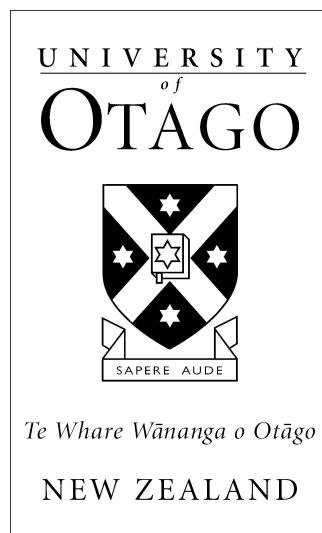


Suitably impressive thesis title



Lennart Golks
Department of Physics
University of Otago

A thesis submitted for the degree of
Doctor of Philosophy

November 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	ix
1 Introduction	1
1.1 Motivation	1
1.2 Research Gap and Contribution	1
1.3 Thesis Structure	2
2 Theoretical and Technical Background	5
2.1 Bayesian Inference	5
2.2 Sampling Methods	8
2.2.1 Sampling from the marginal posterior	9
2.2.2 t-walk sampler as black box	10
2.3 Numerical Approximation Methods - Tensor Train	10
2.3.1 Marginal Functions for quadrature	13
2.3.2 Sampling from the TT approximation – Squared Inverse Rosenblatt Transform	15
2.4 Affine Map	16
2.5 Regularisation	17
3 Forward Model	19
3.1 Singular value decomposition of linear forward model matrix	21
4 Results	25
4.1 Simulate Data based on a ground truth	25
4.2 Set up the Bayesian framework	28
4.2.1 Ozone conditioned on pressure and temperature	29
4.2.2 Pressure over temperature conditioned on noise and ozone	33
4.3 Approximate non-linear forward model with affine Map	37
4.3.1 Sample from marginal posterior distribution for ozone	37
4.3.2 Tensor-train approximation of the marginal posterior distribution for ozone	38
4.3.3 Calculate mean and variance of the full conditional posterior for ozone	39

4.3.4	Asses approximated forward model	40
4.4	Solution by regularisation	41
4.5	Characterise the posterior distribution of ozone with approximated forward model	42
4.5.1	Hyper-parameters samples from and Tensor-train approximation of the marginal posterior distribution	42
4.5.2	Full conditional posterior variance and mean	43
4.6	Posterior distribution for pressure and temperature with approximated forward model	49
4.7	Error analysis	59
5	Conclusions	61
5.1	Regularisation vs MTC	61
5.2	Sampling vs TT	61
5.2.1	Intuioton of TT	62
5.3	Approximation Errors	62
5.4	Atmospheric Physics	62
6	Outlook	65
6.1	Measurement Device	65
6.2	Methods	65
6.2.1	TT approximation	65
6.2.2	Sampling	65
6.2.3	Model	65
Appendices		
A	Correlation Structure	69
B	Mesure theory	71
B.1	probailty measure	71
B.2	σ -algebra	72
C	Additional Figures	73
C.1	Integrated Autocorrelation plots	73
C.1.1	Ozone	73
C.1.2	Pressure and Temperature	75
C.2	Ozone	91
C.3	Pressure over Temperature	93
References		95

List of Figures

2.1	Bayesian Inference DAG	6
2.2	Visualisation of a tensor train	12
2.3	Schematics of the affine map	16
3.1	Schematic of measurement and analysis geometry.	19
3.2	Tangent heights for different sequence of measurements.	22
3.3	Singular values of linear forward model matrix for different sequences of measurements.	23
3.4	Left singular vectors of forward model matrix for one sequence of measurements.	24
4.1	Logarithmic plot of data points at different tangent height.	27
4.2	Complete directed acyclic graph of the forward model.	28
4.3	Plot of the functions $f(\lambda)$ and $g(\lambda)$ for marginal posterior.	32
4.4	Prior Samples of \mathbf{p}/\mathbf{T} according to the respective hyper-prior distribution. .	34
4.5	Prior Samples of \mathbf{T} according to the respective hyper-prior distribution. .	35
4.6	Prior Samples of \mathbf{p} according to the respective hyper-prior distribution. .	36
4.7	Strategy to find affine map.	37
4.10	Scatter plot of samples from marginal posterior, including weighting from TT approximation; additional trace plot of the marginal posterior samples.	41
4.12	Ozone samples of the conditional posterior.	43
4.13	Assessment of affine map.	44
4.14	Plot of the L-curve to find the regularised solution.	45
4.15	Marginal posterior histograms and TT approximation as well as hyper-prior distribution.	46
4.16	Ozone posterior mean and variance and the regularised solution compared to the ground truth.	47
4.18	Histograms and TT approximation of posterior distribution as well as hyper-prior distribution.	50
4.19	Histograms and TT approximation of posterior distribution as well as hyper-prior distribution.	51
4.20	Histograms and TT approximation of posterior distribution as well as hyper-prior distribution.	52

4.21 Histograms and TT approximation of posterior distribution as well as hyper-prior distribution.	53
4.22 Histograms and TT approximation of posterior distribution as well as hyper-prior distribution.	54
4.23 Temperature posterior samples.	55
4.24 Pressure posterior samples.	56
4.25 Correlation plot of samples from TT-approximation	57
4.26 Assessment of Monte-Carlo error.	60
A.1 Correlation structure in between parameters and hyper-parameters	70
C.1 IACT and autocorrelation function for samples $\lambda \sim \pi(\cdot \gamma, \mathbf{y})$	73
C.2 IACT and autocorrelation function for samples $\gamma \sim \pi(\cdot \lambda, \mathbf{y})$	74
C.3 IACT and autocorrelation function for $h_{T,1}$ samples.	75
C.4 IACT and autocorrelation function for h_2 samples.	76
C.5 IACT and autocorrelation function for h_3 samples.	77
C.6 IACT and autocorrelation function for h_4 samples.	78
C.7 IACT and autocorrelation function for h_5 samples.	79
C.8 IACT and autocorrelation function for h_6 samples.	80
C.9 IACT and autocorrelation function for a_0 samples.	81
C.10 IACT and autocorrelation function for a_1 samples.	82
C.11 IACT and autocorrelation function for a_2 samples.	83
C.12 IACT and autocorrelation function for a_3 samples.	84
C.13 IACT and autocorrelation function for a_4 samples.	85
C.14 IACT and autocorrelation function for a_5 samples.	86
C.15 IACT and autocorrelation function for a_6 samples.	87
C.16 IACT and autocorrelation function for T_0 samples.	88
C.17 IACT and autocorrelation function for b samples.	89
C.18 IACT and autocorrelation function for p_0 samples	90
C.19 Directed acyclic graph for ozone retrieval and MTC scheme.	91
C.20 Samples from ozone prior distribution.	92
C.21 Prior distributions $\pi(\mathbf{h}_T)$	93
C.22 Prior samples of $1/T$	94
C.23 T-walk trace	94

1

Introduction

1.1 Motivation

Since the only current ozone limb sounder the MLS on the Aura satellite is drifting away from its orbit will be phased out within 2026, a group around Harald Schwefel proposed to fill that gap with to measure ozone on a much smaller platform such as a cube-sat <https://science.nasa.gov/science-research/earth-science/aura-at-20-years/>. [1] The idea is to target a very narrow frequency band and convert the thermal radiation of ozone from the Terahertz region to the optical domain, so that big cooling devices are not needed. We simulate some data from a limb sounder by solving the radiative transfer equation (RTE) for one specific frequency but leave out any measurement device related details, as they are not available yet. This inverse problem is currently approached in the atmospheric physics community by methods based on optimisation/regularisation from 1970 (which is a very long time ago) [], instead we employ a hierarchically ordered Bayesian framework to recover ozone values.

1.2 Research Gap and Contribution

As, already mentioned the commonly used methods in Atmospheric physics to retrieve ozone from some data are based on regularisation methods, which can lead to unphysical results e.g. negative ozone values [2]. Regularisation methods have the disadvantage that they provide one solution only, which may be biased, and rather arbitrary errors, as "best fit to data is not the best fit to parameters" [3]. Hence, we propose a Bayesian framework, to provide a posterior distribution of unbiased ozone profiles, which naturally includes errors. We compare a regularisation method to a hierachialy ordered linear-Gaussian Bayesian

model, where we separate the posterior distribution over the hyper-parameters, describing noise of the data and smoothnes of the ozone profile, and the posterior distribution over the ozone profile. Fox and Norton call this the marginal and then conditional (MTC) method [] and we are the first to apply it to a physical based problem and hence the first to tackle the RTE with a hierachichally order Bayesian model. Since the RTE is weakly non-linear we do approximate the RTE with an affine map, which is another novelty. Then, instead of sampling from the posterior distributions we are the first to calculate marginal posterior distributions using a tensor-train (TT). Additionally, we are the first to provide pressure, temperature and ozone profile given one set of measurements JOINTLY. []

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 the RTE and some functional dependencies. Using that, in Ch. 4 we use that forward model and simulate some data according to a ground truth. Then to characterise the posterior distributions, we set up a Bayesian framework, where we discuss some prior modelling. Finally, we use the in Ch. 2 introduced methods to aprroximate the non-linear forward model and provide distribution of ozone, pressure and temperature profiles.

421.10046pt

2

Theoretical and Technical Background

In this chapter, we provide brief introductions and derivations to 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, but without derivations. We begin by introducing a general hierarchical Bayesian approach to a linear inverse problem. Next, we provide a small introduction to sampling methods, more specifically, the essentials of Markov-Chain Monte Carlo (MCMC) methods. Lastly, we explain how we approximate functions using a Tensor-Train (TT) approach, which enables us to calculate marginals from the posterior distribution cheaply. Lastly, we provide some background information on affine maps and the Tikhonov regularisation method.

2.1 Bayesian Inference

Assume we observe some data

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

based on a linear forward model \mathbf{A} , a unknown parameter \mathbf{x} and some additive random noise $\boldsymbol{\eta}$. Naturally, due to the noise, we have some uncertainty which we include in the modelling process through the likelihood function $\pi(\mathbf{y}|\boldsymbol{\theta}, \mathbf{x})$ as well as other relevant information about the measurement process. We read $\pi(\mathbf{y}|\boldsymbol{\theta}, \mathbf{x})$ as the distribution over \mathbf{y} conditioned on \mathbf{x} and the hyper-parameter $\boldsymbol{\theta}$. Here $\boldsymbol{\theta}$ may account for multiple variables and is e.g. describing the distribution of the noise vector $\boldsymbol{\eta} \sim \pi_{\boldsymbol{\eta}}(\cdot|\boldsymbol{\theta})$ as well as the prior distribution $\pi(\mathbf{x}|\boldsymbol{\theta})$, which accounts for physical properties or functional dependences of \mathbf{x} . Consequently we define a hyper-prior distribution $\pi(\boldsymbol{\theta})$, where

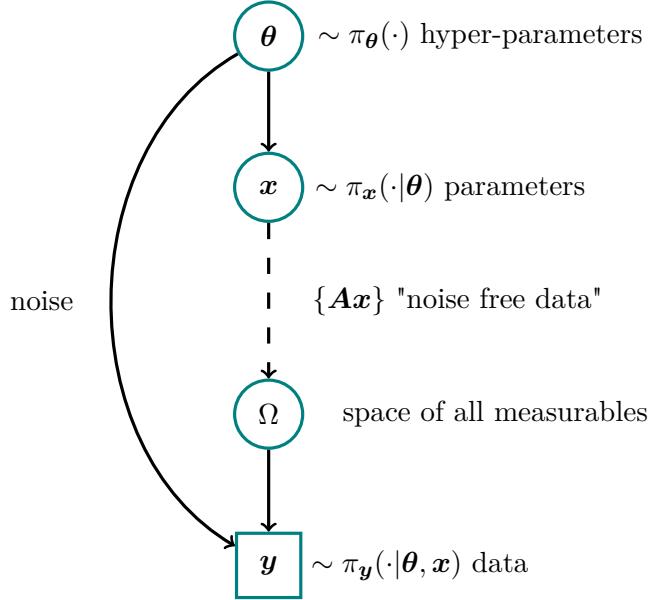


Figure 2.1: The directed acyclic graph (DAG) for a linear inverse problem visualises statistical dependencies as solid line arrows and deterministic dependencies as dotted arrows. The hyper-parameters θ are distributed as the hyper-prior distribution $\pi(\theta)$. The prior distribution $\pi_x(\cdot|\theta)$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

$\pi(x, \theta) = \pi(x|\theta)\pi(\theta)$. Choosing these prior distributions is a delicate topic, as it shall not affect the posterior distribution

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

which according to Bayes theorem gives us a distribution of x and θ given (conditioned) on some data. 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 $h(x_\theta)$, where x may depend on θ , is described as

$$E_{x, \theta|y}[h(x_\theta)] = \underbrace{\int \int h(x_\theta) \pi(x, \theta|y) dx d\theta}_{\mu_{\text{int}}}, \quad (2.3)$$

where $\pi(x, \theta|y)$ is the posterior distribution. This may be a high-dimensional integral and computationally not feasible to solve. Therefore the unbiased sample-based Monte Carlo estimate [4]

$$E_{x, \theta|y}[h(x_\theta)] \approx \underbrace{\frac{1}{N} \sum_{k=1}^N h(x_\theta^{(k)})}_{\mu_{\text{samp}}}, \quad (2.4)$$

for large enough N (law of large numbers [5, Chapter 17]) is often used. Here, the samples $\{\mathbf{x}^{(k)}, \boldsymbol{\theta}^{(k)}\} \sim \pi_{\mathbf{x}, \boldsymbol{\theta}}(\cdot | \mathbf{y})$, for $k = 1, \dots, N$, form a sample set $\mathcal{M} = \{(\mathbf{x}, \boldsymbol{\theta})^{(1)}, \dots, (\mathbf{x}, \boldsymbol{\theta})^{(N)}\}$. 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 [6] and illustrated in Appendix A. If \mathbf{x} can not be parametrised directly in terms of the hyper-parameters $\boldsymbol{\theta}$, i.e., $\mathbf{x}(\boldsymbol{\theta})$, it is beneficial to factorise the posterior distribution as

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

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

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

over the hyper-parameters $\boldsymbol{\theta}$. This approach, known as the marginal and then conditional (MTC) method, is particularly advantageous when \mathbf{x} is high-dimensional (e.g., $\mathbf{x} \in \mathbb{R}^n$ with $n \geq 45$), while $\boldsymbol{\theta}$ is low-dimensional (e.g., two-dimensional). Applying the law of total expectation [7], Eq. (2.3) becomes

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

where, in the case of a linear-Gaussian Bayesian hierarchical model, both the marginal distribution and the inner expectation $\mathbb{E}_{\mathbf{x} | \boldsymbol{\theta}, \mathbf{y}}[h(\mathbf{x}_{\boldsymbol{\theta}})]$ are well defined see Result chapter. Furthermore, the central limit theorem states that the samples mean $\boldsymbol{\mu}_{\text{samp}}^{(i)}$, of independent samples sets \mathcal{M}_i for $i = 1, \dots, n$ of any distribution, converge in distribution to a normal distribution so that

$$\sqrt{n}(\boldsymbol{\mu}_{\text{samp}}^{(i)} - \boldsymbol{\mu}_{\text{int}}) \xrightarrow{\mathcal{D}} \mathcal{N}(0, \sigma^2)[8], \quad (2.8)$$

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

On the Monte-Carlo Error and Integrated Autocorrelation time

To assess the error σ^2 , 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. In general, the error of a Monte-Carlo-based estimate from a sample set \mathcal{M}_i is:

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

Expanding this summation, we see that

$$(\sigma^{(i)})^2 = \frac{1}{N^2} \sum_{k,s=1}^N C(k-s) \quad (2.10)$$

with the auto correlation coefficient $C(k - s) = (h(\mathbf{x}_\theta^{(k)}) - \mu^{(i)})(h(\mathbf{x}_\theta^{(s)}) - \mu^{(i)})$ and define the sample auto correlation function

$$\frac{C(0)}{N} \sum_{k,s=1}^N \frac{C(k - s)}{C(0)} \approx \text{var}(h(\mathbf{x}_\theta)) \sum_{t=-\infty}^{\infty} \rho(t) \quad (2.11)$$

with the normalised auto correlation coefficient $\rho(k - s) = C(k - s)/C(0)$ at lag $k - s$, where $C(0) = \text{var}(h(\mathbf{x}_\theta))$ for $k = s$. Then the estimate for the Monte-Carlo error is:

$$(\sigma^{(i)})^2 \approx \frac{\text{var}(h(\mathbf{x}_\theta))}{N} \underbrace{\sum_{t=-\infty}^{\infty} \rho(t)}_{2\tau_{\text{int}}} = \text{var}(h(\mathbf{x}_\theta)) \frac{2\tau_{\text{int}}}{N}, \quad (2.12)$$

where we define the integrated autocorrelation time (IACT) τ_{int} as in [9] and [], which provides a good estimate on how many steps the sampling algorithm needs to take to produce one independent sample. More specifically, the effective sample size $\frac{2\tau_{\text{int}}}{N}$ gives an estimate of how efficient a sampler is. We calculate the IACT using the Python implementation of [10], accessed [11], which also provides errors on the IACT, which helps us to assess convergence of the chain.

since we expect an exponential decay of the autocorrelate function we can write...

2.2 Sampling Methods

In this section we present the sampling methods used in this thesis and show how these methods draw samples $\mathcal{M} = \{(\mathbf{x}, \boldsymbol{\theta})^{(1)}, \dots, (\mathbf{x}, \boldsymbol{\theta})^{(k)}, \dots, (\mathbf{x}, \boldsymbol{\theta})^{(N)}\} \sim \pi(\mathbf{x}, \boldsymbol{\theta} | \mathbf{y})$ from the desired target distribution, so that we can apply sample-based estimates as in Eq. 2.4. Here, \mathcal{M} denotes a Markov chain, where each new sample $(\mathbf{x}, \boldsymbol{\theta})^{(k)}$ is only affected by the previous one, $(\mathbf{x}, \boldsymbol{\theta})^{(k-1)}$. Markov chain Monte Carlo (MCMC) methods generate such a chain \mathcal{M} using random (Monte Carlo) proposals $(\mathbf{x}, \boldsymbol{\theta})^{(k)} \sim q(\cdot | (\mathbf{x}, \boldsymbol{\theta})^{(k-1)})$ 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 [3, 4].

The ergodicity theorem in [3] 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 [3]). Then the chain converges to the desired target distribution with $\mathcal{M} \sim \pi(\mathbf{x}, \boldsymbol{\theta} | \mathbf{y})$. In practice, one can inspect the trace $\pi(\mathbf{x}^{(k)}, \boldsymbol{\theta}^{(k)} | \mathbf{y})$ for $k = 1, \dots, N$ 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. Nevertheless, we will give a brief overview of the sampling algorithm used.

2.2.1 Sampling from the marginal posterior

As in Eq. 2.5, when using the MTC method we sample from $\pi(\boldsymbol{\theta}|\mathbf{y})$ first and then determine the full conditional $\pi(\mathbf{x}|\mathbf{y})$ as in Eq. 2.7. To sample from $\pi(\boldsymbol{\theta}|\mathbf{y})$, we use a Metropolis-within-Gibbs (MWG) sampler as described in [12]. 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 [13].

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 [13]. We accept and set $\theta_1^{(t)} = \theta_1$ with the acceptance probability

$$\alpha(\theta_1|\theta_1^{(t-1)}) = \min \left\{ 1, \frac{\pi(\theta_1|\theta_2^{(t-1)}, \mathbf{y}) q(\theta_1^{(t-1)}|\theta_1)}{\pi(\theta_1^{(t-1)}|\theta_2^{(t-1)}, \mathbf{y}) q(\theta_1|\theta_1^{(t-1)})} \right\} \quad (2.13)$$

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 $\theta_2^{(t)} \sim \pi(\cdot|\theta_1^{(t)}, \mathbf{y})$, 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 $N = N' - N_{\text{burn-in}}$.

Algorithm 1: Metropolis within Gibbs

```

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

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

5:   Draw  $u \sim \mathcal{U}(0, 1)$ 
6:   if  $\alpha \geq u$  then
7:     Accept and set  $\theta_1^{(t)} = \theta_1$ 
8:   else
9:     Reject and keep  $\theta_1^{(t)} = \theta_1^{(t-1)}$ 
10:  end if
11:  Draw  $\theta_2^{(t)} \sim \pi(\cdot|\theta_1^{(t)}, \mathbf{y})$ 
12: end for
13: Output:  $\boldsymbol{\theta}^{(0)}, \dots, \boldsymbol{\theta}^{(k)}, \dots, \boldsymbol{\theta}^{(N)} \sim \pi(\boldsymbol{\theta}|\mathbf{y})$ 
```

2.2.2 t-walk sampler as 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 algorithm by Christen and Fox [14]. 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.

2.3 Numerical Approximation Methods - Tensor Train

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 $\boldsymbol{x} \in \mathbb{R}^d$. First, we provide a short overview of probability spaces and their associated measures, as a foundation for calculating marginal probability distributions from the tensor train format. Then we explain how we calculate marginal distribution and generate samples via the inverse Rosenblatt transform (IRT). Note that we follow the notation of Cui et al. [15] to introduce this methodology.

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 defined 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 B. We denote

$$\mathbb{P}(A) = \int_A d\mathbb{P} \quad (2.14)$$

as the probability of an event $A \in \mathcal{F}$. By applying the Radon-Nikodym theorem [16], we can change variables

$$\mathbb{P}(A) = \int_A \frac{d\mathbb{P}}{d\boldsymbol{x}} d\boldsymbol{x} = \int_A \pi(\boldsymbol{x}) d\boldsymbol{x}, \quad (2.15)$$

where $d\boldsymbol{x}$ is a reference measure on the same probability space, commonly referred to as the Lebesgue measure. The Radon-Nikodym derivative $\frac{d\mathbb{P}}{d\boldsymbol{x}}$ 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} [17, Chapter 10].

Now, let $X : \Omega \rightarrow \mathbb{R}^d$ 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 . Then the associated PDF $\pi(\boldsymbol{x})$, is a joint density of X , induced by the probability measure on Ω [16, 18]. As in [15], we can define the parameter space as the

Cartesian product $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_d$ with $x_k \in \mathcal{X}_k \subseteq \mathbb{R}$ and $\mathbf{x} = (x_1, \dots, x_k, \dots, x_d)$. The marginal density function for the k -th component is then given by

$$f_{X_k}(x_k) = \int_{\mathcal{X}_1} \cdots \int_{\mathcal{X}_d} \lambda(\mathbf{x}) \pi(\mathbf{x}) dx_1 \cdots dx_{k-1} dx_{k+1} \cdots dx_d, \quad (2.16)$$

where we integrate over all dimensions except the k -th. Here, we introduce a weight function $\lambda(x)$, which can be useful for quadrature rules [19], to which [15] refer to as a "product-form Lebesgue-measurable weighting function" and define it as

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

In the tensor train (TT) format, the integral in Eq. 2.16 for the marginal probability can be computed at a low computational cost as $\pi(\mathbf{x})$ is approximated by

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

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 $\mathbf{x} = (x_1, \dots, x_d)$ 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} and r_k , 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 $\pi(\mathcal{X}) \approx \tilde{\pi}_1 \tilde{\pi}_2 \cdots \tilde{\pi}_d \in \mathbb{R}^d$ 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

$$f_{X_k}(x_k) = \frac{1}{z} \left| \left(\int_{\mathbb{R}} \lambda_1(x_1) \tilde{\pi}_1(x_1) dx_1 \right) \cdots \left(\int_{\mathbb{R}} \lambda_{k-1}(x_{k-1}) \tilde{\pi}_{k-1}(x_{k-1}) dx_{k-1} \right) \right. \\ \left. \lambda_k(x_k) \tilde{\pi}_k(x_k) \right. \\ \left. \left(\int_{\mathbb{R}} \lambda_{k+1}(x_{k+1}) \tilde{\pi}_{k+1}(x_{k+1}) dx_{k+1} \right) \cdots \left(\int_{\mathbb{R}} \lambda_d(x_d) \tilde{\pi}_d(x_d) dx_d \right) \right| \quad (2.17)$$

is computed by integrating over all TT cores except π_k , as in [20], including a normalisation constant z [15].

In practice, tensor train approximations may suffer from numerical instability, in particular because it is not advantageous to approximate the target function $\pi(\mathbf{x})$ in e.g. the logarithmic space. Hence, Cui et al. [15] approximate the square root of the probability density

$$\sqrt{\pi(\mathbf{x})} \approx \sqrt{\tilde{\pi}} = \mathbf{G}_1(x_1), \dots, \mathbf{G}_k(x_k), \dots, \mathbf{G}_d(x_d), \quad (2.18)$$

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

$$G_k^{(\alpha_{k-1}, \alpha_k)}(x_k) = \sum_{i=1}^{n_k} \phi_k^{(i)}(x_k) \mathbf{A}_k[\alpha_{k-1}, i, \alpha_k], \quad k = 1, \dots, d, \quad (2.19)$$

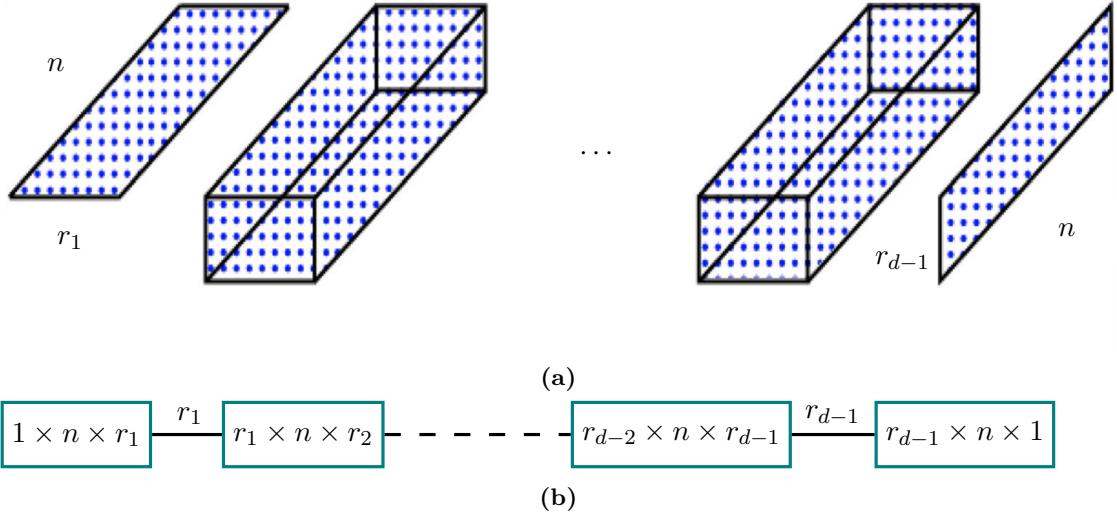


Figure 2.2: Here, we visualise the tensor train cores as two- and three-dimensional matrices. Each core has a length n , corresponding to the number of grid points in one dimension, and the cores are connected through ranks r_k . More specifically, a core $\tilde{\pi}_k$ has dimensions $r_{k-1} \times n \times r_k$, with outer ranks $r_0 = r_d = 1$. Using the TT-format enables us to represent a d -dimensional grid with only dn^2 evaluation points instead of n^d grid points. Figure (a) is adapted from [21].

where $\mathbf{A}_k \in \mathbb{R}^{r_{k-1} \times n_k \times r_k}$ 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 density is written as:

$$\pi(\mathbf{x}) \approx \xi + (\sqrt{\tilde{\pi}})^2(\mathbf{x}), \quad (2.20)$$

where ξ is a positive constant added according to the absolute error and the Lebesgue weighting, see Eq. 2.17, such that

$$0 \leq \xi \leq \frac{1}{\lambda(\mathcal{X})} \|\sqrt{\tilde{\pi}} - \sqrt{\pi}\|_2^2. \quad (2.21)$$

This leads to the normalised target function

$$f_X(x) \approx \frac{1}{z} (\lambda(x)\xi + \lambda(x)\tilde{\pi}(x)), \quad (2.22)$$

where z is the normalisation constant. Given the tensor train approximation of $\sqrt{\pi}$, the marginal function $f_{X_k}(x_k)$ can be expressed as

$$\begin{aligned} f_{X_k}(x_k) &\approx \frac{1}{z} \left(\xi \prod_{i=1}^{k-1} \lambda_i(\mathcal{X}_i) \prod_{i=k+1}^d \lambda_i(\mathcal{X}_i) \right. \\ &\quad + \left(\int_{\mathbb{R}} \lambda_1(x_1) \mathbf{G}_1^2(x_1) dx_1 \right) \cdots \left(\int_{\mathbb{R}} \lambda_{k-1}(x_{k-1}) \mathbf{G}_{k-1}^2(x_{k-1}) dx_{k-1} \right) \\ &\quad \lambda_k(x_k) \mathbf{G}_k^2(x_k) \\ &\quad \left. \left(\int_{\mathbb{R}} \lambda_{k+1}(x_{k+1}) \mathbf{G}_{k+1}^2(x_{k+1}) dx_{k+1} \right) \cdots \left(\int_{\mathbb{R}} \lambda_d(x_d) \mathbf{G}_d^2(x_d) dx_d \right) \right). \end{aligned} \quad (2.23)$$

2.3.1 Marginal Functions for quadrature

Marginal function are handy when approximation integrals and to compute these marginals efficiently, one can use a procedure similar to left and right orthogonalisation of TT-cores [22]. Cui et al. [15] referred to this backwards marginalisation, see Prop. 2, to which I add the forward marginalisation, see Prob. 1. The backwards marginalisation provides us with the coefficient matrices \mathbf{B}_k , while the forward marginalisation gives the coefficient matrices $\mathbf{B}_{\text{pre},k}$. 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 [15]. For this, we define the mass matrix $\mathbf{M}_k \in \mathbb{R}^{n_k \times n_k}$ as

$$\mathbf{M}_k[i, j] = \int_{\mathcal{X}_k} \phi_k^{(i)}(x_k) \phi_k^{(j)}(x_k) \lambda(x_k) dx_k, \quad i, j = 1, \dots, n_k, \quad (2.24)$$

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 \mathbf{B}_k , stated in Proposition 1, is adapted directly from [15].

Proposition 1 (Backwards Marginalisation as in [15]): Starting with the last coordinate $k = d$, we set $\mathbf{B}_d = \mathbf{A}_d$. The following procedure can be used to obtain the coefficient tensor $\mathbf{B}_{k-1} \in \mathbb{R}^{r_{k-2} \times n_{k-1} \times r_{k-1}}$, which we need for defining the marginal function $f_{X_k}(x_k)$:

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

$$\mathbf{C}_k[\alpha_{k-1}, \tau, l_k] = \sum_{i=1}^{n_k} \mathbf{B}_k[\alpha_{k-1}, i, l_k] \mathbf{L}_k[i, \tau]. \quad (2.25)$$

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

$$\mathbf{Q}_k \mathbf{R}_k = (\mathbf{C}_k^{(R)})^\top. \quad (2.26)$$

3. Compute the new coefficient tensor:

$$\mathbf{B}_{k-1}[\alpha_{k-2}, i, l_{k-1}] = \sum_{\alpha_{k-1}=1}^{r_{k-1}} \mathbf{A}_{k-1}[\alpha_{k-2}, i, \alpha_{k-1}] \mathbf{R}_k[l_{k-1}, \alpha_{k-1}]. \quad (2.27)$$

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

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

$$\mathbf{C}_{\text{pre},k}[\alpha_{k-1}, \tau, l_k] = \sum_{i=1}^{n_k} \mathbf{L}_k[i, \tau] \mathbf{B}_{\text{pre},k}[\alpha_{k-1}, i, l_k]. \quad (2.28)$$

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

$$\mathbf{Q}_{\text{pre},k} \mathbf{R}_{\text{pre},k} = (\mathbf{C}_{\text{pre},k}^{(R)}). \quad (2.29)$$

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

$$\mathbf{B}_{\text{pre},k+1}[l_{k+1}, i, \alpha_{k+1}] = \sum_{\alpha_k=1}^{r_k} \mathbf{R}_{\text{pre},k}[l_{k+1}, \alpha_k] \mathbf{A}_{k+1}[\alpha_k, i, \alpha_{k+1}]. \quad (2.30)$$

After computing the coefficient tensors $\mathbf{B}_{\text{pre},k+1}$ as in Prop. 2 and \mathbf{B}_{k+1} from Prop. 1, the marginal PDF of k -th dimension can be expressed as

$$f_{X_k}(x_k) \approx \frac{1}{z} \left(\xi \prod_{i=1}^{k-1} \lambda_i(X_i) \prod_{i=k+1}^d \lambda_i(X_i) + \sum_{l_{k-1}=1}^{r_{k-1}} \sum_{l_k=1}^{r_k} \left(\sum_{i=1}^n \phi_k^{(i)}(x_k) \mathbf{D}_k[l_{k-1}, i, l_k] \right)^2 \right) \lambda_k(x_k), \quad (2.31)$$

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

$$\mathbf{D}_k[l_{k-1}, i, l_k] = \sum_{\alpha_{k-1}=1}^{r_{k-1}} \mathbf{R}_{\text{pre},k-1}[l_{k-1}, \alpha_{k-1}] \mathbf{B}_k[\alpha_{k-1}, i, l_k]. \quad (2.32)$$

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

$$f_{X_1}(x_1) \approx \frac{1}{z} \left(\xi \prod_{i=2}^d \lambda_i(\mathcal{X}_i) + \sum_{l_1=1}^{r_1} \left(\sum_{i=1}^n \phi_1^{(i)}(x_1) \mathbf{D}_1[i, l_1] \right)^2 \right) \lambda_1(x_1), \quad (2.33)$$

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

$$f_{X_d}(x_d) \approx \frac{1}{z} \left(\xi \prod_{i=1}^{d-1} \lambda_i(\mathcal{X}_i) + \sum_{l_{n-1}=1}^{r_{d-1}} \left(\sum_{i=1}^n \phi_1^{(i)}(x_1) \mathbf{D}_d[l_{n-1}, i] \right)^2 \right) \lambda_d(x_d), \quad (2.34)$$

where $\mathbf{D}_d[l_{n-1}, i] = \mathbf{B}_{\text{pre},d}[l_{n-1}, i, \alpha_{n+1}]$ 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 the TT approximation – Squared Inverse Rosenblatt Transform

If instead of evaluating integrals we draw samples from the approximated function via the inverse Rosenblatt transform (IRT), as in [20], to preserve the correlation structure. Since we approximate the square root of the target function Cui et. al. [15] call that the squared inverse Rosenblatt transform (SIRT).

In doing so we start by calculating the Backward marginals as in Prob 1 and draw N uniformly distributed seeds $\{\mathbf{u}^{(1)}, \dots, \mathbf{u}^{(N)}\} \sim \mathcal{U}(0, 1)^d$, where each $\mathbf{u}^{(s)}$ is d-dimensional for $s = 1, \dots, N$. Then we calculate the first marginal $f_{X_1}(x_1)$ as in Eq. 2.33 and normalise with $z = \int_{\mathcal{X}_1} f_{X_1}(x_1) dx_1$. Next we compute the cumulative distribution function (CDF)

$$F_{X_k}(x_k) \approx \int_{-\infty}^{x_k} f_{X_k|X_{<k}}(\tilde{x}_k|x_{k-1}, \dots, x_1) d\tilde{x}_k \quad (2.35)$$

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

$$\mathbf{G}_k(x_k^{(s)}) \approx \frac{x_k^{(s)} - x_k^{(i)}}{x_k^{(i+1)} - x_k^{(i)}} \mathbf{G}_k(x_k^{(i+1)}) + \frac{x_k^{(i+1)} - x_k^{(s)}}{x_k^{(i+1)} - x_k^{(i)}} \mathbf{G}_k(x_k^{(i)}) \quad (2.36)$$

for $x_k^{(i)} \leq x_k^{(s)} \leq x_k^{(i+1)}$ in between two grid points i and $i + 1$ as in [20]. Through $\mathbf{G}_k(x_k^{(s)}) \in \mathbb{R}^{1 \times r_{k-1}}$ 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 \mathbf{B}_k so that the next "conditional marginal" is given as:

$$f_{X_k|X_{<k}}(x_k|x_{k-1}^{(s)}, \dots, x_1^{(s)}) \approx \frac{1}{z} \left(\xi \prod_{i=1}^{k-1} \lambda_i(X_i) \prod_{i=k+1}^d \lambda_i(X_i) + \sum_{l_k=1}^{r_k} \left(\sum_{i=1}^n \phi_k^{(i)}(x_k^{(s)}) \left(\sum_{\alpha_{k-1}=1}^{r_{k-1}} \mathbf{G}_{<k}^{(\alpha_{k-1})}(x_{<k}^{(s)}) \mathbf{B}_k[\alpha_{k-1}, i, l_k] \right) \right)^2 \right) \lambda_k(x_k) \quad (2.37)$$

We repeat the procedure for each $u_k^{(s)} \in \mathbf{u}^{(s)}$ to gain samples $\mathbf{x}^{(s)} \sim f_X(x)$, see algorithmic box 2 for a summarised version.

Note that with Cartesian basis $\sum \phi_k^{(i)}(x_k) \left(\sum \mathbf{G}_{<k}^{(\alpha_{k-1})}(x_{<k}) \mathbf{B}_k[\alpha_{k-1}, i, l_k] \right)^2$ and $\mathbf{G}_{<k}(x_{<k}^{(s)}) = \mathbf{G}_1(x_1^{(s)}) \cdots \mathbf{G}_{k-1}(x_{k-1}^{(s)})$ are simple matrix multiplications for each grid point i or sample $\mathbf{x}^{(s)}$.

Algorithm 2: Squared Inverse Rosenblatt Transform

```

1: Input: seeds  $\{\mathbf{u}^{(1)}, \dots, \mathbf{u}^{(N)}\} \sim \mathcal{U}(0, 1)^d$  and  $\mathbf{B}_1, \dots, \mathbf{B}_d$  from Prob. 1
2: for  $s = 1, \dots, N$  do
3:   for  $k = 1, \dots, d$  do
4:     compute normalised PDF  $f_{X_k|X_{<k}}(x_k|x_{k-1}^{(s)}, \dots, x_1^{(s)})$ , Eq. 2.37
5:     compute cumulative distribution function  $F_{X_k}(x_k)$ , Eq. 2.35
6:     project sample  $x_k^{(s)} = F_{X_k}^{-1}(u_k^{(s)})$ 
7:     interpolate  $\mathbf{G}_k(x_k^{(s)})$ , Eq. 2.36
8:     update  $\mathbf{G}_{\leq k}(x_{\leq k}^{(s)}) = \mathbf{G}_{<k}(x_{<k}^{(s)})\mathbf{G}_k(x_k^{(s)})$ 
9:   end for
10: end for
11: Output: samples  $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}\}$ , where each  $\mathbf{x}^{(s)} \in \mathbb{R}^d$  for  $s = 1, \dots, N$ 

```

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 $\mathbf{M} : \mathbf{A}_L \mathbf{x} \rightarrow \mathbf{A}_{NL} \mathbf{x}$, which approximates the non-linear model using the linear model. 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 [].

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 [23, 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 [23, 24]

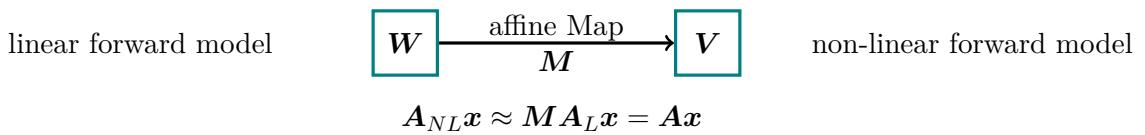


Figure 2.3: This Figure shows the schematic representation of the affine map \mathbf{M} , which approximates the non-linear forward model from the linear forward model. Here, V contains values produced by the linear forward model, and W contains the corresponding values from the non-linear forward model.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

Consequently, to map between the linear and non-linear forward map, we generate two affine subspaces V and W over the same field. Assume we have noise free data vector

$A_{NL}\mathbf{x} \in \mathbb{R}^m$, then the subspace associated with the linear forward model is

$$\mathbf{W} = \begin{bmatrix} | & | & | \\ \mathbf{A}_L\mathbf{x}^{(1)} & \dots & \mathbf{A}_L\mathbf{x}^{(j)} & \dots & \mathbf{A}_L\mathbf{x}^{(m)} \\ | & & | & & | \end{bmatrix} \in \mathbb{R}^{m \times m} \quad (2.38)$$

and with the non-linear forward model is

$$\mathbf{V} = \begin{bmatrix} | & | & | \\ \mathbf{A}_{NL}\mathbf{x}^{(1)} & \dots & \mathbf{A}_{NL}\mathbf{x}^{(j)} & \dots & \mathbf{A}_{NL}\mathbf{x}^{(m)} \\ | & & | & & | \end{bmatrix} = \begin{bmatrix} | & v_1 & | \\ & \vdots & \\ | & v_j & | \\ & \vdots & \\ | & v_m & | \end{bmatrix} \in \mathbb{R}^{m \times m}, \quad (2.39)$$

Here $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}\}$ are m different parameters.

Then we find the affine map

$$\mathbf{V}\mathbf{W}^{-1} = \mathbf{M} = \begin{bmatrix} | & r_0 & | \\ & \vdots & \\ | & r_j & | \\ & \vdots & \\ | & r_m & | \end{bmatrix} \in \mathbb{R}^{m \times m}, \quad (2.40)$$

row wise by solving $v_j = r_j\mathbf{W}$ for r_j , so that $\mathbf{A} = \mathbf{M}\mathbf{A}_{NL} \approx \mathbf{A}_{NL}$. Alternately one could also compute \mathbf{M} using the inverse \mathbf{W}^{-1} .

2.5 Regularisation

As mentioned in the introduction, the currently most used method to analyse any data in atmospheric physics is regularisation. Since we want to show that our methods are computationally comparable if not faster, and provide more information than regularisation, we choose a regulariser closest to our linear-Gaussian Bayesian framework, see section 4.2.

The Tikhonov approach provides one solution \mathbf{x}_λ that minimises both the data misfit norm

$$\|\mathbf{y} - \mathbf{Ax}\| \quad (2.41)$$

and a regularisation semi-norm

$$\lambda \|\mathbf{T}\mathbf{x}\| \quad (2.42)$$

, as described in [12], with a linear forward model matrix \mathbf{A} , the data \mathbf{y} and a regularisation operator \mathbf{T} and the regularisation parameter $\lambda > 0$ which penalise \mathbf{x} accordingly. For a fixed λ , the regularised solution

$$\mathbf{x}_\lambda = \arg \min_{\mathbf{x}} \|\mathbf{y} - \mathbf{Ax}\|^2 + \lambda \|\mathbf{T}\mathbf{x}\|^2 \quad (2.43)$$

is obtained by taking the derivative with respect to \mathbf{x} of the objective function:

$$\nabla_{\mathbf{x}} \left\{ (\mathbf{y} - \mathbf{Ax})^T (\mathbf{y} - \mathbf{Ax}) + \lambda \mathbf{x}^T \mathbf{T}^T \mathbf{T} \mathbf{x} \right\} = 0 \quad (2.44)$$

$$\iff \nabla_{\mathbf{x}} \left\{ \mathbf{y}^T \mathbf{y} + \mathbf{x}^T \mathbf{A}^T \mathbf{Ax} - 2 \mathbf{y}^T \mathbf{Ax} + \lambda \mathbf{x}^T \mathbf{T}^T \mathbf{T} \mathbf{x} \right\} = 0 \quad (2.45)$$

$$\iff 2 \mathbf{A}^T \mathbf{Ax} - 2 \mathbf{A}^T \mathbf{y} + 2 \lambda \mathbf{T}^T \mathbf{T} \mathbf{x} = 0, \quad (2.46)$$

or equivalently the "regularised normal equations" $\mathbf{A}^T \mathbf{y} = \mathbf{A}^T \mathbf{Ax} + \lambda \mathbf{T}^T \mathbf{T} \mathbf{x}$ [25]. Solving this equation yields the regularised solution

$$\mathbf{x}_\lambda = (\mathbf{A}^T \mathbf{A} + \lambda \mathbf{L})^{-1} \mathbf{A}^T \mathbf{y}, \quad (2.47)$$

where we define $\mathbf{L} := \mathbf{T}^T \mathbf{T}$, which typically represents a discrete matrix approximation of a differential operator choice [3].

With \mathbf{T} may represent the first order discrete derivative then the \mathbf{L} is the second order discrete derivative.

To show how the regularisation parameter effects the solution one can do a singular value decomposition of A and the generalised singular value decomposition of A, T as in [26]. Then $\mathbf{A} = \mathbf{U} \Lambda_A \mathbf{M}^{-1}$ and $\mathbf{L} = \mathbf{V} [\Lambda_L, 0] \mathbf{M}^{-1}$ Where the general singular values $\Lambda_A = \text{diag}(\sigma_{A,1}, \dots, \sigma_{A,1}, 1, \dots, 1)$ and $\Lambda_L = \text{diag}(\sigma_{L,1}, \dots, \sigma_{L,1}, 1, \dots, 1)$

Then one can show that the solution is with filter factors

$$f_i = \frac{(\sigma_{A,i}/\sigma_{L,i})^2}{(\sigma_{A,i}/\sigma_{L,i})^2 + \lambda^2} \approx \begin{cases} \frac{(\sigma_{A,i}/\sigma_{L,i})^2}{\lambda^2}, & \sigma_{A,i}/\sigma_{L,i} \gg \lambda \\ 1, & \sigma_{A,i}/\sigma_{L,i} \ll \lambda \end{cases} \text{ for } i = 1, \dots, p \quad (2.48)$$

Then small singular values depend on prior information only for large $(\sigma_{A,i}/\sigma_{L,i})^2$ singular values the solution is unaffected for small $(\sigma_{A,i}/\sigma_{L,i})^2$ singular values the solution is affected by the regularisation parameter

In practice, \mathbf{x}_λ is computed for a range of λ -values and evaluated based on the trade-off between the data misfit and the regularisation norm. The optimal value of λ is often chosen as the point of maximum curvature on the so-called L-curve [27], which we plot in Fig. 4.14. Additionally one can think about it in [12] [28, Fig. 2.13]

3

Forward Model

In this chapter we present the forward model to which we apply all our methodology on. We follow the MIPAS handbook [29] 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.

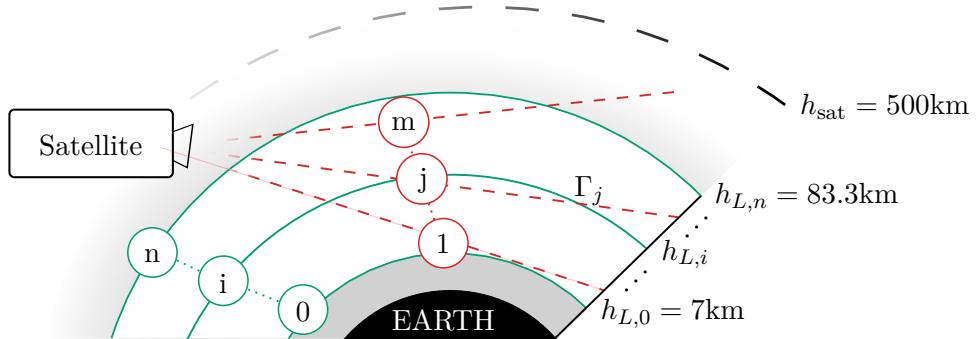


Figure 3.1: Schematic of measurement and analysis geometry, not to scale. The stationary satellite, at a constant height h_{sat} above Earth, takes $m = 41$ measurements along its line-of-sight defining by the line Γ_j . Each measurement has a limb height ℓ_j , $j = 1, 2, \dots, m$ defined as the closest distance of Γ_j to the Earth surface. Between $h_{L,0} = 7\text{km}$ and $h_{L,n} = 83.3\text{km}$, the stratosphere is discretised into $n = 44$ 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 line of sight, see Figure 3.1. One measurement of the thermal radiation if we target one specific molecule, in our case ozone denoted by the ozone volume mixing ratio $x(r)$ at distance r from the satellite,

of at the wave number ν is given by the path integral

$$y_j = \int_{\Gamma_j} B(\nu, T) k(\nu, T) \frac{p(T)}{k_B T(r)} x(r) \tau(r) dr + \eta_j \quad (3.1)$$

$$\tau(r) = \exp \left\{ - \int_{r_{\text{obs}}}^r k(\nu, T) \frac{p(T)}{k_B T(r')} x(r') dr' \right\}, \quad (3.2)$$

which is the radiative transfer equation (RTE) [29] where we define a tangent height h_{ℓ_j} and a pointing direction is Γ_j for each $j = 1, 2, \dots, m$ measurement of the data vector $\mathbf{y} \in \mathbb{R}^m$ including some noise η_j . Within the atmosphere the number density $p(T)/(k_B T(r))$ 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

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

is depend on the line intensity $L(\nu, T_{\text{ref}})$ at reference temperature $T_{\text{ref}} = 296K$, the lower-state energy of the transition E'' , the second radiation constant $c_2 = 1.4387769\text{cmK}$ all provided by the HITRAN database [30]. Since we assume that the measurement deceive as negligible frequency window we neglect line broadening around ν_0 the calculations of $L(\nu, T_{\text{ref}})$ normally include broadening modelled as a convolution of the normalized Lorentz profile (collisional/pressure broadening) and the normalised Doppler (thermal broadening) profile [31]. Additionally we note that we simplify the calculation of $k(\nu, T)$, which usually the sum the individual absorption constant for each targeted molecule weighted by the respective volume mixing ratio [31]. Then the total internal partition function for the lower-state energy is

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

with the statistical weight g'' (also called the degeneracy factor) accounting for the molecules non-rotational and rotational energy states, see [31]. Under the assumption of local thermodynamic equilibrium (LTE) the black body radiation act as a source function

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

with Planck's constant h and velocity of light c [31]. For fundamentals on the Radiative transfer equation we recommend [32, Chapter 1].

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 $i = 1, \dots, n$, with $h_{L,0}$ and $h_{L,n}$. Then we can discretise the ozone, pressure and temperature profiles as a function of height, where 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. Depending on the parameter of interest, which is either the ozone volume mixing ratio $\mathbf{x} = \{x_1, x_2, \dots, x_n\} \in \mathbb{R}^n$ or the fraction of pressure and temperature $\mathbf{p}/\mathbf{T} = \{p_1/T_1, p_2/T_2, \dots, p_n/T_n\} \in \mathbb{R}^n$ we rewrite the integral in Eq. (3.1) for one noise free measurement using the trapezoidal rule as a vector-vector multiplication $\mathbf{A}_j(\mathbf{x}, \mathbf{p}, \mathbf{T}) \mathbf{x}$ or $\mathbf{A}_j(\mathbf{x}, \mathbf{p}, \mathbf{T}) \mathbf{p}/\mathbf{T}$, where the non-linear absorption $\tau(r)$ is included in $\mathbf{A}_j(\mathbf{x}, \mathbf{p}, \mathbf{T}) \in \mathbb{R}^n$ which is the j -th row of the matrix $\mathbf{A}(\mathbf{x}, \mathbf{p}, \mathbf{T})$. Then given a noise vector $\boldsymbol{\eta} \in \mathbb{R}^m$ the data vector

$$\mathbf{y} = \mathbf{A}_{NL} \mathbf{x} + \boldsymbol{\eta} = \mathbf{A}_{NL} \frac{\mathbf{p}}{\mathbf{T}} + \boldsymbol{\eta} \quad (3.6)$$

is based on a matrix-vector multiplication, where we define $\mathbf{A}(\mathbf{x}, \mathbf{p}, \mathbf{T}) \equiv \mathbf{A}_{NL} \in \mathbb{R}^{m \times n}$ for simplicity so that $\mathbf{A}_{NL}\mathbf{x}$ or $\mathbf{A}_{NL}\mathbf{p}/\mathbf{T}$ implies the construction of \mathbf{A}_{NL} . If we neglect the absorption, e.g. set $\tau = 1$ in Eq. (3.2), this problem becomes a linear problem with the forward model given by $\mathbf{A}_L\mathbf{x}$ or $\mathbf{A}_L\mathbf{p}/\mathbf{T}$. Further, we classify the inverse problem as weakly non-linear, see e.g. Fig. 4.13, as neglecting the absorption changes the measurement only slightly.

3.1 Singular value decomposition of linear forward model matrix

In this section we want to give an intuitive way of how we can measure most effectively depending on the information passed through the forward model and how the signal to noise ratio affects that information. One way of doing this is via a singular value decomposition (SVD) of the forward model matrix

$$\mathbf{A} = \sum_{i=1}^r \mathbf{u}_i \sigma_i \mathbf{v}_i^T = \mathbf{U} \Sigma \mathbf{V}^T \quad (3.7)$$

where $r = \min\{m, n\}$ for $\mathbf{A} \in \mathbb{R}^{m \times n}$. One way of thinking about the SVD when considering a parameter \mathbf{x} , which is passed through the forward model as in $\mathbf{A}\mathbf{x}$, is that the right singular vectors \mathbf{v}_i span the parameter space. So if the right singular vectors are flat in height altitudes the forward model does not pick up any structure of the parameter in that region. Then the right singular vectors are weighted with the singular values σ_i , which are ordered in size from the largest σ_1 to the smallest σ_r singular value. Lastly, the left singular vectors \mathbf{u}_i project to the data space. See [3] for a more comprehensive analysis.

If we consider very small singular values below a threshold negligible we can introduce an effective rank $r_{\text{eff}} \leq r$, then $\{\mathbf{v}_{r_{\text{eff}}+1}, \dots, \mathbf{v}_r\}$ spans the null space, see Figure ???. This

threshold may be affected by noise in the data. If the rough assumption that the maximum singular value $\max(y) \approx \sigma_1$ holds and the signal-to-noise-ratio (SNR) is defined as

$$\text{SNR} := \frac{\max(y)}{\text{std. noise}} \quad (3.8)$$

then we can eye ball the information transmitted through the forward model, which are roughly the singular values $s_i \gtrsim \max(y)/\text{SNR}$.

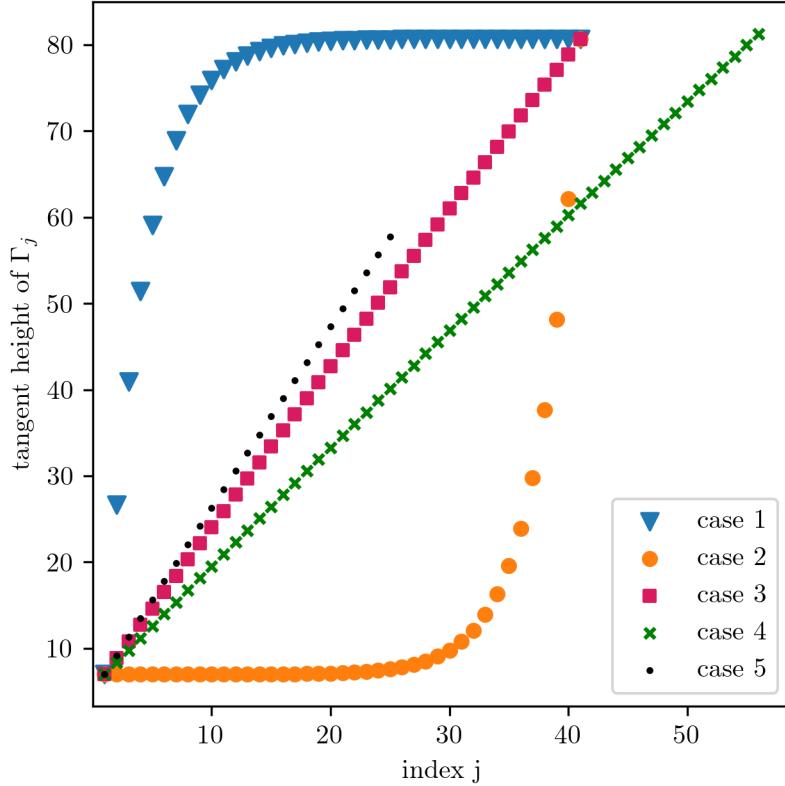


Figure 3.2: We plot the tangent heights for different cases of measurements.

Next we analyse the singular values for a few different measurement scenarios, to see which of those measurement cases is most effective. We test for different pointing accuracies, which determines how well the satellite can point in a certain direction, and we correlate that to the number of measurements. The singular values for measurements with equally distanced tangent heights in between a set atmosphere, see case 3 in Fig. 3.2, are plotted in Fig. 3.3. The pointing accuracy of 150arc sec, was given to us by the team in [34]. We can see that value of the singular values are linearly decreasing in log space (exponentially in normal space) and that we can roughly capture information of the first 15 to 20 singular values. In comparison, if we measure with exponentially decreasing pointing accuracy, see case 1 Fig. 3.2, or increasing pointing accuracy, see case 2 Fig. 3.2, the forward model does not give us more information. We actually get

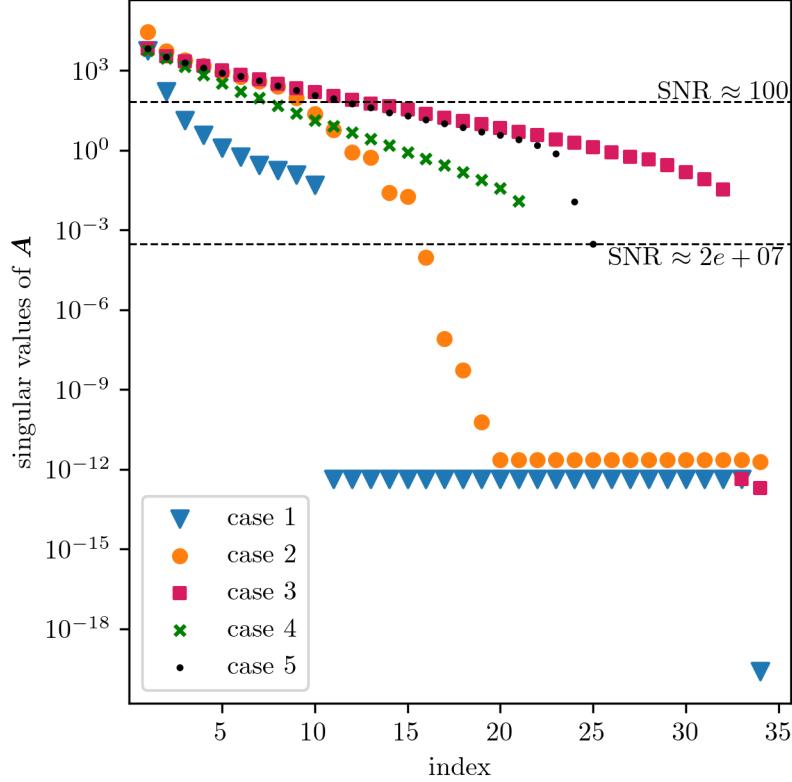


Figure 3.3: We plot the singular values of linear forward model matrix for different sequences of measurements. The corresponding tangent heights of the different cases are plotted in Fig. 3.2. We include an approximate for the disiered Signal to noise ratio if $s_1 \approx \max(\mathbf{y})$ the signal.

fewer large singular values for case 2 and more very small singular values as we measure more in higher altitude where noise is dominant due to decreasing pressure and hence density. Similarly case 2 does also not seem better than case 3, as we do get some larger singular values which then decease rapidly. Also if we would half the pointing accuracy and double the number of measurements, case 4 in Fig. 3.2, we do not increase the amount of information through our forward model. Since I assume it is easier to build a satellite with lower pointing accuracy we find through exploratory analysis that we can tolerate a pointing accuracy of 150arc sec without much loss of information above a SNR of around 100. And since we see that measuring more does not give more information we only measure until around 60 km in height, where the data start to get dominated by noise. Consequently ozone values at higher altitudes are not determined by data but by the prior and given these singular values we can show quite clearly the ill-posedness of this model/problem. Of course, if we increase the SNR by a factor of 10^5 we would be able to see ozone peaks around 60km but that does not seem realistic.

Consequently we proceed with case 5 and plot the parameter space of the model for the first 20 of 25 or so right singular vectors in Fig. 3.4. We observe that we do not

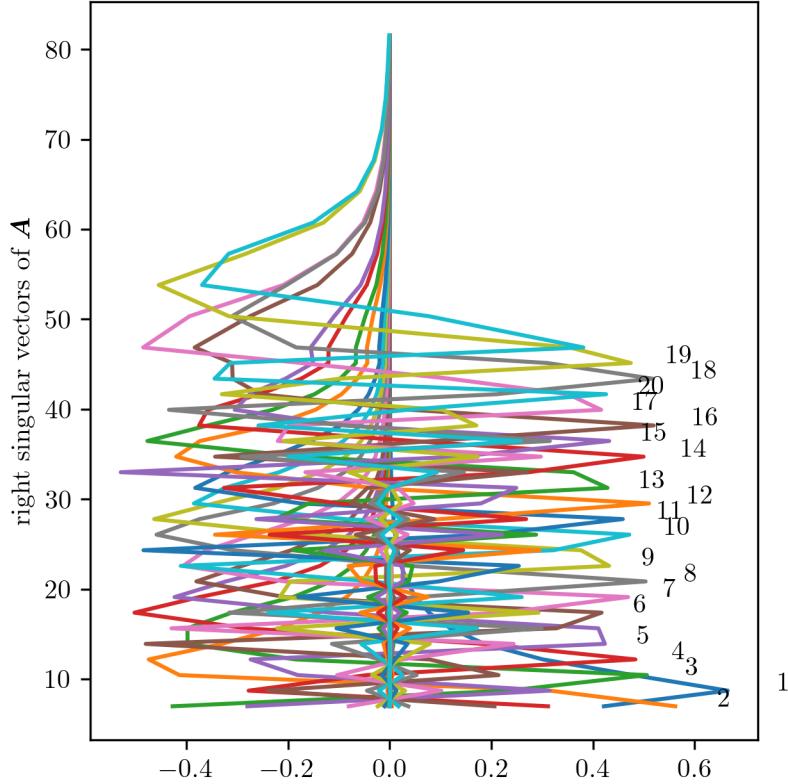


Figure 3.4: We plot the first 20 left singular vectors of forward model matrix for case 5 sequence of measurements, see Fig. 3.3.

Figure 3.5

pick up structures above 60km. The last 5 right singular vectors (null space) include structures above 60 km and hence our model is not sensitive to those.

4

Results

In this chapter, we use the forward model to generate data given an underlying ground truth and then guide the reader towards obtaining the posterior distributions. Once we simulated some data, we established a Bayesian framework in Sec. 4.2, where we discuss the choice of prior distributions and formulate the posterior distributions for ozone and pressure over temperature, respectively. Since our forward model is weakly non-linear we like to approximates the non-linear forward model $\mathbf{A}_{NL} \approx \mathbf{M}\mathbf{A}_{NL}$ with an affine map \mathbf{M} , see Sec. 4.3. In doing so, we sample from the marginal posterior for ozone and compare that to the tensor-train (TT) approximation. Then we calculate the mean and the covariance matrix of the full conditional posterior and use the obtained posterior ozone samples to create two affine subspaces, to which we map in between. Finally, we calculate the posterior distribution for ozone and pressure over temperature with the updated forward map $\mathbf{A} = \mathbf{M}\mathbf{A}_{NL}$ and compare to a ground truth. Lastly, we evaluate some errors occurring during the process. All programming and analysis are done in Python on 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 profile generated from some data [2] of the microwave limb sounder on the aura satellite in the Antarctic region with a peak in high altitude (to show that the data is uninformative in those regions), see Fig. 4.12. The ozone profile from [2] provides ozone volume mixing ratios versus pressure, so we recursively calculate the geometric height with the hydrostatic equilibrium equation

$$\frac{dp}{p} = \frac{-gM}{R^*T} dh, \quad (4.1)$$

subscript i	geometric height h_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.

with the acceleration due to gravity

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

where the polar radius pf the earth is $r_0 \approx 6356$ km, the gravitation at sea level is $g_0 \approx 9.81$ m/s², $R^* = 8.31432 \times 10^{-3}$ Nm/kmol/K and the mean molecular weight of the air is $M = 28.97$ kg/kmol [33]. This holds up to a geometric height of 86km, where ignore a 0.04% change in M from 80km to 86km in geometric altitude.

Following [33] we form a temperature function

$$T(h) = \begin{cases} T_0 & , \quad h = 0 \\ T_0 + a_0 h & , \quad 0 \leq h < h_1 \\ T_0 + a_0 h_1 & , \quad h_1 \leq h < h_2 \\ T_0 + a_0 h_1 + a_1(h_2 - h_1) + a_2(h - h_2) & , \quad h_2 \leq h < h_3 \\ T_0 + a_0 h_1 + a_1(h_2 - h_1) + a_2(h_3 - h_2) + a_3(h - h_3) & , \quad h_3 \leq h < h_4 \\ T_0 + a_0 h_1 + a_1(h_2 - h_1) + a_2(h_3 - h_2) + a_3(h_4 - h_3) \\ \quad + a_4(h - h_4) & , \quad h_4 \leq h < h_5 \\ T_0 + a_0 h_1 + a_1(h_2 - h_1) + a_2(h_3 - h_2) \\ \quad + a_3(h_4 - h_3) + a_4(h_5 - h_4) + a_5(h - h_5) & , \quad h_5 \leq h < h_6 \\ T_0 + a_0 h_1 + a_1(h_2 - h_1) + a_2(h_3 - h_2) \\ \quad + a_3(h_4 - h_3) + a_4(h_5 - h_4) + a_5(h_6 - h_5) & , \quad h_6 \leq h \lesssim 86 \end{cases}$$

with gradient and height values provided by [33], see Tab. 4.1, which act as the ground truth temperature, see Fig. 4.5.

Then we can compute a data vector \mathbf{y} , with $m = 42$ measurements according to the radiative transfer equation (RTE), see Eq. 3.1, determined by the satellite pointing accuracy of 150arcsec as requested by the internal report of the proposed cube-satellite

[34], within an atmosphere $h_{L,0} = 7\text{km}$ and $h_{L,n} = 83.3\text{km}$ with $n = 45$ layers. The height values $h_{L,i}$ for each layer $i = 0, \dots, n$ are defined by the ozone profile from [2] and its pressure values. We target thermal radiation at a wave number $\nu = 7.86\text{cm}^{-1}$, equal to a frequency of roughly 235GHz, where we assume that ozone is the only emitter at that frequency, see [], and calculate the absorption constant $k(\nu, T)$ as in Eq. 3.2, following the HITRAN database [30], which provides the line intensity $L(\nu, T_{\text{ref}})$ for the isotopologue $^{16}\text{O}_3$ with the AFGL Code 666. Lastly we add normally distributed $\nu \sim \mathcal{N}(0, \gamma^{-1} \mathbf{I})$ noise so that we have a voltage Signal-to-Noise (SNR) of 60, similar to THz module on the MLS aura satellite [35].

$$\text{SNR} = \frac{\max(y)}{\text{stdnoise}} = \frac{\max(y)}{\text{stdnoise}} \quad (4.3)$$

Note that we multiply the data with 10^{11} to avoid numerical issues within the inversion process. Can also see that when we plot the data . Now, given the data, we want to

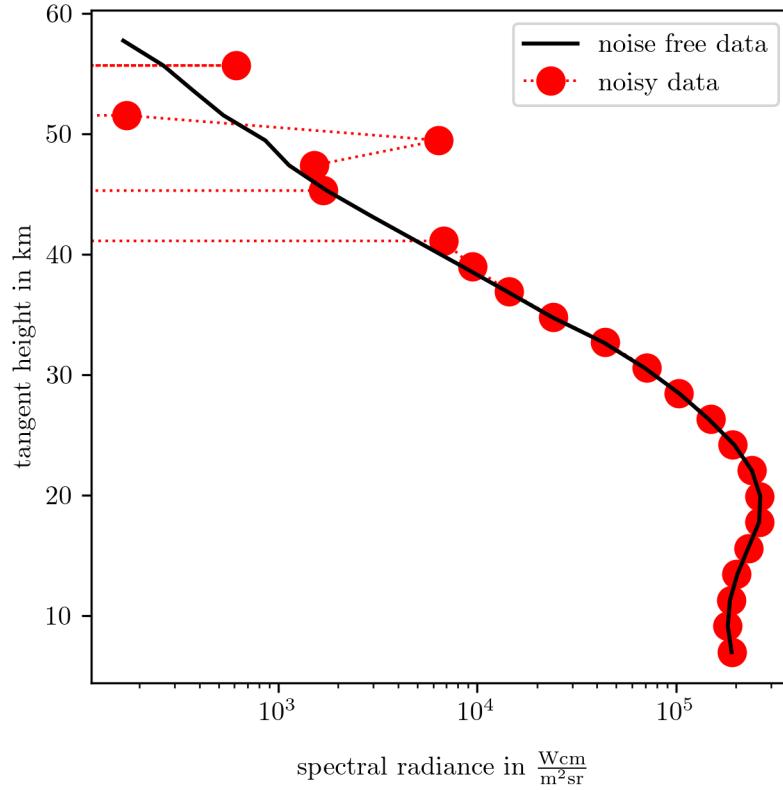


Figure 4.1: Logarithmic plot of data points at different tangent height. Note that negative values are not appearing and we see that at a certain height noise is dominating.

invert the process and determine a posterior distribution over the ozone \mathbf{x} , pressure \mathbf{p} and temperature \mathbf{T} .

4.2 Set up the Bayesian framework

Since the forward model described in Ch. 3 is weakly non-linear we will set up a linear Bayesian hierarchical framework first based on the linear forward model \mathbf{A}_L and then later the approximated version $\mathbf{A}_{NL} \approx \mathbf{M}\mathbf{A}_L$. Furthermore, the noise is normally distributed, so we establish a linear-Gaussian Bayesian hierarchical framework, aiming to recover an ozone profile and a pressure over temperature profile. In doing so, we first draw a directed acyclic graph (DAG) to visualise the measurement and modelling process and determine hyper-parameters and correlations between parameters. Then we define prior distributions over all parameters as well as a likelihood function so that we can formulate the posterior distribution.

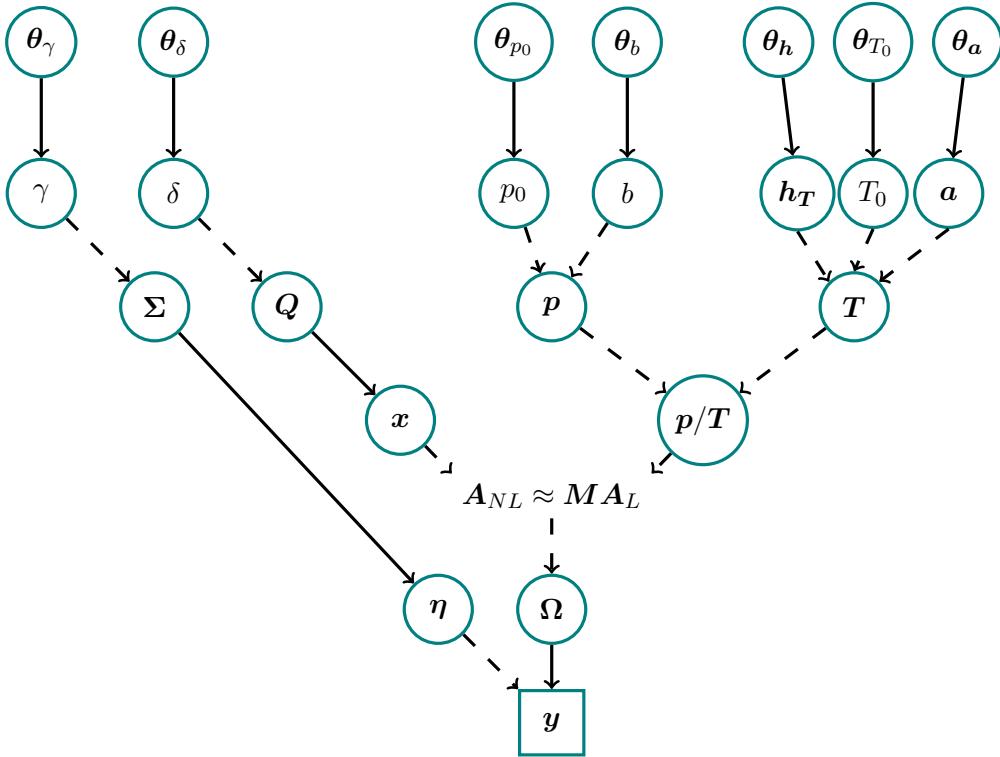


Figure 4.2: Complete directed acyclic graph of the forward model. The hyper-parameters at the top deterministically (dotted line) describe the parameters (\mathbf{p}/\mathbf{T}) or the noise covariance $\Sigma = \gamma^{-1} \mathbf{I}$ of the random (solid line) noise $\boldsymbol{\eta} \sim \mathcal{N}(0, \gamma^{-1} \mathbf{I})$ and precision matrix $\mathbf{Q} = \delta \mathbf{L}$ of the distribution of $\mathbf{x} \sim \mathcal{N}(0, \delta \mathbf{L})$, where \mathbf{L} is a graph Laplacian as in Eq. 4.7. We can group the noise precision γ and the smoothness parameter δ to define the marginal posterior over those hyper-parameters and then condition on them for the conditional posterior distribution, for further details see Fig. C.19. In this whole process where we condition on the pressure \mathbf{p} and temperature \mathbf{T} , which we retrieve separately, see Fig. ???. The hyper-parameters h_0, p_0, b deterministically describe the pressure function in Eq. 4.17, note that we only need three parameters here since $h_0 < h_{L,0}$ and $\mathbf{h} = \{h_1, h_2, h_3, h_4, h_5, h_6\}$, $\mathbf{a} = \{a_0, a_1, a_2, a_3, a_4\}$ and T_0 determine the temperature function. The parameters \mathbf{x} and \mathbf{p}/\mathbf{T} determine the space of all measurable noise free data Ω through the forward model $\mathbf{A}(\mathbf{x}, \mathbf{p}, \mathbf{T})$ from which we randomly observe data set plus some random noise.

We draw a DAG for the measurement and modelling process, where the hyper-hyper-parameters $\theta_\gamma, \theta_\delta, \theta_{p_0}, \theta_b, \theta_{\mathbf{h}}, \theta_{T_0}, \theta_{\mathbf{a}}$ in the top row of Fig. 4.2 determine the hyper-prior distributions $\pi(\gamma, \delta, p_0, b, \mathbf{h}_T, \mathbf{T}_0, \mathbf{a})$ statistically (solid line). Then the hyper-parameters determine the parameters \mathbf{p}/\mathbf{T} deterministically. The temperature function $\mathbf{T} = (T_0, \mathbf{a}, \mathbf{h}_T)$, Eq. 4.3, is determined through \mathbf{a} the temperature gradients at heights \mathbf{h} , see Tab. 4.1, where h_0 is set to zero as we model temperature variability at the sea-level temperature trough the an additional input T_0 . Note that we define an exponential pressure function, Eq. 4.17, later in Sec. ?? so that $\mathbf{p}(p_0, b)$ is defined through the hyper-parameters p_0 (pressure at sea-level) and b (exponential gradient). Since we do not parametrise the ozone profile, we assume a certain smoothness defined through the smoothness hyper-parameter δ and a precision matrix $\mathbf{Q}(\delta)$ which statistically defines a distribution over \mathbf{x} (solid lines). The parameters $\mathbf{x}, \mathbf{p}, \mathbf{T}$ progress deterministically, see RTE in Eq. 3.1, into the forward model \mathbf{A}_{NL} and generate a space of all possible noise free data $\boldsymbol{\Omega}$. From that space of all measurable $\boldsymbol{\Omega}$ we pick one data set to which we add some noise $\boldsymbol{\eta} \sim \mathcal{N}(0, \boldsymbol{\Sigma})$, which is modelled through the hyper-parameter γ and the precision matrix $\boldsymbol{\Sigma} = \gamma^{-1} \mathbf{I}$ so that we obtain the noisy data vector \mathbf{y} . Since the noise is normally distributed, so is the likelihood function $\pi(\mathbf{y}|\mathbf{x}, \mathbf{p}, \mathbf{T})$. Then the joint posterior distribution

$$\pi(p_0, b, \mathbf{h}_T, \mathbf{a}, \delta, \gamma, \mathbf{x}|\mathbf{y}) \propto \pi(\mathbf{y}|\mathbf{x}, \mathbf{p}, \mathbf{T})\pi(p_0, b, \mathbf{h}_T, \mathbf{a}, \delta, \gamma) \quad (4.4)$$

over all 17 hyper-parameters and the parameter $\mathbf{x} \in \mathbb{R}^{45}$ is 62 dimensional. Ideally, we characterise the joint posterior, but this is computationally not feasible. Instead, we factorise the posterior into

$$\pi(p_0, b, \mathbf{h}_T, \mathbf{a}, \delta, \gamma, \mathbf{x}|\mathbf{y}) = \pi(\delta, \gamma, \mathbf{x}|p_0, b, \mathbf{h}_T, \mathbf{a}, \mathbf{y})\pi(p_0, b, \mathbf{h}_T, \mathbf{a}|\delta, \gamma, \mathbf{x}, \mathbf{y}), \quad (4.5)$$

where we either condition on ozone \mathbf{x} and the smoothness hyper-parameter δ as well as the noise hyper-parameter γ or on the fraction \mathbf{p}/\mathbf{T} , pressure over temperature, and its hyper-parameters. Again as in Sec. 3 for brevity we write $\pi(p_0, b, \mathbf{h}_T, \mathbf{a}|\gamma, \mathbf{y})$ and $\pi(\delta, \gamma, \mathbf{x}|\mathbf{y})$, which implies that we conditioned on \mathbf{x} or \mathbf{p} and \mathbf{T} . Next, we need to specify the prior distribution, which we summarise in Tab. 4.2, to formulate the posterior distributions.

4.2.1 Ozone conditioned on pressure and temperature

In this section, we choose prior distributions and describe the approach to evaluate the posterior distribution for ozone $\pi(\delta, \gamma, \mathbf{x}|\mathbf{y})$, including the noise hyper-parameter γ . Assuming Gaussian noise $\boldsymbol{\eta} \sim \mathcal{N}(0, \gamma^{-1} \mathbf{I})$, we define a linear-Gaussian Bayesian hierarchical model [12]

$$\mathbf{y}|\mathbf{x}, \gamma \sim \mathcal{N}(\mathbf{Ax}, \gamma^{-1} \mathbf{I}) \quad (4.6a)$$

$$\mathbf{x}|\delta \sim \mathcal{N}(0, \delta \mathbf{L}) \quad (4.6b)$$

$$\delta, \gamma \sim \pi(\delta, \gamma), \quad (4.6c)$$

model parameters	priors	TT bounds		τ_{int}	Context
		lower	upper		
γ	$\mathcal{T}(1, 10^{-10})$	$5 \cdot 10^{-8}$	$4.5 \cdot 10^{-7}$	9 ± 0.1	\mathbf{y}
δ	$\mathcal{T}(1, 10^{-10})$	-	-	1.5 ± 0.1	\mathbf{x}
λ	-	500	10^4	3.5 ± 0.3	\mathbf{x}
\mathbf{x}	$\mathcal{N}(0, \delta \mathbf{L})$	-	-		\mathbf{x}
h_1	$\mathcal{N}(11, 0.5)$	9.5	12.5	1820 ± 49	\mathbf{p}/\mathbf{T}
h_2	$\mathcal{N}(20.1, 1.6)$	17.7	22.3	454 ± 7	\mathbf{p}/\mathbf{T}
h_3	$\mathcal{N}(32.3, 2.5)$	25.2	39.8	394 ± 5	\mathbf{p}/\mathbf{T}
h_4	$\mathcal{N}(47.4, 5)$	45.9	48.9	324 ± 4	\mathbf{p}/\mathbf{T}
h_5	$\mathcal{N}(51.4, 5)$	49.9	52.9	319 ± 4	\mathbf{p}/\mathbf{T}
h_6	$\mathcal{N}(71.8, 3)$	62.5	80.8	347 ± 5	\mathbf{p}/\mathbf{T}
a_0	$\mathcal{N}(-6.5, 0.01)$	-6.53	-6.47	330 ± 4	\mathbf{p}/\mathbf{T}
a_1	$\mathcal{N}(0, 0.1)$	-0.3	0.3	508 ± 8	\mathbf{p}/\mathbf{T}
a_2	$\mathcal{N}(1, 0.01)$	0.97	1.03	341 ± 5	\mathbf{p}/\mathbf{T}
a_3	$\mathcal{N}(2.8, 0.1)$	2.5	3.1	316 ± 4	\mathbf{p}/\mathbf{T}
a_4	$\mathcal{N}(0, 0.1)$	-0.3	0.3	335 ± 4	\mathbf{p}/\mathbf{T}
a_5	$\mathcal{N}(-2.8, 0.1)$	-3.1	-2.5	335 ± 4	\mathbf{p}/\mathbf{T}
a_6	$\mathcal{N}(-2, 0.01)$	-2.03	-1.97	320 ± 4	\mathbf{p}/\mathbf{T}
T_0	$\mathcal{N}(288.15, 4.5)$	275	302	2446 ± 76	\mathbf{p}/\mathbf{T}
b	$\mathcal{N}(0.167, 5 \cdot 10^{-4})$	0.165	0.171	2813 ± 92	\mathbf{p}/\mathbf{T}
p_0	$\mathcal{N}(1243, 5)$	1229	1259	550 ± 9	\mathbf{p}/\mathbf{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 [36] from posterior samples based on the approximated forward map.

with a normally distributed likelihood $\pi(\mathbf{y}|\mathbf{x}, \gamma)$ including the forward model matrix \mathbf{A} and prior distributions $\pi(\mathbf{x}|\delta)$ and $\pi(\delta, \gamma)$, the noise covariance matrix $\gamma^{-1}\mathbf{I}$, the prior precision matrix $\delta \mathbf{L}$ and the prior mean set to $\mathbf{0}$. The chosen Bayesian model is very similar to a regularisation problem, since we like to show that regularisation is depreciated, and we are able to receive much more meaningful results compared to a regularisation approach.

Prior Modelling

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

First, we set the precision matrix of the prior distribution $\mathbf{x}|\delta$ to

$$\delta\mathbf{L} = \delta \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix} \quad (4.7)$$

which is the 1-dimensional Graph Laplacian as in [12, 37] with Dirichlet boundary condition. This matrix will also act as the regulariser later in the Regularisation section, see Sec. 2.5. For δ and γ we pick relatively uninformative gamma distributions so that $\gamma \sim \mathcal{T}(\boldsymbol{\theta}_\gamma)$ and $\delta \sim \mathcal{T}(\boldsymbol{\theta}_\delta)$, where $\boldsymbol{\theta}_\gamma = \boldsymbol{\theta}_\delta = (1, 10^{-10})$, see Fig. 4.15. These gamma distributions have another advantage when sampling from the marginal posterior distribution $\pi(\gamma, \delta|\mathbf{y})$, where $\pi(\gamma|\lambda, \mathbf{y}) \sim \mathcal{T}(\cdot)$ with the regularisation parameter $\lambda = \delta/\gamma$. We plot the corresponding prior ozone profiles according to $\mathbf{x} \sim \mathcal{N}(0, \delta\mathbf{L})$ in Fig. C.20 and like to note that we should not include negative ozone values but are currently not able to include e.g. a truncated multivariate normal prior distribution for \mathbf{x} .

posterior distribution into marginal and conditional posterior distribution

As noted in Sec. 2.1, we factorise the posterior

$$\pi(\mathbf{x}, \gamma, \delta|\mathbf{y}) \propto \pi(\mathbf{y}|\mathbf{x}, \gamma, \delta)\pi(\mathbf{x}, \gamma, \delta) \quad (4.8)$$

into

$$\pi(\mathbf{x}, \gamma, \delta|\mathbf{y}) = \pi(\mathbf{x}|\gamma, \delta, \mathbf{y})\pi(\gamma, \delta|\mathbf{y}) \quad (4.9)$$

the marginal posterior $\pi(\gamma, \delta|\mathbf{y})$ and conditional posterior $\pi(\mathbf{x}|\gamma, \delta, \mathbf{y})$. Fox and Norton call this method the marginal and then conditional method (MTC) [12], where we break the correlation structure between \mathbf{x} and γ, δ as illustrated in Fig. C.19 and Fig. A.1 by marginalising over \mathbf{x} and evaluating this marginal posterior first and *then* the conditional posterior.

For the linear-Gaussian Bayesian hierarchical model specified in Eq. 4.18, the marginal posterior distribution over the hyper-parameters is given by

$$\pi(\lambda, \gamma|\mathbf{y}) \propto \lambda^{n/2}\gamma^{m/2} \exp\left\{-\frac{1}{2}g(\lambda) - \frac{\gamma}{2}f(\lambda)\right\}\pi(\lambda, \gamma), \quad (4.10)$$

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

$$f(\lambda) = \mathbf{y}^T \mathbf{y} - (\mathbf{A}^T \mathbf{y})^T (\mathbf{A}^T \mathbf{A} + \lambda \mathbf{L})^{-1} (\mathbf{A}^T \mathbf{y}), \quad (4.11a)$$

$$\text{and } g(\lambda) = \log \det(\mathbf{A}^T \mathbf{A} + \lambda \mathbf{L}), \quad (4.11b)$$

see [12, Lemma 2]. When considering \mathbf{x} and \mathbf{y} as a joint multivariate normal distribution or a joint Gaussian Markov random field $(\mathbf{x}^T, \mathbf{y}^T)^T$, then \mathbf{x} conditioned on the hyperparameters γ, δ and the data \mathbf{y} is the normally distributed conditional posterior distribution

$$\mathbf{x}|\delta, \gamma, \mathbf{y} \sim \mathcal{N}\left(\underbrace{(\mathbf{A}^T \mathbf{A} + \delta/\gamma \mathbf{L})^{-1} \mathbf{A}^T \mathbf{y}}_{\mathbf{x}_\lambda}, \underbrace{(\gamma \mathbf{A}^T \mathbf{A} + \delta \mathbf{L})^{-1}}_{\gamma \mathbf{B}_\lambda}\right), \quad (4.12)$$

see [6, 12, 38] for more information. In this thesis, we compute the mean

$$\mu_{\mathbf{x}|\mathbf{y}} = \int \mathbf{x}_\lambda \pi(\lambda|\mathbf{y}) d\lambda \approx \sum \mathbf{x}_{\lambda_i} \pi(\lambda_i|\mathbf{y}), \quad (4.13)$$

and covariance

$$\Sigma_{\mathbf{x}|\mathbf{y}} = \int \gamma^{-1} \pi(\gamma|\mathbf{y}) d\gamma \int \mathbf{B}_\lambda^{-1} \pi(\lambda|\mathbf{y}) d\lambda \approx \sum \gamma_i^{-1} \pi(\gamma_i|\mathbf{y}) \sum \mathbf{B}_{\lambda_i}^{-1} \pi(\lambda_i|\mathbf{y}) \quad (4.14)$$

as weighted expectations, by quadrature [39, Sec. 2.1], with $\sum \pi(\lambda_i|\mathbf{y}) = \sum \pi(\gamma_i|\mathbf{y}) = 1$. If that is too costly, the randomise-then-optimise (RTO) [12, 40] may be a feasible alternative to sample from Eq. 4.12.

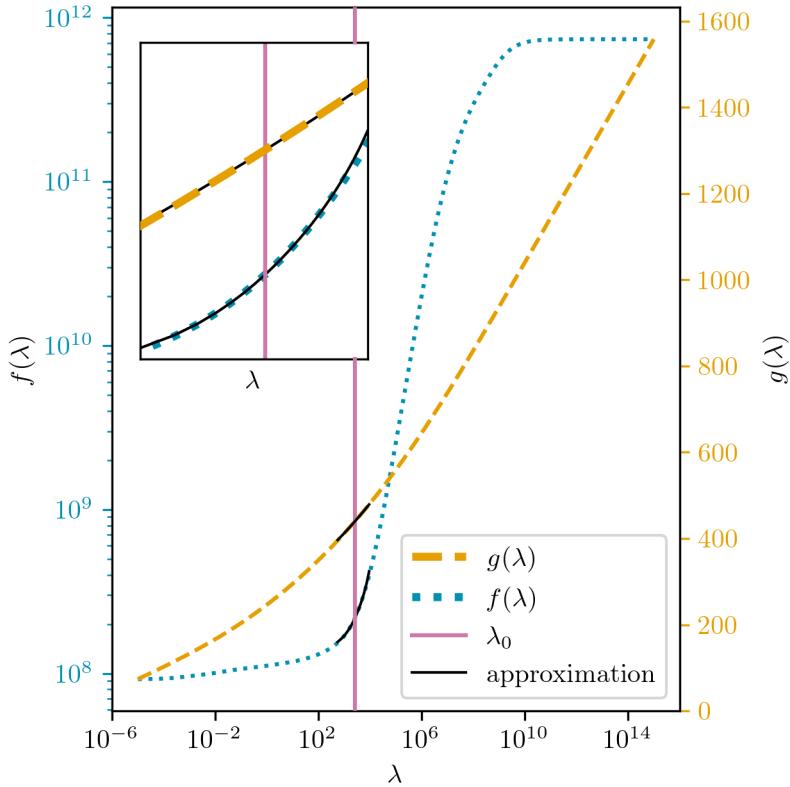


Figure 4.3: Plot of the functions $f(\lambda)$ and $g(\lambda)$ from the marginal posterior for a wide range of $\lambda = \delta/\gamma$. We plot the third order Taylor series in black around the mode of the marginal posterior (vertical line) for the sampling range of λ within the MWG algorithm.

Most of the computational effort lies in the evaluation of $f(\lambda)$ and $g(\lambda)$, see marginal posterior in Eq. 4.10. In Fig. 4.3 we see that $f(\lambda)$ and $g(\lambda)$ are well behaved within the

region of interest. Consequently we approximate $f(\lambda) \approx \tilde{f}(\lambda)$ with a 3rd order Taylor series around the mode λ_0 of $\pi(\lambda, \gamma | \mathbf{y})$. We also note that $\tilde{g}(\lambda) \approx g(\lambda)$ behaves linearly around λ_0 in the log-space. As a result of these observations, the approximations are implicitly given by

$$f^{(r)}(\lambda_0) = (-1)^{r+1} r! (\mathbf{A}^T \mathbf{y})^T (\mathbf{B}_0^{-1} \mathbf{L})^r \mathbf{B}_0^{-1} \mathbf{A}_L^T \mathbf{y} \quad (4.15)$$

$$\text{and } \log \tilde{g}(\lambda) = (\log \lambda - \log \lambda_0) \frac{\log g(\lambda_{\max}) - \log g(\lambda_0)}{\log \lambda_{\max} - \log \lambda_0} + \log g(\lambda_0) \quad (4.16)$$

with $\mathbf{B}_0 = \mathbf{A}^T \mathbf{A} + \lambda_0 \mathbf{L}$. We plot the approximations in Fig. 4.3 and elaborate on approximation errors in sec 4.7

4.2.2 Pressure over temperature conditioned on noise and ozone

First, we observe that we can describe the pressure values in between $h_{L,0} = 7\text{km}$ and $h_{L,n} = 83.3\text{km}$ with an exponential function

$$p(h) = \exp \{-b h\} p_0 \quad , h_{L,n} \leq h \leq h_{L,0} \quad (4.17)$$

so that we parametrize the pressure \mathbf{p} with the hyperparameters p_0, b . Then, within the hierarchical Bayesian framework

$$\mathbf{y} | \mathbf{p}, \mathbf{T}, \gamma \sim \mathcal{N}(\mathbf{A} \mathbf{p} / \mathbf{T}, \gamma^{-1} \mathbf{I}) \quad (4.18a)$$

$$\mathbf{a} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{a}}, \boldsymbol{\Sigma}_{\mathbf{a}}) \quad (4.18b)$$

$$\mathbf{h}_{\mathbf{T}} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{T}}, \boldsymbol{\Sigma}_{\mathbf{h}_{\mathbf{T}}}) \quad (4.18c)$$

$$T_0 \sim \mathcal{N}(\mu_{T_0}, \sigma_{T_0}) \quad (4.18d)$$

$$p_0 \sim \mathcal{N}(\mu_{p_0}, \sigma_{p_0}) \quad (4.18e)$$

$$b \sim \mathcal{N}(\mu_b, \sigma_b) \quad (4.18f)$$

we define a normally distributed likelihood (due to Gaussian noise) and priors, where the hyper-prior means and variances relate to the DAG in Fig. 4.2 so that $\boldsymbol{\theta}_{\mathbf{a}} = (\boldsymbol{\mu}_{\mathbf{a}}, \boldsymbol{\Sigma}_{\mathbf{a}})$, $\boldsymbol{\theta}_{\mathbf{h}_{\mathbf{T}}} = (\boldsymbol{\mu}_{\mathbf{T}}, \boldsymbol{\Sigma}_{\mathbf{h}_{\mathbf{T}}})$, $\boldsymbol{\theta}_{T_0} = (\mu_{T_0}, \sigma_{T_0})$, $\boldsymbol{\theta}_{p_0} = (\mu_{p_0}, \sigma_{p_0})$, and $\boldsymbol{\theta}_b = (\mu_b, \sigma_b)$. Note that we do not include h_0 , from Tab. 4.1, in $\mathbf{h}_{\mathbf{T}}$ since we model temperature variability at sea level through T_0 .

Prior modelling

We summarise the mean and variance in Tab. 4.2 and plot samples from the prior distribution against the ground truth for the pressure \mathbf{p} and temperature \mathbf{T} separately in Fig. 4.6 and 4.5 and jointly as \mathbf{p}/\mathbf{T} in Fig. 4.5. We plot the prior samples against the ground truth for $1/\mathbf{K}$ in Fig. C.22.

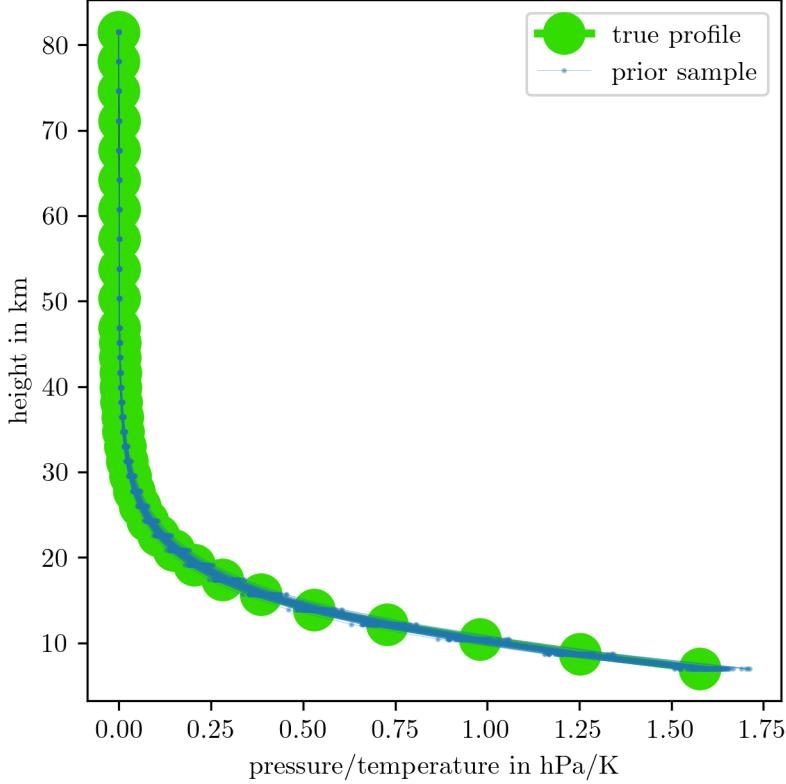


Figure 4.4: We draw samples from the hyper-prior distribution of $h_0, b, p_0, h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4$ and T_0 as defined in table 4.2 and then calculate \mathbf{p}/\mathbf{T} according to the functions in Eq. 4.17 and 4.3.

We carefully choose the hyper-prior distributions \mathbf{h}_T so that the individual distributions for heights $h_1, h_2, h_3, h_4, h_5, h_6$ do not overlap, see Fig. C.21. Additionally, we define the sampling space and the grid for the TT approximation accordingly. We remark that we can already observe in Fig. 4.4 that \mathbf{p}/\mathbf{T} inherits the structure of the pressure function.

posterior distribution

Then we can define the posterior distribution

$$\pi(p_0, b, \mathbf{h}_T, \mathbf{c}_T, \mathbf{a}_T | \mathbf{y}, \gamma, \mathbf{x}) \propto \exp\left\{-\frac{\gamma}{2} \left\|\mathbf{y} - \mathbf{A} \frac{\mathbf{p}}{\mathbf{T}}\right\|^2\right\} \pi(p_0, b, \mathbf{h}_T, \mathbf{c}_T, \mathbf{a}_T), \quad (4.19)$$

which is, conditioned on the noise hyper-parameter γ , the ozone profile \mathbf{x} and the smoothness hyper-parameter δ , a 16 dimensional distribution.

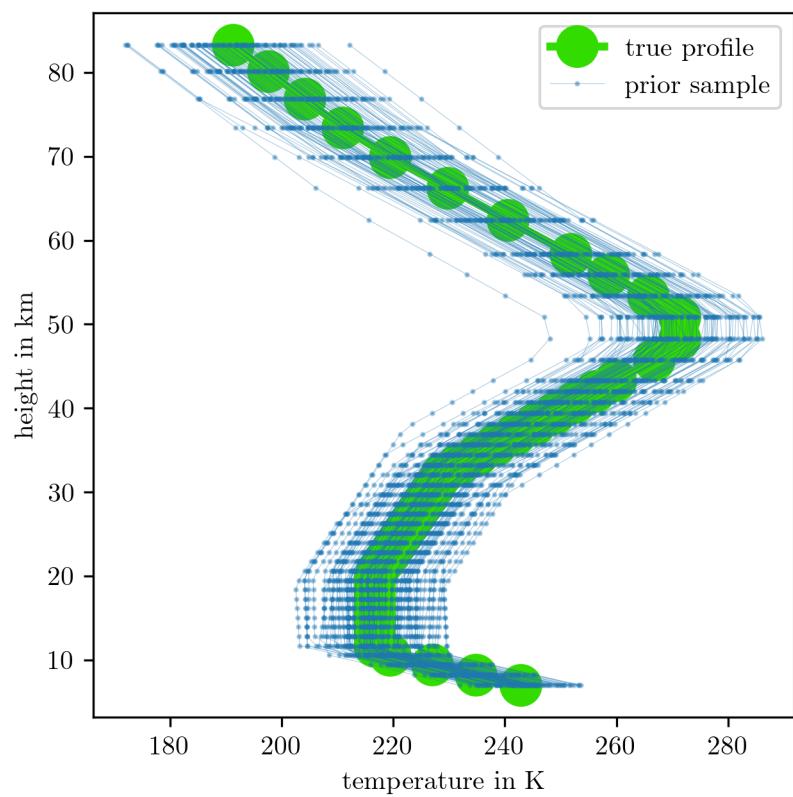


Figure 4.5: We draw samples from the hyper-prior distribution of $h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4$ and T_0 as defined in table 4.2 and then calculate \mathbf{T} according to the function in Eq. 4.3.

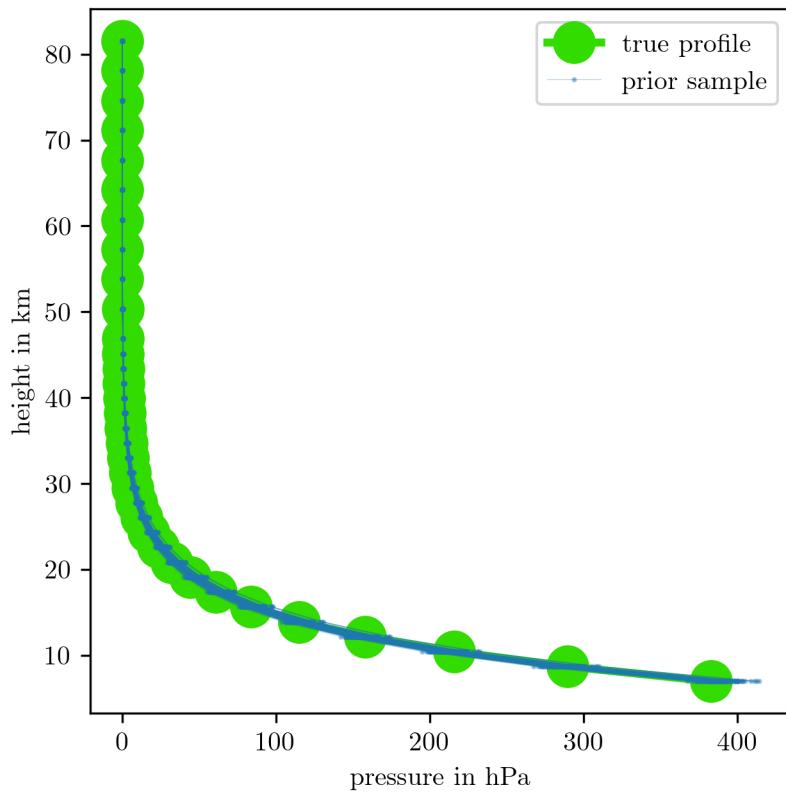


Figure 4.6: We draw samples from the hyper-prior distribution of h_0, b and p_0 as defined in table 4.2 and then calculate \mathbf{p} according to the function in Eq. 4.17.

4.3 Approximate non-linear forward model with affine Map

With the posterior distributions formulated, we can now approximate the non-linear forward model with an affine map \mathbf{M} ; we see Fig. 4.7 for the summarised strategy. We focus on the posterior distribution of ozone profiles by conditioning on pressure and temperature, as this is a quick process when using the MTC method. We approximate and sample from the marginal posterior $\pi(\gamma, \delta | \mathbf{y})$ and then characterise the full conditional posterior distribution $\pi(\mathbf{x} | \mathbf{y})$ based on the linear forward model \mathbf{A}_L , neglecting absorption, see Eq. 3.1. Given samples $\mathbf{x} \sim \pi(\mathbf{x} | \mathbf{y})$ from the full conditional posterior distribution, we can generate two affine subspaces based on the linear and non-linear model and find the mapping between those.

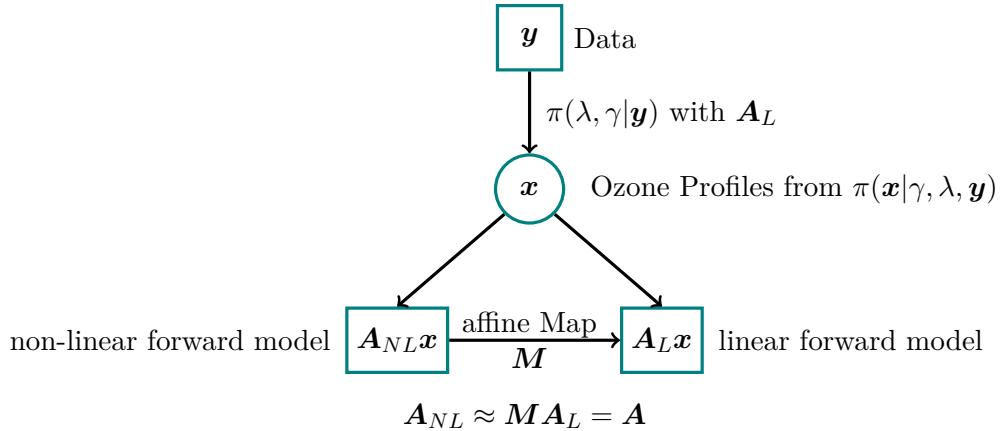


Figure 4.7: The strategy to find the affine map consist of evaluating the marginal posterior for ozone using the linear forward model. Then we draw ozone samples from the conditional posterior and calculate noise free data based on the linear and non-linear forward model. Next we find a mapping in between those two space so that we can approximate the non-linear forward model using an affine map and the linear forward model.

4.3.1 Sample from marginal posterior distribution for ozone

We set $\mathbf{A} = \mathbf{A}_L$ and characterise the marginal posterior $\pi(\lambda, \gamma | \mathbf{y})$ as in Eq. 4.10 by employing a Metropolis within Gibbs (MWG) algorithm, see sec. ???. More specifically, we implement a Metropolis random walk on the full conditional

$$\pi(\lambda | \gamma, \mathbf{y}) \propto \lambda^{n/2 + \alpha_\delta - 1} \exp \left\{ -\frac{1}{2} g(\lambda) - \frac{\gamma}{2} f(\lambda) - \beta_\delta \gamma \lambda \right\} \quad (4.20)$$

and do a Gibbs step on

$$\gamma | \lambda, \mathbf{y} \sim \Gamma \left(\frac{m}{2} + \alpha_\delta + \alpha_\gamma, \frac{1}{2} f(\lambda) + \beta_\gamma + \beta_\delta \lambda \right) \quad (4.21)$$

to generate marginal posterior samples $(\lambda, \gamma)^{(1)}, \dots, (\lambda, \gamma)^{(N)} \sim \pi(\lambda, \gamma | \mathbf{y})$. 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 $d\delta/d\lambda = \gamma$.

Hence we run a Metropolis random walk on $\pi(\lambda|\gamma, \mathbf{y})$, the proposal distribution $q(\lambda'|\lambda^{(k)}) \sim \mathcal{N}(\lambda^{(k)}, 0.8\lambda_0)$ conditioned on the previous sample $\lambda^{(k)}$, with $k = 1, \dots, N$ is symmetric. Then, we accept or reject a new λ' sample by comparing the acceptance ratio

$$\log \left\{ \frac{\pi(\lambda'|\gamma^{(k)}, \mathbf{y})}{\pi(\lambda^{(k)}|\gamma^{(k)}, \mathbf{y})} \right\} = \log\{\pi(\lambda'|\gamma^{(k)}, \mathbf{y})\} - \log\{\pi(\lambda^{(k)}|\gamma^{(k)}, \mathbf{y})\} \quad (4.22)$$

$$= \frac{n}{2}(\log\{\lambda'\} - \log\{\lambda^{(t-1)}\}) + \frac{1}{2}\Delta g + \frac{\gamma^{(t-1)}}{2}\Delta f + \beta_\delta \gamma^{(t-1)}\Delta\lambda, \quad (4.23)$$

where $\Delta\lambda = \lambda' - \lambda^{(k)}$ to a random uniform number in between 0 and 1. Note that since we calculate the acceptance ratio in the log space $\Delta f \approx \tilde{f}(\lambda') - \tilde{f}(\lambda^{(k)}) = \sum f^{(r)}(\lambda_0)\Delta\lambda' - \Delta\lambda^{(k)}$ is a 3rd order taylor approximaton, see Fig. 4.3, where $\Delta\lambda' = \lambda' - \lambda_0$ and $\Delta\lambda^{(k)} = \lambda^{(k)} - \lambda_0$. Similarly we approximate $\Delta g \approx \exp \log \tilde{g}(\lambda') - \exp \log \tilde{g}(\lambda^{(k)})$ as in Eq. 4.11. Lastly, a Gibbs step provides a new $\gamma^{(k+1)} \sim \gamma|\lambda^{(k+1)}, \mathbf{y}$, see Equation (4.21). See Algorithmic Box ?? for a summary of the general version.

We initialise the MWG at the mode $(\lambda^{(0)}, \gamma^{(0)}) = (\lambda_0, \gamma_0)$ and take for $N = 10000$ plus $N_{\text{burn-in}} = 100$ steps in less than 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 []. The samples are plotted in Fig. 4.10 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 [], which gives us $\tau_{\text{int},\gamma} =$ and $\tau_{\text{int},\delta} =$.

4.3.2 Tensor-train approximation of the marginal posterior distribution for ozone

Alternatively, we can approximate the marginal posterior with a tensor-train (TT) of the square root of the marginal posterior on a predefined grid. We define a grid similar to the sampling region of the MWG sampler with 25 grid points in each dimension and use the `tt.cross.rectcross.rect_cross.cross` function from the `ttypy` Python package, based on the rect cross algorithm in []. We set the number of ranks to a constant value $r = 4$ and optimise over those ranks with one sweep to calculate the cores in less than 0.1s. To avoid underflow, we have to add a 'normalisation' constant $c = 460$ so that we approximate $\pi(\lambda|\gamma, \mathbf{y}) = \exp\{\log \pi(\lambda|\gamma, \mathbf{y}) + c\}$. Then we calculate the marginals $\pi(\lambda|\mathbf{y})$ and $\pi(\gamma|\mathbf{y})$ as described in section ??, assuming an absolute error of 1 the constant $\xi = 1/\lambda(\mathcal{X})$, with a diagonal mass matrix $M_k = \text{diag}(\lambda(X_k))$, where $\lambda(X_k)$ is the length of the grid of the k th dimension also known as the Lebesgue measure of a closed interval. We plot the TT approximation as a colour map on top of the obtained samples in the scatter plot in Fig. 4.10.

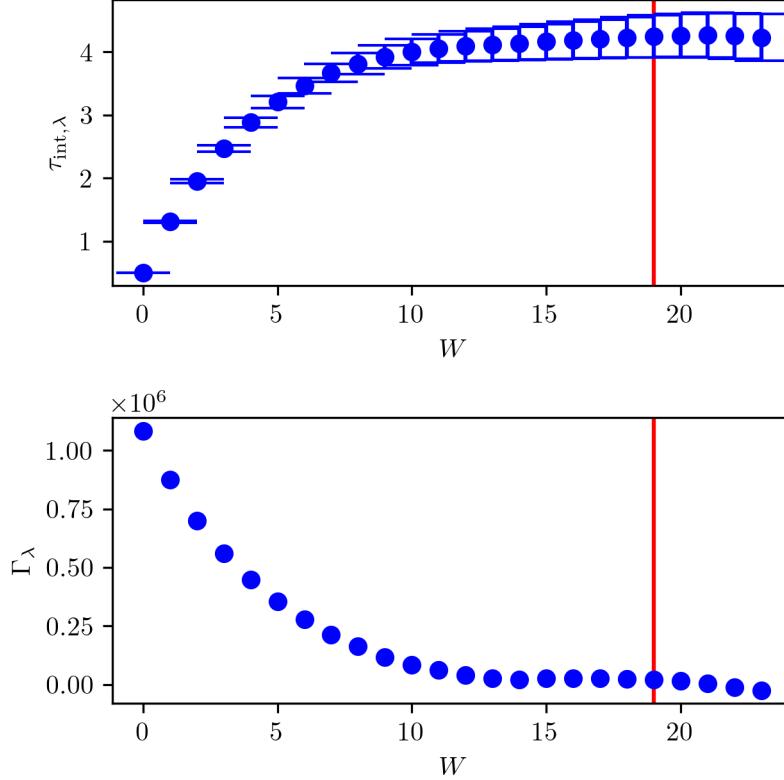


Figure 4.8

4.3.3 Calculate mean and variance of the full conditional posterior for ozone

Based on the marginal posterior distribution $\pi(\gamma, \delta | \mathbf{y})$ we calculate the weighted mean and covariance of the conditional posterior $\pi(\mathbf{x} | \gamma, \delta, \mathbf{y})$ by quadrature as in Eq. 4.13 and Eq. 4.14.

By binning the output samples from the MWG, see Fig. 4.10, into a normalised histogram with 25 bins, we obtain function values for the marginal posterior. With the height of the histogram bars as quadrature weights, e.g. $\pi(\lambda_i | \mathbf{y})$, where λ_i is the centre of each bin we calculate the full conditional mean $\mu_{\mathbf{x}|\mathbf{y}}$ and covariance matrix $\Sigma_{\mathbf{x}|\mathbf{y}}$ as weighted expectations.

Alternatively we use the marginal distributions $\pi(\delta | \mathbf{y})$ and $\pi(\gamma | \mathbf{y})$ from the TT approximation of $\sqrt{\pi(\delta, \gamma | \mathbf{y})}$ to calculate weighted expectations of $\mu_{\mathbf{x}|\mathbf{y}}$ and $\Sigma_{\mathbf{x}|\mathbf{y}}$.

In practice, we have to invert \mathbf{B}_λ and calculate \mathbf{x}_λ , see Eq. 4.12 25 times (TT grid size and number of bins). A feasible method is the Cholesky forward and backwards substitution [], which takes roughly 1s to compute the mean and variance. Note that we reject unphysical samples from the conditional posterior with negative ozone values and plot those in Fig. 4.12, including variance and mean.

Computation time is less than 0.2s.

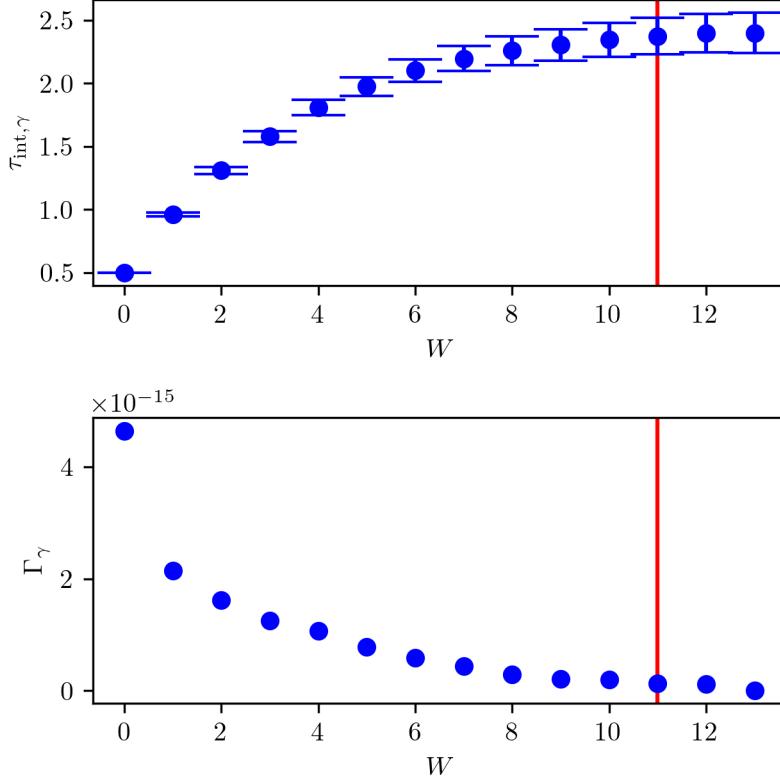


Figure 4.9

4.3.4 Asses approximated forward model

Given m samples $\mathbf{x}^j \sim \pi_{\mathbf{x}}|\mathbf{y}$ for $j = 1, \dots, m$ from the full conditional, as plotted in Fig. 4.12, we are able approximate the non-linear forward model

$$\mathbf{A}_{NL} \approx \mathbf{M}\mathbf{A}_L = \mathbf{A}, \quad (4.24)$$

with the affine map \mathbf{M} and the linear forward model \mathbf{A}_L . In doing so, we can generate two affine subspaces $\mathbf{W} = \{\mathbf{A}_L \mathbf{x}^1, \dots, \mathbf{A}_L \mathbf{x}^m\}$ and $\mathbf{V} = \{\mathbf{A}_{NL} \mathbf{x}^1, \dots, \mathbf{A}_{NL} \mathbf{x}^m\}$. We use the Python function `numpy.linalg.solve` to solve $\mathbf{M}\mathbf{W} = \mathbf{V}$ for each row of \mathbf{M} , see Sec. 2.4 for more details.

We asses the affine map by calculating the relative error $\|\mathbf{M}\mathbf{W} - \mathbf{V}\|/\|\mathbf{M}\mathbf{W}\|$ between the mapped noise free data and the noise free data of non-linear forward model for all of the m ozone samples, which approximnately 0.1%. We display the approximation for one \mathbf{x} sample in Fig. 4.13. Consequently, from here onwards, we use the approximated forward map $\mathbf{A} = \mathbf{M}\mathbf{A}_L$.

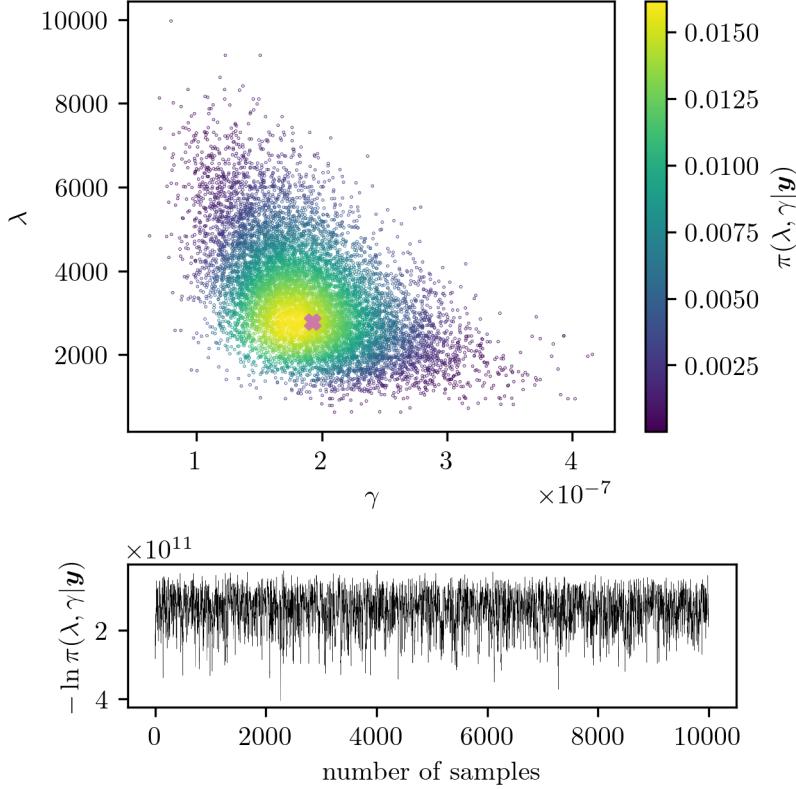


Figure 4.10: We scatter plot the samples of $\lambda = \delta/\gamma$ and γ from the marginal posterior $\pi(\lambda, \gamma|y)$ and colour code the samples using the TT approximation of $\pi(\lambda, \gamma|y)$. The mode of (λ_0, γ_0) of $\pi(\lambda, \gamma|y)$ provided by `scipy.optimize.fmin` is marked with the cross. To show ergodicity we plot the trace of the samples of the Metropolis-within-Gibbs sampler below.

4.4 Solution by regularisation

Since we like to compare the MTC method to regularisation methods, we calculate a solution by Tikhonov regularisation, as this is most similar to our chosen linear-Gaussian Bayesian framework. The Tikhonov regularised solution is defined as [41]

$$\mathbf{x}_\lambda = \arg \min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2 + \lambda \mathbf{x}^T \mathbf{L} \mathbf{x}, \quad (4.25)$$

with the regularisation parameter $\lambda = \delta/\gamma$. The regularised solution is typically calculated by solving the normal equations, see Sec. 2.5,

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

To find the best regularised solution, we use the L-curve method [27]. Within this method we compute \mathbf{x}_λ , for 200 different λ values in between 1 to 10^7 and plot the solution semi norm $\sqrt{\mathbf{x}_\lambda^T \mathbf{L} \mathbf{x}_\lambda}$ against the data misfit norm $\|\mathbf{A}\mathbf{x}_\lambda - \mathbf{y}\|$, see Figure 4.14. The best regularised solution corresponding to the corner of the L-curve is located at the point of maximum curvature, see triangle in Fig. 4.14, which we find with the kneedle algorithm [42] using the python function `kneed.KneeLocator` in less 0.1s.

424.5. Characterise the posterior distribution of ozone with approximated forward model

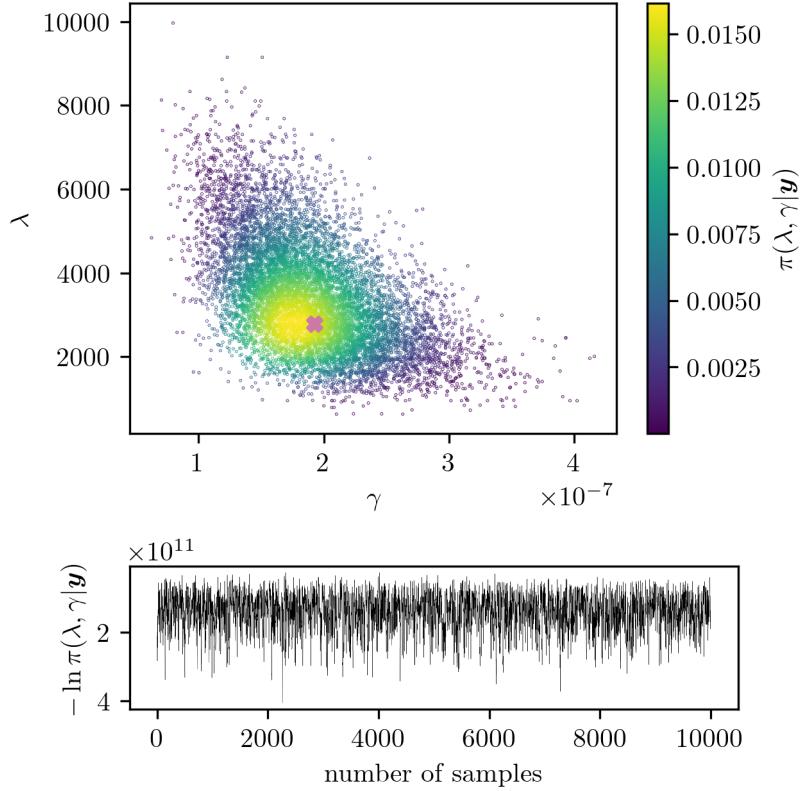


Figure 4.11

4.5 Characterise the posterior distribution of ozone with approximated forward model

With the affine approximation

$$\mathbf{A} = \mathbf{M}\mathbf{A}_L \quad (4.27)$$

of the non-linear forward map, we use the same setup as in Sec. ?? and 4.3.3 to evaluate the marginal posterior and the conditional posterior.

4.5.1 Hyper-parameters samples from and Tensor-train approximation of the marginal posterior distribution

The marginal posterior is defined as in Eq. ?? but with $\mathbf{A} = \mathbf{M}\mathbf{A}_L$. We run the MWG algorithm for $N = 20000$ plus $N_{\text{burn-in}} = 100$ and plot the samples in Fig. 4.15 as well as the marginal approximations provided by the TT decomposition, where we use the same setup as in Sec. 4.3.2.

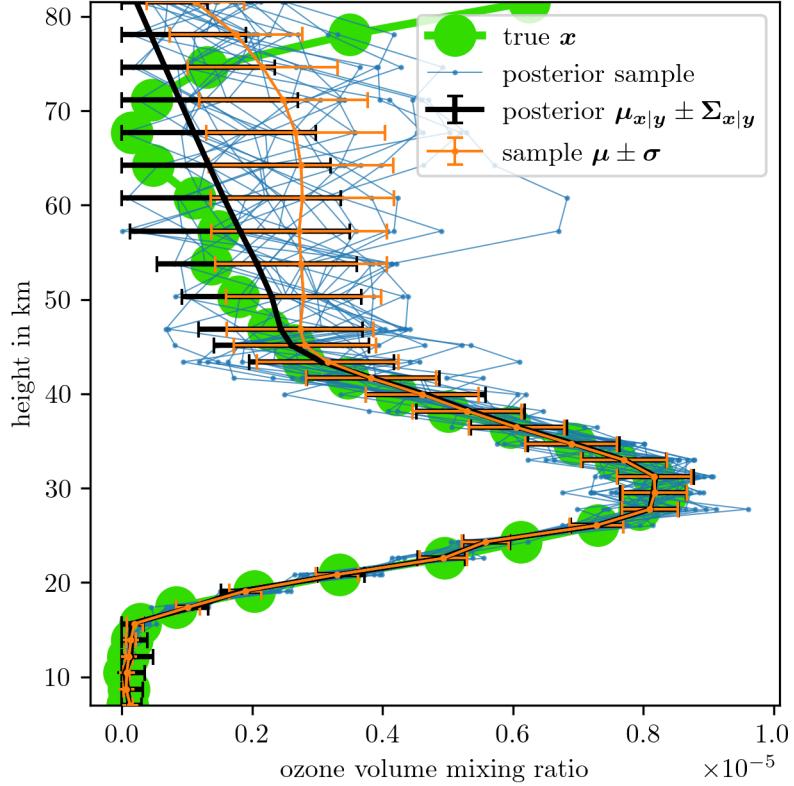


Figure 4.12: We draw samples from the conditional posterior distribution $\pi(\mathbf{x}|\lambda, \gamma, \mathbf{y})$ after characterising the marginal posterior $\pi(\lambda, \gamma|\mathbf{y})$ through sampling or TT approximation using the linear forward map \mathbf{A}_L . Note that we reject samples with unphysical negative values and effectively treat the conditional posterior as a truncated multivariate normal distribution. We will use those samples to find the affine map \mathbf{M} , see section 4.3

4.5.2 Full conditional posterior variance and mean

Next, we characterise the conditional posterior $\pi(\mathbf{x}|\gamma, \delta, \mathbf{y})$ as in Eq. 4.12. Again, we calculate the full conditional mean 4.13 and full conditional covariance matrix 4.14 as weighted expectation over a 25-point grid provided by either the marginal TT approximations or the histogram of samples. We plot the conditional mean and variance in Fig. 4.16 and the regularised solution and one sample from the posterior.

44.5. Characterise the posterior distribution of ozone with approximated forward model

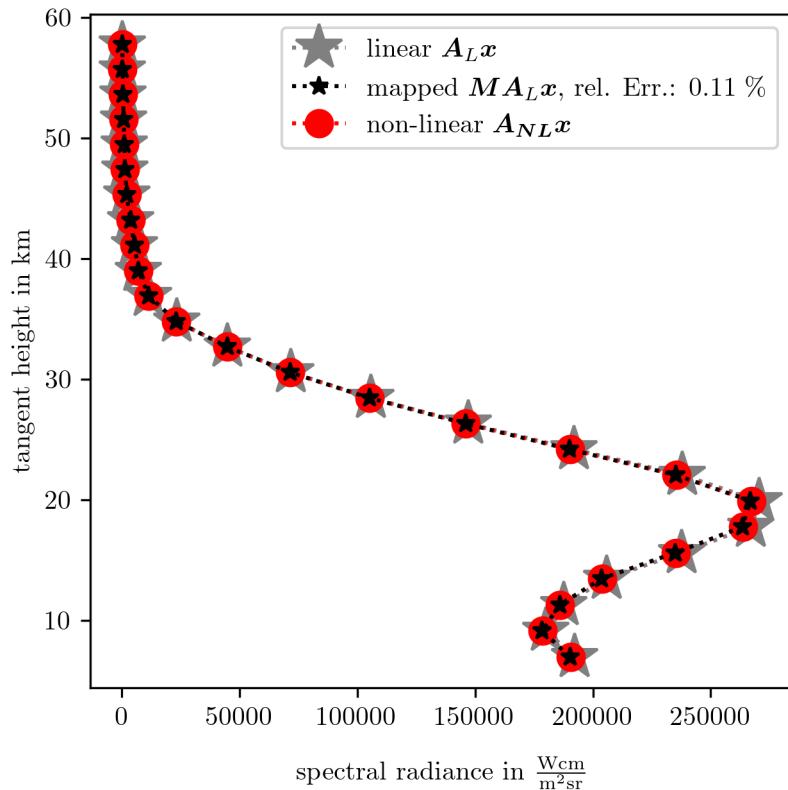


Figure 4.13: We asses how good we can map the linear forward model onto the non-linear forward model using the previous calculated affine map. The gray stars represent noise free linear data, where as the red circles present noise free non-linear data. Then we map the linear noise free data onto the non-linear noise free data and give the relative error in between the mapped noise free data and the non-linear data.

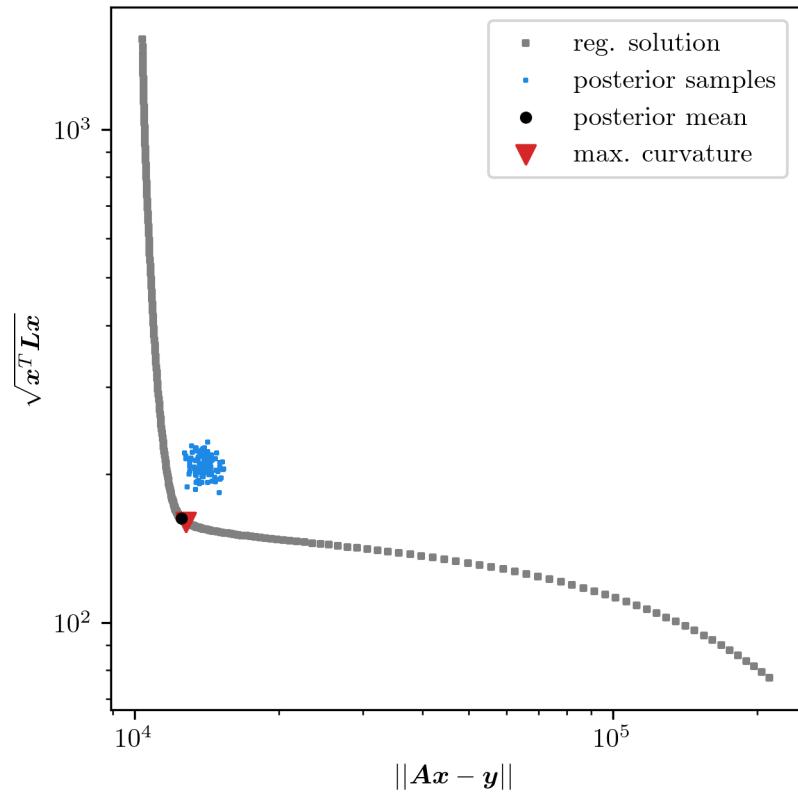


Figure 4.14: We calculate regularised solution as in Eq. ?? and plot the regularised semi norm $\sqrt{\mathbf{x}^T \mathbf{L} \mathbf{x}}$ against the data misfit norm $\|\mathbf{A}\mathbf{x} - \mathbf{y}\|$ to find the regularised solution at the point of maximum curvature of the so-called L-Curve. 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**

464.5. Characterise the posterior distribution of ozone with approximated forward model

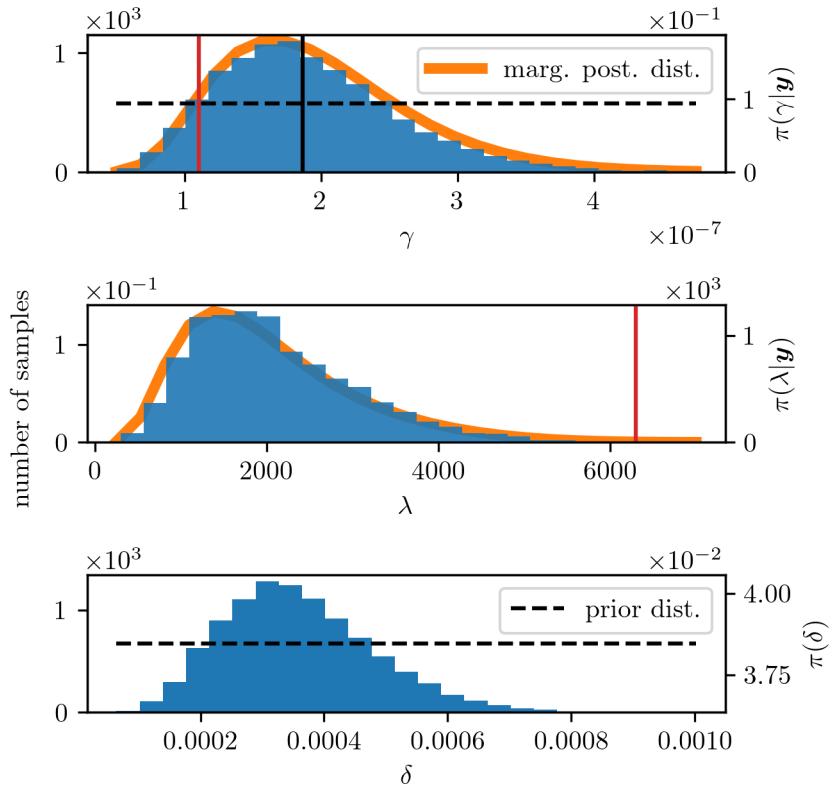


Figure 4.15: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line. Note that we sample λ and γ using the Metropolis-within-Gibbs sampler and can calculate δ for every sample of the marginal posterior, we can not do this for the TT approximation. The regularised parameter corresponding to the regularised solution is marked thought the red vertical line at $\lambda_{\text{reg}} =$.

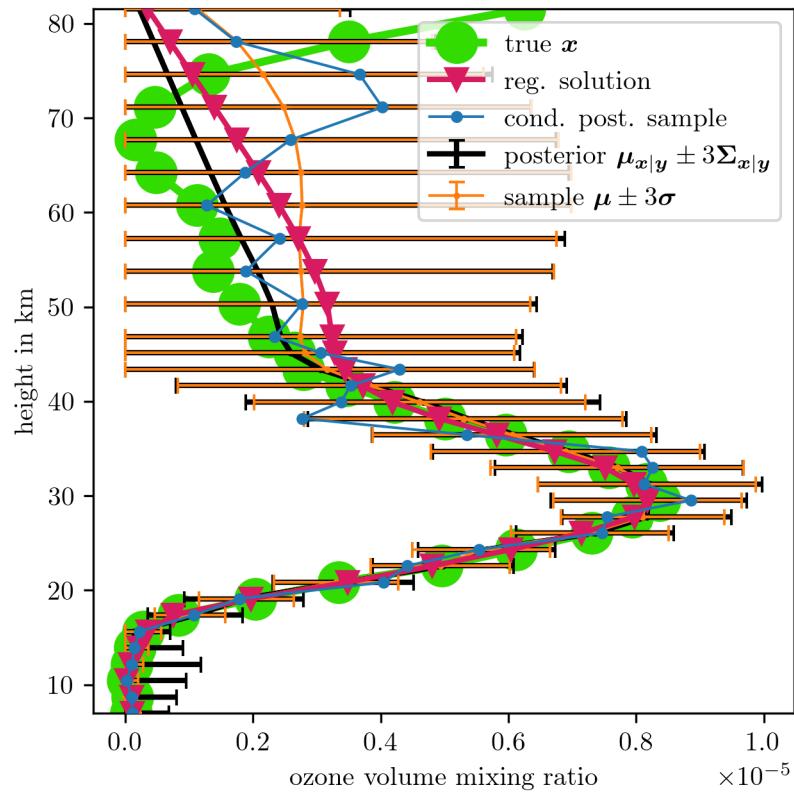


Figure 4.16: We plot the conditional posterior mean and variance in black and the regularised solution on top of the ground truth ozone profile in green. We use the updated forward map MA_L

484.5. Characterise the posterior distribution of ozone with approximated forward model

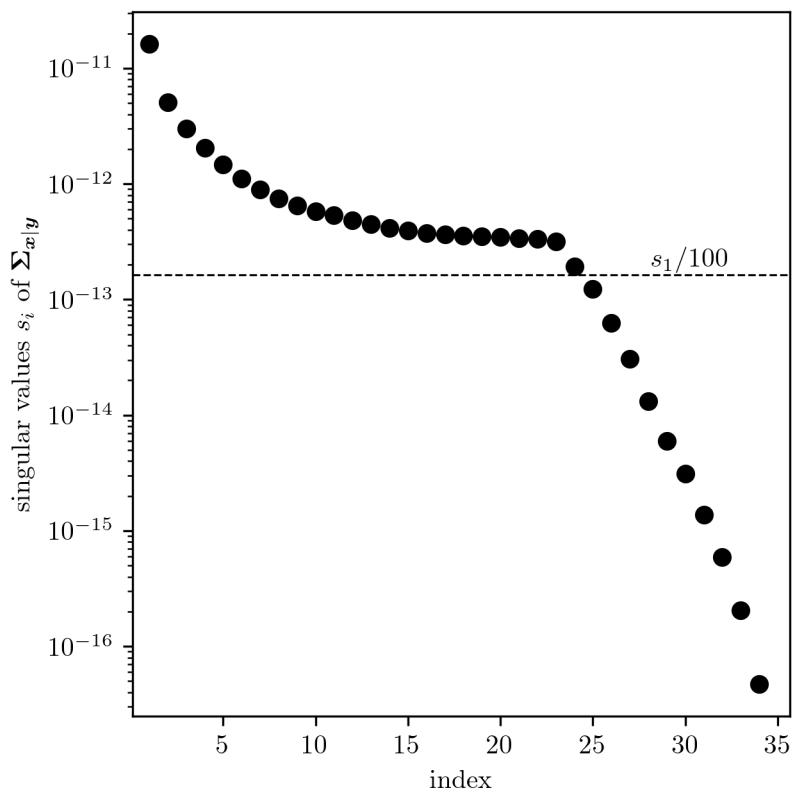


Figure 4.17

4.6 Posterior distribution for pressure and temperature with approximated forward model

The aim now is to characterise the posterior

$$\pi(p_0, b, T_0, \mathbf{h}_T, \mathbf{a}_T | \mathbf{y}, \gamma, \mathbf{x}) \propto \exp\left\{-\frac{\gamma}{2} \left\|\mathbf{y} - \mathbf{A} \frac{\mathbf{p}}{T}\right\|^2 + \ln \pi(p_0, b, T_0, \mathbf{h}_T, \mathbf{a}_T)\right\}, \quad (4.28)$$

conditioned on the ozone sample in Fig. 4.16 and a γ sample from the marginal posterior, and using the approximated forward model $\mathbf{A} = \mathbf{M}\mathbf{A}_L$. We will approximate this posterior with a TT and validate this approximation with samples from the posterior using the `t-walk` [14] implementation in Python [43]. “Conditioning on estimates gives poor predictive densities”. [3]

Again, we define a grid with 25 grid points in each dimension, which also acts as the sampling space. Since we approximate a 16-dimensional function, we have to carefully choose a grid, as we do not want to approximate regions with low probability, and we like to keep the number of grid points low, as this increases computation time. We find the grid, see Tab. 4.2, iteratively by running the t-walk and then computing marginal distributions. Note that we bound the sampling space of the t-walk by the TT-grid. We run the `tt.cross.rectcross.rect_cross.cross` function from the `ttipy` python package [] with constant rank $r = 16$, equal to the dimension of the posterior. Next, we introduce a constant c in the posterior, which acts as a normalisation constant and is needed when approximating the square root of the posterior for pressure and temperature with a tensor-train (TT) to avoid underflow. Then the posterior becomes:

$$\pi(p_0, b, T_0, \mathbf{h}_T, \mathbf{a}_T | \mathbf{y}, \gamma, \mathbf{x}) \propto \exp\left\{-\frac{\gamma}{2} \left\|\mathbf{y} - \mathbf{A} \frac{\mathbf{p}}{T}\right\|^2 + \ln \pi(p_0, b, T_0, \mathbf{h}_T, \mathbf{a}_T) + c\right\}. \quad (4.29)$$

To find the constant c , we evaluate the logarithm of the posterior on 5000 random points and calculate the maximum $c_{\max} < 0$ of those 5000 points. Then we set the constant to a value which pushes the posterior close to the upper numerical limit of our machine, which is approximately e^{700} . Since we approximate the square root, we conservatively set the constant to $c = -c_{\text{diff}} + 325$. It takes roughly 3 to 4min for 15 sweeps by the `cross` to find the optimal tensors. Then we can compute the marginal as in Sec. 2.3, where we set $\xi = 1/\lambda(\mathcal{X})$.

For comparison, we run the t-walk on the posterior as defined in Eq. 4.28 for 5×10^6 steps plus a burn-in period of 10000, which takes around 7 mins on the same laptop. We plot the resulting histograms in Fig. 4.18 to 4.22; additionally, we plot the trace of the samples in Fig. C.23. The integrated autocorrelation times (IACT) for the hyper-parameters range from 0 to 1000 and are summarised in Tab. 4.2.

406. Posterior distribution for pressure and temperature with approximated forward model

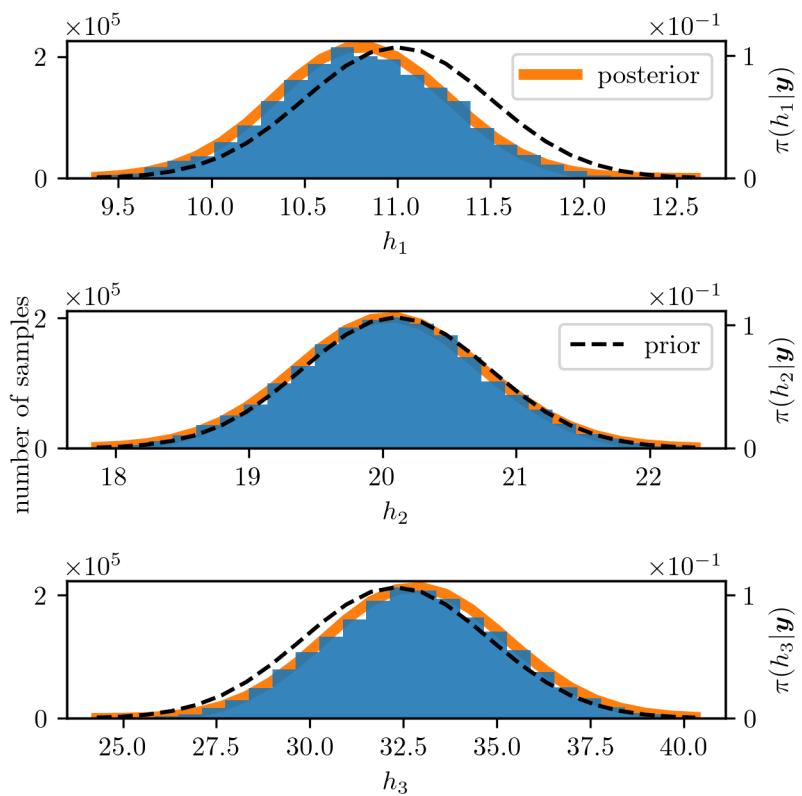


Figure 4.18: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

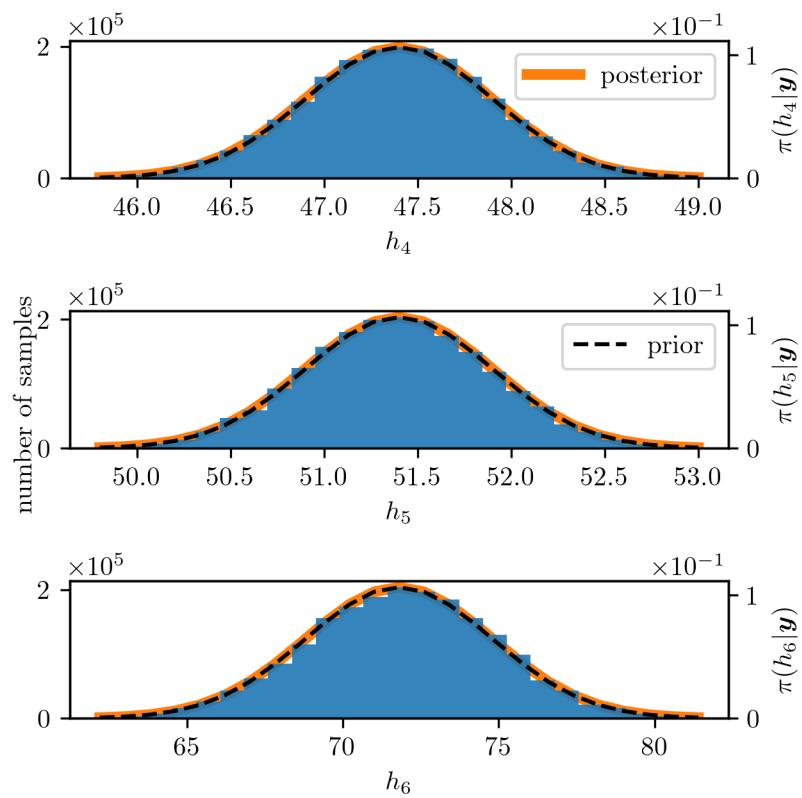


Figure 4.19: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

426. Posterior distribution for pressure and temperature with approximated forward model

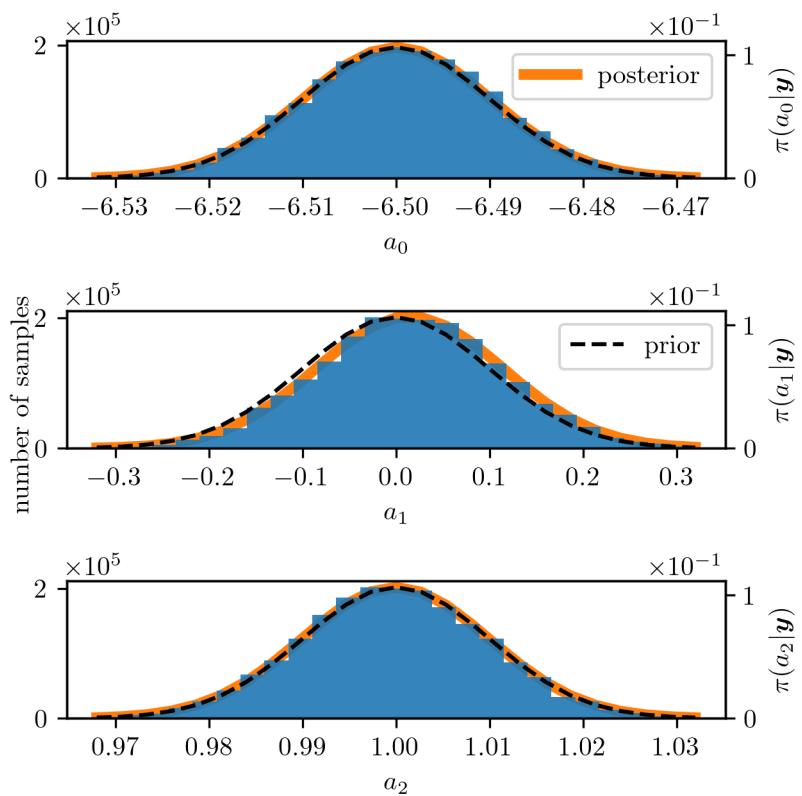


Figure 4.20: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

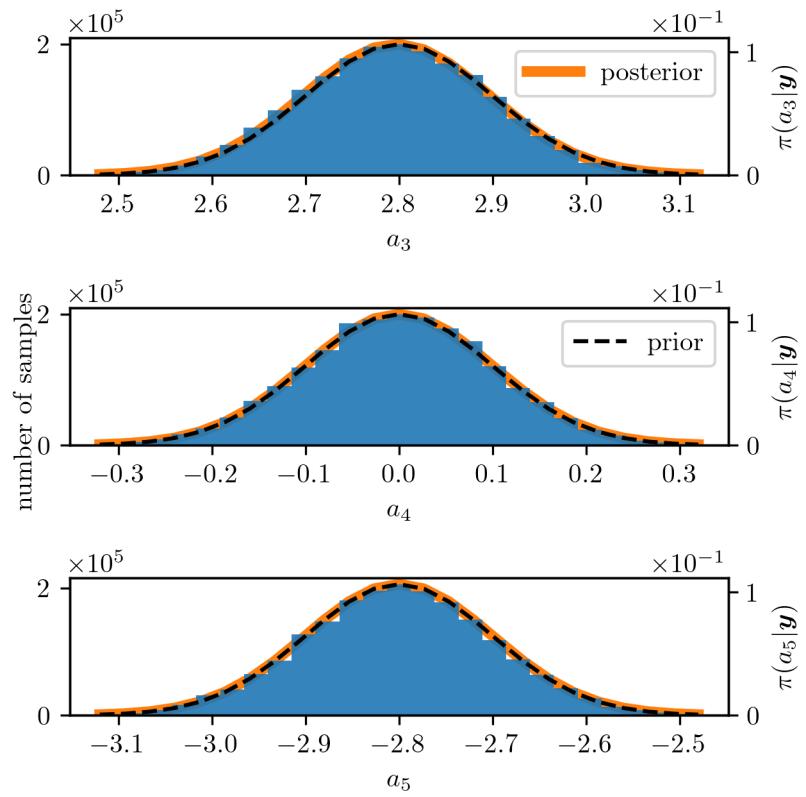


Figure 4.21: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

4.6. Posterior distribution for pressure and temperature with approximated forward model

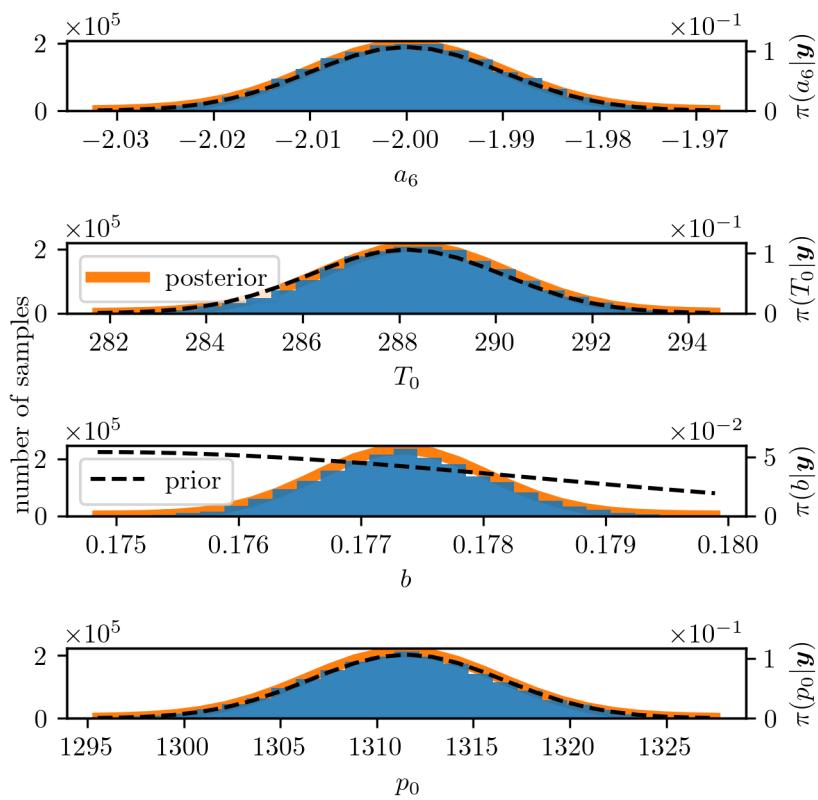


Figure 4.22: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

To obtain temperature and pressure profiles, we can either take samples from the output of the t-walk or generate random values between 0 and 1 and compare them to the cumulative distribution functions. We plot the posterior temperature and pressure profiles in Fig. 4.23 and Fig. 4.24.

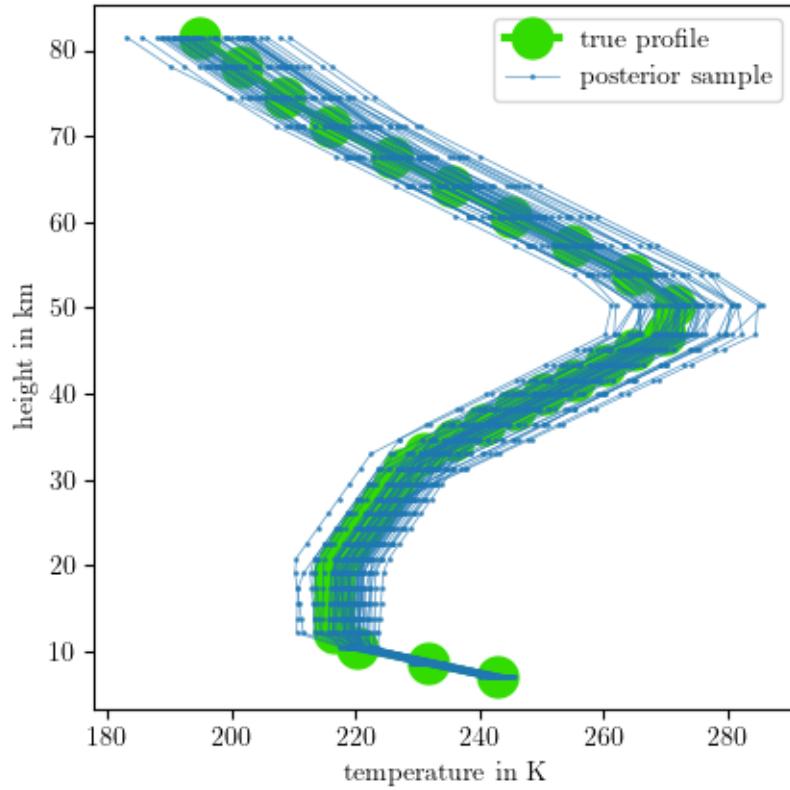


Figure 4.23: We take samples from the posterior distribution, as plotted in Figures 4.18 to 4.21 and plot the corresponding temperature function, see Eq: 4.3.

46. Posterior distribution for pressure and temperature with approximated forward model

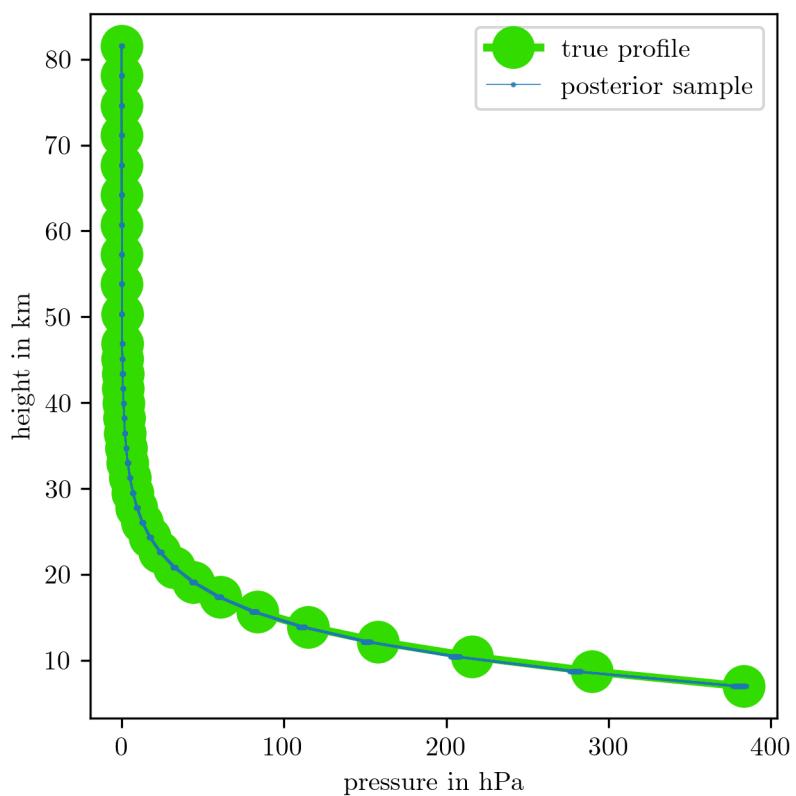
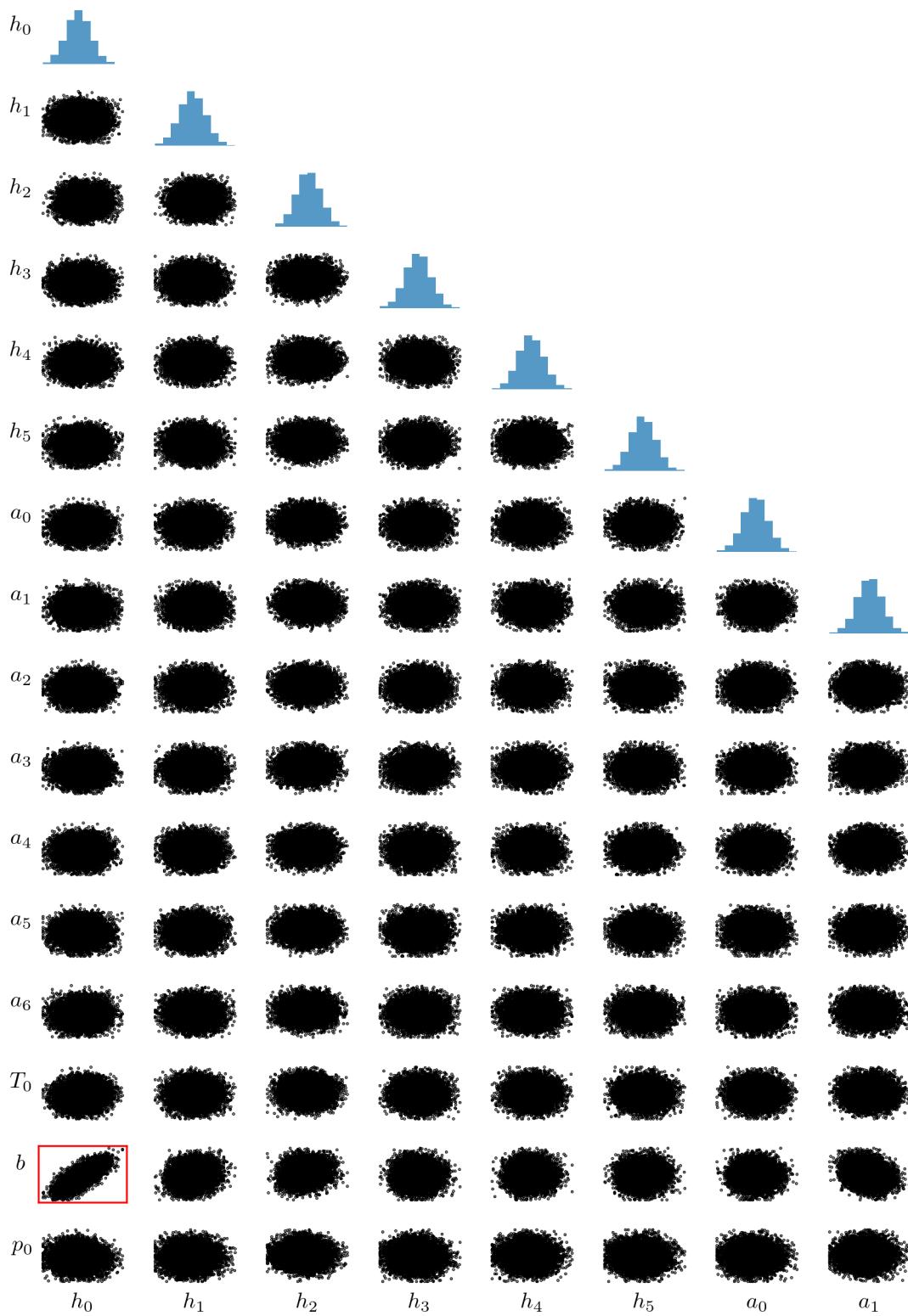
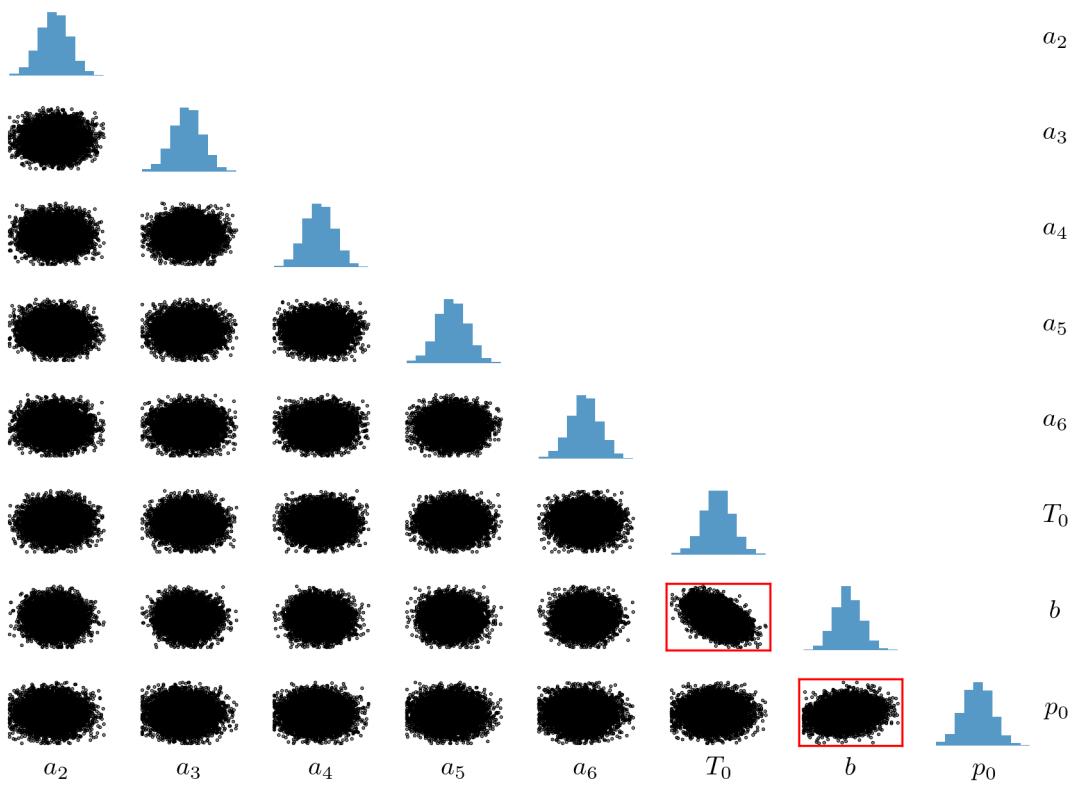


Figure 4.24: We take samples from the posterior distribution, as plotted in Fig. 4.22 and plot the corresponding pressure function, see Eq: 4.17.

**Figure 4.25:** mffjnjjf

486. Posterior distribution for pressure and temperature with approximated forward model



fhgffg

4.7 Error analysis

In this section, we estimate errors due to the function approximations of $f(\lambda)$ and $g(\lambda)$ and how these errors propagate to the marginal posterior. Additionally, we approximate errors of the TT-approximation as well as Monte-Carlo errors when binning up the samples.

Error due to approximation of f and g

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 $\arg \max_{\lambda, \gamma} 0.5\gamma E_f / \log \pi(\lambda, \gamma | \mathbf{y})$ is bound by approximately 5%.

Tensor-train approximation error for the marginal posterior

We calculate the error of the TT approximation of the marginal posterior with the Wasserstein distance $\|x\|$. The wasserstein distance between the normalised true marginal posterior $\pi(\lambda, \gamma | \mathbf{y})$ and the TT approximation $\tilde{\pi}(\lambda, \gamma | \mathbf{y})$ is 0.1.

Error due to grid size and number of sample bins when calculating the full conditinal mean and covariance

When we calculate the mean and covariance matrix of the full conditional $\pi(\mathbf{x} | \mathbf{y})$ we have to bin up the samples of the marginal posterior $\pi(\gamma, \delta | \mathbf{y})$ or use a TT approximation on a predefined grid with a certain number of grid points, we like to give an estimate for this error as well. In doing we bin up samples and use the height $\tilde{\pi}(\boldsymbol{\theta}_d^{(k)})$ for a bin $k = 1, \dots, N_b$ to calculate the mean $\tilde{\mu}_d = \sum_{N_b} \tilde{\pi}(\boldsymbol{\theta}_d^{(k)})$. We compare to the sample mean $\boldsymbol{\mu}_d = \sum_{k=1}^N \boldsymbol{\theta}_d^{(k)} / N$ and calculate the relative error $\|\boldsymbol{\mu}_{\text{samp}} - \boldsymbol{\mu}_{\text{distr}}\| / \|\boldsymbol{\mu}_{\text{samp}}\|$ where $\boldsymbol{\mu}_{\text{samp}} = (\tilde{\mu}_1, \dots, \tilde{\mu}_D)$ and equivalently $\boldsymbol{\mu}_{\text{distr}} = (\tilde{\mu}_1, \dots, \tilde{\mu}_D)$. Here d refers to the $D = 16$ hyper-parameters $\gamma, \lambda, h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4, T_0, p_0, b$.

The relative error behaves proportionally to $1/N$, see Fig. 4.26 and Eq. 2.12, 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. .

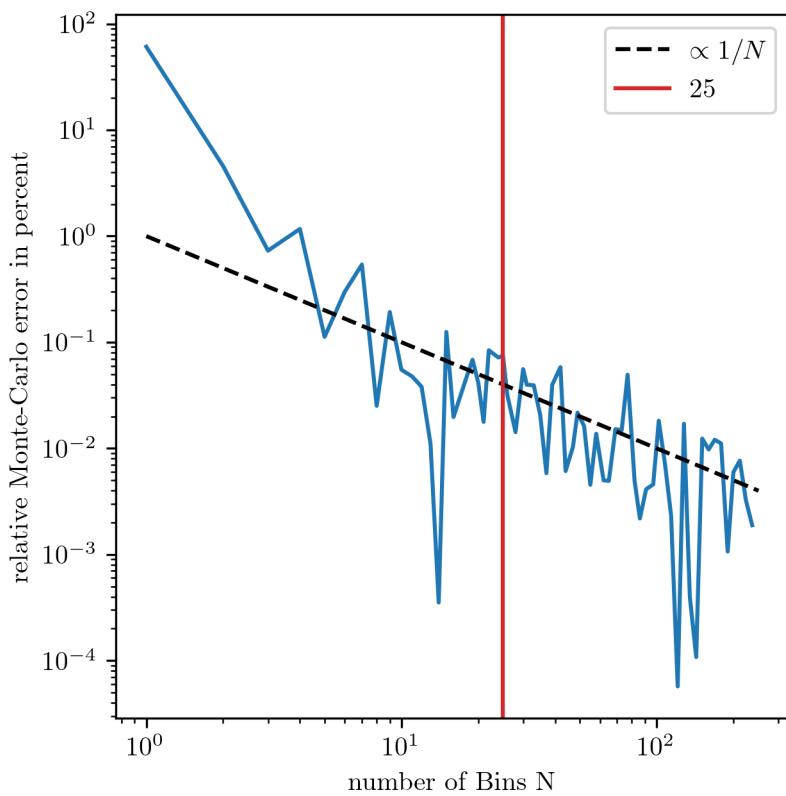


Figure 4.26: Assessment of Monte-Carlo error, where we calculate the relative error of the mean due to binning up the samples compared to the sample mean $\|\boldsymbol{\mu}_{\text{samp}} - \boldsymbol{\mu}_{\text{distr}}\| / \|\boldsymbol{\mu}_{\text{samp}}\|$.

5

Conclusions

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 vs MTC

As already mentioned the regularisation approach only provides one solution, see Fig. 4.16. In Fig. 4.14 we plot samples from the full conditional, which lie 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 vs TT

We can conclude that the TT approximation is faster or as fast as sampling methods. For the marginal posterior $\pi(\gamma, \lambda | \mathbf{y})$ 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.

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 [36] 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.

5.2.1 Intuioton of TT

more correlation hiegher ranks and/or grid point few sweeps and order is important to correlation structure keep ranks as low as possible and incres number of sweeps less correlation fewer rank and gridpoints maybe more cheaper sweeps

5.3 Approximation Errors

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.

5.4 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. 4.16 and Fig. 4.12, 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. 4.18 to 4.22. 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. 4.22. 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.

6

Outlook

6.1 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 $\pi(\mathbf{x}|\mathbf{y}) \approx 1/N_\Gamma \sum_\Gamma \pi(\mathbf{x}, \Gamma^{(k)}|\mathbf{y})$ and include other measurement device specific parameters.

6.2 Methods

Here we point out possibilities for improvement of the currently used methods

6.2.1 TT approximation

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.

6.2.2 Sampling

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 [].

6.2.3 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 have 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.

Appendices

A

Correlation Structure

In the book Gaussian Markov Random Fields [6], Rue and Held demonstrate that a strong correlation between the hyper-parameter μ and the latent field \mathbf{x} can significantly slow down convergence when using samplers, in particular Gibbs samplers. They consider the hierarchical model

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

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

and apply a Gibbs sampler based on the full conditional distributions

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

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

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

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

$$\mu^* \sim q(\mu^* | \mu^{(k-1)}) \quad (\text{A.4})$$

$$\mathbf{x}^{(k)} | \mu^* \sim \mathcal{N}(\mu^* \mathbf{1}, \mathbf{Q}^{-1}) \quad (\text{A.5})$$

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

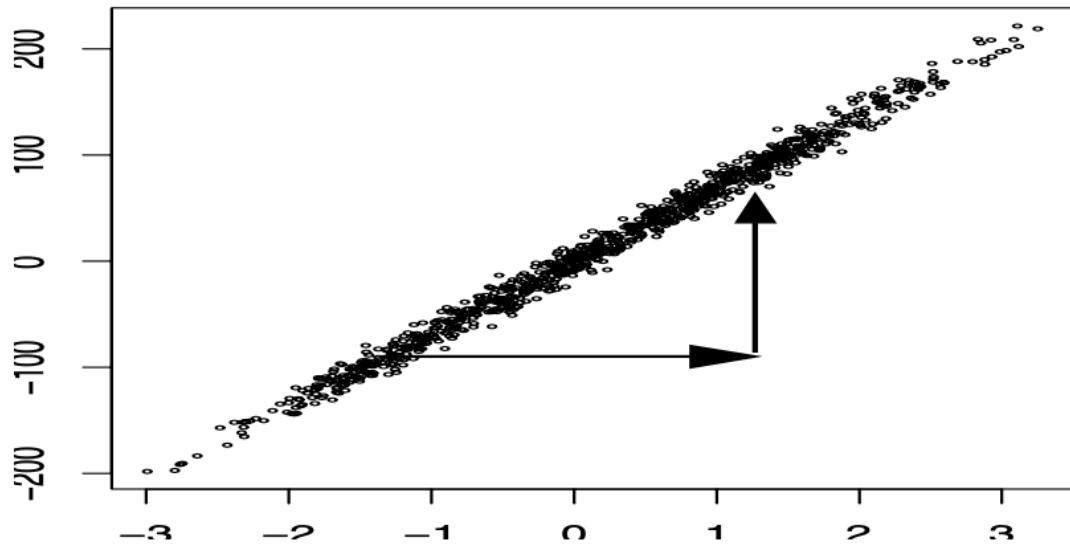


Figure A.1: The figure taken from [6, Figure 4.1 (b)], shows samples from a marginal chain for μ and $\mathbf{1}^T \mathbf{Q} \mathbf{x}^{(k)}$ over 1000 iterations, based on the hierarchical model in Eq. A.1, with an autoregressive process encoded in \mathbf{Q} . The algorithm updates μ and \mathbf{x} successively from their full conditional distributions. The plot displays $(\mu^{(k)}, \mathbf{1}^T \mathbf{Q} \mathbf{x}^{(k)})$, with $\mu^{(k)}$ on the horizontal axis and $\mathbf{1}^T \mathbf{Q} \mathbf{x}^{(k)}$ on the vertical axis. The slow mixing and convergence of μ result from its strong dependence on $\mathbf{1}^T \mathbf{Q} \mathbf{x}^{(k)}$, while the sampler permits only axis-aligned (horizontal and vertical) and does not allow diagonal moves, as illustrated by the arrows.

B

Measure theory

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 $\mathbb{P} : \mathcal{F} \rightarrow \mathbb{R}$ 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 [44] [16] for further reading.

B.1 probailty 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 $\mathbb{P}\left(\bigcup_{j \in \mathbb{N}} A_j\right) = \sum_{j \in \mathbb{N}} \mathbb{P}(A_j)$.

B.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 $A^C := A/\Omega \in \mathcal{F}$
- if $A_1, A_2, \dots \in \mathcal{F}$ 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 $A_1, A_2, \dots \in \mathcal{F}$, 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.

C

Additional Figures

C.1 Integrated Autocorrelation plots

C.1.1 Ozone

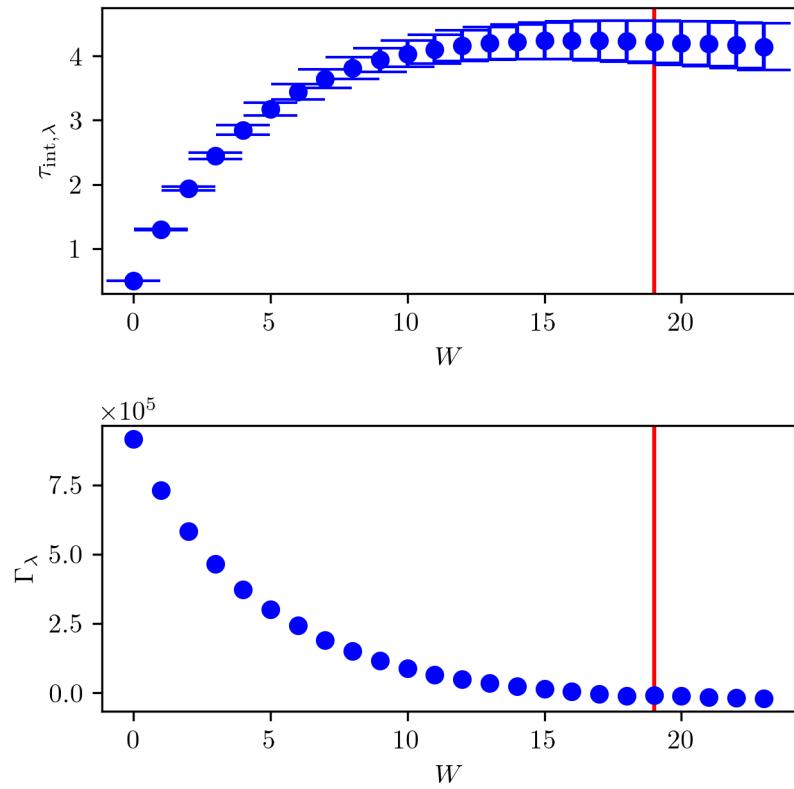


Figure C.1: IACT for samples $\lambda \sim \pi(\cdot | \gamma, \mathbf{y})$ based on the approximated forward model.

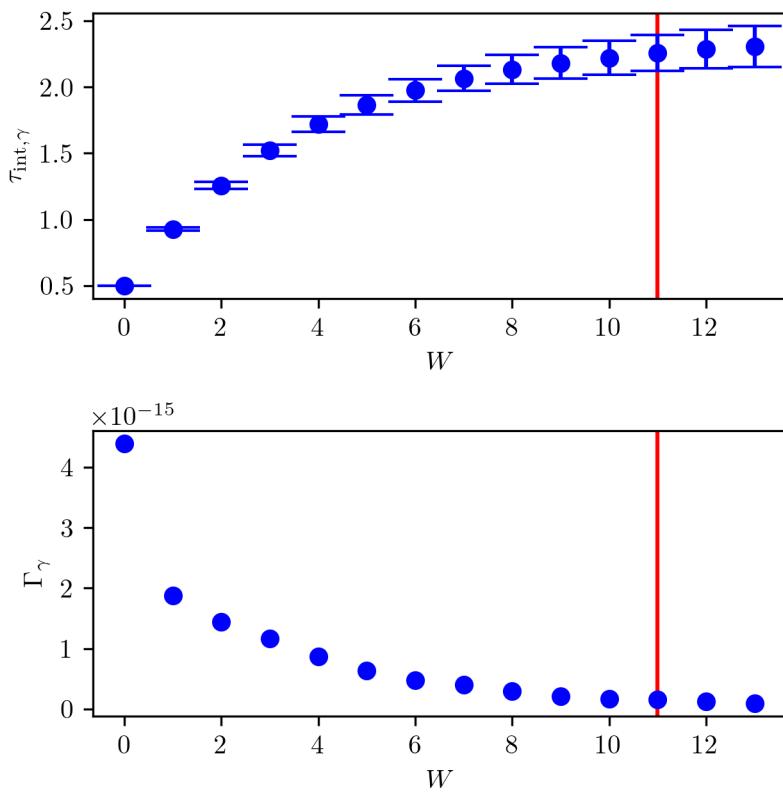


Figure C.2: IACT for samples $\gamma \sim \pi(\cdot | \lambda, \mathbf{y})$ based on the approximated forward model.

C.1.2 Pressure and Temperature

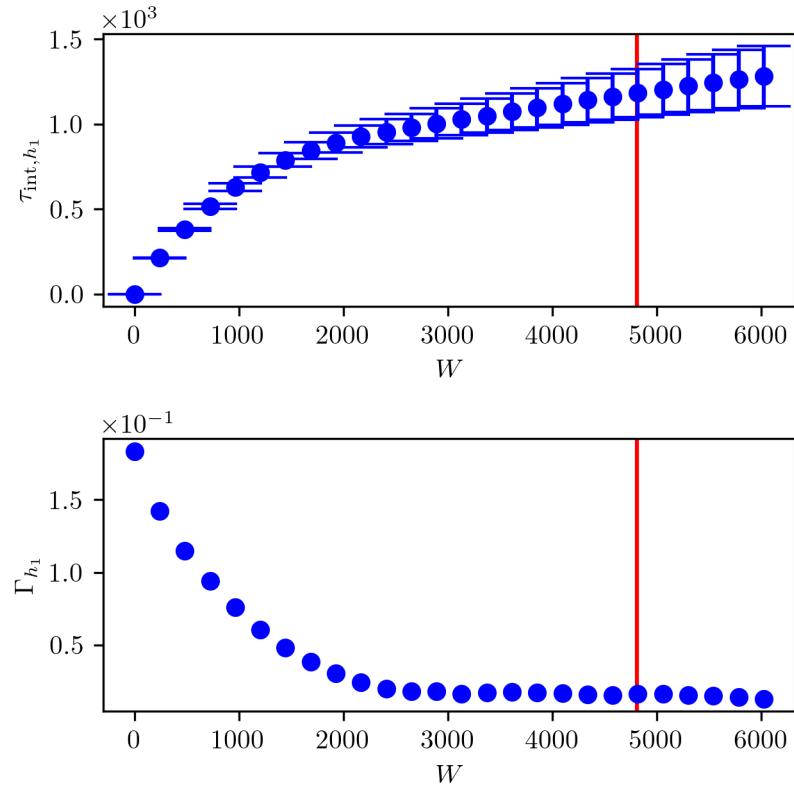


Figure C.3: IACT and autocorrelation function for samples $h_1 \sim \pi(\cdot | h_{T,2}, h_{T,3}, h_{T,4}, h_{T,5}, h_{T,6}, a_0, a_1, a_2, a_3, a_4, a_5, a_6, T_0, b, p_0, \mathbf{y})$

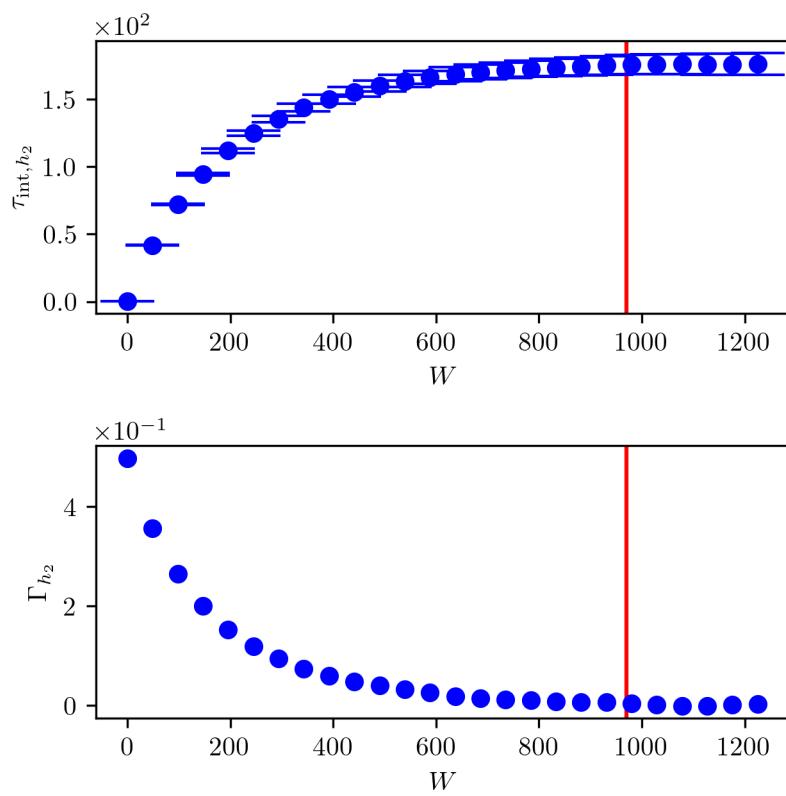


Figure C.4: IACT and autocorrelation function for samples $h_2 \sim \pi(\cdot | h_1, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4, a_5, a_6, T_0, b, p_0, y)$

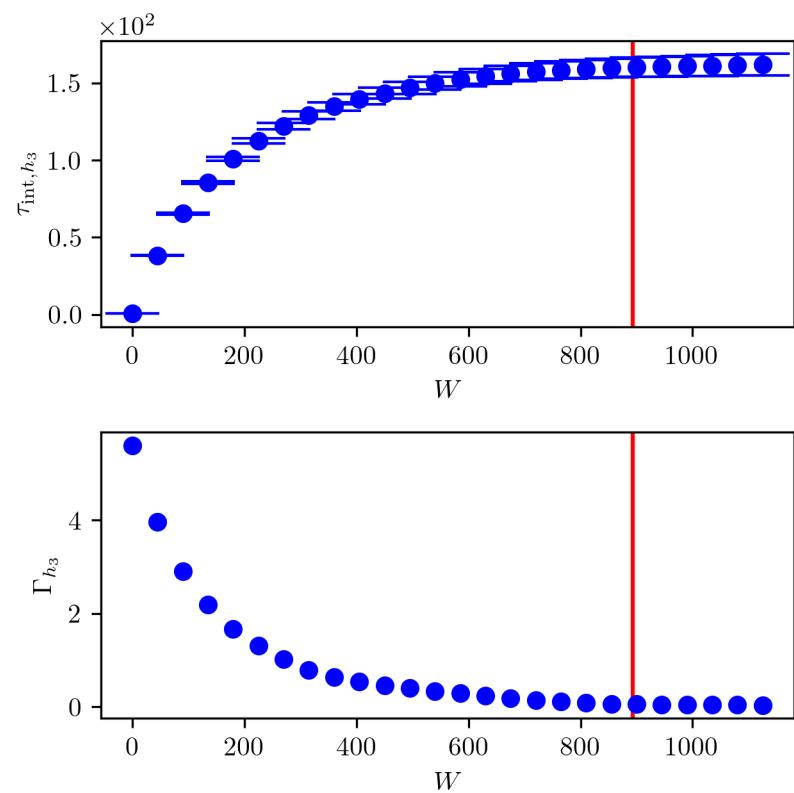


Figure C.5: IACT and autocorrelation function for samples $h_3 \sim \pi(\cdot | h_1, h_2, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4, a_5, a_6, T_0, b, p_0, \mathbf{y})$

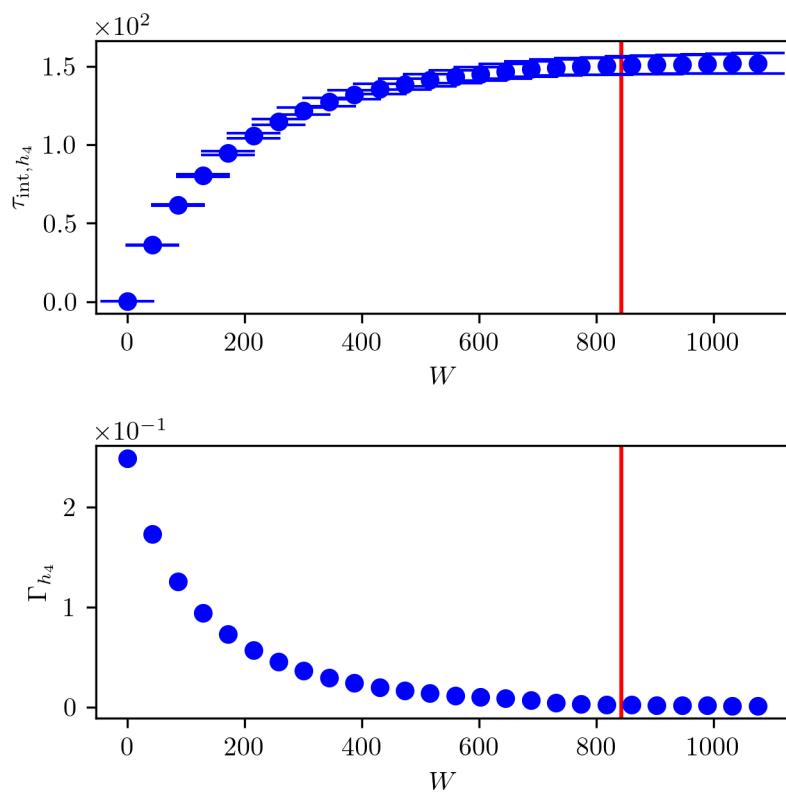


Figure C.6: IACT and autocorrelation function for samples $h_4 \sim \pi(\cdot | h_1, h_2, h_3, h_5, h_6, a_0, a_1, a_2, a_3, a_4, a_5, a_6, T_0, b, p_0, y)$

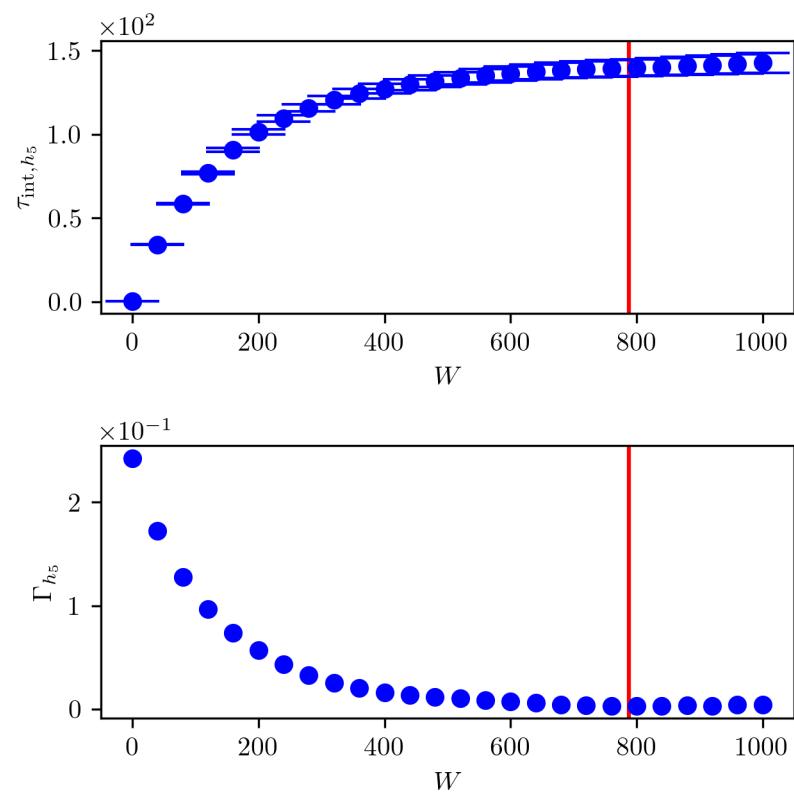


Figure C.7: IACT and autocorrelation function for samples $h_5 \sim \pi(\cdot | h_1, h_2, h_3, h_4, h_6, a_0, a_1, a_2, a_3, a_4, a_5, a_6, T_0, b, p_0, \mathbf{y})$

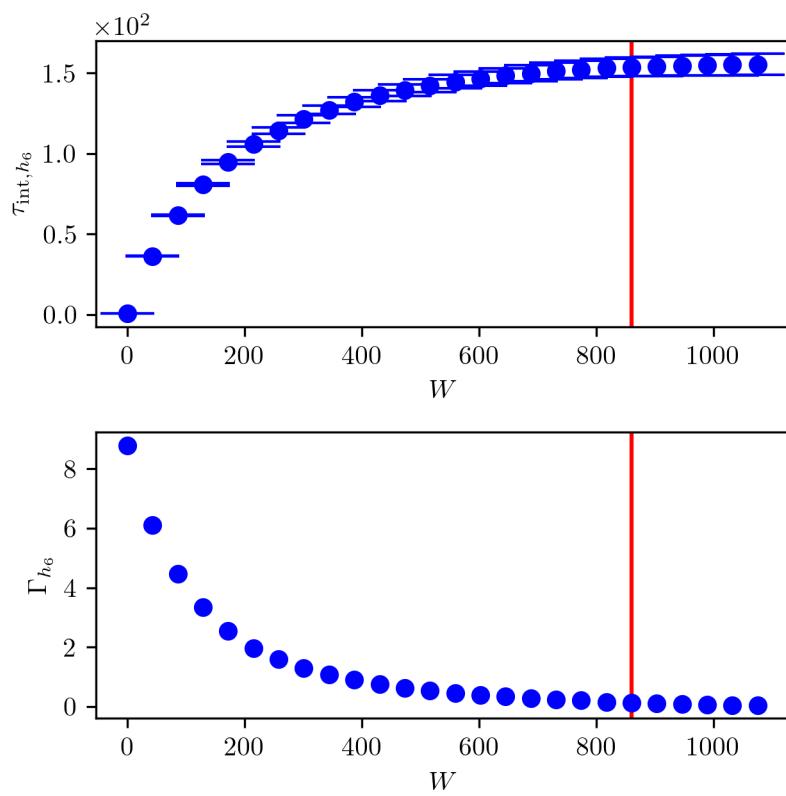


Figure C.8: IACT and autocorrelation function for samples $h_6 \sim \pi(\cdot | h_1, h_2, h_3, h_4, h_5, a_0, a_1, a_2, a_3, a_4, a_5, a_6, T_0, b, p_0, y)$

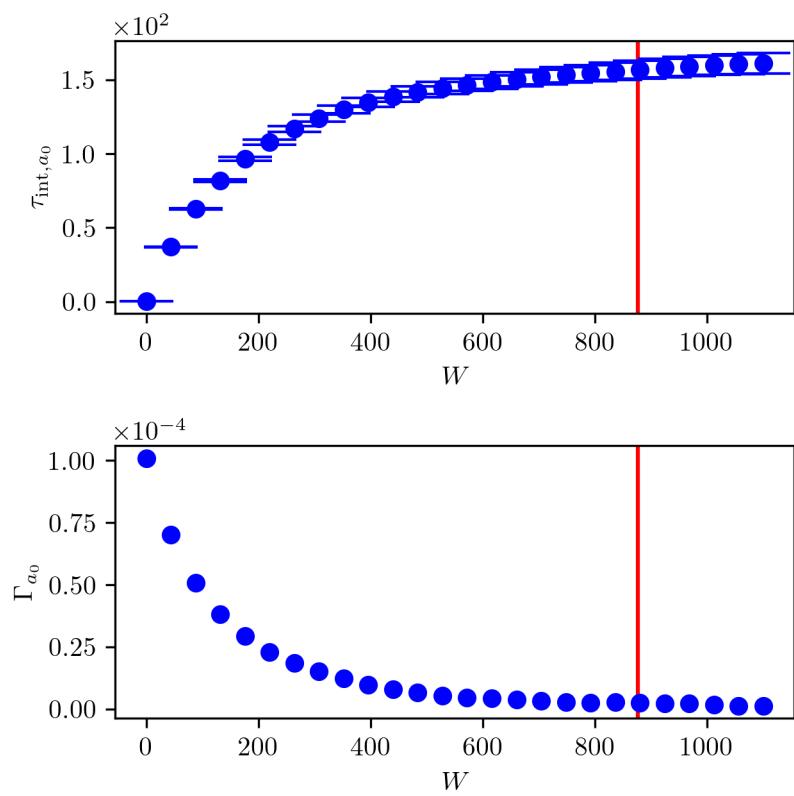


Figure C.9: IACT and autocorrelation function for samples $a_0 \sim \pi(\cdot | h_1, h_2, h_3, h_4, h_5, h_6, a_1, a_2, a_3, a_4, a_5, a_6, T_0, b, p_0, \mathbf{y})$

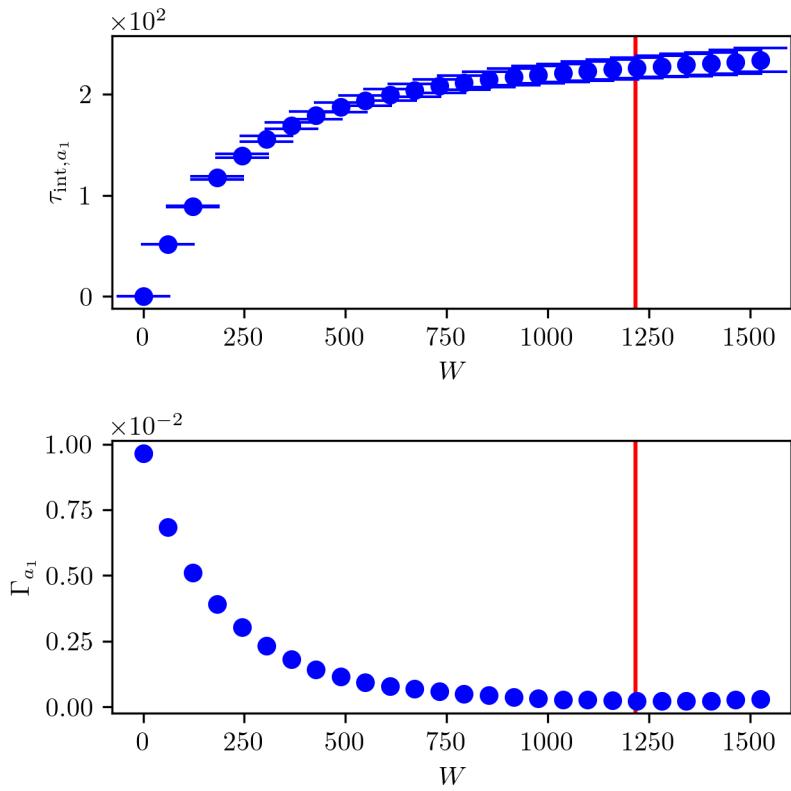


Figure C.10: IACT and autocorrelation function for samples $a_1 \sim \pi(\cdot | h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_2, a_3, a_4, a_5, a_6, T_0, b, p_0, \mathbf{y})$

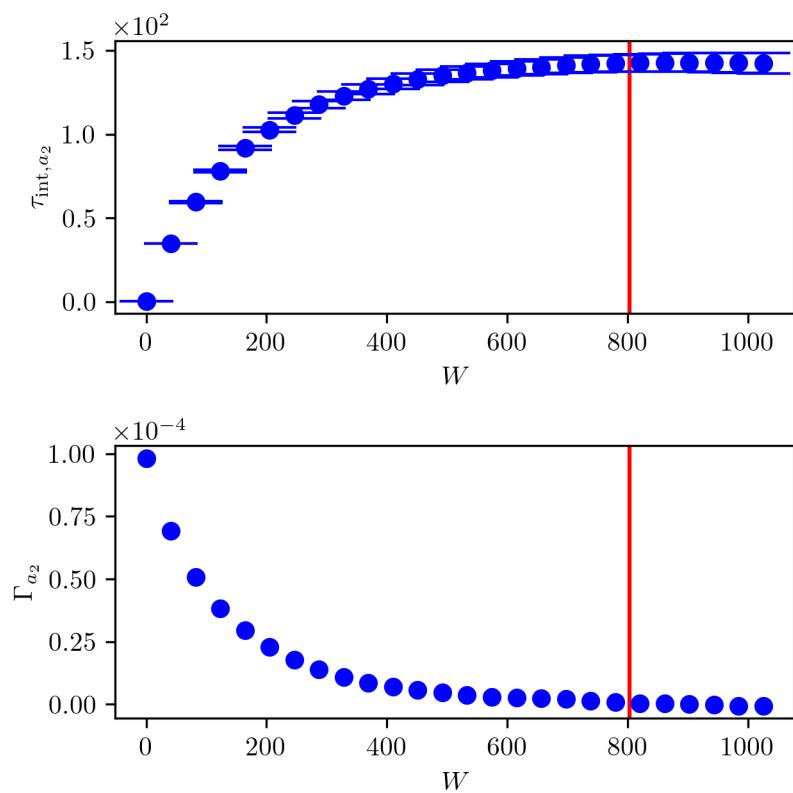


Figure C.11: IACT and autocorrelation function for samples $a_2 \sim \pi(\cdot | h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_3, a_4, a_5, a_6, T_0, b, p_0, \mathbf{y})$

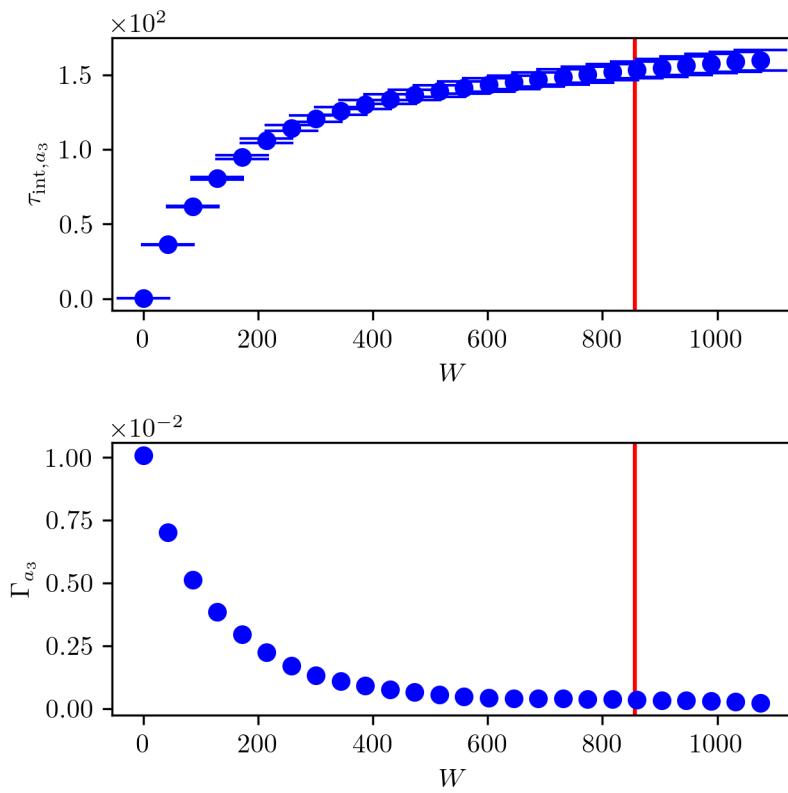


Figure C.12: IACT and autocorrelation function for samples $a_3 \sim \pi(\cdot | h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_4, a_5, a_6, T_0, b, p_0, \mathbf{y})$

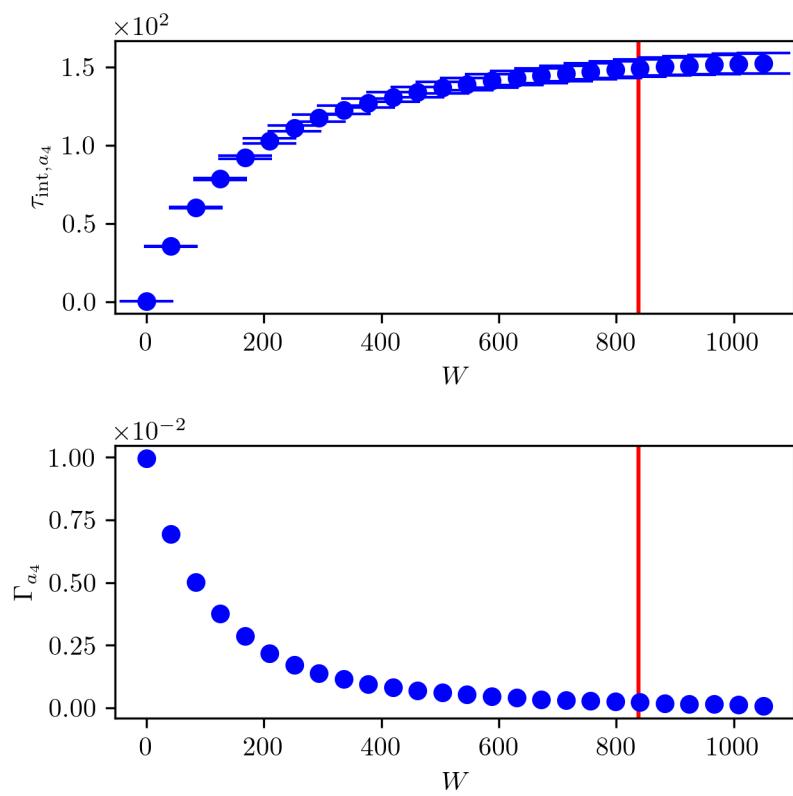


Figure C.13: IACT and autocorrelation function for samples $a_4 \sim \pi(\cdot | h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_4, a_5, a_6, T_0, b, p_0, \mathbf{y})$

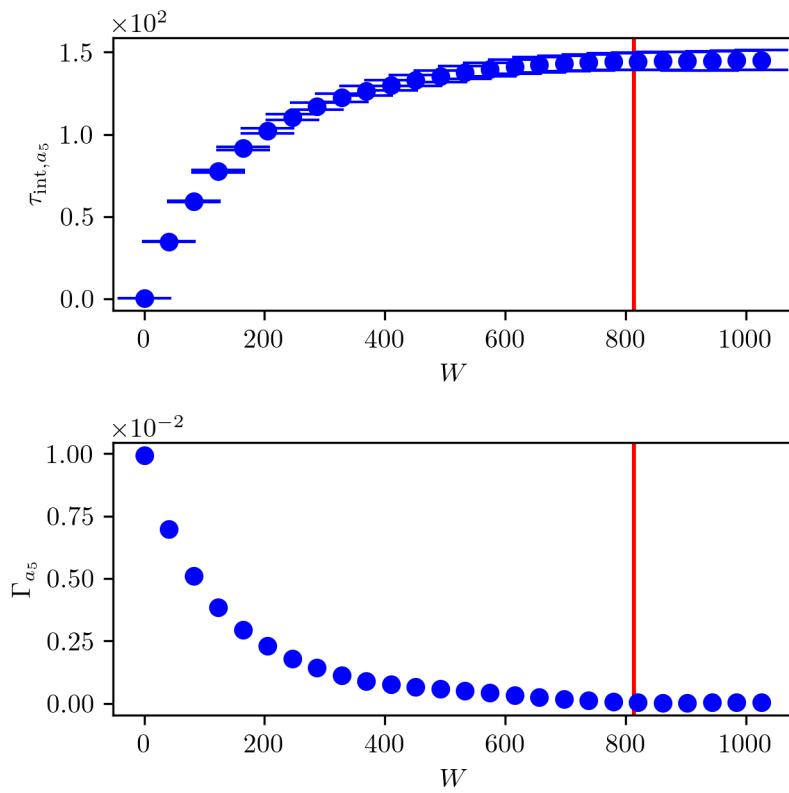


Figure C.14: IACT and autocorrelation function for samples $a_5 \sim \pi(\cdot | h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4, a_5, T_0, b, p_0, \mathbf{y})$

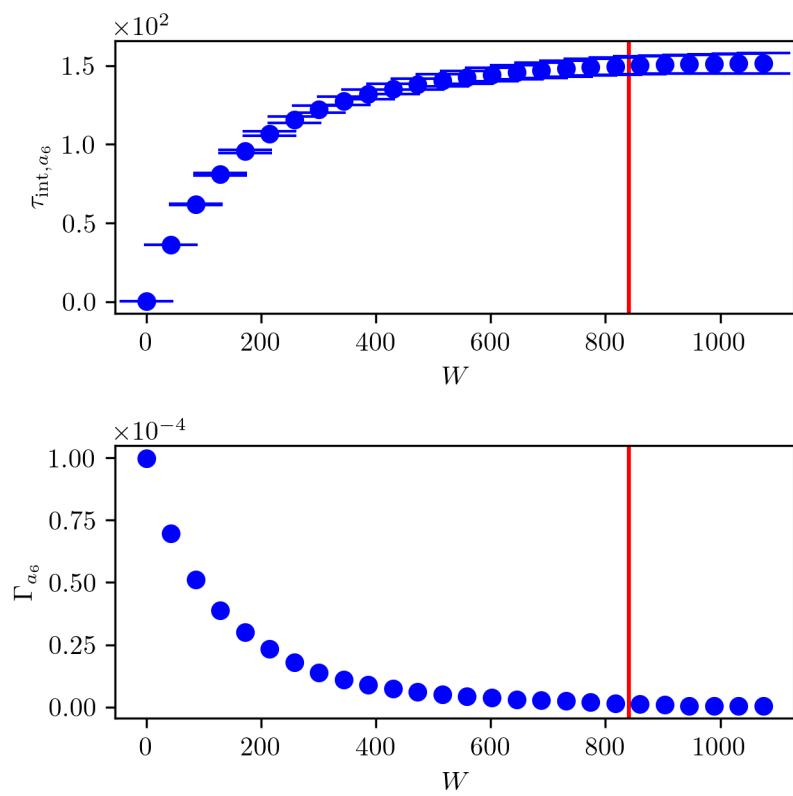


Figure C.15: IACT and autocorrelation function for samples $a_6 \sim \pi(\cdot | h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4, a_5, T_0, b, p_0, \mathbf{y})$

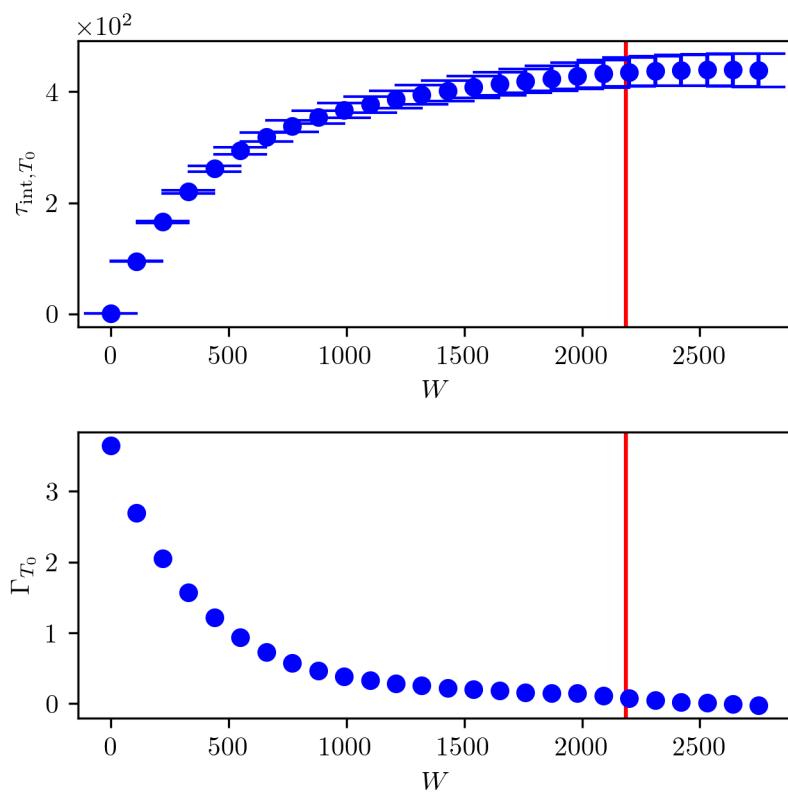


Figure C.16: IACT and autocorrelation function for samples $T_0 \sim \pi(\cdot | h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4, a_5, a_6, b, p_0, y)$

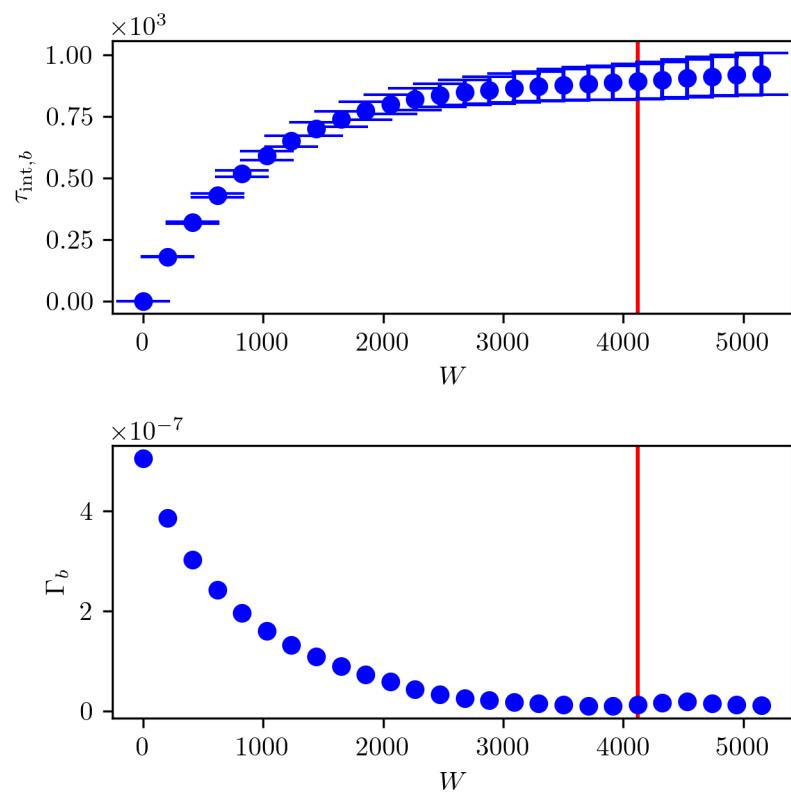


Figure C.17: IACT and autocorrelation function for samples $b \sim \pi(\cdot | h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4, a_5, a_6, T_0, p_0, \mathbf{y})$

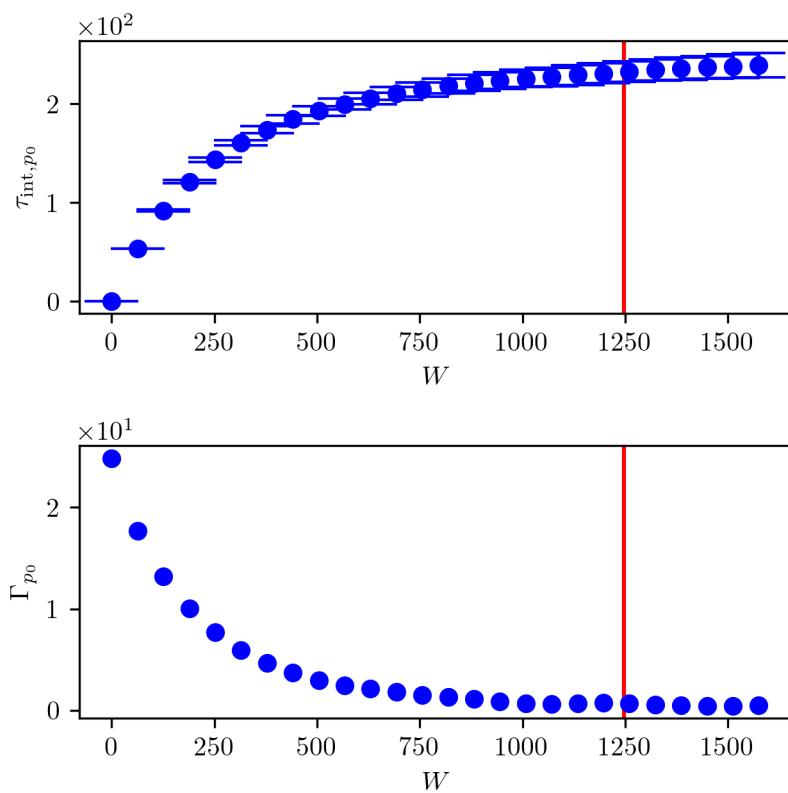


Figure C.18: IACT and autocorrelation function for samples $p_0 \sim \pi(\cdot | h_1, h_2, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4, a_5, a_6, T_0, b, \mathbf{y})$

C.2 Ozone

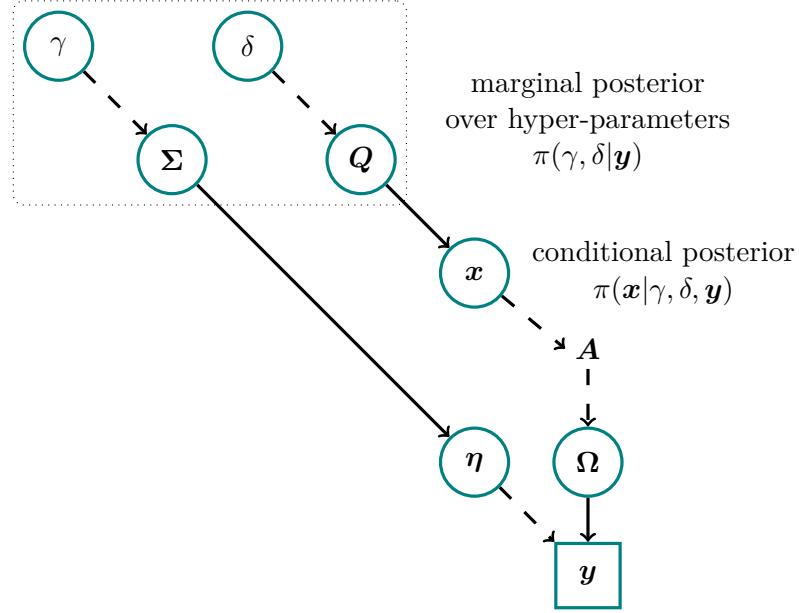


Figure C.19: Directed acyclic graph for modelling and measuring process of ozone highlighting the marginal and then conditional (MTC) scheme. The hyper-parameters δ and γ determine the noise covariance $\Sigma = \gamma^{-1} \mathbf{I}$ for the random noise vector $\eta \sim \mathcal{N}(0, \gamma^{-1} \mathbf{I})$ and the prior precision matrix $Q = \delta \mathbf{L}$ for the normal distribution over $x \sim \mathcal{N}(0, \delta \mathbf{L})$, where \mathbf{L} is a graph Laplacian, see Eq. 4.7. In the MTC scheme we evaluate the marginal posterior over the hyper-parameters $\pi(\gamma, \delta | \mathbf{y})$ as in Eq. ?? first and then the conditional posterior $\pi(x | \gamma, \delta, \mathbf{y})$ as in Eq. 4.12. The MTC scheme allows to evaluate the marginal posterior distribution over the hyper-parameters δ, γ independent of x , breaking the correlation structure. Through the forward model $\mathbf{A}_{NL} \approx \mathbf{M}\mathbf{A}_L$ and the parameter x we generate a space of all measurable from which we randomly observe a data set \mathbf{y} including random noise η .

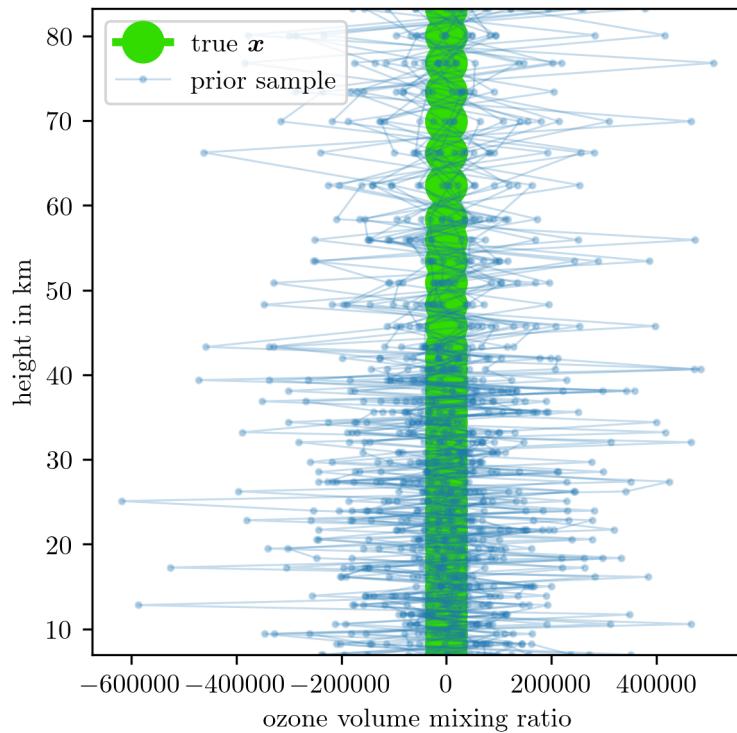


Figure C.20: We draw samples from ozone prior distribution $\mathbf{x} \sim \mathcal{N}(0, \delta \mathbf{L})$ after generating a sample from the hyper-prior distribution $\delta \sim \mathcal{T}(1, 10^{-10})$. Note that since the variance of prior samples is very large compared to the ozone volume mixing ratios, the ozone profile appears to be constant, which is not the case, see e.g. Fig. 4.12.

C.3 Pressure over Temperature

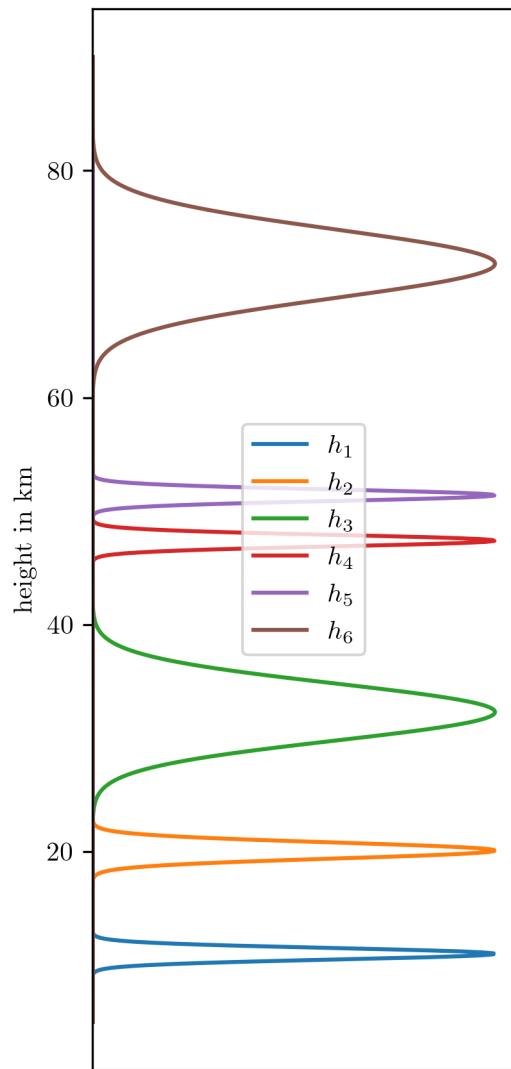


Figure C.21: Prior distributions $\pi(\mathbf{h}_T)$, which we choose so that they do not overlap and not conflict with the temperature function 4.3

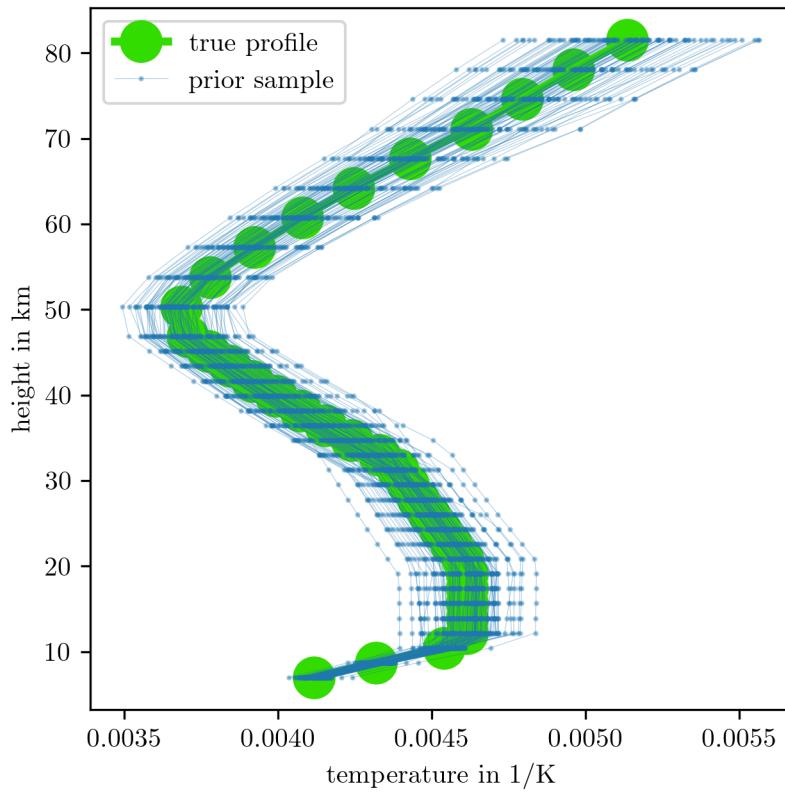


Figure C.22: Prior samples of the inverted temperature profile.

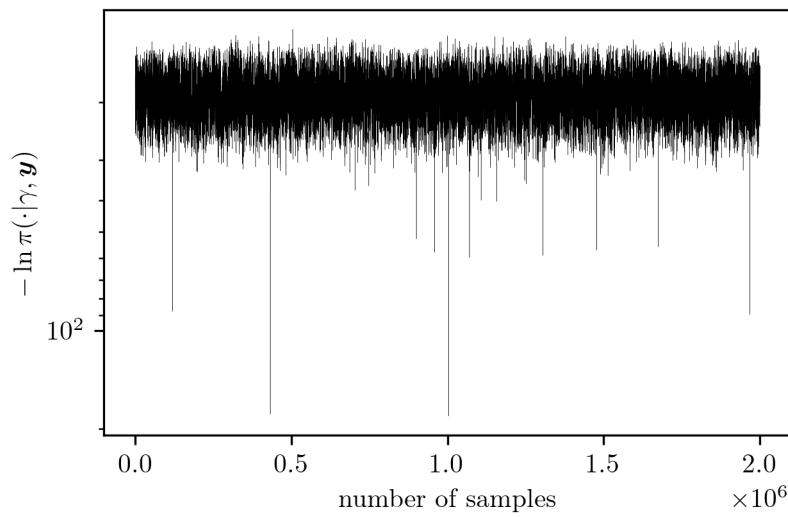


Figure C.23: Output trace of the t-walk on the posterior distribution $\pi(p_0, b, \mathbf{h}_T, \mathbf{a} | \gamma, \mathbf{y})$.

References

- [1] 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.
- [2] Schwartz M. et al. *MLS/Aura Level 2 Ozone (O3) Mixing Ratio V005*. https://disc.gsfc.nasa.gov/datasets/ML203_005/summary?keywords=mls%20o3. [Online; accessed 25/04/24]. 2020.
- [3] 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]. 2016.
- [4] Gareth O. Roberts and Jeffrey S Rosenthal. “General state space Markov chains and MCMC algorithms”. In: *Probability Surveys* 1 (2004), pp. 20–71.
- [5] S. P. Meyn and R.L. Tweedie. *Markov Chains and Stochastic Stability. 2nd Edition*. New York: Cambridge University Press, 2009.
- [6] Havard Rue and Leonhard Held. *Gaussian Markov random fields: theory and applications*. London: CRC press, 2005.
- [7] Charles W. Champ and Andrew V. Sills. “The Generalized Law of Total Covariance”. In: *preprint* (2022). URL: <https://arxiv.org/abs/2205.14525>.
- [8] Charles J Geyer. “Practical markov chain monte carlo”. In: *Statistical science* (1992), pp. 473–483.
- [9] 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.
- [10] Ulli Wolff. “Monte Carlo errors with less errors”. In: *Computer Physics Communications* 156.2 (2004), pp. 143–153. URL: <https://www.sciencedirect.com/science/article/pii/S0010465503004673>.
- [11] Ulli Wolff. “Monte Carlo errors with less errors”. In: *Computer Physics Communications* 156.2 (2004), pp. 143–153.
- [12] 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.
- [13] 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.
- [14] 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.
- [15] 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.
- [16] M. Capiński and P.E. Kopp. *Measure, Integral and Probability. Springer Undergraduate Mathematics Series*. London: Springer-Verlag London, 2004.

- [17] M. Simonnet. *Measures and Probabilities*. New York: Springer-Verlag, 1996.
- [18] Vesa Kaarnioja. *Inverse Problems. Eighth lecture*. <https://vesak90.userpage.fu-berlin.de/ip23/week8.pdf>. [Online; accessed 10/04/25]. 2023.
- [19] Philip J Davis and Philip Rabinowitz. *Methods of numerical integration*. San Diego, CA: Academic Press, Inc., 1984.
- [20] 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.
- [21] 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.
- [22] Ivan V Oseledets. “Tensor-train decomposition”. In: *SIAM Journal on Scientific Computing* 33.5 (2011), pp. 2295–2317.
- [23] Marcel Berger. *Geometry I. 4th Edition*. Berlin Heidelberg: Springer-Verlag, 2009.
- [24] Katsumi Nomizu and Takeshi Sasaki. *Affine differential geometry*. Cambridge: Cambridge University Press, 1994.
- [25] 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.
- [26] Per Christian Hansen. “Regularization, GSVD and truncated GSVD”. In: *BIT numerical mathematics* 29.3 (1989), pp. 491–504.
- [27] 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.
- [28] 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. URL: <https://www.sciencedirect.com/science/article/pii/B9780128235041000131>.
- [29] C. Readings. *Envisat MIPAS An Instrument for Atmospheric Chemistry and Climate Research*. Noordwijk: ESA Publications Division, 2000.
- [30] Iouli E Gordon et al. “The HITRAN2020 molecular spectroscopic database”. In: *Journal of Quantitative Spectroscopy and Radiative Transfer* 277 (2022), p. 107949.
- [31] 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.
- [32] George B. Rybicki and Alan P. Lightman. *Radiative processes in Astrophysics*. Weinheim: Wiley-VCH, 2004.
- [33] 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.
- [34] 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.
- [35] H.M. Pickett. “Microwave Limb Sounder THz module on Aura”. In: *IEEE Transactions on Geoscience and Remote Sensing* 44.5 (2006), pp. 1122–1130.
- [36] Ulli Wolff. *UWerr.m Version6*. <https://www.physik.hu-berlin.de/de/com/ALPHAssoft>. [Online; accessed 5/11/23]. 2004.

- [37] Yu-Xiang Wang et al. "Trend Filtering on Graphs". In: *Journal of Machine Learning Research* 17.105 (2016), pp. 1–41.
- [38] Daniel Simpson, Finn Lindgren, and Håvard Rue. "Think continuous: Markovian Gaussian models in spatial statistics". In: *Spatial Statistics* 1 (2012), pp. 16–29. URL: <https://www.sciencedirect.com/science/article/pii/S2211675312000048>.
- [39] Josef Dick, Frances Y. Kuo, and Ian H. Sloan. "High-dimensional integration: The quasi-Monte Carlo way". In: *Acta Numerica* 22 (2013), 133–288.
- [40] 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.
- [41] Per Christian Hansen. *Discrete Inverse Problems: Insight and Algorithms*. Philadelphia: SIAM, 2010.
- [42] 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.
- [43] J. Andrés Christen and Colin Fox. *The t-walk software*. <https://www.cimat.mx/~jac/twalk/>. [Online; accessed 25/11/24].
- [44] Greg Lawler. *Notes on probability*. <https://www.math.uchicago.edu/~lawler/probnotes.pdf>. [Online; accessed 10/04/25]. 2016.