Suitably impressive thesis title

Lennart Golks
Department of Physics

A thesis submitted for the degree of
Doctor of Philosophy

October 2025

Acknowledgements

Personal

I would like to thank Alex Elliott for his wonderful help and support. None of this would be possible otherwise.

Institutional

If you want to separate out your thanks for funding and institutional support, I don't think there's any rule against it. Of course, you could also just remove the subsections and do one big traditional acknowledgement section.

Abstract

Your abstract text goes here. Check your departmental regulations, but generally this should be less than 300 words. See the beginning of Chapter **??** for more.

Contents

List of Figures

List of Abbreviations

DAG . . . . . Directed Acyclic Graph

HITRAN . . . . . High Resolution Transmission

IATC . . . . . Integrated Autocorrelation Time

IRT . . . . . Inverse Rosenblatt Transform

L . . . . . Linear

MCMC . . . . . Markov Chain Monte-Carlo

MH . . . . . Metropolis-Hastings

MLS . . . . . Microwave Limb Sounder

MTC . . . . . Marginal and Then Conditional

MVN . . . . . Multivariate Normal

MWG . . . . . Metropolis Wihtin Gibbs

NASA . . . . . National Aeronautics and Space Administration

NL . . . . . Non-Linear

RMS . . . . . Root Mean Square

RTE . . . . . Radiative Transfer Equation

RTO . . . . . Randomise Then Optimise

SIRT . . . . . Squared Inverse Rosenblatt Transform

STD . . . . . Standard Deviation

SVD . . . . . Singular Value Decomposition

TT . . . . . Tensor-Train

VMR . . . . . Volume Mixing Ratio

1 Introduction

Here, we briefly describe the currently used standard to retrieve atmospheric trace gas concentrations and what motivates us to employ a hierarchical Bayesian framework on an atmospheric limb sounder measuring ozone, where we contribute to existing methods and how we improve those. Lastly, we provide the reader with the thesis structure.

1.1 Motivation

Since the only currently operating ozone limb sounder, the Microwave Limb Sounder (MLS) on NASA's Aura satellite, is gradually drifting from its orbit and scheduled to be phased out by 2026 [1], a group led by Harald Schwefel has proposed an alternative approach to fill this observational gap using a much smaller platform, such as a disk-shaped resonator mounted on a 6U CubeSat [2]. The proposed system targets a narrow frequency band and converts the thermal radiation emitted by ozone from the terahertz region to the optical domain [3, 4].

This conversion offers a cost-effective and energy-efficient solution, as it circumvents the need for large, energy-hungry cooling devices that are traditionally required to capture terahertz signals. Instead, signal acquisition in the optical domain can be implemented by using compact, cheap, and low-power photonic technologies.

Currently, the inverse problem to retrieve any trace gas from limb-sounding data is approached by the atmospheric physics community using optimisation and regularisation techniques developed in the 1970s [5, 6]. These approaches are based on a "best fit to data but not the best fit to parameters"[7]. Instead, we employ a hierarchically structured Bayesian framework to infer ozone concentrations, where we find a distribution of parameters given some data. This probabilistic approach provides estimates of parameters and their true uncertainties.

1.2 Research Gap and Contribution

As already mentioned, currently the MLS retrieval algorithm [8] is based on the "optimal estimation" method from [5]. This approach iteratively minimises a squared residual norm by fitting parameters to a set of data and penalises against a chosen regularisation. This does not provide comprehensive information about the parameters, the underlying correlation structures, and can lead to unphysical results, e.g. negative ozone concentration values [9]. The errors provided are based on a local derivative of the forward map around one optimal solution, which is inherently highly sensitive to its location. Additionally, these retrievals are conditioned on external estimates of other parameters, such as temperature or pressure [8]. This results in biased solutions, where the bias is then removed based on empirical decisions [10, 11]. Current machine learning efforts condition on one single noise value in the developed model, which is trained for about one month, and do not include noise as a retrieval parameter. Additionally they compare to a "ground truth" provided by the previously described optimal estimation approach [12, 13].

We address these limitations by including measurement noise as a hyper-paramter as well as the smoothness of the ozone profile explicitly in the modelling and inversion process. Then we are able to provide a range of ozone profiles, which are all fitting to the data. Naturally, noise (hyper-parameter) is a random process and according to that noise we deal with distributions over hyper-parameters and parameters (e.g. ozone concentrations) and can provide errors according to those distributions, instead of one "optimal" solution. This approach is called *hierarchical* Bayesian modelling. Livesey et al. [8] report "unexpected spectrally correlated noise" on the MLS aura, so here is another real reason why one should model and estimate noise.

To solve this inverse problem we employ a linear-Gaussian hierarchical Bayesian framework, we apply the marginal and then conditional (MTC) method [14]. Within second we can evaluate distributions over both hyper-parameters (e.g. noise and ozone smoothness) and parameters. This is a fairly new method within the Bayesian community, and we are the first, to our knowledge, to apply it to a forward model based on the radiative transfer equation (RTE). Then, instead of sampling from those posterior distributions, we are the first to utilise a tensor-train (TT) to approximate the posterior distribution, which enables us to provide estimates and uncertainties via quadrature or the inverse Rosenblatt transform (IRT).

Since the RTE is weakly non-linear, we approximate the RTE with an affine map, which seems to be another novelty in the field of atmospheric remote sensing. Additionally, we provide a new approach to tackle this inverse problem by jointly inferring pressure, temperature and ozone profiles given one set of measurements.

1.3 Thesis Structure

In Ch. 2, we give a brief overview of the methods used and provide references for more details. Then, in Ch. 3, we provide the forward model based on a simplified RTE, and discuss how to measure most effectively. Using our findings, we simulate some noisy data for an idealised limb sounder within a simplified atmosphere based on the RTE. Then, in Ch. 4, we setup our linear hierarchical Bayesian model and discuss some prior modelling. Given the simulated data, we provide posterior distributions of our Bayesian framework based on the linearised RTE to then approximate the non-linear forward model with an affine map. We compare a regularisation solution with the posterior distributions of the approximated linear Bayesian model against a ground truth ozone profile, where we also provide posterior distributions over hyper-parameters. Additionally, we condition on an ozone profile and noise sample to give joint pressure and temperature posterior profiles. Furthermore, we assess and discuss some errors of the approximation used to provide arguments for choices made regarding those approximations. Lastly, we discuss our results and provide an outlook, see Ch. 5.

2 Theoretical and Technical Background

In this chapter, we provide introductions and brief derivations of the methods used in this thesis, as well as references for more details. We keep it as general as possible, as the expressions specifically tailored towards the forward map will be presented in the results Chapter 4. We begin by introducing a general hierarchical Bayesian approach to an inverse problem. Next, we provide the basics of Metropolis-Hastings sampling, more specifically, the essentials of Markov-Chain Monte Carlo (MCMC) methods. Further, we explain how we approximate functions using a Tensor-Train (TT) approach, which enables us to calculate marginals from the posterior distribution cheaply. Then, we elaborate on the Wasserstein distance for assessing upper error bounds. Lastly, we provide some background information on affine maps and the Tikhonov regularisation method.

2.1 Hierarchical Bayesian Inference

Assume we observe some data

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{x}) + \boldsymbol{\eta}, \tag{2.1}$$

based on a forward model $\boldsymbol{A}(\boldsymbol{x})$, which may be non-linear, a unknown parameter $\boldsymbol{x}$ and some additive random noise $\boldsymbol{\eta}$. Naturally, due to the noise, which we classify through a hyper-parameter, we deal with a random process, and we wish to include that in our hierarchically ordered modelling. Then define the likelihood function $\pi(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{x})$ according to the nature of the noise as well as all relevant information about the measurement process, captured by the model $\boldsymbol{A}(\boldsymbol{x})$. We read $\pi(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{x})$ as the distribution over $\boldsymbol{y}$ conditioned on $\boldsymbol{x}$ and $\boldsymbol{\theta}$, and $\boldsymbol{\eta} \sim \pi_{\boldsymbol{\eta}}(\cdot|\boldsymbol{\theta})$ as $\boldsymbol{\eta}$ is distributed as $\pi_{\boldsymbol{\eta}}(\cdot|\boldsymbol{\theta})$. Here $\boldsymbol{\theta}$ may account for multiple hyper-parameters, e.g. describing the distribution $\pi_{\boldsymbol{\eta}}(\cdot|\boldsymbol{\theta})$ over the

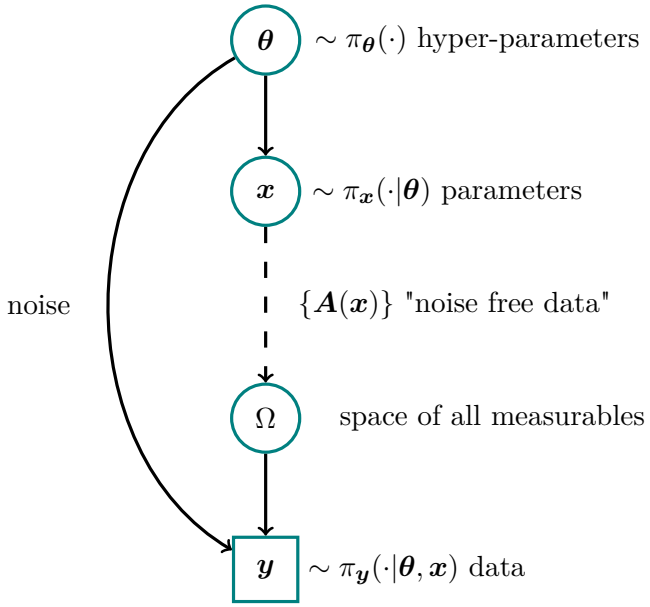


Figure 2.1: The directed acyclic graph (DAG) for an inverse problem visualises statistical dependencies as solid line arrows and deterministic dependencies as dotted arrows. The hyper-parameters $\boldsymbol{\theta}$ are distributed as the hyper-prior distribution $\pi(\boldsymbol{\theta})$. The prior distribution $\pi_{\boldsymbol{x}}(\cdot|\boldsymbol{\theta})$ for the parameter $\boldsymbol{x}$ and the noise are statistically dependent on some of those hyper-parameters. Then a parameter $\boldsymbol{x} \sim \pi_{\boldsymbol{x}}(\cdot|\boldsymbol{\theta})$ is mapped onto the space of all measurables $\boldsymbol{\Omega} = \boldsymbol{A}(\boldsymbol{x})$ deterministically through the forward model. From the space of all measurable noise free data we observe a data set $\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{x}) + \boldsymbol{\eta}$ with some random noise $\boldsymbol{\eta} \sim \pi_{\boldsymbol{\eta}}(\cdot|\boldsymbol{\theta})$, which determines the likelihood function $\pi(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{x})$.

noise vector $\boldsymbol{\eta}$, and describing physical properties or functional dependencies of $\boldsymbol{x}$, e.g. smoothness, trough the prior distribution $\pi(\boldsymbol{x}|\boldsymbol{\theta})$. Consequently we define a hyper-prior distribution $\pi(\boldsymbol{\theta})$, where $\pi(\boldsymbol{x}, \boldsymbol{\theta}) = \pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})$. Choosing these prior distributions is ultimately a modeller's choice and crucial, as it shall not affect the posterior distribution

$$\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) = \frac{\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})\pi(\boldsymbol{x}, \boldsymbol{\theta})}{\pi(\boldsymbol{y})} \propto \pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})\pi(\boldsymbol{x}, \boldsymbol{\theta})\,, \tag{2.2}$$

which according to Bayes' theorem, gives us a distribution of $\boldsymbol{x}$ and $\boldsymbol{\theta}$ given (conditioned on) the data. Note that here we include the hyper-parameters within the posterior distribution, which is the key idea of hierarchical Bayesian modelling, as we not only aim to quantify the posterior distribution over the parameters $\boldsymbol{x}$, but also the posterior distribution over the hyper-parameter $\boldsymbol{\theta}$. We can visualise this hierarchically-ordered correlation structure between parameters as well as how distributions progress through a measurement process, using a directed acyclic graph (DAG), see Figure 2.1.

The expectation of any function $h(\boldsymbol{x}_{\boldsymbol{\theta}})$, where $\boldsymbol{x}$ may depend on $\boldsymbol{\theta}$, is described as

$$\mathrm{E}_{\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y}}[h(\boldsymbol{x}_{\boldsymbol{\theta}})] = \underbrace{\int\int h(\boldsymbol{x}_{\boldsymbol{\theta}})\,\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})\;\mathrm{d}\boldsymbol{x}\;\mathrm{d}\boldsymbol{\theta}}_{\boldsymbol{\mu}_{\mathrm{int}}}\,. \tag{2.3}$$

If that is a high-dimensional integral and computationally not feasible to solve, we approximate

$$\mathrm{E}_{\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y}}[h(\boldsymbol{x}_{\boldsymbol{\theta}})] \approx \underbrace{\frac{1}{N}\sum_{k=1}^{N} h(\boldsymbol{x}_{\boldsymbol{\theta}}^{(k)})}_{\boldsymbol{\mu}_{\text{samp}}}, \tag{2.4}$$

with an the unbiased sample-based Monte Carlo estimate [15] for large enough N (law of large numbers [16, Chapter 17]). Here, the samples $\{\boldsymbol{x}^{(k)}, \boldsymbol{\theta}^{(k)}\} \sim \pi_{\boldsymbol{x},\boldsymbol{\theta}}(\cdot|\boldsymbol{y})$, for $k = 1, \ldots, N$, form a sample set $\mathcal{M} = \{(\boldsymbol{x}, \boldsymbol{\theta})^{(1)}, \ldots, (\boldsymbol{x}, \boldsymbol{\theta})^{(N)}\}$.

Generating a representative sample set quickly from the posterior distribution often presents a significant challenge. This is mainly due to the strong correlations that usually exist between the parameters and hyper-parameters, as discussed by Rue and Held in [17] and illustrated in Appendix A.1. If $\boldsymbol{x}$ cannot be parametrised directly in terms of the hyper-parameters $\boldsymbol{\theta}$, so that $\boldsymbol{x}(\boldsymbol{\theta})$ is function of $\boldsymbol{\theta}$, it is beneficial to factorise the posterior distribution as

$$\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) = \pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})\, \pi(\boldsymbol{\theta}|\boldsymbol{y}), \tag{2.5}$$

into the conditional posterior $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ over the latent field $\boldsymbol{x}$ and the marginal posterior

$$\pi(\boldsymbol{\theta}|\boldsymbol{y}) = \frac{\pi(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{x})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})\pi(\boldsymbol{y})} \propto \frac{\pi(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{x})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})}, \tag{2.6}$$

over the hyper-parameters $\boldsymbol{\theta}$ (see [14, Lemma 2]). In [18], they classify inverse problems into problems with known or unknown conditional posterior distributions, and conclude that if $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y}) = \pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})\pi(\boldsymbol{x}|\boldsymbol{\theta})/\pi(\boldsymbol{y}|\boldsymbol{\theta})$ has a known form, the normalising constant of $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ is available
$\int \pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})\pi(\boldsymbol{x}|\boldsymbol{\theta})\mathrm{d}\boldsymbol{x} = \pi(\boldsymbol{y}|\boldsymbol{\theta}) \propto \pi(\boldsymbol{\theta}|\boldsymbol{y})/\pi(\boldsymbol{\theta})$ and one can almost surely determine the $\boldsymbol{\theta}$-dependence of the marginal posterior $\pi(\boldsymbol{\theta}|\boldsymbol{y})$.

This approach, known as the marginal and then conditional (MTC) method, is particularly advantageous when $\boldsymbol{x} \in \mathbb{R}^n$ is high-dimensional, while $\boldsymbol{\theta} \in \mathbb{R}^{n_{\boldsymbol{\theta}}}$ is low-dimensional, so that $n_{\boldsymbol{\theta}} \ll n$ and evaluation of $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ is cheap. Applying the law of total expectation [19], Eq. (2.3) becomes

$$\mathbb{E}_{\boldsymbol{x}|\boldsymbol{y}}[h(\boldsymbol{x})] = \mathbb{E}_{\boldsymbol{\theta}|\boldsymbol{y}}\left[\mathbb{E}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}[h(\boldsymbol{x}_{\boldsymbol{\theta}})]\right] = \int \mathbb{E}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}\left[h(\boldsymbol{x}_{\boldsymbol{\theta}})\right]\, \pi(\boldsymbol{\theta}|\boldsymbol{y})\, \mathrm{d}\boldsymbol{\theta}, \tag{2.7}$$

where, in the case of a linear-Gaussian hierarchical Bayesian model, both the marginal distribution and the inner expectation $\mathbb{E}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}\left[h(\boldsymbol{x}_{\boldsymbol{\theta}})\right]$ are well defined (see next subsection). Furthermore, the central limit theorem states that the sample mean $\boldsymbol{\mu}_{\text{samp}}^{(i)}$, of independent sample sets $\mathcal{M}_i$ for $i = 1, \ldots, n$ of any distribution, converges to be normally distributed, so that

$$\sqrt{n}(\boldsymbol{\mu}_{\text{samp}}^{(i)} - \boldsymbol{\mu}_{\text{int}}) \xrightarrow{\mathcal{D}} \mathcal{N}(0, \sigma^2)[20], \tag{2.8}$$

and if $\sigma^2 < \infty$ the Monte-Carlo error $\boldsymbol{\mu}_{\text{samp}}^{(i)} - \boldsymbol{\mu}_{\text{int}}$ is bounded.

Integrated Autocorrelation time

To assess the error σ^2 of chain $\mathcal{M}_i$, we ignore systematic error due to initialisation bias (burn-in period), but we have to take into account that samples produced by any system or algorithm are correlated. To derive the integrated autocorrelation time (IATC), we follow the lecture notes [21]. In general, the error of a Monte-Carlo-based estimate from a sample set $\mathcal{M}_i = \{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(k)}, \dots, \boldsymbol{x}^{(s)}, \dots, \boldsymbol{x}^{(N)}\} \sim \pi(\boldsymbol{x}|\boldsymbol{y})$ is:

$$(\sigma^{(i)})^2 = \mathrm{var}(\boldsymbol{\mu}^{(i)}_{\mathrm{samp}}) = \mathrm{var}(\mathrm{E}_{\boldsymbol{x}|\boldsymbol{y}}[h(\boldsymbol{x})]) = \left(\frac{1}{N}\sum_{k=1}^{N} h(\boldsymbol{x}^{(k)}) - \boldsymbol{\mu}^{(i)}\right)^2 . \tag{2.9}$$

Expanding this summation, we see that

$$\left(\frac{1}{N}\sum_{k=1}^{N} h(\boldsymbol{x}^{(k)}) - \boldsymbol{\mu}^{(i)}\right)^2 = \frac{\mathrm{var}(\boldsymbol{\mu}^{(i)}_{\mathrm{samp}})}{N^2}\sum_{k,s=1}^{N} \rho(k-s)\,, \tag{2.10}$$

with the normalised auto correlation coefficient $\rho(k-s) = \Gamma(k-s)/\Gamma(0)$ at lag $k-s$, where the auto correlation coefficient $\Gamma(k-s) = \big(h(\boldsymbol{x}^{(k)}) - \boldsymbol{\mu}^{(i)}\big)\big(h(\boldsymbol{x}^{(s)}) - \boldsymbol{\mu}^{(i)}\big)$ and $\Gamma(0) = \mathrm{var}(\boldsymbol{\mu}^{(i)}_{\mathrm{samp}})$ for $k = s$. Typically $\Gamma(t)$ decays exponentially so that, for $N \gg \tau$, $\Gamma(t) \overset{t\to\infty}{\propto} \exp\{-|t|/\tau\}$ and we can approximate

$$(\sigma^{(i)})^2 \approx \frac{\mathrm{var}(h(\boldsymbol{x}))}{N}\underbrace{\sum_{t=-\infty}^{\infty}\rho(t)}_{:=2\tau_{\mathrm{int}}} = \mathrm{var}(h(\boldsymbol{x}))\frac{2\tau_{\mathrm{int}}}{N}\,, \tag{2.11}$$

where we define the IATC as in [21, pp. 103-105]. See Appendix A.2 and [22] for a more detailed derivation. The IACT provides a good estimate of how many steps the sampling algorithm needs to take to produce one independent sample, accordingly we define the effective sample size as $\frac{2\tau_{\mathrm{int}}}{N}$. Since this is an estimate [23] provides a way to not only calculate the IATC but also to quantify the errors of the estimate of the IATC.

Linear-Gaussian hierarchical Bayesian model

In case of normally distributed noise $\boldsymbol{\eta} \sim \mathcal{N}(0, \boldsymbol{\Sigma}(\boldsymbol{\theta}))$, with zero mean and covariance $\boldsymbol{\Sigma}(\boldsymbol{\theta})$, and a linear model $\boldsymbol{A}$, the data is given as

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}\,, \tag{2.12}$$

and we can derive the marginal and conditional posterior distribution explicitly. We define our hierarchical Bayesian model as

$$\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{A}\boldsymbol{x}, \boldsymbol{\Sigma}(\boldsymbol{\theta})) \tag{2.13a}$$

$$\boldsymbol{x}|\boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{Q}(\boldsymbol{\theta})^{-1}) \tag{2.13b}$$

$$\boldsymbol{\theta} \sim \pi(\boldsymbol{\theta}), \tag{2.13c}$$

with a Gaussian likelihood function $\pi(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{\theta})$, the prior mean $\boldsymbol{\mu}$, prior precision $\boldsymbol{Q}(\boldsymbol{\theta})$ and a hyper-prior distribution $\pi(\boldsymbol{\theta})$. To derive the marginal posterior and the conditional posterior distribution, we consider the joint multivariate Gaussian distribution

$$\begin{pmatrix}\boldsymbol{x}\\ \boldsymbol{y}\end{pmatrix} \sim \mathcal{N}\left[\begin{pmatrix}\boldsymbol{\mu}\\ \boldsymbol{A\mu}\end{pmatrix}, \begin{pmatrix}\boldsymbol{Q}(\boldsymbol{\theta})+\boldsymbol{A}^T\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\boldsymbol{A} & -\boldsymbol{A}^T\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\\ \boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\boldsymbol{A} & \boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\end{pmatrix}^{-1}\right], \tag{2.14}$$

where we provide the joint precision matrix as in [24], see also [14, 17]. Immediately, we formulate the conditional posterior as

$$\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\mu}+(\boldsymbol{Q}(\boldsymbol{\theta})+\boldsymbol{A}^T\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\boldsymbol{A})^{-1}(\boldsymbol{y}-\boldsymbol{A\mu}),(\boldsymbol{Q}(\boldsymbol{\theta})+\boldsymbol{A}^T\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\boldsymbol{A})^{-1}). \tag{2.15}$$

Then the marginal posterior distribution over the hyper-parameters can be derived as in Eq. 2.6, where, as noted in [14], the parameter $\boldsymbol{x}$ cancels and we arrive at

$$\begin{aligned}\pi(\boldsymbol{\theta}|\boldsymbol{y}) \propto &\sqrt{\frac{\det(\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1})\det(\boldsymbol{Q}(\boldsymbol{\theta}))}{\det(\boldsymbol{Q}(\boldsymbol{\theta})+\boldsymbol{A}^T\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\boldsymbol{A})}}\exp\Big\{-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{A\mu})^T\\ &[\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}-\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\boldsymbol{A}(\boldsymbol{Q}(\boldsymbol{\theta})+\boldsymbol{A}^T\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}](\boldsymbol{y}-\boldsymbol{A\mu})\Big\}\pi(\boldsymbol{\theta}).\end{aligned} \tag{2.16}$$

Having the marginal posterior distribution available breaks up the correlation structure between $\boldsymbol{x}$ and $\boldsymbol{\theta}$ (see Appendix A.1), and makes the MTC approach very efficient [14]. This scheme evaluates the marginal posterior values first and then conditions on hyper-parameters to draw posterior samples $\boldsymbol{x} \sim \pi(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta})$ or to evaluate expectation and variance of $\pi(\boldsymbol{x}|\boldsymbol{y})$ by integration over the marginal posterior.

2.2 Sampling Methods

In this section we present the underlying methodology of the sampling methods used in this thesis and show how these methods draw samples
$\mathcal{M} = \{(\boldsymbol{x},\boldsymbol{\theta})^{(1)},\dots,(\boldsymbol{x},\boldsymbol{\theta})^{(k)},\dots,(\boldsymbol{x},\boldsymbol{\theta})^{(N)}\} \sim \pi(\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y})$ from the desired target distribution, so that we can calculate sample-based estimates as in Eq. 2.4. Here, $\mathcal{M}$ denotes a Markov chain, where each new sample $(\boldsymbol{x},\boldsymbol{\theta})^{(k)}$ is only affected by the previous one, $(\boldsymbol{x},\boldsymbol{\theta})^{(k-1)}$. Markov chain Monte Carlo (MCMC) methods generate such a chain $\mathcal{M}$ using random (Monte Carlo) proposals $(\boldsymbol{x},\boldsymbol{\theta})^{(k)} \sim q(\cdot|(\boldsymbol{x},\boldsymbol{\theta})^{(k-1)})$ according to a proposal distribution conditioned on the previous sample (Markov), where ergodicity of the chain $\mathcal{M}$ is a sufficient criterion for using sample-based estimates [7, 15].

The ergodicity theorem in [7] states that, if a Markov chain $\mathcal{M}$ is aperiodic, irreducible, and reversible, then it converges to a unique stationary equilibrium distribution. In other words, if the chain can reach any state from any other state (irreducibility), is not stuck

in periodic cycles (aperiodicity), and is reversible (detailed balance condition [7]). Then the chain converges to the desired target distribution with $\mathcal{M} \sim \pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})$. In practice, one can inspect the trace $\pi(\boldsymbol{x}^{(k)}, \boldsymbol{\theta}^{(k)}|\boldsymbol{y})$ for $k = 1, \ldots, N$ and visually assess convergence and mixing properties of the chain to evaluate ergodicity. The sampling methods used in this thesis possess proven ergodic properties, and we therefore refer the reader to the corresponding literature for further details.

2.2.1 Metropolis within Gibbs

As in Eq. 2.5, when using the MTC method we sample from $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ first and then determine the full conditional $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ as in Eq. 2.7. To sample from $\pi(\boldsymbol{\theta}|\boldsymbol{y})$, we use a Metropolis-within-Gibbs (MWG) sampler as described in [14]. We apply the MWG sample for the two-dimensional case only, with $\boldsymbol{\theta} = (\theta_1, \theta_2)$, where we perform a Metropolis step in the θ_1 direction and a Gibbs step in the θ_2 direction. Ergodicity for this approach is proven in [25].

The Metropolis-within-Gibbs algorithm begins with an initial guess $\boldsymbol{\theta}^{(t)}$ at $t = 0$. We then propose a new sample $\theta_1 \sim q(\theta_1|\theta_1^{(t-1)})$, conditioned on the previous state, using a symmetric proposal distribution $q(\theta_1|\theta_1^{(t-1)}) = q(\theta_1^{(t-1)}|\theta_1)$, which is a special case of the Metropolis-Hastings algorithm [25]. We accept and set $\theta_1^{(t)} = \theta_1$ with the acceptance probability

$$\alpha(\theta_1|\theta_1^{(t-1)}) = \min\left\{1, \frac{\pi(\theta_1|\theta_2^{(t-1)}, \boldsymbol{y})\, \cancel{q(\theta_1^{(t-1)}|\theta_1)}}{\pi(\theta_1^{(t-1)}|\theta_2^{(t-1)}, \boldsymbol{y})\, \cancel{q(\theta_1|\theta_1^{(t-1)})}}\right\} \tag{2.17}$$

or reject and keep $\theta_1^{(t)} = \theta_1^{(t-1)}$, which we do by comparing α to a uniform random number $u \sim \mathcal{U}(0, 1)$.

Next, we perform a Gibbs step in the θ_2 direction, where Gibbs sampling is again a special case of the Metropolis-Hastings algorithm with acceptance probability equal to one, and draw the next sample $\theta_2^{(t)} \sim \pi(\cdot|\theta_1^{(t)}, \boldsymbol{y})$, conditioned on the current value $\theta_1^{(t)}$.

We repeat this procedure N' times and ensure convergence independently of the initial sample (irreducibility) by discarding the initial $N_{\text{burn-in}}$ samples after a so-called burn-in period, resulting in a Markov chain of length $N = N' - N_{\text{burn-in}}$.

2.2.2 `T-walk`Sampler as a Black Box

If the parameters $\boldsymbol{x}$ are functionally dependent on the hyper-parameters $\boldsymbol{\theta}$, i.e., $\boldsymbol{x} = \boldsymbol{x}(\boldsymbol{\theta})$, we can sample directly from the marginal posterior $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ using the `t-walk` sampler as by Christen and Fox [26]. The **t-walk** is employed as a black-box sampler, requiring the specification of the number of samples, burn-in period, support region, and the target distribution. Convergence to the target distribution is guaranteed by construction of the algorithm.

Algorithm 1: Metropolis within Gibbs

Initialise and suppose two dimensional vector $\boldsymbol{\theta}^{(0)} = (\theta_1^{(0)}, \theta_2^{(0)})$
for $k = 1, \ldots, N'$ **do**
  Propose $\theta_1 \sim q(\cdot|\theta_1^{(t-1)}) = q(\theta_1^{(t-1)}|\cdot)$
  Compute

$$\alpha(\theta_1|\theta_1^{(t-1)}) = \min\left\{1, \frac{\pi(\theta_1|\theta_2^{(t-1)}, \boldsymbol{y})\cancel{q(\theta_1^{(t-1)}|\theta_1)}}{\pi(\theta_1^{(t-1)}|\theta_2^{(t-1)}, \boldsymbol{y})\cancel{q(\theta_1|\theta_1^{(t-1)})}}\right\}$$

  Draw $u \sim \mathcal{U}(0, 1)$
  if $\alpha \geq u$ **then**
    Accept and set $\theta_1^{(t)} = \theta_1$
  else
    Reject and keep $\theta_1^{(t)} = \theta_1^{(t-1)}$
  end if
  Draw $\theta_2^{(t)} \sim \pi(\cdot|\theta_1^{(t)}, \boldsymbol{y})$
end for
Output: $\boldsymbol{\theta}^{(0)}, \ldots, \boldsymbol{\theta}^{(k)}, \ldots, \boldsymbol{\theta}^{(N)} \sim \pi(\boldsymbol{\theta}|\boldsymbol{y})$

2.2.3 Randomise then Optimise

If we can not evaluate the mean and variance of the full posterior through integration, e.g. if it is computationally not feasible to solve the integral in Eq. 2.7 due to a large number of hyper-parameters, we need an alternative way to characterise $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$. For the linear-Gaussian Bayesian model we can draw samples from the normal distribution $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ using the randomise then optimise (RTO) [27] scheme after sampling from the marginal posterior $\pi(\boldsymbol{\theta}|\boldsymbol{y})$.

The full conditional distribution can be rewritten as

$$\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}) \propto \pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})\, \pi(\boldsymbol{x}|\boldsymbol{\theta}) \tag{2.18}$$

$$= \exp\left(-\left\|\hat{\boldsymbol{A}}\boldsymbol{x} - \hat{\boldsymbol{y}}\right\|_{L^2}^2\right), \tag{2.19}$$

where we define

$$\hat{\boldsymbol{A}} \coloneqq \begin{bmatrix} \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1/2}\boldsymbol{A}_{\boldsymbol{\theta}} \\ \boldsymbol{Q}_{\boldsymbol{\theta}}^{1/2} \end{bmatrix}, \quad \hat{\boldsymbol{y}} \coloneqq \begin{bmatrix} \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1/2}\boldsymbol{y} \\ \boldsymbol{Q}_{\boldsymbol{\theta}}^{1/2}\boldsymbol{\mu} \end{bmatrix} \quad [28]. \tag{2.20}$$

Here we write $\boldsymbol{A}(\boldsymbol{\theta}) = \boldsymbol{A}_{\boldsymbol{\theta}}$, $\boldsymbol{Q}(\boldsymbol{\theta}) = \boldsymbol{Q}_{\boldsymbol{\theta}}$ and $\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta}) = \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}$, which are all dependent on the hyper-parameters $\boldsymbol{\theta}$. A sample $\boldsymbol{x}^{(k)} \sim \pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ from the conditional posterior is obtained by minimising the following equation with respect to $\hat{\boldsymbol{x}}$:

$$\boldsymbol{x}^{(k)} = \arg\min_{\hat{\boldsymbol{x}}} \|\hat{\boldsymbol{A}}\hat{\boldsymbol{x}} - (\hat{\boldsymbol{y}} + \boldsymbol{b})\|_{L^2}^2, \quad \boldsymbol{b} \sim \mathcal{N}(\boldsymbol{0}, \mathbf{I}), \tag{2.21}$$

where we add a randomised perturbation $\boldsymbol{b}$. Similar to Section 2.5, this expression can be rewritten as

$$\left(\boldsymbol{A}_{\boldsymbol{\theta}}^T \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \boldsymbol{A}_{\boldsymbol{\theta}} + \boldsymbol{Q}_{\boldsymbol{\theta}}\right) \boldsymbol{x}^{(k)} = \boldsymbol{A}_{\boldsymbol{\theta}}^T \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \boldsymbol{y} + \boldsymbol{Q}_{\boldsymbol{\theta}} \boldsymbol{\mu} + \boldsymbol{v}_1 + \boldsymbol{v}_2, \tag{2.22}$$

where the term $-\hat{\boldsymbol{A}}^T \boldsymbol{b}$ is decomposed as $\boldsymbol{v}_1 + \boldsymbol{v}_2$, with $\boldsymbol{v}_1 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{A}_{\boldsymbol{\theta}}^T \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \boldsymbol{A}_{\boldsymbol{\theta}})$ and $\boldsymbol{v}_2 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_{\boldsymbol{\theta}})$, representing independent Gaussian random variables [14, 27].

If the Markov chain over the marginal posterior $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ is ergodic, and the conditional samples $\boldsymbol{x}^{(k)} \sim \pi(\boldsymbol{x}|\boldsymbol{\theta}^{(k)}, \boldsymbol{y})$ are drawn independently, then the resulting joint chain $\{(\boldsymbol{x}, \boldsymbol{\theta})^{(1)}, \ldots, (\boldsymbol{x}, \boldsymbol{\theta})^{(N)}\} \sim \pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) = \pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})\pi(\boldsymbol{\theta}|\boldsymbol{y})$ is also ergodic [29].

2.3 Numerical Approximation – Tensor-Train (TT)

Instead of sampling from a target distribution $\pi(\boldsymbol{x})$ we can approximate that distribution on a d-dimensional grid with far fewer function evaluation compared to sampling methods using a tensor-train (TT) approximation $\tilde{\pi}(\boldsymbol{x}) \approx \pi(\boldsymbol{x})$, with $\boldsymbol{x} \in \mathbb{R}^d$. Here, we explain how to calculate marginal distribution from an approximated probaility density in the TT format and generate samples via the inverse Rosenblatt transform (IRT), following the notation of [30],

As in [30], we can define the parameter space as the product space $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_d$ with $x_k \in \mathcal{X}_k \subseteq \mathbb{R}$ and $\boldsymbol{x} = (x_1, \ldots, x_k, \ldots, x_d)$. Then marginal density function for the k-th component is then given by

$$f_{X_k}(x_k) = \frac{1}{z} \int_{\mathcal{X}_1} \cdots \int_{\mathcal{X}_d} \lambda(\boldsymbol{x})\, \pi(\boldsymbol{x})\, \mathrm{d}x_1 \cdots \mathrm{d}x_{k-1}\, \mathrm{d}x_{k+1} \cdots \mathrm{d}x_d, \tag{2.23}$$

where we integrate over all dimensions except the k-th, and z is a normalisation constant. Here, we introduce a weight function $\lambda(x)$, which can be useful for quadrature rules [31], to which [30] refer to as a "product-form Lebesgue-measurable weighting function" and define it as

$$\lambda(\mathcal{X}) = \prod_{i=1}^{d} \lambda_i(\mathcal{X}_i), \quad \text{where} \quad \lambda_i(\mathcal{X}_i) = \int_{\mathcal{X}_i} \lambda_i(x_i)\, \mathrm{d}x_i. \tag{2.24}$$

In the TT format, the integral in Eq. 2.23 for the marginal probability can be computed at a low computational cost as $\pi(\boldsymbol{x})$ is approximated by

$$\tilde{\pi}(\boldsymbol{x}) = \tilde{\pi}_1(x_1)\tilde{\pi}_2(x_2) \cdots \tilde{\pi}_d(x_d) \in \mathbb{R},$$

which is a sequence of matrix multiplications, with $\tilde{\pi}_k(x_k) \in \mathbb{R}^{r_{k-1} \times r_k}$ for a fixed point $\boldsymbol{x} = (x_1, \ldots, x_d)$ on a predefined d-dimensional discrete univariate grid over the parameter space $\mathcal{X}$. We call $\tilde{\pi}_k \in \mathbb{R}^{r_{k-1} \times n \times r_k}$ a TT-core with ranks $r_{k-1} = r_k = r$, where the outer ranks are $r_0 = r_d = 1$, representing each dimension on n grid points and connecting to neighbouring

dimensions through its ranks. This enables us to approximate $\pi(\mathcal{X}) \approx \tilde{\pi}_1\tilde{\pi}_2\cdots\tilde{\pi}_d$ over the parameter space $\mathcal{X}$ using $2nr + (d-2)nr^2$ evaluation points, as illustrated in Figure 2.2, instead of n^d function evaluation. Consequently, the marginal target distribution

$$f_{X_k}(x_k) \approx \frac{1}{z}\Big| \left(\int_{\mathcal{X}_1} \boldsymbol{\lambda}_1(x_1)\tilde{\pi}_1(x_1)\,\mathrm{d}x_1\right)\cdots\left(\int_{\mathcal{X}_{k-1}} \boldsymbol{\lambda}_{k-1}(x_{k-1})\tilde{\pi}_{k-1}(x_{k-1})\,\mathrm{d}x_{k-1}\right) \boldsymbol{\lambda}_k(x_k)\tilde{\pi}_k(x_k) \left(\int_{\mathcal{X}_{k+1}} \boldsymbol{\lambda}_{k+1}(x_{k+1})\tilde{\pi}_{k+1}(x_{k+1})\,\mathrm{d}x_{k+1}\right)\cdots\left(\int_{\mathcal{X}_d} \boldsymbol{\lambda}_d(x_d)\tilde{\pi}_d(x_d)\,\mathrm{d}x_d\right)\Big| \tag{2.25}$$

is computed by integrating over all TT cores except π_k, as in [32], including a normalisation constant z [30].

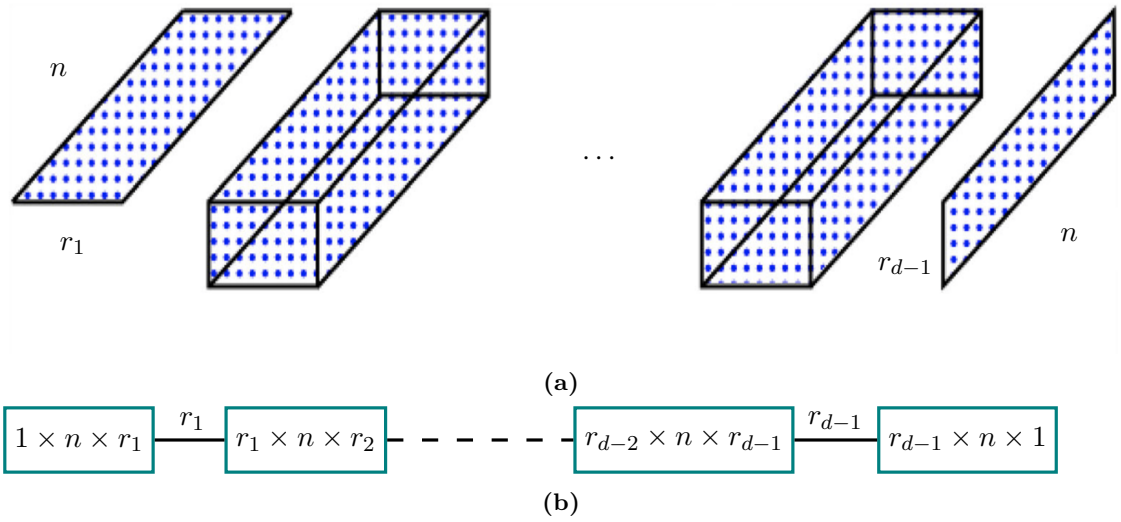


Figure 2.2: Here, we visualise the tensor train cores as two- and three-dimensional matrices. Each core has a length n, corresponding to the number of grid points in one dimension, and the cores are connected through ranks r_k. More specifically, a core $\tilde{\pi}_k$ has dimensions $r_{k-1} \times n \times r_k$, with outer ranks $r_0 = r_d = 1$. Using the TT-format enables us to represent a d-dimensional grid with only $2nr + (d-2)nr^2$ evaluation points instead of n^d grid points. Figure (a) is adapted from [33].

In practice, tensor train approximations may suffer from numerical instability, in particular because it is not advantageous to approximate the target function $\pi(\boldsymbol{x})$ in e.g. the logarithmic space. Hence, Cui et al. [30] approximate the square root of the probability density

$$\sqrt{\pi(\boldsymbol{x})} \approx \tilde{g}(\boldsymbol{x}) = \boldsymbol{G}_1(x_1), \ldots, \boldsymbol{G}_k(x_k), \ldots, \boldsymbol{G}_d(x_d) \text{ [30, Eq. 18]}, \tag{2.26}$$

which ensures positivity. Here, each TT-core is given by

$$G_k^{(\alpha_{k-1},\alpha_k)}(x_k) = \sum_{i=1}^{n_k} \phi_k^{(i)}(x_k)\boldsymbol{A}_k[\alpha_{k-1}, i, \alpha_k], \quad k = 1, \ldots, d, \text{ [30, Eq. 21]}, \tag{2.27}$$

where $\boldsymbol{A}_k \in \mathbb{R}^{r_{k-1}\times n_k \times r_k}$ is the k-th coefficient tensor and $\{\phi_k^{(i)}(x_k)\}_{i=1}^{n_k}$ are the basis functions corresponding to the k-th coordinate. The approximated unnormalised density is written as:

$$\pi(\boldsymbol{x}) \approx \xi + \tilde{g}(\boldsymbol{x})^2 \text{ [30, Eq. 19]}, \tag{2.28}$$

where ξ is a positive constant added according to the ratio of the Lebesgue weighted L2-norm error and the Lebesgue weighting (see Eq. 2.24) such that

$$0 \leq \xi \leq \frac{1}{\lambda(\mathcal{X})}\|\tilde{g} - \sqrt{\pi}\|^2_{L^2_\lambda(\mathcal{X})} \text{ [30, Eq. 35]}. \tag{2.29}$$

This leads to the normalised target function

$$f_X(\boldsymbol{x}) \approx \frac{1}{z}\left(\boldsymbol{\lambda}(\boldsymbol{x})\xi + \boldsymbol{\lambda}(\boldsymbol{x})\tilde{g}(\boldsymbol{x})^2\right) \text{ [30, Eq. 19]}, \tag{2.30}$$

which is the normalisation constant $z = \int_{\mathcal{X}} f_X(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$. Given the tensor train approximation of $\sqrt{\pi}$, the marginal function $f_{X_k}(x_k)$ can be expressed as

$$\begin{aligned} f_{X_k}(x_k) \approx \frac{1}{z}\Bigg(&\xi \prod_{i=1}^{k-1}\lambda_i(\mathcal{X}_i)\prod_{i=k+1}^{d}\lambda_i(\mathcal{X}_i) \\ &+ \left(\int_{\mathcal{X}_1}\lambda_1(x_1)\boldsymbol{G}_1^2(x_1)\,\mathrm{d}x_1\right)\cdots\left(\int_{\mathcal{X}_{k-1}}\lambda_{k-1}(x_{k-1})\boldsymbol{G}_{k-1}^2(x_{k-1})\,\mathrm{d}x_{k-1}\right) \\ &\lambda_k(x_k)\boldsymbol{G}_k^2(x_k) \\ &\left(\int_{\mathcal{X}_{k+1}}\lambda_{k+1}(x_{k+1})\boldsymbol{G}_{k+1}^2(x_{k+1})\,\mathrm{d}x_{k+1}\right)\cdots\left(\int_{\mathcal{X}_d}\lambda_d(x_d)\boldsymbol{G}_d^2(x_d)\,\mathrm{d}x_d\right)\Bigg). \end{aligned} \tag{2.31}$$

2.3.1 Marginal Functions

TT-approximations are handy when approximating integrals, as marginal functions can be easily computed which may simplify the integration significantly. We compute those by a procedure, to which Cui et al. [30] refer to as backwards marginalisation, see Prop. 2, and to which I add the forward marginalisation, see Prop. 1. This is similar to the left and right orthogonalisation of TT-cores [34, 35]. The backwards marginalisation provides us with the coefficient matrices $\boldsymbol{B}_k$, while the forward marginalisation gives the coefficient matrices $\boldsymbol{B}_{\text{pre},k}$. These matrices enable the efficient evaluation of marginal functions since they integrate over the coordinates either left or right of the k-th dimension, as in [30]. In doing so, we define the mass matrix $\boldsymbol{M}_k \in \mathbb{R}^{n_k\times n_k}$ as

$$\boldsymbol{M}_k[i,j] = \int_{\mathcal{X}_k}\phi_k^{(i)}(x_k)\phi_k^{(j)}(x_k)\boldsymbol{\lambda}(x_k)\,\mathrm{d}x_k, \quad i,j = 1,\ldots,n_k, \text{ [30, Eq. 22]}, \tag{2.32}$$

where $\{\phi_k^{(i)}(x_k)\}_{i=1}^{n_k}$ denotes the set of basis functions for the k-th coordinate. The proposition used to compute $\boldsymbol{B}_k$, stated in Prop.1, is adapted directly from [30].

Proposition 1 (Backwards Marginalisation as in [30])**:** Starting with the last coordinate $k = d$, we set $\boldsymbol{B}_d = \boldsymbol{A}_d$. The following procedure can be used to obtain the coefficient tensor $\boldsymbol{B}_{k-1} \in \mathbb{R}^{r_{k-2} \times n_{k-1} \times r_{k-1}}$, which we need for defining the marginal function $f_{X_k}(x_k)$:

1. Use the Cholesky decomposition of the mass matrix, $\boldsymbol{L}_k \boldsymbol{L}_k^\top = \boldsymbol{M}_k \in \mathbb{R}^{n_k \times n_k}$, to construct a tensor $\boldsymbol{C}_k \in \mathbb{R}^{r_{k-1} \times n_k \times r_k}$:

$$\boldsymbol{C}_k[\alpha_{k-1}, \tau, l_k] = \sum_{i=1}^{n_k} \boldsymbol{B}_k[\alpha_{k-1}, i, l_k] \boldsymbol{L}_k[i, \tau] \text{[30, Eq. (27)]}. \tag{2.33}$$

2. Unfold $\boldsymbol{C}_k$ along the first coordinate and compute the thin QR decomposition, so that $\boldsymbol{C}_k^{(R)} \in \mathbb{R}^{r_{k-1} \times (n_k r_k)}$:

$$\boldsymbol{Q}_k \boldsymbol{R}_k = (\boldsymbol{C}_k^{(R)})^\top \text{ [30, Eq. 28]}. \tag{2.34}$$

3. Compute the new coefficient tensor:

$$\boldsymbol{B}_{k-1}[\alpha_{k-2}, i, l_{k-1}] = \sum_{\alpha_{k-1}=1}^{r_{k-1}} \boldsymbol{A}_{k-1}[\alpha_{k-2}, i, \alpha_{k-1}] \boldsymbol{R}_k[l_{k-1}, \alpha_{k-1}] \text{ [30, Eq. 29]}. \tag{2.35}$$

Proposition 2 (Forward Marginalisation)**:** Starting with the first coordinate $k = 1$, we set $\boldsymbol{B}_{\text{pre},1} = \boldsymbol{A}_1$. The following procedure can be used to obtain the coefficient tensor $\boldsymbol{B}_{\text{pre},k+1} \in \mathbb{R}^{r_k \times n_{k+1} \times r_{k+1}}$ for defining the marginal function $f_{X_k}(x_k)$:

1. Use the Cholesky decomposition of the mass matrix, $\boldsymbol{L}_k \boldsymbol{L}_k^\top = \boldsymbol{M}_k \in \mathbb{R}^{n_k \times n_k}$, to construct a tensor $\boldsymbol{C}_k \in \mathbb{R}^{r_{k-1} \times n_k \times r_k}$:

$$\boldsymbol{C}_{\text{pre},k}[\alpha_{k-1}, \tau, l_k] = \sum_{i=1}^{n_k} \boldsymbol{L}_k[i, \tau] \boldsymbol{B}_{\text{pre},k}[\alpha_{k-1}, i, l_k]. \tag{2.36}$$

2. Unfold $\boldsymbol{C}_{\text{pre},k}$ along the first coordinate and compute the thin QR decomposition, so that $\boldsymbol{C}_{\text{pre},k}^{(R)} \in \mathbb{R}^{(r_{k-1} n_k) \times r_k}$:

$$\boldsymbol{Q}_{pre,k} \boldsymbol{R}_{\text{pre},k} = (\boldsymbol{C}_{\text{pre},k}^{(R)}). \tag{2.37}$$

3. Compute the new coefficient tensor $\boldsymbol{B}_{\text{pre},k+1} \in \mathbb{R}^{r_{k-1} \times n_k \times r_k}$:

$$\boldsymbol{B}_{\text{pre},k+1}[l_{k+1}, i, \alpha_{k+1}] = \sum_{\alpha_k=1}^{r_k} \boldsymbol{R}_{\text{pre},k}[l_{k+1}, \alpha_k] \boldsymbol{A}_{k+1}[\alpha_k, i, \alpha_{k+1}]. \tag{2.38}$$

After computing the coefficient tensors $\boldsymbol{B}_{\text{pre},k+1}$ as in Prop. 2 and $\boldsymbol{B}_{k+1}$ from Prop. 1,

the marginal PDF of k-th dimension can be expressed as

$$f_{X_k}(x_k) \approx \frac{1}{z}\left(\xi \prod_{i=1}^{k-1} \lambda_i(X_i) \prod_{i=k+1}^{d} \lambda_i(X_i) + \sum_{l_{k-1}=1}^{r_{k-1}} \sum_{l_k=1}^{r_k} \left(\sum_{i=1}^{n} \phi_k^{(i)}(x_k) \boldsymbol{D}_k[l_{k-1}, i, l_k]\right)^2\right) \lambda_k(x_k), \tag{2.39}$$

where $\boldsymbol{D}_k \in \mathbb{R}^{r_{k-1}\times n\times r_k}$ and $\boldsymbol{R}_{\text{pre},k-1} \in \mathbb{R}^{r_{k-1}\times r_{k-1}}$ and $\boldsymbol{B}_k \in \mathbb{R}^{r_{k-1}\times n\times r_k}$

$$\boldsymbol{D}_k[l_{k-1}, i, l_k] = \sum_{\alpha_{k-1}=1}^{r_{k-1}} \boldsymbol{R}_{\text{pre},k-1}[l_{k-1}, \alpha_{k-1}] \boldsymbol{B}_k[\alpha_{k-1}, i, l_k]. \tag{2.40}$$

For the first dimension, $f_{X_1}(x_1)$ can be expressed as

$$f_{X_1}(x_1) \approx \frac{1}{z}\left(\xi \prod_{i=2}^{d} \lambda_i(\mathcal{X}_i) + \sum_{l_1=1}^{r_1} \left(\sum_{i=1}^{n} \phi_1^{(i)}(x_1) \boldsymbol{D}_1[i, l_1]\right)^2\right) \lambda_1(x_1) \text{ [30, Eq. 30]}, \tag{2.41}$$

where $\boldsymbol{D}_1[i, l_1] = \boldsymbol{B}_1[\alpha_0, i, l_1]$ and $\alpha_0 = 1$, and similarly in the last dimension

$$f_{X_d}(x_d) \approx \frac{1}{z}\left(\xi \prod_{i=1}^{d-1} \lambda_i(\mathcal{X}_i) + \sum_{l_{n-1}=1}^{r_{d-1}} \left(\sum_{i=1}^{n} \phi_1^{(i)}(x_1) \boldsymbol{D}_d[l_{n-1}, i]\right)^2\right) \lambda_d(x_d), \tag{2.42}$$

where $\boldsymbol{D}_d[l_{n-1}, i] = \boldsymbol{B}_{\text{pre},d}[l_{n-1}, i, \alpha_{n+1}]$ and $\alpha_{d+1} = 1$. Note that we calculate the normalisation numerically within the process of finding the marginals so that $\sum f_{X_k}(x_k) = 1$.

2.3.2 Sampling from a TT Approximation

If instead of evaluating integrals we like to draw samples from the approximated function we do this via the inverse Rosenblatt transform (IRT), as in [32], to preserve the correlation structure. Since we approximate the square root of the target function, Cui et. al. [30] call that the squared inverse Rosenblatt transform (SIRT).

Algorithm 2: Squared Inverse Rosenblatt Transform (SIRT)

Input: seeds $\{\boldsymbol{u}^{(1)}, \ldots, \boldsymbol{u}^{(N)}\} \sim \mathcal{U}(0,1)^d$ and $\boldsymbol{B}_1, \ldots, \boldsymbol{B}_d$ from Prop. 1
for $s = 1, \ldots, N$ **do**
  for $k = 1, \ldots, d$ **do**
    compute normalised PDF $f_{X_k|X_{<k}}(x_k|x_{k-1}^{(s)}, \ldots, x_1^{(s)})$, Eq. 2.45
    compute cumulative distribution function $F_{X_k|X_{<k}}(x_k)$, Eq. 2.43,
    project sample $x_k^{(s)} = F^{-1}_{X_k|X_{<k}}(u_k^{(s)})$
    interpolate $\boldsymbol{G}_k(x_k^{(s)})$, Eq. 2.44
    update $\boldsymbol{G}_{\leq k}(x_{\leq k}^{(s)}) = \boldsymbol{G}_{<k}(x_{<k}^{(s)})\boldsymbol{G}_k(x_k^{(s)})$
  end for
end for
Output: samples $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}$, where each $\boldsymbol{x}^{(s)} \in \mathbb{R}^d$ for $s = 1, \ldots, N$

We start by calculating the Backward marginals $\boldsymbol{B}_1, \ldots, \boldsymbol{B}_d$ as in Prop. 1 and draw N uniformly distributed seeds $\{\boldsymbol{u}^{(1)}, \ldots, \boldsymbol{u}^{(N)}\} \sim \mathcal{U}(0,1)^d$, where each $\boldsymbol{u}^{(s)}$ is d-dimensional

for $s = 1, \dots, N$. Then we calculate the first marginal $f_{X_1}(x_1)$ as in Eq. 2.41 and normalise with $z = \int_{\mathcal{X}_1} f_{X_1}(x_1)dx_1$. Next, we compute the cumulative distribution function (CDF)

$$F_{X_k|X_{<k}}(x_k) \approx \int_{-\infty}^{x_k} f_{X_k|X_{<k}}(\hat{x}_k|x_{k-1}, \dots, x_1)\, \mathrm{d}\hat{x}_k \text{ [30, Eq. 17]} \tag{2.43}$$

for the first dimension $k = 1$ and then project the seed on the parameter space $x_k^{(s)} = F^{-1}_{X_k|X_{<k}}(u_k^{(s)})$. Once that is done, we use a piecewise polynomial interpolation

$$\boldsymbol{G}_k(x_k^{(s)}) \approx \frac{x_k^{(s)} - x_k^{(i)}}{x_k^{(i+1)} - x_k^{(i)}} \boldsymbol{G}_k(x_k^{(i+1)}) + \frac{x_k^{(i+1)} - x_k^{(s)}}{x_k^{(i+1)} - x_k^{(i)}} \boldsymbol{G}_k(x_k^{(i)})\,, \tag{2.44}$$

for $x_k^{(i)} \leq x_k^{(s)} \leq x_k^{(i+1)}$ in between two grid points i and $i+1$ as in [32]. Through $\boldsymbol{G}_k(x_k^{(s)}) \in \mathbb{R}^{1 \times r_{k-1}}$ we condition on the previous samples, which denotes the product of all approximated tensors of the previous $k-1$ samples to preserve the correlation structure. Then we marginalise over the dimensions $k+1, \dots, d$ via $\boldsymbol{B}_k$ so that the next "conditional marginal" is given as:

$$f_{X_k|X_{<k}}(x_k|x_{k-1}^{(s)}, \dots, x_1^{(s)}) \approx \frac{1}{z}\Bigg(\xi \prod_{i=k+1}^{d} \boldsymbol{\lambda}_i(X_i) + \sum_{l_k=1}^{r_k}\Bigg(\sum_{i=1}^{n} \phi_k^{(i)}(x_k^{(s)})\Bigg(\sum_{\alpha_{k-1}=1}^{r_{k-1}} \boldsymbol{G}_{<k}^{(\alpha_{k-1})}(x_{<k}^{(s)}) \boldsymbol{B}_k[\alpha_{k-1}, i, l_k]\Bigg)\Bigg)^2\Bigg)\boldsymbol{\lambda}_k(x_k) \text{ [30, Eq. 31].} \tag{2.45}$$

We repeat the procedure for each $u_k^{(s)} \in \boldsymbol{u}^{(s)}$ to gain samples $\boldsymbol{x}^{(s)} \sim f_X(\boldsymbol{x})$, see algorithmic box 3 for a summarised version.

MH - correction step

Since the samples using the SIRT scheme are samples from an approximation it is sensible to correct those using a Metropolis-Hastings (MH) importance sampler. In doing so we compute the acceptance probability $\alpha = \min(w^{(s+1)}/w^{(s)}, 1)$, where

$$w(x) = \frac{\pi(\boldsymbol{x})}{f_X(\boldsymbol{x})} = \frac{\pi(\boldsymbol{x})}{\gamma + \tilde{g}(\boldsymbol{x})^2} \tag{2.46}$$

is the importance ratio. In practise we calculate the importance ratio in the log space, where $\log f_X(\boldsymbol{x}) = \log f_{X_1}(x_1) + \log f_{X_2|X_1}(x_2|x_1) + \cdots + \log f_{X_k|X_{<k}}(x_k|x_{k-1}, \dots, x_1)$ is given as in Eq. 2.45, see [32]. We refer to this as the SIRT-MH scheme, which provides the corrected chain $\{\boldsymbol{x}_{\text{MH}}^{(1)}, \dots, \boldsymbol{x}_{\text{MH}}^{(N)}\} \sim \pi(\boldsymbol{x})$.

Algorithm 3: MH correction step

Input: samples $\{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(N+1)}\}$, where each $\boldsymbol{x}^{(s)} \in \mathbb{R}^d$ for $s = 1, \dots, N+1$
for $s = 1, \dots, N$ **do**
  compute MH ratio $\frac{w^{(s+1)}}{w^{(s)}} = \frac{\pi(\boldsymbol{x}^{(s+1)})}{\pi(\boldsymbol{x}^{(s)})} \frac{f_X(\boldsymbol{x}^{(s)})}{f_X(\boldsymbol{x}^{(s+1)})}$
  compute acceptance probability $\alpha = \min(w^{(s+1)}/w^{(s)}, 1)$
  Draw $u \sim \mathcal{U}(0,1)$
  if $\alpha \geq u$ **then**
    Accept and set $\boldsymbol{x}_{\text{MH}}^{(s+1)} = \boldsymbol{x}^{(s+1)}$
  else
    Reject and keep $\boldsymbol{x}_{\text{MH}}^{(s+1)} = \boldsymbol{x}^{(s)}$
  end if
end for
Output: corrected sample chain $\{\boldsymbol{x}_{\text{MH}}^{(1)}, \dots, \boldsymbol{x}_{\text{MH}}^{(N)}\}$, where each $\boldsymbol{x}_{\text{MH}}^{(s)} \in \mathbb{R}^d$ for $s = 1, \dots, N$

2.3.3 On the Error of a TT Approximation

A straightforward way to asses the error from the TT approximation is to calculate the relative root mean squared error (RMS)

$$\left(\frac{\int_{\mathcal{X}} (\pi(\boldsymbol{x}) - (\gamma + \tilde{g}(\boldsymbol{x})^2))^2 \boldsymbol{\lambda}(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x}}{\int_{\mathcal{X}} \pi(\boldsymbol{x})^2 \boldsymbol{\lambda}(x) \, \mathrm{d}\boldsymbol{x}} \right)^{1/2} = \frac{\|\pi(\boldsymbol{x}) - (\gamma + \tilde{g}(\boldsymbol{x})^2)\|_{L^2_{\boldsymbol{\lambda}}(\mathcal{X})}}{\|\pi(\boldsymbol{x})\|_{L^2_{\boldsymbol{\lambda}}(\mathcal{X})}} \,. \tag{2.47}$$

We can approximate the this integral as

$$\left(\frac{1}{N} \sum_{i=1}^{N} \left(\pi(\boldsymbol{x}^{(i)}) - (\gamma + \tilde{g}(\boldsymbol{x}^{(i)})^2) \right)^2 \boldsymbol{\lambda}(\boldsymbol{x}) \right)^{1/2} \approx \left(\int_{\mathcal{X}} \left(\pi(\boldsymbol{x}) - (\gamma + \tilde{g}(\boldsymbol{x})^2) \right)^2 \boldsymbol{\lambda}(x) \, \mathrm{d}\boldsymbol{x} \right)^{1/2} \tag{2.48}$$

and similarly $\int_{\mathcal{X}} \pi(\boldsymbol{x})^2 \boldsymbol{\lambda}(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x}$.

Absolute Error Bound

One way to assess the error between two distributions is to calculate the Wasserstein distance, because the Kantorovich-Rubinstein duality, as in [36, 37], says that the 1-Wasserstein distance is equal to the upper bound of differences in expectations of a function h between two probability distributions.

We define the 1-Wasserstein distance as

$$W_1(\pi, \tilde{\pi}) = \inf_{\nu \in \Pi(\pi, \tilde{\pi})} \int_{\mathcal{X} \times \mathcal{X}} c(\boldsymbol{x}, \tilde{\boldsymbol{x}}) \, \nu(\boldsymbol{x}, \tilde{\boldsymbol{x}}) d\boldsymbol{x} d\tilde{\boldsymbol{x}} \,, \tag{2.49}$$

where ν couples $\boldsymbol{x}$ and $\tilde{\boldsymbol{x}}$ so that the integral over the distance $c(\boldsymbol{x}, \tilde{\boldsymbol{x}})$ weighted by the probability measures π and $\tilde{\pi}$ is the greatest lower bound of all integrals with respect to a ν in the set of all couplings $\Pi(\pi, \tilde{\pi})$. Often ν is called a transport plan, where

$c(\boldsymbol{x}, \tilde{\boldsymbol{x}})$ is the (ground) cost function, and $\nu(\boldsymbol{x}, \tilde{\boldsymbol{x}})$ is related to the mass which has to be transported and the 1-Wasserstein distance is the earth mover distance. On the other hand (Kantorovich-Rubinstein duality), we can describe the 1-Wasserstein distance

$$W_1(\pi, \tilde{\pi}) = \sup_{\|h(\boldsymbol{x})-h(\tilde{\boldsymbol{x}})\|_{L^2} \leq \|\boldsymbol{x}-\tilde{\boldsymbol{x}}\|_{L^2}} \left\{ \int_{\mathcal{X}} h(\boldsymbol{x}) d\pi(\boldsymbol{x}) - \int_{\mathcal{X}} h(\tilde{\boldsymbol{x}}) d\tilde{\pi}(\tilde{\boldsymbol{x}}) \right\} \tag{2.50}$$

$$= \sup_{\|h(\boldsymbol{x})-h(\tilde{\boldsymbol{x}})\|_{L^2} \leq \|\boldsymbol{x}-\tilde{\boldsymbol{x}}\|_{L^2}} \left\{ \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\pi}[h(\boldsymbol{x})] - \mathop{\mathbb{E}}_{\tilde{\boldsymbol{x}}\sim\tilde{\pi}}[h(\tilde{\boldsymbol{x}})] \right\}. \tag{2.51}$$

as the lowest upper bound of differences in expecations of the 1-Lipschitz function h in between the two distributions π and $\tilde{\pi}$, with distance measure $c(\boldsymbol{x}, \tilde{\boldsymbol{x}}) = \|\boldsymbol{x} - \tilde{\boldsymbol{x}}\|_{L^2}$ for $\mathcal{X} \in \mathbb{R}^d$. For two sample sets $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\} \sim \pi$ and $\{\tilde{\boldsymbol{x}}^{(1)}, \ldots, \tilde{\boldsymbol{x}}^{(M)}\} \sim \tilde{\pi}$ the calculation of the Wasserstein distance becomes an optimisation problem, that is to find the best coupling of samples weighted by their distribution value according to an appropriate distance measure [38]. More specifically the 1-Wasserstein distance becomes

$$W_1(\pi, \tilde{\pi}) = \min_{\nu \in \Pi(\pi, \tilde{\pi})} \sum_{j=1}^{M} \sum_{i=1}^{N} \nu_{ij} \|\boldsymbol{x}^{(i)} - \tilde{\boldsymbol{x}}^{(j)}\|_{L^2}\,, \tag{2.52}$$

where the transport plan $\nu \in \mathbb{R}_{\geq 0}^{N \times M}$ defines the copling $\nu_{ij} \in \nu$ as $\nu_{ij} \coloneqq \pi(\boldsymbol{x}^{(i)})\tilde{\pi}(\tilde{\boldsymbol{x}}^{(j)})$ similar to [38, Eq. 3.166]. Additionally we require that $\sum_{i=1}^{N} \pi(\boldsymbol{x}^{(i)}) = \sum_{j=1}^{M} \tilde{\pi}(\tilde{\boldsymbol{x}}^{(j)}) = 1$. This gives us an upper bound of the absolute error in between the expected value of any 1-Lipschitz function h, e.g the upper bound of absolute differences in means related to the probaility measures π and $\tilde{\pi}$.

2.4 Affine Map

The forward map, which we introduce in Ch. 3, poses a weakly non-linear forward problem, which we could tackle by treating the problem as a linear problem and then iteratively updating the non-linear part after each parameter sample. Instead, we approximate the non-linear model using an affine map $\boldsymbol{M} : \boldsymbol{A}_L \to \boldsymbol{A}_{NL}$, which maps from the linear model to the non-linear model, so that we set $\boldsymbol{A} = \boldsymbol{M}\boldsymbol{A}_{NL} \approx \boldsymbol{A}_{NL}$. Here, we give a brief introduction to affine maps and present our approach to calculating the affine map deterministically. Alternatively, one can also determine this map using other methods, e.g. machine learning methods or matrix inversion.

An affine map is any linear map between two vector spaces or affine spaces, where an affine space does not need to preserve a zero origin, see [39, Def. 2.3.1]. In other words, an affine map does not need to map to the origin of the associated vector space or be a linear map on vector spaces, including a translation, or, in the words of my supervisor, C. F., an affine map is a Taylor series of first order. For more information on affine spaces and maps, we refer to the books [39, 40]

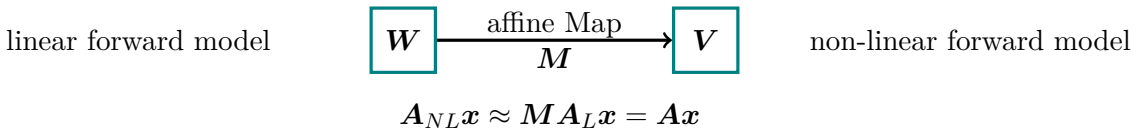


Figure 2.3: This Figure shows the schematic representation of the affine map $\boldsymbol{M}$, which approximates the non-linear forward model from the linear forward model. Here, $\boldsymbol{V}$ contains values produced by the linear forward model, and $\boldsymbol{W}$ contains the corresponding values from the non-linear forward model. Both $\boldsymbol{V}$ and $\boldsymbol{W}$ are affine subspaces over the same field. The affine map $\boldsymbol{M}$ projects elements from the linear forward model space $\boldsymbol{V}$ onto their counterparts in the non-linear forward model space $\boldsymbol{W}$.

2.5 Regularisation

As mentioned in the introduction, the currently most used method to analyse data in atmospheric physics is regularisation-based. Since we want to show that our methods are computationally comparable if not faster, and provide more information than regularisation, we choose a linear-Gaussian Bayesian framework closest to our regulariser, see section **??**.

The Tikhonov regularisation approach provides one solution $\boldsymbol{x}_\lambda$ that minimises both the data misfit norm

$$\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_{L^2} \tag{2.53}$$

and a regularisation semi-norm

$$\lambda \|\boldsymbol{T}\boldsymbol{x}\|_{L^2} \ , \tag{2.54}$$

for a given regularisation parameter $\lambda > 0$ as described in [14], with a linear forward model matrix $\boldsymbol{A}$, the data $\boldsymbol{y}$, a regularisation operator $\boldsymbol{T}$. The regularisation parameter weights the semi-norm and penalises $\boldsymbol{x}$ according to that. If λ is large, then the effect of the data on the solution $\boldsymbol{x}_\lambda$ is small or negligible. If λ is small, the solution $\boldsymbol{x}_\lambda$ will be dominated by the noisy data, resulting in an overfitted $\boldsymbol{x}_\lambda$. We refer to [41] and [7] for a more comprehensive analysis of the effects of the regularisation parameter on the solution, e.g. due to small singular values of the forward model.

For a fixed λ, the regularised solution

$$\boldsymbol{x}_\lambda = \arg\min_{\boldsymbol{x}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_{L^2}^2 + \lambda \|\boldsymbol{T}\boldsymbol{x}\|_{L^2}^2 \tag{2.55}$$

is obtained by taking the derivative with respect to $\boldsymbol{x}$:

$$\nabla_{\boldsymbol{x}} \left\{ (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x})^T (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}) + \lambda \boldsymbol{x}^T \boldsymbol{T}^T \boldsymbol{T}\boldsymbol{x} \right\} = 0 \tag{2.56}$$

$$\iff \quad \nabla_{\boldsymbol{x}} \left\{ \boldsymbol{y}^T\boldsymbol{y} + \boldsymbol{x}^T \boldsymbol{A}^T \boldsymbol{A}\boldsymbol{x} - 2\boldsymbol{y}^T \boldsymbol{A}\boldsymbol{x} + \lambda \boldsymbol{x}^T \boldsymbol{T}^T \boldsymbol{T}\boldsymbol{x} \right\} = 0 \tag{2.57}$$

$$\iff \quad 2\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{x} - 2\boldsymbol{A}^T\boldsymbol{y} + 2\lambda \boldsymbol{T}^T\boldsymbol{T}\boldsymbol{x} = 0, \tag{2.58}$$

also know as the "regularised normal equations" $\boldsymbol{A}^T\boldsymbol{y} = \boldsymbol{A}^T\boldsymbol{A}\boldsymbol{x} + \lambda\boldsymbol{T}^T\boldsymbol{T}\boldsymbol{x}$ [42]. Solving this equation yields the regularised solution

$$\boldsymbol{x}_\lambda = (\boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L})^{-1}\boldsymbol{A}^T\boldsymbol{y}\,, \tag{2.59}$$

where we define $\boldsymbol{L} \coloneqq \boldsymbol{T}^T\boldsymbol{T}$, which typically represents a discrete matrix approximation of a differential operator choice [7]. For example

$$\boldsymbol{T} = \frac{1}{h}\begin{bmatrix} -1 & 1 & & & \\ 0 & -1 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 0 & -1 & 1 \\ & & & 0 & -1 \end{bmatrix}, \tag{2.60}$$

is the first order derivative with equal spacing h as in [7] then

$$\boldsymbol{L} = \frac{1}{h^2}\begin{bmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix}, \tag{2.61}$$

is the second order derivative with Neumann boundary conditions, see [43].

In practice, $\boldsymbol{x}_\lambda$ is computed for a range of λ-values and evaluated based on the trade-off between the data misfit and the regularisation semi-norm. The optimal value of λ corresponds to the point of maximum curvature of the so-called L-curve [44], where the data misfit norm versus the regularisation semi-norm is plotted, see Fig. 4.13.

Additionally one can think about regularisation as a Lagrange multiplier $\mathcal{L}(\boldsymbol{x}, \lambda) :- \lambda\boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x} + \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_{L^2}$, which minimises $\boldsymbol{x}_\lambda = \arg\min_{\boldsymbol{x}}\boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x}$ with respect to a constant $\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_{L^2}$, see [14, fn. 6] and [45, Fig. 2.13]. So every solution $\boldsymbol{x}_\lambda$ is an extremum (the most regularised solution for a given data misfit norm) and almost every sample of the posterior, which represents a feasible solution given the data, has a higher $\boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x}$ value and lays above the L-Curve.

3
Forward Model

In this chapter, we present the forward model on which we apply all our methodology. We follow the Michelson Interferometer for Passive Atmospheric Sounding (MIPAS) handbook [46] and simulate data according to a cloud-free atmosphere in local thermodynamic equilibrium and assume a measurement instrument with infinite spectral resolution and no pointing errors. We do not include any other instrument specific details, such as sensor area or antenna response, as they are not available to us.

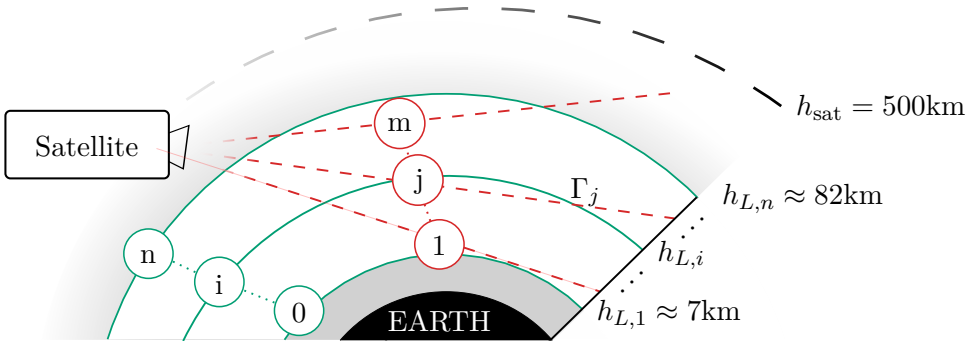


Figure 3.1: Schematic of measurement and analysis geometry, not to scale. The stationary satellite, at a constant height h_{sat} above Earth, takes $m = 30$ measurements along its line-of-sight defined by the line Γ_j. Each measurement has a limb height ℓ_j, $j = 1, 2, \ldots, m$ defined as the closest distance of Γ_j to the Earth's surface. Between $h_{L,0} \approx 7$km and $h_{L,n} \approx 82$km, the stratosphere is discretised into $n = 34$ layers as illustrated by the solid green lines.

A satellite at a constant height h_{sat} points through the atmosphere (limb-sounding) and measures thermal radiation of gas molecules along its straight line of sight Γ_j, see Figure 3.1. One measurement of the thermal radiation of one specific molecule, in our case ozone, denoted by the ozone volume mixing ratio (VMR) $x(r)$ at distance r from

the satellite, at the wave number ν, is given by the path integral

$$y_j = \int_{\Gamma_j} B(\nu, T) k(\nu, T) \frac{p(T)}{k_{\mathrm{B}} T(r)} x(r) \tau(r) \mathrm{d}r + \eta_j \tag{3.1}$$

$$\tau(r) = \exp\Big\{ - \int_{r_{\mathrm{obs}}}^{r} k(\nu, T) \frac{p(T)}{k_B T(r')} x(r') \mathrm{d}r' \Big\}, \tag{3.2}$$

which is the radiative transfer equation (RTE) [46]. For more information on the processes within the atmosphere for ozone, we refer to [47]. We define a tangent height h_{ℓ_j} and Γ_j for each $j = 1, 2, \ldots, m$, so that the data vector $\boldsymbol{y} \in \mathbb{R}^m$ including some noise η_j. Within the atmosphere, the number density $p(T)/(k_{\mathrm{B}} T(r))$ of molecules is dependent on the pressure $p(T)$, the temperature $T(r)$, and the Boltzmann constant k_{B}. The factor $\tau(r) \leq 1$ accounts for re-absorption of the radiation along the line-of-sight, which makes the RTE non-linear. The absorption constant

$$k(\nu, T) = L(\nu, T_{\mathrm{ref}}) \frac{Q(T_{\mathrm{ref}})}{Q(T)} \frac{\exp\{-c_2 E''/T\}}{\exp\{-c_2 E''/T_{\mathrm{ref}}\}} \frac{1 - \exp\{-c_2 \nu / T\}}{1 - \exp\{-c_2 \nu / T_{\mathrm{ref}}\}} \tag{3.3}$$

is dependent on the line intensity $L(\nu, T_{\mathrm{ref}})$ at reference temperature $T_{\mathrm{ref}} = 296K$, the lower-state energy of the transition E'', the second radiation constant $c2 = 1.4387769\mathrm{cmK}$ all provided by the HITRAN database [48]. Since we assume that the measurement device has negligible frequency window we neglect line broadening around ν_0 for the calculations of $L(\nu, T_{\mathrm{ref}})$, which would normally be modelled as a convolution of the normalised Lorentz profile (collisional/pressure broadening) and the normalised Doppler (thermal broadening) profile [46]. Additionally, since we target one specific molecule, we simplify the calculation of $k(\nu, T)$, which usually involves summing the individual absorption constants for each targeted molecule weighted by the respective volume mixing ratio [46]. The total internal partition function for the lower-state energy is given as:

$$Q(T) = g'' \exp\{-\frac{c_2 E''}{T}\}, \tag{3.4}$$

with the statistical weight g'' (also called the degeneracy factor) accounting for the molecule's non-rotational and rotational energy states, see [49]. Under the assumption of local thermodynamic equilibrium (LTE), the black body radiation acts as a source function

$$B(\nu, T) = \frac{2hc^2\nu^3}{\exp\{\frac{hc\nu}{k_B T}\} - 1}, \tag{3.5}$$

with Planck's constant h and speed of light c. For fundamentals on the Radiative transfer equation we recommend [50, Chapter 1], and for a more comprehensive model we refer to [51]

To enable matrix-vector multiplication, we discretise the atmosphere in n layers, where the i^{th} layer is defined by two spheres of radii $h_{L,i-1} < h_{L,i}$, for $i = 1, \ldots, n$, with $h_{L,0}$ and

$h_{L,n}$. Then we can discretise the ozone, pressure and temperature profiles as a function of height; in between the heights $h_{L,i-1}$ and $h_{L,i}$, each of the ozone concentration x_i, the pressure p_i, the temperature T_i, as well as all other height dependent parameters are assumed to be constant. Above $h_{L,n}$ and below $h_{L,0}$, the ozone concentration is set to zero, so no signal can be obtained. We rewrite the integral in Eq. (3.1) for one noise free measurement, using the trapezoidal rule, as a vector-vector multiplication $\boldsymbol{A_j}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T})$, where the ozone volume mixing ratio $\boldsymbol{x} = \{x_1, x_2, \ldots, x_n\} \in \mathbb{R}^n$. Note that since we aim to provide estimates over pressure $\boldsymbol{p}$ and temperature $\boldsymbol{T}$, we explicitly include them as parameters in our forward model. Here, the non-linear absorption $\tau(r)$ is another vector-vector multiplication and included in each entry of $\boldsymbol{A_j}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T}) \in \mathbb{R}^n$ which is the j-th row of the matrix $\boldsymbol{A}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T})$. Then given a noise vector $\boldsymbol{\eta} \in \mathbb{R}^m$ the data vector is given as:

$$\boldsymbol{y} = \boldsymbol{A}_{NL} + \boldsymbol{\eta}\,. \tag{3.6}$$

Here we define the non-linear forward model matrix as $\boldsymbol{A}_{NL} \coloneqq \boldsymbol{A}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T}) \in \mathbb{R}^{m \times n}$ for simplicity. Similarly we define $\boldsymbol{A}_L$, which denotes the linear forward model matrix and neglects absorption (e.g. set $\tau = 1$ in Eq. (3.2)). Then we can compute noise-free linear data as matrix-vector multiplication $\boldsymbol{A}_L\boldsymbol{x}$. Further, we classify the inverse problem as weakly non-linear, see e.g. Fig. 4.9, as neglecting the absorption changes the measurements only slightly.

3.1 Singular Value Decomposition of the Forward Model

Before simulating some data, we provide a quick and intuitive way of assessing if the data collection is effective, how much information is passed through the forward model, depending on how we measure and how the signal to noise ratio (SNR) affects that information. One way of doing this is via a singular value decomposition (SVD) of the forward model matrix

$$\boldsymbol{A}_L = \sum_{i=1}^{r} \boldsymbol{u}_i \sigma_i \boldsymbol{v}_i^T = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T \tag{3.7}$$

where $r = \min\{m, n\}$ for a forward model $\boldsymbol{A}_L \in \mathbb{R}^{m \times n}$. Consider noise-free measurements $\boldsymbol{A}_L\boldsymbol{x}$ for a satellite at a fixed height of $h_{\text{sat}} = 500\text{km}$ above sea level, where $\boldsymbol{x}$ is the ozone VMR. The SVD gives us information on how information is picked up from the parameter space by the forward model, described through the right singular values $\boldsymbol{v}_i$. The singular values σ_i, ordered in size from the largest σ_1 to the smallest σ_r, weight that information from the right singular values to the left singular values $\boldsymbol{u}_i$, which project onto the data space. For a large singular value, we can say that the forward model is informative about structures in the corresponding right singular vector and vice versa. Note that we obtain

the same results using the non-linear forward model $\boldsymbol{A}(\boldsymbol{x})$ to do this analysis, where we would rewrite to the matrix-vector multiplication $\boldsymbol{A}_{NL}\boldsymbol{x}$, where $\boldsymbol{A}_{NL}$ is depend on $\boldsymbol{x}$.

Further, for very small singular values $\sigma_i \ll \sigma_1/\text{SNR}$ below the RMS noise level or the noise standard deviation (STD), we can introduce an effective rank $r_{\text{eff}} \leq r$. Then information of parameter space spanned by $\{\boldsymbol{v}_{r_{\text{eff}}+1}, \ldots, \boldsymbol{v}_r\}$ is not passed through the forward model and the data is noise dominated in the corresponding data space, see Figure 3.6. This is based on the rough assumption that if we define the SNR as

$$\text{SNR} := \frac{\max(y)}{\text{STD noise}} = \frac{\text{peak signal}}{\text{RMS noise}} \text{[52]} \tag{3.8}$$

then the maximum singular value $\max(y) \approx \sigma_1$ and the information transmitted through the forward model corresponds roughly to the singular values $\sigma_i \gtrsim \max(y)/\text{SNR}$. See [7] for a more comprehensive analysis.

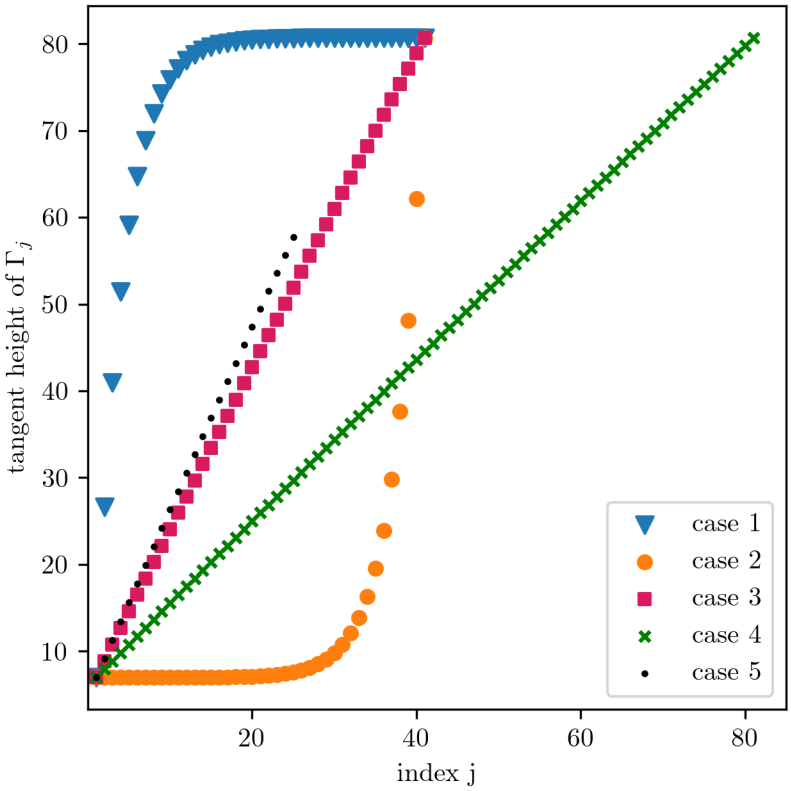


Figure 3.2: We plot the tangent heights for different cases of measurements.

Next, we plot the singular values for 5 different measurement scenarios, where we either measure at equidistance spaced tangent heights or collect more data from high signal regions at low altitudes, to see which of the tested cases is most effective. We assess the number of singular values above and below a certain SNR visually. Our objective is to measure ozone $\boldsymbol{x}$ so our forward model $\boldsymbol{A}$ includes temperature and pressure, the

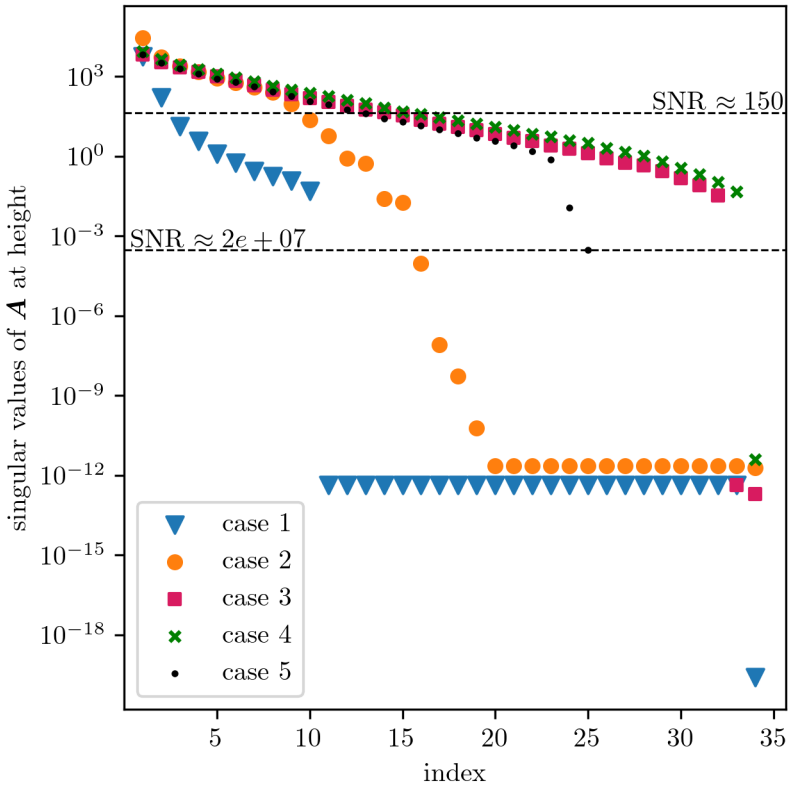


Figure 3.3: We plot the singular values of the forward model matrix for different sequences of measurements. The corresponding tangent heights of the test cases are plotted in Fig. 3.2. The dotted vertical line marks where the singular values are dominated by noise according to a SNR.

latter is dominant, see Fig. 4.17, and decreases exponentially in height and hence does affect the information passed through the model. If the pressure is high, the noise is low, and if the pressure is low, the data is noise-dominated. We start with case 3 in Fig. 3.2 where measurements are spaced according to a pointing accuracy of 150arc sec, given to us by the team of the University of New South Wales Canberra Space [53]. The pointing accuracy determines how well the satellite can point in a certain direction and, hence, roughly the spacing in between two measurements. The corresponding singular values are plotted in Fig. 3.3, of which the first 25 decrease linearly in log-space and about 10-15 singular values lie above the SNR. In comparison, if we measure a lot of times in regions where the data is noise-dominated (high altitude), case 1, we do obtain more information since the singular values decrease rapidly. Measuring lots of times at low altitudes, where the data is informative, and less at higher altitudes, case 2, does not seem optimal either, as we observe one larger singular value, but the other singular values decrease faster compared to case 3. Now, if we double the number of measurements compared to case 3, see case 4, we do get slightly larger singular values, but not significantly so that it would be worth the engineering effort required to achieve that. The measurements with

equidistance-spaced tangent height seem to be most informative. By exploratory analysis, we find that we can tolerate a slightly larger distance between tangent heights (pointing accuracy of 175arc sec) than required by [53], see case 5. In that case, we also stop measuring when the signal is too noisy and decrease the number of measurements taken without losing information. We note that if one wanted to obtain all information provided by the forward model, we would need a signal-to-noise ratio of roughly 10^7.

In principle, we show that it does not depend on how one measures, one can not get more information by measuring more in regions where the information content is low or high. This contradicts the current measurement setup on the AURA MLS [8], which reports high noise in lower atmospheric regions, due to thermal radiation from the earth, and measures more in those regions.

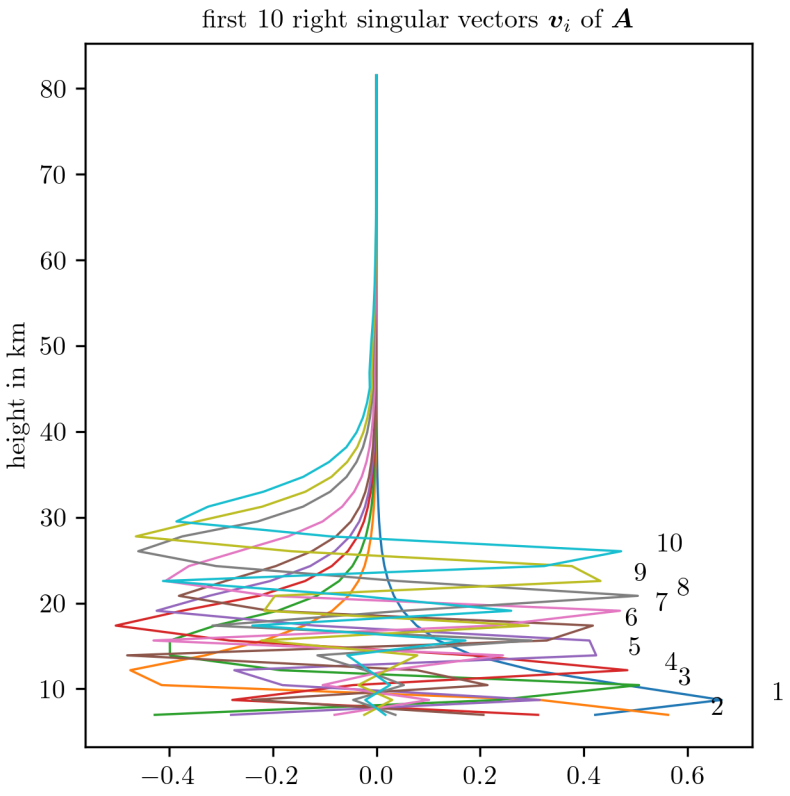


Figure 3.4: We plot the first 10 right singular vectors of the forward model matrix for case 5 sequence of measurements, see Fig. 3.3. These singular vectors correspond to high singular values of the forward model, see Fig. 3.3.

Consequently, we proceed with case 5 and plot the right singular vectors of the forward model versus height in the atmosphere to see where our model is most informative, or which structures of the parameter space are picked up by the model. The first 10 right singular vectors, in Fig. 3.4, corresponding to the 10 largest singular values, pick up structures in lower atmospheric regions. So we can assume that, given some data, we will

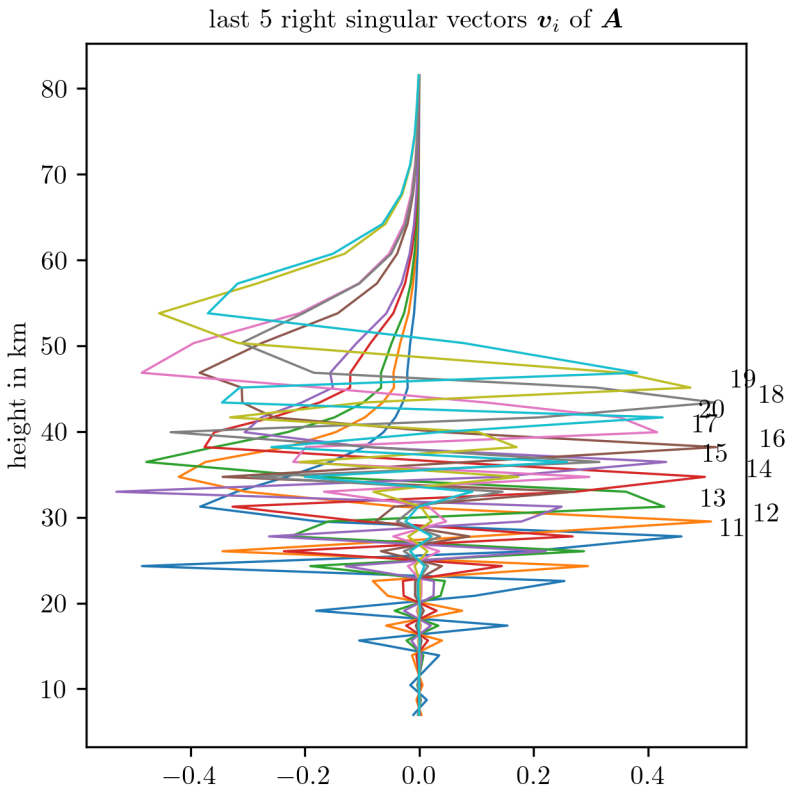


Figure 3.5: We plot the right singular vectors with index $i = 11, \ldots, 19$ of the forward model matrix for case 5 sequence of measurements, see Fig. 3.3. These singular vectors correspond to singular values around the noise level of the measurement, see Fig. 3.3.

be able to provide good reconstructions of the parameter in lower altitudes. Next, we plot the right singular vectors corresponding to the singular values σ_j for $j = 11, \ldots, 20$ in Fig. 3.5, where the nose starts to dominate the data. These singular values lie in regions around the SNR, see Fig. 3.3, and pick up values in the middle of the atmosphere. Consequently, we expect a higher uncertainty of reconstructed parameter values in the middle atmospheric regions. The singular vectors corresponding to the last 5 singular values pick up structures in higher altitudes, but since the singular values are very small, we will not be able to retrieve those structures. More specifically, the retrieved parameter values at higher altitudes will be fully determined by the prior or, in the case of regularisation, by the regulariser [7].

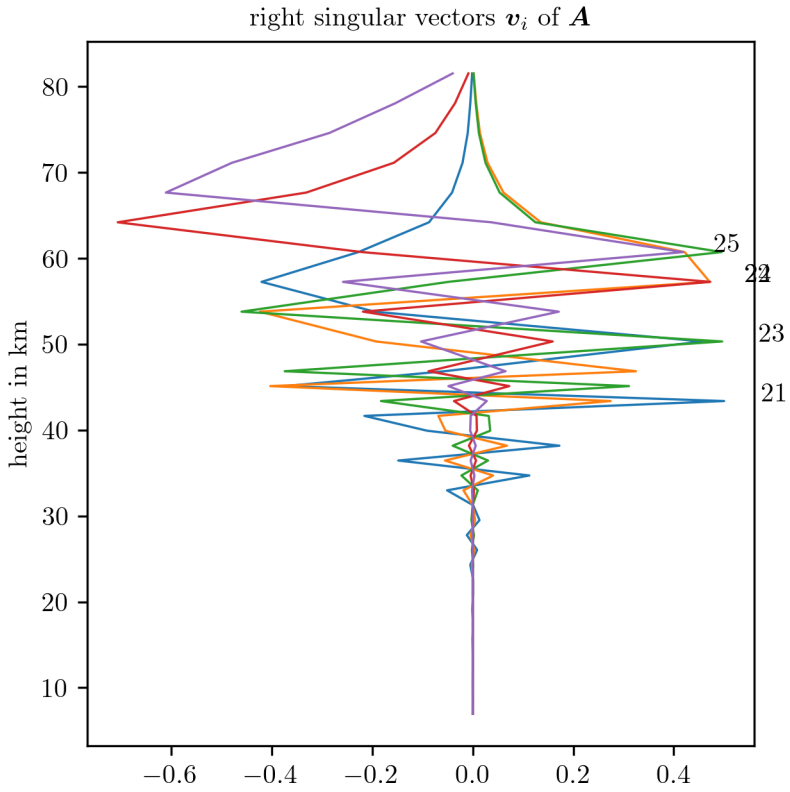


Figure 3.6: We plot the last 5 right singular vectors of the forward model matrix for the case 5 sequence of measurements, as displayed in Fig. 3.2. These singular vectors correspond to small singular values of the forward model, see Fig. 3.3.

4
Results and Conclusions

In this chapter, we use the forward model to generate some data given an underlying ground truth and then guide the reader through the process of setting up a Bayesian framework and ultimately obtaining the posterior distributions of parameters of interest, such as ozone concentration or pressure and temperature profiles. We are using a directed acyclic graph (DAG) to visualise hierarchical and correlation structures of a Bayesian model. Next, we established a choice of prior distributions within our Bayesian model and formulated the posterior distributions. Based on the linear forward model $\boldsymbol{A}_L$, we characterise the marginal posterior for ozone and compare that to the TT approximation. Then we calculate the mean and the covariance matrix of the full posterior for ozone, which we use to find affine map to approximate the non-linear forward model $\boldsymbol{A}_{NL} \approx \boldsymbol{M}\boldsymbol{A}_{NL}$ (see Sec. 4.3). In Sec. 4.4, we repeat the MTC scheme to provide a posterior distribution of ozone based on the approximate forward model and compare to a regularisation approach and a ground truth. Additionally, in sec 4.5, we extend the hierarchical Bayesian model and MTC scheme to jointly provide conditional posterior distributions of ozone, temperature and pressure. Here we elaborate on some aspects of prior modelling and on our findings when using a TT to approximate the higher dimensional marginal posterior. Lastly, we evaluate some errors occurring during the process. All of programming and analysis is done in Python and the reported computation times correspond to a MacBook Pro from 2019 with a 2.4 GHz quad-core Intel Core i5 processor.

4.1 Simulate Data based on a Ground Truth

We take a ground truth ozone VMR at distinct pressure values generated from some data [9] of the MLS on the Aura satellite within the Antarctic region and with a peak in high altitude, see Fig. 4.6, which seems to be a typical nighttime profile [47].

We target Ozone at a frequency of 235.71 GHz, which lies within the region where the MLS observers ozone [10, 54]. The corresponding wave number is $\nu = 7.86\text{cm}^{-1}$. We recursively calculate the geometric height with the hydrostatic equilibrium equation

$$\frac{\mathrm{d}p}{p} = \frac{-gM}{R^*T}\mathrm{d}h\,, \tag{4.1}$$

with the acceleration due to gravity

$$g = g_0\left(\frac{r_0}{r_0+h}\right), \tag{4.2}$$

where the polar radius pf the earth is $r_0 \approx 6356\,\text{km}$, the gravitation at sea level is $g_0 \approx 9.81\text{m/s}^2$, $R^* = 8.31432 \times 10^{-3}\text{Nm/kmol/K}$ and the mean molecular weight of the air is $M = 28.97\text{kg/kmol}$ [55]. This holds up to a geometric height of 86km, where we ignore a 0.04% non-linear change in M from 80km to 86km in geometric altitude.

Following [55] we form the temperature function

$$T(h) = \begin{cases} T_0 & , h = 0 \\ T_0 + a_0 h & , 0 \le h < h_{T,1} \\ T_0 + a_0 h_{T,1} & , h_{T,1} \le h < h_{T,2} \\ T_0 + a_0 h_{T,1} + a_1(h_{T,2} - h_{T,1}) + a_2(h - h_{T,2}) & , h_{T,2} \le h < h_{T,3} \\ T_0 + a_0 h_{T,1} + a_1(h_{T,2} - h_{T,1}) & \\ \quad + a_2(h_{T,3} - h_{T,2}) + a_3(h - h_{T,3}) & , h_{T,3} \le h < h_{T,4} \\ T_0 + a_0 h_{T,1} + a_1(h_{T,2} - h_{T,1}) & \\ \quad + a_2(h_{T,3} - h_{T,2}) + a_3(h_{T,4} - h_{T,3}) + a_4(h - h_{T,4}) & , h_{T,4} \le h < h_{T,5} \\ T_0 + a_0 h_{T,1} + a_1(h_{T,2} - h_{T,1}) & \\ \quad + a_2(h_{T,3} - h_{T,2}) + a_3(h_{T,4} - h_{T,3}) + a_4(h_{T,5} - h_{T,4}) & \\ \quad + a_5(h - h_{T,5}) & , h_{T,5} \le h < h_{T,6} \\ T_0 + a_0 h_{T,1} + a_1(h_{T,2} - h_{T,1}) & \\ \quad + a_2(h_{T,3} - h_{T,2}) + a_3(h_{T,4} - h_{T,3}) + a_4(h_{T,5} - h_{T,4}) & \\ \quad + a_5(h_{T,6} - h_{T,5}) + a_6(h - h_{T,6}) & , h_{T,6} \le h \lesssim 86 \end{cases} \tag{4.3}$$

with gradient and height values provided by [55], see Tab. 4.1. This acts as the ground truth temperature profile, see Fig. 4.15.

subscript i	geometric height $h_{T,i}$ in km	gradient a_i
0	0	-6.5
1	11	0
2	20.1	1
3	32.2	2.8
4	47.4	0
5	51.4	-2.8
6	71.8	-2

Table 4.1: Definition of height depending temperature gradients.

Then we can compute a data vector $\boldsymbol{y} = \boldsymbol{A}_{NL} + \boldsymbol{\eta}$, with $m = 30$ measurements according to the radiative transfer equation (RTE), see Eq. 3.1 which we solve using the trapezoidal integration rule, determined by the satellite pointing accuracy of 175arc sec, see Fig. 3.2. We assume an atmosphere between $h_{L,1} = 7$km and $h_{L,n} = 83.3$km with $n = 45$ equidistant layers. The height value $h_{L,i}$ for each layer $i = 1, \ldots, n$ is defined by the pressure values from [9] and the hydrostatic equilibrium equation, see Eq. 4.1. We calculate the absorption constant $k(\nu, T)$ as in Eq. 3.2, following the *HITRAN* database [48], which provides the line intensity $L(\nu, T_{\text{ref}})$ for the isotopologue $^{16}O_3$ with the AFGL Code 666. This gives us a non-linear forward model matrix $\boldsymbol{A}_{NL} = \boldsymbol{A}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T}) \in \mathbb{R}^{m \times n}$, where $\boldsymbol{x} \in \mathbb{R}^n$ is vector related to the ozone VMR, $\boldsymbol{p} \in \mathbb{R}^n$ is the vector describing the pressure values and $\boldsymbol{T} \in \mathbb{R}^n$ the temperature values.

Lastly we add normally distributed $\boldsymbol{\eta} \sim \mathcal{N}(0, \gamma^{-1}\boldsymbol{I})$ noise so that the SNR is 150, see Eq. 3.8, similar to [11], where a signal with a maximal spectral intensity of around 100K and a noise range of 0.4 to 1.6K is reported. We note that the methods used in this thesis will work with different SNRs or other frequencies.

When we plot the data in Fig. 4.1, we see that, as mentioned in Section 3.1, the data is noise-dominated in higher altitudes. Now, given the data, we like to determine the posterior distributions over ozone $\boldsymbol{x}$, pressure $\boldsymbol{p}$ and temperature $\boldsymbol{T}$ at different heights.

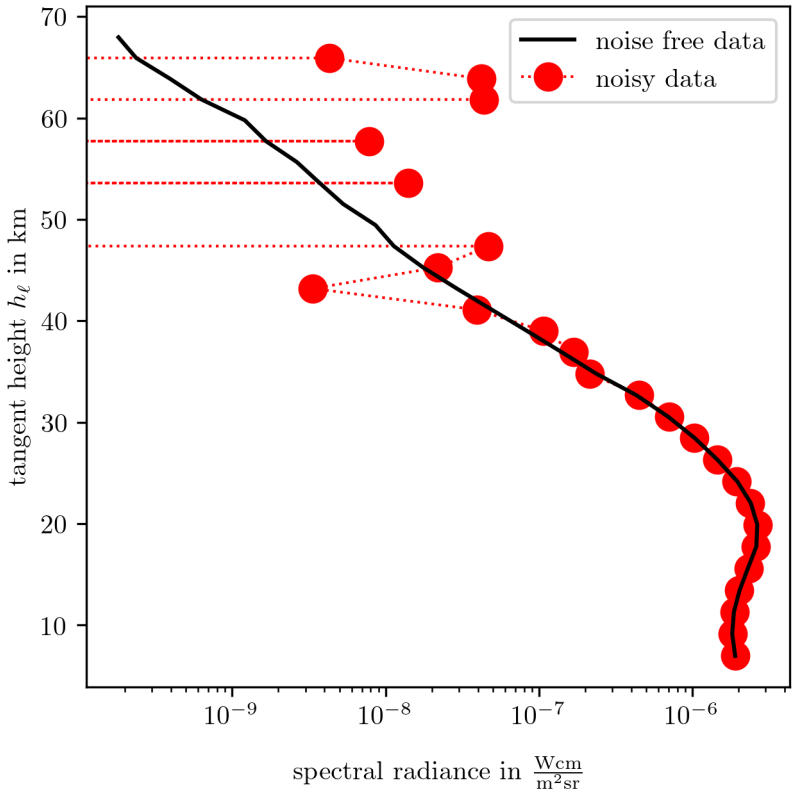


Figure 4.1: Logarithmic plot of data points at different tangent height. Note that negative values are not appearing, and we see that the noise is dominating at high altitudes.

4.2 Hierarchical Bayesian Framework for Ozone

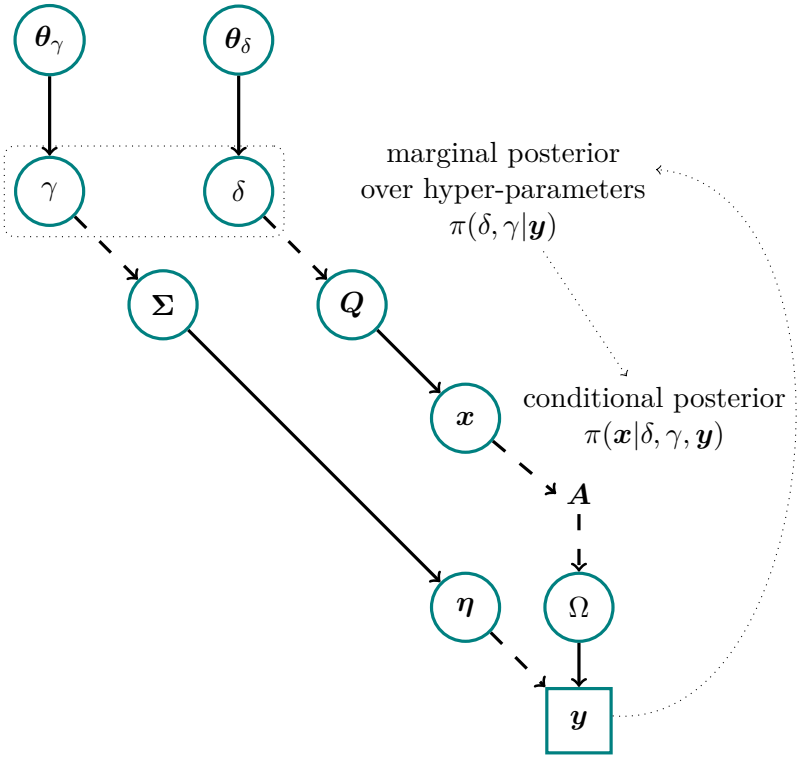


Figure 4.2: DAG for visualisation of hierarchical modelling and measuring process of ozone including the MTC scheme. The hyper-parameter γ deterministically (dotted line) sets the noise covariance $\boldsymbol{\Sigma} = \gamma^{-1}\boldsymbol{I}$ and hence the random (solid line) noise vector $\boldsymbol{\eta} \sim \mathcal{N}(0, \gamma^{-1}\boldsymbol{I})$. The hyper-parameter δ determines (dotted line) the prior precision matrix $\boldsymbol{Q} = \delta\boldsymbol{L}$ for the normally distributed (solid line) prior $\boldsymbol{x}|\delta \sim \mathcal{N}(0, \delta\boldsymbol{L})$, where $\boldsymbol{L}$ is a graph Laplacian, see Eq. 4.5. The hyper-prior distributions (solid line) $\pi(\delta, \gamma)$ are defined by $\boldsymbol{\theta}_\gamma$ and $\boldsymbol{\theta}_\delta$. Through the linear forward model $\boldsymbol{A}$ we generate a space of all measurable noise free data $\boldsymbol{Ax}$ from which we randomly observe a data set $\boldsymbol{y}$ including some added noise $\boldsymbol{\eta}$. Within the MTC scheme we evaluate the marginal posterior over the hyper-parameters $\pi(\gamma, \delta|\boldsymbol{y})$ first and then the conditional posterior $\pi(\boldsymbol{x}|\delta, \gamma, \boldsymbol{y})$. This breaks the correlation structure of $\boldsymbol{x}$ and δ and γ, and allows to evaluate the marginal posterior independent of $\boldsymbol{x}$.

In this section, we setup the hierarchically-ordered linear-Gaussian Bayesian framework to determine the ozone posterior distribution, conditioned on ground truth temperature and pressure. Where for now we define the forward model matrix $\boldsymbol{A} = \boldsymbol{A}_L$ and define the distributions of that Bayesian model, similarly to a regularisation approach, as:

$$\boldsymbol{y}|\boldsymbol{x}, \gamma, \delta \sim \mathcal{N}(\boldsymbol{A}\,\boldsymbol{x}, \gamma^{-1}\boldsymbol{I}) \tag{4.4a}$$

$$\boldsymbol{x}|\delta \sim \mathcal{N}(\boldsymbol{0}, (\delta\boldsymbol{L})^{-1}) \tag{4.4b}$$

$$\delta \sim \Gamma(\alpha_\delta = 1, \beta_\delta = 10^{-35}) \tag{4.4c}$$

$$\gamma \sim \Gamma(\alpha_\gamma = 1, \beta_\gamma = 10^{-35})\,. \tag{4.4d}$$

Assuming Gaussian noise $\boldsymbol{\eta} \sim \mathcal{N}(0, \gamma^{-1}\boldsymbol{I})$, the likelihood function is a normal distributions with mean $\boldsymbol{Ax}$ and covariance matrix $\gamma^{-1}\boldsymbol{I}$. We define the normal prior-distribution

$\pi(\boldsymbol{x}|\delta)$, with zero mean and precision matrix $\delta\boldsymbol{L}$, where δ is a smoothness hyper-parameter and $\boldsymbol{L}$ is the second order discrete derivate operator (see Eq. 4.5). Here the hyper-prior distributions $\pi(\delta)$ and $\pi(\gamma)$ are gamma distributions with shape α and rate β.

We can visualise this hierarchical structure and the correlations in between different hyper-parameters and parameters through a DAG. The hyper-parameter γ sets the noise covariance deterministically (dotted line), but is it self statistically (solid line) defined by the hyper-prior distribution $\pi(\gamma)$. In this case θ_γ determines the hyper-prior distribution $\pi(\gamma)$, and similarly θ_δ for $\pi(\delta)$, which then deterministically sets the prior precision $\boldsymbol{Q}$. In our case, δ accounts for smoothness of the ozone profile. Then $\boldsymbol{Ax}$ determines the space of all measurable noise-free data sets Ω, through the linear forward model, from which we observe a data set $\boldsymbol{y}$ including some noise $\boldsymbol{\eta}$. From that data we then "reverse the arrows" to determine the posterior distribution over the parameter $\boldsymbol{x}$. Since noise is a random process with a defined distribution, the posterior distribution $\pi(\boldsymbol{x}|\boldsymbol{y})$ is well defined. Usually, due to underlying correlation structures, evaluating this posterior poses a signifiant challenge, here the MTC scheme provides the marginal posterior $\pi(\delta, \gamma|\boldsymbol{y})$ first and then the conditional posterior $\pi(\boldsymbol{x}|\delta, \gamma, \boldsymbol{y})$.

4.2.1 Prior Modelling

To complete the Bayesian framework, we have to define prior distributions over the hyper-parameters and parameters. Ideally, we define the prior distributions as uninformative as possible, and include functional dependencies and physical properties.

By choosing a normally distributed prior $\pi(\boldsymbol{x}|\delta)$ with zero mean and no other restrictions, we can already see that our model is not taking into account that ozone values can not be negative. As already mentioned, we set the precision matrix of that prior distribution to

$$\delta\boldsymbol{L} = \delta \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix} \tag{4.5}$$

which is a 1-dimensional Graph Laplacian as in [14, 43] with Dirichlet boundary condition. This matrix will also act as the regulariser later in the Regularisation section, see Sec. 2.5. We reduce the dimension of $\boldsymbol{x}$ from 45 to 34 by discarding every second ozone VMR over a height of $\approx$ 47km. Doing that, while not changing $\boldsymbol{L}$, we effectively induce a larger correlation between points at higher altitude. We plot the corresponding prior ozone profiles according to $\boldsymbol{x} \sim \mathcal{N}(0, (\delta\boldsymbol{L})^{-1})$ in Fig. B.1.

For δ and γ we pick relatively uninformative gamma distributions so that $\gamma \sim \mathcal{T}(\boldsymbol{\theta_\gamma}) \propto \gamma^{\alpha_\gamma - 1} \exp(-\beta_\gamma \gamma)$ and $\delta \sim \mathcal{T}(\boldsymbol{\theta_\delta})$, where $\boldsymbol{\theta_\gamma} = \{\alpha_\gamma, \beta_\gamma\} = \{\alpha_\delta, \beta_\delta\} = \boldsymbol{\theta_\delta} = (1, 10^{-35})$, see Fig. 4.10, similar to [14]. Those gamma distributions have another advantage when using a MWG algorithm to sample from the marginal posterior distribution $\pi(\delta, \gamma|\boldsymbol{y})$. In doing so, we introduce the regularisation parameter $\lambda = \delta/\gamma$ so that $\pi(\gamma|\lambda, \boldsymbol{y}) \sim \mathcal{T}(\cdot)$ is a gamma distribution and easy to sample from.

4.2.2 Posterior Distribution – linear Model

As explained in Sec. 2.1 we factorise the posterior

$$\pi(\boldsymbol{x}, \delta, \gamma|\boldsymbol{y}) \propto \pi(\boldsymbol{y}|\boldsymbol{x}, \delta, \gamma)\pi(\boldsymbol{x}, \delta, \gamma) \tag{4.6}$$

into

$$\pi(\boldsymbol{x}, \delta, \gamma|\boldsymbol{y}) = \pi(\boldsymbol{x}|\delta, \gamma, \boldsymbol{y})\pi(\delta, \gamma|\boldsymbol{y}) \tag{4.7}$$

the marginal posterior $\pi(\delta, \gamma|\boldsymbol{y})$ and conditional posterior $\pi(\boldsymbol{x}|\delta, \gamma, \boldsymbol{y})$ (see Eq. 2.5). As discussed in sec. 2.1, for the linear-Gaussian case, $\boldsymbol{x}$ cancels in the marginal posterior over the hyper-parameters. Following the MTC scheme, we characterise the marginal posterior first and *then* the conditional posterior of $\pi(\boldsymbol{x}|\delta, \gamma)$.

Marginal Posterior

Consequently, for the hierarchical model specified in Eq. 4.4, the marginal posterior distribution over the hyper-parameters is given by

$$\pi(\lambda, \gamma|\boldsymbol{y}) \propto \lambda^{n/2+\alpha_\delta-1}\gamma^{m/2+\alpha_\delta+\alpha_\gamma-1} \exp\Big\{-\frac{1}{2}g(\lambda) - \frac{\gamma}{2}f(\lambda) - \beta_\delta\lambda\gamma - \beta_\gamma\gamma\Big\}, \tag{4.8}$$

with the introduced regularisation parameter $\lambda = \delta/\gamma$, and

$$f(\lambda) = \boldsymbol{y}^T\boldsymbol{y} - (\boldsymbol{A}^T\boldsymbol{y})^T(\boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L})^{-1}(\boldsymbol{A}^T\boldsymbol{y})\,, \tag{4.9a}$$

$$\text{and } g(\lambda) = \log\det(\boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L})\,. \tag{4.9b}$$

Note that, when changing variables from $\delta = \lambda\gamma$ to λ the hyper-prior distribution changes to $\pi(\lambda) \propto \lambda^{\alpha_\delta-1}\gamma^{\alpha_\delta}\exp(-\beta_\delta\lambda\gamma)$, due to $\mathrm{d}\delta/\mathrm{d}\lambda = \gamma$. Most of the computational effort, for each function evaluation of the marginal posterior in Eq. 4.8, lies in the calculation of $f(\lambda)$ in Eq. 4.30a and $g(\lambda)$ in Eq. 4.30b. In Fig. 4.3 we see that $f(\lambda)$ and $g(\lambda)$ are well behaved within the region of interest and approximate $f(\lambda) \approx \tilde{f}(\lambda)$ with a 3rd order Taylor series around the mode λ_0 of $\pi(\lambda, \gamma|\boldsymbol{y})$. We also note that $\tilde{g}(\lambda) \approx g(\lambda)$ behaves linearly around λ_0 in the log-space. The approximations are implicitly given by

$$f^{(r)}(\lambda_0) = (-1)^{r+1}(\boldsymbol{A}^T\boldsymbol{y})^T(\boldsymbol{B}_0^{-1}\boldsymbol{L})^r\boldsymbol{B}_0^{-1}\boldsymbol{A}_L^T\boldsymbol{y} \tag{4.10}$$

$$\text{and } \log\tilde{g}(\lambda) = (\log\lambda - \log\lambda_0)\frac{\log g(\lambda_{\max}) - \log g(\lambda_0)}{\log\lambda_{\max} - \log\lambda_0} + \log g(\lambda_0) \tag{4.11}$$

with $\boldsymbol{B}_0 = \boldsymbol{A}^T\boldsymbol{A} + \lambda_0\boldsymbol{L}$. We plot the approximations

$$\tilde{f}(\lambda) = \sum_{r=0}^{3} f^{(r)}(\lambda_0)(\lambda - \lambda_0)^r\,, \tag{4.12a}$$

$$\text{and } \tilde{g}(\lambda) = \exp\log\tilde{g}(\lambda)\,, \tag{4.12b}$$

in Fig. 4.3 and elaborate on the approximation errors in Sec 4.2.2. Note that usually a Taylor series includes a factor $(r!)^{-1}$, in this case it cancels in $f^{(r)}(\lambda_0)$, see [14].

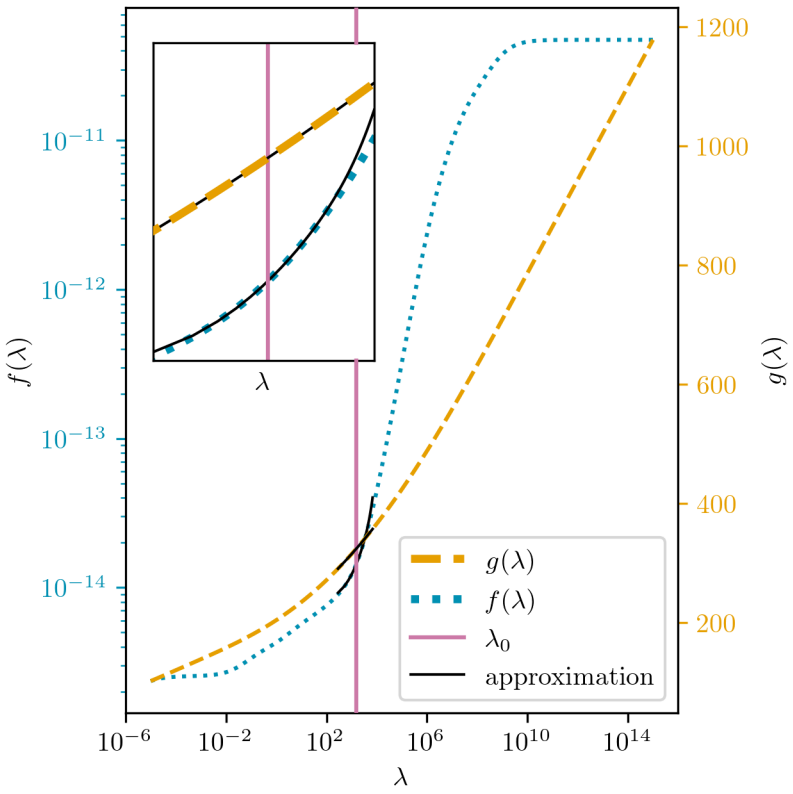


Figure 4.3: Plot of the functions $f(\lambda)$ and $g(\lambda)$ from the marginal posterior for a wide range of $\lambda = \delta/\gamma$. We plot the third-order Taylor series in black around the mode of the marginal posterior (vertical line) for the sampling range of λ within the MWG algorithm.

Error due to Approximation of f and g We report an maximum approximation error in between f and g For 2nd order taylor 3rd tayle and 4th order taylor

then we neglect ther erro in g beacus ethabsolute in f and g are and propagtion error in marginal is due to f is realtive RMS and the maximum error is ...

When approximating the functions $f(\lambda)$ and $g(\lambda)$, we find that the 3rd-order Taylor series of $f(\lambda)$ and a linear approximation of $g(\lambda)$ in log-space give the smallest error. The Taylor series truncation error of $f(\lambda)$ is bounded by the fourth order Taylor series $E_f = \arg\max_{\lambda} f^{(4)}(\lambda_0)/4!\,(\lambda - \lambda_0)^4$ and corresponds to an relative error bounded by 20%. Since the maximum absolute error of the approximation $\arg\max_{\lambda} |\tilde{g}(\lambda) - g(\lambda)| \approx 1$ corresponds to an relative error of approximately 0.3% and is small compared to $E_f \approx 1e8$ we ignore the approximation error of $g(\lambda)$. Then the maximum relative propagation error $\arg\max_{\lambda,\gamma} 0.5\gamma E_f / \log \pi(\lambda, \gamma|\boldsymbol{y})$ is bound by approximately 5%.

Sample from Marginal Posterior Using these approximation we can either utilise a TT approximation of the marginal posterior, see Sec. 2.3.1, over a predefined grid and calculate the marginals $\pi(\gamma|\boldsymbol{y})$ and $\pi(\lambda|\boldsymbol{y})$, or employ a Metropolis within Gibbs (MWG)

sampler to sample from $\pi(\lambda, \gamma|\boldsymbol{y})$, see sec. 2.2.1. More specifically, we implement a Metropolis random walk on

$$\pi(\lambda|\gamma, \boldsymbol{y}) \propto \lambda^{n/2+\alpha_\delta-1} \exp\left\{-\frac{1}{2}g(\lambda) - \frac{\gamma}{2}f(\lambda) - \beta_\delta\gamma\lambda\right\}. \tag{4.13}$$

We accept or reject a proposal $\lambda' \sim \mathcal{N}(0, \sigma_\lambda)$ according to the acceptance ratio in log space

$$\log\left\{\frac{\pi(\lambda'|\gamma^{(k)}, \boldsymbol{y})}{\pi(\lambda^{(k)}|\gamma^{(k)}, \boldsymbol{y})}\right\} = \log\{\pi(\lambda'|\gamma^{(k)}, \boldsymbol{y})\} - \log\{\pi(\lambda^{(k)}|\gamma^{(k)}, \boldsymbol{y})\} \tag{4.14}$$

$$= \frac{n}{2}(\log\{\lambda'\} - \log\{\lambda^{(k)}\}) + \frac{1}{2}\Delta g + \frac{\gamma^{(k)}}{2}\Delta f + \beta_\delta\gamma^{(k)}\Delta\lambda\,, \tag{4.15}$$

where $\Delta\lambda = \lambda' - \lambda^{(k)}$ and $\Delta f \approx \tilde{f}(\lambda') - \tilde{f}(\lambda^{(k)}) = \sum_{r=1}^{3} f^{(r)}(\lambda_0)(\Delta\lambda' - \Delta\lambda^{(k)})^r$, with $\Delta\lambda' = \lambda' - \lambda_0$ and $\Delta\lambda^{(k)} = \lambda^{(k)} - \lambda_0$. Similarly we approximate $\Delta g \approx \tilde{g}(\lambda') - \tilde{g}(\lambda^{(k)})$. Lastly, we do a Gibbs step on

$$\gamma^{(k+1)}|\lambda^{(k+1)}, \boldsymbol{y} \sim \Gamma\left(\frac{m}{2} + \alpha_\delta + \alpha_\gamma, \frac{1}{2}f(\lambda^{(k+1)}) + \beta_\gamma + \beta_\delta\lambda^{(k+1)}\right) \tag{4.16}$$

to generate marginal posterior samples $(\lambda, \gamma)^{(1)}, \ldots, (\lambda, \gamma)^{(N)} \sim \pi(\lambda, \gamma|\boldsymbol{y})$.

The mode (λ_0, γ_0) of $\pi(\lambda, \gamma|\boldsymbol{y})$ is provided by the `scipy.optimize.fmin` function and we approximate $f(\lambda)$ and $g(\lambda)$ accordingly. In doing so, we compute the vector $\boldsymbol{B}_0^{-1}\boldsymbol{A}^T\boldsymbol{y} = (\boldsymbol{A}^T\boldsymbol{A} + \lambda_0\boldsymbol{L})\boldsymbol{A}^T\boldsymbol{y}$, the matrix $\boldsymbol{B}_0^{-1}\boldsymbol{L}$ and the determinant in $g(\lambda)$ using Cholesky decomposition. Then we approximate $f(\lambda)$ with a 3rd order Taylor series and $g(\lambda)$ with a linear approximation in the log-space, where for the approximation in $g(\lambda)$ we set $\lambda_{\max}$ to the maximum value of λ on the TT-grid (see next Paragraph). To sample from $\pi(\lambda, \gamma|\boldsymbol{y})$ we employ the MWG algorithm, see Alg. Box 1, initialised at the mode $(\lambda^{(0)}, \gamma^{(0)}) = (\lambda_0, \gamma_0)$ and take $N = 10000$ plus $N_{\text{burn-in}} = 100$ steps in approximately 0.3s. The standard deviation of the normal proposal distribution is set to $\sigma_\lambda = 0.8\lambda_0$ so that the acceptance rate is ≈ 0.5 as suggested in [56]. The samples are plotted in Fig. 4.5 as a 2D scatter plot, as well as the trace of the MWG to show ergodicity. We calculate the integrated autocorrelation time (IACT) with the Python implementation of [23], provided by [57], which gives us $\tau_{\text{int},\gamma} =$ and $\tau_{\text{int},\delta} =$, see Fig. 4.4 and Fig. B.2.

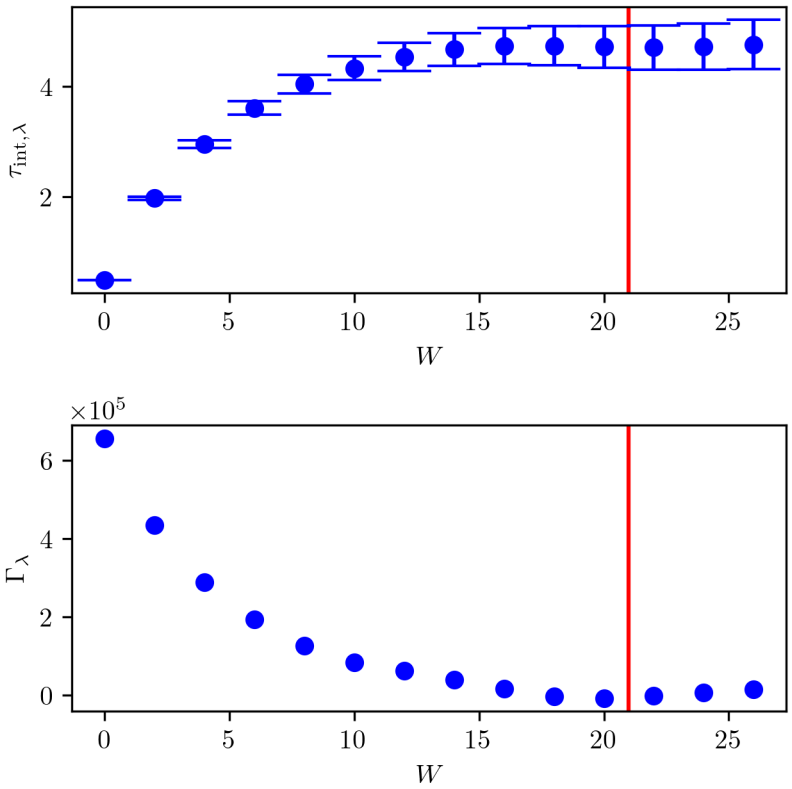


Figure 4.4: Here the autocorrelation function Γ_λ at different lags W is plotted as well as the IATC $\tau_{\text{int},\lambda}$ for the samples from $\pi(\gamma, \lambda|\boldsymbol{y})$ based on the linear forward model.

TT Approximation of Marginal Posterior We approximate the square root of marginal posterior on a predefined univariate grid, where $\gamma = [0.25 \times 10^{15}, 5.5 \times 10^{15}]$ and $\lambda = [100, 5000]$. We set the number of grid points to $n = 20$ and the number of ranks $r = 5$, which we keep constant. Since we do not approximate $\sqrt{\pi(\lambda, \gamma|\boldsymbol{y})}$ in the log-space we introduce a "normalisation constant" $c = 340$. This avoids underflow so that the values $\sqrt{\pi(\lambda, \gamma|\boldsymbol{y})} = \exp\{0.5 \log \pi(\lambda, \gamma|\boldsymbol{y}) + c\}$ are within computer precision. Then we initialise the `tt.cross.rectcross.rect_cross.cross` function, based on the TT cross algorithm in [58, 59], from the Python package `ttpy` [60], with a random tensor and do 1 sweep to obtain a TT approximation of $\pi(\lambda, \gamma|\boldsymbol{y})$. This takes about $0.1s$ for 400 function evaluations. Then we compute the marginals $\pi(\lambda|\boldsymbol{y})$ and $\pi(\gamma|\boldsymbol{y})$ as described in Sec. 2.3.1. In doing so we calculate the coefficient tensor $\boldsymbol{B}$ and $\boldsymbol{B}_{\text{pre}}$ as in Prop. 1 and 2, where we set $\xi = 1/\boldsymbol{\lambda}(\mathcal{X})$ and $\boldsymbol{\lambda}(x) = 1$, so that for Cartesian basis $\boldsymbol{M}_k = \text{diag}(\boldsymbol{\lambda}_k(\mathcal{X}_k))$ in Eq. 12. We plot the TT approximation as a colour map on top of the obtained samples in the scatter plot in Fig. 4.5. We report a RMS error over the whole grid of ...

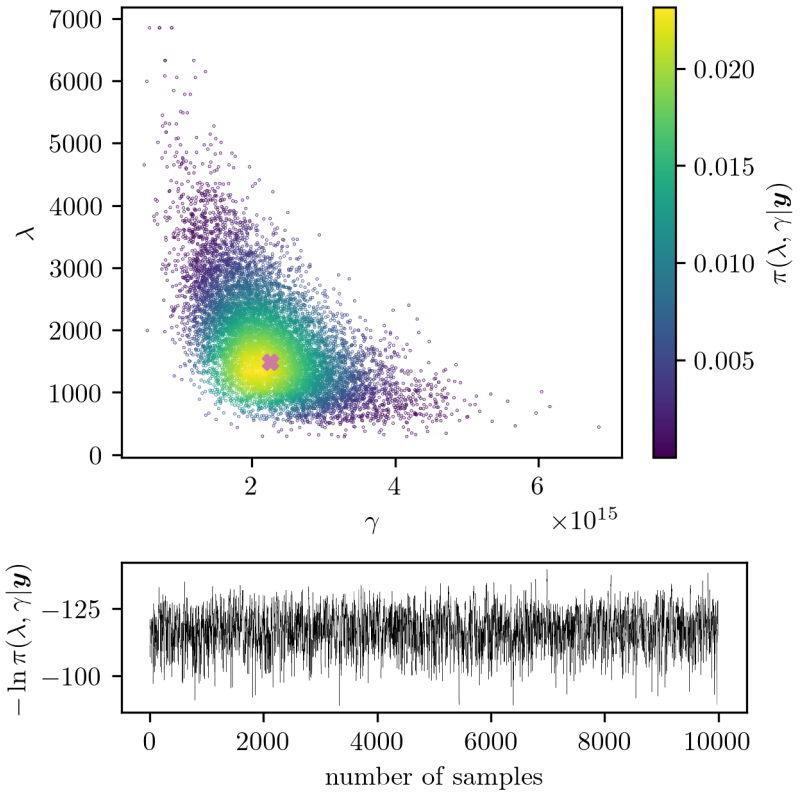


Figure 4.5: We scatter plot the samples of $\lambda = \delta/\gamma$ and γ from the marginal posterior $\pi(\lambda, \gamma|\boldsymbol{y})$ and colour code the samples using the TT approximation of $\pi(\lambda, \gamma|\boldsymbol{y})$. The mode of (λ_0, γ_0) of $\pi(\lambda, \gamma|\boldsymbol{y})$ is marked by the pink cross. To show ergodicity we plot the trace of the samples of the MWG sampler below.

Full Posterior Ozone Mean and Variance

Then we evaluate the normally distributed conditional posterior distribution

$$\boldsymbol{x}|\delta,\gamma,\boldsymbol{y} \sim \mathcal{N}\big(\underbrace{(\boldsymbol{A}^T\boldsymbol{A}+\delta/\gamma\boldsymbol{L})^{-1}\boldsymbol{A}^T\boldsymbol{y}}_{\boldsymbol{x}_\lambda}, \underbrace{(\gamma\boldsymbol{A}^T\boldsymbol{A}+\delta\boldsymbol{L})}_{\gamma\boldsymbol{B}_\lambda}{}^{-1}\big)\,, \tag{4.17}$$

as in Eq. 2.15, with $\lambda = \delta/\gamma$ In this thesis, we compute the mean

$$\mu_{\boldsymbol{x}|\boldsymbol{y}} = \int \boldsymbol{x}_\lambda \pi(\lambda|\boldsymbol{y})\mathrm{d}\lambda \approx \sum \boldsymbol{x}_{\lambda_i}\pi(\lambda_i|\boldsymbol{y})\,, \tag{4.18}$$

and covariance

$$\Sigma_{\boldsymbol{x}|\boldsymbol{y}} = \int \gamma^{-1}\pi(\gamma|\boldsymbol{y})\,\mathrm{d}\gamma \int \boldsymbol{B}_\lambda^{-1}\,\pi(\lambda|\boldsymbol{y})\,\mathrm{d}\lambda \approx \sum {\gamma_i}^{-1}\pi(\gamma_i|\boldsymbol{y}) \sum \boldsymbol{B}_{\lambda_i}^{-1}\pi(\lambda_i|\boldsymbol{y}) \tag{4.19}$$

of $\pi(\boldsymbol{x}|\delta,\gamma,\boldsymbol{y})$ as weighted expectations, by quadrature [61, Sec. 2.1], with $\sum\pi(\lambda_i|\boldsymbol{y}) = \sum\pi(\gamma_i|\boldsymbol{y}) = 1$. The weights $\pi(\lambda_i|\boldsymbol{y})$ and $\pi(\gamma_i|\boldsymbol{y})$ are either given by the TT approximation or by the bins for the sample-based histograms. If calculating the variance is too costly, the RTO method (see Sec. 2.2.3) may be a feasible alternative to draw a sample from Eq. 4.17.

Based on the marginal posterior distribution $\pi(\lambda,\gamma|\boldsymbol{y})$ we calculate the mean and covariance of the conditional posterior $\pi(\boldsymbol{x}|\lambda,\gamma,\boldsymbol{y})$ by quadrature as in Eq. 4.18 and Eq. 4.19. We can either use the sample based histogram bins as weights or the TT approximation to integrate over marginal approximations $\pi(\lambda|\boldsymbol{y})$ and $\pi(\gamma|\boldsymbol{y})$. More precisely, for the sample based evaluation, the height of the histogram bars act as quadrature weights, e.g. $\pi(\lambda_i|\boldsymbol{y})$ at the centre λ_i of each bin. Then, we obtain the full conditional mean $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}$ and covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{x}|\boldsymbol{y}}$ as weighted expectations. Again, we use Cholesky decomposition to invert $\boldsymbol{B}_\lambda = \boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L}$ and to calculate $\boldsymbol{x}_\lambda = (\boldsymbol{A}^T\boldsymbol{A}+\lambda\boldsymbol{L})^{-1}\boldsymbol{A}^T\boldsymbol{y}$. In total we have to evaluate $\boldsymbol{x}_\lambda$ and invert $\boldsymbol{B}_\lambda$ 20 times to obtain mean and covariance of $\pi(\boldsymbol{x}|\boldsymbol{y})$, Fig. 4.6, which takes less than 0.2s. We plot posterior samples of $\pi(\boldsymbol{x}|\boldsymbol{y})$ in Fig. 4.6, where we set negative ozone VMR to zero. The fact that we have to deal with negative ozone values is due to the poor prior choice of $\pi(\boldsymbol{x}|\delta)$. Note, that the sample mean is slight larger than posterior mean at heights where the data is noise dominated, and the ozone values are determined by the prior, or where the ground truth is close to zero. This indicates that we should use a different more physical based prior or model to parametrise the ozone profile. Note, that the posterior samples do not represent the ozone peak at around 80km.

Errors due to number of grid points we plot RMS for mean and variance compared to a solutop with 200 grid points. The relative error behaves proportionally to $1/N$, see Fig. 4.7 and Eq. A.11, and we consider a relative error less than 0.1% good enough. This happens roughly at a bin size of 25, which is our TT grid size. Note that we exclude the error due to τ_{int} the IACT and that we choose the grid according to the sampled values so that the sampling region is the same as the region in which we approximate the posterior distributions. . The error of is influeced by rank and gridsize interploation error

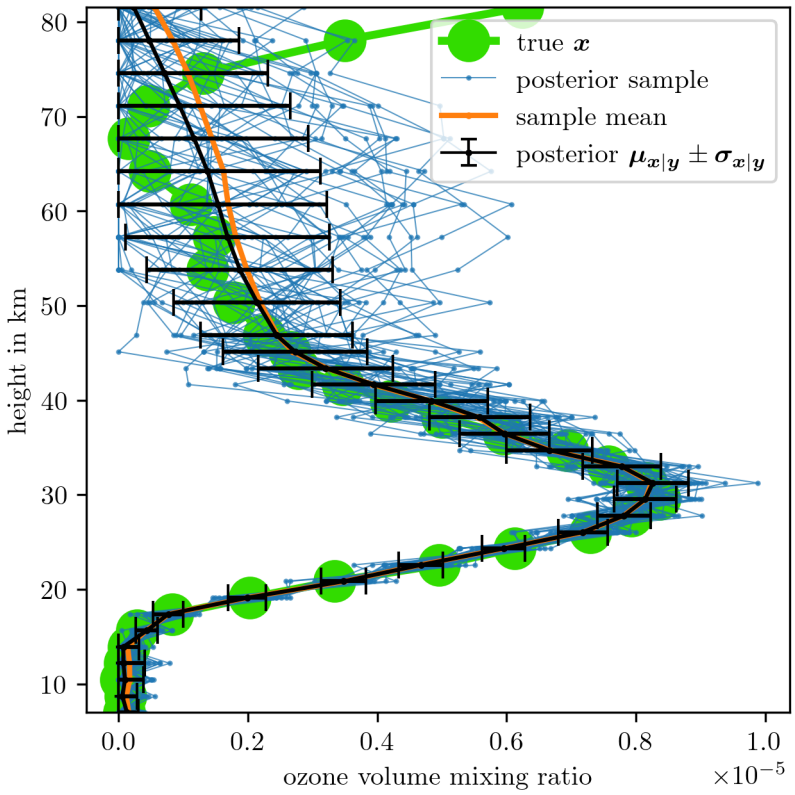


Figure 4.6: We draw ozone samples from the full posterior distribution $\pi(\boldsymbol{x}|\boldsymbol{y})$ after characterising mean and covariance of $\pi(\boldsymbol{x}|\boldsymbol{y})$ by weighted expectations over the marginal posterior $\pi(\lambda, \gamma|\boldsymbol{y})$. we determine $\pi(\lambda, \gamma|\boldsymbol{y})$ either through sampling or via TT approximation based on the linear forward map $\boldsymbol{A}_L$. Note that we set negative values ozone VMR values to zero. We will use those samples to find the affine map $\boldsymbol{M}$, see section 4.3

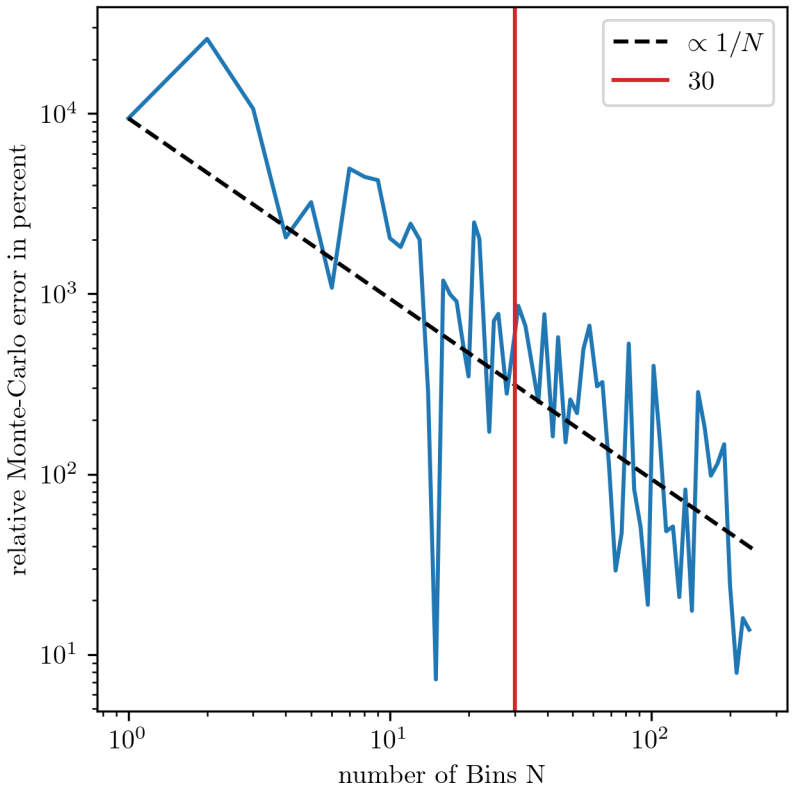


Figure 4.7: Assessment of Monte-Carlo error, where we calculate the relative error of the mean due to binning up the samples compared to the sample mean $||\boldsymbol{\mu}_{\text{samp}} - \boldsymbol{\mu}_{\text{distr}}||/||\boldsymbol{\mu}_{\text{samp}}||$.

4.3 Approximate non-linear Forward Model with an Affine Map

Given the posterior distribution for ozone $\pi(\boldsymbol{x}|\boldsymbol{y})$, we can now approximate the non-linear forward model

$$\boldsymbol{A}_{NL} \approx \boldsymbol{M}\boldsymbol{A}_L\,, \tag{4.20}$$

with an affine map $\boldsymbol{M}$, see Fig. 4.8 for the summarised strategy. Here we write $\boldsymbol{A}_{NL}\boldsymbol{x}$, which implies that we construct the non-linear forward model and compute non-linear noise-free data $\boldsymbol{A}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T})$ based on ground truth pressure and temperature. Using

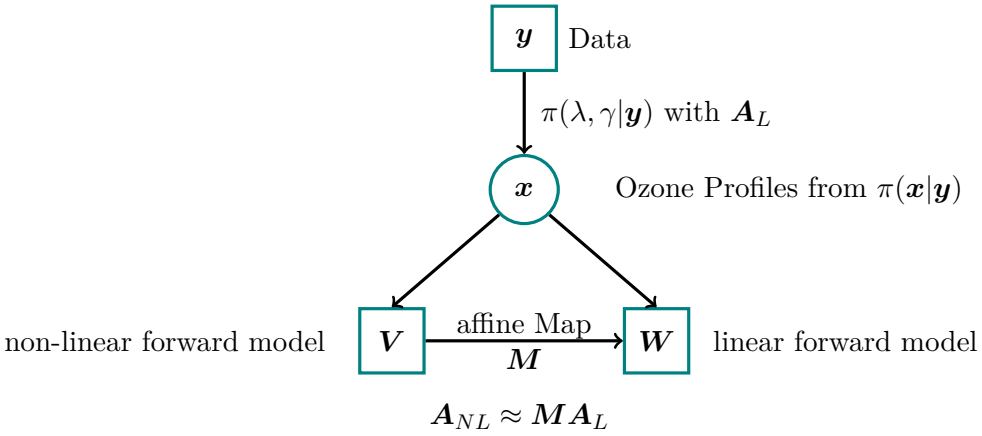


Figure 4.8: The strategy to find the affine map consist of evaluating the marginal posterior for ozone using the linear forward model. Then we draw ozone samples from the conditional posterior and calculate noise free data based on the linear and non-linear forward model. Next we find a mapping in between those two space so that we can approximate the non-linear forward model using an affine map and the linear forward model.

posterior ozone samples we generate two affine subspaces and then find the mapping between those. The subspace $\boldsymbol{W}$ is created by noise free data based on the linear model $\boldsymbol{A}_L$ and $\boldsymbol{V}$ by noise free data based on the non-linear model $\boldsymbol{A}_{NL}$, given m samples $\boldsymbol{x}^{(j)} \sim \pi(\boldsymbol{x}|\boldsymbol{y})$ for $j = 1, \ldots, m$. We report a relative RMS difference between $\boldsymbol{W}$ and $\boldsymbol{V}$ of about 1%, which we aim to reduce through the affine map $\boldsymbol{M}$. More specifically, the affine subspace associated with the linear forward model is

$$\boldsymbol{W} = \begin{bmatrix} | & & | & & | \\ \boldsymbol{A}_L\boldsymbol{x}^{(1)} & \cdots & \boldsymbol{A}_L\boldsymbol{x}^{(j)} & \cdots & \boldsymbol{A}_L\boldsymbol{x}^{(m)} \\ | & & | & & | \end{bmatrix} \in \mathbb{R}^{m\times m} \tag{4.21}$$

and with the non-linear forward model is

$$\boldsymbol{V} = \begin{bmatrix} | & & | & & | \\ \boldsymbol{A}_{NL}\boldsymbol{x}^{(1)} & \cdots & \boldsymbol{A}_{NL}\boldsymbol{x}^{(j)} & \cdots & \boldsymbol{A}_{NL}\boldsymbol{x}^{(m)} \\ | & & | & & | \end{bmatrix} = \begin{bmatrix} — & v_1 & — \\ & \vdots & \\ — & v_j & — \\ & \vdots & \\ — & v_m & — \end{bmatrix} \in \mathbb{R}^{m\times m}. \quad (4.22)$$

Then the we calculate affine map

$$\boldsymbol{V}\boldsymbol{W}^{-1} = \boldsymbol{M} = \begin{bmatrix} — & r_1 & — \\ & \vdots & \\ — & r_j & — \\ & \vdots & \\ — & r_m & — \end{bmatrix} \in \mathbb{R}^{m\times m}. \quad (4.23)$$

by solving $v_j = r_j\boldsymbol{W}$ for each row r_j in $\boldsymbol{M}$, where $j = 1, \ldots, m$, using the Python function `numpy.linalg.solve`. We can do that because every measurement in the data vector $\boldsymbol{y}$ is independent of each other, and then every row v_j of $\boldsymbol{V} \in \mathbb{R}^{m\times m}$ is independent of each other as well.

We asses the affine map by calculating the relative RMS difference $\|\boldsymbol{M}\boldsymbol{W}-\boldsymbol{V}\|_{L^2}/\|\boldsymbol{M}\boldsymbol{W}\|_{L^2}$ between the mapped linear noise free data and the non-linear noise free data, which is approximately 0.001%. In Fig. 4.9, we show the mapping for one posterior ozone sample which has not been used to create this mapping. In other word this is an unseen event not in the training data. The relative RMS error for this approximation is roughly 0.03% and much smaller than the relative difference between noise free linear data and non-linear data. Consequently, from here onwards, we use the approximated forward map.

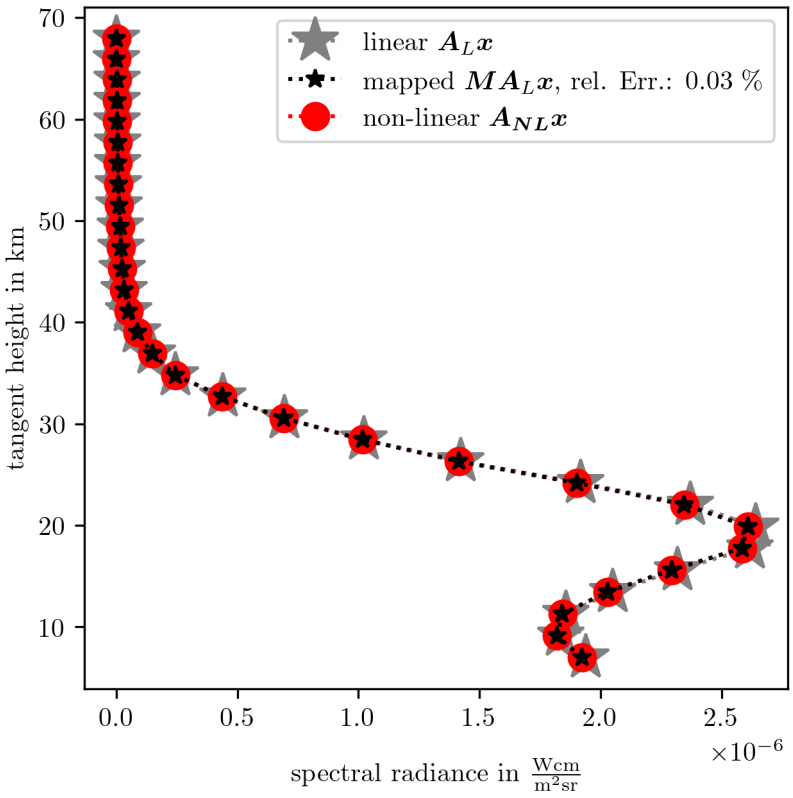


Figure 4.9: We asses how good we can map a new ozone sample $\boldsymbol{x} \sim \pi(\boldsymbol{x}|\boldsymbol{y})$ from the linear forward model onto the non-linear forward model using the previous calculated affine map $\boldsymbol{M}$. The sample has not been used to create this affine map. The gray stars represent noise free linear data, where as the red circles present noise free non-linear data. Then we map the linear noise free data onto the non-linear noise free data, black start, and provide the relative RMS error in between the mapped noise free data and the non-linear data.

4.4 Regularisation Solution vs. Bayesian Approach – approximated Model

With the affine approximation we define the forward model matrix

$$\boldsymbol{A} \coloneqq \boldsymbol{M}\boldsymbol{A}_L \tag{4.24}$$

using the affine map $\boldsymbol{M}$. Here we compare the posterior distribution of ozone to a regularisation approach.

4.4.1 Posterior Distribution for Ozone

Again we us the MTC scheme and the exact same setup as in Sec. 4.2.2 to evaluate the marginal posterior and then the conditional posterior of ozone.

Marginal Posterior

The marginal posterior is defined as in Eq. 4.8, but with updated forward model. We initialise the MWG at the mode of $\pi(\lambda, \gamma|\boldsymbol{y})$ and approximate $f(\lambda)$ and $g(\lambda)$ around the mode as in Eq. 4.12a and Eq. 4.12b. Then we run the MWG algorithm for $N = 10000$ plus $N_{\text{burn-in}} = 100$ steps and plot the samples in Fig. 4.10 as well as the marginal approximations provided by the TT decomposition on the same grid as previously defined with same number of ranks (see Sec. 4.2.2).

Full Posterior Ozone Mean and Variance

Next, we characterise the conditional posterior $\pi(\boldsymbol{x}|\boldsymbol{y})$ as in Eq. 4.17. Again, we calculate the full posterior mean $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}$, see Eq. 4.18, and covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{x}|\boldsymbol{y}}$ 4.19 as weighted expectation over a 20-point grid provided by either the marginal TT approximations of $\pi(\gamma|\boldsymbol{y})$ and $\pi(\lambda|\boldsymbol{y})$ or by the bins of the sample histogram as quadrature weights. We plot the conditional mean and variance in Fig. 4.11, the regularised solution (see next section), and one sample from the posterior, which represents a feasible solution to this inverse problem. We can see that the ground truth lays within 3 times of the STD (accounting for $\approx 99\%$ of all posterior samples) around the mean except for the peak at around 80km. We also note that compared to the previously calculated mean and variance based on the linear forward model, see 4.6, the approximated based posterior distribution does not differ significantly. This is expected since the difference between the linear and non-linear forward map of $\approx 1\%$ is small.

Additionally in Fig. 4.12, we plot the singular values of the covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{x}|\boldsymbol{y}}$, to visualise how many ozone values are informative. We observe that the last 10 singular values are very small and correspond to ozone values at the high altitudes

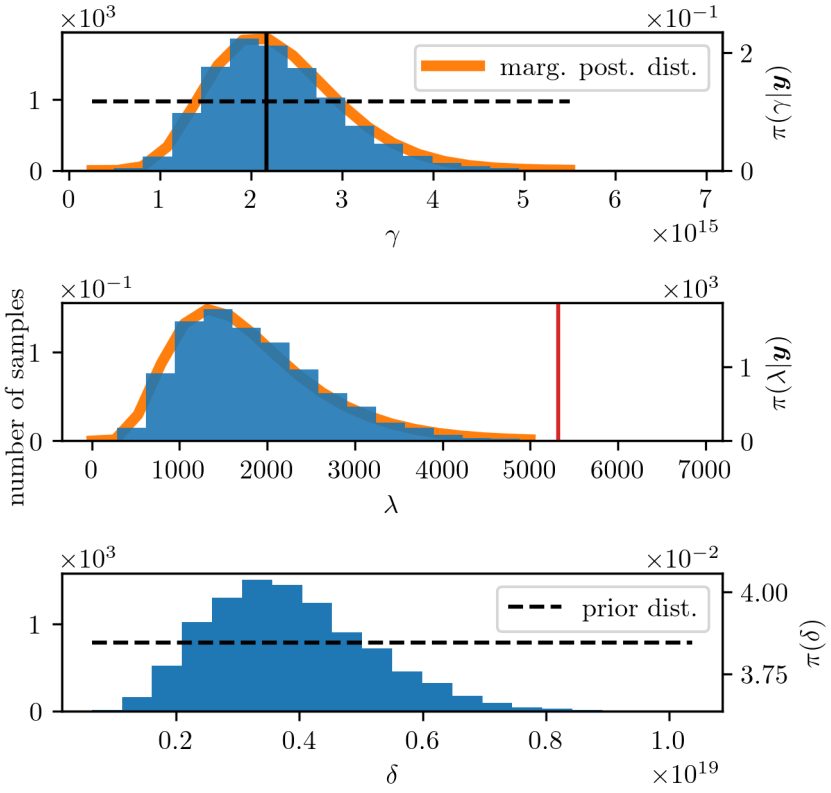


Figure 4.10: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line. Note that we sample λ and γ using the Metropolis-within-Gibbs sampler and can calculate δ for every sample of the marginal posterior, we can not do this for the TT approximation. The regularised parameter corresponding to the regularised solution is marked thought the red vertical line at $\lambda_{\text{reg}} =$.

$\gtrsim$ 45km with a large variance, see Fig. 4.11. This is also roughly the region where we introduce a higher correlation structure due (see Eq. 4.5 Sec. 4.2.1) and the samples are determined by the prior.

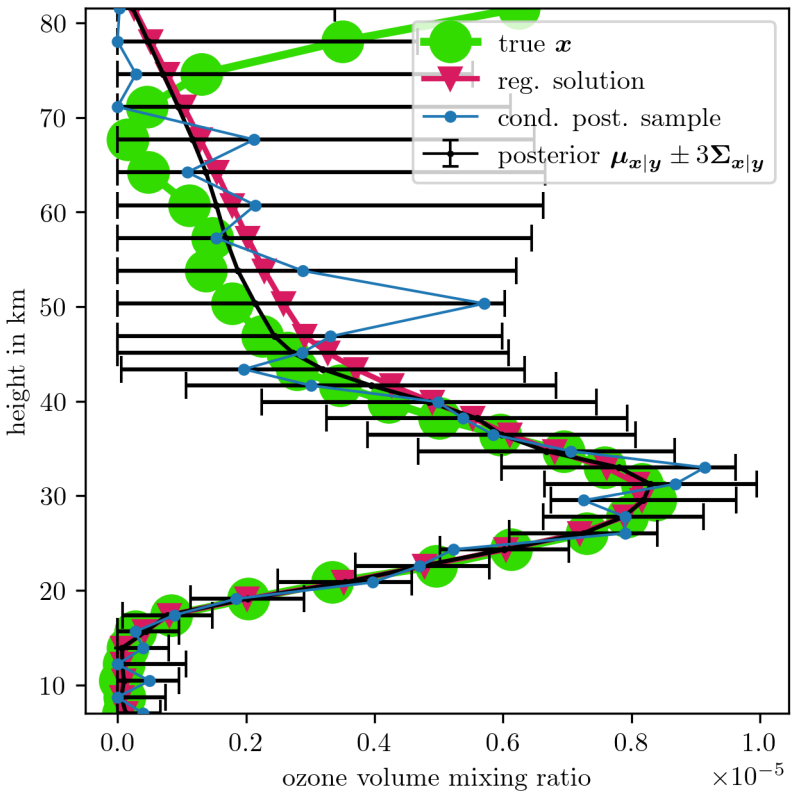


Figure 4.11: We plot the conditional posterior mean and variance in black and the regularised solution on top of the ground truth ozone profile in green. We use the updated forward map $\boldsymbol{MA}_L$

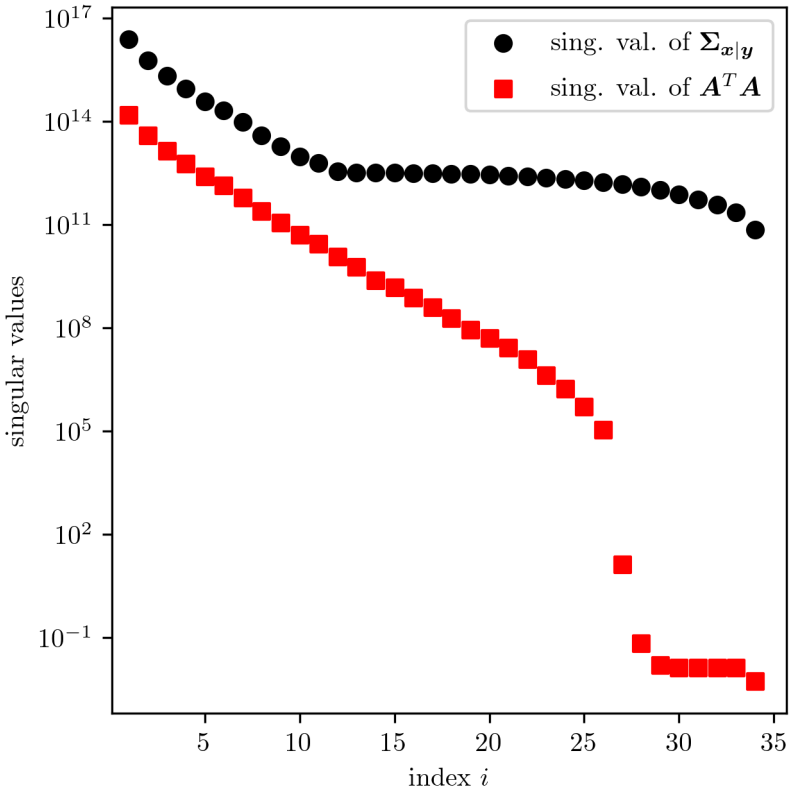


Figure 4.12: Singular values of the covariance matrix of $\boldsymbol{\Sigma}_{\boldsymbol{x}|\boldsymbol{y}}^{-1} = (\sum(\boldsymbol{A}^T\boldsymbol{A} + \boldsymbol{L})^{-1})^{-1}$ of the posterior distribution $\pi(\boldsymbol{x}|\boldsymbol{y})$ for ozone.

4.4.2 Solution by Regularisation

Since we claim that Bayesian analysis is superior to regularisation methods we compare the MTC method to a Tikhonov regularisation solution, see Sec. 2.5 and [14]. This is most similar to our chosen linear-Gaussian Bayesian framework. The Tikhonov regularised solution is defined as in [14, 63]

$$\boldsymbol{x}_\lambda = \arg\min_{\boldsymbol{x}} \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \lambda \boldsymbol{x}^T \boldsymbol{L}\boldsymbol{x}\,, \tag{4.25}$$

with the regularisation parameter λ. The regularised solution is typically calculated by solving the normal equations, see Sec. 2.5,

$$\boldsymbol{x}_\lambda = (\boldsymbol{A}^T\boldsymbol{A} + \lambda \boldsymbol{L})^{-1}\boldsymbol{A}^T\boldsymbol{y}\,. \tag{4.26}$$

To find the best regularised solution, we use the L-curve method [44]. Within this method we compute $\boldsymbol{x}_\lambda$, for 200 different λ values in between $10^{0.5}$ to $10^{6.5}$ and plot the solution semi norm $\sqrt{\boldsymbol{x}_\lambda^T \mathbf{L}\boldsymbol{x}_\lambda}$ against the data misfit norm $\|\boldsymbol{A}\boldsymbol{x}_\lambda - \boldsymbol{x}\|$, see Figure 4.13. The best regularised solution corresponding to the corner of the L-curve is located at the point of maximum curvature, see triangle in Fig. 4.13, which we find with the kneedle algorithm [64] using the python function `kneed.KneeLocator` in less 0.1s.

We plot the regularised solution in Fig. 4.11 and observe that it is very similar to the posterior mean. It is pretty clear that the regularised solution accounts for only one possible solution and does not provide uncertainties. The regularised solution is not similar to the samples drawn from the posterior $\pi(\boldsymbol{x}|\boldsymbol{y})$, see also Fig. 4.6. In Fig. 4.13, the samples of $\pi(\boldsymbol{x}|\boldsymbol{y})$ lie above the L-Curve where as the posterior mean and the regularised solution lie on the L-Curve. This does make sense, if one thinks about the mean (average over non-smooth samples) and the regularised solution as extremely smooth ozone profiles, see also Sec. 2.5. In contrast the samples are less regularised and hence lie above the L-Curve, but have a similar data misfit norm and as already mentioned are all feasible solution to the data. Neither the regularisation solution nor the posterior ozone profiles capture the ozone peak of the ground truth at high altitudes.

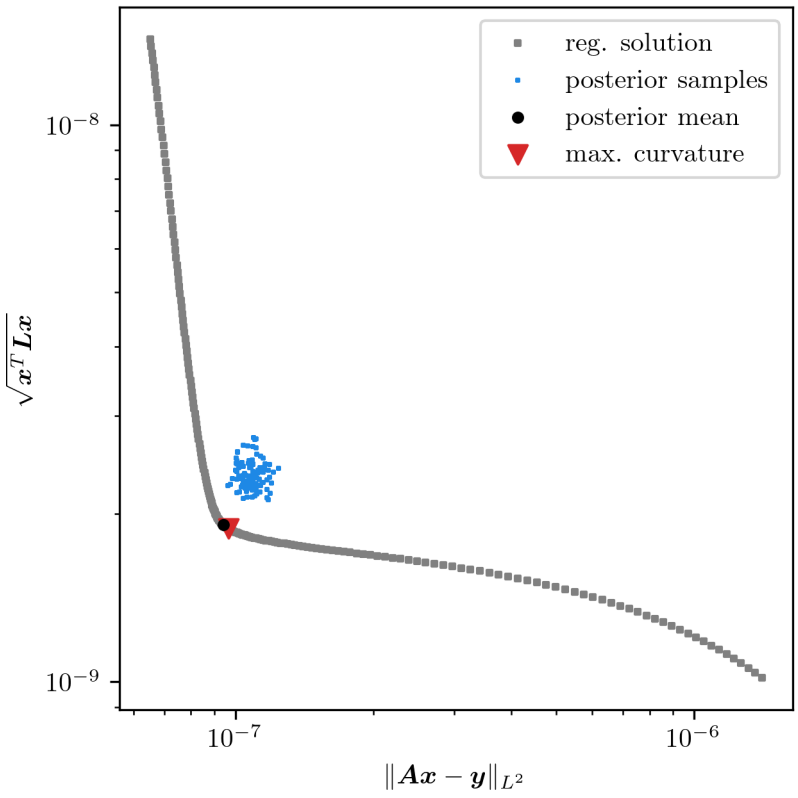


Figure 4.13: We calculate regularised solution as in Eq. **??** and plot the regularised semi norm $\sqrt{\boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x}}$ against the data misfit norm $||\boldsymbol{A}\boldsymbol{x}-\boldsymbol{y}||$ to find the regularised solution at the point of maximum curvature of the so-called L-Curve. Additionally we calculate the data misfit norm and the regularised norm for the ozone posterior and for samples of the conditional posterior distribution. make box around Kneedle reagion

4.5 Hierarchical Bayesian Framework for Ozone, Pressure and Temperature

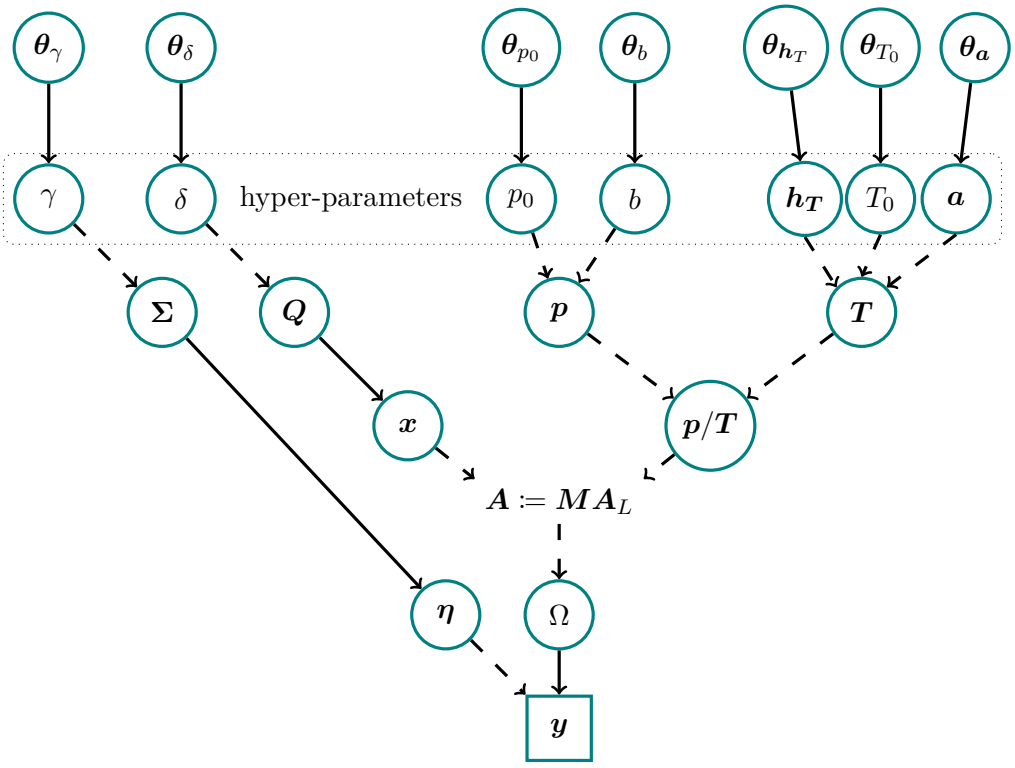


Figure 4.14: DAG of Bayesian model for ozone, pressure and temperature. The hyper-parameters $\boldsymbol{h}_T = \{h_{T,1}, h_{T,2}, h_{T,3}, h_{T,4}, h_{T,5}, h_{T,6}\}$, $\boldsymbol{a} = \{a_0, a_1, a_2, a_3, a_4, a_5, a_6\}$, T_0, b and p_0 deterministically (dotted line) describe pressure parameter $\boldsymbol{p}$ through the function in Eq. 4.28, and temperature parameter $\boldsymbol{T}$ through the function in Eq. 4.3. In this case, we decide the hyper-prior distribution $\pi(\boldsymbol{h}_T, \boldsymbol{a}, b, p_0)$ to be a normal distribution, determined by $\boldsymbol{\theta}_{\boldsymbol{h}_T}, \boldsymbol{\theta}_{\boldsymbol{a}}, \boldsymbol{\theta}_{T_0}, \boldsymbol{\theta}_b, \boldsymbol{\theta}_{p_0}$, which represent mean and variances. The ozone parameter $\boldsymbol{x}$ is statistically (solid line) described by the prior distribution $\boldsymbol{x}|\delta \sim \mathcal{N}(0, \delta\boldsymbol{L})$. Here the hyper-parameter δ accounts for smoothness in the ozone profile and defines the precision matrix $\boldsymbol{Q} = \delta\boldsymbol{L}$, where $\boldsymbol{L}$ is graph Laplacian as in Eq. 4.5. The noise covariance $\boldsymbol{\Sigma} = \gamma^{-1}\boldsymbol{I}$ of the random noise vector $\boldsymbol{\eta} \sim \mathcal{N}(0, \gamma^{-1}\boldsymbol{I})$ is defined by the noise hyper-parameter γ. As in Sec. 4.2 described, $\boldsymbol{\theta}_\delta$ and $\boldsymbol{\theta}_\gamma$ define the hyper-priors $\pi(\delta, \gamma)$. Then, we randomly observe a data set $\boldsymbol{y}$ from the space of all measurables Ω through the approximated froward model $\boldsymbol{M}\boldsymbol{A}_L$ including some added noise. Given the data we like to determine the posterior distribution over the hyper-parameters $\pi(\boldsymbol{h}_T, \boldsymbol{a}, b, p_0, \delta, \gamma|\boldsymbol{y})$ first and then $\pi(\boldsymbol{x}|\boldsymbol{h}_T, \boldsymbol{a}, b, p_0, \delta, \gamma, \boldsymbol{y})$ utilising the MTC scheme.

As in Sec. 4.2, we use the DAG in Fig. 4.14 to visualise the measurement process and correlations in between parameters. We already see that the parameters pressure $\boldsymbol{p}$, temperature $\boldsymbol{T}$ and ozone $\boldsymbol{x}$ are correlated, and progress deterministically (dashed line) into the forward model, via $\boldsymbol{x} \times \boldsymbol{p}/\boldsymbol{T}$. Through their respective prior distributions they generate a space of all possible noise free data Ω, from which we observe some data included some added normally distributed noise $\boldsymbol{\eta}$. This hierarchical Bayesian framework includes the hyper-parameters $p_0, b, \boldsymbol{h}_T, \boldsymbol{T}_0, \boldsymbol{a}$, for pressure and temperature,

		TT bounds			
model parameters	priors	lower	upper	τ_{int}	Context
γ	$\mathcal{T}(1, 10^{-10})$	$5\,10^{-8}$	$4.5\,10^{-7}$	9 ± 0.1	$\boldsymbol{y}$
δ	$\mathcal{T}(1, 10^{-10})$	-	-	1.5 ± 0.1	$\boldsymbol{x}$
$\lambda = \delta/\gamma$	-	500	10^4	3.5 ± 0.3	$\boldsymbol{x}$
$\boldsymbol{x}$	$\mathcal{N}(0, \delta \boldsymbol{L})$	-	-	-	$\boldsymbol{x}$
p_0	$\mathcal{N}(1243, 5)$	1229	1259	550 ± 9	$\boldsymbol{p}/\boldsymbol{T}$
T_0	$\mathcal{N}(288.15, 4.5)$	275	302	2446 ± 76	$\boldsymbol{p}/\boldsymbol{T}$
$h_{T,1}$	$\mathcal{N}(11, 0.5)$	9.5	12.5	1820 ± 49	$\boldsymbol{p}/\boldsymbol{T}$
b	$\mathcal{N}(0.167, 5\,10^{-4})$	0.165	0.171	2813 ± 92	$\boldsymbol{p}/\boldsymbol{T}$
$h_{T,3}$	$\mathcal{N}(32.3, 2.5)$	25.2	39.8	394 ± 5	$\boldsymbol{p}/\boldsymbol{T}$
a_0	$\mathcal{N}(-6.5, 0.01)$	-6.53	-6.47	330 ± 4	$\boldsymbol{p}/\boldsymbol{T}$
$h_{T,2}$	$\mathcal{N}(20.1, 1.6)$	17.7	22.3	454 ± 7	$\boldsymbol{p}/\boldsymbol{T}$
a_1	$\mathcal{N}(0, 0.1)$	-0.3	0.3	508 ± 8	$\boldsymbol{p}/\boldsymbol{T}$
a_2	$\mathcal{N}(1, 0.01)$	0.97	1.03	341 ± 5	$\boldsymbol{p}/\boldsymbol{T}$
a_3	$\mathcal{N}(2.8, 0.1)$	2.5	3.1	316 ± 4	$\boldsymbol{p}/\boldsymbol{T}$
$h_{T,4}$	$\mathcal{N}(47.4, 5)$	45.9	48.9	324 ± 4	$\boldsymbol{p}/\boldsymbol{T}$
a_4	$\mathcal{N}(0, 0.1)$	-0.3	0.3	335 ± 4	$\boldsymbol{p}/\boldsymbol{T}$
$h_{T,5}$	$\mathcal{N}(51.4, 5)$	49.9	52.9	319 ± 4	$\boldsymbol{p}/\boldsymbol{T}$
a_5	$\mathcal{N}(-2.8, 0.1)$	-3.1	-2.5	335 ± 4	$\boldsymbol{p}/\boldsymbol{T}$
$h_{T,6}$	$\mathcal{N}(71.8, 3)$	62.5	80.8	347 ± 5	$\boldsymbol{p}/\boldsymbol{T}$
a_6	$\mathcal{N}(-2, 0.01)$	-2.03	-1.97	320 ± 4	$\boldsymbol{p}/\boldsymbol{T}$

Table 4.2: Summary of relevant parameter characteristics, bounds and sampling statistics. We denote $\mathcal{N}(\mu, \sigma)$ as the Gaussian and $\mathcal{T}(\alpha = \text{scale}, \beta = \text{rate})$ as the gamma distribution. The IACT τ_{int} is estimated as in [65] from posterior samples based on the approximated forward map.

δ for ozone smoothness and γ for noise precision. Here pressure $\boldsymbol{p}$ and temperature $\boldsymbol{T}$ are functionally dependent on $p_0, b, \boldsymbol{h}_T, \boldsymbol{T}_0, \boldsymbol{a}$, see Eq. 4.3 and Eq. 4.28. Each of those hyper-parameters is described by the hyper-prior distribution $\pi(\gamma, \delta, p_0, b, \boldsymbol{h}_T, \boldsymbol{T}_0, \boldsymbol{a})$ defined by us. Here $\boldsymbol{\theta}_\gamma, \boldsymbol{\theta}_\delta, \boldsymbol{\theta}_{p_0}, \boldsymbol{\theta}_b, \boldsymbol{\theta}_{\boldsymbol{h}}, \boldsymbol{\theta}_{T_0}, \boldsymbol{\theta}_{\boldsymbol{a}}$ determine gamma distributions $\gamma, \delta \sim \pi(\boldsymbol{\theta}_\gamma, \boldsymbol{\theta}_\delta)$, so that e.g. $\gamma \sim \Gamma(\alpha_\gamma, \beta_\gamma)$ with $\boldsymbol{\theta}_\gamma = \{\alpha_\gamma, \beta_\gamma\}$, and a normal distribution $p_0, b, \boldsymbol{h}_T, \boldsymbol{T}_0, \boldsymbol{a} \sim \pi(\boldsymbol{\theta}_{p_0}, \boldsymbol{\theta}_b, \boldsymbol{\theta}_{\boldsymbol{h}}, \boldsymbol{\theta}_{T_0}, \boldsymbol{\theta}_{\boldsymbol{a}})$, so that e.g. $b \sim \mathcal{N}(\mu_b, \sigma_b)$ and $\boldsymbol{\theta}_b = \{\mu_b, \sigma_b\}$. We denote the forward model as $\boldsymbol{A} \coloneqq \boldsymbol{M}\boldsymbol{A}_L$.

Then, we set up the hierarchical Bayesian framework

$$\boldsymbol{y}|\boldsymbol{x}, p_0, b, \boldsymbol{h_T}, T_0, \boldsymbol{a}, \delta, \gamma \sim \mathcal{N}(\boldsymbol{A}(p_0, b, \boldsymbol{h_T}, T_0, \boldsymbol{a})\, \boldsymbol{x}, \gamma^{-1}\boldsymbol{I}) \tag{4.27a}$$

$$\boldsymbol{x}|\delta \sim \mathcal{N}(\boldsymbol{0}, (\delta \boldsymbol{L})^{-1}) \tag{4.27b}$$

$$\delta \sim \Gamma(\alpha_\delta, \beta_\delta) \tag{4.27c}$$

$$\gamma \sim \Gamma(\alpha_\gamma, \beta_\gamma) \tag{4.27d}$$

$$\boldsymbol{a} \sim \mathcal{N}(\boldsymbol{\mu_a}, \boldsymbol{\Sigma_a}) \tag{4.27e}$$

$$\boldsymbol{h_T} \sim \mathcal{N}(\boldsymbol{\mu}_T, \boldsymbol{\Sigma}_{\boldsymbol{h}_T}) \tag{4.27f}$$

$$T_0 \sim \mathcal{N}(\mu_{T_0}, \sigma_{T_0}) \tag{4.27g}$$

$$p_0 \sim \mathcal{N}(\mu_{p_0}, \sigma_{p_0}) \tag{4.27h}$$

$$b \sim \mathcal{N}(\mu_b, \sigma_b) \tag{4.27i}$$

and define a normally distributed likelihood (due to Gaussian noise) and normally distributed priors, where the hyper-prior means and variances are described through $\boldsymbol{\theta_a} = (\boldsymbol{\mu_a}, \boldsymbol{\Sigma_a})$, $\boldsymbol{\theta}_{\boldsymbol{h}_T} = (\boldsymbol{\mu_T}, \boldsymbol{\Sigma}_{\boldsymbol{h}_T})$, $\boldsymbol{\theta}_{T_0} = (\mu_{T_0}, \sigma_{T_0})$, $\boldsymbol{\theta}_{p_0} = (\mu_{p_0}, \sigma_{p_0})$, and $\boldsymbol{\theta}_b = (\mu_b, \sigma_b)$. which we summarise in Tab. 4.2. Before we formulate the posterior distribution we carefully define $\boldsymbol{\theta}_\gamma, \boldsymbol{\theta}_\delta, \boldsymbol{\theta}_{p_0}, \boldsymbol{\theta}_b, \boldsymbol{\theta_h}, \boldsymbol{\theta}_{T_0}, \boldsymbol{\theta_a}$.

4.5.1 Prior Modelling

The first we do, is to describe the pressure $\boldsymbol{p}$ in between $h_{L,0} \approx$ 7km and $h_{L,n} \approx$ 82km with an exponential function

$$p(h) = \exp(-b\,h)\; p_0 \quad , h_{L,0} \leq h \leq h_{L,n} \tag{4.28}$$

dependent on two hyper-parameters p_0, b, see Fig. 4.16. Similarly, the temperature as described in Eq. 4.3 can be parametrised with 14 hyper-parameters $\boldsymbol{h_T} = \{h_{T,1}, h_{T,2}, h_{T,3}, h_{T,4}, h_{T,5}, h_{T,6}\}$, $\boldsymbol{a} = \{a_0, a_1, a_2, a_3, a_4, a_5, a_6\}$ and T_0 (see Fig. 4.15 and Eq. 4.3). To complete the model, we have to define a sensible hyper-prior distribution $\pi(\boldsymbol{h_T}, \boldsymbol{a}, T_0)$. In doing so, we choose the variance and mean of the normally distributed hyper-prior distribution $\pi(\boldsymbol{h_T})$ so that the temperature profile maintains its structure, $h_{T,i} < h_{T,i+1}$ for $i = 1, \ldots, 5$ (see Fig. B.5). Further, we define $\pi(\boldsymbol{a})$ as normally distributed, because we find (see Fig. 4.17) that the data is uninformative about the temperature profile. Similarly, we set $\pi(T_0)$ to a normal distribution so that it mimics a daily temperature variability of roughly 30K. The hyper-prior distribution $\pi(p_0, b)$ for pressure-related hyper-parameters is also normally distributed. We choose the variance for $\pi(p_0)$ so that p_0 has a variability of around 80hPa, close to what we can observe when looking at weather data. Note that we fit one exponential function to ground truth pressure values between $h_{L,0} \approx$ 7km and

$h_{L,n} \approx 82$, so that the pressure values p_0 at sea level may be different to true observed pressure values due to that approximation. We set means of the normal distribution of $\pi(\boldsymbol{h}_T, \boldsymbol{a}, T_0, p_0, b)$ to the ground truth values of $\boldsymbol{p}$ and $\boldsymbol{T}$, and summarise the hyper-prior means and variances in Tab. 4.2.

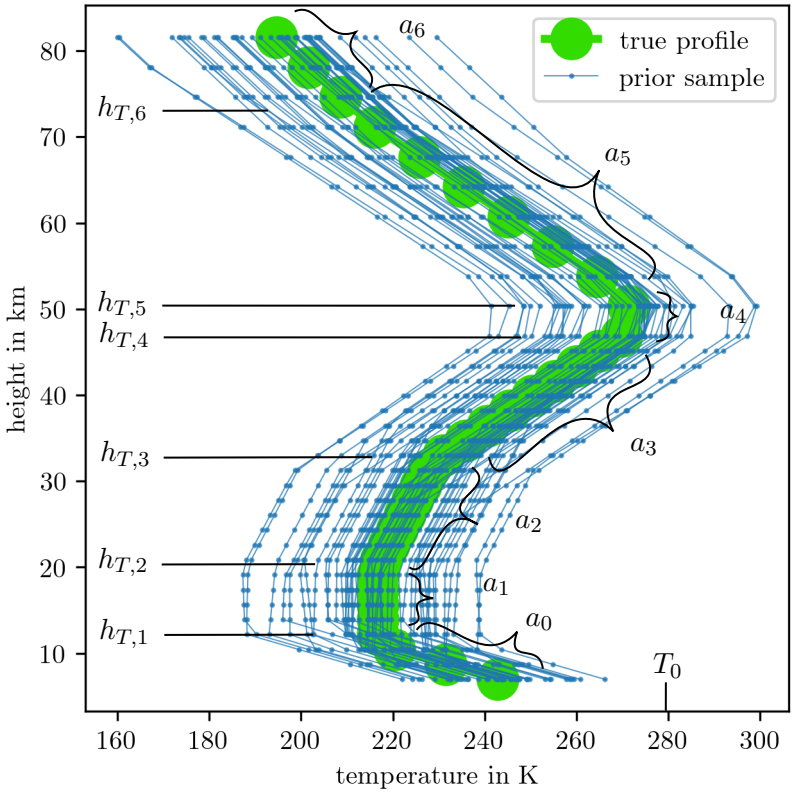


Figure 4.15: We draw samples from the hyper-prior distribution of $h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4$ and T_0 as defined in table 4.2 and then calculate $\boldsymbol{T}$ according to the function in Eq. 4.3.

We plot prior samples of the pressure $\boldsymbol{p}$ in Fig. 4.16, the temperature $\boldsymbol{T}$ in Fig. 4.15 and the ratio $\boldsymbol{p}/\boldsymbol{T}$ in Fig. 4.17 against the ground truth profiles. Additionally, we plot prior samples of $1/\boldsymbol{T}$ in Fig. B.6. Here we already observe that $\boldsymbol{p}/\boldsymbol{T}$ inherits the structure of the pressure function and hence the data is uninformative about the temperature, and that is one of the reasons why we chose those normally, rather informative, hyper-prior distributions.

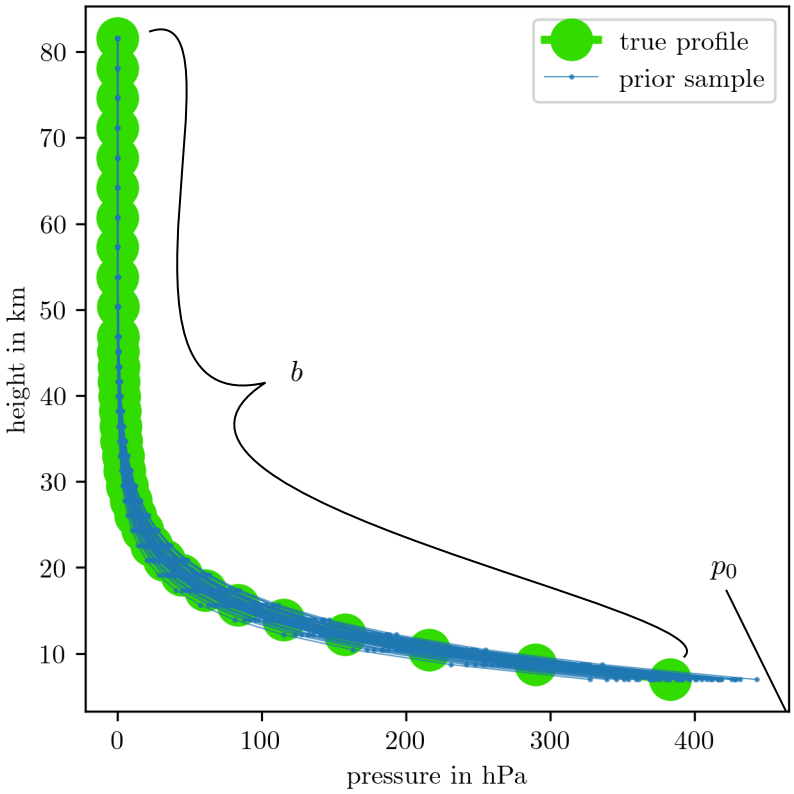


Figure 4.16: We draw samples from the hyper-prior distribution of h_0, b and p_0 as defined in table 4.2 and then calculate $\boldsymbol{p}$ according to the function in Eq. 4.28.

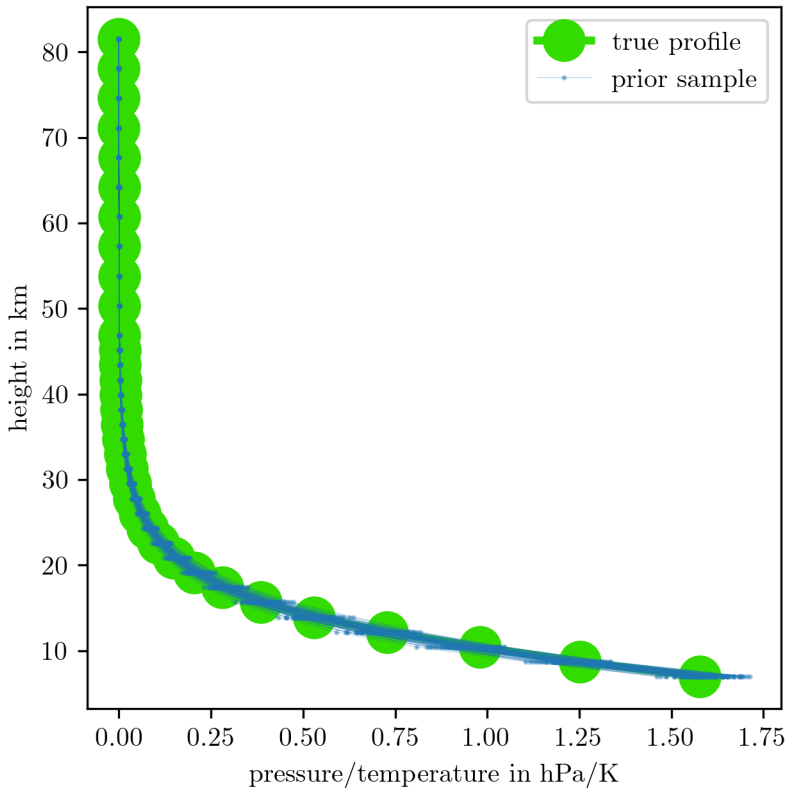


Figure 4.17: We draw samples from the hyper-prior distribution of $h_0, b, p_0, h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4$ and T_0 as defined in table 4.2 and then calculate $\boldsymbol{p}/\boldsymbol{T}$ according to the functions in Eq. 4.28 and 4.3.

4.5.2 Posterior Distribution

Here, we define the marginal and then conditional posterior distribution for the described Bayesian model. We either use the `t-walk` algorithm [26] to draw samples from $\pi(p_0, b, \boldsymbol{h_T}, \boldsymbol{c_T}, \boldsymbol{a_T}, \delta, \gamma|\boldsymbol{y})$ or we utilise a TT approximation on a predefined grid to draw samples via the SIRT method with an MH correction step from that marginal posterior. We define $\boldsymbol{\theta_{p,T}} \coloneqq \{p_0, b, \boldsymbol{h_T}, \boldsymbol{c_T}, \boldsymbol{a_T}\}$, which includes all hyper-parameters related to pressure and temperature, for brevity. Lastly, we use the RTO Method to draw samples from the conditional $\pi(\boldsymbol{x}|p_0, b, \boldsymbol{h_T}, \boldsymbol{c_T}, \boldsymbol{a_T}, \delta, \gamma, \boldsymbol{y})$.

Marginal Posterior Distribution

The marginal posterior is given as

$$\pi(\lambda, \boldsymbol{\theta_{p,T}}, \gamma|\boldsymbol{y}) \propto \lambda^{n/2}\gamma^{m/2} \exp\left\{-\frac{1}{2}g(\lambda, \boldsymbol{\theta_{p,T}}) - \frac{\gamma}{2}f(\lambda, \boldsymbol{\theta_{p,T}})\right\}\pi(\lambda, \boldsymbol{\theta_{p,T}}, \gamma)\,, \tag{4.29}$$

with $\lambda = \delta/\gamma$,

$$f(\lambda, \boldsymbol{\theta_{p,T}}) = \boldsymbol{y}^T\boldsymbol{y} - \left(\boldsymbol{A}(\boldsymbol{\theta_{p,T}})^T\boldsymbol{y}\right)^T\left(\boldsymbol{A}(\boldsymbol{\theta_{p,T}})^T\boldsymbol{A}(\boldsymbol{\theta_{p,T}}) + \lambda\boldsymbol{L}\right)^{-1}\left(\boldsymbol{A}(\boldsymbol{\theta_{p,T}})^T\boldsymbol{y}\right), \tag{4.30a}$$

$$\text{and } g(\lambda, \boldsymbol{\theta_{p,T}}) = \log\det\left(\boldsymbol{A}(\boldsymbol{\theta_{p,T}})^T\boldsymbol{A}(\boldsymbol{\theta_{p,T}}) + \lambda\boldsymbol{L}\right). \tag{4.30b}$$

Here we use the approximated forward model $\boldsymbol{A}(\boldsymbol{\theta_{p,T}}) \coloneqq \boldsymbol{M}\boldsymbol{A}(\boldsymbol{\theta_{p,T}})_L$ and do not approximate f and g but calculate $\boldsymbol{A}(\boldsymbol{\theta_{p,T}})$ for each function evaluation directly.

Sampling from Marginal Posterior For a ground truth, we run the `t-walk` [26] algorithm on $\pi(\lambda, \boldsymbol{\theta_{p,T}}, \gamma|\boldsymbol{y})$ for $N = 1000 \times 1000$ steps with a burn-in period of $N_{\text{burn-in}} = 100 \times 1000$, since we expect a maximal IATC, through some pre-analysis, of around $\tau_{\text{int,max}} = 500$, see Tab. 4.2, to generate around 1000 independent samples from the posterior. We initialise the Python implementation of the `t-walk`[66] around the hyper-prior mean values and take a total number of $N' = N + N_{\text{burn-in}} = 1100000$ steps around 5 mins. We iteratively define the TT grid according to the results of the `t-walk`, and also use these boundaries for the hyper-parameters when running the `t-walk` (see Tab. 4.2). We plot the resulting histograms in Fig. 4.20 to 4.24 and the trace of the samples in Fig. B.7.

TT Approximation of Marginal Posterior The aim now is to approximate the square root of the marginal posterior

$$\pi(\lambda, \boldsymbol{\theta_{p,T}}, \gamma|\boldsymbol{y}) \propto \exp\Big\{\log\pi(\lambda, \boldsymbol{\theta_{p,T}}, \gamma|\boldsymbol{y})) + c\Big\}\,, \tag{4.31}$$

where we introduce a "normalisation constant" $c = -100$, similar to Sec. 4.2.2, to avoid under or overflow. In doing so we run the `tt.cross.rectcross.rect_cross.cross` function from the `ttpy` python package [60]. We compute the marginal as in Sec. 2.3, where we set $\xi = 1/\boldsymbol{\lambda}(\mathcal{X})$ and $\boldsymbol{\lambda}(x) = 1$ so that for Cartesian basis $\boldsymbol{M}_k = \text{diag}(\boldsymbol{\lambda}_k(\mathcal{X}_k))$. To draw samples from this TT approximation we use the SIRT-MH scheme as in Sec. **??**

Correlation structure First, we arrange the order of hyper-parameters according to their correlation structure to make the TT approximation more efficient. In doing so, highly correlated hyper-parameter pairs are next each other, so that their TT cores are direct neighbours and linked through their shared rank. See Fig.4.18, where we scatter plot the samples from the TT approximation of $\sqrt{\pi(p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a_T}|\boldsymbol{y}, \gamma, \boldsymbol{x})}$ via SIRT-MH scheme and plot the Pearson correlation coefficient. A coefficient close to 1 or -1 corresponds to high correlation, whereas low correlation between hyper-parameters has a coefficient close to zero. We observe that the λ hyper-parameters is highly correlated with b the hyper-parameter describing the pressure gradient and with γ. Additionally, $h_{T,1}$ and T_0 describing the temperature at low altitudes are correlated to b as well, since those do very slightly influence "the smoothness" of $\boldsymbol{p}/\boldsymbol{T}$, hard to see in Fig. 4.17. The IACTs in Tab. 4.2 agree with these results. Note that the correlation for parameters describing temperature in higher altitudes and surprisingly p_0 are very much uncorrelated. Next we aim to find the optimal rank and grid size to approximate $\sqrt{\pi(p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a_T}|\boldsymbol{y}, \gamma, \boldsymbol{x})}$, so that the number of function evaluation is as low as possible without too much sacrifice in accuracy.

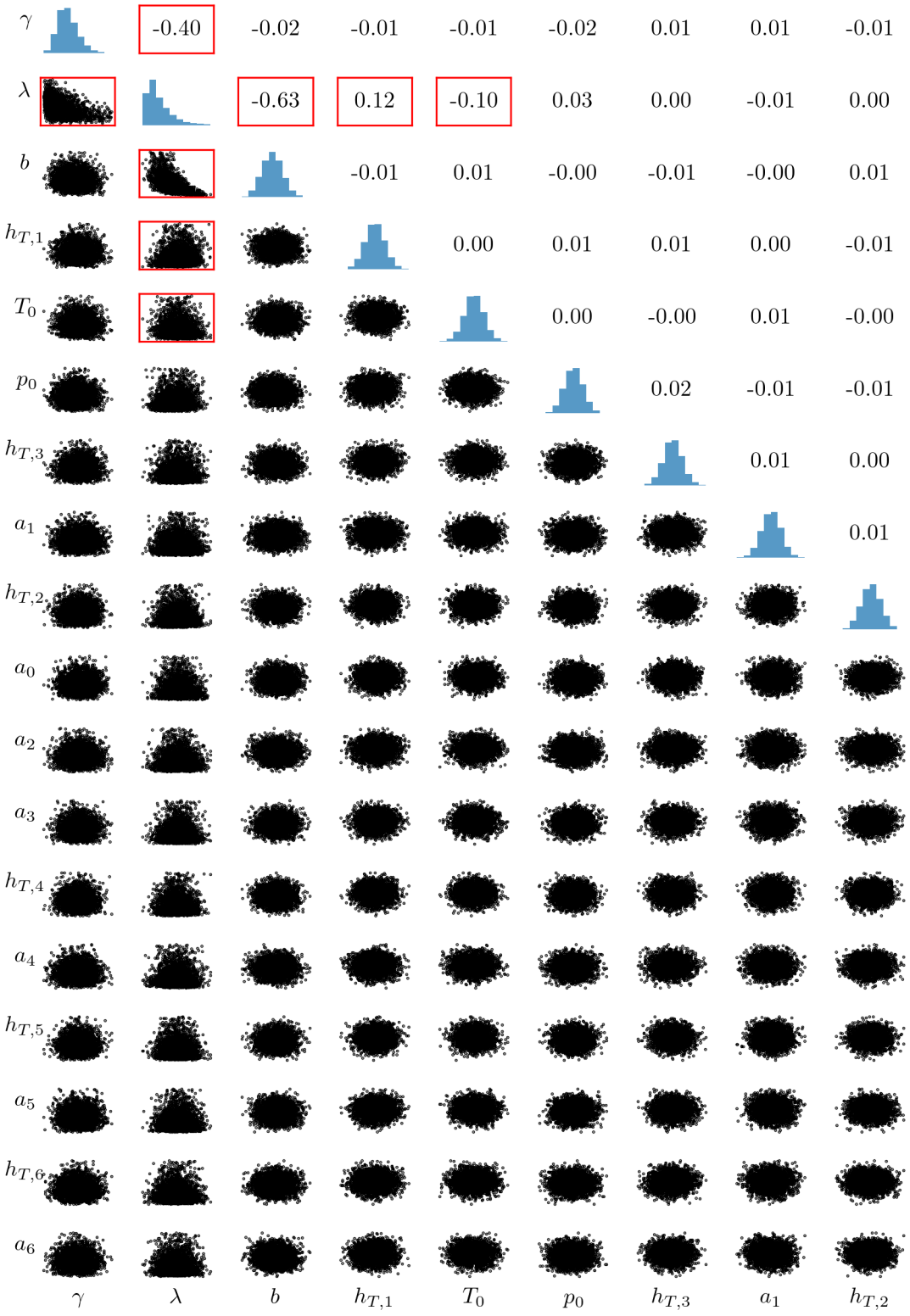
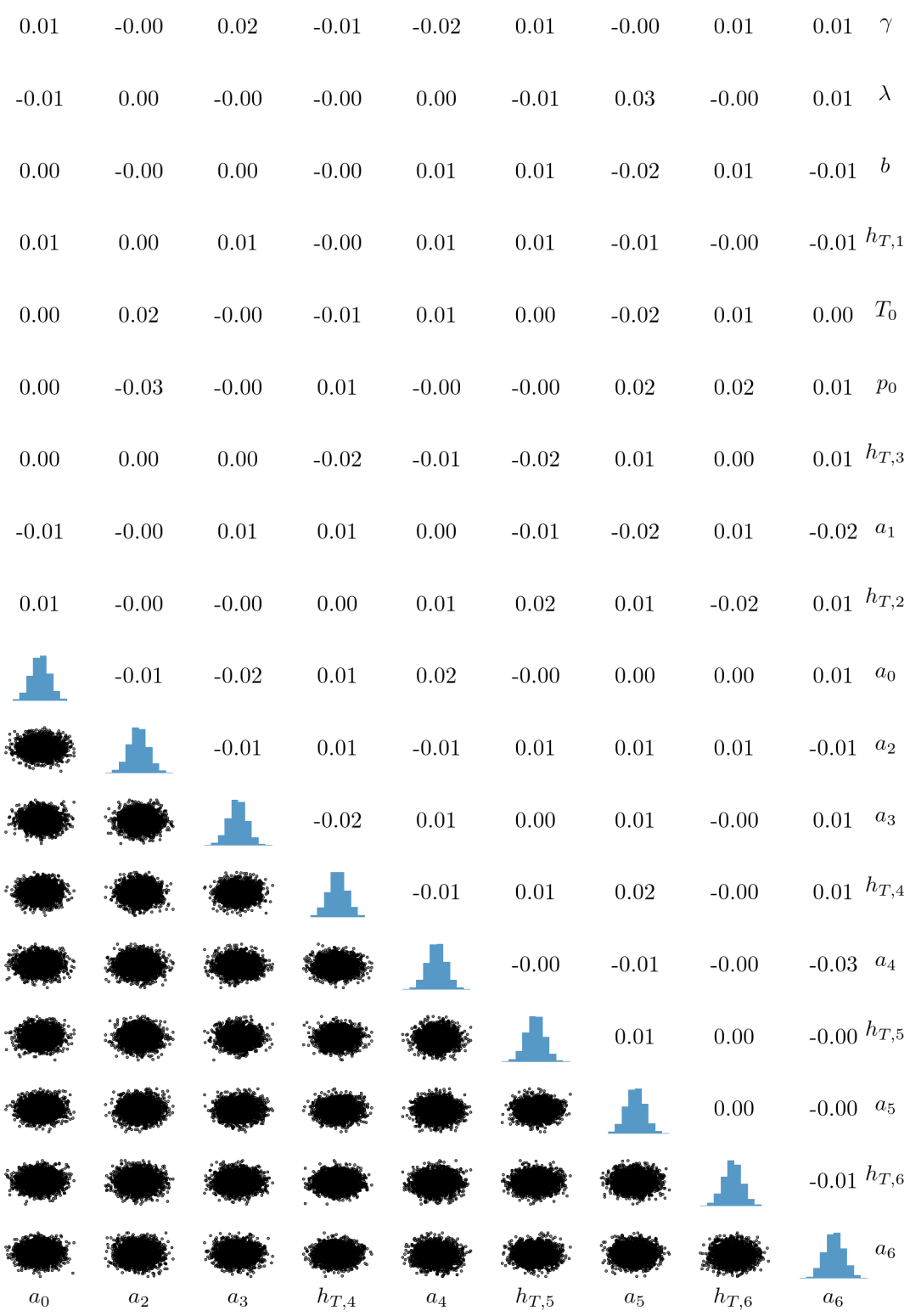


Figure 4.18: Scatter plot of samples from TT-approximation of $\sqrt{\pi(p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a_T}|\boldsymbol{y}, \gamma, \boldsymbol{x})}$ via SIRT scheme. We plot the Pearson correlation coefficient ranging from -1 to 1 for each hyper-parameter pair.



Correlation plot of samples from TT-approximation of $\sqrt{\pi(p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a_T}|\boldsymbol{y}, \gamma, \boldsymbol{x})}$ via SIRT scheme.

Find optimal Rank and Grid Size Next, we choose a relatively large number of grid points of $n = 150$ and calculate different error measures for deceasing number of ranks to find the optimal number of ranks. Then we fix a small but tolerable rank and decrease the number of grid points until sufficient accuracy.

We calculate the 1-Wasserstein distance, see Sec. 2.3.3, between 5000 independent samples from the SIRT-MH scheme, weighted with the TT approximation of marginal posterior, and 4101 independent samples from the `t-walk`, weighted by the true posterior value. To calculate the 1-Wasserstein distance, as in Eq. 2.52, we use the `SamplesLoss("sinkhorn", p=1, blur=0.05, scaling=0.8)` function with default settings from the python package `geomloss` [67]. This provides the unbiased Sinkhorn divergence which converges towards the Wasserstein distance and can be understood as the generalised Quicksort algorithm [38]. Here, p = 1 defines the distance measure $\|\boldsymbol{x} - \tilde{\boldsymbol{x}}\|_{L^2}$, the blur parameter can be understood as an entropic penalty and the scaling parameter specifies the trade-off between speed (scaling $< .4$) and accuracy (scaling $> .9$) [67]. Additionally we use the marginal functions of each TT approximation to calculate the weighted means $\boldsymbol{\mu}_{\mathrm{TT}} \in \mathbb{R}^{18}$ of each hyper-parameter and then the relative RMS deffierence $\|\boldsymbol{\mu}_{\mathrm{TT}} - \boldsymbol{\mu}_{\texttt{t-walk}}\|_{L^2} / \|\boldsymbol{\mu}_{\texttt{t-walk}}\|_{L^2}$ compared to "true means" $\boldsymbol{\mu}_{\texttt{t-walk}}$from the `t-walk`. Additionally, we compare the mean $\boldsymbol{\mu}_{\text{SIRT-MH}}$ based on the samples from the SIRT-MH scheme with the "true" `t-walk` mean. We plot all of these measures in Fig. 4.19 and decide that a rank $r = 10$ sufficient because the 1-Wasserstein distance is relatively stable for $r \geq 10$. We decide that a grid size $n = 40$ is large enough based in the relative differences of the samples based and integrated mean compared to the `t-walk` mean, which also seem to be relatively stable for $n \geq 40$.

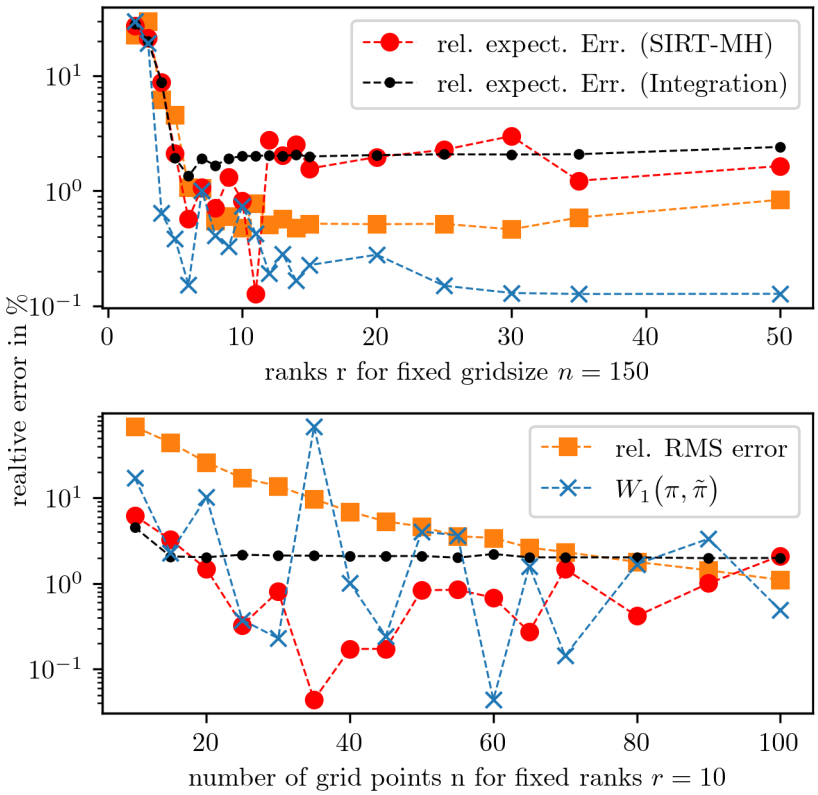


Figure 4.19: Given a TT approximation of $\sqrt{\pi(\lambda, \boldsymbol{\theta}_{\boldsymbol{p},\boldsymbol{T}}, \gamma|\boldsymbol{y})}$ we calculate the realtive RMS error and the 1-Wasserstein distance between approximated values at sample points provided by the SIRT-MH and the true function values. Further we use the marginal function from the TT approximation to calculate mean values of each hyper-parameter as weighted expectations and compare to the sample mean provided by the `t-walk`. Additionally we compare sample based mean from the SIRT-MH and the `t-walk`. We plot the relative RMS difference between those calculations. Note that the y-axis represents the recreative errors related to all measures but W_1. We do so since the each hyper-parameters has a different scale we are only interested in the trend of the W_1.

To decrease the number of functions evaluation and define ranks $r = [1, 10, 10, 10, 10, 10, 5, 5, 5, 5, 5, 3, 2,$ according to the correlation structure of $\pi(\lambda, \boldsymbol{\theta}_{\boldsymbol{p},\boldsymbol{T}}, \gamma|\boldsymbol{y})$ (see Fig. 4.18). We need one sweep in `tt.cross.rectcross.rect_cross.cross`, reducing the computation time to $\approx$ 0.5min and the number of functions evaluations to 48240. Note that we can initialise the `tt.cross.rectcross.rect_cross.cross` algorithm at a previously calculated approximation. We plot the marginals for each hyper-parameter in Fig. 4.20 to Fig. 4.25 and samples in Fig. 4.18 .

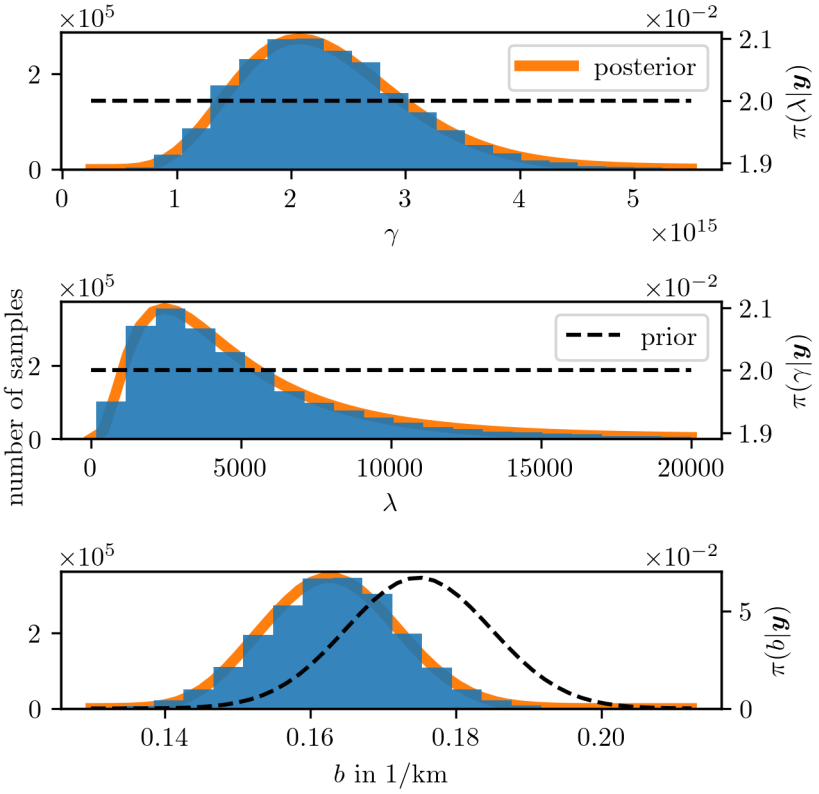


Figure 4.20: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

We observe that the marginal do not differ significantly from the `t-walk` samples and report a relative RMS error between true and TT approximated function values at SIRT-MH samples of $\approx 7\%$. We report IATC for the SIRT-MH around $2-4$ and the IATC given in Tab. 4.2. Additionally, we see when comparing the marginal posterior distributions to the prior distributions of the hyper-parameters that the hyper-parameter b is the only hyper-parameter related to pressure and temperature which is affected by the data.

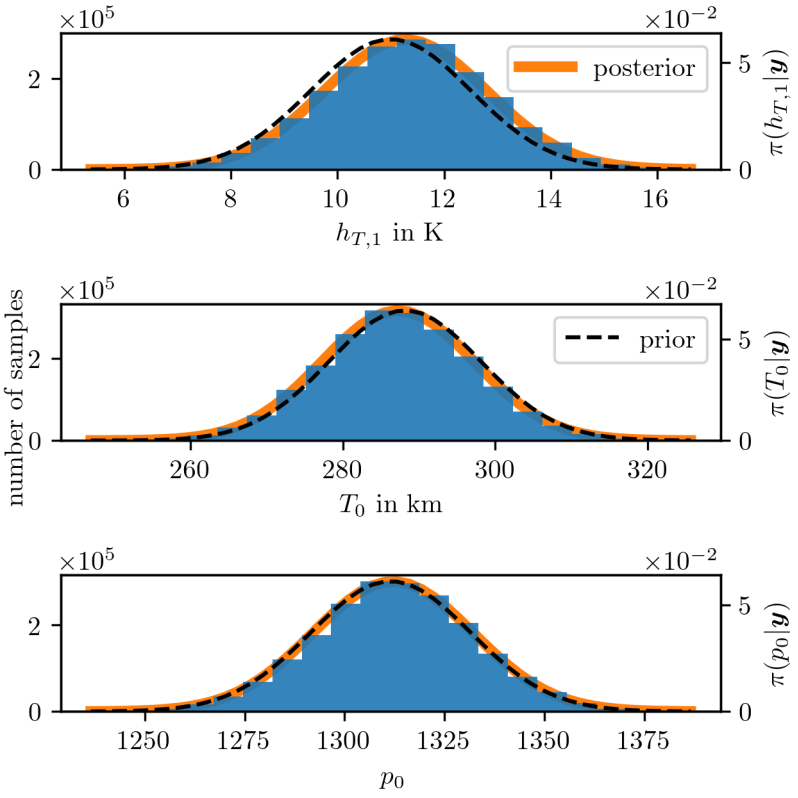


Figure 4.21: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

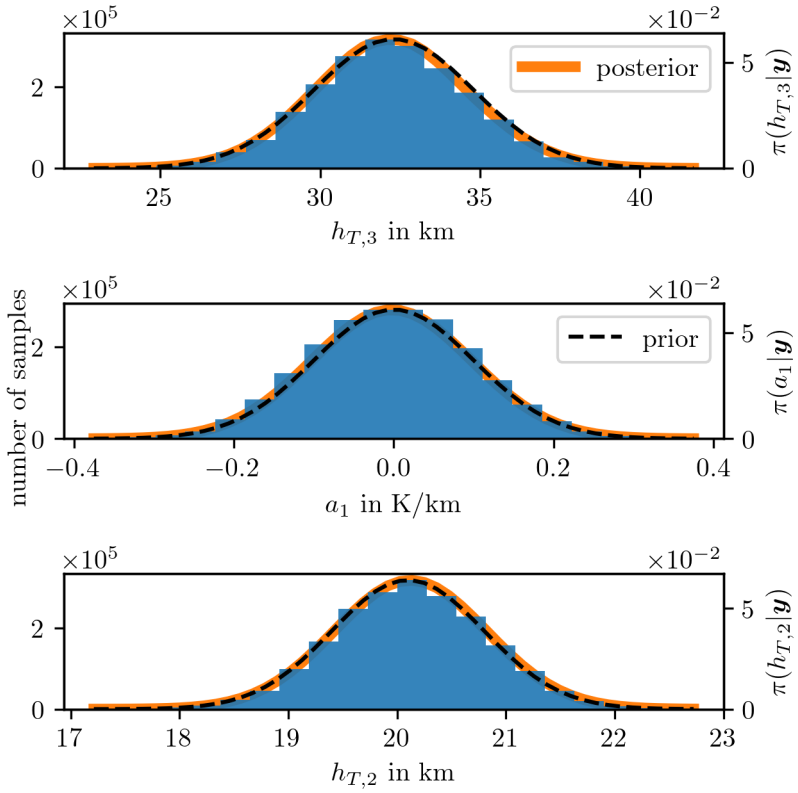


Figure 4.22: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

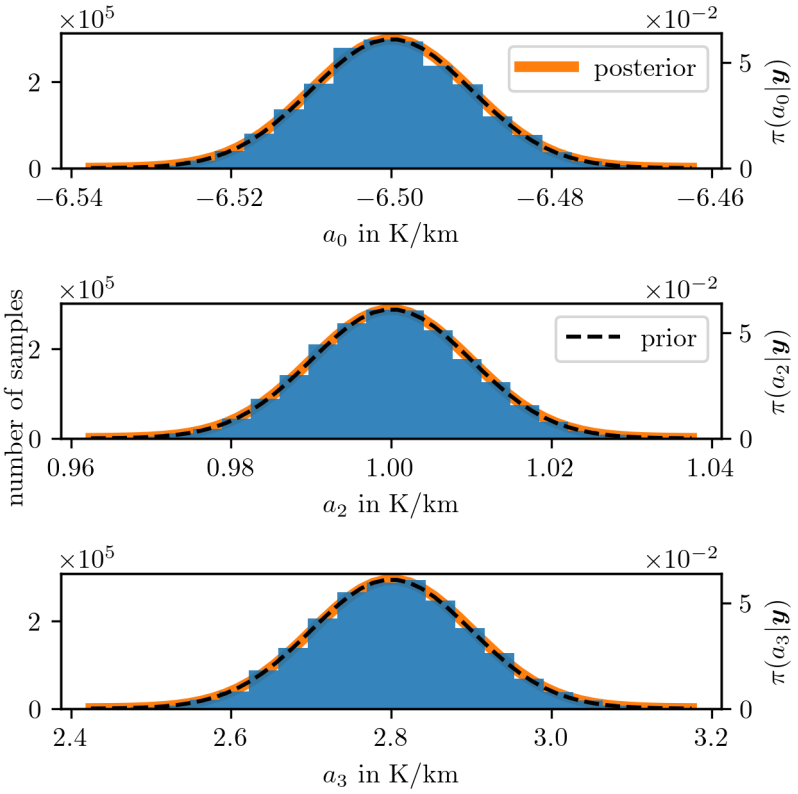


Figure 4.23: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

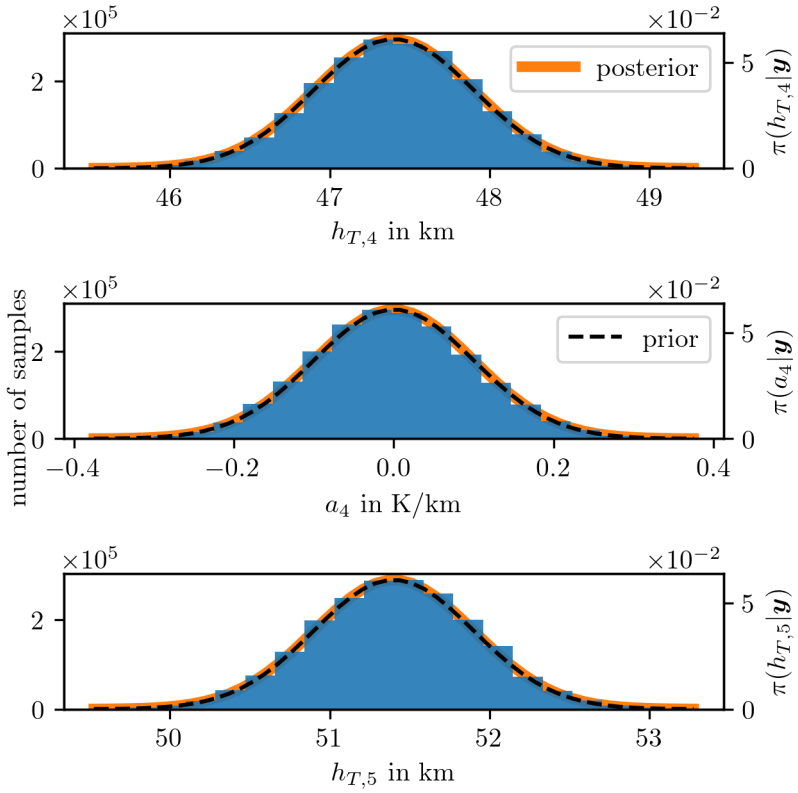


Figure 4.24: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

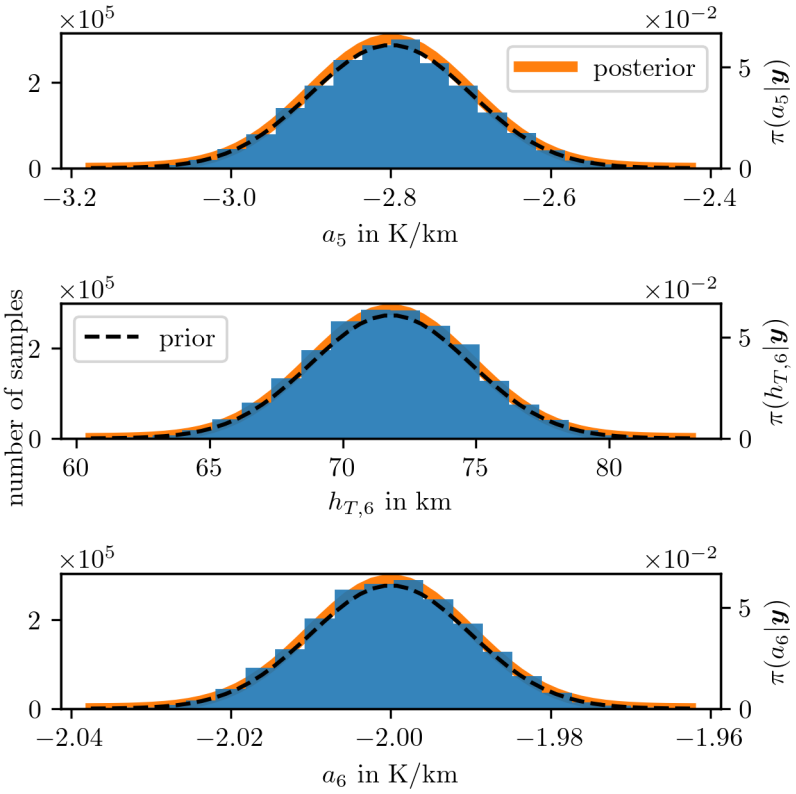


Figure 4.25: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

Conditional Posterior Distribution

We use the RTO method (see Sec. 2.2.3) to obtain ozone samples from the conditional posterior To obtain posterior pressure and temperature samples take hyper-parameter samples directly from the marginal posterior. Then we calculate pressure and temperature profile according to their respective function (see Eq. 4.28 and Eq. 4.3). We plot the posterior temperature and pressure profiles in Fig. 4.26 and Fig. 4.27 and posterior ozone in 4.28.

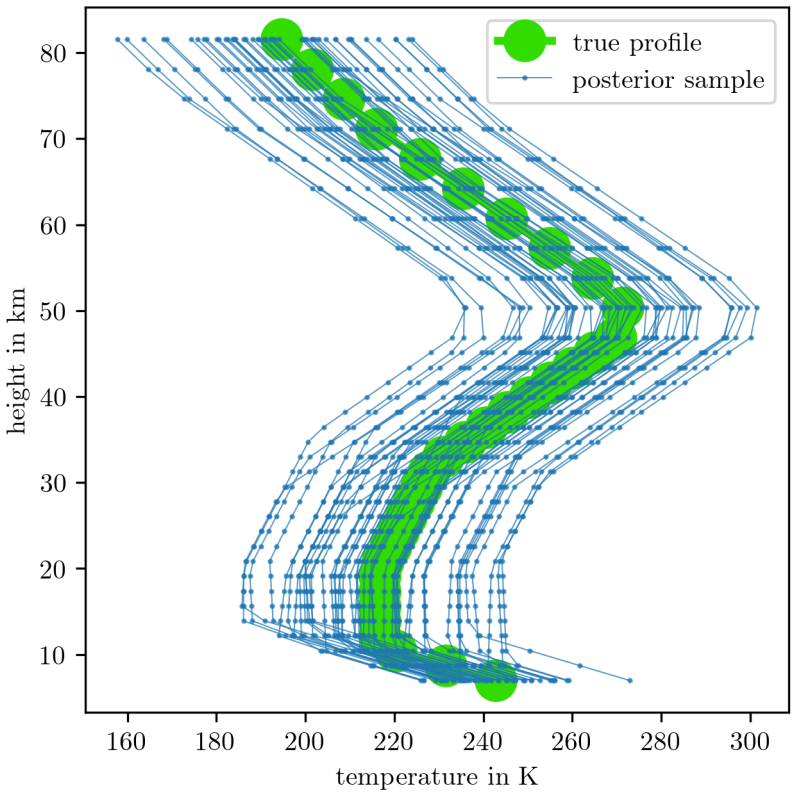


Figure 4.26: We take samples from the posterior distribution, as plotted in Figures 4.20 to 4.23 and plot the corresponding temperature function, see Eq: 4.3.

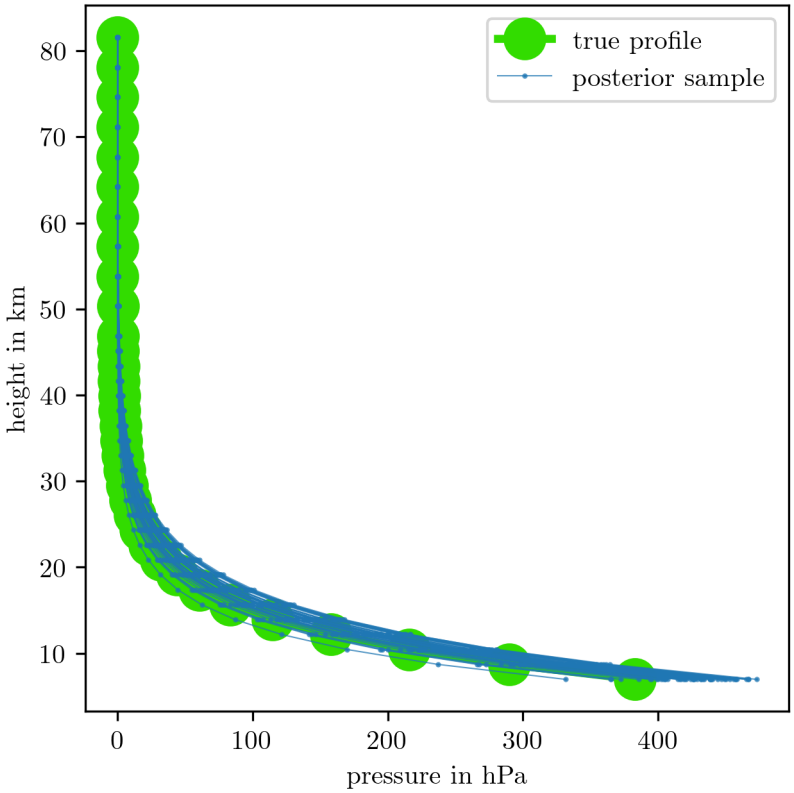


Figure 4.27: We take samples from the posterior distribution, as plotted in Fig. 4.24 and plot the corresponding pressure function, see Eq: 4.28.

We observe that pressure and ozone are highly correlated. Since the hyper-parameter b is smaller that the ground truth value the posterior pressure profile does not exponentially decrease as much compared the ground truth. This results in posterior pressure values which are slightly larger than the ground truth. In comparison, the corresponding posterior ozone profiles have much smaller individual ozone VMRs compared to the ground truth, but still a similar structure compared to the ground truth. Again we are not able to recover the peak at large altitude.

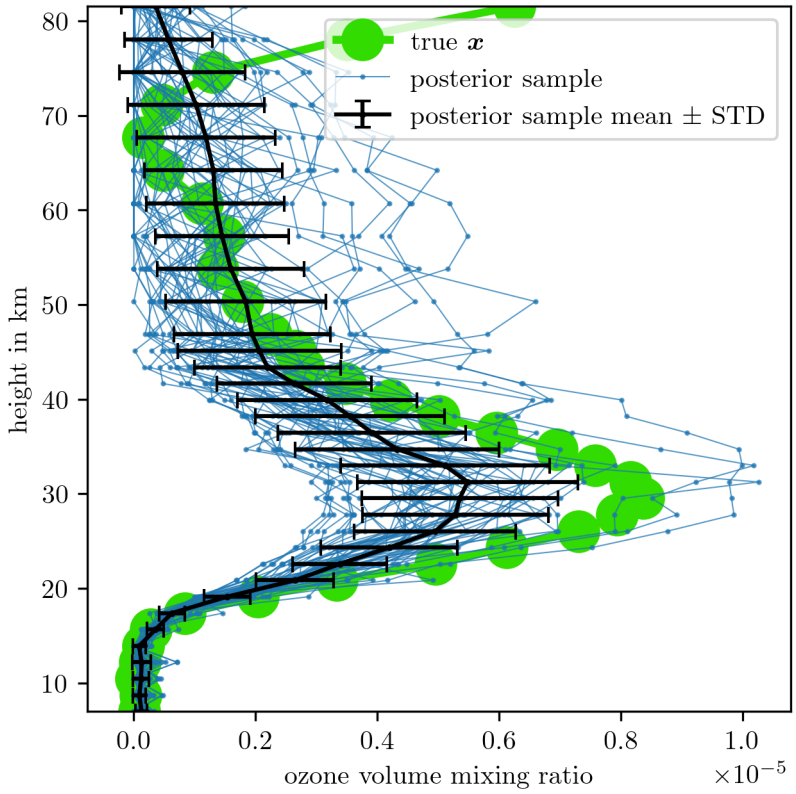


Figure 4.28: We take samples from the posterior distribution, as plotted in Fig. 4.24 and plot the corresponding pressure function, see Eq: 4.28.

5 Summary and Outlook

In this chapter we draw conclusion based on the results from the previous chapter. We compare the regularised solution to the mean and to the samples from the full posterior. We elaborate on the occurring approximation errors. We compare the marginal posterior distributions based on the drawn samples and from the TT-decomposition. While elaboration about the different methods, we also elaborate on how informative the data and what the means in terms of ozone, pressure and temperature profile.

5.1 Regularisation Solution vs. Hierarchical Bayesian Approach

200 functino ecalution of $\boldsymbol{x}_\lambda$ vs 10000 samples vs 600 funciton evalution of marg plus 20 $\boldsymbol{x}_\lambda$ and 20 $\boldsymbol{B}_\lambda^{-1}$ time similar within a second but no varicne As already mentioned the regularisation approach only provides on solution, see Fig. 4.11. In Fig. 4.13 we plot samples from the full conditional, which lay above L-Curve and make sense in terms of the Lagrange multipliers as the point on the L-Curve can be seen as extreme values. So the regularised estimate does not correlate to posterior solutions of the inverse problem. We note that the mean of full conditional is very similar to the regularised solution but is also some sort of an extreme value.

In comparison to the regularisation solution we can provide the mean and variance of the full conditional posterior distribution, as well as the sample mean.

5.2 Sampling Methods vs. TT Approximation

10000 samples vs 600 funciton evalution of marg similair time due approxiamtions of f and g the recto corss semms to be optimised than my code

We can conclude that the TT approximation is faster or as fast as sampling methods. For the marginal posterior $\pi(\gamma, \lambda|\boldsymbol{y})$ the calculation of the TT-cores takes 0.1s, which we consider similar to the sampling time of 0.5s. But the TT approximation needs less function evaluations than the MWG sampler. More precisely, the TT needs $n_{\text{tot}} = 2n_{\text{sweeps}}((d-2)r^2n + 2nr) = 400$ function evaluations, with number of sweeps $n_{\text{sweeps}} = 2$ and rank $r = 10$ and grid size $n = 25$ compared to 10000 samples. Error due to approcimation of f and g but also due TT

for PT once trained signifincaly faster but t walk is not the optimal sampler since it doesn consiger correaltopij structure Times

When approximating the posterior distribution of the temperature pressure ratio we are much faster compared to sampling methods. Since the parameter space is 16-dimensional we have to run the t-walk for about 2 million steps. In addition of checking the trace of the samples, we also estimate the IATC with [65] see Tab. 4.2. Since for shorter chains with a sample size of 10^6 the error for the IATC estimate is much larger we decide to a sample size of $4 \cdot 10^6$ is sufficient. This comes with a sampling time of 20mins, much larger compare to the 2.5min. Which makes sense as we need $n_{\text{tot}} = 2n_{\text{sweeps}}((d-2)r^2n + 2nr) = 384838438$ function evaluations. We also note that we do run into problems especially in higher dimensional functions as we have a large range of values and hence introduce the constant c as already mentioned. The t-walk is more robust but the TT approximation is faster.

But both the samples and the TT approximation point towards the same results.

Reduce correlation structure by rotating coordinate system Errors

Within the TT approximation we run into numerical problems. One way of solving this issue could be to use a different basis set such as Lagrange polynomials as these exactly fit to a Gaussian or Chebychev polynomial as basis functions. Another idea is to use different reference measure for integration, such as a Gaussian measure instead of the current Lebesgue measure. Or that the TT finds normalisation constants automatically.

The t-walk is a robust easy to implement sampling method, of course one could employ a more efficient sampler such as a gibbs sampler or something similar [].

We consider the approximation errors of the functions $f(\lambda)$, $g(\lambda)$ and propagation error into the marginal posterior for sampling of about 10% good enough. The TT approximation error from the marginal posterior is with about 10% also good enough since we do not believe that our model is accurate enough to capture those differences.

When approximation the affine map we get an relative error of about 0.4%, which is much smaller than the relative difference in between noise free and noisy data of approximately of 1.7%. We like to note that the relative difference The error linear to non-linear. Low rank bound as in [68]

5.3 Atmospheric Physics

Here we want to say how informative the data is and what we can about the ozone pressure and temperature profiles.

So all the samples as in Fig. 4.11 and Fig. 4.6, present valid solutions to the inverse problem. Hence, we can see that the variability of ozone in the upper atmosphere is large and that we do not capture the ozone peak around 80km. The posterior temperature profiles is similar to the prior profiles, as also seen in marginal posterior Fig. 4.20 to 4.24. We can already see that in the prior analys, as the pressure temperature ratio does inherit the exponential structure of the pressure profile. So the posterior pressure profile is much more informative, see marginal for b in Fig. 4.24. So we can retrieve an informative pressure profile for the pressure but not for temperature.

Ideally we should do this iteratively update ozone and then temperature and pressure until proven convergence.

We coudl fix that by choosing a more restirctve prio rfor rpessure but that pwould not be objective INstead we should really work on a better mode kf or ozxone. We can get rid of the pressure skew at a SNR of $\approx$ 10000

5.3.1 Measurement Device

Then we can include more measurement specific details such as the pointing accuracy. Then we could sample measurement N_Γ geometries $\Gamma^{(k)} \sim \pi(\Gamma)$ so that the posterior $\pi(\boldsymbol{x}|\boldsymbol{y}) \approx 1/N_\Gamma \sum_\Gamma \pi(\boldsymbol{x}, \Gamma^{(k)}|\boldsymbol{y})$ and include other measurement device specific parameters.

5.3.2 Model

Since we have to truncate the full conditional at the end the model is not accurate enough to eliminate those values. This was to show that we can to a more comprehensive analysis compared to a regularised method. Ideally we like to use a more accurate model where we parametrise ozone, similar to the pressure and temperature profile. In doing so one would has to know much more about ozone in different altitudes. Then we possibly could employ a different graph Laplacian based on a different structure of ozone. And when we approximate the non-linear forward map with a affine map using a linear solver we could of course use other methods such as the machine learning methods.

References

[1] Bryan Duncan. *Aura at 20 Years*. https://science.nasa.gov/science-research/earth-science/aura-at-20-years/. [Online; accessed 31/08/25]. NASA's Goddard Space Flight Center (GSFC), 2024.

[2] Susan L Ustin and Elizabeth McPhee Middleton. "Current and near-term Earth-observing environmental satellites, their missions, characteristics, instruments, and applications". In: *Sensors* 24.11 (2024), p. 3488.

[3] Mallika Irene Suresh et al. "Multichannel upconversion of terahertz radiation in an optical disk resonator". In: *Opt. Express* 33.5 (2025), pp. 10302–10311.

[4] Florian Sedlmeir et al. "Detecting THz in the telecom range: All resonant THz up-conversion in a whispering gallery mode resonator". In: *2014 39th International Conference on Infrared, Millimeter, and Terahertz waves (IRMMW-THz)*. 2014, pp. 1–2.

[5] Clive D Rodgers. "Retrieval of atmospheric temperature and composition from remote measurements of thermal radiation". In: *Reviews of Geophysics* 14.4 (1976), pp. 609–624.

[6] Nathaniel J. Livesey et al. *Earth Observing System (EOS) Microwave Limb Sounder (MLS) Version 5.0x Level 2 and 3 data quality and description document*. Version 5.0-1.1a. NASA Goddard Earth Sciences Data and Information Services Center, 2022.

[7] Sze M Tan, Colin Fox, and Geoff K. Nicholls. *Course notes for ELEC 445 – Inverse Problems and Imaging*. https://coursesupport.physics.otago.ac.nz/wiki/pmwiki.php/ELEC445/HomePage. [Online; accessed 10/12/23]. Physics Department, University of Otago, 2016.

[8] Nathaniel J Livesey et al. "Retrieval algorithms for the EOS Microwave limb sounder (MLS)". In: *IEEE Transactions on Geoscience and Remote Sensing* 44.5 (2006), pp. 1144–1155.

[9] Schwartz M. et al. *MLS/Aura Level 2 Ozone (O3) Mixing Ratio V005*. https://disc.gsfc.nasa.gov/datasets/ML2O3_005/summary?keywords=mls%20o3. [Online; accessed 25/04/24]. NASA Goddard Earth Sciences Data and Information Services Center, 2020.

[10] N. J. Livesey et al. "Validation of Aura Microwave Limb Sounder O3 and CO observations in the upper troposphere and lower stratosphere". In: *Journal of Geophysical Research: Atmospheres* 113.D15 (2008).

[11] L. Froidevaux et al. "Validation of Aura Microwave Limb Sounder stratospheric ozone measurements". In: *Journal of Geophysical Research: Atmospheres* 113.D15 (2008).

[12] F. Werner et al. "Applying machine learning to improve the near-real-time products of the Aura Microwave Limb Sounder". In: *Atmospheric Measurement Techniques* 16.11 (2023), pp. 2733–2751.

[13] Pasquale Sellitto et al. "Neural network algorithms for ozone profile retrieval from ESA-envisat sciamachy and NASA-AURA OMI satellite data". In: vol. 3. Aug. 2008, pp. III –170.

[14] Colin Fox and Richard A Norton. "Fast sampling in a linear-Gaussian inverse problem". In: *SIAM/ASA Journal on Uncertainty Quantification* 4.1 (2016), pp. 1191–1218.

[15] Gareth O. Roberts and Jeffrey S Rosenthal. “General state space Markov chains and MCMC algorithms”. In: *Probability Surveys* 1 (2004), pp. 20–71.

[16] S. P. Meyn and R.L. Tweedie. *Markov Chains and Stochastic Stability. 2nd Edition*. New York: Cambridge University Press, 2009.

[17] Havard Rue and Leonhard Held. *Gaussian Markov random fields: theory and applications*. London: CRC press, 2005.

[18] Richard A Norton, J Andrés Christen, and Colin Fox. “Sampling hyperparameters in hierarchical models: improving on Gibbs for high-dimensional latent fields and large datasets”. In: *Communications in Statistics-Simulation and Computation* 47.9 (2018), pp. 2639–2655.

[19] Charles W. Champ and Andrew V. Sills. “The Generalized Law of Total Covariance”. In: *preprint* (2022).

[20] Charles J Geyer. “Practical markov chain monte carlo”. In: *Statistical science* (1992), pp. 473–483.

[21] U. Wolff and B. Bunk. *Lecture Notes on Computational Physics II [in german]*. `www-com.physik.hu-berlin.de/comphys/comphys.htm`. [Online; accessed 29/08/25]. Humboldt University, Berlin, 2016.

[22] A. Sokal. “Monte Carlo Methods in Statistical Mechanics: Foundations and New Algorithms”. In: *Functional Integration: Basics and Applications*. Ed. by Cecile DeWitt-Morette, Pierre Cartier, and Antoine Folacci. Boston, MA: Springer US, 1997, pp. 131–192.

[23] Ulli Wolff. “Monte Carlo errors with less errors”. In: *Computer Physics Communications* 156.2 (2004), pp. 143–153.

[24] Daniel Simpson, Finn Lindgren, and Håvard Rue. “Think continuous: Markovian Gaussian models in spatial statistics”. In: *Spatial Statistics* 1 (2012), pp. 16–29.

[25] Gareth O. Roberts and Jeffrey S Rosenthal. “Harris recurrence of Metropolis-within-Gibbs and trans-dimensional Markov chains”. In: *The Annals of Applied Probability* 16.4 (2006), 2123–2139.

[26] J. Andrés Christen and Colin Fox. “A general purpose sampling algorithm for continuous distributions (the t-walk)”. In: *Bayesian Analysis* 5.2 (2010), pp. 263 –281.

[27] Johnathan M Bardsley. “MCMC-based image reconstruction with uncertainty quantification”. In: *SIAM Journal on Scientific Computing* 34.3 (2012), A1316–A1332.

[28] Johnathan M Bardsley et al. “Randomize-then-optimize: A method for sampling from posterior distributions in nonlinear inverse problems”. In: *SIAM Journal on Scientific Computing* 36.4 (2014), A1895–A1910.

[29] Dootika Vats et al. “Understanding Linchpin Variables in Markov Chain Monte Carlo”. In: *preprint* (2022). URL: `https://arxiv.org/pdf/2210.13574`.

[30] Tiangang Cui and Sergey Dolgov. “Deep composition of tensor-trains using squared inverse rosenblatt transports”. In: *Foundations of Computational Mathematics* 22.6 (2022), pp. 1863–1922.

[31] Philip J Davis and Philip Rabinowitz. *Methods of numerical integration*. San Diego, CA: Academic Press, Inc., 1984.

[32] Sergey Dolgov et al. “Approximation and sampling of multivariate probability distributions in the tensor train decomposition”. In: *Statistics and Computing* 30 (2020), pp. 603–625.

[33] Colin Fox et al. “Grid methods for Bayes-optimal continuous-discrete filtering and utilizing a functional tensor train representation”. In: *Inverse Problems in Science and Engineering* 29.8 (2021), pp. 1199–1217.

[34] Ivan V Oseledets. "Tensor-train decomposition". In: *SIAM Journal on Scientific Computing* 33.5 (2011), pp. 2295–2317.

[35] Ivan Oseledets. "DMRG Approach to Fast Linear Algebra in the TT-Format". In: *Computational Methods in Applied Mathematics* 11.3 (2011), pp. 382–393.

[36] John Thickstun. *Kantorovich-rubinstein duality*. `https://courses.cs.washington.edu/courses/cse599i/20au/resources/L12_duality.pdf`. [Online; accessed 31/08/25]. University of Washington, 2019.

[37] Luigi Ambrosio, Elia Brué, and Daniele Semola. *Lectures on Optimal Transport*. Cham: Springer Nature Switzerland, 2024.

[38] Jean Feydy. "Analyse de données géométriques, au delà des convolutions". Thesis. Université Paris-Saclay, July 2020. URL: `https://theses.hal.science/tel-02945979`.

[39] Marcel Berger. *Geometry I. 4th Edition*. Berlin Heidelberg: Springer-Verlag, 2009.

[40] Katsumi Nomizu and Takeshi Sasaki. *Affine differential geometry*. Cambridge: Cambridge University Press, 1994.

[41] Per Christian Hansen. "Regularization, GSVD and truncated GSVD". In: *BIT numerical mathematics* 29.3 (1989), pp. 491–504.

[42] Per Christian Hansen. "The L-Curve and its Use in the Numerical Treatment of Inverse Problems". English. In: *Computational Inverse Problems in Electrocardiology*. Ed. by P. Johnston. WIT Press, 2001, pp. 119–142.

[43] Yu-Xiang Wang et al. "Trend Filtering on Graphs". In: *Journal of Machine Learning Research* 17.105 (2016), pp. 1–41.

[44] Per Christian Hansen and Dianne Prost O'Leary. "The use of the L-curve in the regularization of discrete ill-posed problems". In: *SIAM Journal on Scientific Computing* 14.6 (1993), pp. 1487–1503.

[45] KC Santosh, Nibaran Das, and Swarnendu Ghosh. "Chapter 3 - Deep learning models". In: *Deep Learning Models for Medical Imaging*. Ed. by KC Santosh, Nibaran Das, and Swarnendu Ghosh. Primers in Biomedical Imaging Devices and Systems. Academic Press, 2022, pp. 65–97.

[46] C. Readings. *Envisat MIPAS An Instrument for Atmospheric Chemistry and Climate Research*. Noordwijk: ESA Publications Division, 2000.

[47] Jae N. Lee and Dong L. Wu. "Solar Cycle Modulation of Nighttime Ozone Near the Mesopause as Observed by MLS". In: *Earth and Space Science* 7.4 (2020).

[48] Iouli E Gordon et al. "The HITRAN2020 molecular spectroscopic database". In: *Journal of Quantitative Spectroscopy and Radiative Transfer* 277 (2022), p. 107949.

[49] Marie Šimečková et al. "Einstein A-coefficients and statistical weights for molecular absorption transitions in the HITRAN database". In: *Journal of Quantitative Spectroscopy and Radiative Transfer* 98.1 (2006), pp. 130–155.

[50] George B. Rybicki and Alan P. Lightman. *Radiative processes in Astrophysics*. Weinheim: Wiley-VCH, 2004.

[51] W.G. Read et al. "The clear-sky unpolarized forward model for the EOS aura microwave limb sounder (MLS)". In: *IEEE Transactions on Geoscience and Remote Sensing* 44.5 (2006), pp. 1367–1379.

[52] Colin Fox. *Blokkurs on computing MCMC for inverse problems*. unpublished. Physics Department, University of Otago, 2025.

[53] Australian National Concurrent Design Facility. *CubeSat Microwave Radiometer Mission to Support Global Ozone Layer Monitoring. Concept Study - Summary Report*. unpublished, internal report. Canberra BC: UNSW Canberra Space, 2023.

[54] Joe W Waters et al. “The earth observing system microwave limb sounder (EOS MLS) on the Aura satellite”. In: *IEEE Transactions on Geoscience and Remote Sensing* 44.5 (2006), pp. 1075–1092.

[55] *U.S. Standard Atmosphere, 1976*. Washington, D.C.: United States. National Oceanic and Atmospheric Administration, United States Committee on Extension to the Standard Atmosphere, 1976.

[56] Gareth Roberts. *ST911 Fundamentals of Statistical Inference - Part III*. Department of Statistics, University of Warwick, 2015.

[57] Drik Hesse. *py-uwerr; Python implementation of Monte Carlo error analysis a la Wolff.* `https://github.com/dhesse/py-uwerr`. [Online; accessed 09/09/25].

[58] Ivan Oseledets and Eugene Tyrtyshnikov. “TT-cross approximation for multidimensional arrays”. In: *Linear Algebra and its Applications* 432.1 (2010), pp. 70–88.

[59] Sergey Dolgov and Robert Scheichl. “A Hybrid Alternating Least Squares–TT-Cross Algorithm for Parametric PDEs”. In: *SIAM/ASA Journal on Uncertainty Quantification* 7.1 (2019), pp. 260–291.

[60] I. V. Oseledets et al. *ttpy - a Python implementation of the TT-toolbox.* `https://github.com/oseledets/ttpy`. [accessed 23/06/25]. 2018.

[61] Josef Dick, Frances Y. Kuo, and Ian H. Sloan. “High-dimensional integration: The quasi-Monte Carlo way”. In: *Acta Numerica* 22 (2013), 133–288.

[62] Johnathan M Bardsley et al. “Randomize-then-optimize for sampling and uncertainty quantification in electrical impedance tomography”. In: *SIAM/ASA Journal on Uncertainty Quantification* 3.1 (2015), pp. 1136–1158.

[63] Per Christian Hansen. *Discrete Inverse Problems: Insight and Algorithms*. Philadelphia: SIAM, 2010.

[64] Ville Satopää et al. “Finding a "Kneedle" in a Haystack: Detecting Knee Points in System Behavior”. In: *2011 31st International Conference on Distributed Computing Systems Workshops*. IEEE. 2011, pp. 166–171.

[65] Ulli Wolff. *UWerr.m Version6*. `https://www.physik.hu-berlin.de/de/com/ALPHAsoft`. [Online; accessed 5/11/23]. Humboldt-Universität to Berlin, 2004.

[66] J. Andrés Christen and Colin Fox. *The t-walk software.* `https://www.cimat.mx/~jac/twalk/`. [Online; accessed 25/11/24]. CIMAT, Mexico, and University of Otago, New Zealand.

[67] Jean Feydy. *GeomLoss – Geometric Loss functions between sampled measures, images and volumes*. `https://www.kernel-operations.io/geomloss/api/pytorch-api.html`. [Online; accessed 12/09/25].

[68] Paul B. Rohrbach et al. “Rank Bounds for Approximating Gaussian Densities in the Tensor-Train Format”. In: *SIAM/ASA Journal on Uncertainty Quantification* 10.3 (2022), pp. 1191–1224.

[69] M. Capiński and P.E. Kopp. *Measure, Integral and Probability. Springer Undergraduate Mathematics Series*. London: Springer-Verlag London, 2004.

[70] M. Simonnet. *Measures and Probabilities*. New York: Springer-Verlag, 1996.

[71] Vesa Kaarnioja. *Inverse Problems. Eighth lecture.* `https://vesak90.userpage.fu-berlin.de/ip23/week8.pdf`. [Online; accessed 10/04/25]. Freie Universität Berlin, Department of Mathematics and Computer Science, 2023.

[72] Greg Lawler. *Notes on probability.* `https://www.math.uchicago.edu/~lawler/probnotes.pdf`. [Online; accessed 10/04/25]. The University of Chicago, Department of Mathematics, 2016.

Appendices

A

Theoretical and technical background

A.1 Correlation Structure

In the book Gaussian Markov Random Fields [17], Rue and Held demonstrate that a strong correlation between the hyper-parameter μ and the latent field $\boldsymbol{x}$ can significantly slow down convergence when using samplers, in particularly Gibbs samplers. They consider the hierarchical model

$$\mu \sim \mathcal{N}(0, 1) \tag{A.1a}$$

$$\boldsymbol{x}|\mu \sim \mathcal{N}(\mu\mathbf{1}, \boldsymbol{Q}^{-1}), \tag{A.1b}$$

and apply a Gibbs sampler based on the full conditional distributions

$$\mu^{(k)}|\boldsymbol{x}^{(k)} \sim \mathcal{N}\left(\frac{\mathbf{1}^T\boldsymbol{Q}\boldsymbol{x}^{(k-1)}}{1+\mathbf{1}^T\boldsymbol{Q}\mathbf{1}}, \left(1+\mathbf{1}^T\boldsymbol{Q}\mathbf{1}\right)^{-1}\right) \tag{A.2}$$

$$\boldsymbol{x}^{(k)}|\mu^{(k)} \sim \mathcal{N}(\mu^{(k)}\mathbf{1}, \boldsymbol{Q}^{-1}). \tag{A.3}$$

As illustrated in Figure A.1, when the sampler is restricted to steps only in the μ-direction (horizontal axis) or the $\boldsymbol{x}$-direction (vertical axis), it requires many iterations to adequately explore the parameter space. This inefficiency arises from the high correlation between μ and $\boldsymbol{x}$, visible in Figure A.1 as a 'squeeze' of the distribution.

A solution to the slow mixing problem is to update $(\mu, \boldsymbol{x})$ jointly. Since here μ is one dimensional, effectively only marginal density of μ is needed.

$$\mu^\star \sim q(\mu^\star|\mu^{(k-1)}) \tag{A.4}$$

$$\boldsymbol{x}^{(k)}|\mu^\star \sim \mathcal{N}(\mu^\star\mathbf{1}, \boldsymbol{Q}^{-1}) \tag{A.5}$$

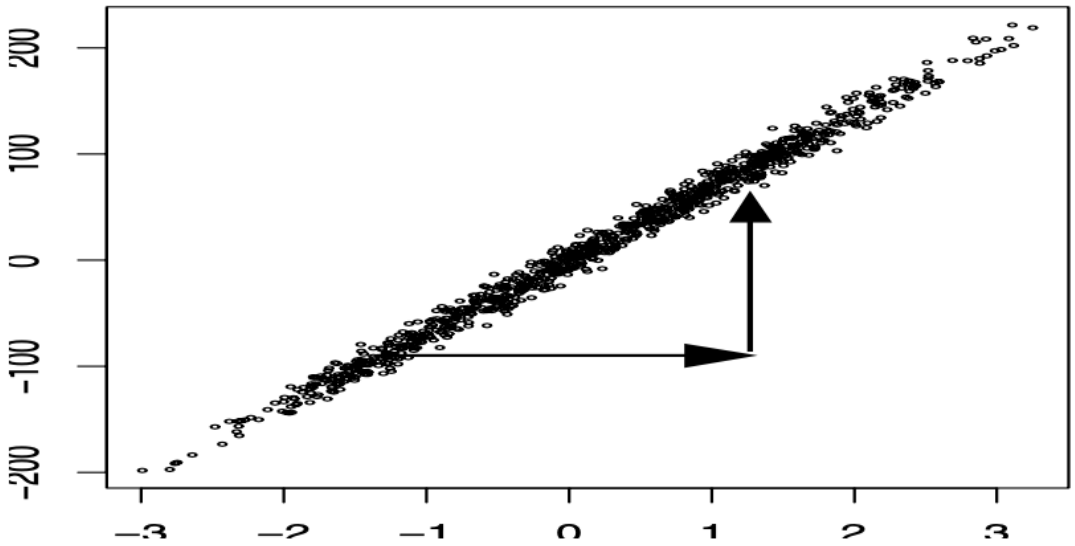


Figure A.1: The figure taken from [17, Figure 4.1 (b)], shows samples from a marginal chain for μ and $\mathbf{1}^T\boldsymbol{Q}\boldsymbol{x}^{(k)}$ over 1000 iterations, based on the hierarchical model in Eq. A.1, with an autoregressive process encoded in $\boldsymbol{Q}$. The algorithm updates μ and $\boldsymbol{x}$ successively from their full conditional distributions. The plot displays $(\mu^{(k)}, \mathbf{1}^T\boldsymbol{Q}\boldsymbol{x}^{(k)})$, with $\mu^{(k)}$ on the horizontal axis and $\mathbf{1}^T\boldsymbol{Q}\boldsymbol{x}^{(k)}$ on the vertical axis. The slow mixing and convergence of μ result from its strong dependence on $\mathbf{1}^T\boldsymbol{Q}\boldsymbol{x}^{(k)}$, while the sampler permits only axis-aligned (horizontal and vertical) and does not allow diagonal moves, as illustrated by the arrows.

With a simple MCMC algorithm targeting μ one can explore the sample space efficiently and only draw a corresponding sample for $\boldsymbol{x}$ from its full conditional once, for instance, the proposal $\mu^\star$ has been accepted.

A.2 On the Monte-Carlo Error and Integrated Autocorrelation time

To assess the error σ^2 of chain $\mathcal{M}_i$, we ignore systematic error due to initialisation bias (burn-in period), but we have to take into account that samples produced by any system or algorithm are correlated. To derive the integrated autocorrelation time (IATC), we follow the lecture notes [21]. In general, the error of a Monte-Carlo-based estimate from a sample set $\mathcal{M}_i = \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(k)}, \ldots, \boldsymbol{x}^{(s)}, \ldots, \boldsymbol{x}^{(N)}\} \sim \pi(\boldsymbol{x}|\boldsymbol{y})$ is:

$$(\sigma^{(i)})^2 = \mathrm{var}(\boldsymbol{\mu}^{(i)}_{\mathrm{samp}}) = \mathrm{var}(\mathrm{E}_{\boldsymbol{x}|\boldsymbol{y}}[h(\boldsymbol{x})]) = \left(\frac{1}{N}\sum_{k=1}^{N} h(\boldsymbol{x}^{(k)}) - \boldsymbol{\mu}^{(i)}\right)^2. \tag{A.6}$$

Expanding this summation, we see that

$$(\sigma^{(i)})^2 = \frac{1}{N^2}\sum_{k,s=1}^{N} \Gamma(k-s) \tag{A.7}$$

with the auto correlation coefficient $\Gamma(k-s) = (h(\boldsymbol{x}^{(k)}) - \boldsymbol{\mu}^{(i)})(h(\boldsymbol{x}^{(s)}) - \boldsymbol{\mu}^{(i)})$. Next we rewrite

$$\sum_{k,s=1}^{N} \Gamma(k-s) = \mathrm{var}(\boldsymbol{\mu}_{\mathrm{samp}}^{(i)}) \sum_{k,s=1}^{N} \frac{\Gamma(k-s)}{\Gamma(0)} = \mathrm{var}(\boldsymbol{\mu}_{\mathrm{samp}}^{(i)}) \sum_{k,s=1}^{N} \rho(k-s)\,, \tag{A.8}$$

with the normalised auto correlation coefficient $\rho(k-s) = \Gamma(k-s)/\Gamma(0)$ at lag $k-s$ and $\Gamma(0) = \mathrm{var}(\boldsymbol{\mu}_{\mathrm{samp}}^{(i)})$ for $k = s$. Typically $\Gamma(t)$ decays exponentially so that, for $N \gg \tau$, $\Gamma(t) \overset{t\to\infty}{\propto} \exp\{-|t|/\tau\}$ and we can approximate

$$\sum_{k,s=1}^{N} \rho(k-s) = N \sum_{t=-(N-1)}^{N-1} \left(1 - \frac{t}{N}\right)\rho(t) \approx N \sum_{t=-\infty}^{\infty} \rho(t) \coloneqq 2N\tau_{\mathrm{int}}\,, \tag{A.9}$$

see [22], and define the IATC as in [21, pp. 103-105]. If $\tau \gg 1$ one can show that $\tau_{\mathrm{int}} \approx \tau$

$$\sum_{t=-\infty}^{\infty} \rho(t) = 1 + 2\sum_{t=1}^{\infty} \left(e^{-1/\tau}\right)^t = 1 + 2\frac{e^{-1/\tau}}{1-e^{-1/\tau}} \approx 1 + 2\frac{1-1/\tau}{1/\tau} = 2\tau - 1 \approx 2\tau_{int} \tag{A.10}$$

where we use the geometric power series $\sum_{n=0}^{\infty} x^n = 1/(1+x)$ and the Taylor series $e^x \approx 1 + x$ for small x. Consequently, the estimate for the Monte-Carlo error is:

$$(\sigma^{(i)})^2 \approx \frac{\mathrm{var}(h(\boldsymbol{x}))}{N} \underbrace{\sum_{t=-\infty}^{\infty} \rho(t)}_{\coloneqq 2\tau_{\mathrm{int}}} = \mathrm{var}(h(\boldsymbol{x}))\frac{2\tau_{\mathrm{int}}}{N}\,, \tag{A.11}$$

where we define the IACT provides a good estimate of how many steps the sampling algorithm needs to take to produce one independent sample. More specifically, the effective sample size $\frac{2\tau_{\mathrm{int}}}{N}$ gives an estimate of how efficient a sampler is.

A.3 Measure theory

Assume that the triple $(\Omega, \mathcal{F}, \mathbb{P})$ defines a probability space, where Ω denotes the complete sample space, $\mathcal{F}$ is a σ-algebra consisting of a collection of countable subsets $\{A_n\}_{n\in\mathbb{N}}$ with $A_n \subseteq \Omega$, and $\mathbb{P}$ is a probability measure on $\mathcal{F}$. The formal conditions for $\mathbb{P}$ to be a probability measure, and for $\mathcal{F}$ to be a σ-algebra over Ω, are given in Appendix A.3. We denote

$$\mathbb{P}(A) = \int_A \mathrm{d}\mathbb{P} \tag{A.12}$$

as the probability of an event $A \in \mathcal{F}$. By applying the Radon-Nikodym theorem [69], we can change variables

$$\mathbb{P}(A) = \int_A \frac{\mathrm{d}\mathbb{P}}{\mathrm{d}\boldsymbol{x}}\,\mathrm{d}\boldsymbol{x} = \int_A \pi(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}, \tag{A.13}$$

where $\mathrm{d}\boldsymbol{x}$ is a reference measure on the same probability space, commonly referred to as the Lebesgue measure. The Radon-Nikodym derivative $\frac{\mathrm{d}\mathbb{P}}{\mathrm{d}\boldsymbol{x}}$ of $\mathbb{P}$ with respect to $\boldsymbol{x}$ is often interpreted as the probability density function (PDF) $\pi(\boldsymbol{x})$. Thus, we say that $\mathbb{P}$ has a density $\pi(\boldsymbol{x})$ with respect to $\boldsymbol{x}$ [70, Chapter 10].

Now, let $X : \Omega \longrightarrow \mathbb{R}^d$ be a d-dimensional random variable mapping from the probability space $(\Omega, \mathcal{F}, \mathbb{P})$ to the measurable space $(\mathbb{R}^d, \mathcal{X})$, where $\mathcal{X}$ is a collection of subsets in $\mathbb{R}^d$ [71]. Then the associated PDF $\pi(\boldsymbol{x})$ is a joint density of X, induced by the probability measure on Ω [69, 71].

Recall the probability space $(\Omega, \mathcal{F}, \mathbb{P})$, where Ω denotes the sample space, and $\mathcal{F}$ is a collection of countable subsets $\{A_n\}_{n\in\mathbb{N}}$ of Ω. Each $A_n \subseteq \Omega$ is called an event, and a map $\mathbb{P} : \mathcal{F} \to \mathbb{R}$ is referred to as a measure. In the following, we describe the conditions required for $\mathcal{F}$ to be a σ-algebra, and for $\mathbb{P}$ to qualify as a probability measure. We refer to [72] [69] for further reading.

A.3.1 Probability Measure

For a probability measure, we require:

- $\mathbb{P}(\Omega) = 1$ and $\mathbb{P}(\emptyset) = 0$
- $\mathbb{P}(A) \in [0, 1]$
- $\mathbb{P}(\bigcup_{j\in\mathbb{N}} A_j) = \sum_{j\in\mathbb{N}} \mathbb{P}(A_j)$ if we have pairwise disjoint sets or $A_i \cap A_j = \emptyset$ for $i \neq j$

In other words, the probability assigned to the entire sample space must be equal to one, $\mathbb{P}(\Omega) = 1$, and the probability of the empty set must be zero, $\mathbb{P}(\emptyset) = 0$. For any subset $A \subseteq \Omega$, the probability $\mathbb{P}(A)$ must lie between zero and one, i.e., $\mathbb{P}(A) \in [0, 1]$. If e.g. two subsets A and B are disjoint (i.e., $A \cap B = \emptyset$), then the probability of their union satisfies $\mathbb{P}(A \cup B) = \mathbb{P}(A) + \mathbb{P}(B)$. This property must also hold for a countable sequence of disjoint sets $\{A_j\}_{j\in\mathbb{N}}$, such that $\mathbb{P}\left(\bigcup_{j\in\mathbb{N}} A_j\right) = \sum_{j\in\mathbb{N}} \mathbb{P}(A_j)$.

A.3.2 σ-Algebra

A collections of subsets $\mathcal{F}$ is called σ-algebra if:

- $\emptyset, \Omega \in \mathcal{F}$,
- if $A \in \mathcal{F}$ then $A^C \coloneqq A/\Omega \in \mathcal{F}$
- if $A_1, A_2, \cdots \in \mathcal{F}$ then $\bigcup_{j\in\mathbb{N}} A_j \in \mathcal{F}$

In other words, the empty set $\emptyset$ and the entire sample space Ω must always be elements of $\mathcal{F}$. If a set $A \in \mathcal{F}$, then its complement $A^C = \Omega \setminus A$ must also be in $\mathcal{F}$. If, in terms of a probability measure, we are able to assign a probability $\mathbb{P}(A)$ to an event A, we must also be able to assign a probability to the event "not A", i.e., $\mathbb{P}(A^C)$. Finally, if a countable collection of sets $A_1, A_2, \cdots \in \mathcal{F}$, then their union $\bigcup_{j\in\mathbb{N}} A_j$ must also be in $\mathcal{F}$. These three properties define the requirements for $\mathcal{F}$ to be a σ-algebra.

A.4 Python Code

18.0pt plus 2.0pt minus 1.0pt

```python
def MargBack(TTCore, univarGrid):
  ''' Backward marginalisation, see Prop. 1 as in SIRT from Cui et al. [30] '''

  dim = len(univarGrid)
  B = dim * [None]  # coeffTensor
  B[-1] = TTCore[-1]
  R = [None] * dim
  C = [None] * dim

  for k in range(dim - 1, 0, -1):
    r_kmin1, n, r_k = np.shape(TTCore[k])
    # Eq.  , [30, Eq.  22] !! we set Lebesgue Measure to const = one
    M = np.identity(n) * (univarGrid[k][-1] - univarGrid[k][0])  # Mass matrix
    L = scy.linalg.cholesky(M)

    # construct Tensor C Eq. (27)
    C[k] = np.zeros((r_kmin1, n, r_k))
    for alpha in range(0, r_kmin1):
      for l in range(0, r_k):
        C[k][alpha, :, l] = B[k][alpha, :, l] @ L[:, :]

    # unfold along first coordinate and compute thin QR decomposition of C^T
    # Eq. (28)
    Q, R[k] = np.linalg.qr(C[k].reshape((r_kmin1, n * r_k), order='C').transpose()
    , mode='reduced')

    # compute next coefficient tensor
    # Eq. (29)
    r_kmin2, n, r_kmin1 = np.shape(TTCore[k - 1])
    B[k - 1] = np.zeros(np.shape(TTCore[k - 1]))
    for alpha_2 in range(0, r_kmin2):
    #for i in range(0, n):
      for l_1 in range(0, r_kmin1):
        B[k - 1][alpha_2, :, l_1] = TTCore[k - 1][alpha_2, :, :] @ R[k][l_1, :]

return B
```

Listing A.1: Pyhton code to calculate Backward marginals, as in Prop. 1 and [30].

```python
def MargForw(TTCore, univarGrid):
''' Forward marginalisation, see Prop. 2, similar to backward marginalsiation as in Cui et al. [30] '''
# compute pre marginal coefficients sarting at dim = 1, k = 0
BPre = dim * [None]  # coeffTensor
LebLam = 1  # !! Lebesgue Measure
BPre[0] = TTCore[0]
RPre = [None] * dim
CPre = [None] * dim

for k in range(0, dim-1):
r_kmin1, n, r_k = np.shape(TTCore[k])
#  Eq.  , [30, Eq.  22] !! we set Lebesgue Measure to const = one
M = np.identity(n) * (univarGrid[k][-1] - univarGrid[k][0])  # Mass matrix
L = scy.linalg.cholesky(M)

# construct Tensor C [30, Eq.  (27)]
CPre[k] = np.zeros((r_kmin1, n, r_k))
for alpha in range(0, r_kmin1):
for l in range(0, r_k):
CPre[k][alpha, :, l] = BPre[k][alpha, :, l] @ L[:, :]

# unfold along first coordinate and compute thin QR decomposition of C
# 3.1 [30, Eq.  (28)]
Q, RPre[k] = np.linalg.qr(CPre[k].reshape((r_kmin1 * n, r_k), order='C'), mode='reduced')

# compute next coefficient tensor [30, Eq.  (29)]
r_k, n, r_kpls1 = np.shape(TTCore[k + 1])
BPre[k + 1] = np.zeros(np.shape(TTCore[k + 1]))
for alpha_1 in range(0, r_kpls1):
for l_1 in range(0, r_k):
BPre[k + 1][l_1, :, alpha_1] = RPre[k][l_1,:] @ TTCore[k + 1][:, :, alpha_1]

return BPre

```

Listing A.2: Pyhton code to calculate forward marginals, as in Prop. 2.

```python
def SIRT(seeds, SQTT, univarGrid, BackMarg, absError):
  ''' do squared inverse rosenblatt transform (SIRT) as in Cui et al. [30] '''
  dim, numbSampl = seeds.shape
  sampls = np.zeros(seeds.shape) # samples from approximated PDF
  probVal = np.zeros(seeds.shape) # PDF values, for MH-correction step
  Approx = np.zeros(seeds.shape[1]) # TT-Approx., to compare to true function

  # lebesgue measure for qudrature Eq. 2.24
  WholeLebLam = np.zeros(dim)
  for k in range(0, dim):
    WholeLebLam[k] = (univarGrid[k][-1] - univarGrid[k][0])
  lamX = np.ones(dim)
  for k in range(1, dim):
    lamX[k - 1] = np.prod(WholeLebLam[k:])

  # error as in Eq. 2.29 [30, Eq. (35)]
  gamError = absError / np.prod(WholeLebLam)

  # sample from first dimension [30, Eq. (30)]
  firstMarg = gamError * lamX[0] + np.sum(BackMarg[0][0, :, :] ** 2, 1)
  # cumulative distribution function, normalised numerically Eq. 2.43 [30, Eq. (17)]
  firstCDF = np.cumsum(firstMarg / np.sum(firstMarg))
  # draw samples as 'inverse transform'
  sampls[0] = np.interp(seeds[0], firstCDF, univarGrid[0])
  probVal[0] = np.interp(sampls[0], univarGrid[0], firstMarg / np.sum(firstMarg))

  # sample from other dimensions
  for n in range(0, numbSampl):
    # interpolate linear on grid in first dimension Eq. 2.44 [32]
    CurrApprCore = LinInterPolTT(SQTT[0], univarGrid[0], sampls[0][n])
    for d in range(1, dim):
      # marginal function condition on previous samples
      rank_min, gridSize, rank_pls = BackMarg[d].shape
      MargDep = np.zeros((BackMarg[d].shape))
      for r in range(0, rank_min):
        # condition on previous samples
        MargDep[r, :, :] = CurrApprCore[0, r] * BackMarg[d][r, :, :]

      # Eq. 2.45 [30, Eq. (31)]
      currMarg = gamError * lamX[d] + np.sum(np.sum(np.copy(MargDep), axis=0)** 2,
        axis=1)
      # Eq. 2.43 [30, Eq. (17)]
      currCDF = np.cumsum(currMarg / np.sum(currMarg))

      # draw sample as 'inverse transform'
      sampls[d][n] = np.interp(seeds[d][n], currCDF, univarGrid[d])
      probVal[d][n] = np.interp(sampls[d][n], univarGrid[d],
        currMarg / np.sum(currMarg))
      # piecew. poly. interpol., Eq. 2.44 [32], cond. on sampl. for next PDF
      CurrApprCore = np.copy(CurrApprCore) @ LinInterPolTT(SQTT[d], univarGrid[d],
        sampls[d][n])

  Approx[n] = gamError + CurrApprCore ** 2

  return sampls, probVal, Approx
```

Listing A.3: Pyhton code to draw samples via SIRT, as in Alg. Box 3.

B
Additional Figures

B.1 Ozone

B.1.1 Ozone Prior

B.1.2 Integrated Autocorrelation plots

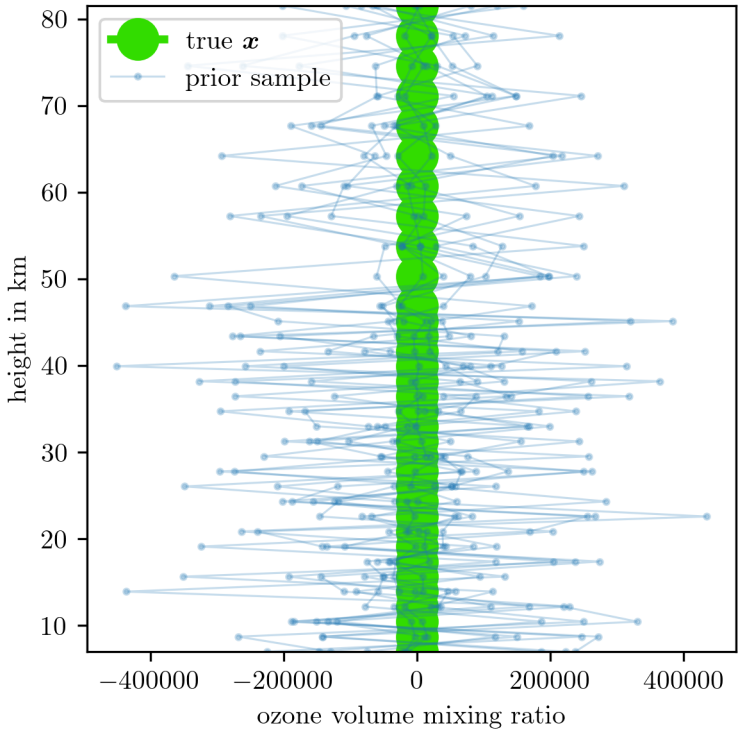


Figure B.1: We draw samples from ozone prior distribution $\boldsymbol{x} \sim \mathcal{N}(0, \delta \boldsymbol{L})$ after generating a sample from the hyper-prior distribution $\delta \sim \mathcal{T}(1, 10^{-10})$. Note that since the variance of prior samples is very large compared to the ozone volume mixing ratios, the ozone profile appears to be constant, which is not the case, see e.g. Fig. 4.6.

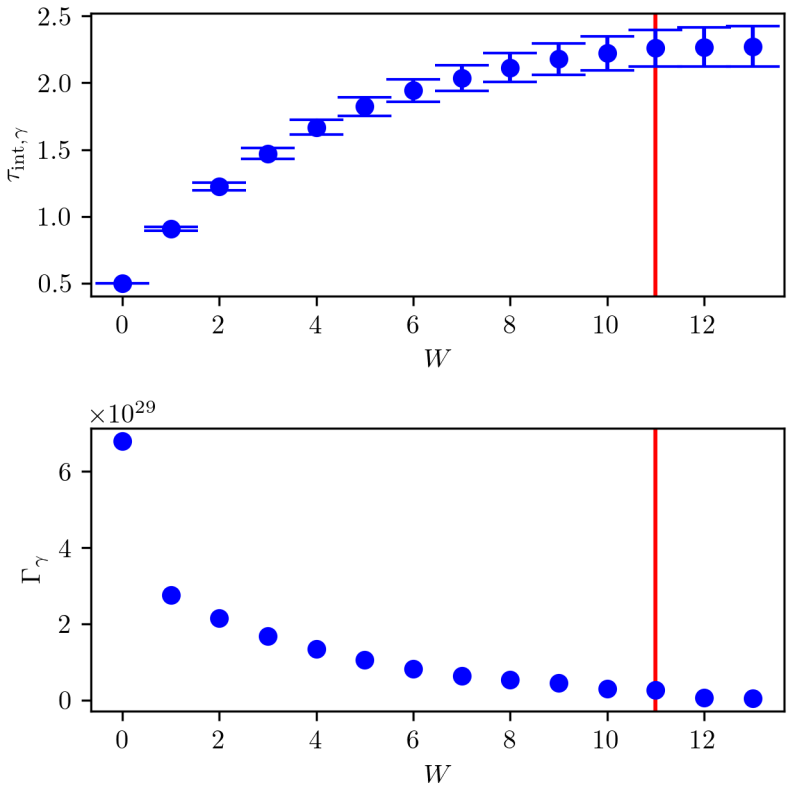


Figure B.2: Here the autocorrelation function Γ_γ at different lags W is plotted as well as the IATC $\tau_{\text{int},\gamma}$ for the samples from $\pi(\gamma, \lambda|\boldsymbol{y})$ based on the linear forward model.

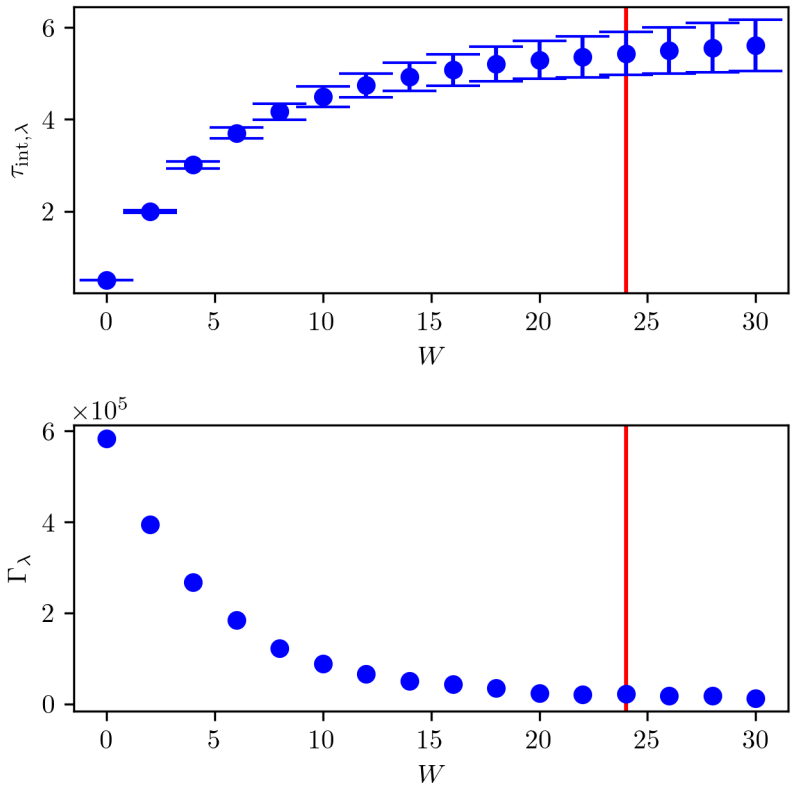


Figure B.3: IACT for samples $\lambda \sim \pi(\cdot|\gamma, \boldsymbol{y})$ based on the approximated forward model.

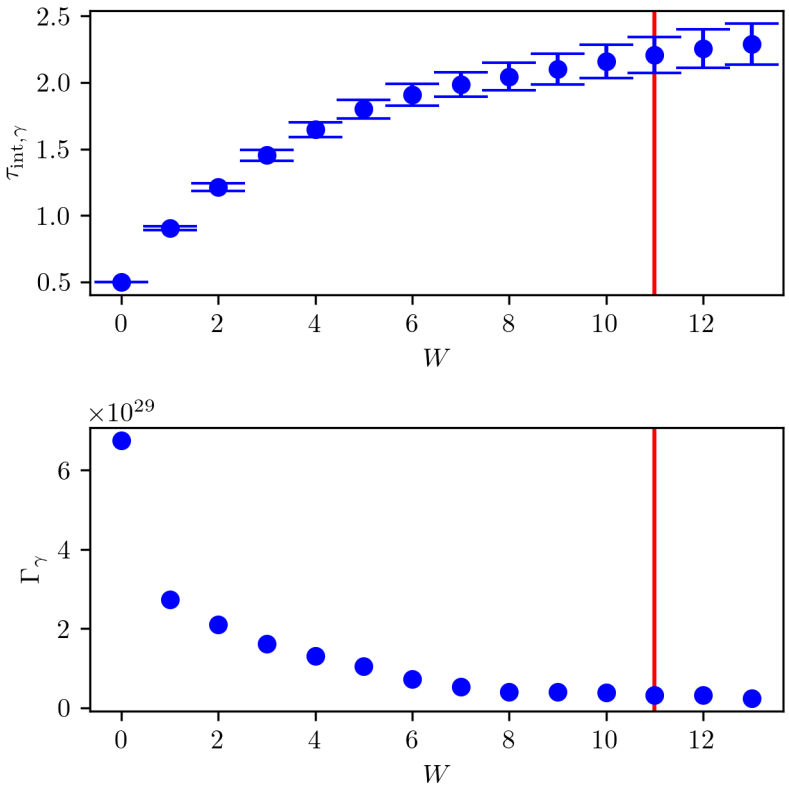


Figure B.4: IACT for samples $\gamma \sim \pi(\cdot|\lambda, \boldsymbol{y})$ based on the approximated forward model.

B.2 Pressure and Temperature

B.2.1 Priors

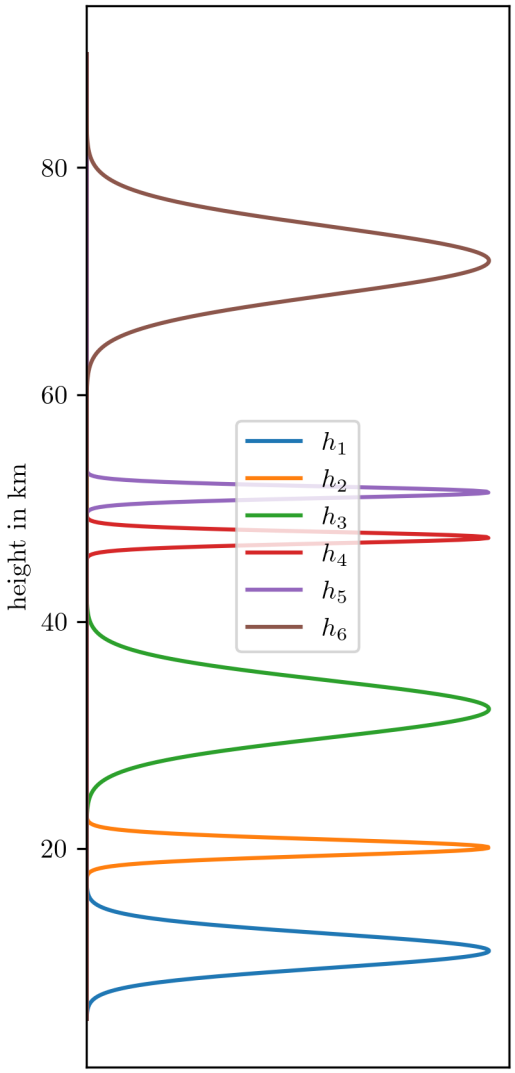


Figure B.5: Prior distributions $\pi(\boldsymbol{h_T})$, which we choose so that they do not overlap and not conflict with the temperature function 4.3

B.2.2 T-walk Trace

B.2.3 Integrated Autocorrelation Plots

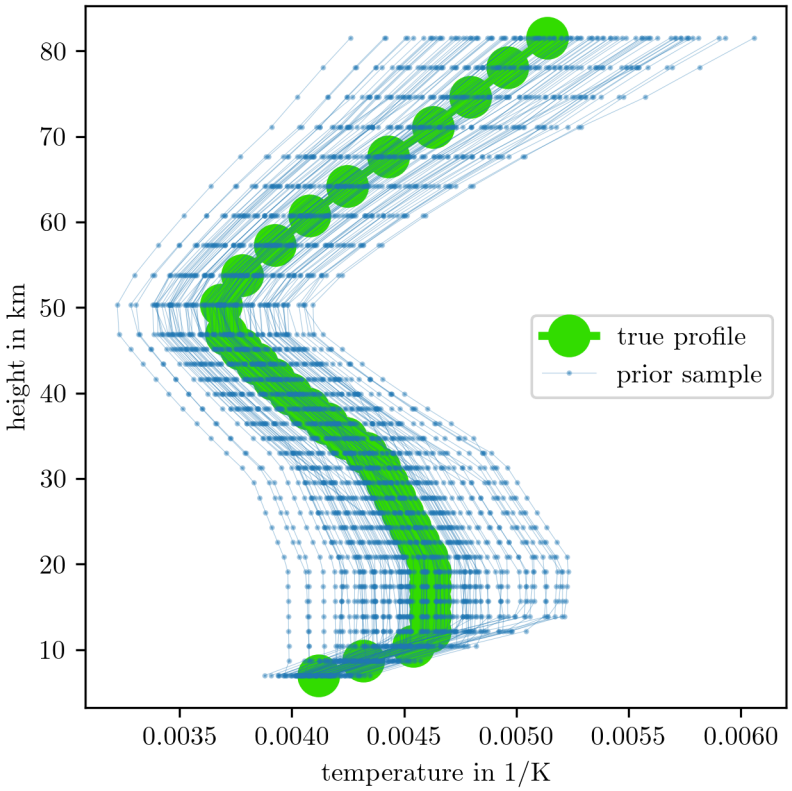


Figure B.6: Prior samples of the inverted temperature profile.

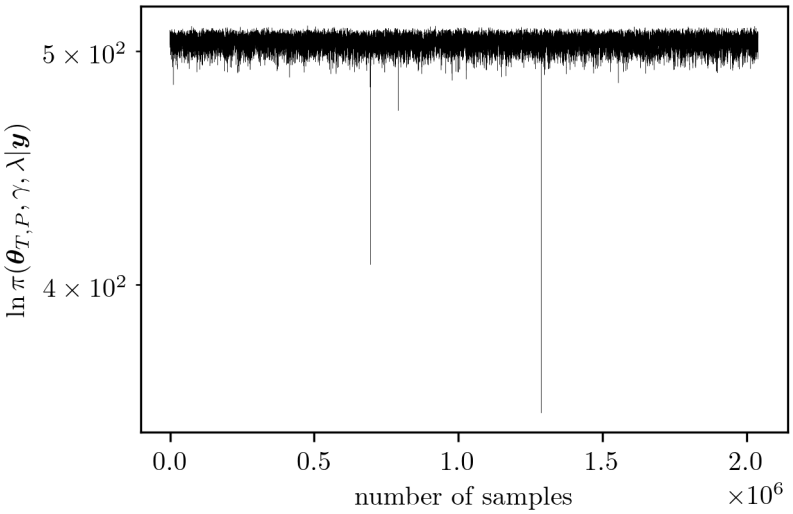


Figure B.7: Output trace of the t-walk on the posterior distribution $\pi(p_0, b, \boldsymbol{h_T}, \boldsymbol{a} | \gamma, \boldsymbol{y})$.

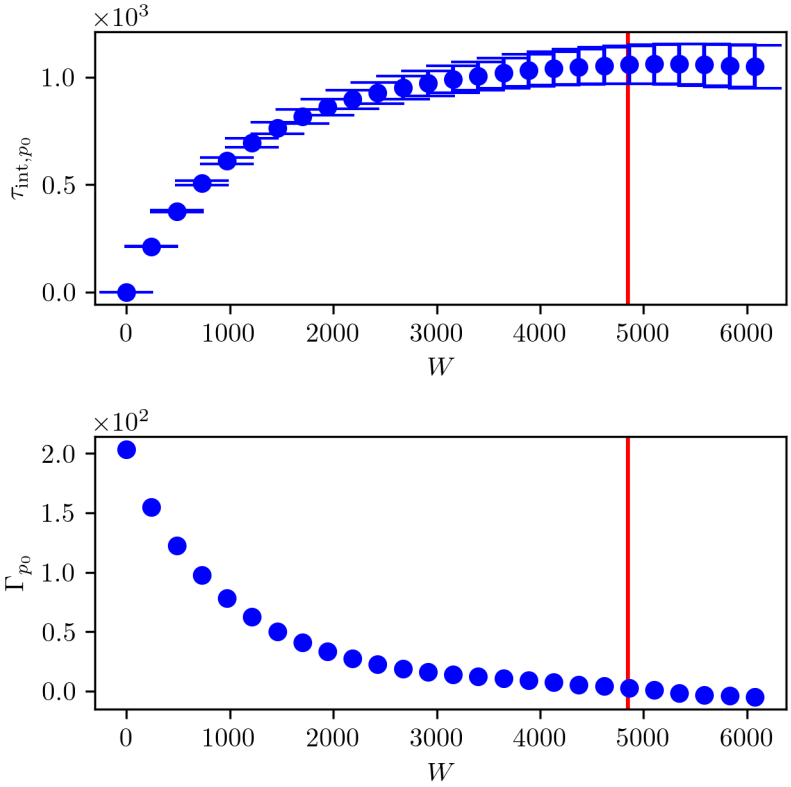


Figure B.8: IACT and autocorrelation function for samples $h_1 \sim \pi(\cdot|h_{T,2}, h_{T,3}, h_{T,4}, h_{T,5}, h_{T,6}, a_0, a_1, a_2, a_3, a_4, a_5, a_6, T_0, b, p_0, \boldsymbol{y})$

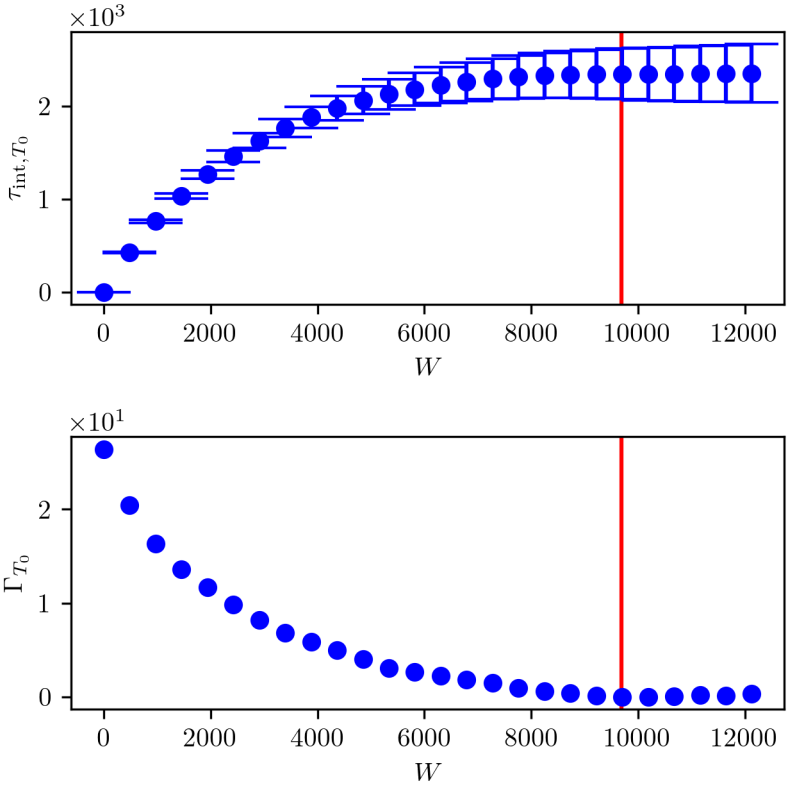


Figure B.9: IACT and autocorrelation function for samples $h_2 \sim \pi(\cdot|h_1, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4, a_5, a_6, T_0, b, p_0, \boldsymbol{y})$

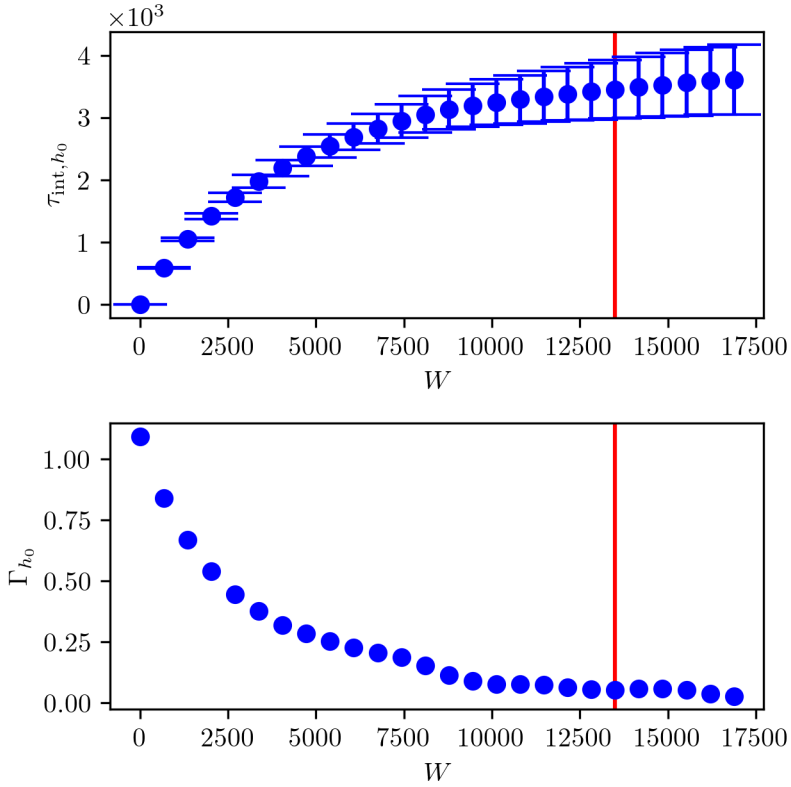


Figure B.10: IACT and autocorrelation function for samples $h_3 \sim \pi(\cdot|h_1, h_2, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4, a_5, a_6, T_0, b, p_0, \boldsymbol{y})$

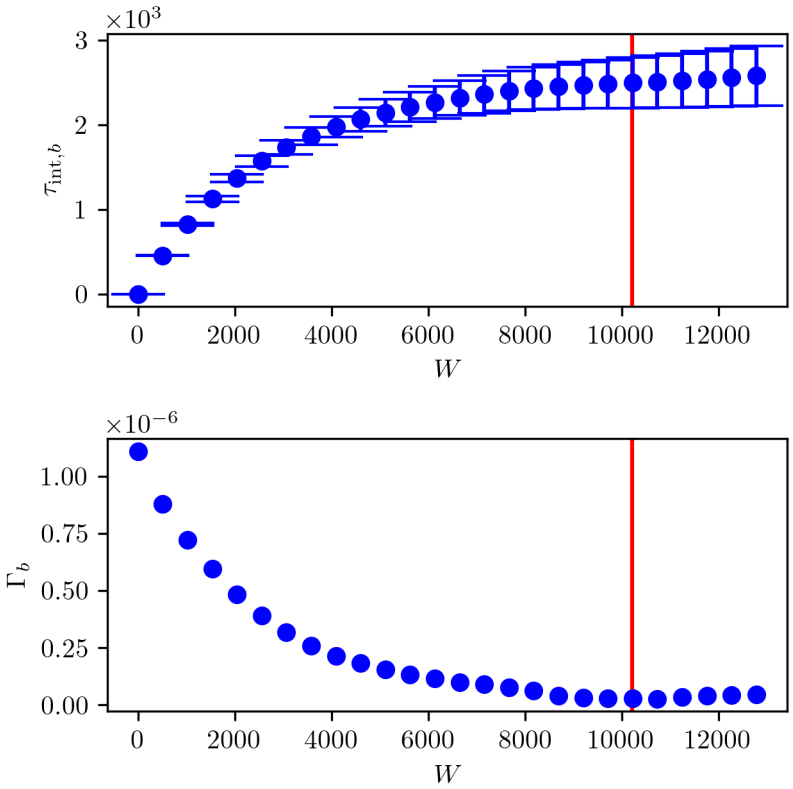


Figure B.11: IACT and autocorrelation function for samples $h_4 \sim \pi(\cdot|h_1, h_2, h_3, h_5, h_6, a_0, a_1, a_2, a_3, a_4, a_5, a_6, T_0, b, p_0, \boldsymbol{y})$

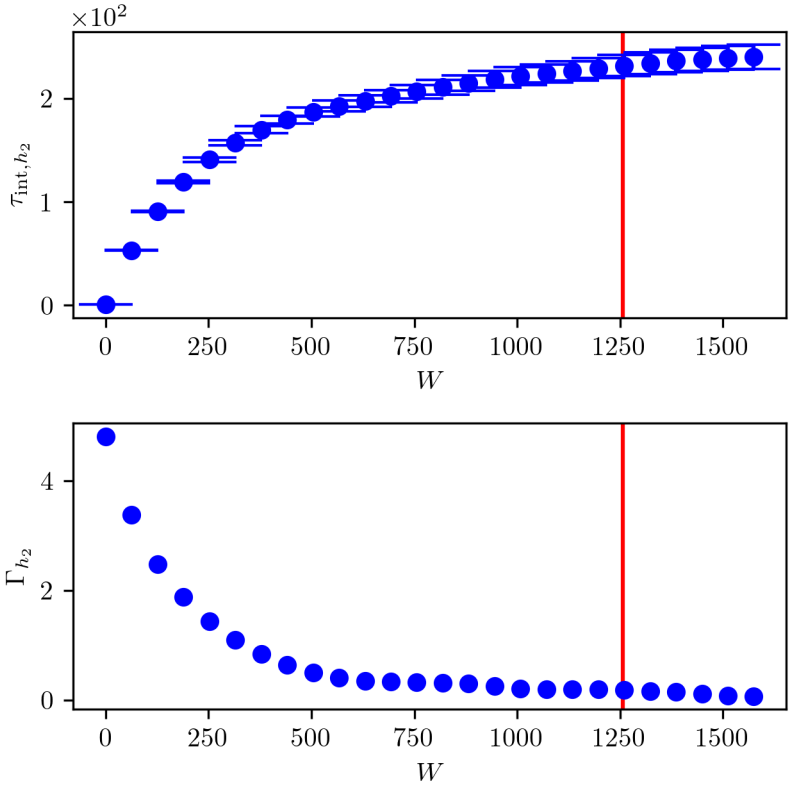


Figure B.12: IACT and autocorrelation function for samples $h_5 \sim \pi(\cdot|h_1, h_2, h_3, h_4, h_6, a_0, a_1, a_2, a_3, a_4, a_5, a_6, T_0, b, p_0, \boldsymbol{y})$

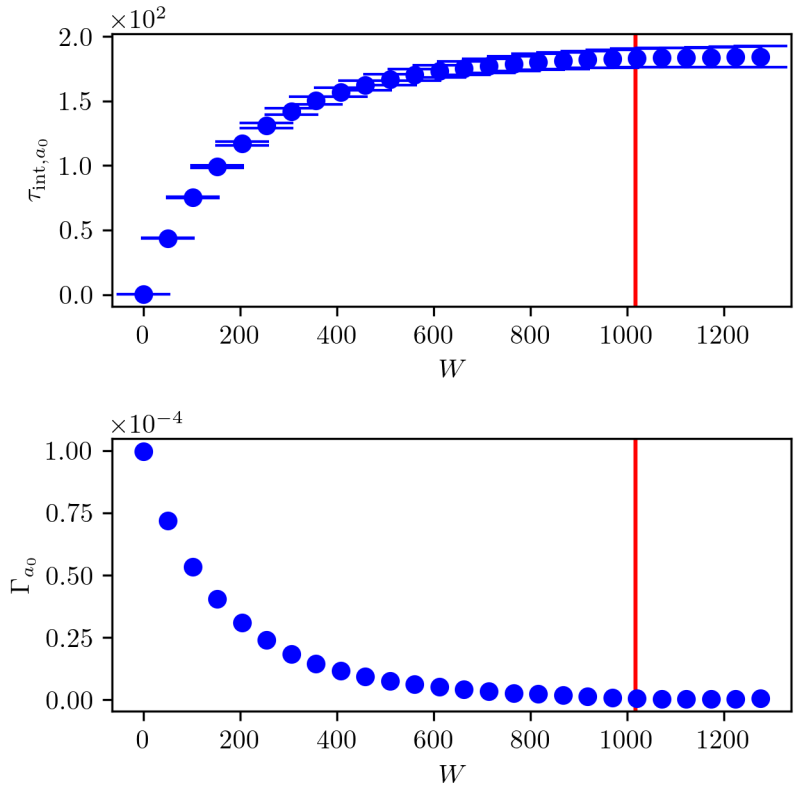


Figure B.13: IACT and autocorrelation function for samples $h_6 \sim \pi(\cdot|h_1, h_2, h_3, h_4, h_5, a_0, a_1, a_2, a_3, a_4, a_5, a_6, T_0, b, p_0, \boldsymbol{y})$

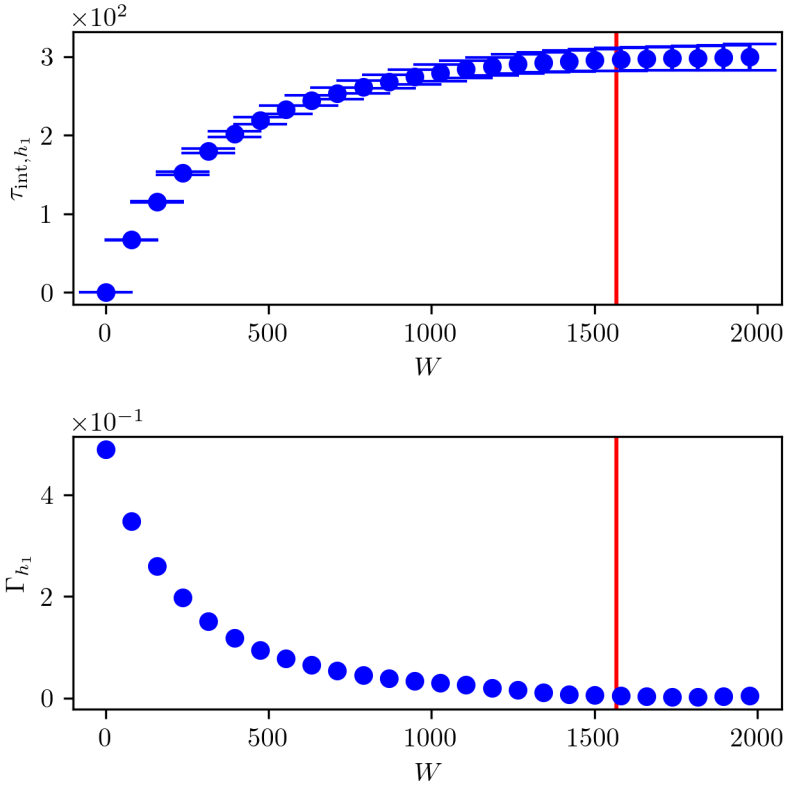


Figure B.14: IACT and autocorrelation function for samples $a_0 \sim \pi(\cdot|h_1, h_2, h_3, h_4, h_5, h_6, a_1, a_2, a_3, a_4, a_5, a_6, T_0, b, p_0, \boldsymbol{y})$

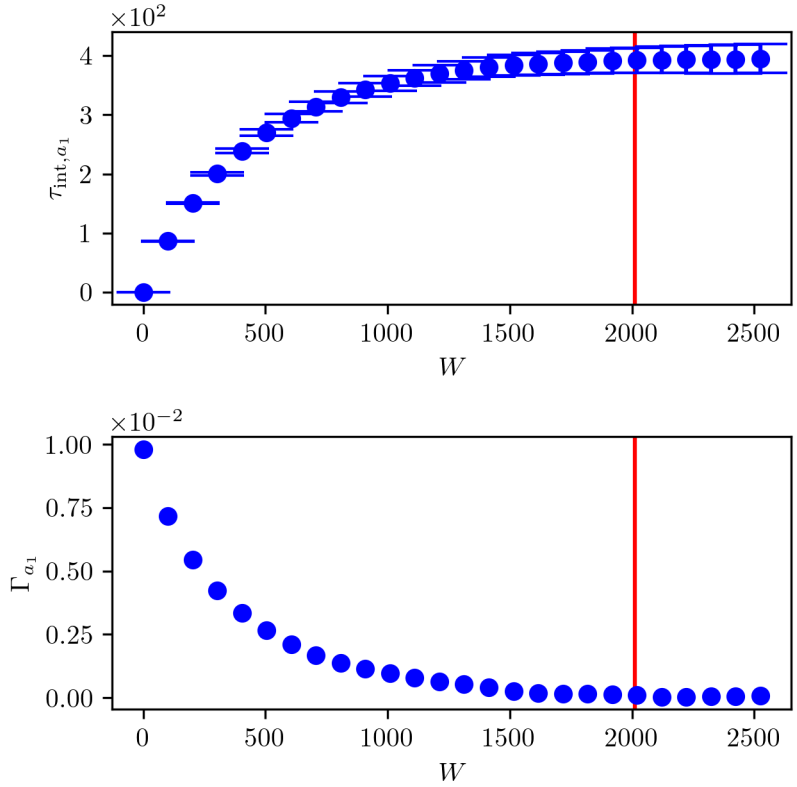


Figure B.15: IACT and autocorrelation function for samples $a_1 \sim \pi(\cdot|h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_2, a_3, a_4, a_5, a_6, T_0, b, p_0, \boldsymbol{y})$

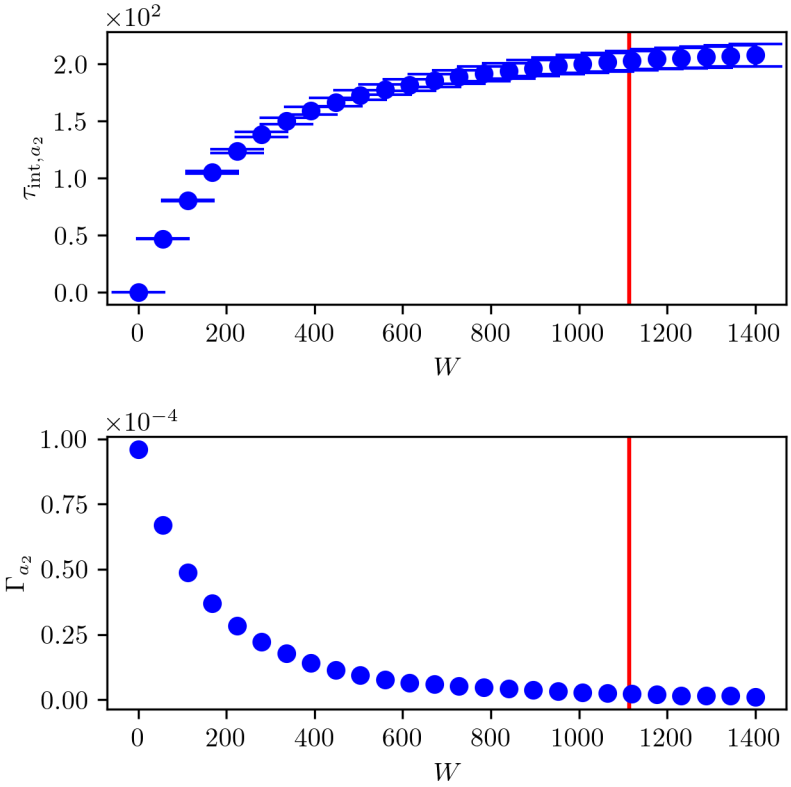


Figure B.16: IACT and autocorrelation function for samples $a_2 \sim \pi(\cdot|h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_3, a_4, a_5, a_6, T_0, b, p_0, \boldsymbol{y})$

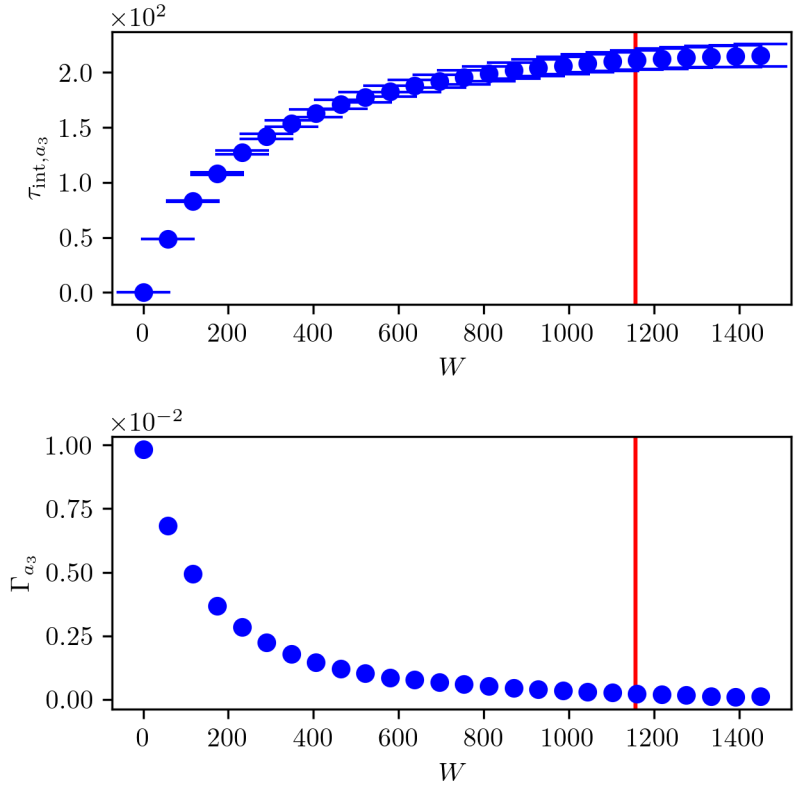


Figure B.17: IACT and autocorrelation function for samples $a_3 \sim \pi(\cdot|h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_4, a_5, a_6, T_0, b, p_0, \boldsymbol{y})$

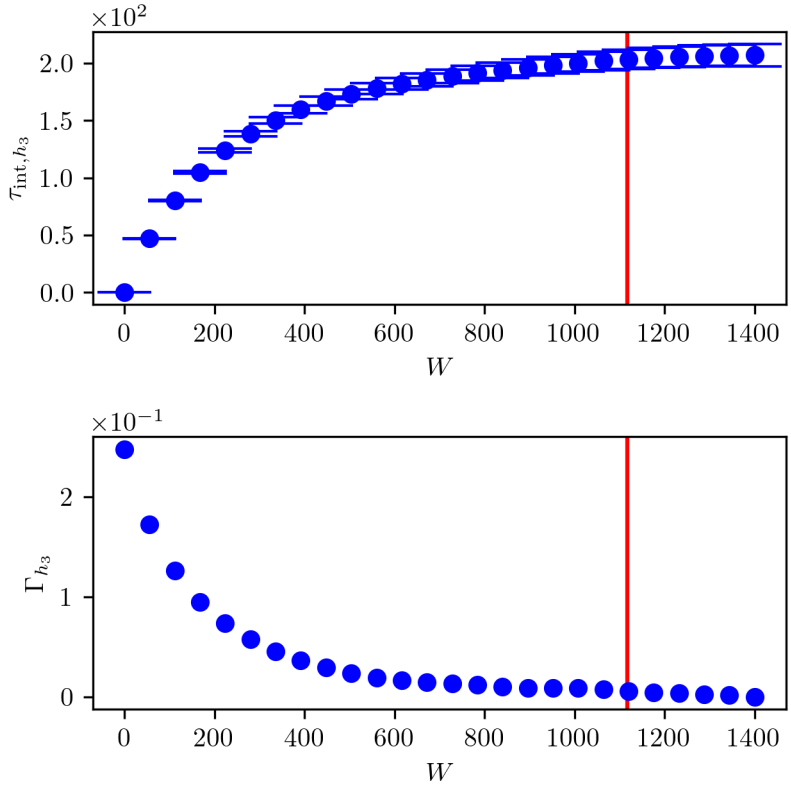


Figure B.18: IACT and autocorrelation function for samples $a_4 \sim \pi(\cdot|h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_4, a_5, a_6, T_0, b, p_0, \boldsymbol{y})$

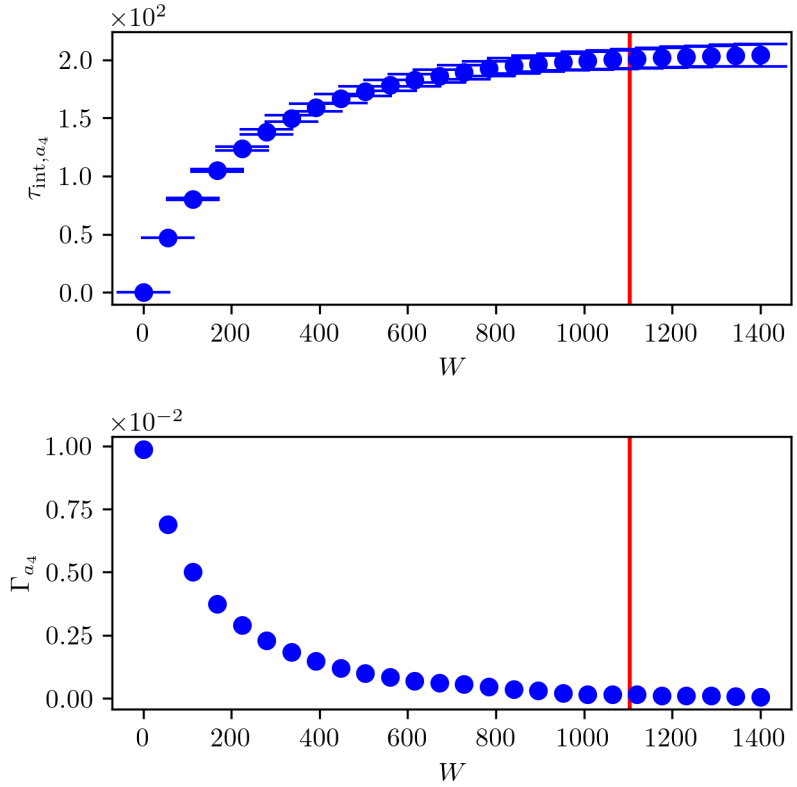


Figure B.19: IACT and autocorrelation function for samples $a_5 \sim \pi(\cdot|h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4, a_5, T_0, b, p_0, \boldsymbol{y})$

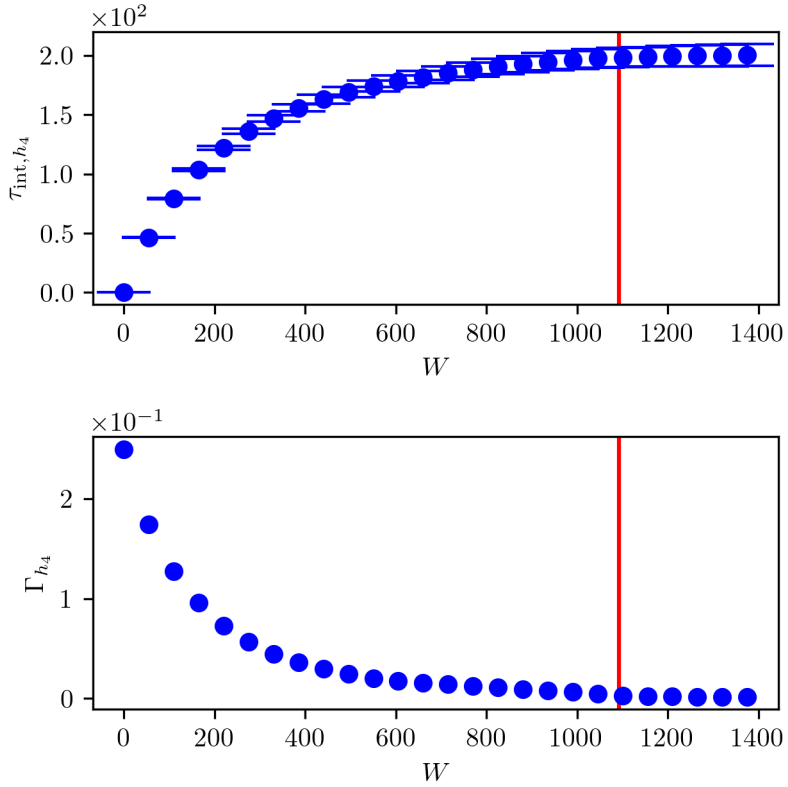


Figure B.20: IACT and autocorrelation function for samples $a_6 \sim \pi(\cdot|h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4, a_5, T_0, b, p_0, \boldsymbol{y})$

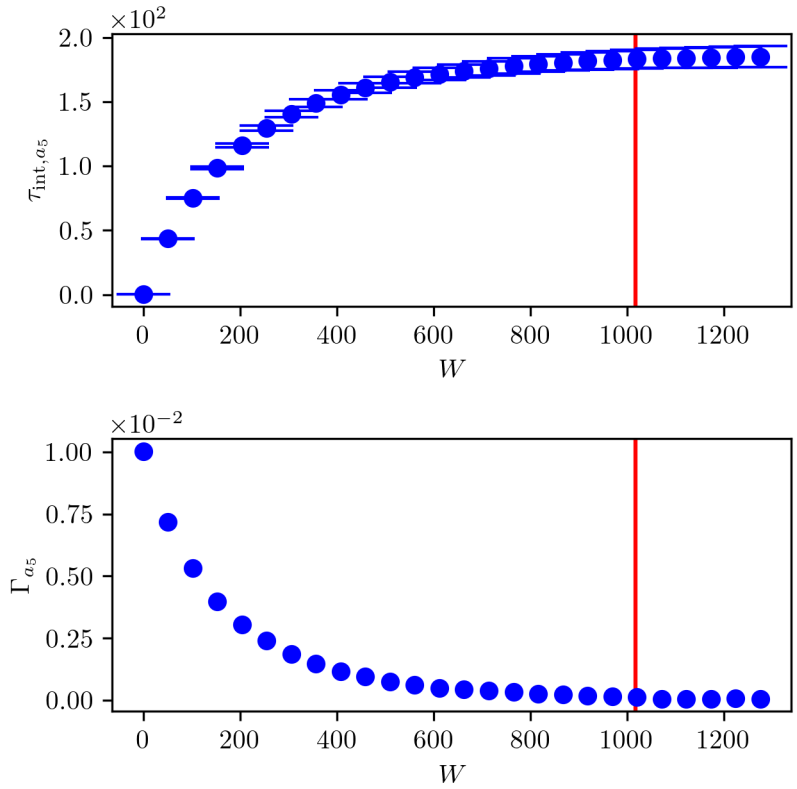


Figure B.21: IACT and autocorrelation function for samples $T_0 \sim \pi(\cdot|h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4, a_5, a_6, b, p_0, \boldsymbol{y})$

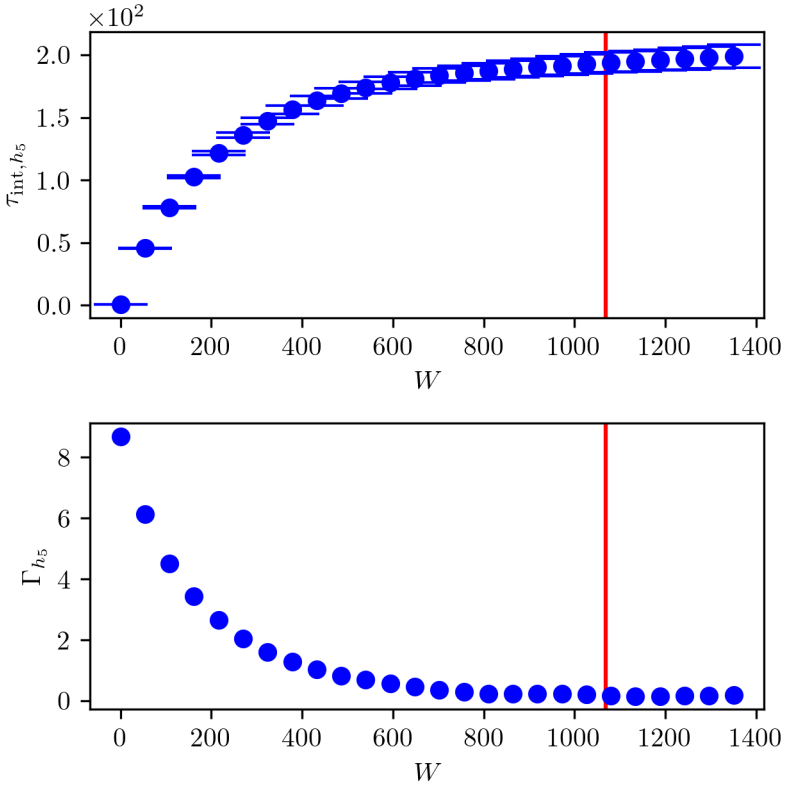


Figure B.22: IACT and autocorrelation function for samples $b \sim \pi(\cdot|h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4, a_5, a_6, T_0, p_0, \boldsymbol{y})$

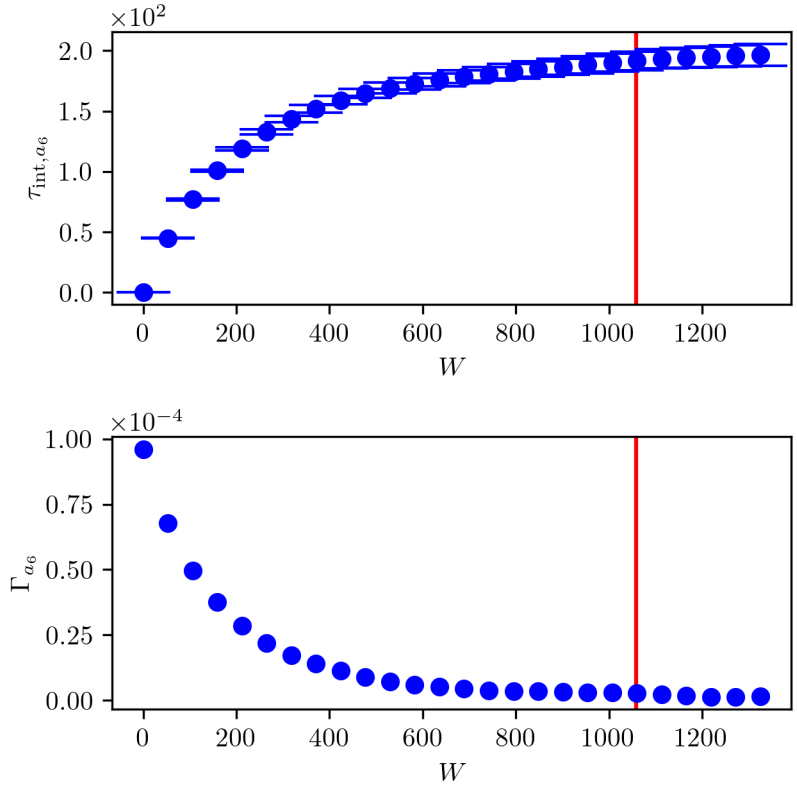


Figure B.23: IACT and autocorrelation function for samples $p_0 \sim \pi(\cdot|h_1, h_2, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4, a_5, a_6, T_0, b, \boldsymbol{y})$