h)p(h). \quad (2.113)$$

In Bayesian framework we formulate our prior beliefs which we update once we acquire more data. In general, Variational Bayes methods focus on approximating the posterior distribution $p(h | y)$ with some distribution $q(h|y)$. Further, it is common to choose blocks of the parameters and impose independence for these blocks

$$p(h|y) \approx q(h|y) = q(h_1|y)q(h_2|y). \quad (2.114)$$

By construction the posterior of the blocks of the parameters is independent. In the literature this is referred to as *the mean-field assumption*.

To find the optimal approximation we minimize the Kullback-Leibler (KL) divergence from $q(h|y)$ to $p(h|y)$

$$\tilde{p}(h|y) = \arg \min_{q(h_1|\cdot)q(h_2|\cdot)} KL(q(h_1|y)q(h_2|y) || p(h|y)). \quad (2.115)$$

Distributional approximation can be viewed as an optimization problem, i.e. an optimal distribution has to be chosen from the space of all possible distributions, and the KL divergence is chosen as a loss function (Šmídl and Quinn, 2006). Salimans et al. (2013) propose a specific approach to the minimization problem of KL divergence which is based on the similarities between the optimal solution to the problem and linear regression. The general idea of the approach summarizes in the following steps:

- initialize all the posterior approximations $q(\theta)$;
- iterate over the parameters updating every one of them given the others;
- repeat until convergence.

Consider the stochastic volatility model

$$y_t = \beta \exp(h_t/2) \epsilon_t, \quad (2.116)$$

$$h_{t+1} = \phi h_t + \eta_{t+1}, \quad (2.117)$$

with $h_1 \sim N(0, \sigma^2/(1 - \phi^2))$ and $\epsilon_t \sim N(0, 1)$, $\eta_t \sim N(0, \sigma_\eta^2)$.

We specify our a priory beliefs in the following manner

$$p(\beta) \propto \beta^{-1} \quad (\phi + 1)/2 \sim \text{Beta}(20, 1.5), \quad \sigma^2 \sim \text{IG}(5, 0.25). \quad (2.118)$$

Application of Variational Bayes method requires specification of the posterior approximations $q(\theta)$. It is convenient to assume hierarchical structure

of the prior, in that case it factorizes to

$$p(\phi, \sigma^2, \beta, f) = p(\phi)p(\sigma^2)p(f|\phi, \sigma^2)p(y|f), \quad (2.119)$$

where $f = (\log(\beta), h')$. The hierarchical structure of the prior leads to the following factorization of the posterior approximation

$$\begin{aligned} q_\xi(\sigma_\eta^2, f|f) &= q_\xi(\sigma_\eta^2)q_\xi(f|\phi, \sigma^2) = \\ &= \frac{q_\xi(\sigma_\eta^2|\phi)p(f|\phi, \sigma^2)q_\xi(y|f)}{q_\xi(y|\phi, \sigma^2)}. \end{aligned} \quad (2.120)$$

Thus, the posterior approximations can be chosen as follows

$$q_\xi((\phi + 1)/2) = \text{Beta}(\xi_1, \xi_2), \quad (2.121)$$

$$q_\xi(\sigma^2|\phi) \sim IG(\xi_3, \xi_4 + \xi_5\phi^2), \quad (2.122)$$

$$q(\log(\beta), h|\phi, \sigma^2) = N(m, V), \quad (2.123)$$

where

$$V^{-1} = P(\phi, \sigma^2) + \xi_6, \quad m = V^{-1}\xi_7,$$

with $P(\phi, \sigma^2)$ precision matrix of $p(\log(\beta), h|\phi, \sigma^2)$.

Once the posterior approximations are initialized, we proceed with the next step and iterate over the parameters. The parameters are updated in blocks that correspond to the factorization of the posterior approximations. First, we update the block for the persistence parameter in the latent process $q_\xi(\phi)$

$$\phi^* = s_1(\xi, z_1^*), \text{ with } s_1(\cdot) \text{ and } z_1^* \text{ such that } \sigma^{2*} \sim q_\xi(\sigma^2|\phi^*) \quad (2.124)$$

$$\sigma^{2*} = s_2(\xi, z_2^*, \phi^*), \text{ with } s_2(\cdot) \text{ and } z_2^* \text{ such that } \sigma^{2*} \sim q_\xi(\sigma^2 | \phi^*) \quad (2.125)$$

$$\hat{C}_1 = \nabla_\xi[s_1(\xi, z_1^*)]\nabla_\phi[T_1(\phi^*)] \quad (2.126)$$

$$\hat{g}_1 \approx \nabla_\xi[s_1(\xi, z_1^*)]\{\nabla_\phi[\log p(\phi^*) + \log q_\xi(y|\phi^*, \sigma^{2*}) - \log q_\xi(\sigma^{2*}|\phi^*)]\}, \quad (2.127)$$

Second, we update the block for the variance of the latent process $q_\xi(\sigma^2|\phi)$

$$\hat{C}_2 = \nabla_\xi[s_2(\xi, z_2^*, \phi^*)]\nabla_{\sigma^2}[T_2(\sigma^{2*})] \quad (2.128)$$

$$\hat{g}_2 \approx \nabla_\xi[s_2(\xi, z_2^*, \phi^*)]\nabla_{\sigma^2}[\log p(\sigma^{2*}) + \log q_\xi(y|\phi^*, \sigma^{2*})], \quad (2.129)$$

$T_2(\sigma^{2*})$ are the sufficient statistics of $q_\xi(\sigma^2|\phi)$. The last update is the update of the likelihood approximation

$$a_{t+1} = (1 - \omega)a_t + \omega\mathbb{E}_{q_\xi(f|\phi^*, \sigma^{2*})}[\nabla_f \log p(y|f)] \quad (2.130)$$

$$z_{t+1} = (1 - \omega)z_t + \omega\mathbb{E}_{q_\xi(f|\phi^*, \sigma^{2*})}[f] \quad (2.131)$$

$$\xi_{6,t+1} = (1 - \omega)\xi_{6,t} - \omega\mathbb{E}_{q_\xi(f|\phi^*, \sigma^{2*})}[\nabla_f \nabla_f \log p(y|f)] \quad (2.132)$$

$$\xi_{7,t+1} = a_{t+1} + \xi_{6,t+1}z_{t+1}. \quad (2.133)$$

For more extensive derivations of the updates we refer the reader to Salimans et al. (2013). Further, one might wonder how is the latent process is estimated in this procedure. Salimans et al. (2013) propose to use Kalman filter to estimate the filtering distribution. Even though it is a valid approach that is also is undertaken in Quasi-maximum likelihood method, it weakness lies in the linearization of the observation equation which implies that the distribution of the noise process is not Gaussian anymore. Even in univariate case this can lead to troublesome situations, one would expect the method perform worse in multivariate case.

Table 2.16: **Estimated parameters of the approximate posterior distribution**

TS	ξ_1	ξ_2	ξ_3	ξ_4	ξ_5
1	31.134	9.357	19.185	0.465	-0.205
2	28.272	5.203	12.665	0.451	-0.181
3	30.335	7.103	13.964	0.456	-0.203
4	38.815	2.151	17.903	0.756	-0.429
5	38.815	2.151	17.903	0.756	-0.429

Note: This table shows parameters of the approximate posterior distribution estimated with fixed-form variational Bayes.

Table 2.17: **True parameters of the model**

TS	β	ϕ	σ_η
1	1.189	0.959	0.089
2	0.903	0.967	0.076
3	0.938	0.764	0.080
4	0.942	0.981	0.100
5	0.553	0.840	0.215

Note: This table shows ground truth that we have assumed for the model.

2 ADVANCES IN BAYESIAN COMPUTATION WITH APPLICATION TO STOCHASTIC VOLATILITY MODELS

Figure 2.22: **Fixed-form Variational Bayes Approximation versus HMC.**

Note: This figure illustrates performance of Fixed-form VB and HMC for three different data generating processes. Every line represents a different data generating process. The true parameters are as follows: (a) $\beta = 1.189$, (b) $\phi = 0.959$, (c) $\sigma = 0.089$; (d) $\beta = 0.903$, $\phi = 0.967$, $\sigma = 0.076$; (g) $\beta = 0.938$, (h) $\phi = 0.764$, (i) $\sigma = 0.080$.

Figure 2.23: **Fixed-form Variational Bayes Approximation versus HMC.**

Note: This figure illustrates performance of Fixed-form VB and HMC for three different data generating processes. Every line represents a different data generating process. The true parameters are as follows: (a) $\beta = 0.942$, (b) $\phi = 0.981$, (c) $\sigma = 0.1$; (d) $\beta = 0.553$, $\phi = 0.840$, $\sigma = 0.215$.

2.7.3 Overall comments on Approximate Methods

As one can see from the illustrative example, in some case variational Bayes performs quite well, however there are also cases when the estimates obtained with VB are far off from the ground truth. The challenge with stochastic volatility models remains the same: it is difficult to estimate the latent states. In the approach of Salimans et al. (2013) this is done via Kalman filtering. So the drawback of linearization of the model remains and we expect this linearization what results in poor performance in some of the simulation examples. In this respect, the possible combination of VB and SMC can be of interest. Note, that VB is also not very flexible. By that we mean that with any new distributional specification analytical derivations have to be adjusted and re-evaluated.

Integrated Nested Laplace Approximation is another approach that works well considering how fast the method is. However, the sparse matrix computations that are used in univariate models are not applicable to multivariate case since the precision matrix in Equation(2.108) is not sparse any more. Thus, the method loses its computational attractiveness.

An approach that we have not considered in this paper is Expectation propagation algorithm. In particular, Cseke and Heskes (2011) propose a way to improve approximate marginals $p(x_t|\theta, y)$ in latent Gaussian fields by using EP. The motivation of the approach builds on the fact that EP can give better approximations than Laplace approximation in this case. The improvements, however, would come at computational costs. In univariate case the extra computational costs do not play significant role as the algorithm can be parallelized. However, it is hard to say how big the difference would be in multivariate case both in terms of improvement in the estimation and loss in computational speed.

2.8 Discussion

In this paper, we reviewed multiple methods for the estimation of nonlinear state-space models and stochastic volatility models in particular that

appeared in Bayesian statistics and machine learning in recent years. State-space models can be powerful tools for modeling latent variables in different scientific fields. However, already univariately they are challenging to estimate. The main aim of this paper was to review and understand the existing methods of estimation and define the direction one can undertake for the estimation of *multivariate* nonlinear state-space models. The challenge rises from both statistical and computational perspectives. By that we mean it is hard to develop methods that both provide sufficiently good results from the estimation point of view and are computationally feasible. We have reviewed a number of methods that allow a trade-off between these two aspects. In particular, we have considered particle Markov Chain Monte Carlo and reviewed multiple particle filtering approaches for this method, Riemann Manifold Langevin Hamiltonian Monte Carlo, Integrated Nested Laplace Approximation, Variational Bayes. All these methods are equipped with the ability to estimate models with intractable likelihoods. While reviewing these methods, performing comparison studies and having the goal in mind of multivariate estimation of this model we have come to the following conclusions.

First, particle Markov Chain Monte Carlo methods are the most flexible methods that are also easy in implementation. Unfortunately, intuitiveness of the methods does not imply fast computations. Additional difficulty (besides computational complexity) one faces in the high-dimensional model is a slow exploration of the parameter space. This problem might be partially solved by introducing mixture proposal distributions which with a certain probability make either big or small proposal step. While this problem comes from the MCMC part of the estimation procedure, the problem with the variance of the estimated likelihood rises from the particle filtering side. It is documented that the variance of the estimated likelihood grows exponentially with growing dimension Chopin et al. (2004). As we showed in this chapter, using look-ahead approached such as auxiliary particle filter can lead to significant improvements (even though not in every single case). However, one can take a step further and use even more advanced methods that take into account not only y_t in the proposal distribution of the par-

ticles, but all available observations $y_{t:T}$. Despite this being an appealing solution theoretically, in practice, one must find an efficient implementation of this method. For example, iterated auxiliary particle filter that aims to solve this problem involves as expensive optimization procedure, especially when it comes to multivariate models with complex dynamics in the latent process. This problem can be potentially solved by applying approximate techniques to this optimization problem.

Another promising approach we have considered in this review is Riemann Manifold Hamiltonian Monte Carlo. This approach exploits geometric information about the posterior distribution which leads to very good mixing of the Markov Chains. The tuning of the algorithm in the multivariate case might be challenging, however, so is tuning of particle MCMC methods.

Further, two approximate methods have been reviewed. We ran simulation studies for Fixed-form Variational Bayes and Integrated Nested Laplace Approximation (INLA). Both methods are very fast, but INLA in terms of the computational speed outperforms all the other methods considered (the estimation of the model is done in seconds). The computational efficiency of the algorithm diminishes with the dimension as it relies on the sparse matrix computations, and in particular on the sparseness of the precision matrix in the approximation. Unfortunately, in multivariate case we do not have the sparseness and hence computational costs of the method increase drastically.

Thus, we are still limited in the dimensionality of the model we can estimate. In the next chapter we propose to use stochastic volatility model to estimate the spillover effects on financial markets. In this chapter we use particle Markov Chain Monte Carlo with mixture proposals and auxiliary particle filter for the estimation of the model. We discuss limitations of the method and propose directions for further research regarding extensions of the methods to higher dimensions.

2.A Appendix: Simulated Data

Table 2.18: **Parameter values for time series generated in the Data Set 1.**

TS	β	ϕ	σ_η
1	0.367	0.965	0.134
2	1.189	0.959	0.089
3	1.937	0.898	0.433
4	0.903	0.967	0.076
5	0.938	0.764	0.080
6	1.435	0.793	0.049
7	1.588	0.919	0.275
8	0.092	0.857	0.110
9	0.942	0.981	0.100
10	0.553	0.840	0.215

Note: This table shows true values for the time series generated from stochastic volatility model which were used for comparison of some methods discussed in this paper. This data set in the paper usually is referred to as 'Data Set 1'.

Table 2.19: **Parameter values for time series generated in the Data Set 1.**

TS	μ	ϕ	σ_η	TS	μ	ϕ	σ_η
1	-0.124	1.000	0.009	26	-0.188	1.000	0.010
2	0.443	1.000	0.007	27	0.103	1.000	0.007
3	0.136	1.000	0.014	28	-0.056	1.000	0.009
4	0.265	1.000	0.018	29	-0.418	1.000	0.012
5	-0.320	1.000	0.009	30	0.194	1.000	0.011
6	0.002	1.000	0.007	31	-0.178	1.000	0.008
7	0.077	1.000	0.007	32	-0.115	1.000	0.006
8	0.102	1.000	0.010	33	-0.187	1.000	0.007
9	-0.105	1.000	0.014	34	0.683	1.000	0.018
10	0.158	1.000	0.004	35	-0.082	1.000	0.007
11	-0.153	1.000	0.010	36	0.078	0.859	0.015
12	0.299	1.000	0.012	37	-0.011	1.000	0.027
13	0.090	1.000	0.014	38	0.396	1.000	0.053
14	-0.142	1.000	0.010	39	0.365	1.000	0.010
15	0.096	1.000	0.011	40	0.174	1.000	0.017
16	0.024	1.000	0.032	41	0.174	1.000	0.025
17	-0.407	1.000	0.008	42	-0.188	0.999	0.007
18	-0.165	1.000	0.005	43	-0.430	1.000	0.031
19	-0.264	1.000	0.026	44	-0.139	1.000	0.026
20	0.137	1.000	0.007	45	-0.123	1.000	0.013
21	-0.010	1.000	0.006	46	0.038	1.000	0.014
22	-0.007	1.000	0.013	47	0.140	1.000	0.004
23	-0.122	1.000	0.007	48	-0.260	0.998	0.017
24	-0.071	0.938	0.007	49	0.195	1.000	0.023
25	-0.214	1.000	0.005	50	-0.150	0.987	0.006

Note: These tables show true values for the time series generated from stochastic volatility model which were used for comparison of some methods discussed in this paper. This data set in the paper usually is referred to as 'Data Set 2'.

Chapter 3

Volatility Spillovers with Application to Stochastic Volatility Models

Co-movements in financial time series suggest the presence of volatility spillover effects among financial markets. Understanding fundamentals behind this phenomena is important for portfolio managers and policy makers. Currently in the literature GARCH-type models is the dominating approach for detecting volatility spillovers. The inference is often based on notions of causality in mean and variance. In this paper, we aim to analyze volatility spillovers using a more natural approach for volatility modeling: multivariate stochastic volatility models (MSVM). The structure of MSVM allows to test for causality in volatility processes directly, and in contrast to GARCH models, causality in variance and causality in volatility do not coincide in this framework. However, due to the presence of latent volatility processes estimation of this class of models is a difficult task, it has been shown that standard methods of estimation such as quasi-maximum likelihood and GMM do not perform sufficiently well. We use alternative methods of estimation: particle Markov Chain Monte Carlo (PMCMC). The notion of Granger-causality is used to test for causal links in volatility processes. We further adopt spillover measures used in Vector Autoregres-

sive Framework for measuring the spillover effects in the latent volatility process. Based on the developed methodology for estimation, we discuss further possibilities for model selection and graphical modeling for causal inference in this framework.

3.1 Introduction

Many researchers have documented that financial time series exhibit co-movements, see for example Lee and Kim (1993), Morck et al. (2000), Forbes and Rigobon (2002), Chiang et al. (2007), Corsetti et al. (2005). We observe this kind of behavior since economies of different countries (and regions on country level) are dependent and interrelated with each other through different economic channels. These dependencies can be studied from different perspectives, but in this paper, we will focus on the interconnectedness of financial markets. In particular, we are interested in developing a coherent methodology for measuring the strength and direction of these effects. In the literature synchronicity in the movements of the returns of financial assets is often explained through *volatility spillovers*. Volatility is often assumed to be a phenomenon that drives financial returns and researchers define it differently depending on a particular model specification. In generalized autoregressive conditional heteroskedasticity (GARCH) models Bollerslev (1986) it is defined as the conditional variance of the stock returns and in the alternative model specification – stochastic (latent) volatility models Andersen and Benzoni (2014) – it is an independent latent stochastic process. Note, that strictly speaking GARCH models are also stochastic volatility (SV) models, however, following the general trend in the literature when we talk about SV models we mean the latter. Volatility spillovers are often believed to increase during crisis periods, see for example Diebold and Yilmaz (2012). It is also straightforward to intuitively understand the phenomena in this context. As a toy example imagine that the economies of Australia and New Zealand are closely related, and moreover, the economy of Australia has strong effects on the economy of New Zealand. In this case, one would expect that if a crisis

happens in Australia than it would also transmit to New Zealand. Knowing 'how much' of the shock would transmit and for how long the effect would last can help the policymakers and portfolio managers in their decision regarding creating bailout funds and portfolio diversification. Thus, it is natural to assume that volatility spillovers can happen both contemporaneously and with lagged effect, correspondingly through correlation and Granger-causality. In GARCH-type models spillover effects are defined through Granger-causality in variance, see for example Hafner and Herwartz (2008). However, inference, in this case, involves a battery of tests that are not intuitive and easily interpretable. The clear advantage of the GARCH-type models is the easiness of estimation. However, the disadvantage is lack of intuitive interpretability. Stochastic (latent) volatility model can provide an intuitive framework for measuring volatility spillover effects. In its multivariate specification, the latent volatility process has $\text{VAR}(p)$ form. Hence, one can extend the interpretability of the $\text{VAR}(p)$ model for identifying and measuring volatility spillover effects. In particular, all Granger-causal relationships are encoded in autoregressive matrices which makes it easy to understand the dependence structure in the model once estimated. Moreover, it allows for high degree of flexibility in the dependence modeling: both positive and negative correlation and both positive and negative Granger-causal feedback are allowed in the model which is not always the case in GARCH-type approaches.

In this paper, we propose a Bayesian framework for measuring volatility spillover effects on financial markets with multivariate stochastic volatility models. We define volatility spillovers through the concept of Granger-causality and further propose measures of the volatility spillover effects in this framework. The measures are constructed for the $\text{VAR}(p)$ part of the model as in Diebold and Yilmaz (2012) and then incorporated into SV approach. We carry out the estimation of the multivariate stochastic volatility model using a combination of auxiliary particle filter Pitt and Shephard (1999) and particle Markov Chain Monte Carlo methods Andrieu et al. (2010). These algorithms present a flexible framework for estimation of the nonlinear state-space model. In multivariate case, the implementa-

tion is not straightforward. Thus we discuss the issues and details of our implementation. We estimate the model in both examples on the simulated data and empirical applications up to dimension $n = 5$. It means that we have five time series of log-returns that are driven by a five-dimensional VAR system of latent volatility processes. In this case we deal with large number of parameters, $3n^2 = 45$. When implementing Metropolis-Hastings algorithm Chib and Greenberg (1995), which is a Markov Chain Monte Carlo algorithm suitable for our model estimation, the exploration of the parameter space would be very slow and hence this would compromise the convergence of the algorithm. To overcome this issue, we use a mixture proposal which allows us to explore small local neighborhoods sufficiently well, but also make large proposals steps. This is not the only problem that has to be tackled from the estimation point of view, however, it is one of the most important to estimate the model in higher dimensions.

It is also important to mention, that there exist approaches for measuring spillover effects with realized volatility. Including intraday financial data into a model usually successfully works in forecasting of financial time series Chiriac and Voev (2011). At this point, however, our focus is not to perform forecasting. We are instead interested in developing a flexible framework for detecting and measuring fundamental relationships between financial markets. Thus, since forecasting is not the goal of our exercise realized volatility might not benefit our purpose. It is rather a real-world phenomenon than a modeling approach, and since it is known to be quite a noisy measure, it might not provide extra information about the dynamic relationships but instead cover them with an extra noise in the model. This, however, is an open question that has to be answered which is outside of the scope of this paper, we leave it for the future. To answer, however, whether realized volatility can give extra insights into spillover analysis one can build upon the framework proposed in this paper. One can add an extra equation for the realized volatility in our model and define it as the volatility plus some noise. This would provide an extra linear equation in the model and might ease the computations. Further, the appropriate measure can be applied. One would have then to compare the results obtained

by estimation of the stochastic volatility model and stochastic volatility together with realized volatility measure. This framework can be of a particular interest to the people who are doing forecasting exercises for the financial markets.

We further proceed with the formulation of the multivariate stochastic volatility models in Section 2 of this Chapter, in Section 3 we define volatility spillover effects through the concept of Granger-causality and discuss the measures that can be used for accessing the strength of these effects, in Section 4 we discuss the estimation of the model, associated challenges and implementation details of the approach, in Section 5 and 6 demonstrate some simulation experiments and empirical applications, both sections consider different dimensions of the model. In simulation examples, we discuss the model in two-, three- and five-dimensional cases. In empirical studies we consider bivariate model for exchange rates between Australia and New-Zealand and five-dimensional model for the US, the UK, Germany, Japan and China.

3.2 Multivariate Stochastic Volatility Model

Stochastic (latent) volatility models belong to the class of nonlinear state-space models and are used for modeling log-returns on financial markets. Log-returns are usually defined as follows. Assume that P_t is the price of the asset at time t . Then, the return at time t is defined through the Equation (3.1).

$$1 + R_t = P_t/P_{t-1}, \quad (3.1)$$

and thereby the log-return is defined through the following expression

$$y_t = \log(1 + R_t) = \log(P_t/P_{t-1}). \quad (3.2)$$

In the model we are considering the log-return defined in Equation (3.2) is modeled through an unobserved independent stochastic process of volatility h_t . In multivariate case, we deal with multiple time series and the

3 VOLATILITY SPILLOVERS WITH APPLICATION TO STOCHASTIC VOLATILITY MODELS

corresponding number of latent volatility processes. The unobserved multivariate volatility process takes the form of vector autoregressive process of order p , $\text{VAR}(p)$, the full model reads

$$y_t = \Omega_t \epsilon_t, \quad (3.3)$$

$$h_t = \mu + \sum_{u=1}^p \Phi^u (h_{t-u} - \mu) + \eta_t, \quad (3.4)$$

with

$$\Omega_t = \begin{pmatrix} \exp(h_{1t}/2) & 0 & \dots & 0 \\ 0 & \exp(h_{2t}/2) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & \exp(h_{nt}/2) \end{pmatrix},$$

$$\Phi_u = \begin{pmatrix} \phi_{11}^u & \phi_{12}^u & \dots & \phi_{1n}^u \\ \phi_{21}^u & \phi_{22}^u & \dots & \phi_{2n}^u \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{n1}^u & \dots & \dots & \phi_{nn}^u \end{pmatrix},$$

$$\Sigma_\epsilon = \begin{pmatrix} 1 & \sigma_{\epsilon_{12}} & \dots & \sigma_{\epsilon_{1n}} \\ \sigma_{\epsilon_{21}} & 1 & \dots & \sigma_{\epsilon_{2n}} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{\epsilon_{n1}} & \dots & \dots & 1 \end{pmatrix}, \quad \Sigma_\eta = \begin{pmatrix} \sigma_{\eta_1}^2 & 0 & \dots & 0 \\ 0 & \sigma_{\eta_2}^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & \sigma_{\eta_n}^2 \end{pmatrix}.$$

where (3.4) is the state equation of the latent process h_t and (3.3) is the observation equation that depends non-linearly on the latent state. Further, $\epsilon_t \stackrel{iid}{\sim} N(0, \Sigma_\epsilon)$ and $\eta \stackrel{iid}{\sim} N(0, \Sigma_\eta)$. Note, that in this model volatility is modeled stochastically (in the GARCH-type models, volatility is deterministic). The unobserved volatility process can be modeled by vector autoregressive

process of order p , in this paper we limit our attention to the case of $p = 1$. Generalization to higher orders of $\text{VAR}(p)$ can be done by incorporating appropriate sparse priors into the model with the off-shelf choice being a Minnesota (Litterman) prior. Other sparse priors such as spike and slab prior or horseshoe prior (George and McCulloch, 1993; Ishwaran and Rao, 2005) can be used depending on a particular inference interest. We describe the application of these priors in the discussion on model selection for stochastic volatility models. Another restriction that we will impose for most of the simulation studies is the diagonal structure of the matrix Σ_η . Generalization to the non-diagonal structure is straightforward, and we discuss it in the section on the estimation. We impose this restriction to ease the computational burden.

3.3 Measuring spillover effects

3.3.1 Granger-causality in volatility

One of the advantages of multivariate stochastic volatility models over the GARCH-type of models is the straightforward interpretation of the volatility spillover effects. While in GARCH framework usually volatility spillovers are defined through causality in variance, in the MSVM classical definition of Granger-causality in multiple time series applies to the latent process of the model which has the form of $\text{VAR}(p)$. Below we first define Granger noncausality in general sense and then we show how this concept can be used in our framework for inferring spillover effects between financial markets.

Assume we have multiple time series processes $V = (X, Y, Z)$. Let us denote the following infomations sets: $\mathcal{I}_t^*$ for X^t, Y^t, Z^t and $\mathcal{I}_{-X,t}^*$ for Y^t, Z^t . The definition of Granger non-causality follows.

Definition 3.1. (Granger non-causality). The series X is Granger non-causal for the series Y with respect to $V = (X, Y, Z)$ if $Y_{t+1} \perp\!\!\!\perp X^t \mid Y^t, Z^t$ for all $t \in \mathbb{Z}$; otherwise we say that X Granger-causes Y with respect to V Eichler (2013).

Further, recall that the latent volatility process can be written in VAR(p) form, unlike in Equation (3.4) here we do not use matrix notation

$$h_{j,t} = \mu_j + \sum_{u=1}^p \sum_{i=1}^n \phi_{j,i}^u (h_{i,t-u} - \mu_i) + \eta_{j,t}. \quad (3.5)$$

Definition 3.1 implies that the volatility h_i does not Granger-cause h_j if $h_{j,t}$ does not depend on h_i^{t-1} given the past of the remaining volatility variables. Thus, by defining the volatility as in Equation (3.5), we say that $h_{j,t}$ does not depend on h_i^{t-1} if and only if $\phi_{j,i}^u = 0$ for all $u > 0$. For the simplicity in the remainder of this chapter we will assume that $p = 1$. In this case Equation (3.5) simplifies to

$$h_{j,t} = \mu_j + \sum_{k=1}^n \phi_{j,i} (h_{i,t-1} - \mu_i) + \eta_{j,t}. \quad (3.6)$$

In this case, we say that there is no volatility spillover from financial market j to i if the condition of Granger non-causality is satisfied, i.e. $\phi_{j,i} = 0$.

Granger-causal structure of the volatility process is specified by matrix Φ and has an intuitive interpretation which makes the approach attractive. Additionally, we do not impose a positive constraint on the individual coefficients of the matrix Φ . Hence our framework provides a flexible tool for also detecting a negative relationship between the financial markets. GARCH approaches usually allow only for the positive feedback, only some papers (Conrad and Karanasos, 2010) have attempted to relax this constraint in GARCH framework.

3.3.2 Graphical representation of the causal structure

Graphical models have been used for causal inference in many fields, Dahlhaus and Eichler (2003) proposed to use graphical model for causal inference in time series data. A model represented by a graph gives an intuitive representation of the underlying causal structure and can be a useful tool for causal inference. In this subsection, we define basic concepts and give an example of a graphical representation of the SV model in bivariate case.

Bivariate stochastic volatility model reads as follows

$$\begin{pmatrix} y_{1,t} \\ y_{2,t} \end{pmatrix} = \begin{pmatrix} \exp(\frac{h_{1,t}}{2}) & 0 \\ 0 & \exp(\frac{h_{2,t}}{2}) \end{pmatrix} \begin{pmatrix} \epsilon_{1,t} \\ \epsilon_{2,t} \end{pmatrix} \quad (3.7)$$

$$\begin{pmatrix} h_{1,t+1} \\ h_{2,t+1} \end{pmatrix} = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} + \begin{pmatrix} \phi_{1,1} & \phi_{1,2} \\ \phi_{2,1} & \phi_{2,2} \end{pmatrix} \begin{pmatrix} h_{1,t} - \mu_1 \\ h_{2,t} - \mu_2 \end{pmatrix} + \begin{pmatrix} \eta_{1,t} \\ \eta_{2,t} \end{pmatrix} \quad (3.8)$$

where

$$\begin{pmatrix} \epsilon_{1,t} \\ \epsilon_{2,t} \end{pmatrix} \sim N(0, \Sigma_\epsilon), \quad \Sigma_\epsilon = \begin{pmatrix} 1 & \rho_{\epsilon_{1,2}} \\ \rho_{\epsilon_{1,2}} & 1 \end{pmatrix}, \quad (3.9)$$

$$\begin{pmatrix} \eta_{1,t} \\ \eta_{2,t} \end{pmatrix} \sim N(0, \Sigma_\eta), \quad \Sigma_\eta = \begin{pmatrix} \sigma_{\eta_{1,1}}^2 & \sigma_{\eta_{1,2}} \\ \sigma_{\eta_{2,1}} & \sigma_{\eta_{2,2}}^2 \end{pmatrix}. \quad (3.10)$$

Ultimately, we are also interested in building a graphical model for the high dimensional system to detect spillover effects. It is, however, the first step for our goal in mind since before one can apply the graphical approach to this problem, one should be able to estimate the model in an efficient way, which means obtaining sufficiently accurate results in feasible computational time. At the moment, the computational complexity of the estimation prevents one from doing so. In particular, if one wants to discover not only the parameter space but also the model space, the problem becomes infeasible computationally (see an example with spike and slab priors on Section 5). The challenge can be addressed from two sides. The first side is the estimation of the model. By combining approximate and exact methods, one could achieve estimation feasibility in higher-dimension. It is, however, is not a trivial problem as we discuss in Chapter 2 of this thesis. The approximate methods, such as Integrated Nested Laplace Approximation or Variational Bayes, lose the advantages they have in the univariate case when we extend the methods to multivariate dimensions. For example, the computational complexity of INLA significantly increases

Figure 3.1: **Graphical representation of a bivariate stochastic volatility model.**

Note: This graph represents dependence structure over time in the bivariate stochastic volatility model described in Equations 3.7 – 3.10. Directed solid edges represent a Granger-causal structure which is encoded in coefficients ϕ_{ij} of the matrix Φ . Dashed undirected edges correspond to the contemporaneous correlation between volatility processes or between log-returns. As log-returns are observed we mark the corresponding vertices with grey color, the vertices corresponding to the latent volatility process are white.

with increased dimensionality. The reason for that is that the sparse matrix computations which make the method computationally fast in the univariate case are not applicable in multivariate extensions as with increased dimension the sparseness is lost. The second way to build graphical models would be to use a combination of the SMC/PMCMC with sparse priors that do not move in the model space, such as horseshoe prior (Carvalho et al. (2009)), as this again would make model selection feasible. We show in Section 5 on a trivariate example that spike and slab priors might work very well, however, the method is too expensive computationally to try it in higher dimensions.

3.3.3 Forecast Error Variance Decomposition

The spillover effects measures are build upon the idea of *forecast error variance decomposition*. These measures were developed for VAR(p) framework and were introduced in Diebold and Yilmaz (2012). As the latent process in stochastic volatility model is defined through VAR(p) process, measures developed in Diebold and Yilmaz (2012) are directly applicable. As we discuss further, we estimate the model in the Bayesian framework. It means we estimated the posterior distributions of the measures discussed in this section and provide summary statistics and highest posterior density intervals for them. The one-step prediction for the volatility process at time T is

$$h_{T+1|T} = \mu + \Phi(h_T - \mu), \quad (3.11)$$

and similarly we can derive the m -steps ahead forecast as

$$h_{T+m|T} = \mu + \Phi(h_{T+m-1} - \mu), \quad (3.12)$$

where $h_{T+k|T} = h_{T+k}$ if $k \leq 0$. Then we can denote the m -step forecast error for a process h_t as follows

$$h_{T+m} - h_{t+m|T} = \sum_{i=0}^{m-1} \Psi_i \eta_{T+m-i}, \quad (3.13)$$

for the forecast that was made based on the information set available at time T , $\mathcal{I}(T)$, and where Ψ_i 's can be obtained recursively from the equation

$$\Psi_i = \sum_{j=1}^i \Psi_{i-k} \Phi_j, \quad \text{for } i = 1, 2, \dots \quad (3.14)$$

Since we consider the case of diagonal variance-covariance matrix of the error terms in volatility process we get the following forecast error variance of the k -th element of h_{T+m} (see Diebold and Yilmaz (2012) on how to deal with non-diagonal matrices)

$$\sigma_k^2(M) = \sum_{j=0}^{M-1} (\psi_{k1,j}^2 + \dots + \psi_{kK,j}^2) = \sum_{j=1}^K (\psi_{kj,0}^2 + \dots + \psi_{kj,M-1}^2), \quad (3.15)$$

with $\psi_{nm,j}$ being (n, m) -th element of Ψ_j . Forecast error variance decomposition helps to identify information flow from one variable to the others in the autoregressive system which is exactly what we are interested in to analyze in the unobserved volatility process.

3.3.4 Variance Shares

The next step for constructing volatility spillover measures is defining the variance shares. Variance shares correspond to the m -step-ahead error variance in forecasting volatility h_i . This error is combined from two sources, first, the shocks that are due to the process h_i itself and, second, due to the shocks that happen in h_j .

$$\theta_{ij}(M) = \frac{\sigma_{\eta ii}^{-1} \sum_{m=0}^{M-1} (e_i' \Psi_m \Sigma_{\eta} e_j)^2}{\sum_{m=0}^{M-1} (e_i' \Psi_m \Sigma_{\eta} \Psi_m' e_i)}, \quad (3.16)$$

where Σ_η is the variance-covariance matrix of the volatility process, $\sigma_{\eta jj}$ is the standard deviation of the volatility error terms, e_i represents the selection vector and $e_i = 1$ for the i -th element and is equal to zero for the rest. Further, the variance shares are normalized for construction of the spillover measures

$$\tilde{\theta}_{ij}(m) = \frac{\theta_{ij}(m)}{\sum_{j=1}^n \theta_{ij}(m)}, \quad (3.17)$$

so that $\sum_{j=1}^n \tilde{\theta}_{ij}(m) = 1$ and $\sum_{i,j=1}^n \tilde{\theta}_{ij}(m) = n$.

3.3.5 Total Spillover Effect

Individual variance shares can be used to measure the total spillover effect in the system. As the name suggests, it tells us about how much of spillovers in total happen in the system.

$$S(M) = \frac{\sum_{i,j=1, i \neq j}^n \tilde{\theta}_{ij}(M)}{\sum_{i,j=1}^n \tilde{\theta}_{ij}(M)} = \frac{\sum_{i,j=1, i \neq j}^n \tilde{\theta}_{ij}(M)}{n}. \quad (3.18)$$

This measure shows how much the individual volatility spillovers affect the total forecast error variance in the system.

3.3.6 Directional Spillover Effect

A measure of directional spillover effect allows us to infer about the strength of the spillover effects taking into account the direction of the information flow. First, we can define the directional spillover to market/country i from all other markets/countries j

$$S_{i\cdot}(M) = \frac{\sum_{j=1, j \neq i}^n \tilde{\theta}_{ij}(M)}{\sum_{j=1}^n \tilde{\theta}_{ij}(M)}. \quad (3.19)$$

Second, we can also define the directional spillover from market/country i to all other markets/countries j

$$S_{\cdot i}(M) = \frac{\sum_{j=1, j \neq i}^n \tilde{\theta}_{ji}(M)}{\sum_{j=1}^n \tilde{\theta}_{ji}(M)}. \quad (3.20)$$

The former measure summarizes information about how much of the volatility was received by a certain market/country, the latter tells us about how much of the volatility that market/country transmitted to the others.

3.4 Estimation

The structure of stochastic volatility model exhibits non-linearity and latent states which complicate estimation, in particular, likelihood function of the model is analytically intractable. Standard techniques such as GMM and QML (Nelson, 1988; Harvey et al., 1994) estimation do not provide satisfactory results, in particular in small samples. So we resort to Bayesian computation methods which often are computationally expensive for this class of models, but at the same time most promising in this framework. In this section, we discuss how *particle filtering* can be used for estimation of the likelihood and then we proceed with describing Markov Chain Monte Carlo algorithms that we use for making draws from the posterior distribution of the parameters of the model.

3.4.1 Outline of the estimation

We proceed with the estimation of the model in the Bayesian framework. Thus, it is useful to rewrite the model in the dynamic Bayesian form. It reads

$$y_t \sim p(y_t | h_{0:t}, \theta, y_{1:t-1}) = N(0, \Omega \Sigma_\epsilon \Omega) \quad (3.21)$$

$$h_t \sim p(h_t, \theta | h_{0:t-1}, \theta, y_{1:t-1}) = N(\mu + \Phi h_{t-1}, \Sigma_\eta) \quad (3.22)$$

$$h_0 \sim \pi(h_0|\theta) \quad (3.23)$$

$$\theta \sim \pi(\theta) \quad (3.24)$$

where (3.21) is the observation (measurement) equation, (3.22) is the state dynamics equation and (3.23) and (3.24) are prior distributions. Figure 3.2 illustrates dynamics of the model in univariate case. At every time step volatility process h_t , which is given in state equation drives asset log-return of the same period y_t which is given in the observation equation. The challenge of the estimation is in filtering out unobserved volatility process from the noisy observations. Since the dependence between the states and observations is nonlinear Kalman filter is not directly applicable, but the problem can be tackled with particle filtering. We further discuss how particle filters are applied in the Bayesian framework and provide details of the algorithm that we use to estimate the likelihood of the model.

The Bayes' rule is the foundation of the estimation procedure

Figure 3.2: **Graphical representation of a stochastic volatility model, where h is a latent volatility process and y is observable log-returns.**

Note: At every time point t the observation y_t depends only on the current state of the latent process h_t . For nonlinear state-space models this dependence is nonlinear.

$$p(h_{0:T}|y_{1:T}) = \frac{p(y_{1:T}|h_{0:T})\pi(h_{0:T})}{p(y_{1:T})}, \quad (3.25)$$

where $\pi(h_{0:T})$ is the prior distribution on the parameters of the volatility process, $p(y_{1:T}|h_{0:T})$ is the likelihood of the observations, $p(y_{1:T})$ is the normalization constant defined as follows

$$p(y_{1:T}) = \int p(y_{1:T}|h_{0:T})p(h_{0:T})dh_{0:T}. \quad (3.26)$$

When the normalizing constant is omitted, Bayes' rule can be formulated in terms of proportionality

$$p(h_{0:T}|y_{1:T}) \propto p(y_{1:T}|h_{0:T})\pi(h_{0:T}) \quad (3.27)$$

Hence, here our goal is to estimate the joint posterior by combining the likelihood with our prior beliefs. The main difficulty that researchers face when estimating nonlinear state space models is intractability of the integral when evaluating the likelihood of the data. In our case, it is the following integral

$$p(y_{1:T}) = \int p(y_{1:T}|h_{0:T})\pi(h_{0:T})dh_{0:T}. \quad (3.28)$$

Dynamic estimation of the model presumes that we need to recompute posterior distribution every time we get a new observation and that eventually leads to the intractability of the integral. As it is not possible to solve the integral in closed form, we need to use numerical methods of for its evaluation and learn about parameters sequentially. Sequential Monte Carlo methods (or – particle filtering) provide us with a solution to this problem. Particle filtering for nonlinear state space models was proposed in Gordon et al. (1993) and Kitagawa (1996). The idea of particle filters is that we estimate the filtering density by an average of particle samples, namely $p(h_t, \theta|y_{1:t})$ is approximated in the following manner

$$p_N(h_t, \theta|Y_{1:t}) = \sum_{i=1}^N \omega_t^{(i)} \delta((h_t^{(i)}, \theta))(h_t, \theta), \quad (3.29)$$

Where $\delta(h_t^{(i)}, \theta)$ is Dirac delta function with the mass centered in $h_t^{(i)}$, In the above equation, N is the number of particles that we use for approximation, and $\omega_t^{(i)}$ is a weight associated with a particular particle. The weights tell us about the difference between the actual target and the approximation, in other words how well the sampled particles can explain the observed data. Particle filters exploit Bayesian recursion equations, and the general outline of a particle filter consists of four steps: initialization, prediction, updating, and resampling (Gall et al. (2007)). These steps are usually present with some degree of variation in all particle filter algorithms.

$$\begin{aligned}
 p^N(h_{t+1}|y_{t+1}) &\propto \underbrace{\int p(y_{t+1}|h_{t+1}) p(h_{t+1}|h_t) p^N(h_t|y_t) dh_t}_{\text{Update}} \underbrace{\quad}_{\text{Predict}} \\
 &\propto \sum_{i=1}^N p(y_{t+1}|h_{t+1}) p(h_{t+1}|h_t^{(i)}) \omega_t^{(i)}
 \end{aligned} \tag{3.30}$$

$p^N(h_{t+1}|y_{t+1})$ is a finite mixture distribution and different sampling methods generate alternative particle filtering algorithms.

- Initialization: given the prior distribution $\pi(\theta_0)$ we draw N independent random samples $\{h_i^0\}_{i=1}^N$ from it, these samples are called particles.
- Prediction: for every particle we make a prediction $p(h_t|h_{t-1})$

$$p(h_t|y_{1:t-1}) = \int p(h_t|h_{t-1}) p(h_{t-1}|y_{1:t-1}) dh_{t-1} \tag{3.31}$$

- Updating: During updating we assign a weight to every particle according to the likelihood function of the observation process. Every weight can be interpreted as our "confidence" about this particle

$$p(h_t|y_{1:t}) = \frac{1}{Z_t} p(y_t|h_t) p(h_t|y_{1:t-1}) \tag{3.32}$$

- Resampling: this is selection procedure, the particles with higher weights are selected, and the particles with lower weights are dropped. We resample to obtain draws from $p(h_t|y_t)$.

Particle filters exhibit two well-known problems; both are related to resampling step of the algorithm. The first problem is called *degeneracy* of a particle filter and it happens when resampling step is either omitted or is not performed often enough. Indeed, if we do not implement resampling step often enough after few iterations, only a few particles will have weights that are different from zero. On the other hand, if resampling is performed too often, then another problem arises that is called *sample impoverishment*. In this case, due to frequent resampling particles with smaller weights are removed, and particles with larger weights are duplicated which eventually leads to a sample of N equivalent particles and hence lack of diversity in the particle set. One of the solutions to these two problems is resampling not at every iteration of the algorithm. Resampling is performed when effective number of particles is not sufficient Särkkä (2013)

$$n_{eff} \approx \frac{1}{\sum_{i=1}^N (\omega_t^{(i)})^2}. \quad (3.33)$$

Generally, performing resampling step only when n_{eff} is below a certain threshold can help to find the balance between degeneracy and impoverishment problems. But that is not a sufficient solution invariably.

3.4.2 Simulated Likelihood with ASIR

As we mentioned before, Sequential Monte Carlo methods allow us to deal with the problems when the likelihood is intractable. Using particle filtering techniques, we can approximate the likelihood and then use this approximation in the construction of the Markov Chain for drawing from the posterior distribution. Degeneracy and impoverishment of the particle set can be tackled with optimal resampling schemes. However, in the case of multivariate models, there is another major problem, and that is the variance of the approximate likelihood which grows exponentially with

the dimension of the model Chopin et al. (2004). It results in the failure of standard particle filter algorithm to provide sufficiently good results in the multivariate framework. One of the solutions that have been proposed in the literature is Auxiliary Particle Filter (APF) introduced in Pitt and Shephard (1999). Standard particle filter scheme such as Sequential Importance Resampling (SIR) algorithm on the propagation stage of the particles makes proposal from $p(h_t|h_{t-1})$, APF exploits all information available at time t and makes a proposal from $p(h_t|h_{t-1}, y_t)$. It can be thought of as a blind and non-blind proposal for the particles. In the former case, we do not take into account the observation of the current period to make proposals for the particles. In the latter case, we do take it into account, and that allows us to make better proposals for the particle set. Below we give the general form of Auxiliary SIR algorithm Pitt et al. (2010) that was first introduced in Pitt and Shephard (1999). In Algorithm 14 T is the sample size of time series, and N is the number of particles. Figures 3.3, 3.4, 3.5 demonstrate the difference between SIR and ASIR in the univariate case of stochastic volatility model and Figure 3.6 in the bivariate case. The black line is the true log-volatility and the red one is the volatility filtered out from the observations y_t . The filtering was done conditioning on the true values of the parameters of the model. The difference is particularly visible for the case when the variance of the latent process is relatively small, see Figure 3.5.

In Algorithm 14 we take μ as the initial state h_0 . Further, the resampling method needs to be specified. We use systematic resampling, for an overview of resampling schemes for particle filtering see Douc and Cappé (2005).

ASIR as it will be apparent from our simulation results, works better than standard BPF for a multivariate framework and it also does a good job for a various parameters values (which is not the case for other methods, for example, QML fails to estimate the variance of the volatility process when it is low already in the univariate case). However, this algorithm still can be improved, and we give more details on that in the discussion. We have implemented more advanced algorithms, but since these methods are

Algorithm 14 Simulated Likelihood with ASIR

1: Initialization: draw N samples from the prior

$$h_0^{(i)} \sim p(h_0 \mid \theta), \quad i = 1, \dots, N \quad (3.34)$$

and set $w_0^{(i)} = 1/N$, for all $i = 1, \dots, N$

2: **for** t in $1, \dots, T$ **do**

3: Draw samples from importance distribution

$$h_t^{(i)} \sim g(h_t \mid h_{t-1}^{(i)}, y_{t+1}, \theta) \quad (3.35)$$

4: Compute the weights

$$v_t^{(i)} = \frac{p(y_t \mid h_t^{(i)})p(h_t^{(i)} \mid h_{t-1}^{(i)})}{g(y_t \mid h_{t-1}^{(i)})g(h_t^{(i)} \mid h_{t-1}^{(i)}, y_t)}, \quad (3.36)$$

5: Compute the estimate of $p(y_t \mid y_{1:t-1}, \theta)$

$$\hat{p}(y_t \mid y_{1:t-1}, \theta) = \sum_t w_{t-1}^{(i)} v_t^{(i)}$$

6: Compute normalized weights

$$w_t^{(i)} \propto w_{t-1}^{(i)} v_t^{(i)} \quad (3.37)$$

7: If the effective number of particles is too low, perform resampling.

8: **end for**

Figure 3.3: Estimate of the log-volatility with blind (left) and non-blind (right) proposals for the particles $N = 100$, $\sigma_\eta^2 = 0.5$.

Note: This figure illustrates the filtered volatility process (red) against the true volatility process (black) estimated using particle filter with blind proposals (left) and non-blind proposals (right). In this case the true variance is $\sigma_\eta^2 = 0.5$.

Figure 3.4: Estimate of the log-volatility with blind (left) and non-blind (right) proposals for the particles $N = 100$, $\sigma_\eta^2 = 0.1$.

Note: This figure illustrates the filtered volatility process (red) against the true volatility process (black) estimated using particle filter with blind proposals (left) and non-blind proposals (right). In this case the true variance is $\sigma_\eta^2 = 0.1$.

Figure 3.5: **Estimate of the log-volatility with blind (left) and non-blind(right) proposals for the particles $N = 1000$, $\sigma_\eta^2 = 0.05$.**

Note: This figure illustrates the filtered volatility process (red) against the true volatility process (black) estimated using particle filter with blind proposals (left) and non-blind proposals (right). In this case the true variance is $\sigma_\eta^2 = 0.01$. Clearly, particle filter with blind proposals performs much worse.

Figure 3.6: **Estimate of the log-volatility with blind (left) and non-blind(right) proposals for the bivariate model.**

Note: This figure illustrates the filtered volatility process (red) against the true volatility process (black) estimated using particle filter with blind proposals (left) and non-blind proposals (right) for a bivariate process.

computationally very intensive, the results in this paper are obtained with Algorithm 14.

3.4.3 Sampling from the posterior

An off-shelf solution for sampling from the posterior distribution is the Metropolis-Hastings algorithm, see Algorithm 15 for the pseudo-code and Robert and Casella (1999) for more details and examples. The general idea behind it is that when we have a target distribution, say $p(\cdot)$, we can construct a Markov Chain that is ergodic and stationary with respect to our target distribution which would converge to it. This approach can be in particular applied to sample from the posterior distribution when direct sampling is hard or impossible. In the Algorithm 15, θ is the vector of parameters in our model for which we wish to obtain samples from the posterior distribution and $q(\cdot)$ is proposal distribution. In each iteration of the algorithm, a draw from $q(\cdot)$ is obtained. This draw is denoted θ_t^c and it is either accepted or rejected with probability ρ defined in Algorithm 15. Here it is important to note, that the performance of the algorithm will highly depend on the proposal distribution $q(y|\theta^{(t)})$. The optimal construction of the algorithm should be such that the parameter space is explored sufficiently well. It means that the local neighborhoods should be explored, but also that the chain should be able to move away from the local maxima. Generally, it is advised to have the acceptance rate around 25% (Roberts et al., 1997; Chib and Greenberg, 1995). Further, note that the Algorithm 15 requires the evaluation of the likelihood which in our case is not possible to do analytically. Here, the idea of Andrieu et al. (2010), the method known as particle Markov Chain Monte Carlo (PMCMC), becomes important. We can estimate the likelihood with the Sequential Monte Carlo scheme and use this estimate in the Metropolis-Hastings algorithm. This methodology has been discussed in particular with application to Gibbs sampler and Metropolis-Hastings algorithm, but can also be applied to other samplers of a similar nature. When dealing with multivariate models in particular similar algorithms that allow better

mixing are of interest. In practice performance of the Metropolis-Hastings algorithm is not necessarily satisfactory given the only finite computational time available in real-life applications. The difficulties arise once one has to make proposals in high-dimensional parameter space (Chib et al., 2009). One of the problems is that when dealing with multivariate proposals, it is much harder for the chain to escape the area of the local maxima of the likelihood. This problem can be addressed with mixture proposals. The small world sampler that we present below addresses the issue by mixing long and short-range proposals, i.e., two chains, but generally more than two distributions can be used in the mixture proposals. The idea behind the small-world sampler is in making interchangeably small and big proposals with either predetermined or adaptive rate for them. In that case, we can make sure that in the higher-dimensional model we do not get stuck in the point of local maximum likelihood and explore the parameter space sufficiently well. As the result of PMCMC, we provide the summary of the

Algorithm 15 Metropolis-Hastings Algorithm

- 1: Given $\theta^{(t)}$,
- 2: Generate $\theta_t^c \sim q(y|\theta^{(t)})$,
- 3: Take

$$\theta^{(t+1)} = \begin{cases} \theta_t^c, & \text{with probability } \rho(\theta^{(t)}, \theta_t^c) \\ \theta^{(t)} & \text{with probability } 1 - \rho(\theta^{(t)}, \theta_t^c), \end{cases}$$

where

$$\rho(\theta, \theta^c) = \min \left(\frac{f(\theta^c) q(\theta|\theta^c)}{f(\theta_t) q(\theta^c|\theta)} \right)$$

Algorithm 16 The Small World Sampler

- 1: With probability $(1 - s)$ simulate y from density $l(x, y)$ and compute $a = \frac{\pi(y) l(y, x)}{\pi(x) l(x, y)}$;
 - 2: Otherwise, simulate y from density $h(x, y)$ and compute $a = \frac{\pi(y) h(y, x)}{\pi(x) h(x, y)}$;
 - 3: Set $X_{n+1} \leftarrow y$ with probability $\min(1, a)$, otherwise set $X_{n+1} \leftarrow x$.
-

posterior distributions of the parameters of the model and the volatility spillover measures.

3.4.4 Implementation details

In this section, we briefly discuss implementation details of the algorithms considered in the previous section. We wrote R and Fortran custom code to run our simulation and empirical examples. In particular, we use Fortran for the estimation of the likelihood with auxiliary particle filter. As the procedure is sequential, writing it in Fortran/C can have large computational benefits. We gained a factor of 10 speed gain when implementing likelihood estimation in Fortran in comparison to R implementation.

We implement Metropolis-Hastings algorithm with multiple adjustments. First, we split the parameters into blocks $\theta = (\mu, \Phi, \Sigma_\epsilon, \Sigma_\eta)$. Further, we make separate random walk proposals for each row of these matrices, except for μ which is already a vector. Since we aim to estimate the model in multivariate case, it is not feasible to make proposals for all parameters at once as it would be nearly impossible to achieve a decent acceptance rate. However, this might be necessary for the calibration stage of the step size proposal. Further, we also use mixture random walk proposals. We find that a good start for the calibration of the algorithm is to use about 30% of big steps in the proposal distribution and smaller steps otherwise. Finding appropriate mixture distribution can be problem and model dependent and generally more than just two distributions can be used.

3.5 Simulation Experiments

In this section, we present preliminary simulation results for bivariate, trivariate and five-dimensional models. We use particle filtering for estimation of simulated likelihood and then the small world sampler to get draws from the posterior distribution. We show that already in this set-up estimation of volatility spillover effects can be achieved quite well. However, there are a few directions in which the current methodology can be

improved. We review these directions in the discussion at the end of this chapter.

3.5.1 Prior distributions

In our simulation and empirical studies we use prior distributions that are typically used in stochastic volatility framework, see for example Yu and Meyer (2006).

Commonly used priors	Alternative prior specification
<ul style="list-style-type: none"> • $\mu_i \sim N(0, \sigma_{\mu_i})$ • $\phi_{ii}^* \sim \text{Beta}(20, 1.5)$, with $\phi_{ii}^* = (\phi_{ii} + 1)/2$ • $\phi_{ij} \sim N(0, 1)$ • $\sigma_{\eta_i} \sim \text{IG}(2.5, 0.025)$ • $\rho_\epsilon, \rho_\eta \sim U(-1, 1)$ 	<ul style="list-style-type: none"> • $\Phi \sim$ Minnesota prior (Litterman et al. (1984)); • $\Phi \sim$ Horseshoe prior (Carvalho et al. (2009)); • $\Sigma_\epsilon, \Sigma_\eta \sim$ Inverse-Wishart prior.

Of course in the high-dimensional case, we face difficulty if selection a prior for matrix Φ , that is a matrix that is responsible for Granger-causal structure in the volatility process and hence in high-dimensional cases we would want to impose some sparsity constraint on it. Thus, when extending the model to higher dimensions, one might want to consider a Minnesota prior or a Horseshoe prior.

3.5.2 Bivariate simulated example

We start the illustration of measuring spillover effects with a multivariate stochastic volatility model on bivariate simulated data. Thus, the model reads

$$\begin{pmatrix} y_{1t} \\ y_{2t} \end{pmatrix} = \begin{pmatrix} \exp(\frac{h_{1t}}{2}) & 0 \\ 0 & \exp(\frac{h_{2t}}{2}) \end{pmatrix} \begin{pmatrix} \epsilon_{1t} \\ \epsilon_{2t} \end{pmatrix} \quad (3.38)$$

$$\begin{pmatrix} h_{1,t+1} \\ h_{2,t+1} \end{pmatrix} = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} + \begin{pmatrix} \phi_{11} & \phi_{12} \\ \phi_{21} & \phi_{22} \end{pmatrix} \begin{pmatrix} h_{1,t} - \mu_1 \\ h_{2,t} - \mu_2 \end{pmatrix} + \begin{pmatrix} \eta_{1t} \\ \eta_{2t} \end{pmatrix} \quad (3.39)$$

where $\epsilon_t = (\epsilon_{1t}, \epsilon_{2t}) \stackrel{iid}{\sim} N(0, \Sigma_\epsilon)$ and $\eta_t = (\eta_{1t}, \eta_{2t}) \stackrel{iid}{\sim} N(0, \Sigma_\eta)$. Here $\{h_{1t}\}_{t=1}^T$ and $\{h_{2t}\}_{t=1}^T$ are two latent volatility process which drive the dynamics of log-returns. The dependence is specified through the matrix Φ which encodes Granger-causal structure. The coefficients in autoregressive matrix Φ can be both positive and negative which allows for both positive and negative Granger-causal feedback. Standard conditions on stationarity apply from the VAR(p) model. Contemporaneous correlation between the log-returns is taken into account through the matrix Σ_ϵ . The parameters for the simulation experiment are chosen according to the values that have been found in empirical studies and are common for the data we are interested in. The data generating process has the following parameters:

$$\begin{aligned} \mu &= \begin{pmatrix} 0.025 \\ 0.025 \end{pmatrix}, & \phi &= \begin{pmatrix} 0.7 & 0.50 \\ 0.0 & 0.85 \end{pmatrix}, \\ \Sigma_\eta &= \begin{pmatrix} 0.04 & 0.00 \\ 0.00 & 0.07 \end{pmatrix}, & \Sigma_\epsilon &= \begin{pmatrix} 1 & 0.6 \\ 0.6 & 1 \end{pmatrix} \end{aligned} \quad (3.40)$$

We are in particular interested in how well the proposed approach captures Granger-causal relationship and how that translated into the estimation of the volatility spillover measures.

Posterior Estimation

We estimate the posterior distribution of the parameters of the model using particle Markov Chain Monte Carlo. The Figure 3.7 illustrates the trace of the posterior draws of the parameters for the bivariate stochastic volatility model. In this example, the number of the MCMC steps is 20000, and one can see that the algorithm converges quickly and that mixing of the posterior is quite good. The highest posterior density intervals are quite small

for all the parameters, and both Granger-causal effects and contemporaneous correlation are captured well. Figure 3.8 shows the autocorrelation for the posterior of the parameters of the model. The rapid decrease of which is a good sign of well-mixed chains. As you can see from the table below, most of the true parameter values fall into the highest posterior density intervals.

Spillover Measures

We further provide posterior summaries for spillover measures, in particular for the total spillover effects, directional and net spillover effects. Figure 3.9 shows the posterior draws and histogram for the total spillover effect measure which provides the visual illustration of the directional spillover effects. Table 3.2 summarizes information about the posterior distribution. Figure 3.10 shows posterior draws for the directional spillover measure and Table 3.4 summarizes the posterior distribution of these measures up to $M = 12$ steps ahead which allow us to access for how long these effects last. The directional spillover effects correspond to the true causal structure in the volatility process.

3.5.3 Trivariate example

In this section, we present results for a trivariate stochastic volatility model. The figure 3.11 shows simulated data for this example. The model reads

$$\begin{pmatrix} y_{1t} \\ y_{2t} \\ y_{3t} \end{pmatrix} = \begin{pmatrix} \exp(\frac{h_{1t}}{2}) & 0 & 0 \\ 0 & \exp(\frac{h_{2t}}{2}) & 0 \\ 0 & 0 & \exp(\frac{h_{3t}}{2}) \end{pmatrix} \begin{pmatrix} \epsilon_{1t} \\ \epsilon_{2t} \\ \epsilon_{3t} \end{pmatrix} \quad (3.41)$$

$$\begin{pmatrix} h_{1,t+1} \\ h_{2,t+1} \\ h_{3,t+1} \end{pmatrix} = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{pmatrix} + \begin{pmatrix} \phi_{11} & \phi_{12} & \phi_{13} \\ \phi_{21} & \phi_{22} & \phi_{23} \\ \phi_{31} & \phi_{32} & \phi_{33} \end{pmatrix} \begin{pmatrix} h_{1,t} - \mu_1 \\ h_{2,t} - \mu_2 \\ h_{3,t} - \mu_3 \end{pmatrix} + \begin{pmatrix} \eta_{1t} \\ \eta_{2t} \\ \eta_{3t} \end{pmatrix} \quad (3.42)$$

Figure 3.7: Trace plots of the MCMC chains for bivariate simulated example, respectively μ , Φ , Σ_η , ρ .

Note: This figure shows trace plots (in black) for parameters of the MSVM estimated with particle Markov Chain Monte Carlo methods, the blue line shows the true values of a parameter. The mixing of the chains appears to be quite good, the convergence happens quite fast, and the Granger-causal effects, as well as contemporaneous correlation, are captured well.

3 VOLATILITY SPILLOVERS WITH APPLICATION TO STOCHASTIC VOLATILITY MODELS

Figure 3.8: **ACF** for the posterior draws of the parameters.

Note: This figure shows autocorrelation function for the draws from the posterior distribution in the estimation of MSVM with particle Markov Chain Monte Carlo. This figure indicates good mixing of the chains.

Table 3.1: **Posterior results of the parameters in bivariate simulated example.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	95% <i>HPD_l</i>	95% <i>HPD_u</i>	<i>True</i>
μ_1	-0.07781	-0.07702	-0.28952	-0.27143	0.11153	0.025
μ_2	0.04991	0.05046	-0.03023	-0.05448	0.15461	0.025
ϕ_{11}	0.86208	0.86377	0.86588	0.77330	0.93850	0.7
ϕ_{21}	-0.00717	-0.00835	-0.00949	-0.03169	0.01869	0.0
ϕ_{12}	0.41983	0.40479	0.43705	0.19762	0.66214	0.5
ϕ_{22}	0.93091	0.93878	0.91039	0.84490	0.99853	0.85
σ_{η_1}	0.01223	0.01001	0.00817	0.00213	0.02885	0.04
σ_{η_2}	0.02156	0.01837	0.01067	0.00447	0.04628	0.07
ρ	0.57503	0.57604	0.57650	0.53144	0.61779	0.6

Note: This table shows posterior moments and highest posterior density intervals for the parameters estimated in the bivariate simulated example with particle Markov Chain Monte Carlo.

Figure 3.9: **Trace plot of the MCMC chain and histogram of the measure for total spillover in the bivariate simulated example calculated for lags $m = 1, \dots, 12$.**

Note: This figure illustrates trace plot of the total spillover measure (left) and the histogram of the draws (right).

3 VOLATILITY SPILLOVERS WITH APPLICATION TO STOCHASTIC VOLATILITY MODELS

Figure 3.10: **Histogram of the Directional Spillover Measures for bivariate simulated example.**

Note: These histograms show the directional spillover effects and how the effect changes over different lags. The corresponding DGP can be found in Equation 3.40. Subfigure (a) shows directional spillover to y_1 , subfigure (b) shows directional spillover to y_2 , subfigure (c) shows directional spillover to from y_1 and subfigure (d) shows directional spillover to from y_2 .

where $\epsilon_t = (\epsilon_{1t}, \epsilon_{2t}, \epsilon_{3t}) \stackrel{iid}{\sim} N(0, \Sigma_\epsilon)$ and $\eta = (\eta_{1t}, \eta_{2t}, \eta_{3t}) \stackrel{iid}{\sim} N(0, \Sigma_\eta)$. Similarly to the example in the previous subsection, $\{h_{1t}\}_{t=1}^T$, $\{h_{2t}\}_{t=1}^T$ and $\{h_{3t}\}_{t=1}^T$ are latent processes of the log-volatility that drive the dynamics of the log-returns. The Granger-causal structure is encoded in the matrix Φ and contemporaneous correlation is accounted for in matrix Σ_ϵ . The data generating process has the following parameters:

$$\mu = \begin{pmatrix} 0.025 \\ 0.025 \\ 0.025 \end{pmatrix}, \quad \phi = \begin{pmatrix} 0.8 & 0.0 & 0.5 \\ 0.0 & 0.8 & 0.4 \\ 0.0 & 0.0 & 0.9 \end{pmatrix},$$

$$\Sigma_\eta = \begin{pmatrix} 0.01 & 0.0 & 0.0 \\ 0.0 & 0.01 & 0.0 \\ 0.0 & 0.0 & 0.01 \end{pmatrix}, \quad \Sigma_\epsilon = \begin{pmatrix} 1 & 0.3 & 0.3 \\ 0.3 & 1 & 0.3 \\ 0.3 & 0.3 & 1 \end{pmatrix}.$$

First, we use the Metropolis-Hastings algorithm for making draws from the posterior distribution of the parameters. However, as it is seen from

Table 3.2: Posterior results for Total Spillover Measure for bivariate simulated example, $m = 1, \dots, 12$.

m	Mean	Median	Mode	95% HPD_l	95% HPD_u
1	0.1476	0.1454	0.0830	0.0100	0.2993
2	0.1194	0.1122	0.0702	0.0103	0.2765
3	0.0984	0.0887	0.0596	0.0075	0.2550
4	0.0830	0.0712	0.0509	0.0056	0.2363
5	0.0715	0.0600	0.0440	0.0025	0.2092
6	0.0627	0.0514	0.0383	0.0023	0.1923
7	0.0557	0.0449	0.0338	0.0033	0.1766
8	0.0502	0.0397	0.0302	0.0028	0.1629
9	0.0456	0.0356	0.0272	0.0014	0.1491
10	0.0418	0.0323	0.0247	0.0012	0.1384
11	0.0386	0.0295	0.0226	0.0011	0.1290
12	0.0358	0.0272	0.0208	0.0010	0.1209

Note: This table shows posterior moments and highest posterior density intervals for the total spillover measure at different lags.

3 VOLATILITY SPILLOVERS WITH APPLICATION TO STOCHASTIC VOLATILITY MODELS

Table 3.3: **Directional Spillover $S_{:,i}(1)$ transmitted by process i for bivariate simulated examples at lags $m = 1, \dots, 12$.**

$S_{:,i}(m)$	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	95% HPD_l	95% HPD_u
$S_{:,1}(1)$	0.0002	0.0001	0.000	0.0000	0.3944
$S_{:,2}(1)$	0.2984	0.2795	0.226	0.0201	0.6116
$S_{:,1}(2)$	0.0001	0.000	0.0000	0.0000	0.3579
$S_{:,2}(2)$	0.2352	0.204	0.1601	0.0027	0.5428
$S_{:,1}(3)$	0.0001	0.0000	0.0000	0.0000	0.3312
$S_{:,2}(3)$	0.1924	0.1565	0.1188	0.0019	0.4921
$S_{:,1}(4)$	0.0000	0.0000	0.0000	0.0000	0.3099
$S_{:,2}(4)$	0.1626	0.1251	0.0931	0.0014	0.4486
$S_{:,1}(5)$	0.0000	0.0000	0.0000	0.0000	0.2909
$S_{:,2}(5)$	0.1409	0.1037	0.0761	0.0011	0.4102
$S_{:,1}(6)$	0.0000	0.0000	0.0000	0.000	0.2726
$S_{:,2}(6)$	0.1245	0.0884	0.0643	0.0009	0.3747
$S_{:,1}(7)$	0.0000	0.0000	0.0000	0.000	0.2581
$S_{:,2}(7)$	0.1117	0.0769	0.0556	0.0008	0.3478
$S_{:,1}(8)$	0.0000	0.0000	0.000	0.000	0.2436
$S_{:,2}(8)$	0.1013	0.0681	0.049	0.0007	0.3221
$S_{:,1}(9)$	0.0000	0.000	0.0000	0.000	0.2306
$S_{:,2}(9)$	0.0928	0.061	0.0438	0.0006	0.2998
$S_{:,1}(10)$	0.0000	0.0000	0.0000	0.000	0.2186
$S_{:,2}(10)$	0.0857	0.0552	0.0396	0.0005	0.2797
$S_{:,1}(11)$	0.0000	0.0000	0.0000	0.000	0.2072
$S_{:,2}(11)$	0.0797	0.0505	0.0361	0.0005	0.2614
$S_{:,1}(12)$	0.0000	0.0000	0.0000	0.000	0.1968
$S_{:,2}(12)$	0.0745	0.0465	0.0332	0.0004	0.2451

Note: This table shows measures for directional transmitted spillover effects at 12 lags for bivariate simulated example. See Equation 3.40 for the specification of the data generating process. Coefficient $\phi_{12} = 0.5$ and $\phi_{21} = 0$ which corresponds to Granger-causal feedback from process 2 to 1, but not the other way around.

Table 3.4: **Directional Spillover $S_{\cdot i}(1)$ received by process i for bivariate simulated examples at lags $m = 1, \dots, 12$.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	95% HPD_t	95% HPD_u
$S_{1.}(1)$	0.2170	0.2185	0.1843	0.0363	0.3944
$S_{2.}(1)$	0.0003	0.0001	0.0000	0.0000	0.0010
$S_{1.}(2)$	0.1779	0.1695	0.138	0.0119	0.3579
$S_{2.}(2)$	0.0001	0.0000	0.000	0.0000	0.0005
$S_{1.}(3)$	0.1499	0.1353	0.1062	0.0033	0.3312
$S_{2.}(3)$	0.0001	0.0000	0.0000	0.0000	0.0003
$S_{1.}(4)$	0.1296	0.1112	0.0851	0.0016	0.3099
$S_{2.}(4)$	0.0001	0.0000	0.0000	0.0000	0.0002
$S_{1.}(5)$	0.1144	0.094	0.0707	0.0011	0.2909
$S_{2.}(5)$	0.0000	0.000	0.0000	0.0000	0.0002
$S_{1.}(6)$	0.1025	0.0812	0.0604	0.0009	0.2726
$S_{2.}(6)$	0.0000	0.0000	0.0000	0.000	0.0001
$S_{1.}(7)$	0.093	0.0714	0.0527	0.0008	0.2581
$S_{2.}(7)$	0.000	0.0000	0.0000	0.000	0.0001
$S_{1.}(8)$	0.0852	0.0637	0.0467	0.0007	0.2436
$S_{2.}(8)$	0.0000	0.0000	0.0000	0.000	0.0001
$S_{1.}(9)$	0.0787	0.0575	0.042	0.0006	0.2306
$S_{2.}(9)$	0.0000	0.0000	0.000	0.000	0.0001
$S_{1.}(10)$	0.0732	0.0524	0.0381	0.0005	0.2186
$S_{2.}(10)$	0.0000	0.0000	0.0000	0.000	0.0001
$S_{1.}(11)$	0.0684	0.0481	0.0349	0.0005	0.2072
$S_{2.}(11)$	0.0000	0.0000	0.0000	0.000	0.0001
$S_{1.}(12)$	0.0643	0.0444	0.0322	0.0004	0.1968
$S_{2.}(12)$	0.0000	0.0000	0.0000	0.000	0.0001

Note: This table shows measures for directional received spillover effects at 12 lags for bivariate simulated example. See Equation 3.40 for the specification of the data generating process. Coefficient $\phi_{12} = 0.5$ and $\phi_{21} = 0$ which corresponds to Granger-causal feedback from process 2 to 1, but not the other way around.

Figure 3.11: **Simulated Data for Trivariate SV Model.**

Note: This figure illustrates the data generated for the trivariate simulated example. The first row indicates the log-returns y_t^i , $i = 1, 2, 3$, and the second row indicates corresponding latent volatility processes h_t^i , $i = 1, 2, 3$.

the posterior draws in Figure 3.12 and autocorrelation in Figure 3.13, even with quite well calibrated Metropolis-Hastings algorithms (i.e., with an acceptance rate varying around 20%) we cannot get proper mixing in the MCMC chain. It suggests that alternative solutions have to be implemented such as the small-world sampler, see Algorithm 16.

Figure 3.12: Traceplots for the trivariate model with standard particle filter.

Note: This figure illustrates trace plots of the parameters in MSVM for trivariate simulated example with particle Markov Chain Monte Carlo and standard (bootstrap) particle filter.

3 VOLATILITY SPILLOVERS WITH APPLICATION TO STOCHASTIC VOLATILITY MODELS

Figure 3.13: Autocorrelation for the trivariate model with standard particle filter.

Note: Autocorrelation confirms suggestion from the trace-plots in Figure 3.13 that the mixing of the Markov chain is not very good.

Figure 3.14: **Histograms for the trivariate model with standard particle filter.**

3 VOLATILITY SPILLOVERS WITH APPLICATION TO STOCHASTIC VOLATILITY MODELS

In Figures 3.15, 3.16, 3.17 we present the results obtained with the small world sampler. The reader can notice that the mixing of the Φ matrix improves and convergence to the true values happens much faster. It is done without spending as much time for the calibration of the algorithm as in the case of standard Metropolis-Hastings implementation. The algorithm estimates non-zero values of Φ quite well which can be seen from the high posterior density intervals. Table 3.5 shows the posterior results for Φ matrix for the trivariate simulation example.

Table 3.5: **Posterior results for the Φ matrix with bootstrap particle filter used in particle MCMC.**

	ϕ_{11}	ϕ_{21}	ϕ_{31}	ϕ_{12}	ϕ_{22}	ϕ_{32}	ϕ_{13}	ϕ_{23}	ϕ_{33}
mean	0.808	0.026	-0.082	0.127	0.732	0.169	0.229	0.486	0.804
median	0.806	0.028	-0.078	0.117	0.738	0.167	0.241	0.477	0.800
mode	0.800	0.037	-0.082	0.091	0.741	0.136	0.248	0.415	0.791
95%HPD _l	0.787	-0.035	-0.178	0.055	0.695	0.096	0.083	0.373	0.709
95%HPD _u	0.834	0.074	-0.001	0.211	0.768	0.255	0.339	0.692	0.907
true	0.8	0.0	0.0	0.0	0.8	0.0	0.5	0.4	0.9

Note: This table indicated posterior moments and highest posterior density intervals for the Φ matrix, in this case, standard (bootstrap) particle filter was used in particle Markov Chain Monte Carlo. The true value of one of the non-zero parameters, ϕ_{23} lies inside the highest posterior density interval. The second one, ϕ_{13} , however, is not captured very well.

Table 3.6: **Posterior results for the Φ matrix with auxiliary particle filter used in particle MCMC.**

	ϕ_{11}	ϕ_{21}	ϕ_{31}	ϕ_{12}	ϕ_{22}	ϕ_{32}	ϕ_{13}	ϕ_{23}	ϕ_{33}
mean	0.829	-0.031	-0.039	-0.129	0.798	0.046	0.687	0.522	0.914
median	0.837	-0.030	-0.041	-0.141	0.800	0.046	0.708	0.542	0.914
mode	0.840	-0.021	-0.052	-0.179	0.800	0.049	0.731	0.565	0.929
95%HPD _l	0.741	-0.127	-0.106	-0.205	0.745	-0.025	0.436	0.338	0.855
95%HPD _u	0.906	0.033	0.011	0.005	0.836	0.128	0.910	0.714	1.004
true	0.8	0.0	0.0	0.0	0.8	0.0	0.5	0.4	0.9

Note: This table indicated posterior results for the Φ matrix, in this case an auxiliary particle filter was used in particle Markov Chain Monte Carlo. Note that true values are always captures in the highest posterior density intervals.

Figure 3.15: Trace plots for the trivariate simulated example with auxiliary particle filter.

Note: This figure shows trace plots for the trivariate simulated example in which auxiliary particle filter was used to estimate the likelihood in particle Markov Chain Monte Carlo. Note, that in this case mixing is better than in the case of the bootstrap particle filter, see Figure 3.12.

3 VOLATILITY SPILLOVERS WITH APPLICATION TO STOCHASTIC VOLATILITY MODELS

Figure 3.16: Autocorrelations for the trivariate model with auxiliary particle filter.

Note: This figure shows autocorrelations for the trivariate simulated example in which an auxiliary particle filter was used to estimate the likelihood in particle Markov Chain Monte Carlo. Note, that in this case mixing appears to be much better than in the case when we used the bootstrap particle filter, see Figure 3.13.

Figure 3.17: **Histograms for the trivariate model with auxiliary particle filter.**

Note: This figure shows histograms for the posterior draws in the trivariate simulated example. An auxiliary particle filter was used to estimate the likelihood in particle Markov Chain Monte Carlo.

3.5.4 Five-dimensional example

In this section, we demonstrate the estimation of a five-dimensional stochastic volatility model. The higher is the dimension of the stochastic volatility model, the more computational time we need for estimation. First of all, for the accuracy of the estimation of the simulated likelihood, it is necessary to use a more substantial number of particles. The increase in the number of particles linearly translates into the increase of the computational time. Second, in the high-dimensional space, it can happen that we will move to the "true parameters" much slower than in low-dimensional cases. The results presented in the graph and the table below are based only on 5000 MCMC runs. In this example, one can observe that parameters start to converge quite fast. However, it cannot be guaranteed to happen every time. While starting exploring the parameter space, we can easily move in the wrong direction and get stuck in the area of a local maximum likelihood for a while. Big-world proposals ensure that eventually, we would get out of the area of local maximum likelihood, but of course, that would require quite a few iterations. In practice, that is far from being ideal since we would like to reduce computational burden. Hence, a good starting point must be found to ensure faster convergence of the algorithm. It is possible to use different methods for choosing a nice starting point, and we mention some of them in the discussion section.

Figure 3.18: Trace plots of the parameter samples for Φ matrix with particle Markov Chain Monte Carlo in five-dimensional simulated example.

Note: In this figure every graphs corresponds to a trace plot of parameters ϕ_{ij} , $i = 1, 2, \dots, 5$, $j = 1, 2, \dots, 5$, of the autoregressive matrix Φ . The horizontal lines of the figure correspond to index i and vertical lines to index j . True parameters from which the data were generated are represented by the straight line.

Table 3.7: Posterior results of the Φ matrix for five-dimensional example.

	ϕ_{11}	ϕ_{21}	ϕ_{31}	ϕ_{41}	ϕ_{51}	ϕ_{12}	ϕ_{22}	ϕ_{32}	ϕ_{42}	ϕ_{52}	ϕ_{13}	ϕ_{23}	ϕ_{33}
mean	0.839	-0.103	-0.045	-0.033	0.071	0.054	0.705	-0.043	-0.119	0.135	0.338	0.149	0.909
median	0.831	-0.109	-0.049	-0.032	0.061	0.092	0.715	-0.038	-0.109	0.123	0.402	0.157	0.909
mode	0.824	-0.075	-0.054	-0.032	0.089	0.083	0.746	-0.081	-0.108	0.17	0.145	0.148	0.85
95%HPD _l	0.809	-0.147	-0.078	-0.073	0.007	-0.091	0.661	-0.144	-0.244	-0.055	0.021	0.022	0.829
95%HPD _u	0.876	-0.055	0.003	-0.007	0.148	0.180	0.746	0.036	-0.039	0.339	0.525	0.271	1.007
true	0.80	0.00	0.00	0.00	0.00	0.00	0.85	0.00	0.00	0.00	0.50	0.00	0.92
	ϕ_{43}	ϕ_{53}	ϕ_{14}	ϕ_{24}	ϕ_{34}	ϕ_{44}	ϕ_{54}	ϕ_{15}	ϕ_{25}	ϕ_{35}	ϕ_{45}	ϕ_{55}	
mean	0.153	0.007	0.167	0.252	0.172	0.59	0.772	-0.018	0.092	0.063	0.021	0.802	
median	0.161	0.023	0.142	0.259	0.1	0.601	0.757	-0.039	0.093	0.059	0.01	0.799	
mode	0.316	-0.211	0.147	0.065	0.079	0.618	0.778	-0.081	0.04	0.07	0.005	0.836	
95%HPD _l	-0.017	-0.225	-0.019	0.015	-0.029	0.507	0.587	-0.100	0.036	-0.012	-0.006	0.773	
95%HPD _u	0.249	0.205	0.484	0.508	0.439	0.677	0.989	0.108	0.139	0.164	0.055	0.836	
true	0.00	0.40	0.00	0.40	0.20	0.80	0.60	0.00	0.00	0.00	0.00	0.77	

Note: This table shows posterior results for the posterior distribution of the matrix Φ in the five-dimensional simulation example. The highest posterior density intervals often capture the true values of the parameter. In the cases, when the true value does not lie in the highest posterior density interval, in particular in some cases when the true parameter is zero, it indicates that the true value is very small.

3.5.5 Model Selection with Spike and Slab Prior

The primary focus of this paper is *measuring* volatility spillover effects. However, one also might be interested in making a statement regarding the presence or lack thereof. In the applications of our interest, one, in particular, might want to decide whether $\phi_{ij} = 0$ which corresponds to the case of Granger non-causality between volatility processes. The simplest way to decide that would be to use highest posterior density intervals that one can get as a side product of particle Markov Chain Monte Carlo inference, for the mathematical formulation of the problem, in this case, see the Appendix. However, the outcome might highly depend on the length of the chain and quality of the mixing of the Markov Chains which is harder to get in high-dimensional cases. As an alternative approach in this section, we propose using *spike and slab priors* for model selection problem. The idea behind this approach is to specify two regimes, *spike* with Dirac mass at zero, and *slab* in which parameter is assumed to have a Normal distribution.

$$p(\phi_{ij}|z_{ij}) = z_{ij}N(0, \sigma^2) + (1 - z_{ij})\mathbb{I}\{|\lambda| < 1\}. \quad (3.43)$$

where $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_k)$ is vector of eigenvalues of Φ and $I(\cdot)$ is indicator function, δ_0 is Dirac delta function at point 0. So the left-hand side of the sum represents the spike component in 0, and the right-hand side represents the slab component.

Figure 3.19: **Spike and slab prior.**

Note: This figure illustrates spike and slab prior expressed in Equation (3.43). There are two components: slab regime which is Gaussian distribution and spike regime which corresponds to a parameter being equal to zero.

Figure 3.20: **Trace plots of the posterior inclusion probabilities for elements of Φ**

Note: This figure shows trace plots of the Markov Chain Monte Carlo for parameters of the matrix Φ . Correspondingly for ϕ_{21} , ϕ_{31} , ϕ_{12} , ϕ_{23} , ϕ_{31} , ϕ_{32} the posterior inclusion probabilities are 0.314, 0.539, 0.544, 0.804, 0.388, 0.123. Note, that with this algorithm we manage to capture all zero elements of matrix Φ . However, this algorithm requires one to spend a lot of time on calibration of the proposals. Moreover, implementing it in high-dimensional case is too expensive.

In the following steps we outline the algorithm that we propose for model selection in low-dimensional multivariate stochastic volatility models with spike and slab prior incorporated into particle Markov Chain Monte Carlo.

- Initialize starting values for model selection $\theta = (\mu, \Phi, \Sigma_\eta, \rho)$.
- Initialize variance Σ_ν that defines size of the step for every block of the parameter vector θ .
- Initialize matrix Z such that the diagonal elements are equal to 1 and nondiagonal are zero. Element z_{ij} of this matrix attributes coefficient ϕ_{ij} either to the spike or to the slab component.
- **For** $l = 1, \dots, M$ **do**:
 1. Make proposals for nondiagonal elements of Φ , $\phi_{ij}^l \sim N(\phi_{ij}^{l-1}, \nu_{\phi_{ij}})$.
 2. Sample $1, \dots, k$, where k is the number of nondiagonal elements of Φ , let $\mathbf{s}$ be vector of these sampled indexes.
 3. **For** $g = 1, \dots, k$ **do**:
 - (a) Update element of Φ that corresponds to $\mathbf{s}(g)$, i.e. calculate $p(Y|Z)$ for two models, for z_{ij} in both regime 0 and 1.
 - (b) Calculate $p(Z) = \text{Bin}(p_Z + c_0, k - p_Z + d_0)$, where p_Z is number of nonzero elements in Z (excluding diagonal). If $c_0 = 1$ and $d_0 = 0$ then the prior is uninformative. Recall:

$$\pi \sim \text{Bin}(c_0, d_0),$$

where $p(z_{ij} = 1) = \pi$.

- (c) Using Sequential Monte Carlo compute

$$p(Z|Y) \propto p(Y|Z)P(Z).$$

- (d) Choose maximum from two $p(Z|Y)$ where in one case z_{ij} was 0 and in another 1. Move to the next step g . In the end

of the cycle $g = 1, \dots, k$ record the "final" model, this is the chosen model for l' th Markov Chain Monte Carlo step.

4. Conditioning on the chosen model Z at this iteration l update other parameters θ .
5. Record new parameters θ and likelihood, go to the next step $l + 1$.

The final model selection can be based on different rules. One can use, for example, the median probability model. In that case, we choose the regimes, 0 or 1, that correspond to the posterior inclusion probability being larger than 0.5. Our simulations with spike and slab priors show that in trivariate case the posterior inclusion probabilities "pick" the right non-zero elements of the matrix Φ . In other words, the elements that are non-zero have posterior inclusion probabilities larger than the threshold of 0.5. However, the procedure itself has a heavy computational burden: parameters of the matrix Z are updated sequentially. Thus, such model selection algorithm is only suitable for the low dimensional models. In higher dimensions, one would have to consider other alternatives.

3.6 Empirical applications

3.6.1 Spillovers between Australia and New Zealand

We consider application to the exchange rate between Australian and New Zealand dollars. These economies are often believed to have strong relations, see for example Choi et al. (2007), Brailsford (1996). In particular, Brailsford (1996) suggests that there are spillover effects from Australia to New Zealand and strong correlation effects. In this section we demonstrate the methodology for measuring spillover effects between exchange rates of Australia and New Zealand. Economies of these two countries are interconnected, and spillover effects have been studied in multiple papers. Judging from the posterior results, there is a small volatility spillover effect from New Zealand to Australia and even less spillover effect in the opposite direction. The correlation between the exchange rates tends to be high with the

posterior mean 0.74. We also do believe that using simulated likelihood we manage to estimate the variance of the volatility processes more accurately. In comparison to, for example, Integrated Nested Laplace Approximation approach which tends to overestimate the variance quite severely (Yu and Meyer (2006), Martino (2007)). We also try to estimate this model with non-blind proposals for the particles. The results are similar, but it does allow us to reduce the number of particles needed for estimation and hence slightly reduces computational time.

Figure 3.21: **Australian/US dollar (upper image) and New Zealand/US dollar (lower image).**

3 VOLATILITY SPILLOVERS WITH APPLICATION TO STOCHASTIC VOLATILITY MODELS

Table 3.8: **Descriptive statistics of the exchange rates between Australia and New Zealand.**

	Aus/US	NZ/US
Mean	-0.04	-0.07
Std.dev.	1.00	1.00
Skewness	0.06	-0.12
Kurtosis	3.37	4.99

Note: This table shows descriptive statistics for the exchange rate data of Australia and New Zealand. Both time series exhibit similar behaviour.

Figure 3.22: **Trace plots of particle Markov Chain Monte Carlo in Australia/New Zealand Empirical application.**

Note: This figure shows trace plots for empirical application on Australia/New Zealand exchange rate data. The mixing of the chains appears to be quite good. Table 3.9 indicates corresponding posterior moments and highest posterior density intervals for the parameters.

Table 3.9: **Posterior results of the parameters in MSVM for the Australia/New Zealand empirical application.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	95% <i>HPD_l</i>	95% <i>HPD_u</i>
μ_1	-0.22527	-0.22055	-0.16383	-0.51039	0.05681
μ_2	-0.44367	-0.42704	-0.37741	-0.91139	0.05046
ϕ_{11}	0.86676	0.88331	0.92113	0.71584	0.97563
ϕ_{21}	0.12785	0.10820	0.05697	-0.01993	0.29306
ϕ_{12}	0.07101	0.06099	0.05698	0.00535	0.16404
ϕ_{22}	0.91722	0.92977	0.94479	0.82405	0.99846
σ_{η_1}	0.00803	0.00710	0.00443	0.00234	0.01647
σ_{η_2}	0.01295	0.01055	0.00994	0.00248	0.02933
σ_ρ	0.74279	0.74370	0.74205	0.70254	0.78594

Note: This table shows posterior moments and highest posterior density intervals for the MSVM estimated with particle Markov Chain Monte Carlo for the Australia/New Zealand empirical application. We do not observe strong indication for spillover effects since the coefficients of autoregressive matrix Φ are estimated to be rather small. The dependency between two economies is observable from the contemporaneous correlation that is estimated to be between 0.7 and 0.78.

3 VOLATILITY SPILLOVERS WITH APPLICATION TO STOCHASTIC VOLATILITY MODELS

Figure 3.23: **Posterior draws and histograms for directional spillover measure to h_i in Australia/New Zealand empirical application.**

Note: This figure shows trace plot and histogram for directional spillover measure, in this case indicating spillovers received by country i , $i = 1$ for Australia and $i = 2$ for New Zealand respectively. The directional spillover measure does not indicate either of the countries being a receiver of volatility spillovers.

Figure 3.24: **Posterior draws and histogram for directional spillover measure from h_i in Australia/New Zealand empirical application.**

Note: This figure shows trace plot and histogram for directional spillover measure, in this case indicating spillovers transmitted by country i , $i = 1$ for Australia and $i = 2$ for New Zealand respectively. The directional spillover measure does not indicate either of the countries being a transmitter of volatility spillovers.

Figure 3.25: **Posterior draws and histogram for total spillover measure in Australia/New Zealand empirical application.**

Note: This figure shows trace plot and histogram for total spillover effects in the system. No significant spillover effects are indicated by the measure.

3 VOLATILITY SPILLOVERS WITH APPLICATION TO STOCHASTIC VOLATILITY MODELS

Table 3.10: **Directional Spillover, $S_i(m)$ transmitted by $i = 1, 2$ in Australia/New Zealand empirical application at lags $m = 1, 2, \dots, 12$.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	95% HPD_l	95% HPD_u
$S_1(1)$	0.0160	0.0039	0.0218	0.0000	0.0785
$S_2(1)$	0.0206	0.0078	0.0018	0.0000	0.0851
$S_1(2)$	0.0088	0.0020	0.0117	0.0000	0.0429
$S_2(2)$	0.0109	0.0039	0.0009	0.0000	0.0448
$S_1(3)$	0.0060	0.0013	0.0078	0.0000	0.0294
$S_2(3)$	0.0074	0.0026	0.0006	0.0000	0.0303
$S_1(4)$	0.0046	0.0010	0.0059	0.0000	0.0224
$S_2(4)$	0.0056	0.0019	0.0004	0.0000	0.0229
$S_1(5)$	0.0037	0.0008	0.0047	0.0000	0.0180
$S_2(5)$	0.0045	0.0016	0.0003	0.0000	0.0184
$S_1(6)$	0.0031	0.0007	0.0039	0.0000	0.0151
$S_2(6)$	0.0038	0.0013	0.0003	0.0000	0.0154
$S_1(7)$	0.0026	0.0006	0.0034	0.0000	0.0130
$S_2(7)$	0.0032	0.0011	0.0002	0.0000	0.0132
$S_1(8)$	0.0023	0.0005	0.0029	0.0000	0.0114
$S_2(8)$	0.0028	0.0010	0.0002	0.0000	0.0116
$S_1(9)$	0.0021	0.0004	0.0026	0.0000	0.0102
$S_2(9)$	0.0025	0.0009	0.0002	0.0000	0.0103
$S_1(10)$	0.0019	0.0004	0.0024	0.0000	0.0092
$S_2(10)$	0.0023	0.0008	0.0002	0.0000	0.0093
$S_1(11)$	0.0017	0.0004	0.0021	0.0000	0.0084
$S_2(11)$	0.0021	0.0007	0.0002	0.0000	0.0084
$S_1(12)$	0.0015	0.0003	0.0020	0.0000	0.0077
$S_2(12)$	0.0019	0.0006	0.0001	0.0000	0.0077

Note: This table shows measures for transmitted directional spillover effects for the bivariate empirical application with Australia/New Zealand data. The reader can observe that no significant directional transmitted spillover effects were detected.

Table 3.11: **Directional Spillover $S_{i.(1)}$ received by $i = 1, 2$ in Australia/New Zealand empirical application at lags $m = 1, 2, \dots, 12$.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	95% <i>HPD_l</i>	95% <i>HPD_u</i>
$S_{1.(1)}$	0.0193	0.0078	0.0018	0.0000	0.0785
$S_{2.(1)}$	0.0150	0.0039	0.0213	0.0000	0.0683
$S_{1.(2)}$	0.0105	0.0039	0.0009	0.0000	0.0429
$S_{2.(2)}$	0.0085	0.0020	0.0115	0.0000	0.0389
$S_{1.(3)}$	0.0072	0.0026	0.0006	0.0000	0.0294
$S_{2.(3)}$	0.0058	0.0013	0.0078	0.0000	0.0268
$S_{1.(4)}$	0.0055	0.0019	0.0004	0.0000	0.0224
$S_{2.(4)}$	0.0045	0.0010	0.0058	0.0000	0.0203
$S_{1.(5)}$	0.0044	0.0016	0.0004	0.0000	0.0180
$S_{2.(5)}$	0.0036	0.0008	0.0047	0.0000	0.0163
$S_{1.(6)}$	0.0037	0.0013	0.0003	0.0000	0.0151
$S_{2.(6)}$	0.0030	0.0007	0.0039	0.0000	0.0137
$S_{1.(7)}$	0.0032	0.0011	0.0002	0.0000	0.0130
$S_{2.(7)}$	0.0026	0.0006	0.0034	0.0000	0.0117
$S_{1.(8)}$	0.0028	0.0010	0.0002	0.0000	0.0114
$S_{2.(8)}$	0.0023	0.0005	0.0029	0.0000	0.0103
$S_{1.(9)}$	0.0025	0.0009	0.0002	0.0000	0.0102
$S_{2.(9)}$	0.0020	0.0004	0.0026	0.0000	0.0092
$S_{1.(10)}$	0.0022	0.0008	0.0002	0.0000	0.0092
$S_{2.(10)}$	0.0018	0.0004	0.0024	0.0000	0.0083
$S_{1.(11)}$	0.0020	0.0007	0.0002	0.0000	0.0084
$S_{2.(11)}$	0.0017	0.0004	0.0021	0.0000	0.0075
$S_{1.(12)}$	0.0019	0.0006	0.0001	0.0000	0.0077
$S_{2.(12)}$	0.0015	0.0003	0.0020	0.0000	0.0069

Note: This table shows measures for received directional spillover effects for the bivariate empirical application with Australia/New Zealand data. The reader can observe that no significant directional received spillover effects were detected.

Table 3.12: **Posterior results for the total spillover measure in Australia/New Zealand empirical application.**

<i>Mean</i>	<i>Median</i>	<i>Mode</i>	95% <i>HPD_l</i>	95% <i>HPD_u</i>
0.0183	0.0097	0.0118	0.0000	0.0658
0.0099	0.0049	0.0063	0.0000	0.0363
0.0067	0.0033	0.0042	0.0000	0.0247
0.0051	0.0025	0.0032	0.0000	0.0187
0.0041	0.0020	0.0025	0.0000	0.0150
0.0034	0.0016	0.0021	0.0000	0.0126
0.0029	0.0014	0.0018	0.0000	0.0108
0.0026	0.0012	0.0016	0.0000	0.0095
0.0023	0.0011	0.0014	0.0000	0.0084
0.0021	0.0010	0.0013	0.0000	0.0076
0.0019	0.0009	0.0012	0.0000	0.0069
0.0017	0.0008	0.0011	0.0000	0.0064

Note: This table shows the measure for total spillover effects in the system for bivariate empirical application with Australia/New Zealand data. The reader can observe that no significant spillover effects were detected in the system.

We applied the proposed framework of estimation and measuring volatility spillover effects with a multivariate stochastic volatility model for the exchange rates of Australia and New Zealand. Our results suggest the following. The economies are indeed strongly interrelated. However, the relationship is represented by rather high correlation, and no significant spillover effects were found in our sample.

3.6.2 Five-dimensional Application

In this empirical application, we use stock market indexes of five countries: the US, the UK, Germany, Japan, China. The indexes in corresponding order are S&P 500, FTSE 100, DAX, Nikkei 225, HSI.¹ We analyze daily returns, and the sample runs from 03/01/2000 until 17/05/2001. Descriptive statistics for this dataset are given in Table 3.13.

¹The data were downloaded from <http://www.oxford-man.ox.ac.uk/>

Tables 3.14, 3.15 and 3.16 present the moments of the posterior distributions for the parameters of the multivariate stochastic volatility model estimated with particle Markov Chain Monte Carlo methods. Further, Table 3.17 shows total spillover measure for 5 lags. The table indicates that there is a moderate amount of the spillover effects in the system during the period considered. Table 3.18 indicates directional spillover effects for transmitters of volatility spillovers in the system. In particular, the US appears to be a strong transmitter of the spillover effects with spillover measure being equal to 0.75. Further, the UK is a strong transmitter of spillover effects with spillover measure being 0.7. The directional spillovers transmitted by Germany have measure 0.48 and Japan 0.29. China appears not to transmit any significant volatility spillovers.

Table 3.19 indicates directional spillover effects for receivers of spillover effects. There are multiple observations to be made from this table. First, during the time period considered Germany and China appear to be the main receivers of volatility spillover effects. Japan also received a moderate amount of volatility spillovers from other countries. The US and the UK appear to be the least affected by volatility spillover effects.

Overall, in the sample period, we are considering the US and the UK are the primary transmitters of the spillover effects, while Germany, China,

Table 3.13: Descriptive Statistics for Five Dimensional Empirical Application.

	S&P 500	FTSE 100	DAX	Nikkei 225	HSI
Mean	-0.0007	-0.0007	-0.0010	-0.0010	-0.0011
Std.dev.	0.0132	0.0109	0.0127	0.0143	0.0136
Skewness	-0.1619	-0.1282	0.2016	0.0667	-0.0703
Kurtosis	4.2839	3.3129	3.2530	3.1541	3.5277

Note: This table shows descriptive statistics for the five dimensional empirical application. Multiple stock market indices are considered. S&P 500 corresponds to American stock market index, FTSE 100 corresponds to London stock exchange index, DAX to German stock index, Nikkei 25 to Tokyo stock exchange and HSI to Hong Kong.

and Japan are the primary receivers of the spillover effects. The results are as to be expected from other studies Gamba-Santamaria et al. (2017). The US often appears to be a strong transmitter of the volatility spillovers while China is mainly a receiver of the spillover effects.

Table 3.14: **Posterior results for vector μ in the five-dimensional empirical example.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	95% <i>HPD_l</i>	95% <i>HPD_u</i>
μ_1	0.0438	0.0505	0.0603	-0.0152	0.0856
μ_2	0.0473	0.0512	0.0557	-0.0044	0.0920
μ_3	0.0472	0.0522	0.0111	0.0015	0.0785
μ_4	0.0429	0.0453	0.0053	-0.0372	0.0907
μ_5	0.0439	0.0399	0.0469	0.0004	0.0911

Note: This table shows posterior moments and highest posterior density interval for the scale parameter μ in the latent volatility process.

Table 3.15: **Posterior results for matrix Φ in the five-dimensional empirical application.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	95% <i>HPD_l</i>	95% <i>HPD_u</i>
ϕ_{11}	0.9534	0.9631	0.9762	0.9030	0.9878
ϕ_{21}	0.2549	0.2198	0.3690	0.1675	0.3872
ϕ_{31}	-0.4136	-0.3234	-0.7357	-0.7357	-0.1150
ϕ_{41}	-0.5176	-0.5300	-0.5384	-0.5909	-0.4198
ϕ_{51}	0.1002	0.0688	0.1183	0.0001	0.2477
ϕ_{12}	0.0683	0.0686	0.0012	0.0001	0.1843
ϕ_{22}	0.8285	0.8577	0.7327	0.7256	0.9205
ϕ_{32}	0.2831	0.2572	0.4140	0.1094	0.5351
ϕ_{42}	0.3627	0.4030	0.4195	0.0612	0.5261
ϕ_{52}	0.2436	0.2485	0.1899	0.0531	0.3850
ϕ_{13}	0.5825	0.5449	0.5323	0.4243	0.9999
ϕ_{23}	0.3347	0.3659	0.4072	0.1049	0.4515
ϕ_{33}	0.7818	0.7601	0.6846	0.6846	0.9006
ϕ_{43}	0.0974	0.0666	0.0199	0.0010	0.2564
ϕ_{53}	0.5337	0.4805	0.4794	0.2960	0.7752
ϕ_{14}	0.4463	0.4385	0.4013	0.3489	0.5747
ϕ_{24}	0.2537	0.3037	0.0218	0.0218	0.4671
ϕ_{34}	0.1111	0.0665	0.2769	0.0050	0.2769
ϕ_{44}	0.8652	0.8178	0.8101	0.7531	0.9849
ϕ_{54}	0.1975	0.1977	0.0417	0.0409	0.4187
ϕ_{15}	0.1097	0.1098	0.0626	0.0626	0.1503
ϕ_{25}	0.0245	0.0244	0.0244	0.0006	0.0636
ϕ_{35}	0.0434	0.0281	0.0266	0.0000	0.1249
ϕ_{45}	0.0968	0.0852	0.0785	0.0056	0.1737
ϕ_{55}	0.8599	0.8807	0.7678	0.7678	0.9453

Note: This table shows posterior moments and highest posterior density interval for the autoregressive matrix Φ in the latent volatility process.

Table 3.16: **Posterior results for covariances and variances in the five-dimensional empirical example.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	95% <i>HPD_l</i>	95% <i>HPD_u</i>
σ_{η_1}	0.2646	0.2494	0.2548	0.2153	0.3721
σ_{η_2}	0.2429	0.2476	0.2224	0.1592	0.3430
σ_{η_3}	0.0487	0.0206	0.0212	0.0049	0.1663
σ_{η_4}	0.0791	0.0478	0.0060	0.0052	0.2179
σ_{η_5}	0.0347	0.0201	0.0048	0.0038	0.0930
$\sigma_{\epsilon_{21}}$	0.2477	0.2728	0.1820	0.1820	0.3059
$\sigma_{\epsilon_{31}}$	0.4355	0.4498	0.5074	0.3188	0.5074
$\sigma_{\epsilon_{41}}$	0.5134	0.5120	0.6279	0.3725	0.6279
$\sigma_{\epsilon_{51}}$	0.5080	0.5296	0.5922	0.3147	0.5922
$\sigma_{\epsilon_{32}}$	0.3793	0.3750	0.4551	0.2907	0.4551
$\sigma_{\epsilon_{42}}$	0.4433	0.4475	0.4862	0.3681	0.4862
$\sigma_{\epsilon_{52}}$	0.4436	0.4463	0.5434	0.2952	0.5434
$\sigma_{\epsilon_{43}}$	0.2918	0.2956	0.2956	0.2501	0.3521
$\sigma_{\epsilon_{53}}$	0.3055	0.3077	0.3077	0.2792	0.3362
$\sigma_{\epsilon_{54}}$	0.2600	0.2526	0.1654	0.1654	0.3918

Note: This table shows posterior moments and highest posterior density interval for the variances in the latent volatility process σ_{η_i} , $i = 1, \dots, 5$ and covariances in the log-returns $\sigma_{\epsilon_{ij}}$, $i = 1, \dots, 5$, $j = 1, \dots, 5$, $i \neq j$.

Table 3.17: **Posterior results the total spillover measure in five-dimensional application with $m = 5$ lags.**

<i>m</i>	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	95% <i>HPD_l</i>	95% <i>HPD_u</i>
1	0.45	0.44	0.44	0.25	0.59
2	0.37	0.37	0.37	0.16	0.51
3	0.32	0.33	0.33	0.09	0.45
4	0.29	0.30	0.30	0.07	0.42
5	0.27	0.28	0.28	0.06	0.39

Note: This table shows total spillover measure for 5 lags in the five-dimensional empirical application. The posterior moments and highest posterior density intervals indicate that strong spillover effects are consistently present in the system.

Table 3.18: **Directional Spillover $S_{\cdot i}(m)$ transmitted by country i at lag m .**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	95% <i>HPD_l</i>	95% <i>HPD_u</i>
$S_1(1)$	0.75	0.77	0.65	0.52	0.39
$S_2(1)$	0.70	0.73	0.61	0.31	0.92
$S_3(1)$	0.48	0.48	0.70	0.04	0.83
$S_4(1)$	0.29	0.24	0.22	0.10	0.56
$S_5(1)$	0.01	0.01	0.00	0.00	0.03
$S_1(2)$	0.70	0.73	0.63	0.39	0.27
$S_2(2)$	0.60	0.62	0.48	0.18	0.89
$S_3(2)$	0.39	0.34	0.64	0.03	0.83
$S_4(2)$	0.16	0.12	0.12	0.03	0.38
$S_5(2)$	0.00	0.00	0.00	0.00	0.01
$S_1(3)$	0.66	0.68	0.61	0.31	0.21
$S_2(3)$	0.53	0.52	0.38	0.12	0.87
$S_3(3)$	0.33	0.26	0.59	0.01	0.79
$S_4(3)$	0.09	0.07	0.07	0.02	0.29
$S_5(3)$	0.00	0.00	0.00	0.00	0.01
$S_1(4)$	0.63	0.65	0.60	0.24	0.17
$S_2(4)$	0.47	0.44	0.31	0.09	0.85
$S_3(4)$	0.30	0.20	0.54	0.00	0.74
$S_4(4)$	0.07	0.04	0.05	0.00	0.22
$S_5(4)$	0.00	0.00	0.00	0.00	0.01
$S_1(5)$	0.60	0.62	0.60	0.21	0.15
$S_2(5)$	0.43	0.39	0.26	0.02	0.77
$S_3(5)$	0.27	0.18	0.49	0.00	0.69
$S_4(5)$	0.05	0.03	0.04	0.00	0.18
$S_5(5)$	0.00	0.00	0.00	0.00	0.00

Note: This table shows directional spillover effects, it indicates the volatility spillovers which were transmitted by country i , correspondingly the US, the UK, Germany, Japan and China are encoded as $i = 1, 2, 3, 4, 5$.

3 VOLATILITY SPILLOVERS WITH APPLICATION TO STOCHASTIC VOLATILITY MODELS

Table 3.19: **Directional Spillover $S_{i.(m)}$ received by country i at lag m .**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	95% <i>HPD_l</i>	95% <i>HPD_u</i>
$S_{1.}(1)$	0.27	0.27	0.38	0.13	0.39
$S_{2.}(1)$	0.21	0.21	0.30	0.08	0.34
$S_{3.}(1)$	0.56	0.55	0.68	0.28	0.76
$S_{4.}(1)$	0.39	0.36	0.30	0.23	0.57
$S_{5.}(1)$	0.47	0.47	0.46	0.21	0.62
$S_{1.}(2)$	0.15	0.14	0.27	0.05	0.27
$S_{2.}(2)$	0.11	0.10	0.19	0.02	0.21
$S_{3.}(2)$	0.48	0.46	0.62	0.15	0.74
$S_{4.}(2)$	0.34	0.32	0.24	0.19	0.54
$S_{5.}(2)$	0.40	0.41	0.43	0.14	0.60
$S_{1.}(3)$	0.10	0.08	0.21	0.02	0.21
$S_{2.}(3)$	0.07	0.06	0.14	0.02	0.15
$S_{3.}(3)$	0.43	0.41	0.58	0.10	0.71
$S_{4.}(3)$	0.32	0.28	0.19	0.15	0.53
$S_{5.}(3)$	0.35	0.37	0.40	0.08	0.56
$S_{1.}(4)$	0.07	0.06	0.17	0.01	0.17
$S_{2.}(4)$	0.05	0.04	0.11	0.01	0.12
$S_{3.}(4)$	0.40	0.37	0.55	0.08	0.68
$S_{4.}(4)$	0.30	0.25	0.16	0.12	0.51
$S_{5.}(4)$	0.31	0.33	0.37	0.06	0.53
$S_{1.}(5)$	0.06	0.04	0.15	0.01	0.15
$S_{2.}(5)$	0.04	0.03	0.09	0.01	0.10
$S_{3.}(5)$	0.37	0.34	0.53	0.07	0.66
$S_{4.}(5)$	0.28	0.23	0.14	0.10	0.50
$S_{5.}(5)$	0.29	0.30	0.34	0.05	0.50

Note: This table shows directional spillover effects, it indicates the volatility spillovers which were received by country i , correspondingly the US, the UK, Germany, Japan and China are encoded as $i = 1, 2, 3, 4, 5$.

3.7 Discussion

In this chapter, we propose a fully Bayesian approach for detection and measuring of volatility spillover effects between financial markets. Our approach is based on the definition of volatility spillovers through Granger-causality in the latent volatility process of the multivariate stochastic volatility model. As the volatility process has the form of vector autoregressive model corresponding theory can be applied for identifying and measuring spillover effects on financial markets. We apply measures developed in Diebold and Yilmaz (2012) to the latent volatility process, and estimate posterior distributions of thereof. The advantage of our approach is its flexibility and interpretability. The flexibility of the multivariate stochastic volatility models allows contemporaneous correlation to be both positive and negative. Moreover, since the volatility process is defined through Vector Autoregressive model, we can account for Granger-causality in the volatility process. Thus, we can distinguish between contemporaneous and lagged relationships and take both into account when constructing volatility spillover measures. The Bayesian framework allows us to obtain not only point estimates, but also posterior distributions of the parameters and spillover measures. We report the highest posterior density intervals to quantify the uncertainty about the parameters and spillover measures of interest. Besides developing volatility spillover measures, we also propose an approach to model selection and decision making regarding autoregressive coefficients in the volatility process (and thus regarding Granger-causal relationships). Our approach is based on incorporation of spike and slab prior into particle Markov Chain Monte Carlo framework. Spike and slab prior allows us to move simultaneously in the parameter and models space. In the trivariate example we show that this approach works well in the low dimensional case, however, for higher dimensions, it is computationally costly.

We estimate the model with particle Markov Chain Monte Carlo (PMCMC) methods and discuss issues that had to be resolved to estimate the model in dimensionalities considered. In the five-dimensional case, two crucial steps have to be taken into account. First, the variance of the estimated

likelihood grows exponentially with the dimensionality of the model. The estimation is improved by using the auxiliary particle filter. Second, with the five-dimensional model, we have 45 parameters to be estimated. In this case mixing of the Markov Chains is of concern. Mixture proposals can significantly improve the performance of the Metropolis-Hastings algorithm. Finally, to improve convergence results using appropriate methods on a simplified model and using the results as a starting point in PMCMC can be helpful.

We consider two empirical applications. The first one is for detecting spillover effects between exchange rates of Australia and New Zealand. We find that there are insignificant Granger-causal relationships and rather high correlation for these time series data. In our five-dimensional application, we consider financial indexes of the following countries: United States, United Kingdom, Germany, Japan, and China. Our results indicate that the US and the UK are main volatility spillovers transmitters during the sample period 03/01/2000 until 17/05/2001 while Germany, China, and Japan are main volatility spillovers receivers.

Finally, we discuss the way in which the current approach can be improved and extended to higher dimensions. First, the high variance of the estimated likelihood in multivariate models has to be addressed with different methods. Most promising once would be Particle Efficient Importance Sampler (Scharth and Kohn, 2016) and iterated Auxiliary Particle Filter (Guarniero et al., 2017). Both methods are appealing. However, both Scharth and Kohn (2016) and Guarniero et al. (2017) use stochastic volatility model with diagonal structure in the autoregressive matrix to demonstrate the methods. Thus, they have to be adapted for the case when Granger-causal relationships are taken into account. Second, with increasing dimensionality of the model the parameter space also grows. Using a random walk proposal in the Metropolis-Hastings algorithm, in this case, can result in poor mixing of the Markov chains. Recently, Hamiltonian Monte Carlo methods with Langevin dynamics have gained popularity in the literature, see Girolami and Calderhead (2011b). These methods avoid drawbacks from which the random walk proposals suffer. While with ran-

dom walk proposals, we are likely to move in any direction of the parameter space with equal probability, Hamiltonian Monte Carlo methods exploit differential geometry to make proposals which are more likely to be accepted. It leads to a more efficient exploration of the parameter space and better mixing of the Markov chains.

Finally, when the estimation is improved it would be interesting from both methodological and applied points of view to extend the model and allow for time variation in both autoregressive matrix and correlation coefficients. It has been shown in the literature, for example, see Gamba-Santamaria et al. (2017), that spillover effects can change over time.

3.A Appendix: Five-dimensional empirical example

Figure 3.26: Histograms of the posterior draws for matrix Φ in the five-dimensional empirical example.

Note: This figure shows histograms for elements of the Φ matrix sampled with PMCMC, blue lines indicate zero point.

3.B Appendix: Resampling

Resampling is an essential step in Sequential Monte Carlo algorithms. The idea behind it is quite simple: as we sequentially progress with a version of *importance sampling* algorithm over time, the algorithm becomes unstable, and particles degenerate. To avoid this problem and make sure that the population of particles has some diversity one must perform *resampling* step. However, one can exploit different resampling schemes. Especially in high dimensions, the quality of resampling can affect the results significantly. In the implementation of our chosen methodology, we use *systematic* resampling. For a general review of resampling schemes see Douc and Cappé (2005) and for some theoretical properties Gerber et al. (2017).

3.C Appendix: Posterior Draws

The standard off-shelf solution for making draws from the posterior distribution is Metropolis-Hastings algorithms. However, in high dimensions, it does not provide good mixing. As an alternative, in particular, in this paper, we used mixed proposals in the algorithm to improve mixing. However, once the dimension increases further, that might not be enough. In that case a possible solution can be STEEP sampler (Guan and Stephens (2012)). STEEP sampler combines two algorithms: the small-world sample and equi-energy sampler. Generally, more than two chains can be used in this algorithm (as well as in the small-world sampler algorithm). We do not apply this algorithm in our simulations since we manage to get good acceptance rates with the small world sampler. However, once we go to higher dimensions most likely, we would have to use STEEP sampler or similar alternatives.

Let $\pi(x) \propto \pi(x)^{1/t}$ for some $t > 1$. $\{Y_n\}$ and $\{X_n\}$ sample π_t and π respectively, and $Y_n = y_n$ and $X_n = x_n$.

3 VOLATILITY SPILLOVERS WITH APPLICATION TO STOCHASTIC VOLATILITY MODELS

Algorithm 17 Systematic Resampling

```

1:  $[\{h_t^i\}_{i=1}^N] = \text{Resample}[\{h_t^{(m)}, \omega_t^{(m)}\}_{m=1}^M, N]$ 
2:
3:  $[\{Q_t^m\}_{m=1}^M] = \text{CumulativeSum}[\{\omega_t^{(m)}\}_{m=1}^M]$ 
4:
5:  $i = 0$ 
6:
7:  $m = 1$ 
8:
9:  $u_0 \sim U(0, 1/N]$ 
10:
11: WHILE  $(n \leq N)$ 
12:
13:  $u = u_0 + n/N$ 
14:
15: WHILE  $(Q_t^{(m)} < u)$ 
16:
17:  $m = m + 1$ 
18:
19: END
20:
21:  $n = n + 1$ 
22:
23:  $\tilde{h}_t^{(i)} = h_t^{(m)}$ 
24:

```

Algorithm 18 STEEP Sampler

- 1: Simulate $Y_{n+1} \mid Y_n = y_n$ using Small World Sampler to obtain y_{n+1} .
Update empirical measure $\xi_{n+1} = n/(n+1)\xi_n + 1/(n+1)\delta(y_{n+1})$.
 - 2: Simulate $X_{n+1} \mid X_n = x_n$ using Small World Sampler. Modify step (2)
so that $y \sim \xi_{n+1}$ and compute $a = \frac{\pi(y)}{\pi(x_n)} \frac{\pi_t(x_n)}{\pi_t(y)}$. Set $X_{n+1} \leftarrow y$ with
probability $\min(1, a)$, otherwise set $X_{n+1} \leftarrow x_n$.
 - 3: Set $n \leftarrow n + 1$, goto step 1.
-

Chapter 4

Multivariate Models for Count Data

Count data appears in many research fields and exhibits certain features that make modeling challenging. Most popular approaches can be classified into observation and parameter driven models. One of the popular observation driven approaches is the *autoregressive conditional intensity* model (ACI) also sometimes referred to as the *integer-valued generalized autoregressive conditional heteroskedastic* model (INGARCH). These models can be easily estimated under the assumption of no contemporaneous correlation between time series. Implementation of the quasi-maximum likelihood approach is straightforward, and estimation is fast. The challenging part starts when one wants to include contemporaneous dependence in the model. In bivariate case, it is relatively simple with a trivariate reduction technique, even though quite restrictive since covariance, in this case, has to be positive. Extending this method to higher dimensions is also cumbersome. On the contrary, parameter driven state-space models allow for high degree of flexibility in modeling the dependence. Through the latent process, one can infer about both negative and positive contemporaneous correlation and Granger-causal relationships. The price of this flexibility comes at a high computational cost. The presence of the latent process leads to the intractability of the likelihood, and thus appropriate methods

have to be applied to the estimation problem. In this paper, we review multivariate approaches to modeling count data and in particular compare observation and parameter driven approaches in the multivariate framework. We look at the fit the models provide together with the predictive performance for three data applications: transaction data on financial markets, bank failures, and car crash counts.

4.1 Introduction

Count time series appear in many scientific fields, and there are various approaches for modeling time series of counts dynamically. The data takes positive integer values, and as an example, one can think of the number of infection diseases, number of road accidents, number of bank failures. Frequently a Poisson distribution is used to model time series of counts. Even though in the univariate framework it fits the purpose well, the multivariate Poisson distribution brings some challenges in time series modeling of count data. The probability mass function of multivariate Poisson distribution is complex and makes estimation cumbersome. In particular, it slows down the estimation when there are high count values in the data. Moreover, in multivariate Poisson distribution, the covariance structure is modeled through a common Poisson random variable which leads to only positive contemporaneous correlation. It is also very restrictive to assume that all of the time series have the same covariance term and hence one needs to complicate the inference even further to make the analysis realistic. Karlis and Meligkotsidou (2005) discuss how to relax this assumption and propose models with covariance structure for every pair of variables. However, extending their framework to high-dimensional models would be hard. Also, some other issues have to be addressed in this model. For example, the model does not account for potential overdispersion – a property that is common for count data. The authors propose Bayesian and classical inferential approaches for multivariate Poisson regression and compare them. They find that computationally Bayesian and classical approaches are of the similar order. In this paper, we compare two classes of mod-

els rather than estimation approaches: observation driven and parameter driven models. Observation driven models are estimated using the quasi-maximum likelihood method. Parameter driven models, however, fit very well into the Bayesian paradigm and that is what we use for estimation. Certain advantages come together with this framework, such as naturally obtained from the posterior distribution uncertainty about parameters of the model and forecast of the multivariate time series (Zellner, 1971).

Other methods can also address the problem of modeling the joint distribution of count data. One of the frequently discussed methods in the literature is the copula approach. A number of papers has considered this strategy for various count time series models (Heinen and Rengifo, 2007; Nikoloulopoulos and Karlis, 2009; Andreassen, 2013; Lennon, 2016). Even though copulas seem to be an attractive way of modeling dependency in count time series, several issues arise in their applications for count data such as unidentifiability. Even though we do not discuss this method in detail, we give an overview for modeling time series of counts with copulas at the end of this paper and refer to Genest and Nešlehová (2007) for more details on the problems in applying copulas to count data.

Both approaches of modeling joint distribution of count data – multivariate Poisson distribution and copulas – can be incorporated in the autoregressive conditional intensity (ACI) framework, often also referred to as an integer-valued generalized autoregressive conditional heteroskedasticity model (INGARCH). This model belongs to the class of observation driven models as opposed to parameter driven models, the classification that was proposed by Cox et al. (1981). ACI models have been dominating the literature for a quite long time despite their restrictiveness: these models allow only for positive coefficients in the equation for conditional intensity. Bounded parameters lead to several problems besides potentially unrealistic dependence structure for some data. In particular, the problem of calculating confidence sets for the parameters that are close to or on the boundary rises and has not been yet solved in the literature. Another observation driven model that has been proposed as an alternative to ACI framework is log-linear model, see Fokianos and Tjøstheim (2011), a mul-

tivariate extension of which has been considered in Doukhan et al. (2017). Even though the problem of modeling joint distribution remains, the advantage of this approach is that no restrictions on the parameter space are required due to the log-transform of the data.

Another class of models that can be considered for modeling count data, but is rarely used in the literature is parameter driven models, and in particular non-linear state-space models. In this framework, the observations are driven by an independent unobserved stochastic process which, for instance, can be a (vector) autoregressive process ($VAR(p)$). These models have been discussed extensively in the univariate case, see for example Davis et al. (1999). However, multivariately these models are rarely used in applications due to computationally demanding methods that have to be used for the estimation. According to our knowledge, only one very recent study has considered them multivariately Zhang et al. (2017). Non-linear state-space models are capable of modeling and inferring complex dependence structures in the data. They allow for both negative and positive contemporaneous correlation, as well as for both negative and positive Granger-causal feedback. Thereby, these models avoid the problem of modeling joint distribution of time series of counts and provide a coherent inferential tool in the Bayesian framework. This is what distinguishes our approach from the approach discussed in Zhang et al. (2017) who consider frequentist estimation of these models. We also compare SSM to log-linear models instead of MACI models since they allow for negative dependence between the intensities and hence appear to be a more natural competitor of SSM models.

In this chapter, we focus on the multivariate comparison of autoregressive conditional intensity models (observation driven) and non-linear state-space models (parameter driven). We compare the models on the simulated data in Section 4.6 and on multiple empirical data sets in section 4.7. In particular, we consider data sets for car crash counts in the Netherlands, bank failure data for Russian and the U.S. and transaction data on financial markets. In section 4.8 we briefly discuss copula approach and section 4.9 concludes this chapter.

4.2 Count Time Series

Count time series appear in many scientific fields, for example, in finance we observe the number of transactions for a particular stock in a certain period of time. Other examples from economic applications include the number of strikes per month, the number of bank failures per year, the founding rate of national labor unions (Winkelmann, 2013). Schoenmaker (1996) discuss contagion risk in banking. Hence one can analyze the number of bank failures and corresponding dependencies between them in different countries. Other examples are infectious diseases data, sports data (such as a number of goals scored in soccer, for example), the number of earthquakes can be analyzed in seismology. The natural distribution to model count data is Poisson distribution.¹

4.2.1 Typical examples of count data

Count data appears in various fields of studies. Among others, the examples of these data are in economics – the number of bank failures, epidemiology – the incidence of infectious diseases, industrial control – counts of faults, criminology – the number of crimes, actuarial science – the number of claims filed, seismology – the number of earthquakes. We here will consider some examples of the count data and discuss particular properties that one desires to model when dealing with such data. Two standard data sets that have been used for univariate illustration and comparison are asthma presentations at a Sydney hospital and a monthly number of cases of poliomyelitis in the U.S. (1970 - 1983). Both data sets are coming from epidemiology. Figures 4.1– 4.6 illustrate different example of count data. Further, Karlis and Meligkotsidou (2005) consider an application from criminology. In particular, they look into four different types of criminal counts and explain them with the variables such as Gross Domestic Product, the unemployment rate, the level of urbanization of the city and location of

¹Sometimes, depending on a particular problem and model, also the negative binomial distribution can be used. The problem of choosing appropriate distribution in a regression for an ecological study is considered in Ver Hoef and Boveng (2007). A negative binomial version of INGARCH is discussed in Zhu (2011).

the city (border of the country or not). Park and Lord (2007) consider multivariate Poisson log-normal models with correlations for modeling the crash frequency by severity. The authors demonstrate that accounting for the correlations in the multivariate model one can improve the accuracy of the estimation.

4.2.2 Poisson distribution

The Poisson distribution has played an important role in modeling count time series data as its interpretation is the number of independent events that occur in a time period. These are the kind of data we will be considering in this paper. One example from economics can be a number of transactions for a given time interval, see the next section for more examples of the count data from different disciplines. Hence, a random variable x takes integer values in $\{0, 1, \dots\}$. The mean of the Poisson distribution, λ , describes the average occurrences per interval. Further, the probability mass function of the Poisson distribution reads

$$p(x) = \frac{\lambda^x e^{-\lambda}}{x!}, \quad x = 0, 1, 2, 3, \dots, \quad (4.1)$$

and $\mathbb{E}(x) = \text{Var}(x) = \lambda$. In this paper, we are interested in multivariate models and hence in the joint probability distribution of the random variables. When we have two Poisson random variables, say x_1 and x_2 , then the joint probability mass function reads

$$p(x_1, x_2) = \prod_{i=1}^2 \frac{e^{-\lambda_i} \lambda_i^{x_i}}{x_i!}. \quad (4.2)$$

This extension to multivariate Poisson distribution is quite naive since we assume independence between random variables. Such a model would ignore potential dependency of the real world data. Many of the count time series models take their origins in the idea of Poisson regression model, an extensive overview of these models is given in Fokianos (2012). Note, that Equation (4.2) does not account for possible correlation between X_1 and

Figure 4.1: Asthma presentations at a Sydney hospital.

Figure 4.3: Yearly bank failures in the U.S.

Figure 4.5: Number of car crashes near Utrecht.

Figure 4.2: Monthly number of poliomyelitis cases in the U.S.

Figure 4.4: Transaction counts for BMW.

Figure 4.6: Number of earthquakes.

Note: This figure illustrates typical examples of count data coming from applications in different scientific fields.

X_2 . One way to use the Poisson distribution for modeling count data in multivariate case and incorporate correlation structure is through the so-called *trivariate reduction* (Liu, 2012; Andreassen, 2013). The idea is that correlation can be modeled through the third Poisson variable. Assume we have three independent random variables $x_i \sim \text{Poisson}(\lambda_{t,i} - \varphi)$, where $0 \leq \varphi \leq \min\{\lambda_{t,1}, \lambda_{t,2}\}$. Define $Y_{t,1} = X_1 + X_2$ and $Y_{t,2} = X_2 + X_3$. In this way, the random variable X_2 is exploited to model the dependence between Y_1 and Y_2 . The restriction of this approach is that the correlation is the same between all the series (in case one wants to model the systems beyond bivariate case) and the dependence can only be positive. We further discuss the trivariate reduction technique in the context of ACI/INGARCH models. In particular, the difficulties of extending this to higher dimensions is of interest and presents one with a challenging task.

4.3 Observation driven models

In this section we consider two observation driven models: multivariate autoregressive conditional intensity model and log-linear analog of it. Both of these models are characterized by the dynamics that depends on the past of the process itself and some noise. Both models have been considered in Doukhan et al. (2017) where the authors discussed some theoretical properties and proposed to use copula approach for modeling joint count distribution. Copulas is a flexible tool for modeling dependence structure. However, there are some problems associated with their usage in count time series models which we discuss in section 4.8. We first will consider both models under independence assumption in the Poisson random variables, and at the end of this section, we discuss the extension of modeling multiple count time series with multivariate Poisson distribution.

4.3.1 MACI (INGARCH)

The Poisson integer-valued generalized autoregressive conditional heteroscedastic process (INGARCH) models (Ferland et al., 2006) are built upon GARCH

framework and are capable of analyzing time series of counts. In the literature, these models appear as multivariate autoregressive conditional intensity models (MACI). As for GARCH-type models, it is assumed that the conditional mean of the process at time t depends on the value of the process at period $t - 1$ and its conditional mean at time $t - 1$. The time series of counts follow Poisson process with the conditional mean λ_t , i.e.

$$X_{i,t} \mid \mathcal{F}_{t-1} \sim \text{Poisson}(\lambda_{i,t}), \quad i = 1, \dots, n, \quad (4.3)$$

The corresponding joint p.m.f. reads

$$P(X_{1t} = x_{1t}, \dots, X_{nt} = x_{nt} \mid \mathcal{F}_{t-1}) = \prod_{i=1}^n \frac{e^{-\lambda_{it}} \lambda_{it}^{x_{it}}}{x_{it}!}. \quad (4.4)$$

The dynamics of the conditional intensity $\lambda_t = \mathbb{E}[X_t \mid \mathcal{F}_{t-1}]$ follows

$$\lambda_t = \omega + \sum_{i=1}^n \mathbf{A}^i \lambda_{t-i} + \sum_{j=1}^q \mathbf{B}^j X_{t-j}. \quad (4.5)$$

Note, the elements of ω , $\mathbf{a}^i$, $\mathbf{b}^j$ are assumed to be positive to ensure positivity of the intensity process λ .² Note also that in the above model we assume no contemporaneous correlation in the counts. Consider the bivariate case for the conditional intensity process

$$\begin{bmatrix} \lambda_{1t} \\ \lambda_{2t} \end{bmatrix} = \begin{bmatrix} \omega_1 \\ \omega_2 \end{bmatrix} + \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} \lambda_{1t-i} \\ \lambda_{2t-i} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} \begin{bmatrix} X_{1t-j} \\ X_{2t-j} \end{bmatrix}, \quad t = 0, \pm 1, \pm 2, \dots \quad (4.6)$$

From Equation (4.6) it is clear that when $\mathbf{A}$ and $\mathbf{B}$ are diagonal, there is no dependence structure between the intensities. Further, when $a_{12} = 0$ and $b_{12} = 0$ then the intensity of the first process, $\lambda_{1,t}$, depends only on its own past while the second process can depend on the dynamics of the first one. Finally, if we restrict $\mathbf{A}$ being diagonal and $\mathbf{B}$ being non-diagonal

² Doukhan et al. (2017) argue that the condition $\|\mathbf{A} + \mathbf{B}\|_2 < 1$ guaranties stationarity.

then every intensity process would depend on its past and possibly on the past of all of the observations. This constraint is relevant when one wants to apply graphical modeling to this problem.

4.3.2 Quasi-maximum likelihood for MACI models

In this section, we discuss how the inference for MACI/INGARCH models can be executed. The details for the multivariate case have also been discussed in Doukhan et al. (2017). For these models, we will be working in classical estimation framework and in particular use quasi-maximum likelihood estimation. The conditional quasi-likelihood for this model and $\boldsymbol{\theta}$ reads

$$L(\boldsymbol{\theta}) = \prod_{t=1}^T \prod_{i=1}^n \left(\frac{\exp(-\lambda_{i,t}(\boldsymbol{\theta})) \lambda_{i,t}^{x_{i,t}}(\boldsymbol{\theta})}{x_{i,t}!} \right), \quad (4.7)$$

where $\boldsymbol{\theta}$ are the parameters of interest. It follows the the quasi log-likelihood function is

$$l(\boldsymbol{\theta}) = \sum_{t=1}^T \sum_{i=1}^n (x_{i,t} \log \lambda_{i,t}(\boldsymbol{\theta}) - \lambda_{i,t}(\boldsymbol{\theta})), \quad (4.8)$$

and the corresponding score function reads

$$\begin{aligned} S_T(\boldsymbol{\theta}) &= \sum_{t=1}^T \sum_{i=1}^n \left(\frac{x_{i,t}}{\lambda_{i,t}} - 1 \right) \frac{\partial \lambda_{i,t}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \\ &\sum_{t=1}^T \frac{\partial \boldsymbol{\lambda}_t^T(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \mathbf{D}_t^{-1}(\boldsymbol{\theta}) (\mathbf{X}_t - \boldsymbol{\lambda}_t(\boldsymbol{\theta})) \equiv \sum_{t=1}^T s_t(\boldsymbol{\theta}), \end{aligned} \quad (4.9)$$

where $\partial \boldsymbol{\lambda}_t \backslash \partial \boldsymbol{\theta}^T$ is $n \times d$ matrix with $d \equiv n(1 + 2n)$ being the dimension of the parameter vector $\boldsymbol{\theta}$, $\mathbf{D}_t$ is diagonal matrix with dimensions $n \times n$ and its diagonal elements are $\lambda_{i,t}(\boldsymbol{\theta})$, $i = 1, 2, \dots, n$, $\mathbf{X}_t$ consists of elements $x_{i,t}$, $i = 1, 2, \dots, n$, $t = 1, 2, \dots, T$. Thus the recursions for the quasi-maximum

likelihood estimation follow

$$\frac{\partial \boldsymbol{\lambda}_t}{\partial \boldsymbol{\omega}^T} = \mathbf{I}_n + \mathbf{A} \frac{\partial \boldsymbol{\lambda}_{t-1}}{\partial \boldsymbol{\omega}^T}, \quad (4.10)$$

$$\frac{\partial \boldsymbol{\lambda}_t}{\partial \text{vec}^T(\mathbf{A})} = (\boldsymbol{\lambda}_{t-1} \otimes \mathbf{I}_n)^T + \mathbf{A} \frac{\partial \boldsymbol{\lambda}_{t-1}}{\partial \text{vec}^T(\mathbf{A})}, \quad (4.11)$$

$$\frac{\partial \boldsymbol{\lambda}_t}{\partial \text{vec}^T(\mathbf{B})} = (\mathbf{X}_{t-1} \otimes \mathbf{I}_n)^T + \mathbf{A} \frac{\partial \boldsymbol{\lambda}_{t-1}}{\partial \text{vec}^T(\mathbf{B})}. \quad (4.12)$$

Finally, the Hessian matrix and the conditional information matrix are correspondingly

$$\begin{aligned} \mathbf{H}_T(\boldsymbol{\theta}) = & \sum_{t=1}^T \sum_{i=1}^n \frac{x_{i,t}}{\lambda_{i,t}^2(\boldsymbol{\theta})} \frac{\partial \lambda_{i,t}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \frac{\partial \lambda_{i,t}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^T} - \\ & \sum_{t=1}^T \sum_{i=1}^n \left(\frac{x_{i,t}}{\lambda_{i,t}(\boldsymbol{\theta})} - 1 \right) \frac{\partial^2 \lambda_{i,t}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T}, \end{aligned} \quad (4.13)$$

$$\mathbf{G}_T = \sum_{t=1}^T \frac{\partial \boldsymbol{\lambda}_t^T(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \mathbf{D}_t^{-1}(\boldsymbol{\theta}) \boldsymbol{\Sigma}_t \mathbf{D}_t^{-1}(\boldsymbol{\theta}) \frac{\boldsymbol{\lambda}_t(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^T}. \quad (4.14)$$

Further, one can show that $\mathbf{S}_n(\boldsymbol{\theta}) = 0$ has a unique solution, and moreover this solution $\hat{\boldsymbol{\theta}}$ is strongly consistent and asymptotically normal, for more details on the proof we refer the reader to Doukhan et al. (2017). Note however, that theoretical properties of $\hat{\boldsymbol{\theta}}$ are proven under assumption that the true value $\boldsymbol{\theta}_0$ belongs to the interior of the parameter space $\boldsymbol{\Theta}$. The problems certainly arise when the true parameter is close or on the boundary of the parameter space. Dealing with the theoretical problems of the constrained optimization and parameters near or on the boundary of parameter space is out of the scope of this paper and generally establishing the theory for this case is a complicated task. One of the possible solu-

tions is to exploit bootstrap methods for this task. In particular, we refer the interested reader to Hilmer et al. (2000) for comparison of some bootstrap methods related to this sort of problem and review of other possible approaches.

4.3.3 Log-linear autoregressive model

Log-linear models have appeared in the count data literature in the recent years (Doukhan et al., 2017) and have good potential since they allow for both positive/negative correlation and avoid parameter boundary problems which MACI models suffer from.

$$X_{i,t} \mid \mathcal{F}_{t-1}^{\mathbf{X}, \boldsymbol{\lambda}} \sim \text{Poisson}(\lambda_{i,t}), \quad (4.15)$$

$$\boldsymbol{\nu}_t = \boldsymbol{\omega} + \mathbf{A}\boldsymbol{\nu}_{t-1} + \mathbf{B} \log(\mathbf{X}_{t-1} + \mathbf{1}_n), \quad t \geq 1, \quad (4.16)$$

where $\mathcal{F}_{t-1}^{\mathbf{Y}, \boldsymbol{\lambda}}$ is the σ -field generated by $\{\mathbf{X}_0, \dots, \mathbf{X}_t, \boldsymbol{\lambda}_0\}$, $\mathbf{1}_n$ is the n -dimensional vector of ones, $\boldsymbol{\nu}_t \equiv \log \boldsymbol{\lambda}_t$. Note, that for this model parameters $\boldsymbol{\omega}$, $\mathbf{A}$, and $\mathbf{B}$ do not have to be positive which makes this models more attractive than MACI.

4.3.4 Quasi-maximum likelihood for log-linear models

The inference in log-linear models is very similar to the quasi-maximum likelihood approach derived for MACI models in Section 4.3.2. Only minor adjustments have to be made in corresponding recursions (Doukhan et al., 2017). In particular, the score function for the log-linear model reads

$$S_T(\boldsymbol{\theta}) = \sum_{t=1}^T \sum_{i=1}^n (x_{i,t} - \exp(\nu_{i,t}(\boldsymbol{\theta}))) \frac{\partial \nu_{i,t}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \sum_{t=1}^T \frac{\partial \boldsymbol{\nu}_t^T(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} (\mathbf{X}_t - \exp(\boldsymbol{\nu}_t(\boldsymbol{\theta}))), \quad (4.17)$$

the Hessian matrix is

$$\begin{aligned} \mathbf{H}_T(\boldsymbol{\theta}) = & \sum_{t=1}^T \sum_{i=1}^n \exp(\nu_{i,t}(\boldsymbol{\theta})) \frac{\partial \nu_{i,t}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \frac{\partial \nu_{i,t}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^T} - \\ & \sum_{t=1}^T \sum_{i=1}^n (x_{i,t} - \exp(\nu_{i,t}(\boldsymbol{\theta}))) \frac{\partial^2 \nu_{i,t}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T}, \end{aligned} \quad (4.18)$$

finally the conditional information matrix for the log-linear model reads

$$\mathbf{G}_T(\boldsymbol{\theta}) = \sum_{t=1}^T \sum_{i=1}^n \exp(\nu_{i,t}(\boldsymbol{\theta})) \frac{\partial \nu_{i,t}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \frac{\partial \nu_{i,t}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^T}. \quad (4.19)$$

Doukhan et al. (2017) prove theoretical properties of this model. In particular they show that there exists a unique solution $\hat{\boldsymbol{\theta}}$ which is strongly consistent and asymptotically normal. The authors also show that the condition $\sum_{j=0}^{\infty} \|\mathbf{A}^j \mathbf{B}\|_2 < 1$ guarantees both stationarity and weak dependence.

4.3.5 Multivariate Poisson Distribution

To allow for contemporaneous correlation, we need to use trivariate reduction technique discussed before. We further consider bivariate case to give an example, assume that there are three independent random variables Y_1, Y_2, Y_3 with positive means $\lambda_1, \lambda_2, \lambda_3$ respectively. Define random variables $X_1 = Y_1 + Y_3$ and $X_2 = Y_2 + Y_3$. The new random variables will have means $\lambda_1 + \lambda_3$ and $\lambda_2 + \lambda_3$, where λ_3 would also correspond to the covariance between X_1 and X_2 . The covariance is clearly restricted to be positive, while correlation will lie between 0 and $\min\{\frac{\sqrt{\lambda_1 + \lambda_3}}{\sqrt{\lambda_2 + \lambda_3}}, \frac{\sqrt{\lambda_2 + \lambda_3}}{\sqrt{\lambda_1 + \lambda_3}}\}$. Thereby the joint p.m.f. of interest (alternative to what we have in Equation (4.3)) becomes

$$\begin{aligned} P(X_{1t} = x_{1t}, X_{2t} = x_{2t} \mid \mathcal{F}_{t-1}) = \\ e^{-(\lambda_1 + \lambda_2 + \lambda_3)} \frac{\lambda_1^{x_1} \lambda_2^{x_2}}{x_1! x_2!} \sum_{i=0}^{\min(x_1, x_2)} \binom{x_1}{i} \binom{x_2}{i} i! \left(\frac{\lambda_3}{\lambda_1 \lambda_2} \right)^i. \end{aligned} \quad (4.20)$$

Extension of these approach for modeling contemporaneous correlation in higher dimensions is non-trivial. Assume that we would like to model n Poisson random variables, thus $X_i \sim \text{Poisson}(\lambda_i)$, $i = 1, \dots, n$. By defining a random variable $X_0 \sim \text{Poisson}(\lambda_0)$ and extending the argument of the trivariate reduction we can define random variables $X_1 = Y_1 + Y_0$, $X_2 = Y_2 + Y_0, \dots, X_n = Y_n + Y_0$. The joint p.m.f. is then

$$P(X_1 = x_1, X_2 = x_2, \dots, X_n = x_n) = \exp\left(-\sum_{i=1}^n \lambda_i\right) \prod_{i=1}^n \frac{\lambda_i^{x_i}}{x_i!} \sum_{i=0}^m \sum_{j=1}^n \binom{x_j}{i} i! \left(\frac{\lambda_0}{\prod_{i=1}^n \lambda_i}\right)^i, \quad (4.21)$$

where $m = \min(x_1, x_2, \dots, x_n)$. Note, that this approach assumes that the covariance is the same for all the pairs of Poisson random variables which is very restrictive. Karlis and Meligkotsidou (2005) consider the case with richer covariance structure which we discuss further. For simplicity, assume we want to model trivariate time series of counts Y_1, Y_2, Y_3 . As before, let us specify through X_i and X_{ij} univariate Poisson random variables, i.e. $X_i \sim \text{Poisson}(\lambda_i)$ and $X_{ij} \sim \text{Poisson}(\lambda_{ij})$ with $i, j \in \{1, 2, 3\}$, $i < j$. Then the random variables Y_i with $i \in \{1, 2, 3\}$ are defined in the following way

$$\begin{aligned} Y_1 &= X_1 + X_{12} + X_{13}, \\ Y_2 &= X_2 + X_{12} + X_{23}, \\ Y_3 &= X_3 + X_{13} + X_{23}. \end{aligned} \quad (4.22)$$

Thus, $Y_i \sim \text{Poisson}(\lambda_i + \lambda_{ij} + \lambda_{ik})$, where $i, j, k \in \{1, 2, 3\}$, $i \neq j \neq k$. Finally, these random variables follow the Poisson distribution with $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \lambda_3, \lambda_{12}, \lambda_{13}, \lambda_{23})$, and hence with the mean vector $\mathbf{A}\boldsymbol{\lambda} = (\lambda_1 + \lambda_{12} + \lambda_{13}, \lambda_2 + \lambda_{12} + \lambda_{23}, \lambda_3 + \lambda_{13} + \lambda_{23})'$. The variance-covariance matrix for this distribution is given by

$$\mathbf{A}\boldsymbol{\Sigma}\mathbf{A}' = \begin{pmatrix} \lambda_1 + \lambda_{12} + \lambda_{13} & \lambda_{12} & \lambda_{13} \\ \lambda_{12} & \lambda_2 + \lambda_{12} + \lambda_{23} & \lambda_{23} \\ \lambda_{13} & \lambda_{23} & \lambda_3 + \lambda_{13} + \lambda_{23} \end{pmatrix} \quad (4.23)$$

It is clear from the above examples that modeling of the time series of counts with multivariate Poisson distribution in higher dimensions is restrictive and cumbersome. It is restrictive since it allows only for positive dependency in the data which can be unreasonable for real-world applications. It is cumbersome since the method is only computationally tractable when one has low counts data (see Equation 4.21 in which the number of terms in the sum depends on the number of observed counts). The methods, such as expectation maximization, can be applied in this case. However, they are not trivial and stable in the case of high counts. Moreover, in this case, incorporation of the multivariate Poisson distribution into MACI or log-linear models also affects computational speed significantly, and they lose their attractiveness over more complex models such as nonlinear state-space models that we discuss in the next section.

4.4 Parameter driven model: nonlinear state-space model

The advantage of parameter driven models is clear interpretability of the model parameters and the high degree of flexibility. The model can easily incorporate different distributions and extends easily to the multivariate framework. Moreover, in the Bayesian framework, we have coherent inferential tools derived from the posterior distributions of the parameters, such as highest posterior density intervals. These models also take into account uncertainty about the parameters which is incorporated into predictions. The disadvantage of this approach is challenging estimation procedures that are computationally intensive and hence even though theoretically estimation methodologies are possible to extend to any dimension, in practice that is not feasible due to the time constraints. In this paper, we estimate the parameter-driven model for multivariate count data using Sequential Monte Carlo and particle Markov Chain Monte Carlo. These methods became popular with the availability of more computational power. They are restricted in some ways, and we will discuss these restrictions in the next subsections after introducing the nonlinear state space model (SSM) which

we will compare to the observation driven models.

4.4.1 Multivariate SSM

A state-space model is usually presented by an observation equation and a state equation. The state equation represents a latent process, say h_t , which drives the dynamics of observations y_t . In the model we are considering onwards this dependence between the observations and the state is nonlinear. The multivariate state-space model for count data is defined in the following way

$$X_{it} \sim \text{Poisson}(\lambda_{it}), \quad i = 1, 2, \dots, n \quad (4.24)$$

$$\lambda_t = \beta \exp(h_t) \quad (4.25)$$

$$h_t = \sum_{i=1}^n \Phi^i h_{t-i} + \eta_t, \quad \Sigma_\eta = \begin{pmatrix} \sigma_{\eta_1}^2 & \rho_{\eta_{12}} & \cdots & \rho_{\eta_{1n}} \\ \rho_{\eta_{21}} & \sigma_{\eta_2}^2 & \cdots & \rho_{\eta_{2n}} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{\eta_{n1}} & \cdots & \cdots & \sigma_{\eta_n}^2 \end{pmatrix}, \quad (4.26)$$

where $\eta_t \sim N(0, \Sigma_\eta)$. Equation (4.24) shows that the observations have Poisson distribution with mean λ_t defined through the Equation (4.25), and λ_t nonlinearly depends on the latent process h_t which is defined through Equation (4.26). Note that the latent process is defined through a VAR(p) process, and hence corresponding theory applies. In particular, the stationarity condition is that the roots of the Equation (4.27) must lie outside the unit circle

$$|\lambda^p I_n - \lambda^{p-1} \Phi_1 - \dots \Phi_p| = 0. \quad (4.27)$$

Note, the dependence structure between the counts is modeled through the dependence in the latent process. Conditioned on the latent process $\{\mathbf{h}_t\}_{t=1}^T$ the observations $\{\mathbf{X}_t\}_{t=1}^T$ are independent. Further, since the latent process of the model is a VAR(p), we can account for various dependence structures: positive and negative contemporaneous correlation, and positive and negative Granger-causal feedback.

These models are challenging to estimate, and assumption of $p = 1$ can simplify the inference.³ Bivariate specification of the nonlinear state-space model with the lag $p = 1$ reads

$$X_{it} \sim \text{Poisson}(\lambda_{it}), \quad i = 1, 2 \quad (4.28)$$

$$\lambda_{it} = \beta_i \exp(h_{it}), \quad i = 1, 2, \quad (4.29)$$

$$\begin{pmatrix} h_{1,t+1} \\ h_{2,t+1} \end{pmatrix} = \begin{pmatrix} \phi_{11} & \phi_{12} \\ \phi_{21} & \phi_{22} \end{pmatrix} \begin{pmatrix} h_{1,t} \\ h_{2,t} \end{pmatrix} + \begin{pmatrix} \eta_{1t+1} \\ \eta_{2t+1} \end{pmatrix}, \quad \Sigma_\eta = \begin{pmatrix} \sigma_{\eta_1}^2 & \rho_{\eta_{12}} \\ \rho_{\eta_{21}} & \sigma_{\eta_2}^2 \end{pmatrix}, \quad (4.30)$$

The dependence between the count data is described by the Granger-causal relationship in the latent processes h_{it} and contemporaneous relations that are incorporated in Σ_η . For example, we say that $h_{2,t}$ does not Granger-cause $h_{1,t}$ if $\phi_{12} = 0$. Correlation coefficient ρ in this model allows us to model both positive and negative correlation between the counts.

4.4.2 Bayesian Inference

Estimation of the nonlinear state-space models naturally fits into the Bayesian framework. A method that can be employed for the estimation is particle Markov Chain Monte Carlo (PMCMC) introduced in Andrieu et al. (2010).

³For extending the model to $p > 1$ we advise the reader to consider using sparse priors, such as Minnesota prior, spike and slab or horseshoe prior.

Figure 4.7: **Simulated Data with SSM.**

Note: This is the example of the data simulated with SSM. The sample size is $T=1000$, the data were generated with the parameters $\beta_1 = 2$, $\beta_2 = 3$, $\phi_{11} = 0.8$, $\phi_{21} = 0.5$, $\phi_{12} = 0.0$, $\phi_{22} = 0.8$ $\sigma_{\eta_{11}} = 0.01$, $\sigma_{\eta_{22}} = 0.04$.

The presence of the unobservable process in the model and nonlinear dependence of the observations on this process leads to the intractability of the likelihood. Hence, Sequential Monte Carlo methods, also referred to as particle filtering, can be used for the estimation of the likelihood sequentially. An extensive introduction to nonlinear state-space models and particle filtering can be found, for example, in Särkkä (2013). Recall the Bayes rule on which the inference is based

$$p(h_{1:T}|x_{1:T}) = \frac{p(x_{1:T}|h_{1:T})\pi(h_{1:T})}{p(x_{1:T})}, \quad (4.31)$$

where $\pi(h_{1:T})$ is the prior distribution on the parameters of the volatility process defined by the dynamic model, $p(x_{1:T}|h_{1:T})$ is the likelihood of the observations, $p(x_{1:T})$ is the normalization constant that is ignored during the inference, i.e. we deal with proportionality rather than equality in the Equation (4.31). In complex models, the estimation of the posterior distribution is usually carried out using algorithms such as Metropolis-Hastings (see Algorithm 19). Numerical methods are required since the corresponding distributions are not available in the closed form and hence the posterior samples cannot be sampled directly. Metropolis-Hastings belongs to the class of Markov Chain Monte Carlo methods that are defined as follows by Robert and Casella (2005)

Definition 4.1. A Markov Chain Monte Carlo (MCMC) method for the simulation of a distribution f is any method producing an ergodic Markov chain $(X^{(t)})$ whose stationary distribution is f .

In the Algorithm 19 $f(\cdot)$ is target distribution and $q(\cdot)$ is proposal distribution, and $\rho(\theta, \theta^c)$ is called Metropolis-Hastings ratio. How well we manage to explore the posterior depends on both the proposal distribution and the acceptance probability which are interconnected with each other. When the acceptance rate is too high it is often related to the proposals steps are too small, and the other way around. Overall, either case slows down the convergence of the Markov Chain. General advice for the optimal performance of the algorithm is an acceptance rate that is around

0.234 (Roberts et al., 1997).

Estimation of the likelihood with SMC

As we noted before, the flexibility of the nonlinear state-space models comes at a price of computationally expensive numerical methods that have to be used for the inference. The likelihood of the model, see Equation (4.32), is not available in closed form

$$\begin{aligned} L(x_{1:T}) &= \int p(x_{1:T}, h_{1:T}) dh_{1:T} = \int p(x_{1:T} | h_{1:T}) p(h_{1:T}) dh_{1:T} \\ &= \int p(x_1 | h_1) p(h_1) \prod_{t=2}^T p(x_t | h_t) p(h_t | h_{t-1}) dh_1 \dots h_T. \end{aligned} \quad (4.32)$$

However, the likelihood of nonlinear state-space models can be estimated with Sequential Monte Carlo (SMC) method, also known as particle filters. Algorithm 20 represents a simple version of a particle filter. In that case proposals for the particles are made according to the dynamics of the model, and then the weights are estimated for every particle. As standard particle filters suffer from a problem called degeneracy of the particles, which means that in the sequential procedure the filter tends to end up with only a few particles that have high weights, it is crucial to perform resampling

Algorithm 19 Metropolis-Hastings Algorithm

- 1: Given $\theta^{(t)}$,
- 2: Generate $\theta_t^c \sim q(y | \theta^{(t)})$,
- 3: Take

$$\theta^{(t+1)} = \begin{cases} \theta_t^c, & \text{with probability } \rho(\theta^{(t)}, \theta_t^c) \\ \theta^{(t)} & \text{with probability } 1 - \rho(\theta^{(t)}, \theta_t^c), \end{cases}$$

where

$$\rho(\theta^{(t)}, \theta_t^c) = \min \left(\frac{f(\theta_t^c)}{f(\theta^{(t)})} \frac{q(\theta^{(t)} | \theta_t^c)}{q(\theta_t^c | \theta^{(t)})}, 1 \right)$$

step that is given in the Algorithm 21. The approximate marginal likelihood of the model is a by-product of particle filtering and Alrorithm 20 in particular, it can be estimated as follows

$$p(x_{1:T}|\theta) \approx \prod_t \hat{p}(x_t|x_{1:t-1}, \theta). \quad (4.33)$$

For the purpose of numerical stability one usually estimates the log-likelihood instead

$$\log p_\theta(x_{1:T}) = \log p_\theta(x_1) + \sum_{t=2}^T \log p_\theta(x_t|x_{1:t-1}), \quad (4.34)$$

where $p_\theta(x_t|x_{1:t-1}) = \frac{1}{N} \sum_{i=1}^N \omega_t^{(i)}$ with N being the number of particles. Further, the estimated log-likelihood can be used inside Markov Chain Monte Carlo and in particular in Metropolis-Hastings algorithm. This combination of SMC and MCMC is referred to as particle MCMC and

Algorithm 20 Bootstrap Particle Filter

- 1: Draw a new point $h_t^{(i)}$ for each point in the sample set $\{h_{k-1}^{(i)} : i = 1, \dots, N\}$ from the dynamic model:

$$h_t^{(i)} \sim p(h_t|h_{t-1}^{(i)}), \quad i = 1, \dots, N.$$

- 2: Calculate the weights

$$\omega_t^{(i)} \sim p(x_t|h_t^{(i)}), \quad i = 1, \dots, N,$$

and normalize them to sum to unity.

- 3: Do resampling.
-

Algorithm 21 Resampling

- 1: Interpret each weight $\omega_t^{(i)}$ as the probability of obtaining the sample index i in the set $\{h_t^{(i)} : i = 1, \dots, N\}$.
 - 2: Draw N samples from that discrete distribution defined by the weights and replace the old samples set with this new one.
-

was introduced in Andrieu et al. (2010). Algorithm 22 presents the particle Metropolis-Hastings algorithm where $\hat{p}_{\theta^*}(x_{1:T})$ is estimated using Algorithm 20 with systematic resampling of the particles. General idea of resampling is depicted in Algorithm 21. Resampling step is usually necessary to avoid particle degeneracy. Without resampling step this problem usually appears after a few iterations and results in almost all particles having zero or nearly zero weights (Särkkä, 2013).

Further, the idea of particle Metropolis-Hastings is that the vector of the parameters θ is updated using random-walk proposal, in our case it is $N(\theta_{(i-1)}, \Sigma_\theta)$ where Σ_θ defines the step of the proposal. In comparison to the Algorithm 15 instead of the likelihood we use the estimate of the likelihood with SMC. Note, that the performance of the Metropolis-Hastings now depends not only on the proposal (and acceptance rate), but also on the variance of the estimated likelihood. When this variance is too big, we would not be able to explore the parameter space sufficiently well.

PMH and block updating

Details of implementation of the algorithms discussed above can affect performance and computational speed significantly. One implementation detail is concerned with whether to update one parameter at a time, all pa-

Algorithm 22 Particle Metropolis-Hastings

- 1: Initialize algorithm, $i = 0$
- 2: Set $\theta(0)$ arbitrary and
- 3: Run an SMC algorithm targeting $p_{\theta(0)}(h_{1:T}|x_{1:T})$, sample $h_{1:T}(0) \sim \hat{p}_{\theta(0)}(\cdot|x_{1:T})$ and let $\hat{p}(x_{1:T})$ denote the marginal likelihood estimate, and
- 4: with probability

$$1 \wedge \frac{\hat{p}_{\theta^*}(x_{1:T})p(\theta^*)}{\hat{p}_{\theta_{(i-1)}}(x_{1:T})p(\theta_{(i-1)})} \frac{q(\theta(i-1)|\theta^*)}{q(\theta^*|\theta(i-1))} \quad (4.35)$$

set $\theta(i) = \theta'^*$, $h_{1:T}(i) = h_{1:T}^*$ and $\hat{p}_{\theta(i)}(x_{1:T}) = \hat{p}_{\theta^*}(x_{1:T})$; otherwise set $\theta(i) = \theta(i-1)$, $h_{1:T}(i) = h_{1:T}(i-1)$ and $\hat{p}_{\theta(i)}(x_{1:T}) = \hat{p}_{\theta(i-1)}(x_{1:T})$.

parameters at the same time or perform block updating. We update the parameters of the model in multiple blocks which allows exploring the parameter space in less time while still achieving sufficient acceptance rate in the Metropolis-Hastings algorithm. We use random-walk proposal in the Metropolis-Hastings algorithm with the proposal steps defined by matrices Σ^β , Σ^ϕ , Σ^{σ_η} and Σ^ρ . The parameters are defined on a certain support. In particular, $\beta > 0$, Φ should satisfy the stationarity constraint of an autoregressive vector process, Σ_η should be positive semidefinite. When the proposal candidate θ^c does not satisfy these conditions, it is rejected without evaluation of the likelihood and the chain moves to the next step. The proposal distributions can be chosen differently, for example, for a variance-covariance matrix it is natural to choose the Inverse-Wishart proposal.

$$p(\Sigma_\eta(i)) \sim IW(\tau + 3, \tau \Sigma_\eta(i - 1)), \quad (4.36)$$

In this case, one must take into account that this distribution is not symmetric and adjust the acceptance probability in the Algorithm 22 correspondingly. In our simulation studies, we notice that for a bivariate model normal proposal works better. However, in a higher dimension it might be beneficial to adopt the algorithm to Inverse-Wishart proposal: when the parameter space is large one might run into a problem of high rejection rates due to an inadequate proposal distribution that fails to propose values for the variance-covariance matrix that satisfies the condition of positive semidefiniteness.

Smoothing

When one deals with time series with a large sample size smoothing can become a useful inferential tool. The particle filter provides us with the sequence of distributions $p(h_t|x_t)$, however due to particle degeneracy problem discussed previously, sampling from $p(h_{1:T}|x_{1:T})$ and approximating $p(h_k|x_{1:T})$, $k = 1, \dots, T$, is inefficient. One of the possible solutions to this problem is using so called forward filtering - backward smoothing recursions (Doucet and Johansen, 2009). The algorithms starts with sampling

$h_T^* \sim \hat{p}(h_T|x_{1:T})$, and then backwards for $k = T-1, T-2, \dots, 1$, we sample $h_k^* \sim \hat{p}(h_k|h_{k+1}^*, x_{1:n})$. Then we can approximate the distribution $\hat{p}(h_k|x_{1:T})$ as follows

$$\begin{aligned} \hat{p}(h_k|x_{1:T}) &= \sum_{i=1}^N W_k^i \times \\ &\left[\sum_{j=1}^N W_{k+1}^j \frac{f(h_{k+1}^{*,j}|h_k^{*,i})}{[\sum_{l=1}^N W_k^l f(h_{k+1}^{*,j}|h_k^{*,l})]} \delta_{h_k^{*,i}}(h_k) \right] = \\ &\sum_{i=1}^N W_{k|T}^i \delta_{h_k^{*,i}}(h_k). \end{aligned} \quad (4.37)$$

The smoothing comes at cost of $O(NT)$ operations to sample a path from $p(h_{1:T}|x_{1:T})$ and $O(N^2T)$ operations to approximate $p(h_k|x_{1:T})$. This method works very well. However, its performance comes at the price of high computational costs. Thereby, it is generally recommended to use it when the sample size of the data is large and hence Sequential Monte Carlo is more likely to suffer from particle degeneracy. There exist other methods that are computationally less expensive (Doucet and Johansen, 2009). However, in higher dimensions, they would be less reliable, and it would be recommendable to use more expensive methods.

4.4.3 Illustrative example with bivariate simulated data

This section illustrates the performance of the method on the simulated data which means we know the ground truth and thus can assess how well parameters of the model are estimated. We take the following parameters for the DGP of the simulated data: $\beta_1 = 1$, $\beta_2 = 2$, $\phi_{11} = 0.5$, $\phi_{21} = 0.3$, $\phi_{12} = 0.0$, $\phi_{22} = 0.5$, $\sigma_{\eta_1} = 0.5$, $\sigma_{\eta_2} = 0.5$, $\rho = 0.3$. In this case we have contemporaneous correlation between two time series and Granger-causality from process 1 to 2, i.e. $1 \rightarrow 2$. Further, during estimation we define the proposal is done separately for the blocks of the parameters β_i , $i=1,2$, ϕ_{ij} , $i,j=1,2$, σ_{η_i} , $i=1,2$, and ρ . Corresponding proposal steps in the particle Metropolis-Hastings algorithms are taken as follows $\Sigma^\beta =$

$\begin{pmatrix} 0.1 & 0.0 \\ 0.0 & 0.1 \end{pmatrix}$, $\Sigma^\phi = \begin{pmatrix} 0.00512 & 0.0092 \\ 0.0092 & 0.00152 \end{pmatrix}$, $\Sigma^{\sigma_\eta} = \begin{pmatrix} 0.001 & 0.0 \\ 0.0 & 0.001 \end{pmatrix}$, $\sigma^\rho = 0.009$. These step sizes guarantee acceptance rate between 20 and 40 %.

Table 4.1: Descriptives of the simulated data set.

	x_1	x_2
mean	2.6545	2.612
median	2	2
st.d.	2.3961	2.9938
dispersion	0.8148	1.3137
minimum	0	0
maximum	16	38

Note: In this table you can see descriptive statistics for the simulated data example of this section. We are considering bivariate count time series $x = (x_1, x_2)$.

Figure 4.8: Trace plots, histograms and acf for β_1, β_2 .

Note: This figure shows trace plots of the MCMC chains for parameters β_1, β_2 as well as the corresponding histograms and autocorrelation functions. The red line indicates the value of the parameter with which the data were generated. The autocorrelation indicates quite good mixing of the Markov chains.

Figure 4.9: Trace plots, histograms and acf for ϕ_{11} , ϕ_{21} , ϕ_{12} , ϕ_{22} .

Note: This figure shows trace plots of the MCMC chains for parameters ϕ_{11} , ϕ_{21} , ϕ_{12} and ϕ_{22} respectively. Figures in the middle are corresponding histograms and figures on the right hand side autocorrelation functions. The red line indicates the value of the parameter with which the data were generated. The autocorrelation indicates quite good mixing of the Markov chains.

Figure 4.10: Trace plots, histograms and acf for $\sigma_{\eta_{11}}$, $\sigma_{\eta_{22}}$, ρ .

Note: This figure shows trace plots of the MCMC chains for parameters σ_{11} , σ_{22} , ρ . Figures in the middle are corresponding histograms and figures on the right hand side autocorrelation functions. The red line indicates the value of the parameter with which the data were generated. The autocorrelation indicates quite good mixing of the Markov chains.

4.4.4 Forecasting with SSM

One of the interests of statistical inference can be the ability to perform forecasting exercises and thus handling the uncertainty about the future in the best possible way. In this section, we will discuss how forecasting task fits into the Bayesian framework, and in particular how it can be done for models of our interest.

Remember that we estimated multivariate SSM model for count data in the Bayesian framework. Observing the data $x = (x_1, \dots, x_T)$ we estimated the posterior distribution of the parameters in our model using particle Markov Chain Monte Carlo methods $p(\theta|x)$. Assume that we are interested in predicting the next s observations, i.e. $x_{T+1}, \dots, x_{T+s}$. First, note that the prediction equation for the next step reads

$$p(x_{t+1}|x_t) = \int p(x_{t+1}|h_{t+1})p(h_{t+1}|x_t)dh_t. \quad (4.38)$$

Table 4.2: **Posterior results for the simulated data example where we know the ground truth.**

	<i>Mean</i>	<i>Median</i>	<i>Mode</i>	95% <i>HPD_l</i>	95% <i>HPD_u</i>	<i>True</i>
β_1	1.0435	1.0418	1.0570	0.9036	1.1988	2
β_2	1.0181	1.0114	1.1291	0.7945	1.2355	3
ϕ_{11}	0.5260	0.5268	0.6252	0.2923	0.7403	0.5
ϕ_{21}	0.4062	0.3973	0.5010	0.1573	0.6655	0.3
ϕ_{12}	-0.0085	-0.0114	-0.0225	-0.1399	0.1171	0.0
ϕ_{22}	0.5823	0.5898	0.5602	0.4211	0.7169	0.5
σ_{η_1}	0.6261	0.6296	0.5430	0.4928	0.7427	0.5
σ_{η_2}	0.5555	0.5539	0.5537	0.4531	0.6602	0.5
ρ	0.2148	0.2132	0.2045	0.1269	0.2924	0.3

Note: This table summarizes posterior distributions of the parameters and provides relevant summary statistics. The most right column indicates the values of the parameters with which the data were generated. We observe that the values of the parameters Φ always lie inside the highest posterior density intervals. The same holds for the variance of the latent processes σ_{11} and σ_{22} . The scale parameters β_1 and β_2 are underestimated. The correlation between the latent processes, ρ , is also slightly underestimated.

In the framework of particle Markov Chain Monte Carlo, it is natural to adopt sequential nature of SMC and the fact that we obtain posterior draws in the MCMC part of the algorithm. Thereby, for every vector θ of the parameters drawn in the MCMC, we can propagate the particles obtained at time T and based on those make one-step ahead forecast. The similar idea extends to s -steps ahead forecasts. In this was the uncertainty about the parameters is included in the forecasts.

When forecasting, the most natural but not necessarily the easiest thing to do would be to update the posterior distribution every time we get a new observation. It would mean that we would have as many MCMC chains as we have steps for forecasting. It can be done straightforwardly by reestimating posterior distribution every time or more efficiently by incorporating this into SMC framework. However, if our training data set is sufficiently large, adding an extra estimation into the PMCMC framework should not change the results significantly. Ignoring this update also makes more comparable forecasting performance of the frequentist and Bayesian approaches.

4.5 Model comparison and prediction assessment

As the goal of this paper is to compare some multivariate models of counts, it is important to choose measures with which we can compare them. We are in particular interested in comparing observation and parameter driven approaches. Observation driven models in this comparison are represented by log-linear autoregressive model. It is more flexible than MACI model since it can account for negative correlation and thus it is a fairer competitor. Parameter driven approach is the state-space model where observations are generated from Poisson distribution and dependency is modeled through a latent process. Note that for the latter framework we follow a fully Bayesian approach. Thereby, we compare these two classes of models by model fit and forecasting performance criteria. The standard measures to access the model fit and forecast accuracy would be Mean Squared Error (MSE) and Mean Absolute Error (MAE) defined in Equation (4.39)

Figure 4.11: **Visual representation of forecasting exercise for stochastic volatility model.**

Note: This figure shows how we perform forecasting with the State-Space model for count data. In particular, one can see that we do not re-estimate the posterior distribution every time we receive a data point. This is due to two reasons. First, re-estimating the posterior distribution is quite expensive. Second, it is more natural for our goal since we would like to compare forecasting performance of two different models: state-space model and log-linear version of multivariate autoregressive conditional intensity model. Both frameworks are estimated in different paradigms. While SSM naturally fits into the Bayesian paradigm, the log-linear model is usually estimated using frequentist methods (quasi-maximum likelihood in this case). Since our goal is not to compare the two approaches to statistics, this design of forecasting exercise is fairer.

respectively.

$$MSE = \frac{1}{s} \sum_{i=1}^s (x_i - \hat{x}_i)^2, \quad MAE = \frac{1}{s} \sum_{i=1}^s (x_i - \hat{x}_i). \quad (4.39)$$

In Czado et al. (2009) the authors propose to compare forecast performance using some scoring rules. To define scoring rules, let P be the predictive distribution and x be the counts; then the penalty is defined through $s(P, x)$. Table 4.3 presents some of the scoring rules one can use for comparing the performance of count data models. Note, that in practice, one calculates the mean score

$$S = \frac{1}{n} \sum_{i=1}^n s(P^{(i)}, x^{(i)}). \quad (4.40)$$

Table 4.3: **Scoring rules for assessment of the forecasts.**

Logarithmic score	$\log(P, x) = -\log p_x$
Quadratic score	$qs(P, x) = -2p_x + \ p\ ^2$
Spherical score	$sphs(P, x) = -\frac{p_i}{\ p\ },$ where $\ p\ ^2 = \sum_{k=0}^{\infty} p_k^2$
Rank probability score	$rps(P, x) = \sum_{k=0}^{\infty} \{P_k - 1(x \geq k)\}^2$
Squared error score	$ses(P, x) = (x - \mu_p)^2,$ where μ_p is the mean of P

Note: This table summarizes scoring rules that we use to assess forecasting performance of the models under consideration. These are the rules that were discussed in the literature before, see Czado et al. (2009), and in particular, were proposed for comparison of models for count data.

To compare our results with the conclusions in Zhang et al. (2017) we also report Dawid-Sebastiani (DS) score which is defined in Equation (4.41)

$$DSS_{t,i}(\mathbf{X}_{t,i}) = \frac{X_{t,i} - \mu_{t,i}}{\sigma_{t,i}} + 2\log(\sigma_{t,i}). \quad (4.41)$$

4.6 Simulation Examples

In this section, we demonstrate the performance of the model with some simulated examples. We simulate data from various specifications of SSM and log-linear MACI models. We compare the models in terms of forecasting performance using the scoring rules discussed in the previous section and report estimation results. This simulation exercise is designed to give insights about the performance of the models incorrectly specified and misspecified and correctly specified frameworks.

Tables 4.4 and 4.5 summarize the parameters we have chosen for the data generating processes in our simulation study.

4.6.1 Prediction assessment of the simulation study

In this section, we assess the forecasting performance of two models for multivariate count data: state-space model and log-linear model. The data generating processes are presented in the previous section. Tables 4.6

Table 4.4: True parameters for the data sets generated from the state-space model in the examples of this section.

Data set	β_1	β_2	ϕ_{11}	ϕ_{21}	ϕ_{12}	ϕ_{22}	σ_{η_1}	σ_{η_2}	ρ
SSM_1	1.0	2.0	0.5	0.3	0.0	0.5	0.5	0.5	0.3
SSM_2	1.0	2.0	0.5	0.3	0.0	0.5	0.5	0.5	-0.3
SSM_3	1.0	2.0	0.5	0.3	0.0	0.5	0.5	0.5	0.0

Note: These are the data generating processes we choose for the data sets generated from the state-space framework. The underlying dependence structure includes Granger-causal feedback and positive/negative contemporaneous correlations in two out of three data generating processes: SSM_1 and SSM_2 . The data sets with the third data generating process only include Granger-causal feedback and no contemporaneous correlation.

and 4.7 summarize our findings regarding forecasting performance with scoring rules. In particular, Table 4.6 shows performance of the state-space model and Table 4.7 the performance of the log-linear model. The state-space model often outperforms the log-linear MACI model in most of the data generating processes, including one in the misspecified framework. In the cases when log-linear model outperforms the state-space model it is usually by a small margin. Moreover, most of the time the resulting scores are quite close in that case.

Tables 4.6 and 4.7 summarize forecasting assessment of the study for the simulated data. The rows of the tables correspond to a particular data generating process (see for details of specification Tables 4.4 and 4.5) and columns for a particular scoring rule (see scoring rules specification in Table 4.3). First, let us discuss the performance of the models when the data is generated from the state-space model. In two out of three data generating processes, labeled SSM_1 and SSM_3 , state-space framework

Table 4.5: **True parameters for the data sets generated from the log-linear model in the examples of this section.**

Data set	ω_1	ω_2	a_{11}	a_{22}	b_{11}	b_{12}	b_{21}	b_{22}
LL_1	0.9	0.4	-0.5	0.2	0.5	0.2	0.0	0.4
LL_2	0.2	0.3	0.2	0.4	0.5	0.2	0.0	0.4

Note: This figure shows the parameters of the data generating processes for the data sets generated from the log-linear MACI model.

Table 4.6: **Scores for the forecasting exercise with State-Space Model.**

DGP	log	qs	sph	rps	ds	se
SSM_1	1.484	-0.229	-0.440	0.770	2.352	2.634
SSM_2	1.861	-0.235	-0.487	1.000	3.136	3.551
SSM_3	1.967	-0.224	-0.475	0.948	3.599	4.075
LL_1	1.959	-0.164	-0.405	0.974	2.176	3.214
LL_2	1.351	-0.293	-0.543	0.545	1.103	1.087

Note: This table shows the scores obtained through forecasting exercise with state-space model for various data sets.

Table 4.7: Scores for the forecasting exercise with Log-Linear Model.

DGP	log	qs	sph	rps	ds	se
SSM_1	1.636	-0.321	-0.553	0.999	2.612	3.088
SSM_2	2.089	-0.164	-0.391	1.333	2.614	5.180
SSM_3	1.929	-0.220	-0.469	0.948	3.464	4.187
LL_1	1.985	-0.159	-0.400	0.996	2.238	3.357
LL_2	1.320	-0.309	-0.555	0.555	1.036	1.023

Note: This table shows the scores obtained through forecasting exercise with log-linear model for various data sets.

performs better according to most of the scoring rules and in one, SSM_2 , quadratic score, rank probability score and squared error score indicate better performance of state-space model, while logarithmic score, spherical score, and Dawid-Sebastian score indicate better performance of the log-linear model. What concerns the data generated from the log-linear model, in one case, data generated process labeled LL_1 , state-space model performs better according to all scoring rules, and in the other case, data generated process labeled LL_2 , the log-linear model performs better according to all scoring rules. In both cases for the log-linear data generating processes. However, the scores for forecasts with the state-space model and log-linear model are very close to each other.

4.7 Empirical Applications

4.7.1 Road accidents: Utrecht, Dordrecht and Haarlemmermeer

We analyze car crash data from three road segments in the Netherlands – Utrecht, Dordrecht, and Haarlemmermeer. We estimate three bivariate models and one trivariate model for daily crash counts in 2014. One could expect some correlation in such data due to, for example, similar weather conditions. In particular Brijs et al. (2008) study the effects of weather on car crash counts using INAR model. This study analyses roads

in Utrecht, Dordrecht, and Haarlemmermeer for the year 2001, the authors find somewhat relationship between the weather and the number of car crashes. Other literature makes inconsistent conclusions, and hence we do not expect to find a relationship of a particular sign (negative/positive correlation) if any at all. Here we consider the trivariate model for the car

Table 4.8: **Descriptives for the car crash data set.**

	Utrecht	Dordrecht	Haarlemmermeer
mean	4.7507	1.6137	3.0329
median	4	1	3
st.d.	2.7166	1.3733	2.2358
dispersion	0.327	0.7243	0.7243
minimum	0	0	0
maximum	15	7	7

Note: This table shows descriptive statistics for the data set of car crash counts.

crash count data. We are considering car crash counts on the directions to Utrecht, Dordrecht, and Haarlemmermeer. We estimate state-space framework and log-linear model for the data of 2014 and perform a forecasting exercise for the year 2015. Table 4.9 summarizes the results. In this application the log-linear model outperforms the state-space framework. It can be due to two reasons. First, it might be that the log-linear model is closer to the data generating process. Second, it might be due to the variance of the estimated likelihood. Note, that in this application we estimate the likelihood using the standard bootstrap particle filter. The variance of the likelihood grows with the dimensionality of the model. Thus, the estimation results and hence forecasting performance can be affected by this issue.

4.7.2 Bank Failures

Bank failures have been modeled univariately using a Poisson process (Davutyan, 1989). A number of researches have investigated bank failure data of different time periods and with different applications in mind. For instance, Schoenmaker (1996) discusses contagion risk in banking. Overall, it is rea-

Table 4.9: **Scores for the forecasting exercise with car crash data.**

Model	log	qs	sph	rps	ds	se
Log-Linear	2.3247	-0.1284	-0.3650	1.5450	3.3586	6.0743
SSM	2.6797	-0.0737	-0.2817	1.6784	3.4998	9.5627

Note: This table shows scoring rules for the forecasting exercise in the trivariate model for car crash data.

sonable to assume that sometimes bank failures in different countries are driven by similar economic phenomena, and possibly contagion/spillover effects exist between economies of different countries. In this section, we compare the models for count data using a bivariate data set of bank failures in the U.S. and Russia. We use monthly data from the period January 2006 until December 2011. The sample size is taken as $T = 55$, and we make five steps ahead predictions using the log-linear model and the state-space approach.

Table 4.10: **Descriptives for the bank failures data for the period January 2006 until December 2011.**

	Russia	U.S.
mean	4.0727	7.2182
median	3	7
st.d.	2.7275	6.7706
dispersion	0.4485	0.8798
minimum	0	0
maximum	10	24

Note: This table shows descriptive statistics of the data set for bank failures.

We further investigate how models perform in the forecasting exercise, Table 4.11 summarizes the results. We conclude that both models perform quite well. The state-space approach, however, performs better according to most of the scoring rules.

Table 4.11: **Scores for the forecasting exercise with bank failure data.**

Model	log	qs	sph	rps	ds	se
SSM	1.9026	-0.1738	-0.4189	0.9755	2.1619	3.0841
Log-Linear	2.0244	-0.1623	-0.3996	0.8862	2.2934	4.2031

Note: This table shows scoring rules for the forecasting exercise in the bivariate model for bank failure data.

4.7.3 Transactions

In this empirical application we analyze the number of transactions on 30 seconds intervals for Deutsche Bank AG and Bayer AG⁴. The summary statistics for the data are provided in Table 4.12 and the data are illustrated in Figure 4.12.

Table 4.12: **Descriptives for the transactions data.**

	Deutsche Bank AG	Bayer AG
mean	6.95	7.716
median	5	5
st.d.	8.2462	8.227
dispersion	1.4078	1.1368
minimum	0	0
maximum	67	60

Note: This table shows descriptive statistics of the data set for transactions.

⁴The data was obtained from FactSet

Figure 4.12: **Transaction data for Bayer AG and Deutsche Bank AG.**

Note: This figure illustrate transaction counts for Bayer A and Deutsche Bank AG. In comparison to other empirical applications this data set exhibits rather large counts which is typical for financial markets.

Table 4.13 shows the scoring rules for the forecasting exercise with transactions data. While the state-space model performs better according to most of the scoring rules, in general the scores are very close to each other.

Table 4.13: Scores for the forecasting exercise with transaction data.

Model	log	qs	sph	rps	ds	se
Log-Linear	3.8079	-0.0358	-0.2404	3.0531	7.6096	23.7932
SSM	3.7226	-0.0361	-0.2302	2.8957	6.8750	23.9339

Note: This table shows scoring rules for the forecasting exercise in the transaction applications. SSM model outperforms log-linear model according to four out of six scoring rules.

4.8 A note on copula approach

Copulas provide a framework to modeling dependence dynamics in the data. They have found applications in many fields of studies, for the general overview see Nelsen (2007) and Joe (2014). Copulas appear to be a useful tool when one wants to model dependence, and from the mathematical point of view, the following theorem expresses the general idea behind this concept (for more details see Embrechts, 2009).

Theorem 4.1. (Sklar, 1959) Suppose $X_1, X_2, \dots, X_n$ are random variables with continuous density functions $F_1, F_2, \dots, F_n$ and joint density F , then there exist a unique copula C (a density function on $[0, 1]^n$ with uniform marginals) such that for all $x = (x_1, x_2, \dots, x_n)^T \in \mathbb{R}^n$:

$$F(x_1, x_2, \dots, x_n) = C(F_1(x_1), F_2(x_2), \dots, F_n(x_n)). \quad (4.42)$$

Conversely, given any density functions $F_1, F_2, \dots, F_n$ and copula C , F defined through Eq. 4.42 is an n -variate density function with marginal density functions $F_1, F_2, \dots, F_n$.

Indeed, copulas appear to be a powerful tool for modeling the dependence, and many choices have been proposed in the literature for defining a copula. A precise specification always depends on a particular problem one is considering. In terms of count time-series, a few approaches have been considered. For example, Andreassen (2013) considers the following

way of modeling dependence in the count time series with the application of the Franc copula and Clayton copula. First, note

$$\mathbb{P}_\theta(Y_{t,1} \leq m, Y_{t,2} \leq n \mid \mathcal{I}_{t-1}) = C_\varphi(F_{\lambda_{t,1}}(m), F_{\lambda_{t,2}}(n)), \quad (4.43)$$

where C_φ is a copula and F_λ is the cdf of a Poisson distribution which has the mean λ . The joint pmf of $Y_{t,1} \mid \mathcal{I}_{t-1} \sim \text{Poisson}(\lambda_{t,1})$ and $Y_{t,2} \mid \mathcal{I}_{t-1} \sim \text{Poisson}(\lambda_{t,2})$ reads

$$\begin{aligned} \mathbb{P}_\theta(Y_{t,1} = m, Y_{t,2} = n \mid \mathcal{I}_{t-1}) = \\ C_\varphi(F_{\lambda_{t,1}}(m)F_{\lambda_{t,2}}(n)) - C_\varphi(F_{\lambda_{t,1}}(m-1), F_{\lambda_{t,2}}(n)) \quad (4.44) \\ - C_\varphi(F_{\lambda_{t,1}}(m), F_{\lambda_{t,2}}(n-1)) + C_\varphi(F_{\lambda_{t,1}}(m-1), F_{\lambda_{t,2}}(n-1)). \end{aligned}$$

The the cdf of the Franc copula is defined in the following way

$$C_\varphi(u, v) = \begin{cases} -\frac{1}{\varphi} \log \left(1 - \frac{(1-e^{-\varphi u})(1-e^{-\varphi v})}{1-e^{-\varphi}} \right), & \text{if } \varphi \neq 0 \\ uv, & \text{if } \varphi = 0, \end{cases} \quad (4.45)$$

and the cdf of the Clayton copula reads

$$C_\varphi(u, v) = \begin{cases} (u^{-\varphi} + v^{-\varphi} - 1)^{-1/\varphi}, & \text{if } u, v, \varphi > 0 \\ \text{or both } \varphi < 0 \text{ and } u^{-\varphi} + v^{-\varphi} > 1, \\ uv, & \text{if } \varphi = 0 \\ 0, & \text{if otherwise,} \end{cases} \quad (4.46)$$

where $(u, v) \in [0, 1]^2$ and $\varphi \in [-1, \infty) \setminus \{0\}$. Andreassen (2013) presents preliminary results for modeling bivariate count data with copulas and make the comparison to INGARCH (MACI) model. Preliminary results suggest that copulas can provide a flexible framework and incorporate negative dependency in the model. However, the authors also note that specification of the joint pdf of the counts is challenging, and multivariate extensions (the authors are considering bivariate scenario) is troublesome. Asymp-

otic results for this approach have not yet been established. Another way to specify copula for modeling of the time series of counts is proposed in Doukhan et al. (2017). Let us assume that we are in the framework of multivariate autoregressive conditional intensity

$$X_{i,t} \mid \mathcal{F}_-^{\mathcal{X}, \boldsymbol{\lambda}} \sim \text{Poisson}(\lambda_{i,t}), \quad (4.47)$$

$$\boldsymbol{\lambda}_t = \boldsymbol{\omega} + \mathbf{A}\boldsymbol{\lambda}_{t-1} + \mathbf{B}\mathbf{X}_{t-1}, \quad (4.48)$$

In Doukhan et al. (2017) the authors note the fact that every random variable $X_{i,t}$ is distributed according to Poisson distribution marginally, but the random vector $\mathbf{X}$ does not have to be distributed jointly as multivariate Poisson distribution. Hence the authors construct the model in such a way that arbitrary dependencies are allowed between the individual Poisson components. In more detail, assume that $\boldsymbol{\lambda}_0 = (\lambda_{1,0}, \dots, \lambda_{n,0})$ is a starting value. Then the data can be generated in the following way (Doukhan et al., 2017):

- Let $\mathbf{U}_l = (U_{1,l}, \dots, U_{n,l})$ for $l = 1, 2, \dots, K$, be a sample from n -dimensional copula $C(u_1, \dots, u_n)$. Then $U_{i,l}$, $l = 1, 2, \dots, K$, follow marginally the uniform distribution on $(0,1)$, for $i = 1, 2, \dots, n$.
- Consider the transformation

$$Y_{i,l} = -\frac{\log U_{i,l}}{\lambda_{i,t}}, \quad i = 1, 2, \dots, n.$$

Then the marginal distribution of $X_{i,l}$, $l = 1, 2, \dots, K$ is exponential with parameter $\lambda_{i,0}$, $i = 1, 2, \dots, n$.

- Take a large enough K , and define

$$X_{i,0} = \max_{1 \leq l \leq K} \left\{ \sum_{l=1}^K X_{i,l} \leq 1 \right\}, \quad i = 1, 2, \dots, n.$$

Then $\mathbf{X}_0 = (X_{1,0}, \dots, X_{n,0})$ is marginally a set of first values of a Poisson process with parameter λ_0 .

- Use the appropriate model specification (say, Equations (4.47) and (4.48)) to get λ_1 .
- Repeat the previous steps from the start.

The idea of this approach is to not use the copula for modeling the time series of counts directly, but instead, use it for a vector of associated continuous random variables (Doukhan et al., 2017). The authors show some asymptotic results, derive conditions for ergodicity and stationarity, and estimate the models with quasi-maximum likelihood, and provide preliminary simulation results for bivariate frameworks. However, the authors mention that further work is required to establish properties of multivariate count processes specified in this way and suggest some possible generalizations. Simulation results suggest that when the data is generated from the correct model, the performance is quite good and improves with increasing sample size. The results seem to be reasonably robust to the choice of copulas: the authors use Gaussian copula and compare their results to Clayton copula. To summarize, copulas seem to gain popularity in the literature that develops models and methods for the time series of counts. The approach provides flexibility and allows for modeling of negative dependencies in the data. However, it does have various drawbacks. Formulation of the joint pmf of counts is one challenge, as well as the particular choice of copula when it comes to higher dimensions. Moreover, in the literature, there have appeared concerns about identifiability issues in the copula approach for modeling the time series of counts. In particular, Trivedi and Zimmer (2017) confirm results of Genest and Nešlehová (2007) with the series of simulation studies on identifiability of copulas for discrete outcomes and particularly for count data.

4.9 Discussion

In this paper, we have reviewed multiple approaches for modeling multivariate count time series data. One of the challenges that appear in the literature and have not been resolved is modeling the dependency between counts in a flexible way that would also allow for feasible estimation. We have discussed multivariate autoregressive conditional intensity models (MACI), their log-linear alternative to which we refer as the multivariate log-linear model and nonlinear state-space model. All of the models have advantages and disadvantages. In particular, the nonlinear state-space framework allows for various dependencies that one cannot easily incorporate into MACI or log-linear approach. However, these models can be hard and computationally expensive to estimate, in particular in higher-dimensions. Challenges in estimation rise from different sources. One is intractable likelihood and thus one must either use approximate methods or sampling techniques, i.e., sequential Monte Carlo, for estimation of the likelihood. Further, for estimation of the posterior distribution one has to use Markov Chain Monte Carlo methods. These methods can be hard to scale to larger dimensions. MACI models, on the other hand, are quite restrictive: they restrict coefficients in the model to be positive as well as the correlation between time series. Both assumptions can be unrealistic in many real-world applications. Log-linear model avoids the problem of only positive coefficients in the model by logarithmic transformation of the data. However, estimation can be unstable, and good starting points need to be chosen for the estimation. When the dimension of the model grows, it becomes harder to choose good starting points for the optimization problem. Nevertheless, in terms of flexibility, this model is the best competitor for the state-space approach.

In this chapter, we have also compared log-linear models and state-space models for count data in terms of forecasting performance on multiple simulated data sets and real data applications. We found that on the simulated data state-space framework generally outperforms log-linear model. In the cases when the state-space model performs worse the difference is usually

marginal. Moreover, the computational advantage of log-linear and MACI models decreases with the increase in either dimensionality of the model or the number of counts. It is due to the usage of the multivariate Poisson distribution. As every pairwise correlation has to be modeled as a separate Poisson random variable. Moreover, the sum in the specification of the joint distribution runs through the number of counts. Generally, one could say that estimation of log-linear models much faster than of the state-space models. It is true that in low dimensions and with the small number of counts these models do not require much of computational power, however, once the number of counts increases and once we deal with higher dimensions, the computations become much more extensive due to large sums in the multivariate Poisson distribution. Moreover, while running the model on simulated and empirical data, we found that the estimation can be numerically unstable and can highly depend on the starting values in the estimation procedure. We follow the suggestion of Doukhan et al. (2017), and the first estimate the model for univariate time series. These estimates we further use in multivariate estimation. However, the problem of numerical instability remains especially in small samples according to our experience.

Finally, both approaches have their drawbacks. In particular, the log-linear model seems to have numerical stability issues. Finding optimal starting values for optimization can be a challenge. In the state-space approach, the challenging part is the estimation of the likelihood which is intractable and sample from the posterior distribution. Existing state of the art solutions are computationally very expensive. Thus, exploring approximate methods might be beneficial for application of these models in high dimensional case.

Chapter 5

Conclusion

*“I would rather have questions that can’t be answered than answers that
can’t be questioned.”*

-Richard Feynman

In this chapter, we provide an overall conclusion to the thesis and discuss directions for future research. In the chapters that constitute this work, we have considered Bayesian estimation of nonlinear state-space models. These models allow modeling complex dependence structure in multivariate models with latent variables. However, standard estimation procedures cannot be used due to the intractability of the likelihood.

In Chapter 2 we address the problem of estimation of the stochastic volatility models that belong to the class of non-linear state-space models. We consider various methods of estimation which are originating in Bayesian statistics and machine learning literature. We make an extensive review of possible solutions to this problem and compare some of the promising approaches. By promising we mean approaches that might be successfully extended to the multivariate case and ideally to the high-dimensional case, although the latter seems to be infeasible at the moment. Different methods provide a different degree of accuracy in the estimation and require different computing time for carrying out the inference. Moreover, they are also distinguishable by the degree of manual work required

to calibrate or adjust the method when some assumptions (for instance, distributional assumptions about the noise term) about the model change. It is particularly relevant for practitioners who would not want to do a lot of manual calculations to change and test simple assumptions about the model. Therefore, straightforward adaptation of the code to changes of distribution assumptions (for example, t-student distribution instead of normal distribution for error terms) or regarding parametrization of the model (say, adding leverage effect to stochastic volatility model) can be of an asset. We compare methods such as particle Markov Chain Monte Carlo methods, Hamiltonian Monte Carlo methods, Variational Bayes and Integrated Nested Laplace Approximation. Our goal was to estimate the multivariate non-linear state-space model and in particular multivariate stochastic volatility models. With this goal in mind, we discovered that approximate methods do not provide satisfactory results in the estimation. Thereby, we decide to proceed with particle Markov Chain Monte Carlo methods as the primary framework for multivariate estimation of stochastic volatility models in Chapter 3. Certain schemes in implementation – such as using mixture proposal distributions and block updating – allow the algorithm to perform well while still being feasible in terms of computational time in multivariate models.

In Chapter 3 we build a framework for detecting and measuring volatility spillover effects with multivariate stochastic volatility model. Financial markets are known to exhibit co-movements, and stochastic volatility models provide a very intuitive approach for modeling dependencies between financial markets. When the latent process is modeled through a Vector Autoregressive model ($\text{VAR}(p)$) the autoregressive matrices correspond to Granger-causal feedback and contemporaneous correlation is taken into account in the variance-covariance matrix of the log-returns. Thereby, we can incorporate volatility spillover measures developed for VAR framework and estimate their posterior distributions in a fully Bayesian framework using multivariate stochastic volatility models.

In Chapter 4 we investigate whether the flexibility of a non-linear state-space model has an advantage above simpler, but more restrictive models

for count time series data. First, we consider two observation-driven models in multivariate frameworks, in particular, we look into multivariate Conditional Intensity Model (MACI) and its log-linear model. In order for MACI to be stationary, one must assume that the parameters of the model are positive. It leads to problems when parameters are close or on the boundary of the parameter space since standard errors and confidence intervals cannot be trivially obtained. The log-linear model uses a logarithmic transformation of the data in the conditional intensity equation of the model and by that avoids the problem. Another challenge for both of these models is the specification of the joint distribution of the counts. Log-linear models appear to be less restrictive, and thus we use it as the main competitor of the parameter-driven model based on the state-space framework. In our simulation studies and real data applications, we find that in most of the cases the state-space framework either outperforms log-linear model or at least almost as good. The models were compared based on predictive performance using multiple assessment measures.

The approach that we have reviewed, but have not considered in the simulation studies, is modeling the dependency through copulas. Even though they do provide a flexible framework for modeling dependency and allow for negative dependency too (which is not, for example, possible with multivariate Poisson distribution), copulas have documented problems with identifiability. We find that in terms of forecasting performance non-linear state-space models often outperform log-linear version of MACI models even in the model misspecified scenario. It provides a good motivation for further development of scalable estimation of methods for this class of models.

To sum up all the above, multivariate non-linear state-space models provide a powerful tool for modeling dependence in the data. There are however some issues that can be improved in the future line of research. Below we discuss suggestions for the future research based on the outcomes of this thesis.

Estimation

In Chapter 3 and 4 we used particle Markov Chain Monte Carlo methods for the estimation of the nonlinear state-space model. In Chapter 3 we considered multivariate stochastic volatility model and in Chapter 4 state-space approach for count time series data. Both models can benefit from developing and applying more efficient Sequential Monte Carlo algorithms. In particular, for the stochastic volatility model, we have already used Auxiliary Particle Filter (APF) developed by Pitt and Shephard (1999). However, some schemes could provide a better solution in the multivariate, and potentially high-dimensional, framework. For example, the iterated Auxiliary Particle Filter Guarniero et al. (2017) can provide a solution to the problem if modified appropriately. The idea of the APF is based on the fact that instead of making a proposal for the next generation of particles which is based only on the model dynamics $p(h_t | h_{t-1})$, we can include also the current observation y_t into account. Then the particles are proposed according to $p(h_t | h_{t-1}, y_t)$. The iterated APF and similar methods are based on the idea that we can include all the data we have observed in our proposal. Unfortunately, the current implementation ignores cross-dependence in the particles (we discuss this in Chapter 2) and hence is not suitable for the problems we have considered. A straightforward extension of the same method that would account for cross-dependence in the particles is too computationally expensive to apply. Thereby, one of the possible solutions would be to apply either the Laplace approximation or Expectation Propagation when proposing the next generation of particles. They are not straightforward approaches, but we believe they are worth exploring.

Another improvement in estimation can be achieved through extending methodology of Hamiltonian Monte Carlo (as reviewed in Chapter 2) to the multivariate case in both stochastic volatility model, and state-space model for count time series. We have expected particle Markov Chain Monte Carlo to perform well in the dimensionalities we were considering. Indeed, in 2-dimensional and 3-dimensional cases it provides satisfactory re-

sults. The largest dimensionality we considered was with 5 time series (and correspondingly 5 latent processes). In this case we experienced difficulties with calibration of the algorithm and slower convergence. Hamiltonian Monte Carlo algorithms might be able to handle larger parameter spaces and ensure better mixing in the multivariate framework.

The choice of the prior is another thing one should consider. In Chapter 3 we considered spike and slab priors for model choice. However, the computational cost for this kind of inference is too high since we move not only in parameter space, but also in model space. Ideally, when designing the algorithm for higher dimensions, one also should consider using shrinking priors. An appealing choice for the shrinking priors seems to be either the standard Minnesota prior or more recently proposed Horseshoe prior. We leave this for the future research.

Time varying volatility spillovers

Chapter 3 develops a methodology for measuring volatility spillovers within the framework of multivariate stochastic volatility model. An interesting extension from the point of view of practitioners, especially from policy-makers, would be considered time-varying spillover effects. Adding time variation into the current methodology would be too expensive computationally. However, a promising way to do it through the development of approximate methods in Bayesian framework.

Chapter 6

Valorisation

“It is our responsibility as scientists, knowing the great progress and great value of a satisfactory philosophy of ignorance, the great progress that is the fruit of freedom of thought, to proclaim the value of this freedom, to teach how doubt is not to be feared but welcomed and discussed, and to demand this freedom as our duty to all coming generations.”

-Richard Feynman

In the following addendum, I will discuss the value of this dissertation for society through two prisms. First, through the research questions that I have been investigating. Second, through a personal reflection on the training, I have received from the academic community.

This thesis contributes to the utilization of a specific class of nonlinear state-space models. First, I would like to remind the reader why these models can be useful. In the real world, it often happens that there exist latent or unobservable variable that affects the system we are interested in. This latent variable can be either some aspect of the real world that potentially could be measured but was not for some practical reasons, or an abstract concept, like volatility on the financial markets in our case, that cannot be measured directly at all. State-space models often are chosen to deal with the problem of modeling latent variables. Moreover, the state-space approach can model various types of linear and non-linear relationships in

observations through latent variables. These models can have a wide range of applications in different scientific fields. It is still commonly accepted that non-linear state-space models are hard to estimate. It already happens in the univariate case when a single observation process depends on a single latent process. The challenges of estimation become more severe in multivariate models. However, if one manages to build an efficient estimation procedure, these models can serve well in various scientific fields, including economics, finance, biology, physics, engineering.

As non-linear state-space models are hard to estimate, Chapter 2 of this thesis is particularly concerned with bringing together scientific fields (statistics, machine learning, and econometrics) and comparing methods that can be used for the estimation of models of our interest, namely stochastic volatility models. Having an overview of existing methods from these fields can be helpful in efficiently solving the problem. Non-linear state-space models are quite flexible and the estimation method has to be chosen based on the particular specification of the model. Thus, this chapter can be used as a guideline for choosing a strategy to estimate the model of a particular specification. In our case we use it as a guideline for estimation of the multivariate stochastic volatility models where the latent variables are modeled with Vector Autoregressive process and the dependence between latent variables and observations is non-linear.

Chapter 3 develops a new method for measuring spillover effects on financial markets. This method can be used by portfolio managers and policymakers for taking more effective decisions. Knowing the dependencies between financial markets, portfolio managers can make a better decision regarding portfolio diversification and policymakers can be more effective in making a decision regarding bailout funds. The volatility spillover effects in this framework are defined in a very transparent and intuitive way. Bayesian inference allows us to report the uncertainty in the parameters and spillover measures themselves. Overall, this approach to measuring volatility spillovers can make the decision process more informed. Imagine, that financial markets of country A and country B are interconnected. Moreover, it is known that there is Granger-causal relationship from country A

to country B, i.e. there is a spillover effect from country A to country B. If policymakers of country B know how strong this relationship is, it would help them to create sufficient fund to support their economy in case crisis happens in country A and in case it transmits to their own economy. Thus, the whole society would benefit from it since the consequences of a crisis would be reduced. Finally, in this chapter we also discuss current limitations in estimation of non-linear state-space models and propose directions for future research.

Chapter 4 is concerned with the application of non-linear state-space models in a different scenario: in modeling multivariate count data. These data appear in many scientific fields and has its specific problems. We compare the state-space approach with its most relevant competitor in terms of forecasting performance and conclude that the state-space approach is usually favorable. This conclusion and method can be useful for practitioners who work with count data. The applications we have considered are forecasting of a number of transactions on financial markets, forecasting the number of car crashes and forecasting number of bank failures. However, there are much more different problems that could benefit from modeling with the state-space approach.

Academic research in its many different variations has different societal impacts. Likewise, Ph.D. training has its own specific value to society. Young scientists form their scientific values during this time, which in turn shapes the science of the future. We learn how to conduct scientific investigation, we learn how to teach, how to assimilate and expand knowledge. We learn to pose interesting questions and try to answer them. Moreover, we learn how to deal with the uncertainty, how to live with the fact that many questions have not yet been answered. These are unique skills that take time to be developed and can be important in universities, research institutes, public organizations, and companies. One of the responsibilities I have had was teaching on undergraduate level at the university. Lecturers and tutors and the universities can have profound impact on the students who will, later on, find their careers in academia, public organizations and companies. I believe that we are capable of not only teaching students the

material but also change their attitude towards knowledge. There is room for the development of critical thinking, creativity and accepting that perhaps not all questions can be answered straight away, which does not mean we should not try. I think that I, like many academics before and after me, have the power to show students the importance and impact of science.

Bibliography

- Amari, S.-I. and H. Nagaoka (2007). *Methods of information geometry*, Volume 191. American Mathematical Soc.
- Andersen, T. G. and L. Benzoni (2014). *Stochastic volatility*. Springer.
- Andreassen, C. M. (2013). *Models and inference for correlated count data*. Ph. D. thesis, Department of Mathematics, Aarhus University.
- Andrieu, C., A. Doucet, and R. Holenstein (2010). Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 72(3), 269–342.
- Asai, M., M. McAleer, and J. Yu (2006). Multivariate stochastic volatility: a review. *Econometric Reviews* 25(2-3), 145–175.
- Betancourt, M. (2017). A conceptual introduction to Hamiltonian Monte Carlo. *arXiv preprint arXiv:1701.02434*.
- Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.
- Black, F. (1976). Studies of stock price volatility changes. *Proceedings of the 1976 Meeting of the Business and Economic Statistics Section*, American Statistical Association, Washington DC, 177–181.
- Black, F. and M. Scholes (1973). The pricing of options and corporate liabilities. *Journal of political economy* 81(3), 637–654.

- Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics* 31(3), 307–327.
- Brailsford, T. J. (1996). Volatility spillovers across the Tasman. *Australian Journal of Management* 21(1), 13–27.
- Brijs, T., D. Karlis, and G. Wets (2008). Studying the effect of weather conditions on daily crash counts using a discrete time-series model. *Accident Analysis & Prevention* 40(3), 1180–1190.
- Carvalho, C. M., N. G. Polson, and J. G. Scott (2009). Handling sparsity via the horseshoe. In *International Conference on Artificial Intelligence and Statistics*, pp. 73–80.
- Chiang, T. C., B. N. Jeon, and H. Li (2007). Dynamic correlation analysis of financial contagion: Evidence from Asian markets. *Journal of International Money and finance* 26(7), 1206–1228.
- Chib, S. and E. Greenberg (1995). Understanding the Metropolis-Hastings algorithm. *The American Statistician* 49(4), 327–335.
- Chib, S., Y. Omori, and M. Asai (2009). Multivariate stochastic volatility. In *Handbook of Financial Time Series*, pp. 365–400. Springer.
- Chiriac, R. and V. Voev (2011). Modelling and forecasting multivariate realized volatility. *Journal of Applied Econometrics* 26(6), 922–947.
- Choi, D., V. Fang, and T. Fu (2007). Volatility spillovers between stock market returns and exchange rate changes: the New Zealand case. *Countries, Review of Economics and Statistics* 85, 212–217.
- Chopin, N. et al. (2004). Central limit theorem for sequential Monte Carlo methods and its application to Bayesian inference. *The Annals of Statistics* 32(6), 2385–2411.
- Conrad, C. and M. Karanasos (2010). Negative volatility spillovers in the unrestricted ECCC-GARCH model. *Econometric Theory* 26(3), 838–862.

- Corsetti, G., M. Pericoli, and M. Sbracia (2005). Some contagion, some interdependence: More pitfalls in tests of financial contagion. *Journal of International Money and Finance* 24(8), 1177–1199.
- Cox, D. R., G. Gudmundsson, G. Lindgren, L. Bondesson, E. Harsaae, P. Laake, K. Juselius, and S. L. Lauritzen (1981). Statistical analysis of time series: Some recent developments [with discussion and reply]. *Scandinavian Journal of Statistics*, 93–115.
- Crowell, B. (2003). *Newtonian physics*, Volume 1. Light and Matter.
- Cseke, B. and T. Heskes (2011). Approximate marginals in latent Gaussian models. *Journal of Machine Learning Research* 12(Feb), 417–454.
- Czado, C., T. Gneiting, and L. Held (2009). Predictive model assessment for count data. *Biometrics* 65(4), 1254–1261.
- Dahlhaus, R. and M. Eichler (2003). Causality and graphical models in time series analysis. *Oxford Statistical Science Series*, 115–137.
- Davis, R. A., W. T. Dunsmuir, and Y. Wang (1999). Modeling time series of count data. *Statistics Textbooks and Monographs* 158, 63–114.
- Davutyan, N. (1989). Bank failures as Poisson variates. *Economics Letters* 29(4), 333–338.
- Diebold, F. X. and K. Yilmaz (2012). Better to give than to receive: Predictive directional measurement of volatility spillovers. *International Journal of Forecasting* 28(1), 57–66.
- Douc, R. and O. Cappé (2005). Comparison of resampling schemes for particle filtering. In *ISPA 2005. Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, 2005.*, pp. 64–69. IEEE.
- Doucet, A., N. de Freitas, and N. Gordon (2001). *Sequential Monte Carlo Methods in Practice*. Springer.

- Doucet, A. and A. M. Johansen (2009). A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of nonlinear filtering* 12(656–704), 3.
- Doukhan, P., K. Fokianos, B. Støve, and D. Tjøstheim (2017). Multivariate count autoregression. *arXiv preprint arXiv:1704.02097*.
- Duane, S., A. D. Kennedy, B. J. Pendleton, and D. Roweth (1987). Hybrid Monte Carlo. *Physics letters B* 195(2), 216–222.
- Durbin, J. and S. J. Koopman (2012). *Time series analysis by state space methods*, Volume 38. OUP Oxford.
- Ehlers, R. and M. Zevallos (2015). Bayesian estimation and prediction of stochastic volatility models via INLA. *Communications in Statistics-Simulation and Computation* 44(3), 683–693.
- Eichler, M. (2013). Causal inference with multiple time series: principles and problems. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* 371(1997).
- Embrechts, P. (2009). Copulas: A personal view. *Journal of Risk and Insurance* 76(3), 639–650.
- Ferland, R., A. Latour, and D. Oraichi (2006). Integer-valued GARCH process. *Journal of Time Series Analysis* 27(6), 923–942.
- Fokianos, K. (2012). Count time series models. *Time Series-Methods and Applications, Handbook of Statistics* 30, 315–347.
- Fokianos, K. and D. Tjøstheim (2011). Log-linear Poisson autoregression. *Journal of Multivariate Analysis* 102(3), 563–578.
- Forbes, K. J. and R. Rigobon (2002). No contagion, only interdependence: measuring stock market comovements. *The journal of Finance* 57(5), 2223–2261.

- Gall, J., J. Potthoff, C. Schnörr, B. Rosenhahn, and H.-P. Seidel (2007). Interacting and Annealing Particle Filters: Mathematics and a Recipe for Applications. *Journal of Mathematical Imaging and Vision* 28(1), 1–18.
- Gamba-Santamaria, S., J. E. Gomez-Gonzalez, J. L. Hurtado-Guarin, and L. F. Melo-Velandia (2017). Volatility spillovers among global stock markets: measuring total and directional effects. *Empirical Economics*, 1–19.
- Geman, S. and D. Geman (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence* (6), 721–741.
- Genest, C. and J. Nešlehová (2007). A primer on copulas for count data. *ASTIN Bulletin: The Journal of the IAA* 37(2), 475–515.
- George, E. I. and R. E. McCulloch (1993). Variable selection via Gibbs sampling. *Journal of the American Statistical Association* 88(423), 881–889.
- Gerber, M., N. Chopin, and N. Whiteley (2017). Negative association, ordering and convergence of resampling methods. *arXiv preprint arXiv:1707.01845*.
- Geyer, C. (2011). Introduction to Markov Chain Monte Carlo. *Handbook of markov chain monte carlo* 20116022, 45.
- Girolami, M. and B. Calderhead (2011a). Riemann Manifold Langevin and Hamiltonian Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 73(2), 123–214.
- Girolami, M. and B. Calderhead (2011b). Riemann Manifold Langevin and Hamiltonian Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 73(2), 123–214.
- Gordon, N. J., D. J. Salmond, and A. F. Smith (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. In *IEE Proceedings F-Radar and Signal Processing*, Volume 140, pp. 107–113. IET.

- Guan, Y. and M. Stephens (2012). Small world MCMC with tempering: Ergodicity and spectral gap. *arXiv preprint arXiv:1211.4675*.
- Guarniero, P., A. M. Johansen, and A. Lee (2017). The iterated auxiliary particle filter. *Journal of the American Statistical Association* 112(520), 1636–1647.
- Hafner, C. M. and H. Herwartz (2008). Testing for causality in variance using multivariate GARCH models. *Annales d'Economie et de Statistique*, 215–241.
- Halton, J. H. (1962). Sequential Monte Carlo. In *Mathematical Proceedings of the Cambridge Philosophical Society*, Volume 58, pp. 57–78. Cambridge University Press.
- Harvey, A., E. Ruiz, and N. Shephard (1994). Multivariate stochastic variance models. *The Review of Economic Studies* 61(2), 247–264.
- Harvey, A. C. and N. Shephard (1996). Estimation of an asymmetric stochastic volatility model for asset returns. *Journal of Business & Economic Statistics* 14(4), 429–434.
- Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* 57(1), 97–109.
- Heinen, A. and E. Rengifo (2007). Multivariate autoregressive modeling of time series count data using copulas. *Journal of Empirical Finance* 14(4), 564–583.
- Heston, S. L. (1993). A closed-form solution for options with stochastic volatility with applications to bond and currency options. *The review of financial studies* 6(2), 327–343.
- Hilmer, C. E., M. T. Holt, et al. (2000). A comparison of resampling techniques when parameters are on a boundary: The bootstrap, subsample bootstrap, and subsample jackknife. In *Annual Meeting, July*.

- Hull, J. and A. White (1987). The pricing of options on assets with stochastic volatilities. *The journal of finance* 42(2), 281–300.
- Ishwaran, H. and J. S. Rao (2005). Spike and slab variable selection: frequentist and Bayesian strategies. *Annals of statistics*, 730–773.
- Jacquier, E., N. G. Polson, and P. E. Rossi (2004). Bayesian analysis of stochastic volatility models with fat-tails and correlated errors. *Journal of Econometrics* 122(1), 185–212.
- Joe, H. (2014). *Dependence modeling with copulas*. CRC Press.
- Johansen, A. M. and A. Doucet (2008). A note on auxiliary particle filters. *Statistics & Probability Letters* 78(12), 1498–1504.
- Johnson, H. and D. Shanno (1987). Option pricing when the variance is changing. *Journal of Financial and Quantitative Analysis* 22(2), 143–151.
- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME—Journal of Basic Engineering* 82(Series D), 35–45.
- Karlis, D. and L. Meligkotsidou (2005). Multivariate Poisson regression with covariance structure. *Statistics and Computing* 15(4), 255–265.
- Kim, S., N. Shephard, and S. Chib (1998). Stochastic volatility: likelihood inference and comparison with ARCH models. *The review of economic studies* 65(3), 361–393.
- Kitagawa, G. (1996). Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics* 5(1), 1–25.
- Lee, S. B. and K. J. Kim (1993). Does the October 1987 crash strengthen the co-movements among national stock markets? *Review of Financial Economics* 3(1), 89.

- Lennon, H. (2016). *Gaussian copula modelling for integer-valued time series*. Ph. D. thesis, University of Manchester.
- Lindsten, F., M. I. Jordan, and T. B. Schön (2014). Particle Gibbs with ancestor sampling. *Journal of Machine Learning Research* 15(1), 2145–2184.
- Litterman, R. B. et al. (1984). Specifying vector autoregressions for macroeconomic forecasting. Technical report, Federal Reserve Bank of Minneapolis Minneapolis.
- Liu, H. (2012). *Some models for time series of counts*. Columbia University.
- Mandelbrot, B. B. (1997). The variation of certain speculative prices. In *Fractals and scaling in finance*, pp. 371–418. Springer.
- Marin, J.-M. and C. Robert (2007). *Bayesian core: a practical approach to computational Bayesian statistics*. Springer Science & Business Media.
- Martino, S. (2007). Approximate Bayesian inference for multivariate stochastic volatility models. Technical report, Department of Mathematical Sciences, Norwegian University of Science and Technology, Trondheim, Norway.
- Martino, S., K. Aas, O. Lindqvist, L. R. Neef, and H. Rue (2011). Estimating stochastic volatility models using integrated nested Laplace approximations. *The European Journal of Finance* 17(7), 487–503.
- Metropolis, N., A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller (1953). Equation of state calculations by fast computing machines. *The journal of chemical physics* 21(6), 1087–1092.
- Meyer, K., G. Hall, and D. Offin (2008). *Introduction to Hamiltonian dynamical systems and the N-body problem*, Volume 90. Springer Science & Business Media.
- Morck, R., B. Yeung, and W. Yu (2000). The information content of stock markets: why do emerging markets have synchronous stock price movements? *Journal of financial economics* 58(1-2), 215–260.

- Neal, R. M. et al. (2011). MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo* 2(11).
- Nelsen, R. B. (2007). *An introduction to copulas*. Springer Science & Business Media.
- Nelson, D. B. (1988). The time series behavior of stock market volatility and returns. *Unpublished PhD Dissertation, Massachusetts Institute of Technology, Cambridge, MA*.
- Nikoloulopoulos, A. K. and D. Karlis (2009). Modeling multivariate count data using copulas. *Communications in Statistics-Simulation and Computation* 39(1), 172–187.
- Park, E. and D. Lord (2007). Multivariate Poisson-lognormal models for jointly modeling crash frequency by severity. *Transportation Research Record: Journal of the Transportation Research Board* (2019), 1–6.
- Pitt, M., R. Silva, P. Giordani, and R. Kohn (2010). Auxiliary particle filtering within adaptive Metropolis-Hastings sampling. *arXiv preprint arXiv:1006.1914*.
- Pitt, M. K. and N. Shephard (1999). Filtering via simulation: Auxiliary particle filters. *Journal of the American statistical association* 94(446), 590–599.
- Platanioti, K., E. McCoy, and D. Stephens (2005). A review of stochastic volatility: univariate and multivariate models. Technical report, Working paper.
- Richard, J.-F. and W. Zhang (2007). Efficient high-dimensional importance sampling. *Journal of Econometrics* 141(2), 1385–1411.
- Robert, C. P. and G. Casella (1999). The Metropolis—Hastings algorithm. In *Monte Carlo Statistical Methods*, pp. 231–283. Springer.
- Robert, C. P. and G. Casella (2005). *Monte Carlo Statistical Methods (Springer Texts in Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc.

- Roberts, G. O., A. Gelman, W. R. Gilks, et al. (1997). Weak convergence and optimal scaling of random walk Metropolis algorithms. *The annals of applied probability* 7(1), 110–120.
- Roberts, G. O. and J. S. Rosenthal (1998). Optimal scaling of discrete approximations to Langevin diffusions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 60(1), 255–268.
- Rue, H., S. Martino, and N. Chopin (2009). Approximate Bayesian inference for latent Gaussian models by using Integrated Nested Laplace Approximations. *Journal of the royal statistical society: Series b (statistical methodology)* 71(2), 319–392.
- Ruiz, E. (1994). Quasi-maximum likelihood estimation of stochastic volatility models. *Journal of Econometrics* 63(1), 289–306.
- Salimans, T., D. A. Knowles, et al. (2013). Fixed-form variational posterior approximation through stochastic linear regression. *Bayesian Analysis* 8(4), 837–882.
- Särkkä, S. (2013). *Bayesian filtering and smoothing*, Volume 3. Cambridge University Press.
- Scharth, M. and R. Kohn (2016). Particle efficient importance sampling. *Journal of Econometrics* 190(1), 133–147.
- Schoenmaker, D. (1996). *Contagion risk in banking*. LSE Financial Markets Group.
- Schotman, P. C. and H. K. Van Dijk (1991). On bayesian routes to unit roots. *Journal of Applied Econometrics* 6(4), 387–401.
- Shephard, N. and G. Torben (2009). Stochastic volatility: origins and overview. In *Handbook of financial time series*, pp. 233–254. Springer.
- Sklar, M. (1959). *Fonctions de Répartition À N Dimensions Et Leurs Marges*. Université Paris 8.

- Šmídl, V. and A. Quinn (2006). *The variational Bayes method in signal processing*. Springer Science & Business Media.
- Trivedi, P. and D. Zimmer (2017). A note on identification of bivariate copulas for discrete count data. *Econometrics* 5(1), 10.
- van Ravenzwaaij, D., P. Cassey, and S. D. Brown (2016). A simple introduction to Markov Chain Monte–Carlo sampling. *Psychonomic bulletin & review*, 1–12.
- Ver Hoef, J. M. and P. L. Boveng (2007). Quasi-Poisson vs. negative binomial regression: How should we model overdispersed count data? *Ecology* 88(11), 2766–2772.
- Wiggins, J. B. (1987). Option values under stochastic volatility: Theory and empirical estimates. *Journal of financial economics* 19(2), 351–372.
- Winkelmann, R. (2013). *Econometric analysis of count data*. Springer Science & Business Media.
- Yu, J. (2005). On leverage in a stochastic volatility model. *Journal of Econometrics* 127(2), 165–178.
- Yu, J. and R. Meyer (2006). Multivariate stochastic volatility models: Bayesian estimation and model comparison. *Econometric Reviews* 25(2–3), 361–384.
- Zellner, A. (1971). *An introduction to Bayesian inference in econometrics*, Volume 156. Wiley New York.
- Zhang, C., N. Chen, and Z. Li (2017). State space modeling of autocorrelated multivariate Poisson counts. *IIEE Transactions* 49(5), 518–531.
- Zhu, F. (2011). A negative binomial integer-valued GARCH model. *Journal of Time Series Analysis* 32(1), 54–67.

Nederlandse Samenvatting

Als wetenschappers proberen we de wereld te begrijpen door verhelderende vragen te stellen. Ontwikkelingen op het gebied van statistiek en informatica leveren een steeds breder scala aan methoden om antwoorden op deze vragen te vinden. De analyse van experimentele en observationele data staat hierbij centraal. Zowel binnen als buiten de econometrie moet de onderzoeker echter ook rekening te houden met niet-geobserveerde variabelen. Voorbeelden zijn: (1) variabelen die niet voldoende vaak waargenomen zijn (bijvoorbeeld economische grootheden die slechts maandelijks maar niet wekelijks of dagelijks vastgelegd zijn) of (2) variabelen die we veronderstellen te bestaan, maar die niet waargenomen kunnen worden (bijvoorbeeld de volatiliteit van financiële tijdreeksen). In dit proefschrift modelleren we deze latente variabelen met behulp van niet-lineaire ‘state-space’ modellen. Een voorbeeld van deze soort modellen zijn de zogenaamde stochastische volatiliteitsmodellen. Deze modellen worden gedetailleerd besproken in de hoofdstukken 2 en 3. Het onderwerp van hoofdstuk 4 is teldata. Bij teldata is er veelal sprake van aantallen en zijn de uitkomsten dus gehele getallen. Het gebruik van latente processen maakt het hierbij mogelijk om complexe verbanden in deze tijdreeksen te modelleren.

In hoofdstuk 2 bespreken we de methoden uit de Bayesiaanse statistiek en ‘machine learning’ die recentelijk gebruikt zijn om stochastische volatiliteitsmodellen te schatten. Deze schattingsmethodes zijn meestal al inge-

wikkeld als we een enkele financiële tijdreeks willen analyseren. Een goed begrip van de voor- en nadelen is dus noodzakelijk voordat multivariate uitbreidingen aan bod komen. Er is meestal een compromis nodig tussen de nauwkeurigheid van de schatting en de benodigde rekentijd om tot die schatting te komen. Daarom besteden we aandacht aan zowel nauwkeurige maar gecompliceerde methoden, als snelle maar minder nauwkeurige methoden. We bespreken deze methoden vanuit vier perspectieven. Allereerst bespreken we rekenkundige complexiteit en de nauwkeurigheid van de schatting. Ten tweede bespreken we de flexibiliteit van de schattingsmethoden. Als derde onderzoeken we of deze methoden toegepast kunnen worden in multivariate en hoog-dimensionele modellen. Dit is van belang, aangezien sommige methoden voordelen bieden in het univariate geval (zoals geïntegreerde, geneste, Laplace Approximations met sparse matrix computations) die niet langer gelden in hogere dimensies. Als laatste bespreken we welke methode volgens ons de beste procedure is voor de schatting van multivariate stochastische volatiliteitsmodellen.

In hoofdstuk 3 introduceren we een methode om volatiliteit-spillovers te kwantificeren met behulp van multivariate stochastische modellen. Het begrijpen van de verbanden tussen de financiële markten van verschillende regio's/landen kan beleidsmakers en portfolio managers helpen om betere beslissingen te nemen. Allereerst gebruiken we de inzichten van hoofdstuk 2 om deze multivariate modellen Bayesiaans te schatten. Hierna leggen we uit hoe deze modellen ons in staat stellen de volatiliteit-spillovers te meten. Het Bayesiaanse framework levert ook de posterior verdeling van deze spillovers zodat we het interval met de hoogste posterior density kunnen gebruiken om de onzekerheden omtrent deze volatiliteit-volatiliteit-spillovers te kwantificeren. We illustreren onze methode aan de hand van verschillende gesimuleerde voorbeelden en twee empirische toepassingen. We beperken ons tot een maximum van vijf latente processen. Er zijn meerdere uitbreidingen denkbaar. Een combinatie van exacte methodes en benaderingen kan voordeling zijn bij het vinden van een evenwicht tussen de rekentijd en de precisie van de schatting. Daarnaast kunnen methoden die de geometrische informatie in de posterior distribution gebruiken (zoals Hamiltonian

Monte Carlo) gebruikt worden om nog meer latente processen toe te staan.

Het modelleren van multivariate teldata is het onderwerp van hoofdstuk 4. We bespreken zowel parameter- als observatie-gedreven modellen en vergelijkingen deze modellen op basis van hun voorspellende kwaliteiten. We bespreken allereerst enkele moeilijkheden en mogelijke oplossingen. De univariate Poisson verdeling wordt veelvuldig gebruikt om univariate teldata te beschrijven maar de multivariate uitbreiding is te gecompliceerd om bruikbaar te zijn. Andere modellen, zoals bijvoorbeeld ‘Autoregressive Conditional Intensity’ modellen, vereisen positieve parameters. Deze positieve parameters impliceren dat zulke modellen alleen in staat zijn processen te beschrijven waarin de correlatie positief is. Een log-lineaire specificatie laat ook negatieve parameters toe. Het blijft echter lastig om de multivariate verdeling van teldata correct te beschrijven. Als laatste alternatief stellen we voor om niet-lineaire ‘state space’ modellen te gebruiken. Deze klasse van modellen is zeer flexibel: zowel positieve als negatieve correlaties zijn mogelijk, evenals positieve en negatieve Granger-causal feedback. We vergelijken de modelvoorspellingen aan de hand van verschillende criteria en komen tot de conclusie dat het gebruik van ‘state space’ modellen de voorkeur heeft.

Curriculum Vitae

Yuliya Shapovalova was born on 17 June 1990 in Academic City, Novosibirsk, Russia. Between 2004 and 2007 she attended high school, Gymnasium 5, and specialized in mathematics and economics. She continued her education at Novosibirsk State University where she obtained her Bachelor's degree in economics and cybernetics. Subsequently, she studied econometrics at Maastricht University where she obtained two MSc degrees between 2012 and 2014. Her master theses are entitled "Conditional Independence and Markov Equivalence in (Granger-causal) Graphical Models" and "Graphical Multivariate Stochastic Volatility Models: Estimation and Model Selection".

After graduation, Yuliya started her Ph.D. under the supervision of Dr. Michael Eichler and prof. Jean-Pierre Urbain. Findings of her research are presented in this dissertation. Yuliya presented her research at multiple scientific conferences, workshops, and seminars. In spring 2016 she was a visiting graduate student for two months in Aix-Marseille University with local host Professor Michel Lubrano. In fall 2016 she was a visiting graduate student for two months in University Paris-Dauphine with local host Professor Christian P. Robert. In January 2018 she joined the division of informatics, imaging, and data science at The University of Manchester as a research associate. In September 2018 she joined data science group at Radboud University as postdoctoral researcher.