Hierarchical Bayesian Modelling and Posterior Inference Applied to an Atmospheric Limb-Sounder Measuring Ozone

Lennart Golks
Department of Physics

A thesis submitted for the degree of
Doctor of Philosophy

October 2025

Acknowledgements

Abstract

In this thesis, we develop a hierarchical Bayesian model based on the radiative transfer equation for a simplified atmospheric limb-sounder targeting ozone, as described in [41]. To ensure effective measurements, we briefly assess the informativity of different measurement approaches via a singular value decomposition of the forward model and adapt the data collection accordingly. Following [17], we utilise the marginal and then conditional scheme to provide posterior distributions of hyper-parameters and ozone profiles, and compare with a regularisation approach. After approximating the non-linear forward model with an affine map, we extend our hierarchical Bayesian framework and the marginal and then conditional scheme to jointly infer posterior pressure, temperature and ozone.

The main contribution of this work is the application of tensor-trains to approximate high-dimensional posterior probability distributions [8, 11]. This enables us to generate samples from the target distribution with far fewer function evaluations compared to the t-walk sampling algorithm [6]. Tensor-train methods require a predefined grid and a "normalisation constant" so that function outputs are within computer precision, but once defined, they reduce the function evaluations per independent sample significantly. Another advantage of the tensor-train format is that marginal probability distributions, useful for characterisation of integrals via quadrature, can be calculated at a low computational cost, without any sampling. To further improve tensor-train methods, we suggest future work should focus on lowering tensor ranks, calculating "normalisation constants" to avoid numerical issues and reducing correlation structures between parameters automatically, all of which we currently have to do by exploratory analysis. Additionally, choosing accurate interpolation schemes between grid points is crucial to improving the effectiveness of the approximation.

Our results show that a hierarchical Bayesian approach, which quantifies posterior mean and variance of the parameter (ozone), provides more information than a regularisation approach at comparable computational time. In regions where the signal strength is low and the data is noise-dominated, we can not recover ozone structures from the ground truth. When including pressure and temperature describing hyper-parameters within our hierarchical Bayesian model, we find a strong correlation between ozone and pressure, whereas the model and data are uninformative about temperature. For future work, we recommend developing a more physically informed parametrised model for ozone within the atmosphere, incorporating atmospheric chemistry and other important processes.

Contents

List of Abbreviations

CDF . . . . . Cumulative Distribution Function

DAG . . . . . Directed Acyclic Graph

HITRAN . . . . . High Resolution Transmission

IACT . . . . . Integrated Autocorrelation Time

IRT . . . . . Inverse Rosenblatt Transform

L . . . . . Linear

MCMC . . . . . Markov Chain Monte-Carlo

MH . . . . . Metropolis–Hastings

MIPAS . . . . . Michelson Interferometer for Passive Atmospheric Sounding

MLS . . . . . Microwave Limb Sounder

MTC . . . . . Marginal and then Conditional

MWG . . . . . Metropolis within Gibbs

NASA . . . . . National Aeronautics and Space Administration

PDF . . . . . Probability Density Function

RMS . . . . . Root Mean Square

RTE . . . . . Radiative Transfer Equation

RTO . . . . . Randomise then Optimise

SIRT . . . . . Squared Inverse Rosenblatt Transform

STD . . . . . Standard Deviation

SVD . . . . . Singular Value Decomposition

TT . . . . . Tensor-Train

VMR . . . . . Volume Mixing Ratio

x

x

1 Introduction

Here we briefly describe the standard currently used to retrieve atmospheric trace gas concentrations, e.g. ozone concentration, from limb-sounding measurements and what motivates us to employ a hierarchical Bayesian framework to address this inverse problem. We explain how our approach contributes to and improves upon existing methods. Lastly, we provide the reader with the thesis structure.

1.1 Motivation

Currently, the only operating ozone limb sounder is the Microwave Limb Sounder (MLS) on NASA's Aura satellite. This satellite is gradually drifting away from its orbit and scheduled to be phased out by 2026 [13]. A group led by Harald Schwefel has proposed an alternative approach to fill this observational gap using a much smaller platform such as a 6U CubeSat (roughly 30cm $\times$ 15cm $\times$ 10cm) [60]. The proposed system includes a disk-shaped resonator targeting a narrow frequency band and converting the thermal radiation emitted by ozone molecules from the terahertz region to the optical domain [56, 52]. This frequency conversion offers a cost-effective and energy-efficient solution as it avoids the need for large, energy-hungry cooling devices that are traditionally required to capture terahertz signals. Instead, signal acquisition in the optical domain can be implemented by using compact, cheap, and low-power photonic technologies.

Currently, the inverse problem to retrieve any trace gas from limb-sounding data is approached by the atmospheric physics community using optimisation and regularisation techniques developed in the 1970s [45, 33]. These methods focus on finding the "best fit to data but not the best fit to parameters" [57]. Instead, we employ a hierarchically structured Bayesian framework to provide a distribution of ozone profiles, which represents

multiple possible solutions according to some given data. This probabilistic approach allows us to determine meaningful estimates and uncertainties of parameters.

1.2 Research Gap and Contribution

As already mentioned, currently the MLS retrieval algorithm [32] is based on the "optimal estimation" method from [45]. This approach provides a point estimate by fitting parameters to some data and iteratively minimising a squared residual norm, penalised against a chosen regularisation. However, this does not provide comprehensive information about the parameters' underlying correlation structures can lead to unphysical results, e.g. negative ozone concentration values [51], and biased solutions where the bias is then removed based on empirical decisions [34, 21]. Errors are provided by a local derivative of the forward map at the optimal solution, which is inherently highly sensitive to that specific point in the parameter space. Furthermore, these regularisation methods condition on external point estimates of other parameters such as temperature or pressure [32]. In Bayesian modelling all unknown parameters are treated as random variables [28]. Further, a hierarchically ordered model incorporates unknown hyper-parameters, which, e.g. model the noise and control the smoothness of the ozone profile and includes hyper-parameters and parameters in the retrieval process. Therefore, a hierarchical Bayesian framework is able to model conditional dependences between parameters and hyper-parameters. Furthermore, a Bayesian approach provides posterior probability distributions over a range of feasible solutions for some given data. Livesey et al. [32] report "unexpected spectrally correlated noise" on the MLS Aura, so here is another real reason why one should include noise in the model.

In this thesis the marginal and then conditional (MTC) method [17] is utilised and a hierarchical Bayesian model based on the radiative transfer equation (RTE) is developed to provide posterior distributions of ozone as well as temperature and pressure profiles. First, we neglect the non-linearity of the RTE and employ a linear-Gaussian hierarchical Bayesian model. Since the RTE is weakly non-linear we use the results based on the linearised RTE to find an affine map that approximates the non-linear forward model. This appears to be another novelty in the field of atmospheric remote sensing. Lastly, the hierarchical Bayesian framework is extended to include pressure and temperature. The MTC scheme is a relatively new method within the Bayesian community,and we are the first, to the best of our knowledge, to apply it to a forward model based on the RTE and to jointly provide posterior ozone, pressure, and temperature profiles.

Instead of using sampling algorithms to characterise the posterior probability distribution of the hierarchical Bayesian model we approximate the distribution directly utilising

the tensor-train (TT) format [8]. This allows us to generate independent samples from a TT approximation via a scheme similar to the inverse Rosenblatt transform (IRT) [11] with far fewer function evaluations compared to conventional samplers. Further, in the TT format one can calculate marginal probability distributions of each hyper-parameter and evaluate integrals via quadrature without any sampling.

1.3 Thesis Structure

In Chapter 2, we give a brief overview of the key methods used, along with references for further reading. Chapter 3 introduces the simplified forward model based on the RTE. Here we test several measurement cases and assess the information content using a singular value decomposition (SVD) to understand the forward model and determine the most effective measurement method. Based on those results and a ground truth, we simulate noisy data for an idealised limb sounder within a simplified atmosphere. Then in Chapter 4, a linear-Gaussian hierarchical Bayesian model based on the linearised RTE is constructed. We discuss some prior modelling choices, apply the MTC scheme and compare posterior ozone profiles and a regularisation approach to a ground truth ozone profile. Using those results we find an affine map in Chapter 5 to approximate the non-linear forward model. In Chapter 6, the previously built hierarchical Bayesian model is extended to include hyper-parameters corresponding to pressure and temperature. Again, we touch on some prior modelling choices and use the MTC method to provide joint estimates of posterior ozone, pressure and temperature profiles. Additionally, some important aspects for improving the effectiveness and stability of TT approximations are highlighted. Lastly, in Chapter 7 we summarise and discuss our results and provide an outlook for future work. All programming and analysis in this thesis are done in Python, and the reported computation times are taken on a MacBook Pro from 2019 with a 2.4 GHz quad-core Intel i5 processor.

2

Theoretical and Technical Background

In this chapter, we introduce the hierarchical Bayesian approach to inverse problems, along with key concepts of Markov Chain Monte Carlo (MCMC) methods and tensor-train (TT) approximations for high-dimensional probability distributions. We keep it as general as possible. Specific sampling algorithms are not introduced here, as they are specifically tailored towards the structure of the forward model and the particular problem. Therefore, they will be presented in detail when applied.

2.1 Hierarchical Bayesian Inference

Assume we observe some data

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

based on a forward model $\boldsymbol{A}(\boldsymbol{x})$, which may be non-linear, an unknown parameter vector $\boldsymbol{x}$ and some additive random noise $\boldsymbol{\eta}$.

Naturally, due to the noise, the observation process in Eq. 2.1 is a random process. Hence, in Bayesian modelling, the aim is to determine a probability distribution over the parameter $\boldsymbol{x}$ given some data $\boldsymbol{y}$. Further, a hierarchical Bayesian model incorporates (auxiliary) hyper-parameters $\boldsymbol{\theta}$, and treats unknown hyper-parameters and parameters as random variables [28, Chapter 3].

According to Bayes' theorem, the joint posterior distribution over the parameters $\boldsymbol{x}$ and the hyper-parameter $\boldsymbol{\theta}$ is given as

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

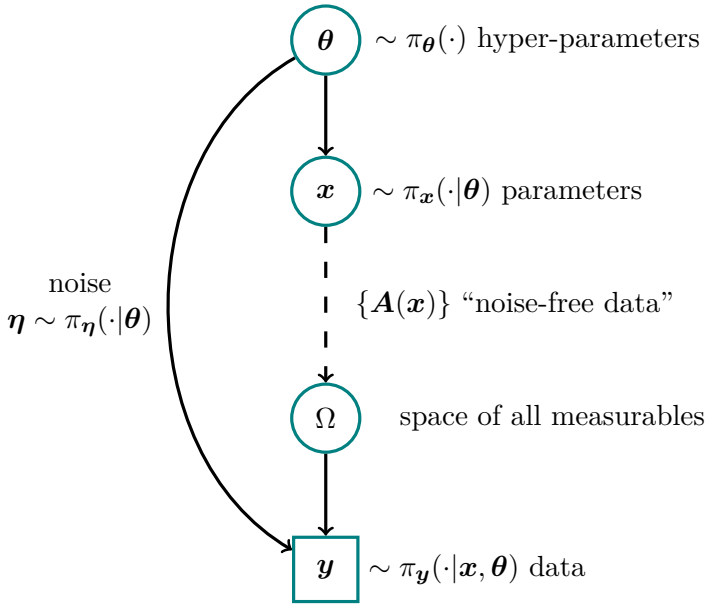


Figure 2.1: A directed acyclic graph (DAG) for an inverse problem visualises statistical dependencies as solid line arrows and deterministic dependencies as dotted arrows. The hyper-parameters $\boldsymbol{\theta}$ are distributed as ($\sim$) the hyper-prior distribution $\pi(\boldsymbol{\theta})$. The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

with finite and non-zero $\pi(\boldsymbol{y})$. The likelihood function $\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})$ is defined by the nature of the noise and the noise-free data $\boldsymbol{A}(\boldsymbol{x})$, which we read as the distribution over $\boldsymbol{y}$ conditioned on $\boldsymbol{x}$ and $\boldsymbol{\theta}$. Here $\boldsymbol{\theta}$ may account for multiple hyper-parameters, e.g. modelling the noise vector $\boldsymbol{\eta} \sim \pi_{\boldsymbol{\eta}}(\cdot|\boldsymbol{\theta})$, where $\sim$ reads as "is distributed as", and describing physical properties or functional dependencies of $\boldsymbol{x}$ such as the smoothness of $\boldsymbol{x}$. Because unknown parameter are treated as random variables the joint prior distribution is introduced as $\pi(\boldsymbol{x}, \boldsymbol{\theta}) = \pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})$ with the parameter prior distribution $\pi(\boldsymbol{x}|\boldsymbol{\theta})$ and the hyper-prior distribution $\pi(\boldsymbol{\theta})$. Choosing these prior distributions is ultimately a modeller's choice and is crucial, as those shall be as uninformative as possible for regions in hyper-parameter and parameter space where the data is informative. If the data is uninformative, the prior distributions can be informative and may represent a rather restrictive range of (physically) feasible hyper-parameters and parameters.

Figure 2.1 visualises the conditional dependencies between hyper-parameters and parameters as well as how distributions progress through to an observation (square box) using a directed acyclic graph (DAG). We plot statistical dependencies as solid arrows and deterministic dependencies as dotted arrows.

Usually, the objective is to calculate the expectation of a function $h(\boldsymbol{x})$, which is defined as

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

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

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

with an the unbiased sample-based Monte-Carlo estimate [43] for large enough N (law of large numbers [35, Chapter 17]). Here the posterior samples $\{\boldsymbol{x}^{(k)},\boldsymbol{\theta}^{(k)}\} \sim \pi_{\boldsymbol{x},\boldsymbol{\theta}}(\cdot|\boldsymbol{y})$, for $k = 1,\ldots,N$, form a sample set $\mathcal{M} = \{(\boldsymbol{x},\boldsymbol{\theta})^{(1)},\ldots,(\boldsymbol{x},\boldsymbol{\theta})^{(N)}\}$. The central limit theorem states that the sample mean $\bar{h}_N^{(i)}$ of independent sample sets $\mathcal{M}^{(i)} \sim \pi(\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y})$, for $i = 1,\ldots,n$ from a distribution, converges to be normally distributed, so that

$$\sqrt{n}(\bar{h}_N^{(i)} - \bar{h}) \xrightarrow{\mathcal{D}} \mathcal{N}(0,\sigma^2)[22], \tag{2.5}$$

and if $\sigma^2 < \infty$ the Monte-Carlo error $\bar{h}_N^{(i)} - \bar{h}$ is bounded. In practice, the Monte-Carlo error from a sample set $\mathcal{M}^{(i)}$ is approximated as

$$(\sigma^{(i)})^2 = \mathrm{Var}(\mathrm{E}_{\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y}}[h(\boldsymbol{x})]) \approx \frac{\mathrm{Var}(h(\boldsymbol{x}))}{N}\underbrace{\left(1 + 2\sum_{t=1}^{W}\frac{\Gamma(t)}{\Gamma(0)}\right)}_{:=\tau_{\mathrm{int}}} = \mathrm{Var}(h(\boldsymbol{x}))\frac{\tau_{\mathrm{int}}}{N}\,, \tag{2.6}$$

where we have to take into account that the samples generated by any system or algorithm are correlated. We define the integrated autocorrelation time (IACT) τ_{int} as in [17], which is twice the value of the IACT in [67, pp. 103-105] and [65, 27]. Here the autocorrelation coefficient $\Gamma(t) \propto \exp\{-|t|/\tau\} \longrightarrow 0$ for $t \to \infty$ at lag t decays exponentially and $\Gamma(0) = \mathrm{Var}(h(\boldsymbol{x}))$. Choosing the summation window W is crucial because it has to be large compared to the decay time τ, but for too large t the autocorrelation coefficient $\Gamma(t)$ is noise-dominated. U. Wolff [65] (and the Python implementation by D. Hesse [27]) provide a way to not only calculate the IACT safely but also to quantify the errors of the estimated IACT.

The IACT provides a good estimate of the number of steps the sampling algorithm needs to take to produce one independent sample. According to the IACT, we define the effective sample size as τ_{int}/N. We point out that for uncorrelated samples $\tau_{\mathrm{int}} = 1$ the error $(\sigma^{(i)})^2$ is a typical Monte-Carlo estimate. See Appendix A.2 and [55, 65, 67] for a more detailed derivation.

2.1.1 Marginal and then Conditional Method

Quickly generating a representative sample set 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 [47] and illustrated in Appendix A.1. Depending on the problem and the available model it is beneficial to factorise the joint posterior distribution

$$\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) = \pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})\, \pi(\boldsymbol{\theta}|\boldsymbol{y}) \tag{2.7}$$

into the full conditional posterior $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ over the latent field $\boldsymbol{x}$ and the marginal posterior $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ over hyper-parameter $\boldsymbol{\theta}$. This approach, known as the MTC method, is particularly advantageous when $\boldsymbol{x} \in \mathbb{R}^n$ is high-dimensional, while $\boldsymbol{\theta}$ is low-dimensional and the evaluation of the marginal posterior

$$\pi(\boldsymbol{\theta}|\boldsymbol{y}) = \frac{\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})\pi(\boldsymbol{y})} \propto \frac{\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})} \tag{2.8}$$

as in [17, Lemma 2] is relatively cheap.

Applying the law of total expectation [5], Eq. (2.3) becomes

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y}}[h(\boldsymbol{x})] &= \int\int h(\boldsymbol{x})\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})\ \mathrm{d}\boldsymbol{x}\, \pi(\boldsymbol{\theta}|\boldsymbol{y})\ \mathrm{d}\boldsymbol{\theta} && (2.9)\\
&= \int \mathbb{E}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}\left[h(\boldsymbol{x})\right]\, \pi(\boldsymbol{\theta}|\boldsymbol{y})\ \mathrm{d}\boldsymbol{\theta} && (2.10)\\
&= \mathbb{E}_{\boldsymbol{\theta}|\boldsymbol{y}}\left[\mathbb{E}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}[h(\boldsymbol{x})]\right]. && (2.11)
\end{aligned}$$

In the case of a linear-Gaussian hierarchical Bayesian model, both the marginal distribution $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ and the inner expectation $\mathbb{E}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}\left[h(\boldsymbol{x})\right]$ are well defined (see next subsection). If the integral in Eq. 2.10 is expensive to calculate, we use sample-based methods to produce a Markov chain $\{(\boldsymbol{x}, \boldsymbol{\theta})^{(1)}, \ldots, (\boldsymbol{x}, \boldsymbol{\theta})^{(N)}\} \sim \pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})$ and sample from $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ first and then draw samples from the full conditional posterior $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ (see Sec. 6.2.2).

Linear-Gaussian hierarchical Bayesian model

In case of normally distributed noise $\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}(\boldsymbol{\theta}))$ with zero mean and covariance $\boldsymbol{\Sigma}(\boldsymbol{\theta})$ and a linear forward model matrix $\boldsymbol{A}$ Eq. 2.1 simplifies to

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

Then we can obtain the marginal and full conditional posterior distribution explicitly. Our hierarchical linear-Gaussian Bayesian model is defined as

$$\begin{aligned}
\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta} &\sim \mathcal{N}(\boldsymbol{A}\boldsymbol{x}, \boldsymbol{\Sigma}(\boldsymbol{\theta})) && (2.13a)\\
\boldsymbol{x}|\boldsymbol{\theta} &\sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{Q}(\boldsymbol{\theta})^{-1}) && (2.13b)\\
\boldsymbol{\theta} &\sim \pi(\boldsymbol{\theta})\,, && (2.13c)
\end{aligned}$$

with a Gaussian likelihood function $\pi(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{\theta})$, a normally distributed prior $\pi(\boldsymbol{x}|\boldsymbol{\theta})$, with prior mean $\boldsymbol{\mu}$ and prior precision $\boldsymbol{Q}(\boldsymbol{\theta})$, and a hyper-prior distribution $\pi(\boldsymbol{\theta})$. For the derivation of the marginal posterior and the full conditional posterior distribution, consider the joint multivariate Gaussian distribution

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

with the joint precision matrix as in [54] (see also [47, 17]). Immediately[1], the full conditional posterior distribution can be formulated as

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

Then the marginal posterior distribution over the hyper-parameters in Eq. 2.8 is derived as

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

where, as noted by Fox and Norton [17], the parameter $\boldsymbol{x}$ cancels. Having the marginal posterior distribution $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ (independent of $\boldsymbol{x}$) available breaks up the correlation structure between $\boldsymbol{x}$ and $\boldsymbol{\theta}$ and makes the MTC approach very efficient [17] (see Appendix A.1). Within this scheme, we evaluate the marginal posterior first and then either condition on hyper-parameters to draw full conditional posterior samples $\boldsymbol{x} \sim \pi(\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y})$ (see Sec. 6.2.2) or evaluate the posterior mean

$$\mu_{\boldsymbol{x}|\boldsymbol{y}} = \int \boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}\pi(\boldsymbol{\theta}|\boldsymbol{y})\,\mathrm{d}\boldsymbol{\theta} \tag{2.17}$$

and the posterior covariance matrix

$$\Sigma_{\boldsymbol{x}|\boldsymbol{y}} = \int \boldsymbol{\Sigma}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}\pi(\boldsymbol{\theta}|\boldsymbol{y})\,\mathrm{d}\boldsymbol{\theta} \tag{2.18}$$

of $\pi(\boldsymbol{x}|\boldsymbol{y})$ by some quadrature rule.

[1] Assume $\begin{pmatrix}\boldsymbol{x}\\ \boldsymbol{y}\end{pmatrix} \sim \mathcal{N}\left[\begin{pmatrix}\boldsymbol{\mu}_{\boldsymbol{x}}\\ \boldsymbol{\mu}_{\boldsymbol{y}}\end{pmatrix}, \begin{pmatrix}\boldsymbol{Q}_{\boldsymbol{xx}} & \boldsymbol{Q}_{\boldsymbol{xy}}\\ \boldsymbol{Q}_{\boldsymbol{yx}} & \boldsymbol{Q}_{\boldsymbol{yy}}\end{pmatrix}^{-1}\right]$, then $\boldsymbol{x}|\boldsymbol{y} \sim \mathcal{N}\big(\boldsymbol{\mu}_{\boldsymbol{x}}-\boldsymbol{Q}_{\boldsymbol{xx}}^{-1}\boldsymbol{Q}_{\boldsymbol{xy}}(\boldsymbol{y}-\boldsymbol{\mu}_{\boldsymbol{y}}), \boldsymbol{Q}_{\boldsymbol{xx}}^{-1}\big)$.

2.2 Sample-Based Estimates via Markov Chain Monte-Carlo Methods

One may use Markov chain Monte-Carlo (MCMC) methods to calculate sample-based estimates of $\mathbb{E}_{\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y}}[h(\boldsymbol{x})]$ as in Eq. 2.4. Within the MTC scheme we draw samples $\{\boldsymbol{\theta}^{(1)},\dots,\boldsymbol{\theta}^{(k)},\dots,\boldsymbol{\theta}^{(N)}\} \sim \pi(\boldsymbol{\theta}|\boldsymbol{y})$ from the marginal posterior first and then characterise the full conditional posterior $\pi(\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y})$. In doing so we generate a Markov chain $\mathcal{M} = \{\boldsymbol{\theta}^{(1)},\dots,\boldsymbol{\theta}^{(k)},\dots,\boldsymbol{\theta}^{(N)}\}$, where every new sample $\boldsymbol{\theta}^{(k)}$ is proposed according to a random proposal $\boldsymbol{\theta}^{(k)} \sim q(\cdot|\boldsymbol{\theta}^{(k-1)})$ and only affected by the previous one $\boldsymbol{\theta}^{(k-1)}$. For large enough N, this chain of random variables can be used to calculate Monte-Carlo estimates, where ergodicity of the Markov chain $\mathcal{M}$ is a sufficient criterion to do so [57, 43].

The ergodicity theorem in [57] 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, the chain can reach any state from any other state (irreducibility), is not stuck in periodic cycles (aperiodicity), and satisfies the detailed balance condition [57] (reversibility). Then the samples in that chain $\mathcal{M} \sim \pi(\boldsymbol{\theta}|\boldsymbol{y})$ are samples from the desired target distribution. In practice, one can inspect the trace $\pi(\boldsymbol{\theta}^{(k)}|\boldsymbol{y})$ for $k = N_{\text{burn-in}},\dots,N$ after a "burn-in" period $N_{\text{burn-in}}$ and visually assess if the chain is consistent with ergodicity. The "burn-in" period $N_{\text{burn-in}}$ removes initialisation bias. The specific sampling methods in this thesis possess proven ergodic properties, and we therefore provide the reader with corresponding literature for further details when the methods are introduced.

If the instance $\boldsymbol{\theta}^{(k)}$ of an ergodic Markov chain represents an independent sample of the marginal posterior $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ and $\boldsymbol{x}^{(k)}$ is a sample from the full conditional posterior $\pi(\boldsymbol{x}|\boldsymbol{\theta}^{(k)},\boldsymbol{y})$, e.g. as in Sec. 6.2.2, then the resulting sample $(\boldsymbol{x}^{(k)},\boldsymbol{\theta}^{(k)})$ is an independent sample from the joint posterior $\pi(\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y})$ [17, 61]. Repeating this procedure gives the chain $\{(\boldsymbol{x},\boldsymbol{\theta})^{(1)},\dots,(\boldsymbol{x},\boldsymbol{\theta})^{(k)},\dots,(\boldsymbol{x},\boldsymbol{\theta})^{(N)}\} \sim \pi(\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y})$ of independent samples from the joint posterior.

2.3 Numerical Function Approximation – Tensor-Train (TT)

Instead of relying on sampling-based methods to explore an unnormalised density function $\pi(\boldsymbol{x})$, which in our case will be the marginal posterior distribution over the hyper-parameters, we can approximate this function using a tensor-train (TT) approximation. The TT approximation $\tilde{\pi}(\boldsymbol{x}) \approx \pi(\boldsymbol{x})$, where $\boldsymbol{x} \in \mathbb{R}^d$, on a d-dimensional grid requires far fewer function evaluations compared to conventional sampling methods. In the following, we describe how to compute a normalised marginal probability density function (PDF) $f_{X_k}(x_k)$, for an $x_k \in \boldsymbol{x}$ and $k = 1,\dots,d$, from a target function $\pi(\boldsymbol{x})$ approximated in TT format. Further, a scheme similar to the inverse Rosenblatt transform (IRT) in [11] is

introduced to generate samples from $\pi(\boldsymbol{x})$. In doing so, we follow the notation and procedure introduced in [8].

As in [8], the parameter space is defined as the product space $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_d$ with $x_k \in \mathcal{X}_k \subseteq \mathbb{R}$. The marginal PDF for the k-th component is then given by

$$f_{X_k}(x_k) = \frac{1}{z} \int_{\mathcal{X}_1} \cdots \int_{\mathcal{X}_{k-1}} \int_{\mathcal{X}_{k+1}} \cdots \int_{\mathcal{X}_d} \boldsymbol{\lambda}(\boldsymbol{x})\, \pi(\boldsymbol{x})\, \mathrm{d}x_1 \cdots \mathrm{d}x_{k-1}\, \mathrm{d}x_{k+1} \cdots \mathrm{d}x_d, \quad (2.19)$$

where we integrate over all dimensions except the k-th, and z is a normalisation constant. Here, Cui and Dolgov [8] refer to $\boldsymbol{\lambda}(x)$ as the "product-form Lebesgue-measurable weighting function", which can be useful for quadrature rules [9], and define it as

$$\boldsymbol{\lambda}(\mathcal{X}) = \prod_{i=1}^{d} \boldsymbol{\lambda}_i(\mathcal{X}_i), \quad \text{where} \quad \boldsymbol{\lambda}_i(\mathcal{X}_i) = \int_{\mathcal{X}_i} \boldsymbol{\lambda}_i(x_i)\, \mathrm{d}x_i. \quad (2.20)$$

To approximate a target function in the TT format, one has to predefine d-dimensional discrete univariate grid over the parameter space $\mathcal{X}$ with n grid points in each direction. In the TT format, the integral in Eq. 2.19 for the marginal PDF can be computed at a low computational cost, as $\pi(\boldsymbol{x})$ is approximated by

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

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 grid point $\boldsymbol{x} = (x_1, \ldots, x_d)$. A TT core $\tilde{\pi}_k \in \mathbb{R}^{r_{k-1} \times n \times r_k}$ has ranks r_{k-1} and r_k, which connect the core to its respective neighbouring dimensions. The outer ranks of a TT are $r_0 = r_d = 1$. Then, a discrete parameter space $\mathcal{X}$ is approximated by $\pi(\mathcal{X}) \approx \tilde{\pi}_1 \tilde{\pi}_2 \cdots \tilde{\pi}_d$ with $2nr + (d-2)nr^2$ evaluation points for fixed ranks $r = r_{k-1} = r_k$, as illustrated in Figure 2.2, instead of n^d function evaluations. Consequently, the marginal PDF

$$\begin{aligned} f_{X_k}(x_k) \approx \frac{1}{z} \Bigg| & \left(\int_{\mathcal{X}_1} \boldsymbol{\lambda}_1(x_1)\tilde{\pi}_1(x_1)\, \mathrm{d}x_1 \right) \cdots \left(\int_{\mathcal{X}_{k-1}} \boldsymbol{\lambda}_{k-1}(x_{k-1})\tilde{\pi}_{k-1}(x_{k-1})\, \mathrm{d}x_{k-1} \right) \\ & \boldsymbol{\lambda}_k(x_k)\tilde{\pi}_k(x_k) \\ & \left(\int_{\mathcal{X}_{k+1}} \boldsymbol{\lambda}_{k+1}(x_{k+1})\tilde{\pi}_{k+1}(x_{k+1})\, \mathrm{d}x_{k+1} \right) \cdots \left(\int_{\mathcal{X}_d} \boldsymbol{\lambda}_d(x_d)\tilde{\pi}_d(x_d)\, \mathrm{d}x_d \right) \Bigg| \end{aligned} \quad (2.21)$$

is computed by integrating over all TT cores except the k-th core π_k, as in [11], and normalised by the constant z [8].

In practice, TT approximations may suffer from numerical instability. In particular when the target function is non-negative the TT approximation can have negative values in regions where true function values are very small. One way to ensure non-negativity

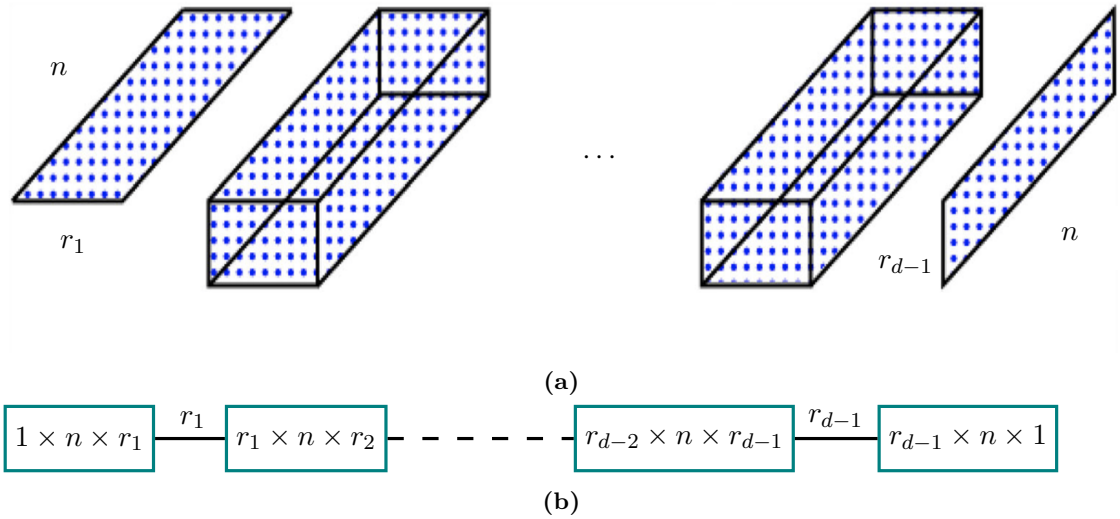


Figure 2.2: Here, we visualise the TT cores as a train of two- and three-dimensional matrices. Each core has a length n, corresponding to the number of grid points in each dimension, and the cores are connected through ranks r_k. More specifically, a core $\tilde{\pi}_k$ has dimensions $r_{k-1} \times n \times r_k$, with outer ranks $r_0 = r_d = 1$. Using the TT format enables us to represent a d-dimensional grid with only $2nr + (d-2)nr^2$ evaluation points instead of n^d grid points. Figure (a) is adapted from [20].

is to square the target function, hence [8] approximate the square root of the target function and define the approximation as [8, Eq. 18]

$$\sqrt{\pi(\boldsymbol{x})} \approx \tilde{g}(\boldsymbol{x}) = \boldsymbol{G}_1(x_1), \ldots, \boldsymbol{G}_k(x_k), \ldots, \boldsymbol{G}_d(x_d)\,. \tag{2.22}$$

Here, each TT core is given by [8, Eq. 21]

$$G_k^{(\alpha_{k-1},\alpha_k)}(x_k) = \sum_{i=1}^{n_k} \phi_k^{(i)}(x_k) \boldsymbol{A}_k[\alpha_{k-1}, i, \alpha_k], \quad k = 1, \ldots, d, \;, \tag{2.23}$$

where $\boldsymbol{A}_k \in \mathbb{R}^{r_{k-1} \times n_k \times r_k}$ is the k-th coefficient tensor and $\{\phi_k^{(i)}(x_k)\}_{i=1}^{n_k}$ are the basis functions corresponding to the k-th coordinate. The approximated unnormalised density function is written as [8, Eq. 19]:

$$\pi(\boldsymbol{x}) \approx \xi + \tilde{g}(\boldsymbol{x})^2\,, \tag{2.24}$$

to ensure a positivity a small constant $\xi > 0$ is added according to the ratio of the Lebesgue weighted L2-norm error and the Lebesgue weighting (see Eq. 2.20 and [8, Eq. 35]) such that

$$0 \leq \xi \leq \frac{1}{\lambda(\mathcal{X})} \|\tilde{g} - \sqrt{\pi}\|^2_{L^2_\lambda(\mathcal{X})}\,. \tag{2.25}$$

This leads to the normalised PDF [8, Eq. 19]

$$f_X(\boldsymbol{x}) \approx \frac{1}{z}\left(\lambda(\boldsymbol{x})\xi + \lambda(\boldsymbol{x})\tilde{g}(\boldsymbol{x})^2\right)\;, \tag{2.26}$$

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

$$\begin{aligned} f_{X_k}(x_k) \approx \frac{1}{z}\Bigg(\xi \prod_{i=1}^{k-1} \boldsymbol{\lambda}_i(\mathcal{X}_i) \prod_{i=k+1}^{d} \boldsymbol{\lambda}_i(\mathcal{X}_i) \\ + \left(\int_{\mathcal{X}_1} \boldsymbol{\lambda}_1(x_1)\boldsymbol{G}_1^2(x_1)\,\mathrm{d}x_1\right) \cdots \left(\int_{\mathcal{X}_{k-1}} \boldsymbol{\lambda}_{k-1}(x_{k-1})\boldsymbol{G}_{k-1}^2(x_{k-1})\,\mathrm{d}x_{k-1}\right) \\ \boldsymbol{\lambda}_k(x_k)\boldsymbol{G}_k^2(x_k) \\ \left(\int_{\mathcal{X}_{k+1}} \boldsymbol{\lambda}_{k+1}(x_{k+1})\boldsymbol{G}_{k+1}^2(x_{k+1})\,\mathrm{d}x_{k+1}\right) \cdots \left(\int_{\mathcal{X}_d} \boldsymbol{\lambda}_d(x_d)\boldsymbol{G}_d^2(x_d)\,\mathrm{d}x_d\right)\Bigg). \end{aligned} \tag{2.27}$$

2.3.1 Marginal Functions

The marginal functions $f_{X_k}(x_k)$ of the PDF $f_X(\boldsymbol{x})$ are computed by a procedure to which Cui and Dolgov [8] refer to as backward marginalisation, see Prop. 2, and to which we add the forward marginalisation, see Prop. 1. This is similar to the left and right orthogonalisation of TT cores [37, 36]. The backward marginalisation provides the coefficient matrices $\boldsymbol{B}_k$, while the forward marginalisation gives the coefficient matrices $\boldsymbol{R}_{\mathrm{pre},k}$. These matrices enable the efficient evaluation of marginal functions since they are formed by integration over the parameter space either left or right of the k-th dimension, as in [8]. In doing so, the mass matrix $\boldsymbol{M}_k \in \mathbb{R}^{n_k \times n_k}$ is defined as in [8, Eq. 22]

$$\boldsymbol{M}_k[i,j] = \int_{\mathcal{X}_k} \phi_k^{(i)}(x_k)\phi_k^{(j)}(x_k)\boldsymbol{\lambda}(x_k)\,\mathrm{d}x_k, \quad i,j = 1,\ldots,n_k, \ , \tag{2.28}$$

where $\{\phi_k^{(i)}(x_k)\}_{i=1}^{n_k}$ denotes the set of basis functions for the k-th coordinate. The proposition used to compute $\boldsymbol{B}_k$, stated in Prop. 1, is adapted directly from [8].

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

$$f_{X_k}(x_k) \approx \frac{1}{z}\left(\xi \prod_{i=1}^{k-1} \boldsymbol{\lambda}_i(X_i) \prod_{i=k+1}^{d} \boldsymbol{\lambda}_i(X_i) + \sum_{l_{k-1}=1}^{r_{k-1}} \sum_{l_k=1}^{r_k} \left(\sum_{i=1}^{n} \phi_k^{(i)}(x_k)\boldsymbol{D}_k[l_{k-1},i,l_k]\right)^2\right)\boldsymbol{\lambda}_k(x_k), \tag{2.29}$$

where $\boldsymbol{D}_k \in \mathbb{R}^{r_{k-1}\times n \times r_k}$ and given as

$$\boldsymbol{D}_k[l_{k-1},i,l_k] = \sum_{\alpha_{k-1}=1}^{r_{k-1}} \boldsymbol{R}_{\mathrm{pre},k-1}[l_{k-1},\alpha_{k-1}]\boldsymbol{B}_k[\alpha_{k-1},i,l_k]\,, \tag{2.30}$$

with $\boldsymbol{R}_{\mathrm{pre},k-1} \in \mathbb{R}^{r_{k-1}\times r_{k-1}}$ and $\boldsymbol{B}_k \in \mathbb{R}^{r_{k-1}\times n \times r_k}$.

For the first dimension, $f_{X_1}(x_1)$ can be expressed as [8, Eq. 30]

$$f_{X_1}(x_1) \approx \frac{1}{z}\left(\xi \prod_{i=2}^{d} \boldsymbol{\lambda}_i(\mathcal{X}_i) + \sum_{l_1=1}^{r_1} \left(\sum_{i=1}^{n} \phi_1^{(i)}(x_1)\boldsymbol{D}_1[i,l_1]\right)^2\right)\boldsymbol{\lambda}_1(x_1)\ , \tag{2.31}$$

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

$$f_{X_d}(x_d) \approx \frac{1}{z}\left(\xi \prod_{i=1}^{d-1} \boldsymbol{\lambda}_i(\mathcal{X}_i) + \sum_{l_{d-1}=1}^{r_{d-1}} \left(\sum_{i=1}^{n} \phi_1^{(i)}(x_1)\boldsymbol{D}_d[l_{d-1}, i]\right)^2\right)\boldsymbol{\lambda}_d(x_d), \tag{2.32}$$

where $\boldsymbol{D}_d[l_{d-1}, i] = \boldsymbol{B}_{\mathrm{pre},d}[l_{d-1}, i, \alpha_{d+1}]$ and $\alpha_{d+1} = 1$. Note that in practice we calculate

z numerically within the process of computing the marginal PDFs so that $\sum f_{X_k}(x_k) = 1$

and for Cartesian basis $\boldsymbol{M}_k = \mathrm{diag}(\boldsymbol{\lambda}_k(\mathcal{X}_k))$ with $\boldsymbol{\lambda}(x) = 1$.

Proposition 1 (Backward Marginalisation as in [8])**:** Starting with the last coordinate $k = d$, we set $\boldsymbol{B}_d = \boldsymbol{A}_d$. The following procedure can be used to obtain the coefficient tensor $\boldsymbol{B}_k \in \mathbb{R}^{r_{k-1}\times n_k \times r_k}$, which is needed for defining the marginal function $f_{X_k}(x_k)$ or to draw samples from $\tilde{\pi}(\boldsymbol{x})$ via the squared IRT scheme (see Alg. Box 1):

1. Use the Cholesky decomposition of the mass matrix, $\boldsymbol{L}_k\boldsymbol{L}_k^\top = \boldsymbol{M}_k \in \mathbb{R}^{n_k\times n_k}$, to construct a tensor $\boldsymbol{C}_k \in \mathbb{R}^{r_{k-1}\times n_k\times r_k}$ [8, Eq. 27]:

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

2. Unfold $\boldsymbol{C}_k$ along the first coordinate and compute the thin QR decomposition, so that $\boldsymbol{C}_k^{(R)} \in \mathbb{R}^{r_{k-1}\times(n_k r_k)}$ [8, Eq. 28]:

$$\boldsymbol{Q}_k\boldsymbol{R}_k = \left(\boldsymbol{C}_k^{(R)}\right)^\top . \tag{2.34}$$

3. Compute the new coefficient tensor [8, Eq. 29]:

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

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

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

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

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

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

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

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

2.3.2 Sampling from a TT Approximation

Instead of evaluating marginal functions for quadrature, the inverse Rosenblatt transform (IRT) provides a scheme to draw samples from an approximated function in the TT format [11]. The idea is that a target function can be represented as the sequence $f_X(\boldsymbol{x}) = f_{X_1}(x_1) f_{X_2|X_1}(x_2|x_1) \cdots f_{X_k|X_{<k}}(x_k|x_{k-1}, \ldots, x_1)$. Within the IRT scheme samples are iteratively drawn from each $f_{X_k|X_{<k}}(x_k|x_{k-1}, \ldots, x_1)$ conditioned on the previous left $k-1$ samples and marginalised over the right $k+1$ dimensions, for $k = 2, \ldots, d-1$. Since the square root of the target function is approximated, Cui and Dolgov [8] call that the squared inverse Rosenblatt transform (SIRT).

Algorithm 1: Squared Inverse Rosenblatt Transform (SIRT)

Input: seeds $\{\boldsymbol{u}^{(1)}, \ldots, \boldsymbol{u}^{(N)}\} \sim \mathcal{U}(0,1)^d$ and $\boldsymbol{B}_1, \ldots, \boldsymbol{B}_d$ from Prop. 1
for $s = 1, \ldots, N$ **do**
  for $k = 1, \ldots, d$ **do**
    compute normalised PDF $f_{X_k|X_{<k}}(x_k|x_{k-1}^{(s)}, \ldots, x_1^{(s)})$, Eq. 2.40
    compute cumulative distribution function $F_{X_k|X_{<k}}(x_k)$, Eq. 2.39,
    project sample $x_k^{(s)} = F^{-1}_{X_k|X_{<k}}(u_k^{(s)})$
    interpolate $\boldsymbol{G}_k(x_k^{(s)})$, Eq. 2.41
    update $\boldsymbol{G}_{\leq k}(x_{\leq k}^{(s)}) = \boldsymbol{G}_{<k}(x_{<k}^{(s)}) \boldsymbol{G}_k(x_k^{(s)})$
  end for
end for
Output: samples $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}$, where each $\boldsymbol{x}^{(s)} \in \mathbb{R}^d$ for $s = 1, \ldots, N$

Given the backward marginal coefficient tensors $\boldsymbol{B}_1, \ldots, \boldsymbol{B}_d$ as in Prop. 1 and N uniformly distributed seeds $\{\boldsymbol{u}^{(1)}, \ldots, \boldsymbol{u}^{(N)}\} \sim \mathcal{U}(0,1)^d$, where each $\boldsymbol{u}^{(s)}$ is d-dimensional for $s = 1, \ldots, N$, the first marginal $f_{X_1}(x_1)$ is calculated as in Eq. 2.31 and normalised with $z = \int_{\mathcal{X}_1} f_{X_1}(x_1)dx_1$. Next, the cumulative distribution function (CDF) $F_{X_1}(x_k) = \int_{-\infty}^{x_k} f_{X_1}(\hat{x}_1)d\hat{x}_1$ is formed, which for the general case is given as [8, Eq. 17]:

$$F_{X_k|X_{<k}}(x_k) = \int_{-\infty}^{x_k} f_{X_k|X_{<k}}(\hat{x}_k|x_{k-1}, \ldots, x_1)\,\mathrm{d}\hat{x}_k \;; \tag{2.39}$$

Then the seed $u_k^{(s)}$ is projected onto the parameter space to generate the sample $x_k^{(s)} = F^{-1}_{X_k|X_{<k}}(u_k^{(s)})$. For $k = 2, \ldots, d$ the general "conditional marginal" is given as [8, Eq. 31]:

$$\begin{aligned} f_{X_k|X_{<k}}(x_k|x_{k-1}^{(s)}, \ldots, x_1^{(s)}) \approx \frac{1}{z}\Bigg(\xi \prod_{i=k+1}^{d} \boldsymbol{\lambda}_i(X_i) + \\ \sum_{l_k=1}^{r_k}\Bigg(\sum_{i=1}^{n} \phi_k^{(i)}(x_k^{(s)})\Big(\sum_{\alpha_{k-1}=1}^{r_{k-1}} \boldsymbol{G}_{<k}^{(\alpha_{k-1})}(x_{<k}^{(s)})\boldsymbol{B}_k[\alpha_{k-1}, i, l_k]\Big)\Bigg)^2\Bigg)\boldsymbol{\lambda}_k(x_k)\;, \end{aligned} \tag{2.40}$$

where we marginalise over the dimensions $k+1, \ldots, d$ via $\boldsymbol{B}_k$ and condition on the previous $k-1$ samples through the product $\boldsymbol{G}_k(x_k^{(s)}) \in \mathbb{R}^{1 \times r_{k-1}}$. Function values between grid points i and $i+1$ are approximated with a piecewise polynomial interpolation

$$\boldsymbol{G}_k(x_k^{(s)}) \approx \frac{x_k^{(s)} - x_k^{(i)}}{x_k^{(i+1)} - x_k^{(i)}}\boldsymbol{G}_k(x_k^{(i+1)}) + \frac{x_k^{(i+1)} - x_k^{(s)}}{x_k^{(i+1)} - x_k^{(i)}}\boldsymbol{G}_k(x_k^{(i)})\,, \tag{2.41}$$

for $x_k^{(i)} \leq x_k^{(s)} \leq x_k^{(i+1)}$ as in [11] for the next "conditional marginal".

The procedure is repeated for each $u_k^{(s)} \in \boldsymbol{u}^{(s)}$ to produce the samples $\boldsymbol{x}^{(s)} \sim f_X(\boldsymbol{x})$, as summarised in Alg. Box 1.

Metropolis–Hastings – correction step

Since the samples by the SIRT scheme are generated from an approximation, it is sensible to correct those using a Metropolis–Hastings (MH) importance step as in [11]. In doing so, we compute the acceptance probability $\alpha = \min(w^{(s+1)}/w^{(s)}, 1)$, where

$$w(x) = \frac{\pi(\boldsymbol{x})}{f_X(\boldsymbol{x})} = \frac{\pi(\boldsymbol{x})}{\xi + \tilde{g}(\boldsymbol{x})^2} \tag{2.42}$$

is the importance ratio. Note the normalising constants in the ratio $w^{(s+1)}/w^{(s)}$ cancel. In practice, the importance ratio is calculated in the log-space so that $\log f_X(\boldsymbol{x}) = \log f_{X_1}(x_1) + \log f_{X_2|X_1}(x_2|x_1) + \cdots + \log f_{X_k|X_{<k}}(x_k|x_{k-1}, \ldots, x_1)$ (see Eq. 2.40). We refer to this as the SIRT-MH scheme, which provides the corrected chain $\{\boldsymbol{x}_{\mathrm{MH}}^{(1)}, \ldots, \boldsymbol{x}_{\mathrm{MH}}^{(N)}\} \sim \pi(\boldsymbol{x})$.

Algorithm 2: MH correction step

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

2.3.3 Error of the TT Approximation

A straightforward way to assess an average error of a TT approximation is to calculate the relative root mean squared (RMS) error

$$\left(\frac{\int_{\mathcal{X}} (\pi(\boldsymbol{x}) - (\xi + \tilde{g}(\boldsymbol{x})^2))^2 \boldsymbol{\lambda}(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x}}{\int_{\mathcal{X}} \pi(\boldsymbol{x})^2 \boldsymbol{\lambda}(x) \, \mathrm{d}\boldsymbol{x}} \right)^{1/2} = \frac{\|\pi(\boldsymbol{x}) - (\xi + \tilde{g}(\boldsymbol{x})^2)\|_{L^2_{\boldsymbol{\lambda}}(\mathcal{X})}}{\|\pi(\boldsymbol{x})\|_{L^2_{\boldsymbol{\lambda}}(\mathcal{X})}} \,. \tag{2.43}$$

The RMS is approximated by

$$\left(\frac{1}{N} \sum_{i=1}^{N} \left(\pi(\boldsymbol{x}^{(i)}) - (\xi + \tilde{g}(\boldsymbol{x}^{(i)})^2) \right)^2 \boldsymbol{\lambda}(\boldsymbol{x}^{(i)}) \right)^{1/2} \approx \left(\int_{\mathcal{X}} (\pi(\boldsymbol{x}) - (\xi + \tilde{g}(\boldsymbol{x})^2) \right)^2 \boldsymbol{\lambda}(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \Bigg)^{1/2} \tag{2.44}$$

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

Absolute error bound

If large errors occur in regions with low probability, the RMS is sensitive to those, whereas the Wasserstein distance weighs differences according to their respective probability values.

The Wasserstein distance is the infimum over all couplings between two probability distributions with respect to some distance measure. The Kantorovich-Rubinstein duality, as in [58, 1], says that the 1-Wasserstein distance is equal to the supremum of differences in expectations over all 1-Lipschitz functions h between two probability distributions. So the 1-Wasserstein distance provides an upper absolute error bound and is defined as

$$W_1(\pi, \tilde{\pi}) = \inf_{\nu \in \Pi(\pi, \tilde{\pi})} \int_{\mathcal{X} \times \mathcal{X}} c_{\mathcal{X}}(\boldsymbol{x}, \tilde{\boldsymbol{x}}) \, \nu(\boldsymbol{x}, \tilde{\boldsymbol{x}}) \, \mathrm{d}\boldsymbol{x} \, \mathrm{d}\tilde{\boldsymbol{x}} \,, \tag{2.45}$$

where ν couples $\boldsymbol{x}$ and $\tilde{\boldsymbol{x}}$ so that the integral over the distance $c_{\mathcal{X}}(\boldsymbol{x}, \tilde{\boldsymbol{x}})$ weighted by the probability measures π and $\tilde{\pi}$ is the greatest lower bound of all integrals with

respect to ν in the set of all couplings $\Pi(\pi, \tilde{\pi})$. Often ν is the transport plan, where $c_{\mathcal{X}}(\boldsymbol{x}, \tilde{\boldsymbol{x}})$ is the (ground) cost function, and $\nu(\boldsymbol{x}, \tilde{\boldsymbol{x}})$ is related to the mass which has to be transported and the 1-Wasserstein distance is the earth mover distance. On the other hand (Kantorovich-Rubinstein duality), the 1-Wasserstein distance

$$W_1(\pi, \tilde{\pi}) = \sup_{h(\boldsymbol{x});\, c_{\mathcal{Y}}(h(\boldsymbol{x}), h(\tilde{\boldsymbol{x}})) \leq c_{\mathcal{X}}(\boldsymbol{x}, \tilde{\boldsymbol{x}})} \left\{ \int_{\mathcal{X}} h(\boldsymbol{x}) \, \mathrm{d}\pi(\boldsymbol{x}) - \int_{\mathcal{X}} h(\tilde{\boldsymbol{x}}) \, \mathrm{d}\tilde{\pi}(\tilde{\boldsymbol{x}}) \right\} \tag{2.46}$$

$$= \sup_{h(\boldsymbol{x});\, c_{\mathcal{Y}}(h(\boldsymbol{x}), h(\tilde{\boldsymbol{x}})) \leq c_{\mathcal{X}}(\boldsymbol{x}, \tilde{\boldsymbol{x}})} \left\{ \mathop{\mathbb{E}}_{\boldsymbol{x} \sim \pi} \big[h(\boldsymbol{x}) \big] - \mathop{\mathbb{E}}_{\tilde{\boldsymbol{x}} \sim \tilde{\pi}} \big[h(\tilde{\boldsymbol{x}}) \big] \right\}. \tag{2.47}$$

is the lowest upper bound of differences in expectations over all 1-Lipschitz function $h(\boldsymbol{x}) : \mathcal{X} \to \mathcal{Y}$ in between the two distributions π and $\tilde{\pi}$, with the distance measure $c_{\mathcal{X}}$ on the set $\mathcal{X}$ forming the metric space $(\mathcal{X}, c_{\mathcal{X}})$ and similarly the metric space $(\mathcal{Y}, c_{\mathcal{Y}})$.

For two sample sets $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\} \sim \pi$ and $\{\tilde{\boldsymbol{x}}^{(1)}, \ldots, \tilde{\boldsymbol{x}}^{(M)}\} \sim \tilde{\pi}$ the calculation of the Wasserstein distance becomes an optimisation problem that is to find the best coupling of samples weighted by their distribution value according to an appropriate distance measure [15], which we set to $c_{\mathcal{X}}(\boldsymbol{x}, \tilde{\boldsymbol{x}}) = \|\boldsymbol{x} - \tilde{\boldsymbol{x}}\|_{L^2}$. More specifically,

$$W_1(\pi, \tilde{\pi}) = \min_{\nu \in \Pi(\pi, \tilde{\pi})} \sum_{j=1}^{M} \sum_{i=1}^{N} \nu_{ij} \|\boldsymbol{x}^{(i)} - \tilde{\boldsymbol{x}}^{(j)}\|_{L^2} \,, \tag{2.48}$$

where the transport plan $\nu \in \mathbb{R}_{\geq 0}^{N \times M}$ defines the coupling $\nu_{ij} \in \nu$ as $\nu_{ij} \coloneqq \pi(\boldsymbol{x}^{(i)}) \tilde{\pi}(\tilde{\boldsymbol{x}}^{(j)})$ similar to [15, Eq. 3.166]. Additionally it is required that $\sum_{i=1}^{N} \pi(\boldsymbol{x}^{(i)}) = \sum_{j=1}^{M} \tilde{\pi}(\tilde{\boldsymbol{x}}^{(j)}) = 1$. This gives us an upper bound of the absolute error between the expected value of any 1-Lipschitz function h.

2.4 Regularisation Approach

The currently most used method to analyse data in atmospheric physics is regularisation-based. Since we want to show that Bayesian methods provide more information than regularisation at a similar computational cost, the chosen regularisation approach is the closest equivalent to the linear-Gaussian Bayesian framework [17] in Sec. 4.1.

For a linear forward model matrix $\boldsymbol{A}$, data $\boldsymbol{y}$ and a regularisation operator $\boldsymbol{T}$, the regularisation approach provides one solution $\boldsymbol{x}$ that minimises both the data misfit norm

$$\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_{L^2} \tag{2.49}$$

and a regularisation norm

$$\|\boldsymbol{T}\boldsymbol{x}\|_{L^2} \,. \tag{2.50}$$

For a fixed regularisation parameter $\lambda > 0$, the regularised solution as in [24, 17, 57] is given by $\boldsymbol{x}_\lambda$ that minimises the weighted sum

$$\boldsymbol{x}_\lambda = \arg\min_{\boldsymbol{x}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_{L^2}^2 + \lambda \|\boldsymbol{T}\boldsymbol{x}\|_{L^2}^2 \,, \tag{2.51}$$

which can be calculated by taking the derivative with respect to $\boldsymbol{x}$:

$$\nabla_{\boldsymbol{x}} \left\{ (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x})^T (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}) + \lambda \boldsymbol{x}^T \boldsymbol{T}^T \boldsymbol{T} \boldsymbol{x} \right\} = 0 \tag{2.52}$$

$$\iff \quad \nabla_{\boldsymbol{x}} \left\{ \boldsymbol{y}^T \boldsymbol{y} + \boldsymbol{x}^T \boldsymbol{A}^T \boldsymbol{A} \boldsymbol{x} - 2\boldsymbol{y}^T \boldsymbol{A} \boldsymbol{x} + \lambda \boldsymbol{x}^T \boldsymbol{T}^T \boldsymbol{T} \boldsymbol{x} \right\} = 0 \tag{2.53}$$

$$\iff \quad 2\boldsymbol{A}^T \boldsymbol{A} \boldsymbol{x} - 2\boldsymbol{A}^T \boldsymbol{y} + 2\lambda \boldsymbol{T}^T \boldsymbol{T} \boldsymbol{x} = 0. \tag{2.54}$$

Eq. 2.54 yields the regularised solution

$$\boldsymbol{x}_\lambda = (\boldsymbol{A}^T \boldsymbol{A} + \lambda \boldsymbol{L})^{-1} \boldsymbol{A}^T \boldsymbol{y} \,, \tag{2.55}$$

where we define $\boldsymbol{L} \coloneqq \boldsymbol{T}^T \boldsymbol{T}$. Typically, $\boldsymbol{L}$ represents a discrete matrix approximation of a differential operator [57]. For example

$$\boldsymbol{T} = \frac{1}{h} \begin{bmatrix} -1 & 1 & & & \\ 0 & -1 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 0 & -1 & 1 \\ & & & 0 & -1 \end{bmatrix} , \tag{2.56}$$

is the first order forward difference operator with equal spacing h as in [57] that approximates the first derivative. Then

$$\boldsymbol{T}^T \boldsymbol{T} = \frac{1}{h^2} \begin{bmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{bmatrix} , \tag{2.57}$$

is a discrete approximation to the second derivative with Neumann boundary conditions [62].

If λ is large, then the effect of the data on the solution $\boldsymbol{x}_\lambda$ is small and dominated by the regulariser, resulting in an under-fitted $\boldsymbol{x}_\lambda$. For example, if the regulariser imposes smoothness, the solutions will be overly smooth and not sensitive to structures from the data. If λ is small, the solution $\boldsymbol{x}_\lambda$ will be dominated by the data misfit norm. Then $\boldsymbol{x}_\lambda$ is sensitive to noise, resulting in an over-fitted solution inheriting the structure of the noise. We refer to [25] and [57] for a more comprehensive analysis on

the effects of the regularisation parameter on the solution, e.g. due to small singular values of the forward model.

In practice, $\boldsymbol{x}_\lambda$ is computed for a range of λ-values and the data misfit norm versus the regularisation norm is plotted in log-space to form an L-curve (see Fig. 4.8). Based on the trade-off between the data misfit and the regularisation norm, the regularisation solution corresponds to the point of maximum curvature of the L-curve [26].

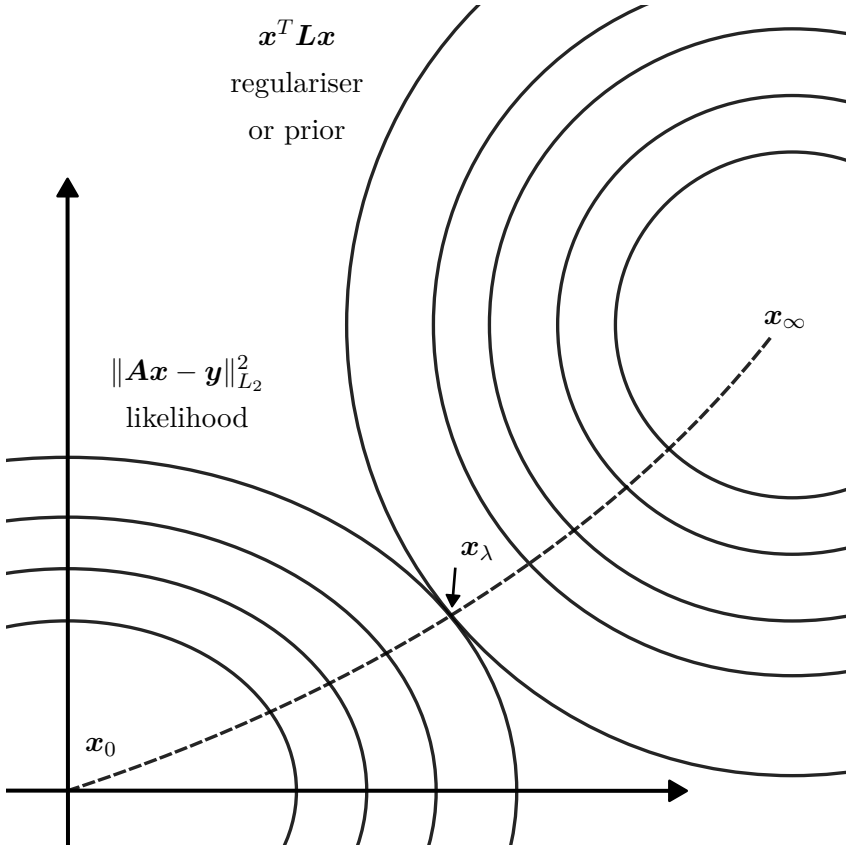


Figure 2.3: This Figure is not to scale and is directly inspired by [19]. One solution $\boldsymbol{x}_\lambda$ is obtained by following a contour line $\|\boldsymbol{A}\boldsymbol{x}-\boldsymbol{y}\|^2_{L^2} = c$ until $\boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x}$ is minimised. In the centre of the likelihood contours a solution $\boldsymbol{x}_0$ uneffected by the regulariser is obtained, whereas the solution $\boldsymbol{x}_\infty$ is determined a-priori.

Alternatively one can introduce a Lagrangian $\mathcal{L}(\boldsymbol{x},\lambda) \coloneqq \|\boldsymbol{A}\boldsymbol{x}-\boldsymbol{y}\|^2_{L^2} + \lambda\,\boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x}$ similar to [31], where λ is a Lagrange multiplier. For a given λ, a solution $\boldsymbol{x}_\lambda$ that minimises $\mathcal{L}(\boldsymbol{x},\lambda)$ is usually obtained by finding the minimum of $\|\boldsymbol{A}\boldsymbol{x}-\boldsymbol{y}\|^2_{L^2}$ with respect to the constant constrained $\boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x} = c$ [49, Fig. 2.13]. This is equivalent to $\boldsymbol{x}_\lambda = \arg\min_{\boldsymbol{x}} \boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x}$ subject to a constant constrain $\|\boldsymbol{A}\boldsymbol{x}-\boldsymbol{y}\|^2_{L^2} = c$ (see [17, fn. 6]). So every solution $\boldsymbol{x}_\lambda$ is extremely regularised for a given data misfit and the L-Curve presents the lower boundary to an open set of $\boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x}$ values. Hence, almost every sample of the posterior, which represents a feasible solution given the data, lies above the L-Curve and is less regularised and because it has a higher $\boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x}$ value.

3

The Forward Model

In this Chapter, we present the forward model to which we apply our entire methodology and conduct a singular value analysis for different measurement scenarios to understand the forward model and to determine a sensible way to measure ozone. We follow the Michelson interferometer for passive atmospheric sounding (MIPAS) handbook [41] and simulate data according to an idealised cloud-free atmosphere in local thermodynamic equilibrium, assuming a measurement instrument with infinite spectral resolution and no pointing errors. This is a simplified forward model, and we do not include any other instrument-specific details, such as sensor area or antenna response, as they are not available to us.

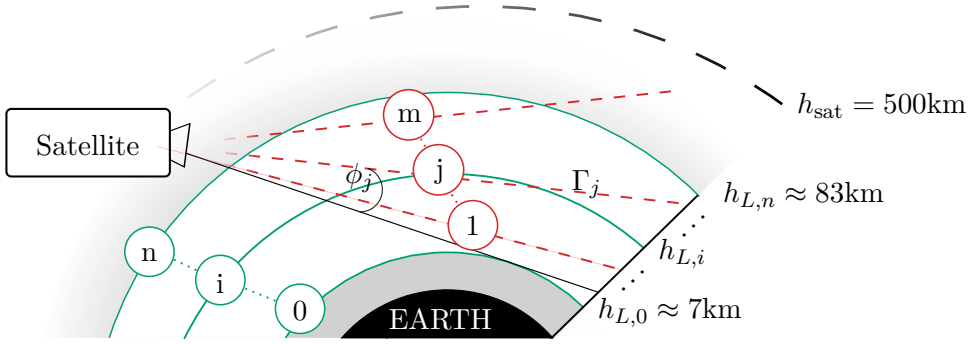


Figure 3.1: Schematic of measurement and analysis geometry, not to scale. The stationary satellite, at a constant height h_{sat} above Earth, takes m measurements along its line-of-sight defined by the line Γ_j. Each measurement has a pointing angle ϕ_j and a tangent height h_{ℓ_j}, $j = 1, 2, \ldots, m$ defined as the closest distance of Γ_j to the Earth's surface. Between $h_{L,0} \approx 7\text{km}$ and $h_{L,n} \approx 83\text{km}$, the atmosphere is discretised into n layers as illustrated by the solid green lines.

3.1 Radiative Transfer Equation

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

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

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

which is the radiative transfer equation (RTE) [41]. For more information on the processes within the atmosphere for ozone, we refer to [30]. We define a tangent height h_{ℓ_j} and Γ_j for each $j = 1, 2, \ldots, m$, so that the data vector $\boldsymbol{y} \in \mathbb{R}^m$ including some additive noise η_j. Additionally, the pointing angle $0 \leq \phi_j < \phi_{\max}$ is defined so that if $\phi = 0$arc sec the satellite points at $h_{L,0}$ and for a pointing angle $\phi_{\max}$ at $h_{L,n}$. Within the atmosphere, the number density $p(T)/(k_{\text{B}} T(r))$ 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

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

is dependent on the line intensity $L(\nu, T_{\text{ref}})$ at reference temperature $T_{\text{ref}} = 296K$, the lower-state energy E'' in cm^{-1} of the targeted transition and the second radiation constant $c_2 \coloneqq hc/k_{\text{B}} \approx 1.44\text{cmK}$ as in the HITRAN database [23], with Planck's constant h and speed of light c. Since we assume that the measurement device has a negligible frequency window, we neglect line broadening around ν for the calculations of $L(\nu, T_{\text{ref}})$, which would normally be modelled as a convolution of the normalised Lorentz profile (collisional/pressure broadening) and the normalised Doppler (thermal broadening) profile [41]. Additionally, we target one specific molecule and calculate $k(\nu, T)$ accordingly, which usually would involve summing the individual absorption constants for multiple radiating molecules weighted by their respective VMR [41]. The total internal partition function is given as:

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

with the statistical weight g'' for the lower and g' for the upper energy state (also called the degeneracy factors) accounting for the molecule's non-rotational and rotational energy

states (see [53]), and the upper state energy $E' = E'' + \nu$. Under the assumption of local thermodynamic equilibrium (LTE), the black body radiation acts as a source function

$$B(\nu, T) = \frac{2hc^2\nu^3}{\exp\{\frac{c_2\nu}{T}\} - 1}\,. \tag{3.5}$$

For fundamentals on the RTE, we recommend [48, Chapter 1], and for a more comprehensive model, we refer to [40].

3.2 Simulate Data Based on a Ground Truth

To calculate the integrals in Eq. 3.1 and Eq. 3.2 numerically, we discretise the atmosphere in n layers and define height values $h_{L,i-1} < h_{L,i}$ with respect to the surface of the earth, for $i = 1, \ldots, n$. The i-th layer is defined by two spheres around the centre of the earth with radii $r_0 + h_{L,i-1}$ and $r_0 + h_{L,i}$. Then the ozone VMR $\boldsymbol{x} = \{x_1, x_2, \ldots, x_n\} \in \mathbb{R}^n$, pressure $\boldsymbol{p} = \{p_1, p_2, \ldots, p_n\} \in \mathbb{R}^n$ and temperature $\boldsymbol{T} = \{T_1, T_2, \ldots, T_n\} \in \mathbb{R}^n$, as well as all other height dependent parameters, are discretised profiles with constant values between the heights $h_{L,i-1} \leq h < h_{L,i}$. Above $h_{L,n}$ and below $h_{L,0}$, the ozone VMR is set to zero, so no signal can be obtained. We evaluate the integral in Eq. (3.1) for one noise-free measurement $A_j(\boldsymbol{p}, \boldsymbol{T}, \boldsymbol{x})$, using the trapezoidal rule. Here, each entry A_j of $\boldsymbol{A}(\boldsymbol{p}, \boldsymbol{T}, \boldsymbol{x}) \in \mathbb{R}^m$ includes multiple evaluations of the integral in Eq. 3.2 to calculate the absorption $\tau(r)$. The data vector

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{x}) + \boldsymbol{\eta} \tag{3.6}$$

includes an additive noise vector $\boldsymbol{\eta} \in \mathbb{R}^m$, where we define the non-linear forward model as $\boldsymbol{A}(\boldsymbol{x}) \coloneqq \boldsymbol{A}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T}) \in \mathbb{R}^m$ for brevity. Similarly, we define $\boldsymbol{A}_L \in \mathbb{R}^{m \times n}$, which denotes the linear forward model matrix and neglects absorption (e.g. set $\tau = 1$ in Eq. (3.2)) and enables matrix-vector multiplication $\boldsymbol{A}_L\boldsymbol{x}$ to compute noise-free linear data. Further, we classify the inverse problem as a weakly non-linear inverse problem, because neglecting the absorption changes the measurements only slightly (about 1%, see Chapter 5).

As the ground truth for our methodology, we consider an ozone profile at distinct pressure values generated from some data [51] of the MLS on the Aura satellite within the Antarctic region. This ozone profile has a peak in the middle atmosphere and a second peak at higher altitudes, see Fig. 4.5, which seems to be a typical nighttime profile [30].

We recursively relate pressure p to geometric height h with the hydrostatic equilibrium equation

$$\frac{\mathrm{d}p}{p} = \frac{-gM}{R^*T}\mathrm{d}h\,, \tag{3.7}$$

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

Table 3.1: Definition of height depending temperature gradients.

starting with a pressure of 1013.25hPa at sea level. The acceleration due to gravity is

$$g = g_0 \left(\frac{r_0}{r_0 + h} \right), \tag{3.8}$$

where the polar radius of the earth is $r_0 \approx 6356\,\text{km}$, the gravitation at sea level is $g_0 \approx 9.81\text{m/s}^2$, $R^* = 8.31432 \times 10^{-3}\text{Nm/kmol/K}$ and the mean molecular weight of the air is $M = 28.97\text{kg/kmol}$ [59]. This holds up to a geometric height of 86km, where we ignore a 0.04% non-linear change in M from 80km to 86km.

Following [59] we form the temperature function

$$T(h) = \begin{cases} T_0 & , h = 0 \\ T_0 + a_0 h & , 0 \leq h < h_{T,1} \\ T_0 + a_0 h_{T,1} & , h_{T,1} \leq h < h_{T,2} \\ T_0 + a_0 h_{T,1} + a_1(h_{T,2} - h_{T,1}) + a_2(h - h_{T,2}) & , h_{T,2} \leq h < h_{T,3} \\ T_0 + a_0 h_{T,1} + a_1(h_{T,2} - h_{T,1}) & \\ \quad + a_2(h_{T,3} - h_{T,2}) + a_3(h - h_{T,3}) & , h_{T,3} \leq h < h_{T,4} \\ T_0 + a_0 h_{T,1} + a_1(h_{T,2} - h_{T,1}) & \\ \quad + a_2(h_{T,3} - h_{T,2}) + a_3(h_{T,4} - h_{T,3}) + a_4(h - h_{T,4}) & , h_{T,4} \leq h < h_{T,5} \\ T_0 + a_0 h_{T,1} + a_1(h_{T,2} - h_{T,1}) & \\ \quad + a_2(h_{T,3} - h_{T,2}) + a_3(h_{T,4} - h_{T,3}) + a_4(h_{T,5} - h_{T,4}) & \\ \quad + a_5(h - h_{T,5}) & , h_{T,5} \leq h < h_{T,6} \\ T_0 + a_0 h_{T,1} + a_1(h_{T,2} - h_{T,1}) & \\ \quad + a_2(h_{T,3} - h_{T,2}) + a_3(h_{T,4} - h_{T,3}) + a_4(h_{T,5} - h_{T,4}) & \\ \quad + a_5(h_{T,6} - h_{T,5}) + a_6(h - h_{T,6}) & , h_{T,6} \leq h \lesssim 86 \end{cases} \tag{3.9}$$

with gradient and height values provided by [59] (see Tab. 3.1) which acts as the ground truth temperature profile (see Fig. 6.2).

One measurement is calculated according to the RTE as in Eq. 3.1 and Eq. 3.2 using the trapezoidal integration rule. We assume an atmosphere between $h_{L,1} = 6.9$km and $h_{L,n} = 83.3$km with $n = 45$ equidistant layers and a satellite at a fixed height of $h_{\text{sat}} = 500$km (see Fig. 3.1). The height value $h_{L,i}$ for each layer $i = 1, \ldots, n$ is defined by the pressure values from [51] and the hydrostatic equilibrium equation, see Eq. 3.7. We target ozone at a frequency of 235.71GHz, which lies within the region where the MLS observes ozone [34, 63]. The corresponding wave number is $\nu = 7.86\text{cm}^{-1}$. We calculate the absorption constant $k(\nu, T)$ as in Eq. 3.2, following the high resolution transmission (HITRAN) database [23], which provides the line intensity $L(\nu, T_{\text{ref}})$ for the isotopologue $^{16}O_3$ with the AFGL Code 666. Lastly, we add independent and identically-distributed Gaussian noise $\boldsymbol{\eta} \sim \mathcal{N}(0, \gamma^{-1}\boldsymbol{I})$ so that the SNR = 150 (see Eq. 3.11) similar to [21], where a signal with a maximal spectral intensity of around 100K and a noise range of 0.4 to 1.6K is reported. We note that the methods used in this thesis will work with different SNRs or other frequencies. Before computing a data vector $\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{x}) + \boldsymbol{\eta}$ we test different measurement strategy and asses the information provided by the forward model via a singular value decomposition (SVD).

3.2.1 Understanding the Forward Model

Through an SVD of the linear forward model matrix we provide a quick and intuitive way of assessing if the data collection is effective, how much information is passed through the forward model, and how the signal-to-noise ratio (SNR) and the measurement strategy affect that information. The SVD of the linear forward model matrix $\boldsymbol{A}_L \in \mathbb{R}^{m \times n}$ is given as

$$\boldsymbol{A}_L = \sum_{i=1}^{r} \boldsymbol{u}_i \sigma_i \boldsymbol{v}_i^T = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T \tag{3.10}$$

with $r = \min\{m, n\}$. Our main objective is to measure ozone $\boldsymbol{x}$, so our forward model $\boldsymbol{A}_L$ includes temperature and pressure, the latter is dominant, see Fig. 6.4, decreases exponentially in height and hence does affect the information passed through the model. If the pressure is high, the signal is large. If the pressure is low, the signal is low, and the data tends to be noise-dominated.

A noise-free measurement vector is given as $\boldsymbol{A}_L\boldsymbol{x}$ so the SVD of the forward model $\boldsymbol{A}_L$ provides information on how the right singular vectors $\boldsymbol{v}_i$ act on the parameter $\boldsymbol{x}$. Then the singular values σ_i, ordered in size from the largest σ_1 to the smallest σ_r, weigh that information from the right singular vectors to the left singular vectors $\boldsymbol{u}_i$. The left singular vectors project $\sigma_i \boldsymbol{v}_i^T \boldsymbol{x}$ onto the data space. For a large singular value, the forward model is informative about parameter structures represented by the corresponding right singular vector. For a small singular value, the forward model is

uninformative about parameter structures represented by the corresponding right singular vector. Further, if we define the SNR as

$$\text{SNR} := \frac{\max(y)}{\text{STD noise}} = \frac{\text{peak signal}}{\text{RMS noise}}\,, \tag{3.11}$$

and roughly assume that the maximum singular value $\sigma_1 \approx \max(y)$ then most of the information transmitted through the forward model corresponds to the singular values $\sigma_i \gtrsim \max(y)/\text{SNR}$ [18]. For very small singular values $\sigma_i \ll \sigma_1/\text{SNR}$ below the RMS noise level or the noise standard deviation (STD), an effective rank $r_{\text{eff}} \leq r$ is introduced and the data space spanned by $\{\boldsymbol{u}_{r_{\text{eff}}+1}, \ldots, \boldsymbol{u}_r\}$ is noise-dominated. For large singular values above the SNR the associated data space is hardly influenced by the noise. Hence we expect reconstructions in the parameter space spanned by the corresponding right singular vectors to be close to the ground truth. For singular values around the SNR the noise is starting to influence the data space. Hence reconstructions in the parameter space spanned by the corresponding right singular vectors are expected to have increasing uncertainties. Reconstructed parameter values in the parameter space spanned by $\{\boldsymbol{v}_{r_{\text{eff}}+1}, \ldots, \boldsymbol{v}_r\}$ (e.g. in Figure 3.6) are expected to have large variances and to be determined by the prior because they correspond to very small singular values. Further, we say a forward model matrix is informative if it has a large effective rank and the singular values decrease gradually. If the effective rank is small and singular values decrease quickly we classify this forward model as uninformative. See [57] for a more comprehensive analysis.

We test five different measurement strategies and plot the tangent heights corresponding to the pointing angles in Fig. 3.2. The measurement test cases are:

- **Case 1** includes 42 measurements between heights of $\approx$ 7km and $\approx$ 83km with pointing angles

$$\phi_j = \left(\frac{-1}{1.25^{j-1}} + 1\right)\phi_{\max}, \qquad \text{for } j = 1, \ldots, 42\,.$$

- **Case 2** includes 42 measurements between heights of $\approx$ 7km and $\approx$ 83km with pointing angles

$$\phi_j = \frac{1.25^{j-1}}{1.25^{m-1}} 0.99\phi_{\max}, \qquad \text{for } j = 1, \ldots, 42\,.$$

- **Case 3** includes 42 measurements between heights of $\approx$ 7km and $\approx$ 83km with pointing accuracy 150arc sec and pointing angles

$$\phi_j = (j-1)150\text{arc sec}, \qquad \text{for } j = 1, \ldots, 42\,.$$

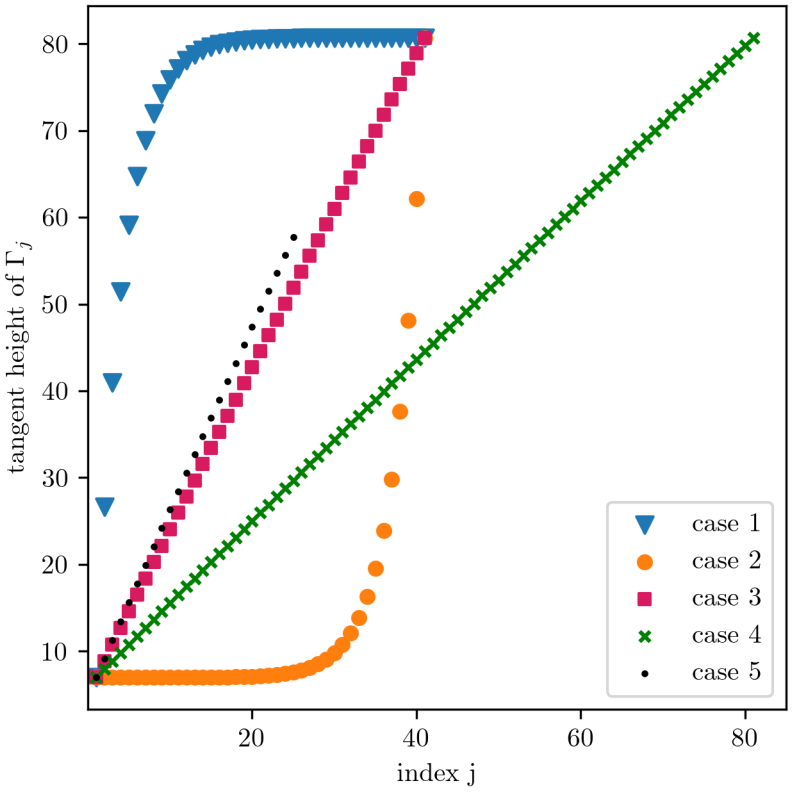


Figure 3.2: Tangent heights for five different sequences of measurements.

- **Case 4** includes 83 measurements between heights of ≈ 7km and ≈ 83km with pointing accuracy 77.5arc sec and pointing angles

$$\phi_j = (j-1)77.5\text{arc sec}, \qquad \text{for } j = 1, \ldots, 83\,.$$

- **Case 5** includes 30 measurements between heights of ≈ 7km and ≈ 68km with pointing accuracy 175arc sec and pointing angles

$$\phi_j = (j-1)175\text{arc sec}, \qquad \text{for } j = 1, \ldots, 30\,.$$

Case 1 collects more data in low signal regions at high altitudes. Case 2 collects more data in high signal regions at low altitudes. Case 3, case 4, and case 5 measure at equidistantly spaced pointing angles corresponding to different pointing accuracies. The pointing accuracy determines how well the satellite can point in a certain direction and, hence the spacing of tangent heights in the atmosphere for a stable satellite at h_{sat}. Case 1, case 2, case 3, and case 4 measure in between heights $h_{L,1} = 6.9$km and $h_{L,n} = 83.3$km, case 5 does not collect measurements in high altitude regions. The pointing accuracy for case 3 in Fig. 3.2 of 150arc sec was given to us by the team of the University of

New South Wales Canberra Space [14]. Case 4 has half the pointing accuracy of case 3, and case 5 has a slightly larger pointing accuracy than case 3. We visually assess the effective rank and how the singular values behave to determine which of the test cases is most effective. More specifically, if the singular values decay fast and only a few singular values are above an SNR of 150, the forward map is rather uninformative. If the singular values decay slowly and more singular values are above an SNR of 150, we classify the forward map as informative.

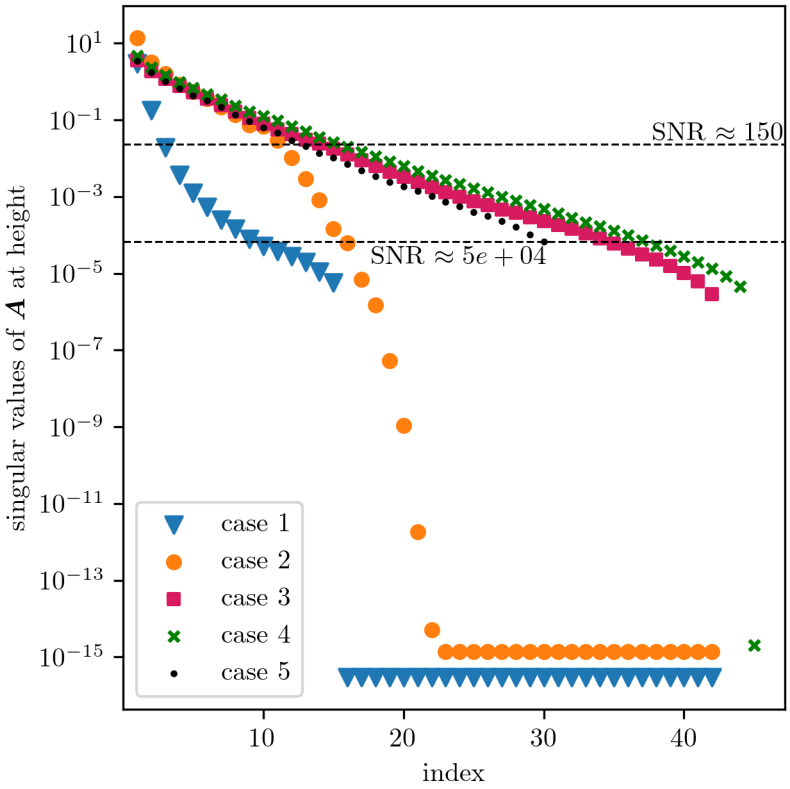


Figure 3.3: Singular values of the forward model matrix for different sequences of measurements. The corresponding tangent heights of the test cases are plotted in Fig. 3.2. The dotted vertical line marks an SNR according to σ_1 of measurement case 5.

In Fig. 3.3, we plot the singular values for each of the measurement cases. The dotted lines in Fig. 3.3 correspond to an SNR of roughly 150 with respect to the largest singular value of case 5 and an SNR according to the lowest singular value of case 5, which would be required to reconstruct all information provided by the forward model. The largest singular value of case 5 has approximately the same value as the largest singular values of case 1, case 3, and case 4, so the dotted line for an SNR of ≈ 150 is indicative for those cases as well. Fig. 3.3 shows that case 2 has the largest singular value of all cases but its singular values decrease faster than the singular values of cases 3, 4, and 5 especially below the SNR of 150. Additionally, case 2 has smaller effective rank than cases 3, 4, and

5. Case 1 does obtain singular values which are decreasing the fastest of all cases and has the largest number of singular values below an SNR of 150 and hence the smallest effective rank. We conclude that neither case 1 nor case 2 is effective.

Cases 3, 4, and 5 with equidistantly spaced pointing angles have similar effective ranks and the singular values do not decrease as quickly compared to case 1 and case 2. Case 4 measures almost twice as much compared to case 3, but does not provide much more information, which would justify the engineering effort required to achieve such pointing accuracy. The slightly larger pointing accuracy in case 5 compared to case 3 provides similar information. The last 5 to 10 singular values of case 3 are so small that the information will be completely covered by the noise. Hence, case 5 does not measure in noise-dominated regions and only up to a height of $\approx$ 68km instead of $\approx$ 83km without losing too much crucial information above the SNR. Note, that if one wanted to obtain all information provided by the forward model, an SNR of roughly 10^4 is required.

In principle, this shows that it does matter how one measures, but one cannot get much more information by measuring more in regions where the information content is low or high. The test cases show that an efficient measurement strategy may consist of equidistantly spaced pointing angles and does stop measuring when the singular values are too low. Consequently, we proceed with case 5 and plot the right singular vectors of the forward model versus height in the atmosphere to see to which parameter structures our model is sensitive.

The parameter space of $\boldsymbol{A}_L$ spanned by the first 10 right singular vectors plotted in Fig. 3.4 corresponds to the 10 largest singular values in Fig. 3.3 and represents parameter structures in the lower atmospheric regions. So we can assume that, given some data, we will be able to provide good reconstructions of the parameter in lower altitudes up to $\approx$ 30km. The right singular vectors in Fig. 3.5 correspond to the singular values σ_j for $j = 11, \ldots, 20$ around the SNR of 150, in Fig. 3.3. This is roughly where the noise starts to dominate the data. The parameter space spanned by those right singular vectors represents parameter values in the middle atmosphere. Consequently, we expect an increasing uncertainty of reconstructed parameter values at heights between $\approx$ 20km and $\approx$ 55km. The singular vectors in Fig. 3.6 corresponding to the smallest 10 singular values and span structures in parameter space at higher altitudes and the corresponding data space is noise-dominated. That is why, we will not be able to reconstruct parameter values from the ground truth above $\approx$ 55km. More specifically, the retrieved parameter values at higher altitudes will be mostly determined by the prior or, in the case of a regularisation approach, by the regulariser [57].

Now we can compute a data vector $\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{x}) + \boldsymbol{\eta}$, with $m = 30$ measurements determined by the satellite pointing accuracy of 175arc sec (see case 5 in Fig. 3.2),

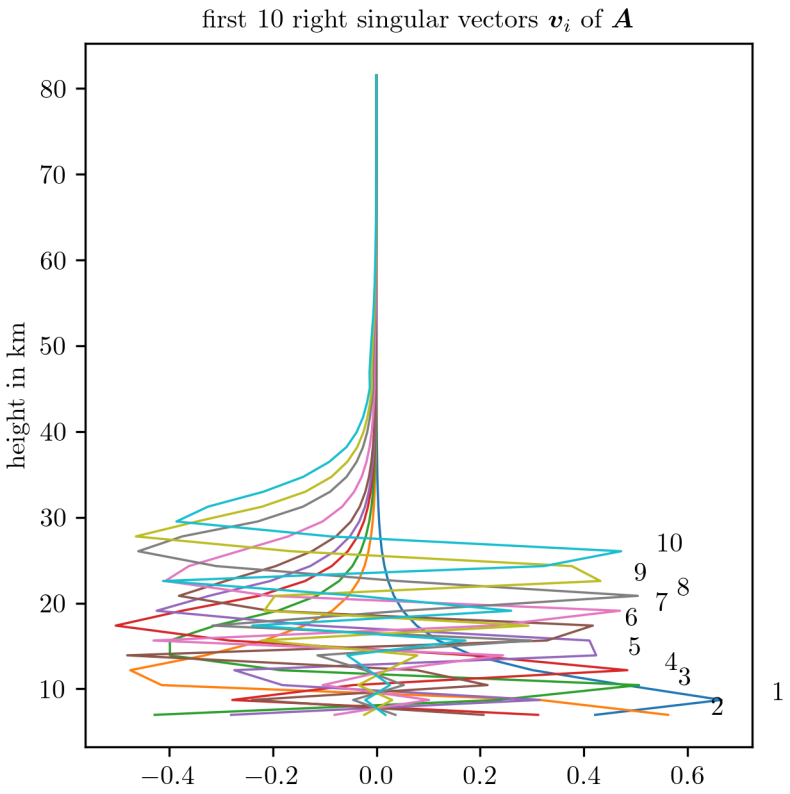


Figure 3.4: First 10 right singular vectors of the forward model matrix for measurements case 5 in Fig. 3.2. These singular vectors correspond to high singular values of the forward model in Fig. 3.3.

according to the RTE as in Eq. 3.1 and Eq. 3.2 using the trapezoidal integration rule. As already mentioned, we set the SNR to 150 and plot the data in Fig. 3.7, which is noise-dominated in higher altitudes. Given the data, we like to determine the posterior distributions over ozone $\boldsymbol{x}$, pressure $\boldsymbol{p}$ and temperature $\boldsymbol{T}$.

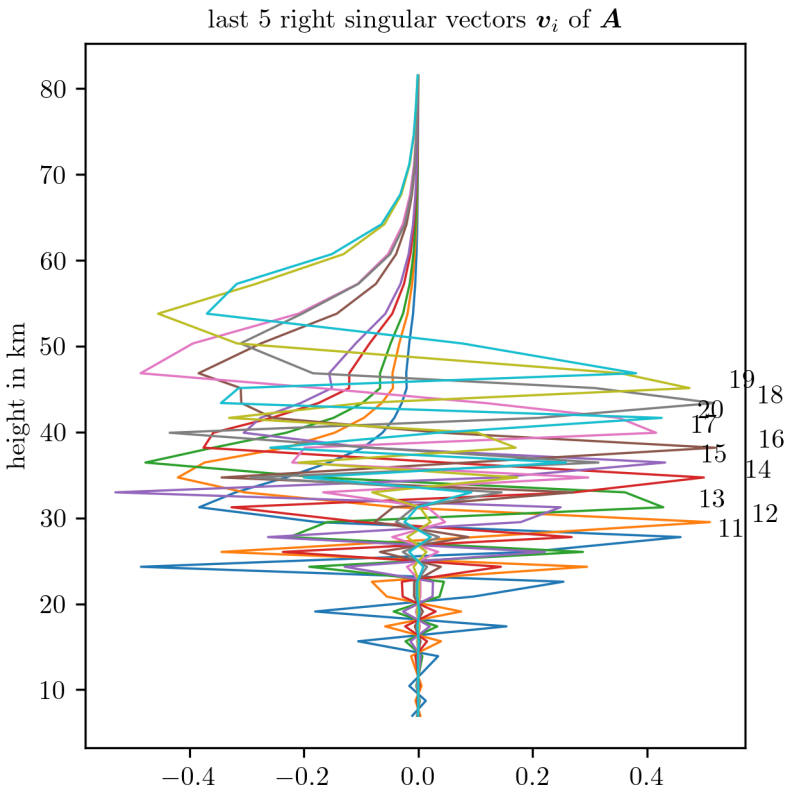


Figure 3.5: Right singular vectors with index $i = 11, \ldots, 19$ of the forward model matrix for measurements case 5 in Fig. 3.2. These singular vectors correspond to singular values in Fig. 3.3, where the noise level is similar to the data.

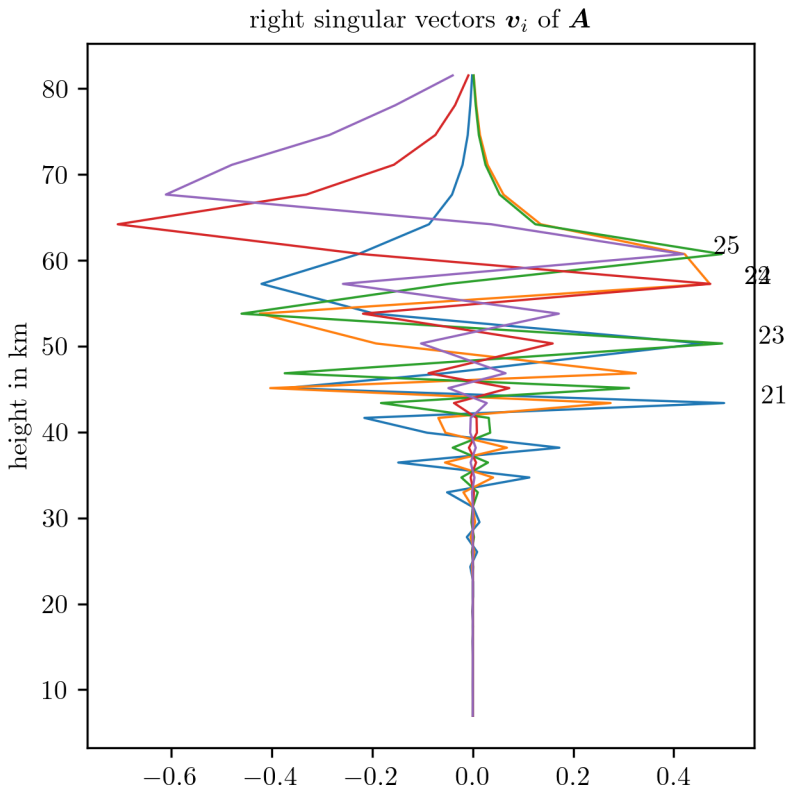


Figure 3.6: Last 10 right singular vectors of the forward model matrix for measurements case 5 in Fig. 3.2. These singular vectors correspond to small singular values of the forward model in Fig. 3.3, where the data is noise-dominated.

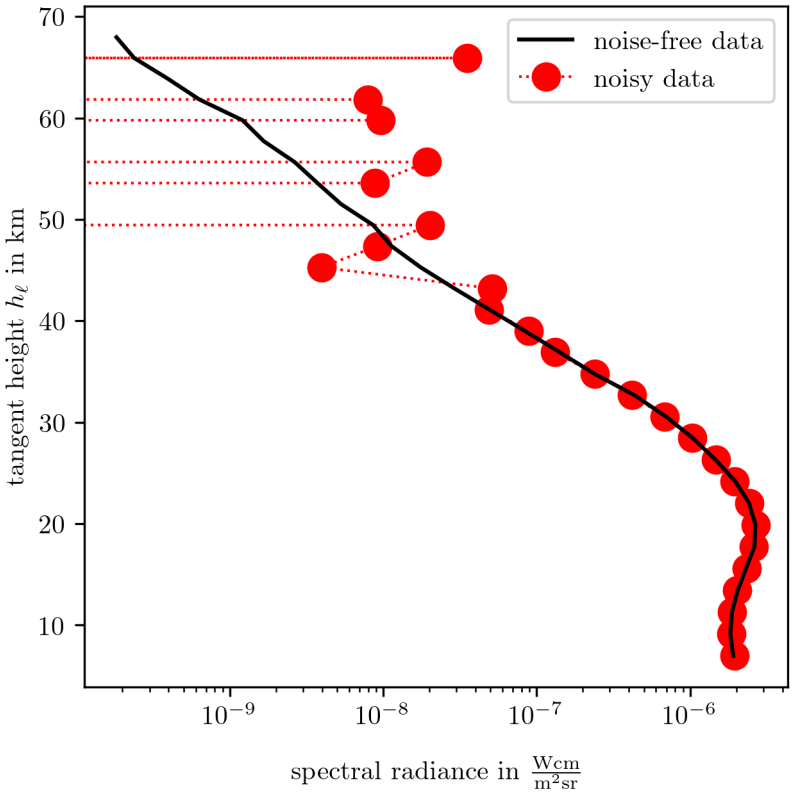


Figure 3.7: Logarithmic plot of data points at different tangent height. Note that negative values are not plotted, and noise is dominating at higher altitudes.

4

Linear Bayesian vs. Regularisation – Ozone

In this Chapter, we guide the reader through the process of setting up a hierarchical Bayesian framework, establishing a choice of prior distributions, and using a DAG to visualise conditional dependencies between hyper-parameters and parameters. Applying the MTC scheme the marginal and then full conditional posterior distributions are explicitly formulated. Here this inverse problem is treated as a linear inverse problem by neglecting the absorption term in RTE (see Eq. 3.1). A Metropolis within Gibbs sampler and a TT approximation to characterise the marginal posterior are utilised. Then we calculate the mean and the covariance matrix of the posterior distribution for ozone and compare it to a regularisation approach.

4.1 Hierarchical Bayesian Framework

In this section, we set up the hierarchically-ordered linear-Gaussian Bayesian framework to determine the ozone posterior distribution, conditioned on ground truth temperature and pressure. For now the forward model matrix is defined as $\boldsymbol{A} \coloneqq \boldsymbol{A}_L$ and the distributions of that Bayesian model are:

$$\boldsymbol{y}|\boldsymbol{x}, \gamma, \delta \sim \mathcal{N}(\boldsymbol{A}\,\boldsymbol{x}, \gamma^{-1}\boldsymbol{I}) \tag{4.1a}$$

$$\boldsymbol{x}|\delta \sim \mathcal{N}(\boldsymbol{0}, (\delta\boldsymbol{L})^{-1}) \tag{4.1b}$$

$$\delta \sim \Gamma(\alpha_\delta, \beta_\delta) \tag{4.1c}$$

$$\gamma \sim \Gamma(\alpha_\gamma, \beta_\gamma)\,. \tag{4.1d}$$

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

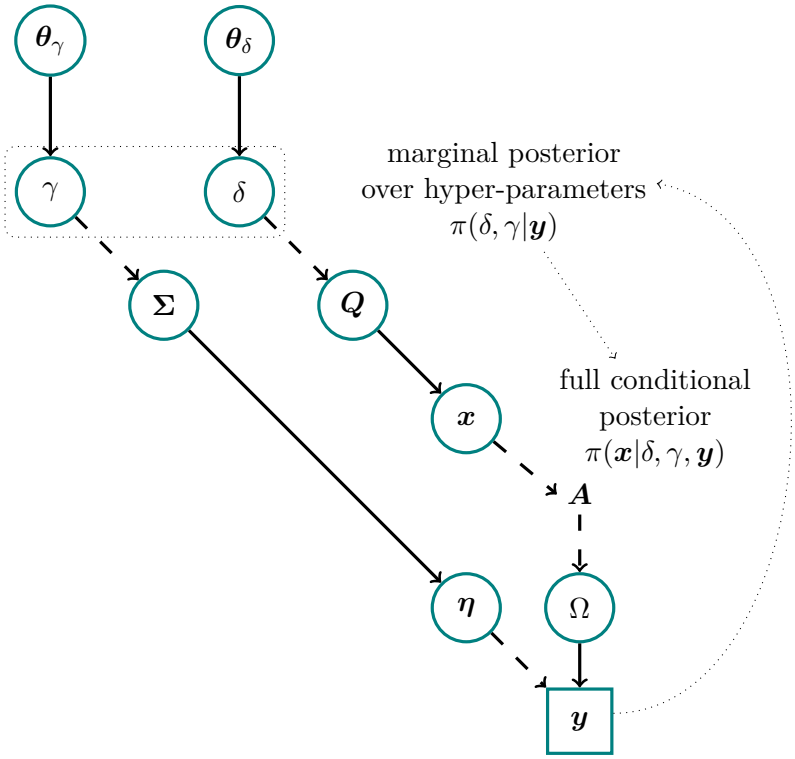


Figure 4.1: DAG for visualisation of hierarchical modelling and measuring process of ozone, including the MTC scheme. The hyper-parameter γ deterministically (dotted line) sets the noise covariance $\boldsymbol{\Sigma} = \gamma^{-1}\boldsymbol{I}$ and hence the random (solid line) noise vector $\boldsymbol{\eta} \sim \mathcal{N}(0, \gamma^{-1}\boldsymbol{I})$. The hyper-parameter δ determines (dotted line) the prior precision matrix $\boldsymbol{Q} = \delta\boldsymbol{L}$ for the normally distributed (solid line) prior $\boldsymbol{x}|\delta \sim \mathcal{N}(0, \delta\boldsymbol{L})$, where $\boldsymbol{L}$ is a graph Laplacian, see Eq. 4.2. The hyper-prior distributions (solid line) $\pi(\delta, \gamma)$ are defined by $\boldsymbol{\theta}_\gamma$ and $\boldsymbol{\theta}_\delta$. Through a linear forward model $\boldsymbol{A}$, we generate a space of all measurable noise-free data $\boldsymbol{Ax}$ from which we randomly observe a data set $\boldsymbol{y}$ including some added noise $\boldsymbol{\eta}$. Within the MTC scheme, we evaluate the marginal posterior over the hyper-parameters $\pi(\gamma, \delta|\boldsymbol{y})$ first and then the full conditional posterior $\pi(\boldsymbol{x}|\delta, \gamma, \boldsymbol{y})$. This breaks the correlation structure of $\boldsymbol{x}$ and δ and γ, and allows us to evaluate the marginal posterior independent of $\boldsymbol{x}$.

$\pi(\boldsymbol{x}|\delta)$ with zero mean and precision matrix $\delta\boldsymbol{L}$, where δ is a smoothness hyper-parameter and $\boldsymbol{L}$ is a discrete approximation to the second derivate operator (see Eq. 4.2). Here the hyper-prior distributions $\pi(\delta)$ and $\pi(\gamma)$ are gamma distributions with shape α and rate β.

We can visualise this hierarchical structure and the conditional dependencies between hyper-parameters and parameters through a DAG, as in Fig. 4.1. The hyper-parameter γ sets the noise covariance deterministically (dotted line), but is itself statistically (solid line) defined by the hyper-prior distribution $\pi(\gamma)$. This is a gamma distribution, where $\boldsymbol{\theta}_\gamma$ determines the shape and rate of $\pi(\gamma)$. Similarly $\boldsymbol{\theta}_\delta$ defines $\pi(\delta)$, where δ accounts for smoothness of the ozone profile and sets the prior precision $\boldsymbol{Q}(\delta)$. Then $\boldsymbol{Ax}$ determines the space of all measurable noise-free data sets Ω through the linear forward model, from which we observe a data set $\boldsymbol{y}$ including some noise $\boldsymbol{\eta}$. Given that data, we "reverse the arrows" to determine the posterior distribution $\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})$ over the parameter $\boldsymbol{x}$ and the hyper-parameters $\boldsymbol{\theta}$. Usually, due to underlying correlation structures, evaluating this posterior

poses a signifiant challenge. The MTC scheme breaks this correlation and provides the marginal posterior $\pi(\delta, \gamma|\boldsymbol{y})$ first and then the full conditional posterior $\pi(\boldsymbol{x}|\delta, \gamma, \boldsymbol{y})$.

4.1.1 Prior Modelling

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

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

$$\delta \boldsymbol{L} = \delta \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix} \tag{4.2}$$

which is discrete approximation to the second derivate operator with Dirichlet boundary condition and defines a 1-dimensional Graph Laplacian as in [62, 17]. This matrix will also act as the regulariser later in Sec. 4.3. We reduce the dimension of $\boldsymbol{x}$ from 45 to 34 by discarding every second ozone VMR over a height of ≈ 47km. Doing that, while not changing $\boldsymbol{L}$, we effectively induce a larger correlation between points at higher altitude. We plot the corresponding prior ozone profiles according to $\boldsymbol{x} \sim \mathcal{N}(0, (\delta \boldsymbol{L})^{-1})$ in Fig. B.1.

For δ and γ we pick relatively uninformative gamma distributions so that $\gamma \sim \mathcal{T}(\boldsymbol{\theta_\gamma}) \propto \gamma^{\alpha_\gamma - 1} \exp(-\beta_\gamma \gamma)$ and $\delta \sim \mathcal{T}(\boldsymbol{\theta_\delta})$, where $\boldsymbol{\theta_\gamma} = \{\alpha_\gamma, \beta_\gamma\} = \{\alpha_\delta, \beta_\delta\} = \boldsymbol{\theta_\delta} = (1, 10^{-35})$ (see Fig. 5.3) similar to [17]. Those gamma distributions have another advantage when using the Metropolis within Gibbs algorithm, as in Sec. 4.2.1, to sample from the marginal posterior distribution $\pi(\delta, \gamma|\boldsymbol{y})$, where then $\pi(\gamma|\lambda, \boldsymbol{y}) \sim \mathcal{T}(\cdot)$ is a gamma distribution with $\lambda = \delta/\gamma$, and easy to sample from.

4.2 Posterior Distribution

As explained in Sec. 2.1.1, we factorise the posterior

$$\pi(\boldsymbol{x}, \delta, \gamma|\boldsymbol{y}) \propto \pi(\boldsymbol{y}|\boldsymbol{x}, \delta, \gamma)\pi(\boldsymbol{x}, \delta, \gamma) \tag{4.3}$$

into

$$\pi(\boldsymbol{x}, \delta, \gamma|\boldsymbol{y}) = \pi(\boldsymbol{x}|\delta, \gamma, \boldsymbol{y})\pi(\delta, \gamma|\boldsymbol{y}) \tag{4.4}$$

the marginal posterior $\pi(\delta, \gamma|\boldsymbol{y})$ and full conditional posterior $\pi(\boldsymbol{x}|\delta, \gamma, \boldsymbol{y})$ (see Eq. 2.7). As discussed in Sec. 2.1.1, for the linear-Gaussian case, $\boldsymbol{x}$ cancels in the marginal posterior over the hyper-parameters. Following the MTC scheme, we characterise the marginal posterior first and then the full conditional posterior.

4.2.1 Marginal Posterior

Consequently, for the hierarchical model specified in Eq. 4.1, the marginal posterior distribution over the hyper-parameters is given by

$$\pi(\lambda, \gamma|\boldsymbol{y}) \propto \lambda^{n/2+\alpha_\delta-1}\gamma^{m/2+\alpha_\delta+\alpha_\gamma-1} \exp\left\{-\frac{1}{2}g(\lambda) - \frac{\gamma}{2}f(\lambda) - \beta_\delta\lambda\gamma - \beta_\gamma\gamma\right\}, \tag{4.5}$$

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

$$f(\lambda) = \boldsymbol{y}^T\boldsymbol{y} - (\boldsymbol{A}^T\boldsymbol{y})^T(\boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L})^{-1}(\boldsymbol{A}^T\boldsymbol{y})\,, \tag{4.6a}$$

$$\text{and } g(\lambda) = \log\det(\boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L})\,. \tag{4.6b}$$

Note that when changing variables from $\delta = \lambda\gamma$ to λ the hyper-prior distribution changes to $\pi(\lambda) \propto \lambda^{\alpha_\delta-1}\gamma^{\alpha_\delta}\exp(-\beta_\delta\lambda\gamma)$, due to $\mathrm{d}\delta/\mathrm{d}\lambda = \gamma$. For each evaluation of the marginal

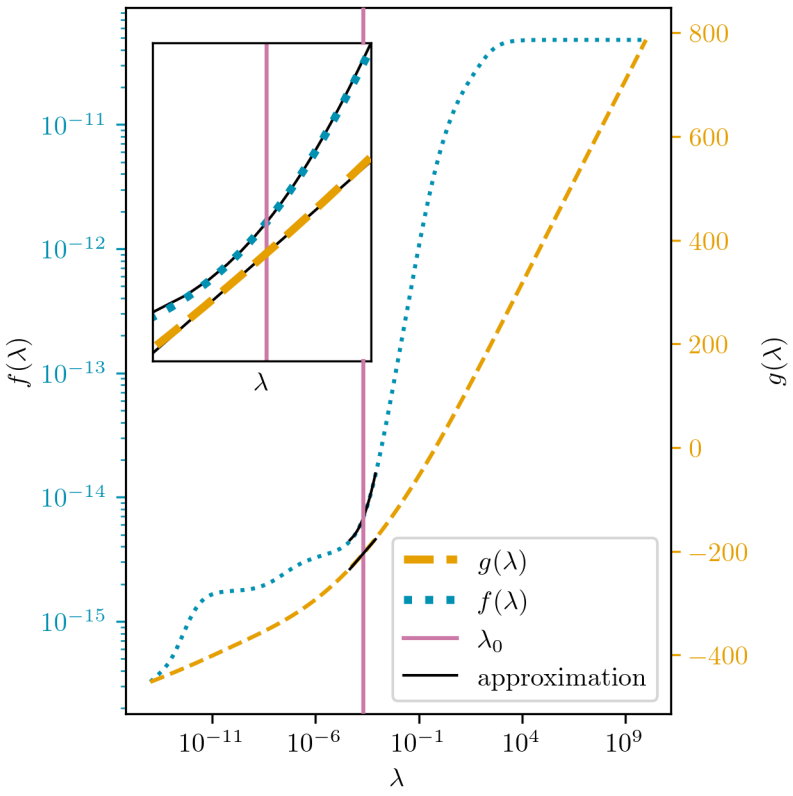


Figure 4.2: Functions $f(\lambda)$ and $g(\lambda)$ from the marginal posterior in Eq. 4.5 for a wide range of $\lambda = \delta/\gamma$. We plot the approximations (see Eq. 4.8 and Eq. 4.9) in black around the mode of the marginal posterior (vertical line) for the sampling range of λ within the MWG algorithm.

posterior most of the computational effort lies in the calculation of $f(\lambda)$ and $g(\lambda)$. Obtaining the Cholesky decomposition $\boldsymbol{A}^T\boldsymbol{A}+\lambda\boldsymbol{L} = \boldsymbol{C}_\lambda\boldsymbol{C}_\lambda^T$ via `numpy.linalg.cholesky` immediately gives $g(\lambda) = 2\sum \log \operatorname{diag}(\boldsymbol{C}_\lambda)$. Additionally to the Cholesky decomposition of $\boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L}$, the Python function `scipy.linalg.cho_solve` is used to solve for $(\boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L})^{-1}(\boldsymbol{A}^T\boldsymbol{y})$ to calculate $f(\lambda)$. If that is not too expensive one may consider a In Fig. 4.2 we see that $f(\lambda)$ and $g(\lambda)$ are well behaved within the region of interest. Because of this, we approximate $f(\lambda) \approx \tilde{f}(\lambda)$ with a Taylor series and $\tilde{g}(\lambda) \approx g(\lambda)$ with a linear approximation in λ log-space around the mode λ_0 of $\pi(\lambda, \gamma|\boldsymbol{y})$. The Taylor series coefficient of $f(\lambda)$ is given by

$$f^{(r)}(\lambda_0) = (-1)^{r+1}(\boldsymbol{A}^T\boldsymbol{y})^T(\boldsymbol{B}_0^{-1}\boldsymbol{L})^r\boldsymbol{B}_0^{-1}\boldsymbol{A}_L^T\boldsymbol{y} \tag{4.7}$$

with $\boldsymbol{B}_0 = \boldsymbol{A}^T\boldsymbol{A} + \lambda_0\boldsymbol{L}$. Note that usually a Taylor series includes a factor $(r!)^{-1}$ [17], which in this case cancels in $f^{(r)}(\lambda_0)$ so that $f(\lambda)$ is approximated as

$$\tilde{f}(\lambda) = \sum_{r=0}^{\infty} f^{(r)}(\lambda_0)(\lambda - \lambda_0)^r\,. \tag{4.8}$$

By exploratory analysis we find that the approximation

$$\tilde{g}(\lambda) = g(\lambda_0) + (\log\lambda - \log\lambda_0)\frac{g(1.25\lambda_0) - g(0.75\lambda_0)}{\log 1.25\lambda_0 - \log 0.75\lambda_0} \tag{4.9}$$

is sufficient. Note that $g(\lambda)$ can be approximated with a Taylor series as well (see [17]). We plot the function $f(\lambda)$ and $g(\lambda)$ and their approximations in Fig. 4.2 and elaborate on the approximation errors in the section below.

Error due to approximation of f and g

To assess the approximation error, we lay a 100-point grid over the sampling region in each dimension and compare the approximations of $f(\lambda)$, $g(\lambda)$ and $\pi(\lambda, \gamma|\boldsymbol{y})$ with their true function values.

Compared to a 2nd, 3-rd or 4-th order Taylor approximation, the 1-st order Taylor approximation of $f(\lambda)$ gives the smallest relative RMS error of $\approx 9\%$ for $\lambda = [10^{-5}, 8\times10^{-4}]$ (TT grid) and a maximum absolute error of $\approx 3 \times 10^{-16}$. Additionally, the linear approximation of $g(\lambda)$ has a relative RMS of $\approx 3\%$ and a maximum absolute error of ≈ 5.

These errors then propagate into the marginal posterior $\pi(\lambda, \gamma|\boldsymbol{y})$ so that the relative RMS error is $\approx 6\%$ over the whole grid. When sampling, we evaluate the acceptance ratio in the log-space, so we report a relative RMS error of $\approx 0.1\%$ for $\log\pi(\lambda|\gamma, \boldsymbol{y})$. We consider this good enough.

Sample from marginal posterior – Metropolis within Gibbs

Using these approximations, a Metropolis within Gibbs (MWG) sampler summarised in Alg. Box 3 is employed to characterise $\pi(\lambda, \gamma|\boldsymbol{y})$ as in [17]. More specifically, we implement a Metropolis random walk on

$$\pi(\lambda|\gamma, \boldsymbol{y}) \propto \lambda^{n/2+\alpha_\delta-1} \exp\left\{-\frac{1}{2}g(\lambda) - \frac{\gamma}{2}f(\lambda) - \beta_\delta\gamma\lambda\right\} \tag{4.10}$$

and do a Gibbs step on

$$\gamma|\lambda, \boldsymbol{y} \sim \Gamma\left(\frac{m}{2} + \alpha_\delta + \alpha_\gamma, \frac{1}{2}f(\lambda + \beta_\gamma + \beta_\delta\lambda\right) . \tag{4.11}$$

Ergodicity for this approach is proven in [44].

Algorithm 3: Metropolis within Gibbs

1: Initialise $(\lambda_1^{(0)}, \gamma_2^{(0)}) = (\lambda_0, \gamma_0)$
2: **for** $k = 0, \ldots, N-1$ **do**
3: Propose $\lambda' \sim q(\cdot|\lambda^{(k)}) = q(\lambda^{(k)}|\cdot)$
4: Compute

$$\alpha(\lambda'|\lambda^{(k)}) = \min\left\{1, \frac{\pi(\lambda'|\gamma^{(k)}, \boldsymbol{y})\cancel{q(\lambda^{(k)}|\lambda')}}{\pi(\lambda^{(k)}|\gamma^{(k)}, \boldsymbol{y})\cancel{q(\lambda'|\lambda^{(k)})}}\right\}$$

5: Draw $u \sim \mathcal{U}(0, 1)$
6: **if** $\alpha \geq u$ **then**
7: Accept and set $\lambda^{(k+1)} = \lambda'$
8: **else**
9: Reject and keep $\lambda^{(k+1)} = \lambda^{(k)}$
10: **end if**
11: Draw $\gamma^{(k+1)} \sim \pi(\cdot|\lambda^{(k+1)}, \boldsymbol{y})$
12: **end for**
13: Output: $(\lambda, \gamma)^{(0)}, \ldots, (\lambda, \gamma)^{(k)}, \ldots, (\lambda, \gamma)^{(N)} \sim \pi(\boldsymbol{\theta}|\boldsymbol{y})$

The MWG algorithm starts at the initial guess $(\lambda^{(0)}, \gamma^{(0)})$ at $k = 0$. We then propose a new sample $\lambda' \sim q(\cdot|\lambda^{(k)})$, conditioned on the previous state, using a symmetric proposal distribution $q(\lambda'|\lambda^{(k)}) = q(\lambda^{(k)}|\lambda')$, which is a Metropolis step and a special case of the Metropolis-Hastings algorithm [44]. We accept and set $\lambda^{(k+1)} = \lambda'$ with the acceptance probability

$$\alpha(\lambda'|\lambda^{(k)}) = \min\left\{1, \frac{\pi(\lambda'|\gamma^{(k)}, \boldsymbol{y})\cancel{q(\lambda^{(k)}|\lambda')}}{\pi(\lambda^{(k)}|\gamma^{(k)}, \boldsymbol{y})\cancel{q(\lambda'|\lambda^{(k)})}}\right\} , \tag{4.12}$$

otherwise reject and keep $\lambda^{(k+1)} = \lambda^{(k)}$. In practice, we calculate the acceptance ratio in log-space, so that

$$\log\left\{\frac{\pi(\lambda'|\gamma^{(k)}, \boldsymbol{y})}{\pi(\lambda^{(k)}|\gamma^{(k)}, \boldsymbol{y})}\right\} = \log\{\pi(\lambda'|\gamma^{(k)}, \boldsymbol{y})\} - \log\{\pi(\lambda^{(k)}|\gamma^{(k)}, \boldsymbol{y})\} \tag{4.13}$$

$$= \frac{n}{2}(\log\{\lambda'\} - \log\{\lambda^{(k)}\}) + \frac{1}{2}\Delta g + \frac{\gamma^{(k)}}{2}\Delta f + \beta_\delta\gamma^{(k)}\Delta\lambda\,, \tag{4.14}$$

where $\Delta\lambda = \lambda' - \lambda^{(k)}$ and $\Delta f \approx \tilde{f}(\lambda') - \tilde{f}(\lambda^{(k)}) = f^{(1)}(\lambda_0)\Delta\lambda' - \Delta\lambda^{(k)}$, with $\Delta\lambda' = \lambda' - \lambda_0$ and $\Delta\lambda^{(k)} = \lambda^{(k)} - \lambda_0$. Similarly we approximate $\Delta g \approx \tilde{g}(\lambda') - \tilde{g}(\lambda^{(k)})$.

Next, we perform a Gibbs step on $\pi(\gamma^{(k+1)}|\lambda^{(k+1)}, \boldsymbol{y})$ conditioned on the prevously drawn $\lambda^{(k+1)}$. Gibbs sampling is again a special case of the Metropolis-Hastings algorithm with acceptance probability equal to one. Repeating this N time give us marginal posterior samples $(\lambda, \gamma)^{(1)}, \ldots, (\lambda, \gamma)^{(N)} \sim \pi(\lambda, \gamma|\boldsymbol{y})$. To remove initialisation bias the first $N_{\text{burn-in}}$ samples are discarded.

Running the MWG sampler $f(\lambda)$ and $g(\lambda)$ are approximated around the mode (λ_0, γ_0) of $\pi(\lambda, \gamma|\boldsymbol{y})$ as previously described. The mode is provided by the `scipy.optimize.fmin` function, with a limit of 25 function evaluations. Initialised at the mode $(\lambda^{(0)}, \gamma^{(0)}) = (\lambda_0, \gamma_0)$ the MWG sampler takes $N = 10100$, which includes burn in period of $N_{\text{burn-in}} = 100$ steps. The standard deviation of the normal proposal distribution $\lambda' \sim \mathcal{N}(\lambda^{(k)}, \sigma_\lambda^2)$ is empirically set to $\sigma_\lambda = 0.8\lambda_0$, so that the acceptance rate is ≈ 0.5 as suggested in [42]. This takes ≈ 0.5s and we plot in Fig. 4.4 as well as the trace of the MWG to show ergodicity. The IACTs is given by twice the value of the Python implementation of [65] provided by [27], so that $\tau_{\text{int},\gamma} \approx 4.4 \pm 0.2$ and $\tau_{\text{int},\lambda} = 10.4 \pm 1.0$ (see Fig. 4.3 and Fig. B.2).

TT approximation of marginal posterior

Alternatively, the square root of the marginal posterior over a predefined grid can be approximated by a TT to calculate the marginals $\pi(\gamma|\boldsymbol{y})$ and $\pi(\lambda|\boldsymbol{y})$ (see Sec. 2.3.1).

The univariate grid is defined over $\gamma = [0.8 \times 10^{15}, 1.2 \times 10^{16}]$ and $\lambda = [10^{-5}, 8 \times 10^{-4}]$ with $n = 20$ grid points (see Fig. 4.7, where we argue for the number of grid points). The "normalisation constant" is set to $c = -150$ so that the values of $\sqrt{\pi(\lambda, \gamma|\boldsymbol{y})} = \exp\{0.5 \log \pi(\lambda, \gamma|\boldsymbol{y}) + c\}$ are within computer precision. Then we initialise the `tt.cross.rectcross.rect_cross.cross` function based on the TT cross algorithm in [39, 12] from the Python package `ttpy` [38] with a random tensor. The number of ranks $r = 4$ is constant and we do one sweep with $2n_{\text{sweeps}}2nr = 400$ function evaluations and obtain a TT approximation of $\pi(\lambda, \gamma|\boldsymbol{y})$ in about 0.02s. Ironically, this the same number of functions evaluations to approximate a 20×20 point grid. The TT format is especially advantageous for larger grid sizes and higher dimensional parameter spaces. To compute the marginals $\pi(\lambda|\boldsymbol{y})$ and $\pi(\gamma|\boldsymbol{y})$ the TT error is set to $\xi = 1/\boldsymbol{\lambda}(\mathcal{X})$ because we observe peak values of around 10^{47}. For Cartesian basis the Mass matrix becomes $\boldsymbol{M}_k = \text{diag}(\boldsymbol{\lambda}_k(\mathcal{X}_k))$ (see Eq. 2.28) with and $\boldsymbol{\lambda}(x) = 1$. The coefficient tensor $\boldsymbol{B}$ and $\boldsymbol{R}_{\text{pre}}$ are calculated as in Prop. 1 and Prop. 2 (see Sec. 2.3.1).

We plot the TT approximation as a colour map on top of the obtained samples in Fig. 4.4. The relative RMS TT approximation error over the whole grid is $\approx 7\%$ and similar to the propagation error in $\pi(\lambda, \gamma|\boldsymbol{y})$ due to the approximations of $f(\lambda)$ and $g(\lambda)$ (see further up).

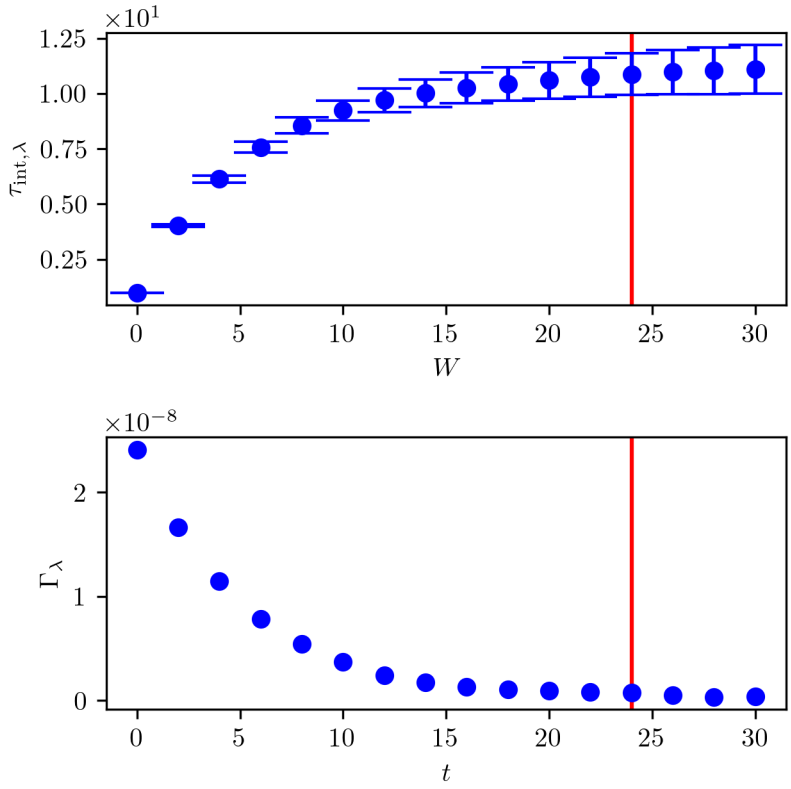


Figure 4.3: Provided by [27], the IACT $\tau_{\text{int},\lambda}$ at summation windows W as well as the estimated autocorrelation function Γ_λ at lag t of the samples $\lambda \sim \pi(\cdot|\boldsymbol{y})$.

4.2.2 Full Conditional Posterior

Finally, we can evaluate the normally distributed full conditional posterior distribution

$$\boldsymbol{x}|\delta,\gamma,\boldsymbol{y} \sim \mathcal{N}\big(\underbrace{(\boldsymbol{A}^T\boldsymbol{A}+\delta/\gamma\boldsymbol{L})^{-1}\boldsymbol{A}^T\boldsymbol{y}}_{\boldsymbol{x}_\lambda}, \underbrace{(\gamma\boldsymbol{A}^T\boldsymbol{A}+\delta\boldsymbol{L})}_{\gamma\boldsymbol{B}_\lambda}{}^{-1}\big)\,, \tag{4.15}$$

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

$$\mu_{\boldsymbol{x}|\boldsymbol{y}} = \int \boldsymbol{x}_\lambda \pi(\lambda|\boldsymbol{y})\,\mathrm{d}\lambda \approx \sum \boldsymbol{x}_{\lambda_i}\pi(\lambda_i|\boldsymbol{y})\,, \tag{4.16}$$

and posterior covariance

$$\Sigma_{\boldsymbol{x}|\boldsymbol{y}} = \int \gamma^{-1}\pi(\gamma|\boldsymbol{y})\;\mathrm{d}\gamma \int \boldsymbol{B}_\lambda^{-1}\,\pi(\lambda|\boldsymbol{y})\,\mathrm{d}\lambda \approx \sum {\gamma_i}^{-1}\pi(\gamma_i|\boldsymbol{y}) \sum \boldsymbol{B}_{\lambda_i}^{-1}\pi(\lambda_i|\boldsymbol{y}) \tag{4.17}$$

of $\pi(\boldsymbol{x}|\boldsymbol{y})$ as weighted expectations over the marginal posterior $\pi(\lambda,\gamma|\boldsymbol{y})$ by quadrature [10, Sec. 2.1] with $\sum \pi(\lambda_i|\boldsymbol{y}) = \sum \pi(\gamma_i|\boldsymbol{y}) = 1$. The weights $\pi(\lambda_i|\boldsymbol{y})$ and $\pi(\gamma_i|\boldsymbol{y})$ are either given by the TT approximation or by the bars of the sample-based histograms. More precisely, the heights of the sample-based histogram bars act as quadrature weights, where λ_i is defined at the centre of each bar. We use Cholesky decomposition of

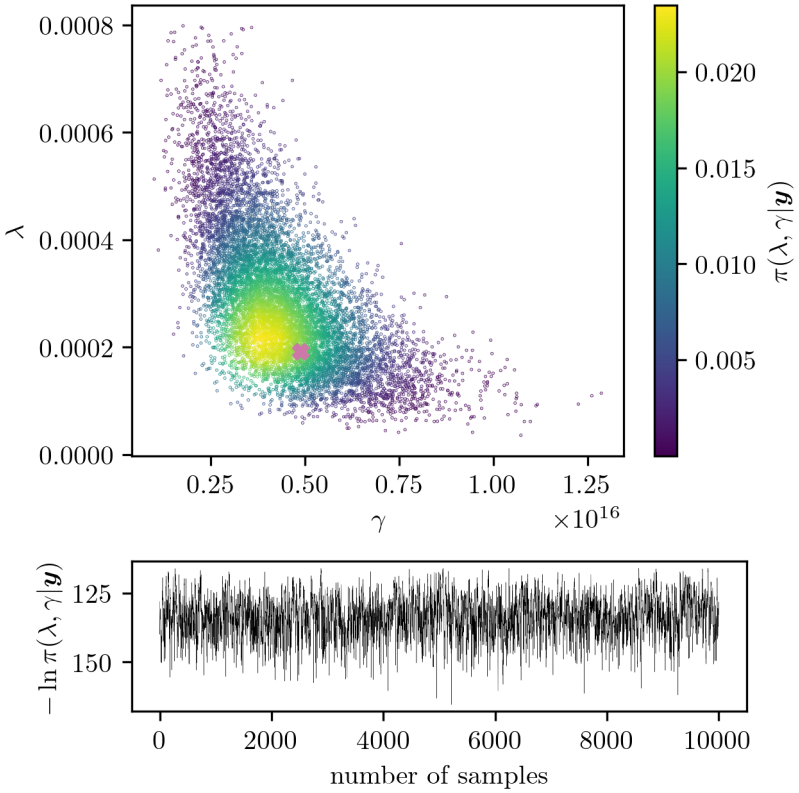


Figure 4.4: Samples from the marginal posterior colour-coded using the TT approximation of $\pi(\lambda, \gamma|\boldsymbol{y})$. The mode of (λ_0, γ_0) of $\pi(\lambda, \gamma|\boldsymbol{y})$ is marked with the pink cross. To show ergodicity, we plot the trace of the samples of the MWG algorithm.

$\boldsymbol{B}_\lambda = \boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L}$ to invert $\boldsymbol{B}_\lambda$ and to calculate $\boldsymbol{x}_\lambda = (\boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L})^{-1}\boldsymbol{A}^T\boldsymbol{y}$ both via `scipy.linalg.cho_solve`. It is sufficient to evaluate $\boldsymbol{x}_\lambda$ and invert $\boldsymbol{B}_\lambda$ 20 times to obtain mean and covariance values of $\pi(\boldsymbol{x}|\boldsymbol{y})$ within a reasonable error (see Fig. 4.7). Finding the mode of $\pi(\lambda, \gamma|\boldsymbol{y})$, running the TT `cross`, calculating the marginals and the posterior mean and variance takes 0.025s. The MWG sampler takes $\approx$ 0.5s for the same results, so most computational effort lays within the sampling procedure and the time to calculate posterior mean and variance is negligible. We plot posterior samples of $\pi(\boldsymbol{x}|\boldsymbol{y})$ in Fig. 4.5 and set negative ozone values to zero, which is observed in almost every sample. The fact that we have to deal with negative ozone values is due to the poor prior choice in $\pi(\boldsymbol{x}|\delta)$. This indicates that one should use a different, more physically based prior or model a parametrised ozone profile. Note that the posterior samples do not capture the second ozone peak at around 80km.

If calculating the variance is too costly, the RTO method (see Sec. 6.2.2) may be a feasible alternative to draw a sample from Eq. 4.15.

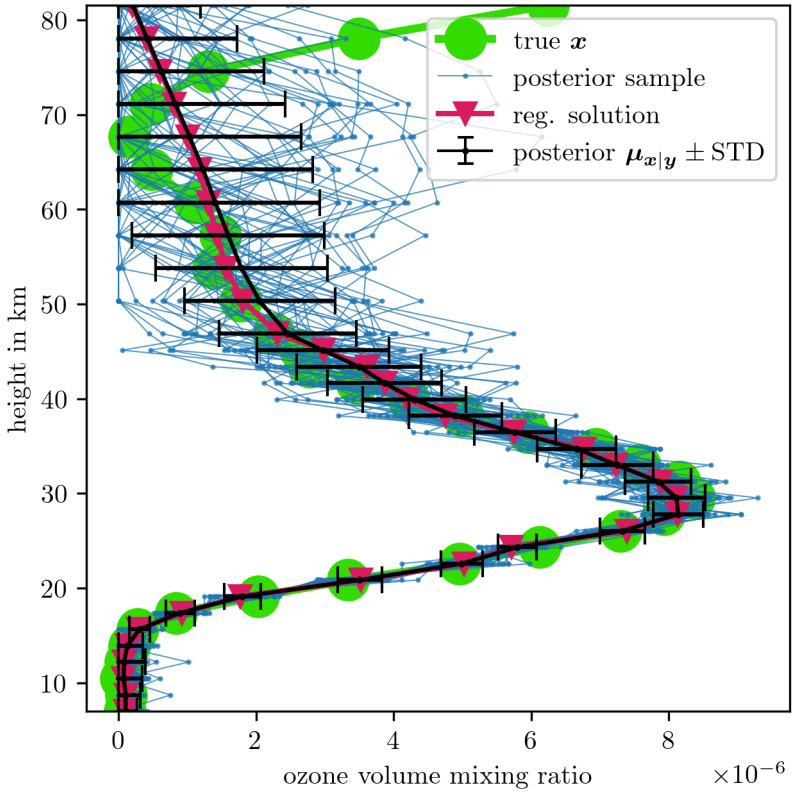


Figure 4.5: Ozone samples from the full posterior distribution $\pi(\boldsymbol{x}|\boldsymbol{y})$ after characterising posterior mean and covariance by weighted expectations over the marginal posterior $\pi(\lambda, \gamma|\boldsymbol{y})$ based on the linear forward map $\boldsymbol{A}_L$. We set negative ozone VMR values to zero.

Eigenvalues full conditional posterior covariance

In Fig. 4.6 the eigenvalues (ordered in size) of the precision matrix $\boldsymbol{Q}_{\boldsymbol{x}|\delta,\gamma,\boldsymbol{y}} = \gamma \boldsymbol{A}^T \boldsymbol{A} + \delta \boldsymbol{L}$ for a random $\delta, \gamma \sim \pi(\delta, \gamma|\boldsymbol{y})$ are plotted and compared to the eigenvalues of the prior $\delta \boldsymbol{L}$ and the forward model $\gamma \boldsymbol{A}^T \boldsymbol{A}$. We observe that the larger eigenvalues of $\boldsymbol{Q}_{\boldsymbol{x}|\delta,\gamma,\boldsymbol{y}}$ are very much the same as the larger eigenvalues of $\gamma \boldsymbol{A}^T \boldsymbol{A}$. Once the eigenvalues of $\gamma \boldsymbol{A}^T \boldsymbol{A}$ are significantly smaller than the eigenvalues of $\boldsymbol{Q}_{\boldsymbol{x}|\delta,\gamma,\boldsymbol{y}}$ the structure of the eigenvalues is dominated by the eigenvalues of $\delta \boldsymbol{L}$. The largest 10 eigenvalues of $\boldsymbol{Q}_{\boldsymbol{x}|\delta,\gamma,\boldsymbol{y}}$ include ozone profile structure at lower altitudes, where the other eigenvector mainly represent structures at higher altitudes (see Fig. B.5 and Fig. B.5). Note that the eigenvalues of each matrix may correspond to different eigenvectors even if the eigenvalues of two matrices are the same.

Errors of Full Posterior Mean and Covariance

In Fig. 4.7, we plot the relative RMS error for the mean $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}$ and covariance $\boldsymbol{\Sigma}_{\boldsymbol{x}|\boldsymbol{y}}$ of $\pi(\boldsymbol{x}|\boldsymbol{y})$ due to grid size or number of bins of the marginal posterior. Those results are obtained by calculating the weighted expectation over normalised histograms of $\pi(\lambda, \gamma|\boldsymbol{y})$, where

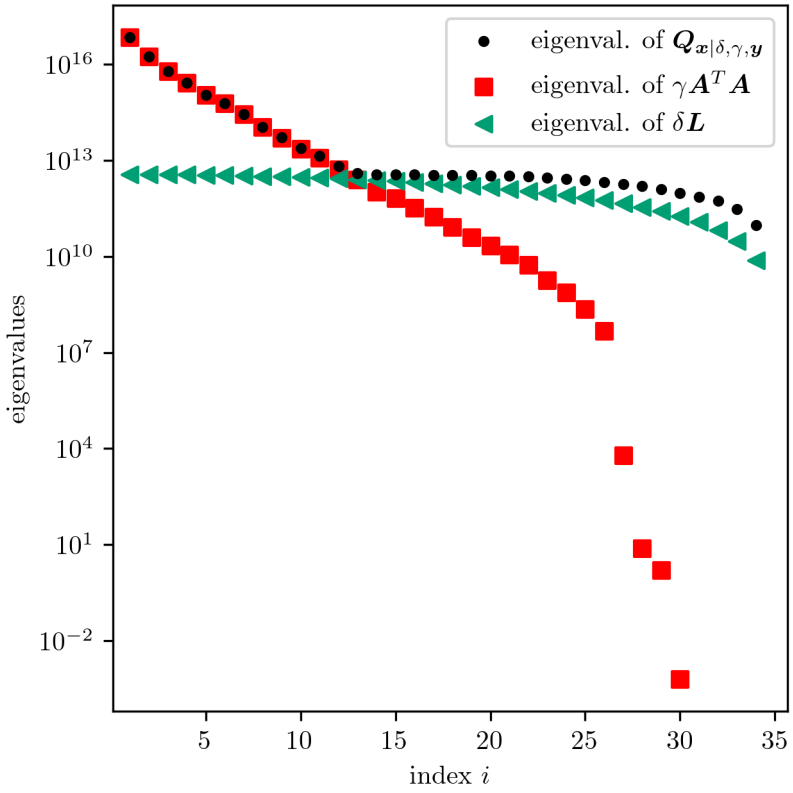


Figure 4.6: Eigenvalues of the precision matrix of $\boldsymbol{Q}_{\boldsymbol{x}|\delta,\gamma,\boldsymbol{y}} = \gamma \boldsymbol{A}^T \boldsymbol{A} + \delta \boldsymbol{L}$ of the full posterior distribution $\pi(\boldsymbol{x}|\delta, \gamma, \boldsymbol{y})$ for ozone. We see that large eigenvalues of $\gamma \boldsymbol{A}^T \boldsymbol{A}$ and $\delta \boldsymbol{L}$ are rather unaffected by the prior compared to small eigenvalues. The eigenspace may differ.

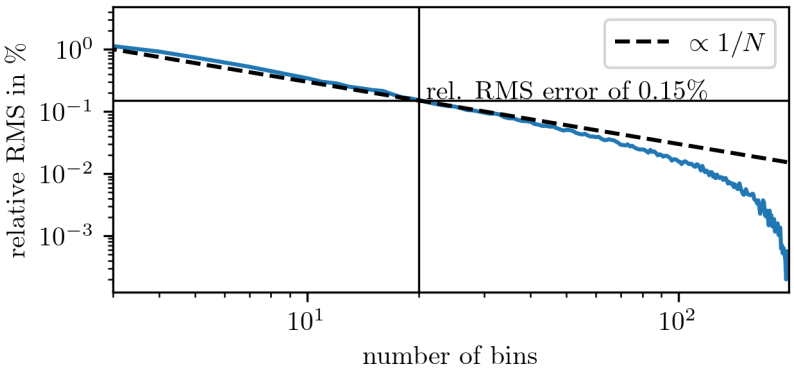


Figure 4.7: Relative RMS error of $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}$ and covariance $\boldsymbol{\Sigma}_{\boldsymbol{x}|\boldsymbol{y}}$ calculated by the weighted expectations and compared to a "ground truth" given by weighted expectations over 200 bins.

the number of bins is increased and compared to a solution calculated from a histogram with 200 bins. The relative error behaves roughly proportional to $1/N$, and we consider a relative RMS error less than 0.5% good enough, which is easily met at 20 bins. This sets the TT grid size and the number of evaluations of $\boldsymbol{x}_\lambda$ in Eq. 4.16 and $(\gamma \boldsymbol{B}_\lambda)^{-1}$ in Eq. 4.17.

4.3 Solution by Regularisation

Since we claim that the Bayesian approach is superior to regularisation methods, we compare the MTC method to a regularisation approach, which is most similar to our chosen linear-Gaussian Bayesian framework [17].

The regularised solution is defined as in [24, 17]

$$\boldsymbol{x}_\lambda = \arg\min_{\boldsymbol{x}} \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_{L^2}^2 + \lambda \boldsymbol{x}^T \boldsymbol{L}\boldsymbol{x}\,, \tag{4.18}$$

with the regularisation parameter λ, linear forward model matrix $\boldsymbol{A}$ and data $\boldsymbol{y}$. A regularised solution

$$\boldsymbol{x}_\lambda = (\boldsymbol{A}^T\boldsymbol{A} + \lambda \boldsymbol{L})^{-1}\boldsymbol{A}^T\boldsymbol{y} \tag{4.19}$$

is calculated as in Sec. 2.4.

To find the regularised solution, we use the L-curve method, and follow [26]. Within this method we compute $\boldsymbol{x}_\lambda$, for 200 different λ values in between 10^{-8} to 10^0 and plot the regularisation norm $\sqrt{\boldsymbol{x}_\lambda^T \mathbf{L}\boldsymbol{x}_\lambda}$ against the data misfit norm $\|\boldsymbol{A}\boldsymbol{x}_\lambda - \boldsymbol{y}\|_{L^2}$ (see Figure 4.8). The regularised solution corresponds to the "corner" of the L-curve at the point of maximum curvature provided by the kneedle algorithm [50] using the function `kneed.KneeLocator` in ≈ 0.015s, which is slightly faster then the TT approach to obtain full posterior mean and covariance. The corresponding regularisation parameter is $\lambda = 1.6 \times 10^{-4}$.

The regularised solution in Fig. 5.4 is very similar to the posterior mean. It is pretty clear that the regularised solution accounts for only one possible solution and does not provide uncertainties. The regularised solution is not similar to the samples drawn from the posterior $\pi(\boldsymbol{x}|\boldsymbol{y})$ (see Fig. 4.5). The samples of $\pi(\boldsymbol{x}|\boldsymbol{y})$ plotted in Fig. 4.8 lie above the L-Curve, whereas the posterior mean and the regularised solution are on the L-Curve. This does make sense, if one thinks about the mean as the (smooth) average over less-smooth samples and the regularised solution as an extremely smooth ozone profile (see Lagrangian in Sec. 2.4). In contrast, the samples are less regularised and hence lie above the L-Curve, but have a similar data misfit norm, and as already mentioned, are all feasible solutions to the data. Neither the regularisation solution nor the posterior ozone profiles capture the second ozone peak of the ground truth at high altitudes.

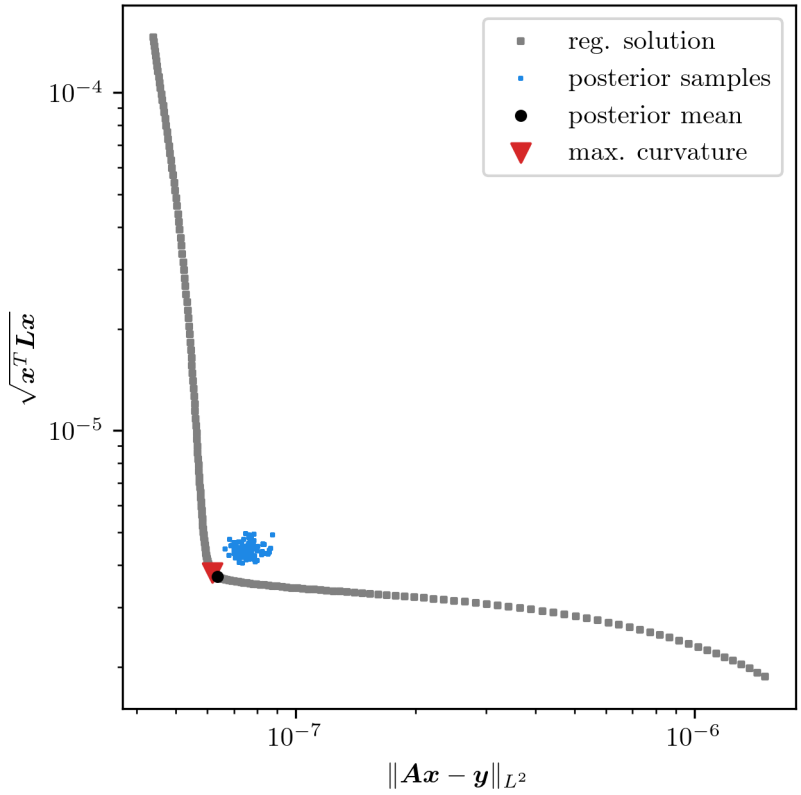


Figure 4.8: L-Curve of regularised semi norm $\sqrt{\boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x}}$ against the data misfit norm $\|\boldsymbol{A}\boldsymbol{x}-\boldsymbol{y}\|_{L^2}$ for different λ values, where $\boldsymbol{x}_\lambda$ is calculated as in Eq. 4.19. The best regularised solution is at the point of maximum curvature (pink triangle). Additionally, we calculate the data misfit norm and the regularised norm for the mean (black circle) and samples (blue squares) of the full posterior of ozone.

5

Affine Approximation of the Non-Linear Model

The forward map, introduced in Chapter 3, poses a weakly non-linear forward problem. One could tackle this non-linear inverse problem by fixing the absorption at a previously obtained parameter state and treating this as a linear inverse problem, and then iteratively update the absorption after each parameter sample. Instead, as in Fig. 5.1 illustrated, we approximate the non-linear model using an affine map, which is a linear map with a translation, e.g. $\boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}$. An affine map $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}$ maps a Gaussian $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ onto a Gaussian $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{A}\boldsymbol{\mu} + \boldsymbol{b}, \boldsymbol{A}^T\boldsymbol{\Sigma}\boldsymbol{A})$. Here we find an affine map $\boldsymbol{M}$ based on the linear model $\boldsymbol{A}_L$ that provides an approximation of the non-linear model $\boldsymbol{A}(\boldsymbol{x})$ for parameters $\boldsymbol{x}$ near the posterior mean $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}$.

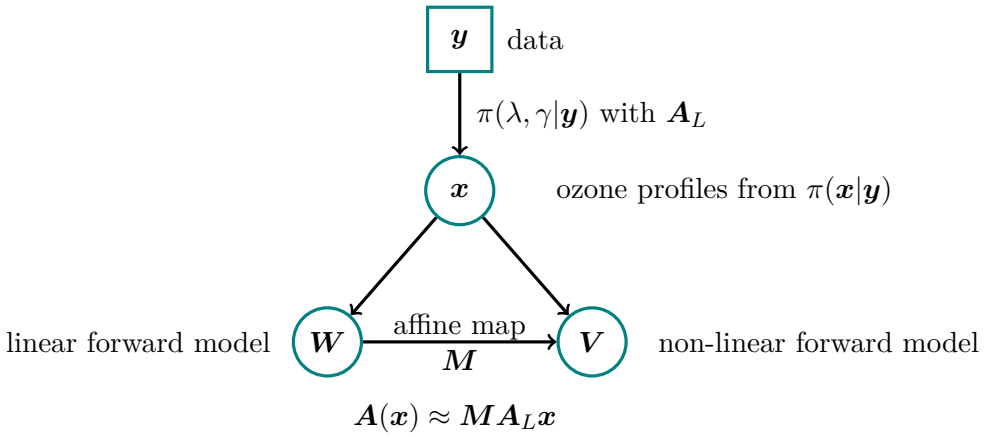


Figure 5.1: The strategy to find the affine map consists of first evaluating the marginal posterior for ozone $\pi(\lambda, \gamma|\boldsymbol{y})$ based on the linear forward model. Based on ozone samples from the full posterior, we find an affine map $\boldsymbol{M}$ which approximates between noise-free linear data $\boldsymbol{A}_L$ and noise-free non-linear data $\boldsymbol{A}(\boldsymbol{x})$.

5.1 Finding an Affine Map

We find an affine map by creating the vector spaces $\boldsymbol{W}$ based on the linear forward model and $\boldsymbol{V}$ based on the non-linear forward model with ground truth pressure and temperature. More specifically $m-1$ samples $\boldsymbol{x}^{(j)} \sim \pi(\boldsymbol{x}|\boldsymbol{y})$, for $j = 2, \ldots, m$, from the posterior and the posterior mean $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}$ generate,

$$\boldsymbol{W} = \begin{bmatrix} | & | & & | & & | \\ \boldsymbol{A}_L\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}} & \boldsymbol{A}_L\boldsymbol{x}^{(2)} & \cdots & \boldsymbol{A}_L\boldsymbol{x}^{(j)} & \cdots & \boldsymbol{A}_L\boldsymbol{x}^{(m)} \\ | & | & & | & & | \end{bmatrix} \in \mathbb{R}^{m\times m}$$

and

$$\boldsymbol{V} = \begin{bmatrix} | & | & & | & & | \\ \boldsymbol{A}(\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}) & \boldsymbol{A}(\boldsymbol{x}^{(2)}) & \cdots & \boldsymbol{A}(\boldsymbol{x}^{(j)}) & \cdots & \boldsymbol{A}(\boldsymbol{x}^{(m)}) \\ | & | & & | & & | \end{bmatrix} = \begin{bmatrix} — & v_1 & — \\ & \vdots & \\ — & v_j & — \\ & \vdots & \\ — & v_m & — \end{bmatrix} \in \mathbb{R}^{m\times m}.$$

Then the non-linear forward model is approximated as

$$\boldsymbol{A}(\boldsymbol{x}) \approx \boldsymbol{M}\boldsymbol{A}_L\boldsymbol{x}\,, \tag{5.1}$$

where we solve $v_j = r_j\boldsymbol{W}$ for each row r_j in

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

using the Python function `numpy.linalg.solve`. This is feasible since every noise-free measurement is independent of each other, and then every row v_j of $\boldsymbol{V} \in \mathbb{R}^{m\times m}$ is independent of each other as well. For an $\boldsymbol{x} = \boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}} + \Delta\boldsymbol{x}$ we rewrite Eq. 5.1 to

$$\boldsymbol{A}(\boldsymbol{x}) \approx \underbrace{\boldsymbol{M}\boldsymbol{A}_L\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}}_{=\boldsymbol{A}(\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}})} + \underbrace{\boldsymbol{M}\boldsymbol{A}_L\Delta\boldsymbol{x}}_{=\boldsymbol{A}'(\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}})\Delta\boldsymbol{x}} \tag{5.2}$$

$$= \underbrace{\boldsymbol{A}'(\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}})\boldsymbol{x}}_{\boldsymbol{A}\boldsymbol{x}} + \underbrace{\boldsymbol{A}(\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}) - \boldsymbol{A}'(\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}})\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}}_{\boldsymbol{b}} \tag{5.3}$$

to show that $\boldsymbol{M} : \boldsymbol{A}_L\boldsymbol{x} \to \boldsymbol{A}(\boldsymbol{x})$ is an affine map.

The relative RMS difference $\|\mathrm{vec}(\boldsymbol{M}\boldsymbol{W}) - \mathrm{vec}(\boldsymbol{V})\|_{L^2}/\|\mathrm{vec}(\boldsymbol{M}\boldsymbol{W})\|_{L^2}$ between the mapped linear noise-free data and the non-linear noise-free data is approximately 0.001%.

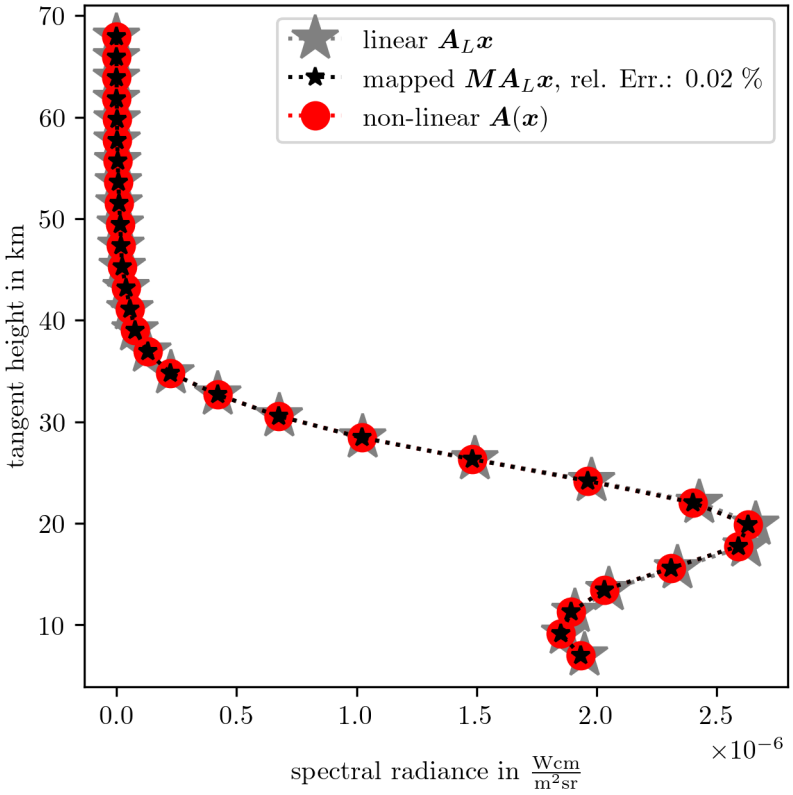


Figure 5.2: Assessment of how well we can approximate noise-free non-linear data $\boldsymbol{A}(\boldsymbol{x})$ (red circles) with noise-free linear data $\boldsymbol{A}_L\boldsymbol{x}$ (grey stars) and the previously calculated affine map $\boldsymbol{M}$. The approximated noise-free data (black stars) has a relative RMS error of $\approx 0.02\%$ compared to the true non-linear noise-free data. The ozone sample to generate this noise-free data has not been used to create the affine map.

This is much smaller than the relative RMS difference between $\boldsymbol{W}$ and $\boldsymbol{V}$ of about 1%. Fig. 5.2 shows the mapping for one posterior ozone sample with a relative RMS error $\approx 0.02\%$. This posterior ozone sample has not been used to create this mapping; in other words, this is an unseen event not occurring in the training data. Consequently, from here onwards, we use the approximated forward map.

5.2 Marginal and then Conditional Posterior – Ozone

Again, we use the MTC scheme and the exact same setup and procedure as in Sec. 4.2 to evaluate the marginal posterior and then the full conditional posterior of ozone with similar computational time.

The marginal posterior is defined as in Eq. 4.5, but with the approximated forward model. The MWG is initialised at the mode of $\pi(\lambda, \gamma|\boldsymbol{y})$ and $f(\lambda)$ and $g(\lambda)$ are approximated around the mode as in Sec. 4.2.1 (see Eq. 4.8 and Eq. 4.9). We take $N = 10000$ plus $N_{\text{burn-in}} = 100$ steps in ≈ 0.5s. The IACTs provided by [27] are

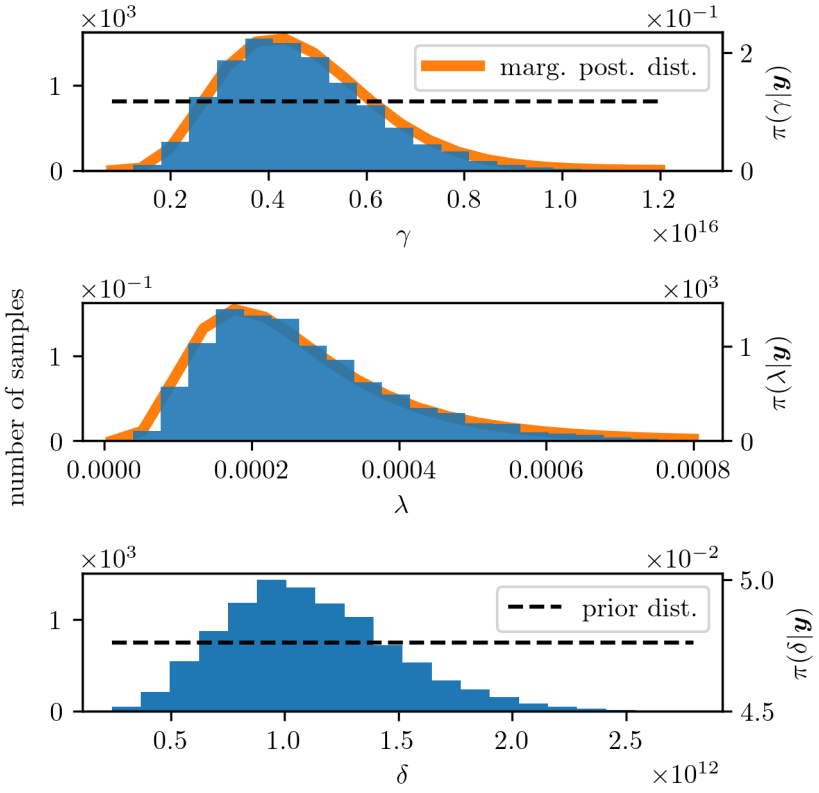


Figure 5.3: The TT approximation of the marginal posterior in orange and the samples as a histogram, as well as the prior distribution with a dotted line. We sample λ and γ using the MWG algorithm and then calculate δ for every sample of the marginal posterior. The regularised parameter corresponding to the best regularised solution (see Fig. 5.4 and Fig. 4.8) is marked with the red vertical line. We mark the ground truth noise precision with the back vertical line.

$\tau_{\text{int},\gamma} \approx 5.2 \pm 0.3$ and $\tau_{\text{int},\lambda} = 11 \pm 1$ (see Fig. B.4 and Fig. B.3) and similar to the previously calculated values. We plot the samples in Fig. 5.3 as well as the TT approximation of the marginal posterior using 400 function evaluations (same grid; same number of ranks; see Sec. 4.2.1). The relative RMS error of the TT approximation over the whole grid and the relative RMS approximation error of $\pi(\lambda, \gamma|\boldsymbol{y})$ due to the approximations of $f(\lambda)$ and $g(\lambda)$ are both $\approx 8\%$.

Again, we calculate the full posterior mean $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}$, see Eq. 4.16, and covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{x}|\boldsymbol{y}}$ 4.17 as weighted expectation. We plot the results and one sample of $\pi(\boldsymbol{x}|\boldsymbol{y})$, which represents a feasible solution to this inverse problem, in Fig. 5.4, as well as the regularised solution (see next section), and one sample from the posterior. We can see that the ground truth lies within the STD around the mean, except for the peak at around 80km. Compared to the previously calculated mean and variance based on the linear forward model $\boldsymbol{A}_L$ (see 4.5), the posterior distribution based on $\boldsymbol{M}\boldsymbol{A}_L$ does not differ significantly. This is expected since the difference between the linear and non-linear forward map of $\approx 1\%$ is small.

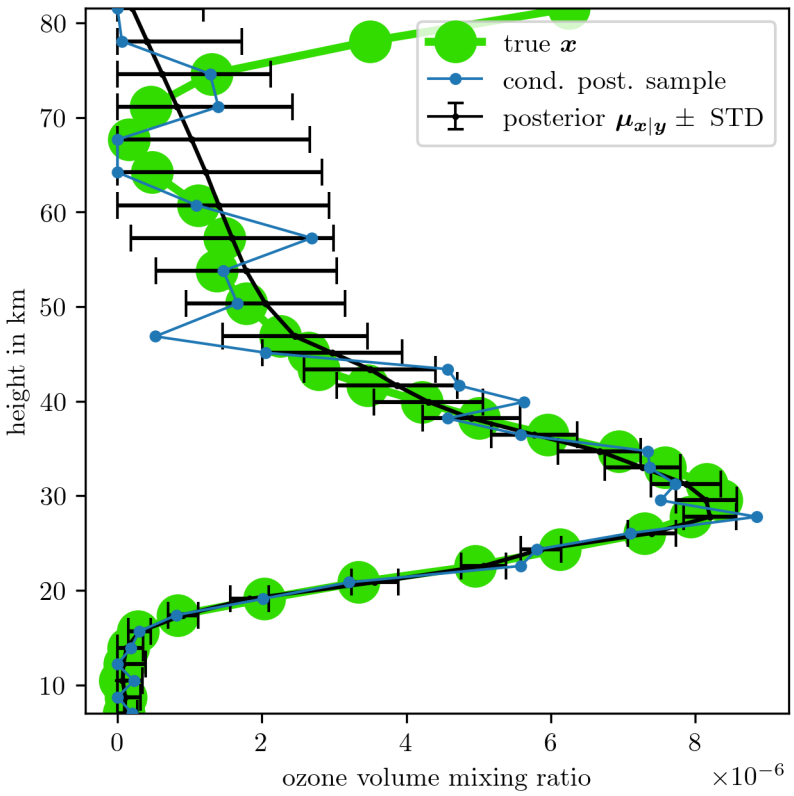


Figure 5.4: Full posterior mean and variance and one ozone sample from the full posterior. We plot the regularised solution on top of the ground truth ozone profile in green. The results are based on the approximated forward model $\boldsymbol{MA}_L$.

6

Joint Retrieval of Ozone, Pressure and Temperature

Here, we extend the hierarchical Bayesian model set up in Sec. 4.1 to include pressure and temperature related hyper-parameters and elaborate on some aspects of prior modelling. The MTC scheme is applied to jointly provide posterior distributions of ozone, pressure and temperature. Additionally, the reader is guided through the process of setting up an efficient TT approximation of the higher-dimensional marginal posterior.

6.1 Hierarchical Bayesian Framework

As in Sec. 4.1, we use a DAG as in Fig. 6.1 to visualise the measurement process and conditional dependencies between pressure $\boldsymbol{p}$, temperature $\boldsymbol{T}$ and ozone $\boldsymbol{x}$, which progress deterministically (dashed line) into the forward model, via $\boldsymbol{x} \times \boldsymbol{p}/\boldsymbol{T}$. Note that other variables in the RTE, such as the internal partition function and the black body radiation, are dependent on temperature (see Sec. 3.1). This hierarchical Bayesian framework includes the hyper-parameters p_0, b for pressure (see Eq. 6.3), $\boldsymbol{a}, \boldsymbol{h}_T, T_0$ for temperature (see Eq. 3.9), δ for ozone smoothness and γ for noise precision. Each of those hyper-parameters is described by the hyper-prior distribution $\pi(p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a}, \delta, \gamma)$ (see Sec. 6.1.1). Through their respective prior distributions, pressure $\boldsymbol{p}$, temperature $\boldsymbol{T}$ and ozone $\boldsymbol{x}$ generate a space of all possible noise-free data Ω from which we observe some data with additive normally distributed noise $\boldsymbol{\eta}$. For brevity, we define a linear forward model matrix as

$$\boldsymbol{A_\theta} \coloneqq \boldsymbol{M}\boldsymbol{A}_L(p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a})\,. \tag{6.1}$$

with $\boldsymbol{\theta} \coloneqq \{p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a}\}$ accounting for the all pressure and temperature related hyper-parameters and $\boldsymbol{M}$ the affine approximation from the previous Chapter.

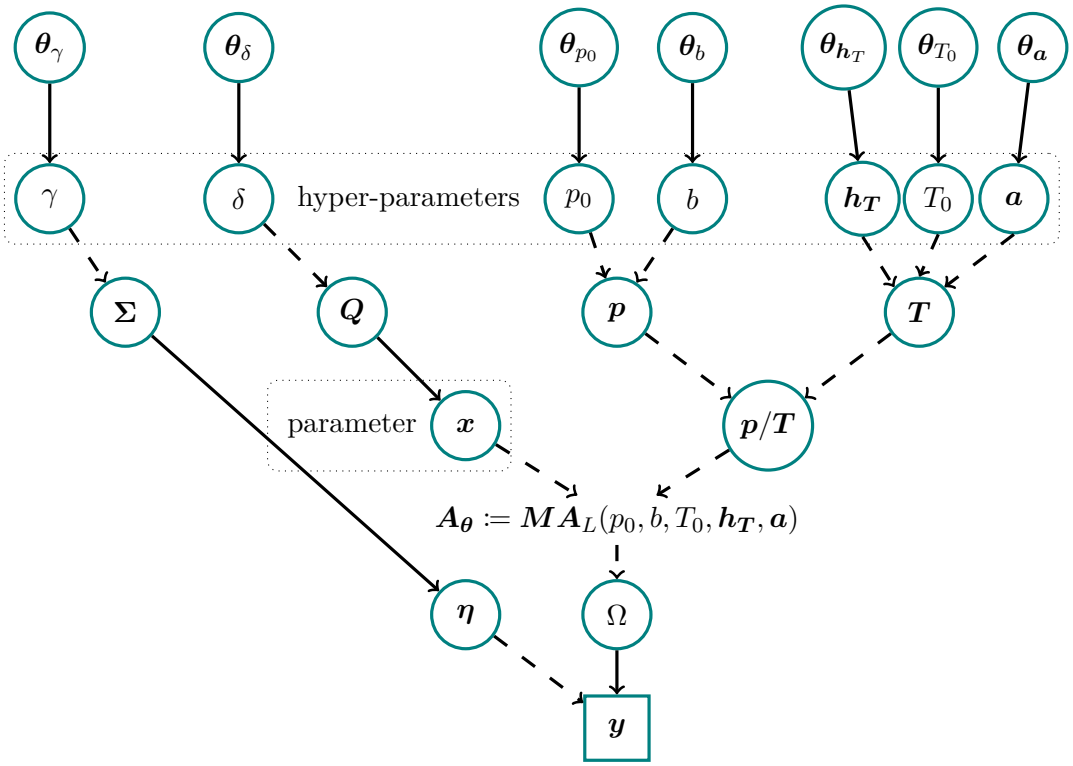


Figure 6.1: DAG of Bayesian model for ozone, pressure and temperature. The hyper-parameters $\boldsymbol{h}_T = \{h_{T,1}, h_{T,2}, h_{T,3}, h_{T,4}, h_{T,5}, h_{T,6}\}$, $\boldsymbol{a} = \{a_0, a_1, a_2, a_3, a_4, a_5, a_6\}$, T_0, b and p_0 deterministically (dotted line) describe pressure parameter $\boldsymbol{p}$ through the function in Eq. 6.3, and temperature parameter $\boldsymbol{T}$ through the function in Eq. 3.9. In this case, we choose the hyper-parameters $\pi(p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a})$ to be a normally distributed a-priori, determined by $\boldsymbol{\theta}_{\boldsymbol{h}_T}, \boldsymbol{\theta}_{\boldsymbol{a}}, \boldsymbol{\theta}_{T_0}, \boldsymbol{\theta}_b, \boldsymbol{\theta}_{p_0}$ which represent mean and variances, e.g. $b \sim \mathcal{N}(\mu_b, \sigma_b^2)$ and $\boldsymbol{\theta}_b = \{\mu_b, \sigma_b\}$. As previously described in Sec. 4.1, $\boldsymbol{\theta}_\gamma, \boldsymbol{\theta}_\delta$ determine gamma distributions, e.g. $\gamma \sim \Gamma(\alpha_\gamma, \beta_\gamma)$ with $\boldsymbol{\theta}_\gamma = \{\alpha_\gamma, \beta_\gamma\}$. The ozone parameter $\boldsymbol{x}$ is statistically (solid line) described by the prior distribution $\boldsymbol{x}|\delta \sim \mathcal{N}(0, (\delta \boldsymbol{L})^{-1})$. Here, the hyper-parameter δ accounts for smoothness in the ozone profile and defines the precision matrix $\boldsymbol{Q} = \delta \boldsymbol{L}$, where $\boldsymbol{L}$ is the graph Laplacian as in Eq. 4.2. The noise covariance $\boldsymbol{\Sigma} = \gamma^{-1}\boldsymbol{I}$ of the random noise vector $\boldsymbol{\eta} \sim \mathcal{N}(0, \gamma^{-1}\boldsymbol{I})$ is defined by the hyper-parameter γ. From the space of all measurables Ω mapped through the approximated forward model $\boldsymbol{A_\theta} \coloneqq \boldsymbol{MA}_L(p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a})$, depending on the hyper-parameter $\boldsymbol{\theta} \coloneqq \{p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a}\}$ a data set $\boldsymbol{y}$ is randomly observed (square box) including some additive noise. Given the data we like to determine the marginal posterior distribution over the hyper-parameters $\pi(\boldsymbol{\theta}, \delta, \gamma|\boldsymbol{y})$ first and then the conditional posterior distribution for ozone $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \delta, \gamma, \boldsymbol{y})$, utilising the MTC scheme.

Then, we set up the hierarchical Bayesian framework

$$\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta}, \delta, \gamma \sim \mathcal{N}(\boldsymbol{A_\theta x}, \gamma^{-1}\boldsymbol{I}) \tag{6.2a}$$

$$\boldsymbol{x}|\delta \sim \mathcal{N}(\boldsymbol{0}, (\delta \boldsymbol{L})^{-1}) \tag{6.2b}$$

$$\delta \sim \Gamma(\alpha_\delta, \beta_\delta) \tag{6.2c}$$

$$\gamma \sim \Gamma(\alpha_\gamma, \beta_\gamma) \tag{6.2d}$$

$$\boldsymbol{a} \sim \mathcal{N}(\boldsymbol{\mu_a}, \boldsymbol{\Sigma_a}) \tag{6.2e}$$

$$\boldsymbol{h_T} \sim \mathcal{N}(\boldsymbol{\mu}_T, \boldsymbol{\Sigma}_{\boldsymbol{h}_T}) \tag{6.2f}$$

$$T_0 \sim \mathcal{N}(\mu_{T_0}, \sigma_{T_0}) \tag{6.2g}$$

$$p_0 \sim \mathcal{N}(\mu_{p_0}, \sigma_{p_0}) \tag{6.2h}$$

$$b \sim \mathcal{N}(\mu_b, \sigma_b) \tag{6.2i}$$

		TT bounds		t-walk	
model parameters	priors	lower	upper	τ_{int}	Context
$\boldsymbol{x}$	$\mathcal{N}(0, (\delta \boldsymbol{L})^{-1})$	-	-	-	$\boldsymbol{x}$
δ	$\mathcal{T}(1, 10^{-35})$	-	-	-	$\boldsymbol{x}$
γ	$\mathcal{T}(1, 10^{-35})$	8×10^{14}	1.2×10^{16}	507 ± 29	$\boldsymbol{y}$
$\lambda = \delta/\gamma$	-	1×10^{-5}	2.5×10^{-3}	979 ± 75	-
b	$\mathcal{N}(0.174, (0.01)^2)$	0.129	0.214	830 ± 60	$\boldsymbol{p}$
$h_{T,1}$	$\mathcal{N}(11, (1.5)^2)$	5.4	16.3	286 ± 13	$\boldsymbol{T}$
T_0	$\mathcal{N}(288.15, (10)^2)$	247	326	279 ± 12	$\boldsymbol{T}$
p_0	$\mathcal{N}(1311, (20)^2)$	1237	1387	279 ± 12	$\boldsymbol{p}$
$h_{T,3}$	$\mathcal{N}(32.3, (2.5)^2)$	22.9	41.7	254 ± 11	$\boldsymbol{T}$
a_1	$\mathcal{N}(0, (0.1)^2)$	-0.38	0.38	295 ± 13	$\boldsymbol{T}$
$h_{T,2}$	$\mathcal{N}(20.1, (0.7)^2)$	17.2	22.7	296 ± 13	$\boldsymbol{T}$
a_0	$\mathcal{N}(-6.5, (0.01)^2)$	-6.54	-6.47	252 ± 10	$\boldsymbol{T}$
a_2	$\mathcal{N}(1, (0.01)^2)$	0.97	1.03	267 ± 11	$\boldsymbol{T}$
a_3	$\mathcal{N}(2.8, (0.1)^2)$	2.5	3.1	267 ± 11	$\boldsymbol{T}$
$h_{T,4}$	$\mathcal{N}(47.4, (0.5)^2)$	45.5	49.3	270 ± 12	$\boldsymbol{T}$
a_4	$\mathcal{N}(0, (0.1)^2)$	-0.38	0.38	254 ± 11	$\boldsymbol{T}$
$h_{T,5}$	$\mathcal{N}(51.4, (0.5)^2)$	49.5	53.3	280 ± 12	$\boldsymbol{T}$
a_5	$\mathcal{N}(-2.8, (0.1)^2)$	-3.18	-2.43	278 ± 12	$\boldsymbol{T}$
$h_{T,6}$	$\mathcal{N}(71.8, (3)^2)$	60.5	83.1	250 ± 10	$\boldsymbol{T}$
a_6	$\mathcal{N}(-2, (0.01)^2)$	-2.04	-1.96	272 ± 12	$\boldsymbol{T}$

Table 6.1: Summary of relevant parameter characteristics, bounds and sampling statistics. We denote $\mathcal{N}(\mu, \sigma^2)$ as the Gaussian and $\mathcal{T}(\alpha = \text{scale}, \beta = \text{rate})$ as the gamma distribution. The IACT τ_{int} is estimated as in [66] from posterior samples based on the approximated forward map.

and define a normally distributed likelihood (due to Gaussian noise) and normally distributed priors. Before we formulate the posterior distribution, we carefully define $\boldsymbol{\theta}_\gamma, \boldsymbol{\theta}_\delta, \boldsymbol{\theta}_{p_0}, \boldsymbol{\theta}_b, \boldsymbol{\theta}_{\boldsymbol{h}}, \boldsymbol{\theta}_{T_0}, \boldsymbol{\theta}_{\boldsymbol{a}}$, the hyper-prior scales, shapes, means and variances, which are explicitly given in Tab. 6.1.

6.1.1 Prior Modelling

We start by describing the pressure $\boldsymbol{p}$ in between $h_{L,0} \approx 7$km and $h_{L,n} \approx 82$km with an exponential function

$$p(h) = \exp(-b\,h)\; p_0 \quad , h_{L,0} \leq h \leq h_{L,n} \tag{6.3}$$

depending on two hyper-parameters p_0, b (see Fig. 6.3). Similarly, the temperature as described in Eq. 3.9 can be parametrised with 14 hyper-parameters

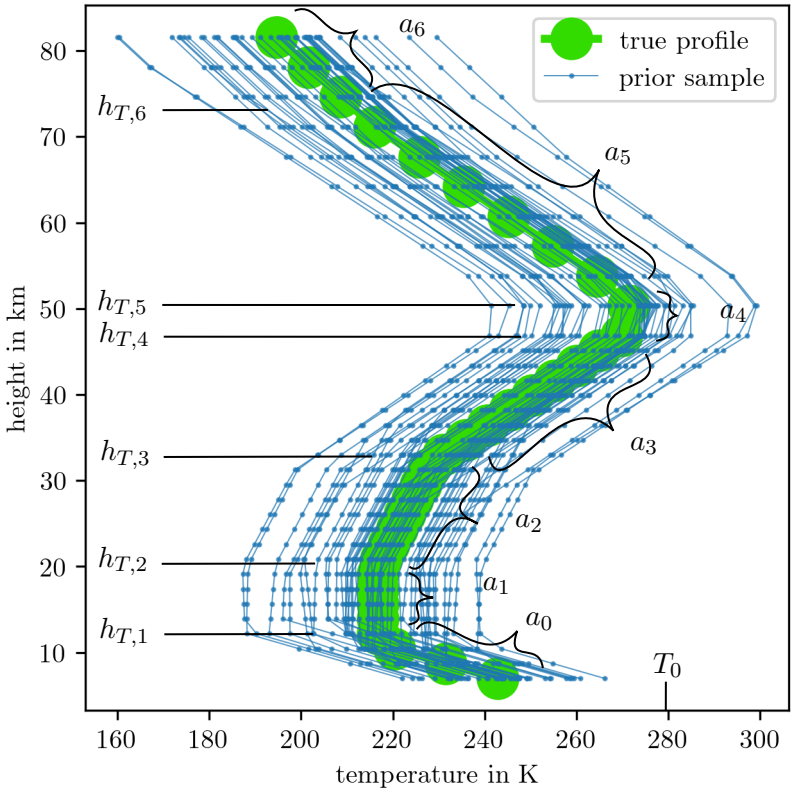


Figure 6.2: Prior samples from the hyper-prior distribution of $\boldsymbol{h}_T$, $\boldsymbol{a}$ and T_0, as defined in Tab. 6.1, where we calculate $\boldsymbol{T}$ according to the function in Eq. 3.9.

$\boldsymbol{h}_T = \{h_{T,1}, h_{T,2}, h_{T,3}, h_{T,4}, h_{T,5}, h_{T,6}\}$, $\boldsymbol{a} = \{a_0, a_1, a_2, a_3, a_4, a_5, a_6\}$ and T_0 (see Fig. 6.2 and Eq. 3.9). We define the Gaussian hyper-prior distributions for $p_0, b, T_0, \boldsymbol{h}_T, \boldsymbol{a}$, and to complete the model we have to define sensible hyper-prior variances and means. The in Sec. 4.1.1 defined gamma distributions $\pi(\delta, \gamma)$ are not changing. We tune the normal distribution $\pi(\boldsymbol{h}_T)$, so that the temperature profile maintains its structure, $h_{T,i} < h_{T,i+1}$ for $i = 1, \ldots, 5$ (see Fig. B.7) and set $\pi(\boldsymbol{a})$ to a normal distribution as well. Similarly, we set $\pi(T_0)$ to a normal distribution, so that it mimics a daily temperature variability of roughly 30K. The hyper-prior distributions are rather informative, because we find that the data and the model (see Fig. 6.4) are uninformative about the temperature profile. The hyper-prior distribution $\pi(p_0, b)$ for pressure-related hyper-parameters is also normally distributed, but with a rather large variance σ_b^2, where p_0 has a variability of around 80hPa, close to what we can observe when looking at weather data. The means of the normal distribution $\pi(p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a})$ are set to the ground truth values of $\boldsymbol{T}$ and $\boldsymbol{p}$. For $\boldsymbol{p}$, this is provided by the Python function `scipy.optimize.curve_fit`. See Tab. 6.1 for a summary of the hyper-prior distributions.

We plot prior samples of the pressure $\boldsymbol{p}$ in Fig. 6.3, the temperature $\boldsymbol{T}$ in Fig. 6.2 and the ratio $\boldsymbol{p}/\boldsymbol{T}$ in Fig. 6.4 against the ground truth profiles, and additionally prior

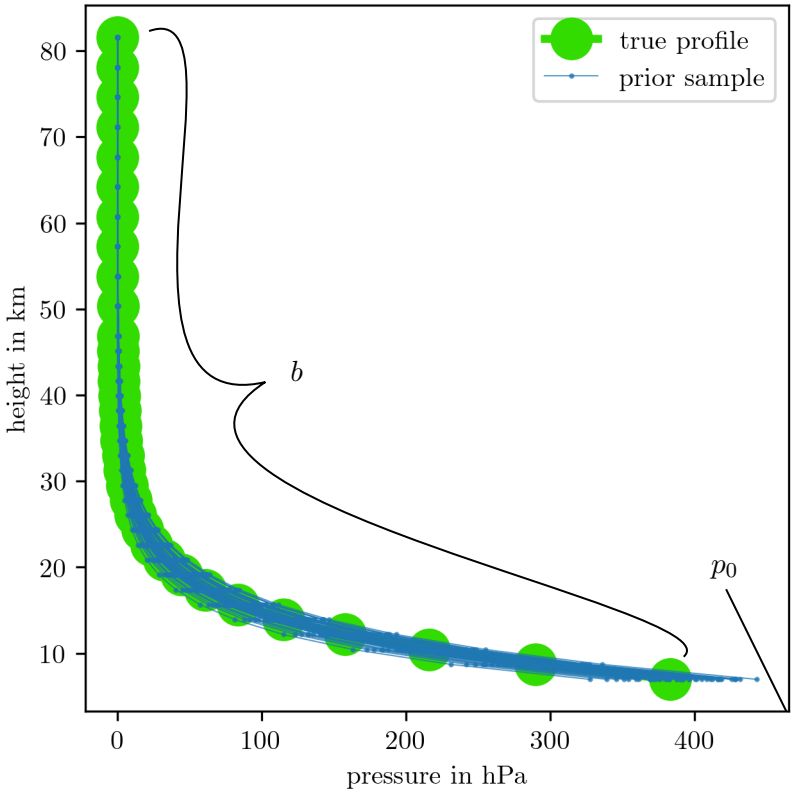


Figure 6.3: Prior samples from the hyper-prior distribution of b and p_0 as defined in Tab. 6.1, where we calculate $\boldsymbol{p}$ according to the function in Eq. 6.3.

samples of $1/\boldsymbol{T}$ in Fig. B.8. Here we already observe that $\boldsymbol{p}/\boldsymbol{T}$ inherits the structure of the pressure function and hence the model is uninformative about the temperature.

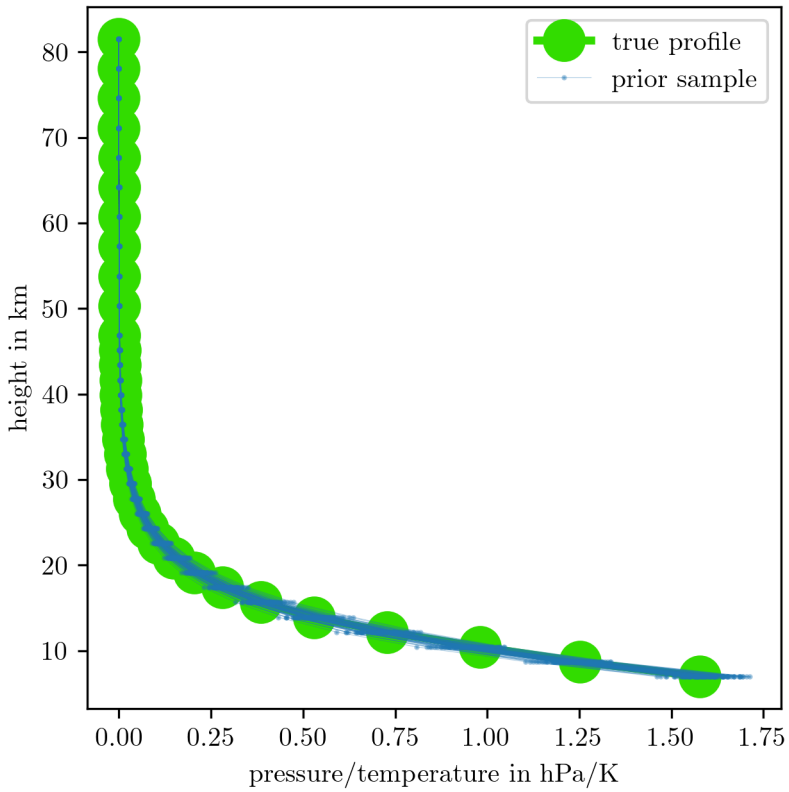


Figure 6.4: Prior samples from the hyper-prior distribution of $\boldsymbol{h}_T$, $\boldsymbol{a}$ and T_0 for temperature as in Eq. 3.9 and b and p_0 for pressure as in Eq. 6.3. We plot $\boldsymbol{p}/\boldsymbol{T}$. The hyper-priors are defined in Tab. 6.1.

6.2 Posterior Distribution

Here, we define the marginal and then the full conditional posterior distribution for the described Bayesian model. We either use the t-walk algorithm [6] to draw samples from $\pi(p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a}, \lambda, \gamma|\boldsymbol{y})$ or we utilise a TT approximation on a predefined grid to generate samples via the SIRT method with an MH correction step. In doing so, we guide the reader through the procedure and point out some key aspects of how we obtain an efficient TT approximation. Lastly, we use the RTO method to draw ozone samples from the full conditional posterior $\pi(\boldsymbol{x}|p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a}, \lambda, \gamma, \boldsymbol{y})$. Recall that the linear forward model matrix is $\boldsymbol{A_\theta}$ is depending on the hyper-parameter defined as $\boldsymbol{\theta} \coloneqq \{p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a}\}$.

6.2.1 Marginal Posterior – Pressure and Temperature

The marginal posterior is given as

$$\pi(\boldsymbol{\theta}, \lambda, \gamma|\boldsymbol{y}) \propto \lambda^{n/2}\gamma^{m/2} \exp\Big\{-\frac{1}{2}g(\boldsymbol{\theta}, \lambda) - \frac{\gamma}{2}f(\boldsymbol{\theta}, \lambda)\Big\}\pi(\boldsymbol{\theta}, \lambda, \gamma)\,, \tag{6.4}$$

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

$$f(\boldsymbol{\theta}, \lambda) = \boldsymbol{y}^T\boldsymbol{y} - (\boldsymbol{A}_{\boldsymbol{\theta}}^T\boldsymbol{y})^T(\boldsymbol{A}_{\boldsymbol{\theta}}^T\boldsymbol{A}_{\boldsymbol{\theta}} + \lambda\boldsymbol{L})^{-1}(\boldsymbol{A}_{\boldsymbol{\theta}}^T\boldsymbol{y})\,, \tag{6.5a}$$

$$\text{and } g(\boldsymbol{\theta}, \lambda) = \log\det(\boldsymbol{A}_{\boldsymbol{\theta}}^T\boldsymbol{A}_{\boldsymbol{\theta}} + \lambda\boldsymbol{L})\,. \tag{6.5b}$$

For each evaluation of $\pi(\boldsymbol{\theta}, \lambda, \gamma|\boldsymbol{y})$ we compose $\boldsymbol{A_\theta}$ as in Chapter 3, and calculate f and g directly using the Cholesky decomposition via the Python functions `np.linalg.cholesky` and `scy.linalg.cho_solve`.

Sampling from the marginal posterior

Since the hierarchical model has 18 hyper-parameters, we utilise the t-walk algorithm by Christen and Fox [6] to sample from the marginal posterior $\pi(\boldsymbol{\theta}, \lambda, \gamma|\boldsymbol{y})$, because it is quick-to-implement and easy-to-use. The t-walk chooses between four different types of steps on the target distribution and is employed as a black-box algorithm in default settings, 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 the construction of this algorithm [6].

Running the t-walk [6] algorithm with the objective to generate 1000 independent samples from the marginal posterior provides a ground truth, which we compare the TT approximation to. The maximum IACT provided by twice the value of [65, 27] (see Tab. 6.1 and Fig. B.9 to Fig. B.26) can be bounded by 1100. Then the t-walk takes $N = 1000 \times 1100$ steps with a burn-in period of $N_{\text{burn-in}} = 100 \times 1100$ for 1000 independent samples. We initialise the Python implementation of the t-walk [7] around the hyper-prior

mean values and the mode of $\pi(\lambda, \gamma|\boldsymbol{y})$. For a total number of $N + N_{\text{burn-in}} = 1210000$ steps within hyper-parameter support bounds given by the iteratively defined TT grid (see Tab. 6.1) a time of ≈ 10 mins is taken. The resulting sample histograms are plotted in Fig. 6.8 to Fig. 6.12 and the trace of the samples in Fig. 6.5.

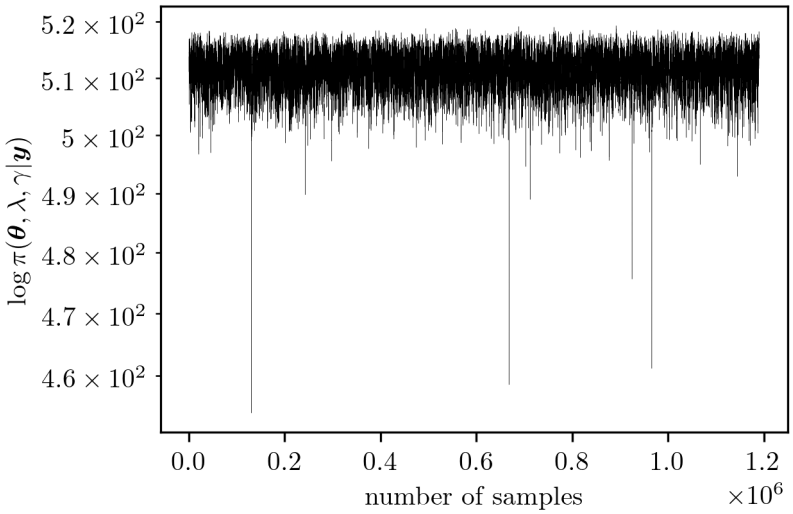


Figure 6.5: Output trace of the marginal posterior distribution $\pi(\boldsymbol{\theta}, \lambda, \gamma|\boldsymbol{y})$ from the t-walk.

TT approximation of marginal posterior

The aim now is to approximate the square root of the marginal posterior

$$\sqrt{\pi(\boldsymbol{\theta}, \lambda, \gamma|\boldsymbol{y})} \propto \exp\Big\{0.5 \log \pi(\boldsymbol{\theta}, \lambda, \gamma|\boldsymbol{y}) + c\Big\}, \tag{6.6}$$

with a "normalisation constant" $c = -200$ to stay within computer precision. In doing so, we run the `tt.cross.rectcross.rect_cross.cross` function from the `ttpy` python package [38] on a grid (see Tab. 6.1) according to the results of the t-walk. When computing the marginal of each hyper-parameter as in Sec. 2.3 the maximum values of $\sqrt{\pi(\boldsymbol{\theta}, \lambda, \gamma|\boldsymbol{y})}$ are around 10^{27} so we set $\xi = 1/\boldsymbol{\lambda}(\mathcal{X})$ with $\boldsymbol{\lambda}(x) = 1$. For Cartesian basis the mass matrix becomes $\boldsymbol{M}_k = \mathrm{diag}(\boldsymbol{\lambda}_k(\mathcal{X}_k))$. To draw samples from the TT approximation the SIRT-MH scheme is used as introduced in Sec. 2.3.2.

Correlation structure First, we order the hyper-parameters according to their correlation structure to improve the efficiency of the TT approximation. Specifically, the hyper-parameter space $\mathcal{X}_\gamma \times \mathcal{X}_\lambda \times \mathcal{X}_b \times \cdots$ is arranged in such a way that highly correlated hyper-parameter pairs are adjacent and directly linked through their shared TT rank. In Fig. 6.6 we plot 1000 independent samples drawn via the SIRT-MH scheme from the TT approximation of $\sqrt{\pi(\boldsymbol{\theta}, \lambda, \gamma|\boldsymbol{y})}$ and the Pearson correlation coefficient between hyper-parameter pairs. A coefficient close to 1 or -1 indicates strong correlation, while values near zero suggest weak or no correlation. We observe that the hyper-parameters λ and b, and λ and γ are highly correlated. Additionally, $h_{T,1}$ describing the temperature at low altitudes (strong signal) is mildly correlated to b. This is because $h_{T,1}$ influences "the smoothness" of $\boldsymbol{p}/\boldsymbol{T}$, which is hard to see in Fig. 6.4. Interestingly, p_0 appears largely uncorrelated with other hyper-parameters, while b is the key parameter linking pressure to ozone and temperature. Hyper-parameters describing temperature at higher altitudes are very much uncorrelated and the IACTs in Tab. 6.1 agree with those results. Alternatively, one could decorrelate the hyper-parameter space, e.g. via Cholesky whitening [29].

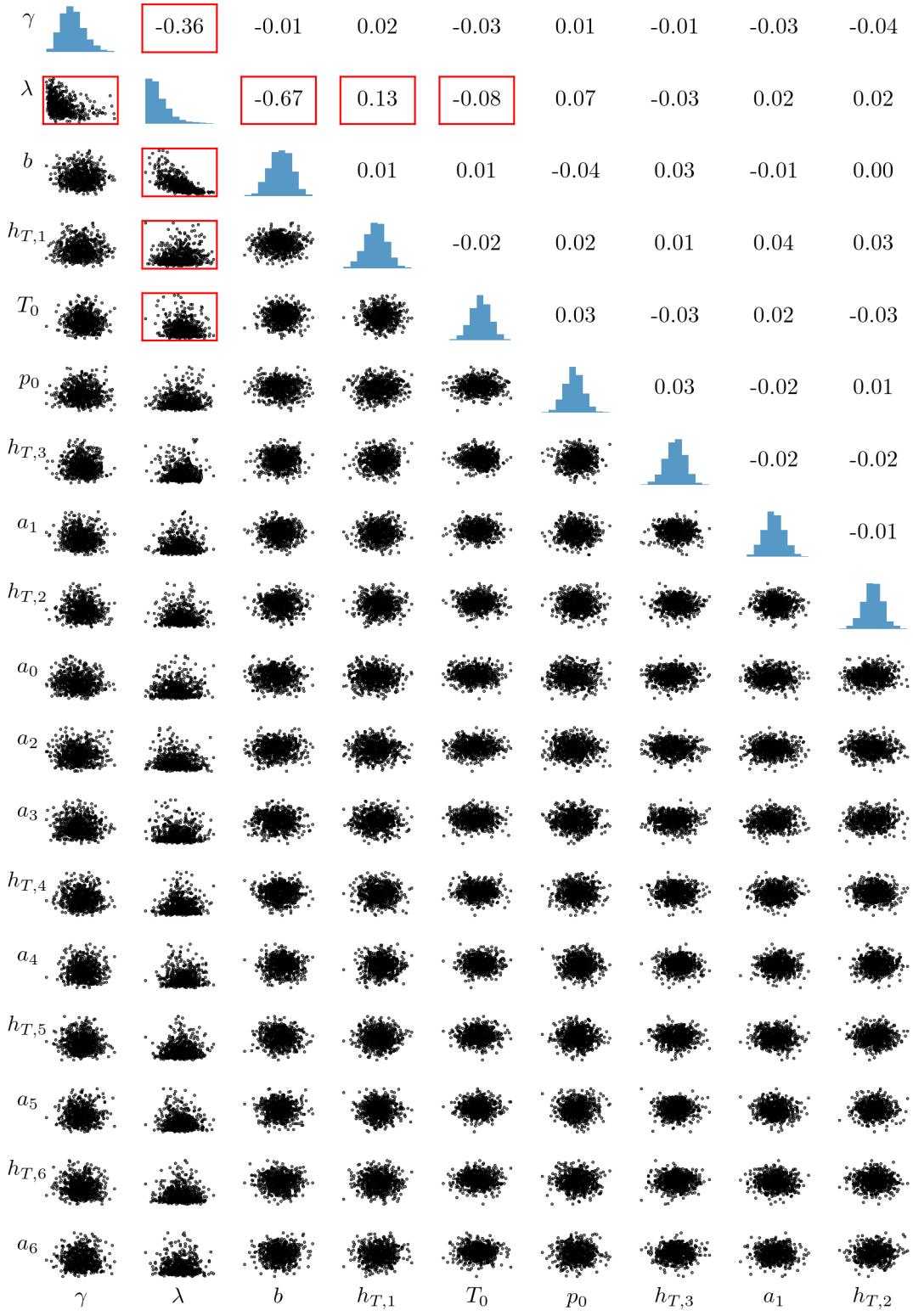
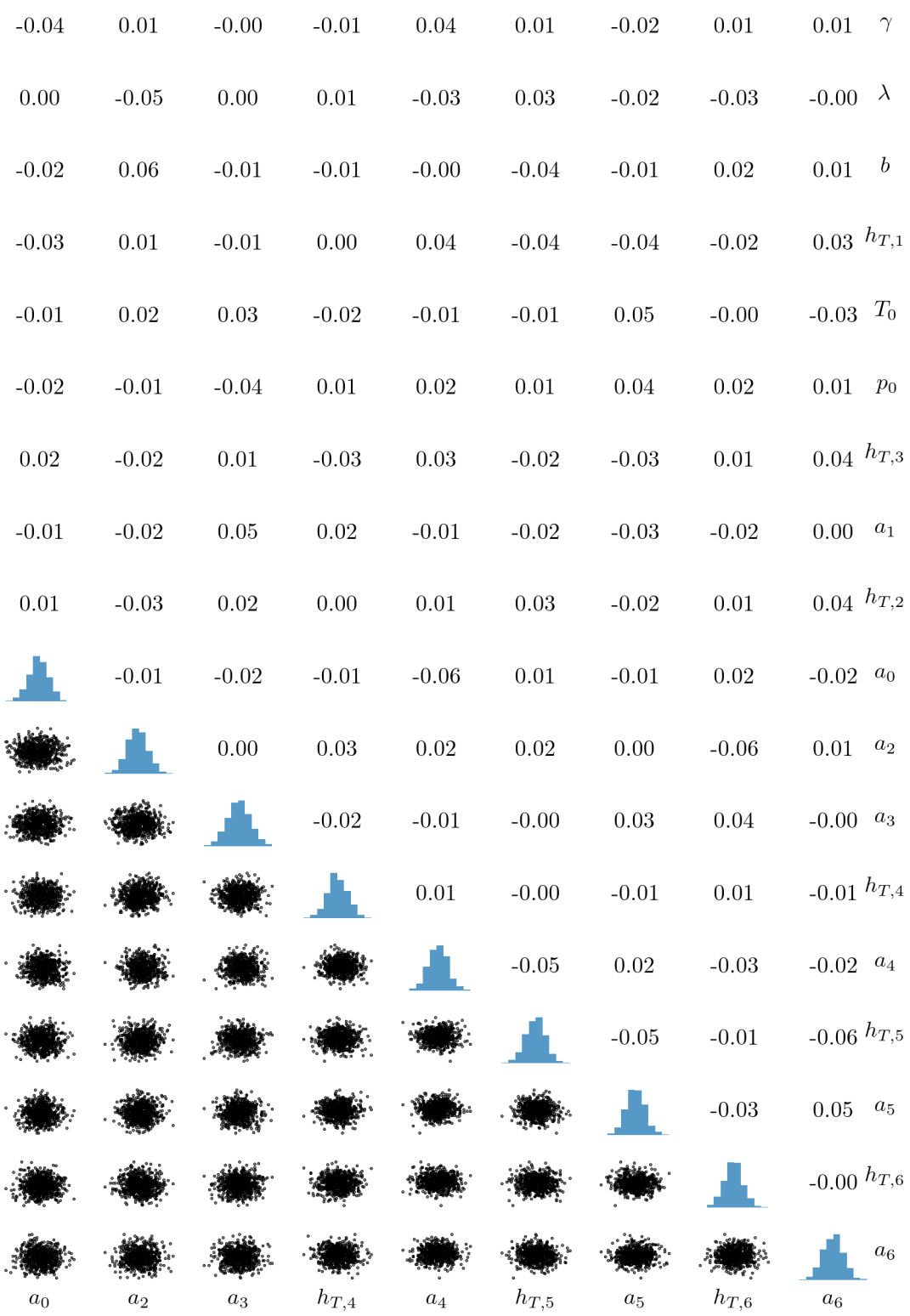


Figure 6.6: Plot of 1000 independent samples from TT approximation of $\sqrt{\pi(\boldsymbol{\theta}, \lambda, \gamma | \boldsymbol{y})}$ via SIRT-MH scheme. We plot the Pearson correlation coefficient ranging from -1 to 1 for each hyper-parameter pair.



Correlation plot of samples from TT-approximation of $\sqrt{\pi(\boldsymbol{\theta}, \lambda, \gamma | \boldsymbol{y})}$ via SIRT-MH scheme.

Find optimal rank and grid size The aim is to decrease the number of function evaluations and to determine the optimal rank and grid size without losing too much accuracy of the marginal posterior approximation. For stable and comparable results, we do five sweeps within

the `tt.cross.rectcross.rect_cross.cross` python function initialised at a random TT, where the ranks between TT cores are constant. Then 1000 independent samples from the TT approximation of the marginal posterior via the SIRT-MH scheme are drawn. First, the number of grid points is set to $n = 150$ and different error measures for ranks $\{2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 20, 25, 30, 35, 50\}$ are calculated. We compare to true marginal posterior function values and 1000 independent t-walk samples to find a small but tolerable rank. Then with a fixed rank the number of grid points $\{10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 80, 90, 100\}$ is decreased until sufficient accuracy.

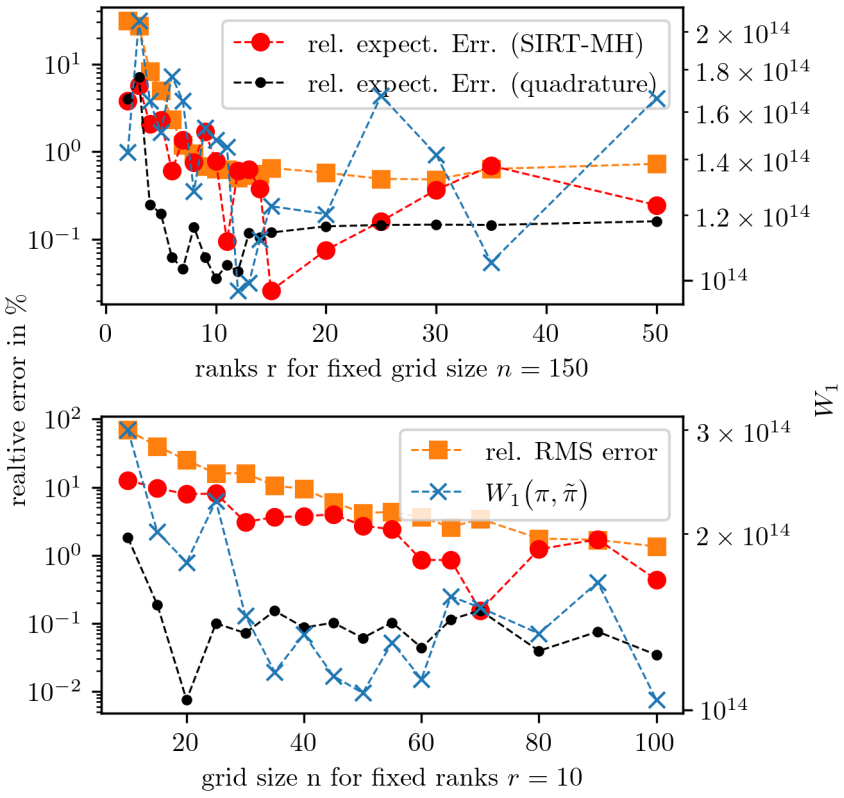


Figure 6.7: Given a TT approximation of $\sqrt{\pi(\boldsymbol{\theta}, \lambda, \gamma|\boldsymbol{y})}$, we calculate the relative RMS error (orange squares) and the 1-Wasserstein distance (blue cross) between approximated values at sample points provided by the SIRT-MH and the true function values. We calculate the relative RMS error between the sample mean provided by the t-walk and mean values for the hyperparameters calculated by quadrature (black dots), where we use the marginal function from the TT approximation as weights. Additionally, we plot the relative RMS error between the sample-based mean from the SIRT-MH and the t-walk (red circles).

One of the error measures is the 1-Wasserstein distance W_1 (blue crosses in Fig. 6.7), as in Eq. 2.48. The 1-Wasserstein distance is calculated between the SIRT-MH samples $\tilde{\boldsymbol{x}} \sim \tilde{\pi}$ weighted with the TT approximation $\tilde{\pi}$ of marginal posterior and the t-walk samples $\boldsymbol{x} \sim \pi$ weighted by π, the true marginal posterior values. We use the `SamplesLoss("sinkhorn", p=1, blur=0.05, scaling=0.8)` function with default settings from the Python package `geomloss` [16] to obtain W_1. This function provides the unbiased Sinkhorn divergence, which converges towards the Wasserstein distance and can be understood as the generalised Quicksort algorithm [15]. Here p = 1 defines the distance measure $\|\boldsymbol{x} - \tilde{\boldsymbol{x}}\|_{L^2}$, the blur parameter is an entropic penalty and the scaling parameter specifies the trade-off between speed (scaling < 0.4) and accuracy (scaling > 0.9) [16]. Additionally, the marginal functions of each TT approximation are used to calculate the quadrature-based means $\boldsymbol{\mu}_{\mathrm{TT}} \in \mathbb{R}^{18}$ of each hyper-parameter by weighted expectations, and to obtain the relative RMS difference $\|\boldsymbol{\mu}_{\mathrm{TT}} - \boldsymbol{\mu}_{\text{t-walk}}\|_{L^2} / \|\boldsymbol{\mu}_{\text{t-walk}}\|_{L^2}$ (black dots in Fig. 6.7). The "true sample-based means" from the t-walk are denoted as $\boldsymbol{\mu}_{\text{t-walk}}$. Further, we calculate the relative RMS between the SIRT-MH sample-based mean $\boldsymbol{\mu}_{\text{SIRT-MH}}$ and $\boldsymbol{\mu}_{\text{t-walk}}$ (red circles in Fig. 6.7), and the relative RMS error at SIRT-MH samples compared to ground truth function values (orange squares in Fig. 6.7).

We plot all of these measures in Fig. 6.7 and observe that a rank $r = 10$ is sufficient because the error measures are relatively stable for $r \geq 10$. For a grid size $n \geq 30$ the relative differences of sample-based means to $\boldsymbol{\mu}_{\text{t-walk}}$ (red circles in Fig. 6.7) and the RMS error at the SIRT-MH samples (orange squares in Fig. 6.7) are around 10% and considered good enough. For an increasing number of grid points the interpolation of function values between grid points is more accurate and the sample-based relative errors decrease, since the chosen linear interpolation (see Eq. 2.41) is a rather rudimentary choice. The quadrature-based relative expectation error (black dots in Fig. 6.7) is almost constant for ranks $\gtrsim 7$ and grid sizes > 20. Since the hyper-parameters have different length scales, we are only interested in the trend of the sample-based 1-Wasserstein distance (blue crosses in Fig. 6.7). The 1-Wasserstein distance is quite fluctuant but decreases with increasing ranks and stays within a similar range for grid sizes $n \geq 30$.

Further, we decrease the number of functions evaluations and define ranks $r = [1, 10, 10, 10, 10, 10, 5, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 1]$ between TT cores harvesting the correlation structure of $\pi(\boldsymbol{\theta}, \lambda, \gamma | \boldsymbol{y})$ even more. One sweep in the `tt.cross.rectcross.rect_cross.cross` initialised at a previously calculated approximation reduces the computation time to $\approx$ 7s and the number of function evaluations to 34080. An average IACT (provided by [65, 27]) of $\approx 1.2 \pm 0.2$ for the samples drawn via the SIRT-MH scheme is calculated. This means that once the TT approximation is available two function evaluations per independent sample are needed. To draw 1000

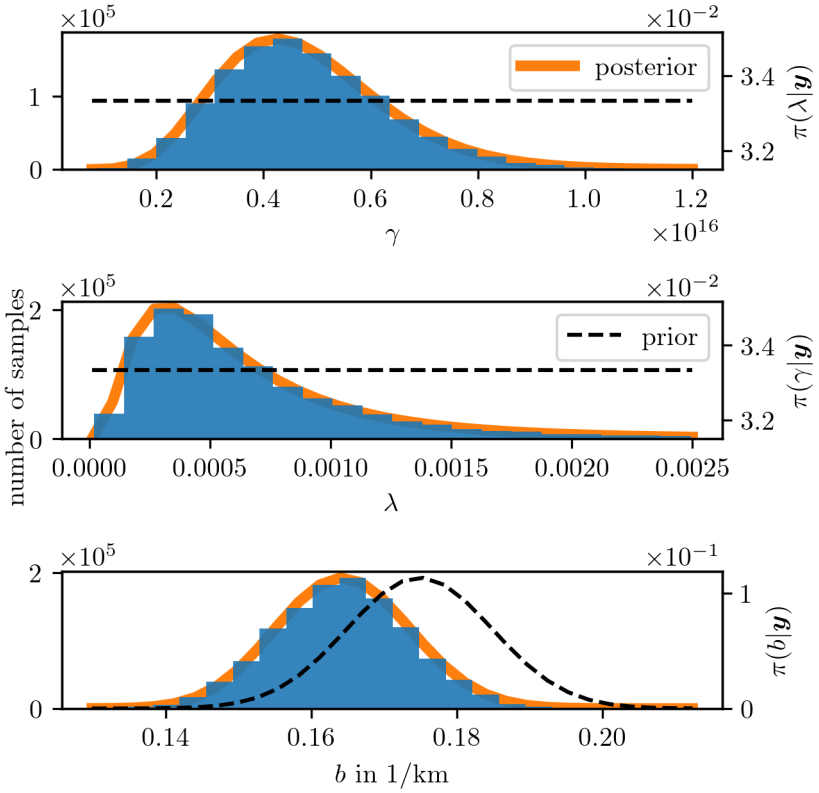


Figure 6.8: TT approximation of the marginal posterior in orange and the samples as a histogram, as well as the prior distribution with a dotted line.

independent samples, including generating a TT approximation, takes ≈ 30s, and we report a relative RMS error of ≈ 12% evaluated over those 1000 independent samples. The relative RMS error over 1000 randomly chosen grid points is ≤ 1% indicating that the linear interpolation causes most of the approximation error. The marginals for each hyper-parameter are plotted in Fig. 6.8 to Fig. 6.13 and the samples in Fig. 6.6. We observe that, besides λ and γ, only the marginal posterior of the b hyper-parameter is seriously affected by the data and has significantly changed compared to the hyper-prior distribution.

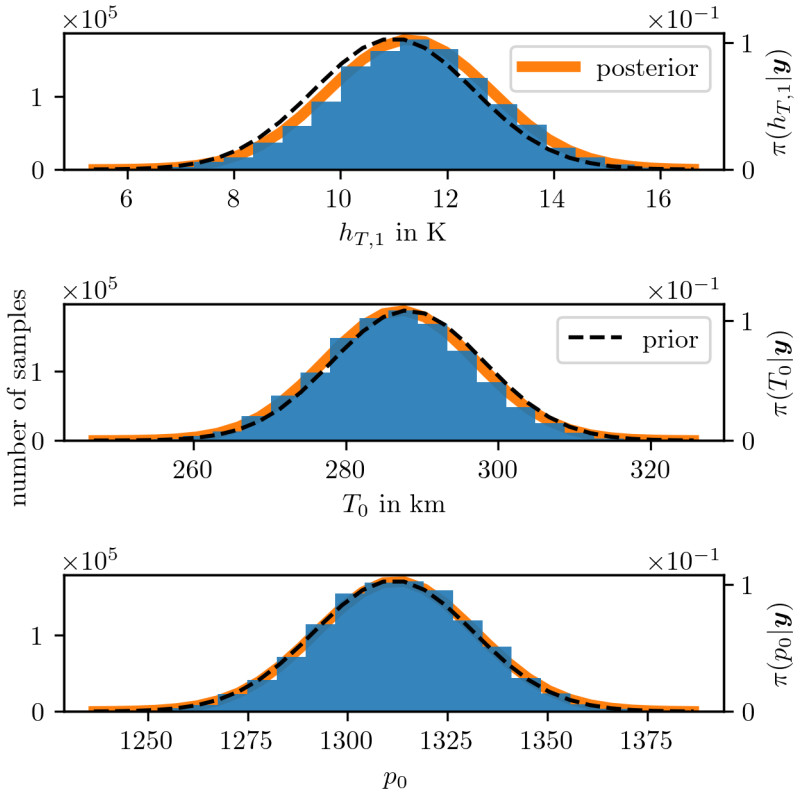


Figure 6.9: TT approximation of the marginal posterior in orange and the samples as a histogram, as well as the prior distribution with a dotted line.

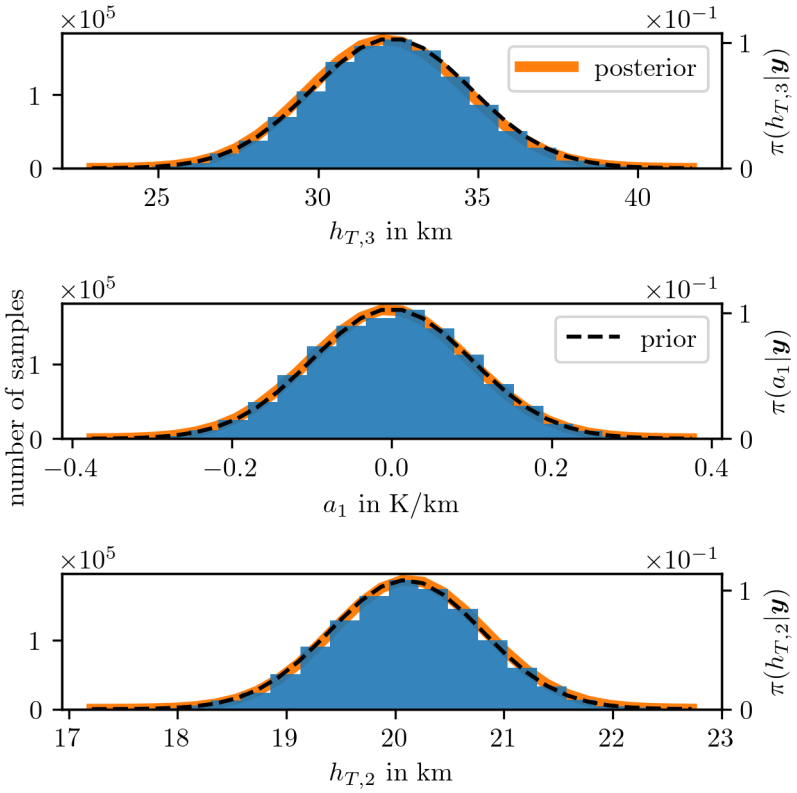


Figure 6.10: TT approximation of the marginal posterior in orange and the samples as a histogram, as well as the prior distribution with a dotted line.

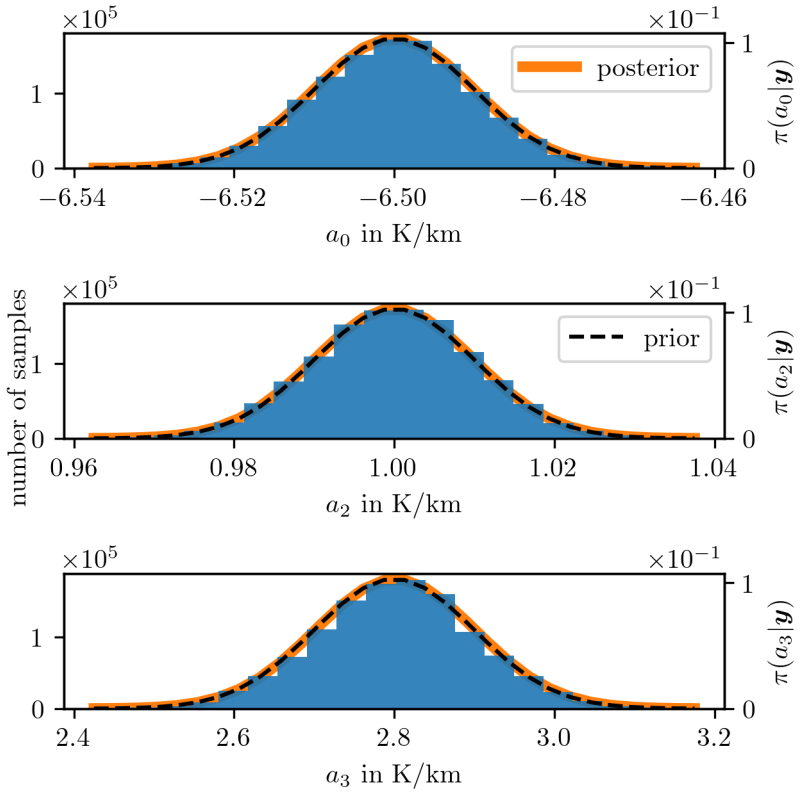


Figure 6.11: TT approximation of the marginal posterior in orange and the samples as a histogram, as well as the prior distribution with a dotted line.

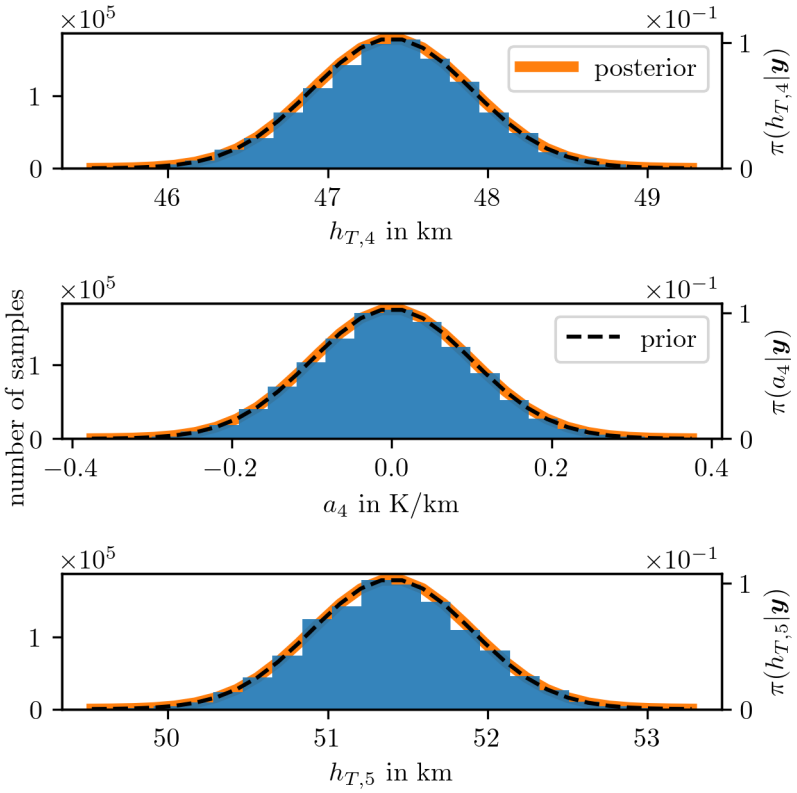


Figure 6.12: TT approximation of the marginal posterior in orange and the samples as a histogram, as well as the prior distribution with a dotted line.

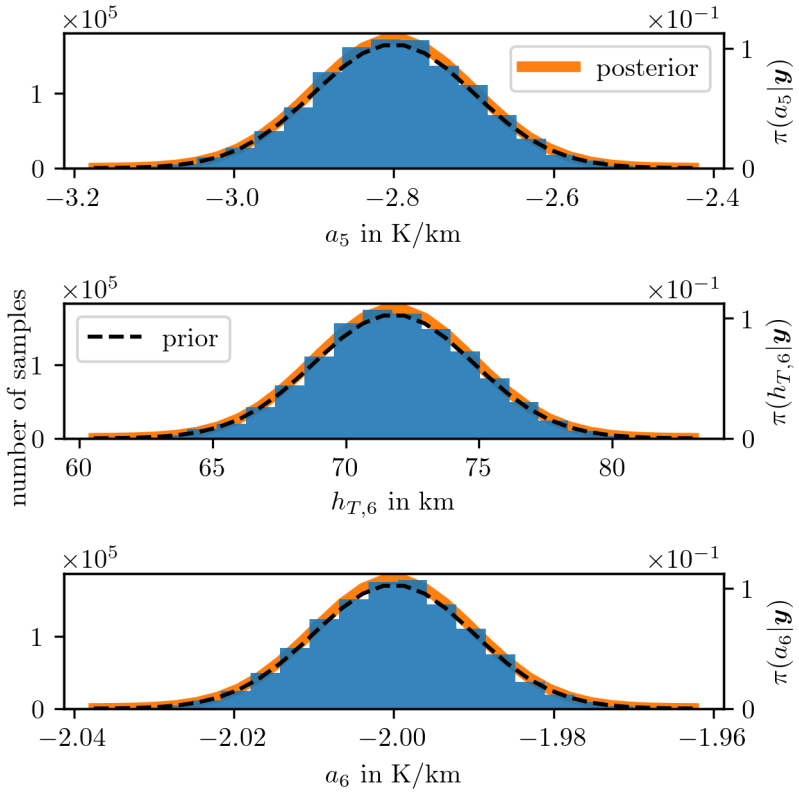


Figure 6.13: TT approximation of the marginal posterior in orange and the samples as a histogram, as well as the prior distribution with a dotted line.

Posterior pressure and temperature

Posterior pressure and temperature profiles are directly obtained by samples from the marginal posterior $\pi(\boldsymbol{\theta}, \lambda, \gamma|\boldsymbol{y})$ and according to their respective function (see Eq. 3.9 and Eq. 6.3). We plot posterior temperature profiles in Fig. 6.14 and pressure profiles in Fig. 6.15. The posterior temperature profiles look (as expected) similar to the prior temperature profiles, whereas the pressure profile has slightly larger values compared to the ground truth. This is because the hyper-parameter b is smaller than its ground truth value (see Fig. 6.8), resulting in the posterior pressure profiles that do not exponentially decrease as fast as the ground truth pressure profile.

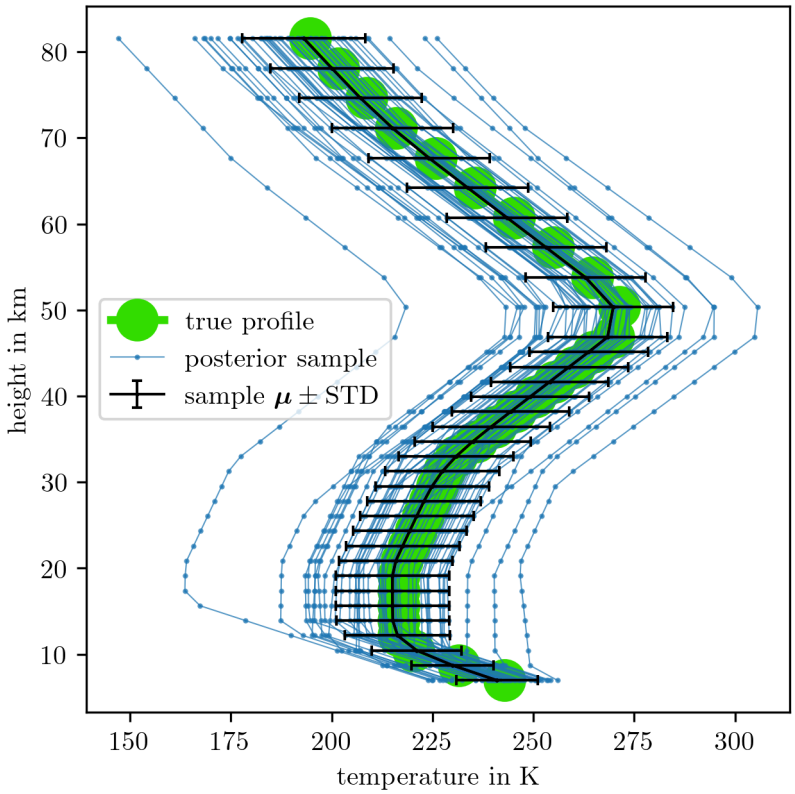


Figure 6.14: According to the hyper-parameter samples from the marginal posterior distribution $\pi(\boldsymbol{\theta}, \lambda, \gamma|\boldsymbol{y})$ we plot the corresponding posterior temperature profile as given by Eq. 3.9.

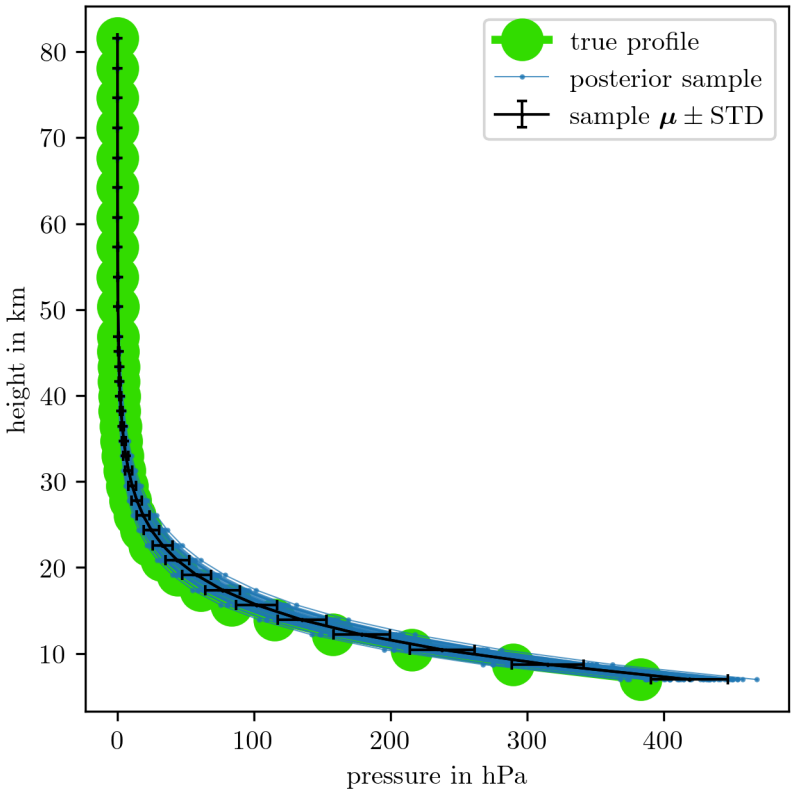


Figure 6.15: According to the hyper-parameter samples from the marginal posterior distribution $\pi(\boldsymbol{\theta}, \lambda, \gamma | \boldsymbol{y})$ we plot the corresponding posterior pressure profile as given by Eq. 6.3.

6.2.2 Full Conditional Posterior – Ozone

Due to the large number of hyper-parameters calculating the posterior mean $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}$ and $\boldsymbol{\Sigma}_{\boldsymbol{x}|\boldsymbol{y}}$ covariance via quadrature as in Eq. 2.17 and 2.18 is computationally not feasible. If the full conditional posterior is a normal distribution the randomise then optimise (RTO) method provides a scheme to obtain an independent ozone sample from $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \delta, \gamma, \boldsymbol{y})$ with $\delta = \lambda\gamma$. We introduce the RTO method for general case first and then draw ozone samples from $\boldsymbol{x}^{(k)} \sim \pi(\boldsymbol{x}|\boldsymbol{\theta}^{(k)}, \delta^{(k)}, \gamma^{(k)}, \boldsymbol{y})$ conditioned on independent marginal posterior samples $\boldsymbol{\theta}^{(k)}, \delta^{(k)}, \gamma^{(k)} \sim \pi(\boldsymbol{\theta}, \delta, \gamma|\boldsymbol{y})$.

Randomise then optimise

As in [2] rewrite the full conditional posterior (see Eq. 2.15) as

$$\pi(\boldsymbol{x}|\boldsymbol{\theta}, \delta, \gamma, \boldsymbol{y}) \propto \pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta}, \gamma)\pi(\boldsymbol{x}|\delta) \tag{6.7}$$

$$\propto \exp\left(-\frac{1}{2}(\boldsymbol{A}_{\boldsymbol{\theta}}\boldsymbol{x} - \boldsymbol{y})^T\boldsymbol{\Sigma}_{\gamma}^{-1}(\boldsymbol{A}_{\boldsymbol{\theta}}\boldsymbol{x} - \boldsymbol{y})\right)\exp\left(-\frac{1}{2}(\boldsymbol{\mu} - \boldsymbol{x})^T\boldsymbol{Q}_{\delta}(\boldsymbol{\mu} - \boldsymbol{x})\right), \tag{6.8}$$

$$= \exp\left(-\frac{1}{2}\left\|\hat{\boldsymbol{A}}\boldsymbol{x} - \hat{\boldsymbol{y}}\right\|_{L^2}^2\right), \tag{6.9}$$

where

$$\hat{\boldsymbol{A}} \coloneqq \begin{bmatrix}\boldsymbol{\Sigma}_{\gamma}^{-1/2}\boldsymbol{A}_{\boldsymbol{\theta}} \\ \boldsymbol{Q}_{\delta}^{1/2}\end{bmatrix}, \quad \hat{\boldsymbol{y}} \coloneqq \begin{bmatrix}\boldsymbol{\Sigma}_{\gamma}^{-1/2}\boldsymbol{y} \\ \boldsymbol{Q}_{\delta}^{1/2}\boldsymbol{\mu}\end{bmatrix}, \tag{6.10}$$

$\boldsymbol{Q}_{\delta}$ is the prior precision, $\boldsymbol{\mu}$ the prior mean and $\boldsymbol{\Sigma}_{\gamma}$ the noise covariance (see also [3, 4]). A sample $\boldsymbol{x}^{(k)}$ from the full conditional posterior $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \delta, \gamma, \boldsymbol{y})$ is obtained by minimising the following equation:

$$\boldsymbol{x}^{(k)} = \arg\min_{\boldsymbol{x}} \|\hat{\boldsymbol{A}}\boldsymbol{x} - (\hat{\boldsymbol{y}} + \boldsymbol{b})\|_{L^2}^2, \quad \boldsymbol{b} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_m), \tag{6.11}$$

where a random perturbation $\boldsymbol{b}$ is added. Similar to Section 2.4, this expression becomes

$$\left(\boldsymbol{A}_{\boldsymbol{\theta}}^T\boldsymbol{\Sigma}_{\gamma}^{-1}\boldsymbol{A}_{\boldsymbol{\theta}} + \boldsymbol{Q}_{\delta}\right)\boldsymbol{x}^{(k)} = \boldsymbol{A}_{\boldsymbol{\theta}}^T\boldsymbol{\Sigma}_{\gamma}^{-1}\boldsymbol{y} + \boldsymbol{Q}_{\delta}\boldsymbol{\mu} + \boldsymbol{v}_1 + \boldsymbol{v}_2, \tag{6.12}$$

with $\boldsymbol{v}_1 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{A}_{\boldsymbol{\theta}}^T\boldsymbol{\Sigma}_{\gamma}^{-1}\boldsymbol{A}_{\boldsymbol{\theta}})$ and $\boldsymbol{v}_2 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_{\delta})$, representing independent Gaussian random variables [2, 17].

Posterior ozone samples

More explicitly conditioned on an independent $\boldsymbol{\theta}^{(k)}, \lambda^{(k)}, \gamma^{(k)} \sim \pi(\boldsymbol{\theta}, \lambda, \gamma|\boldsymbol{y})$ and one independent full conditional posterior sample is given as

$$\boldsymbol{x}^{(k)} = \Big(\underbrace{\gamma^{(k)}\boldsymbol{A}_{\boldsymbol{\theta}^{(k)}}^T\boldsymbol{A}_{\boldsymbol{\theta}^{(k)}} + \delta^{(k)}\boldsymbol{L}}_{\boldsymbol{B}^{(k)}}\Big)^{-1}\left(\gamma^{(k)}\boldsymbol{A}_{\boldsymbol{\theta}^{(k)}}^T\boldsymbol{y} + \sqrt{\gamma^{(k)}}\boldsymbol{A}_{\boldsymbol{\theta}^{(k)}}^T\boldsymbol{v}_1 + \sqrt{\delta^{(k)}}\boldsymbol{L}^{1/2}\boldsymbol{v}_2\right) \tag{6.13}$$

with $\boldsymbol{v}_1 \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_m)$, $\boldsymbol{v}_2 \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_n)$, $\boldsymbol{Q}_\delta = \delta \boldsymbol{L}$, $\boldsymbol{\Sigma}_\gamma^{-1} = \gamma \boldsymbol{I}$ and $\boldsymbol{L}^{1/2}$ is the Cholesky factorisation of $\boldsymbol{L}$ [2]. If n is large and calculating the Cholesky decomposition of $\boldsymbol{L}$ or constructing $\boldsymbol{L}$ is expensive it is recommended to represent $\boldsymbol{L}$ as a sum over cliques e.g. small 2×2 rank-1 matrices (see $\boldsymbol{L}$ in Eq. 2.57). Then a sample $\boldsymbol{v}_2 \sim \mathcal{N}(\mathbf{0}, \boldsymbol{Q}_\delta)$ can be obtained by drawing n random variables from $\mathcal{N}(0, 1)$ without explicitly forming $\boldsymbol{L}$ [17].

We obtain the Cholesky factorisation of $\boldsymbol{B}^{(k)}$ and $\boldsymbol{L}$ from the Python function `numpy.linalg.cholesky` and solve for $\boldsymbol{x}^{(k)}$ using `scipy.linalg.cho_solve`. We plot 100 samples in Fig. 6.16 and a sample mean based on 1000 samples. The posterior ozone mean is much smaller than the ground truth especially around the ozone peak. Compared to the posterior pressure in Fig. 6.15, which is slightly larger than the ground truth, we can conclude that pressure and ozone are highly correlated. Additionally, the individual posterior samples are more prior-dominated through larger λ values (see Fig. 6.8) and hence smoother compared to the previously calculated ozone posterior profiles. Again, we are not able to recover the second ozone peak at high altitudes.

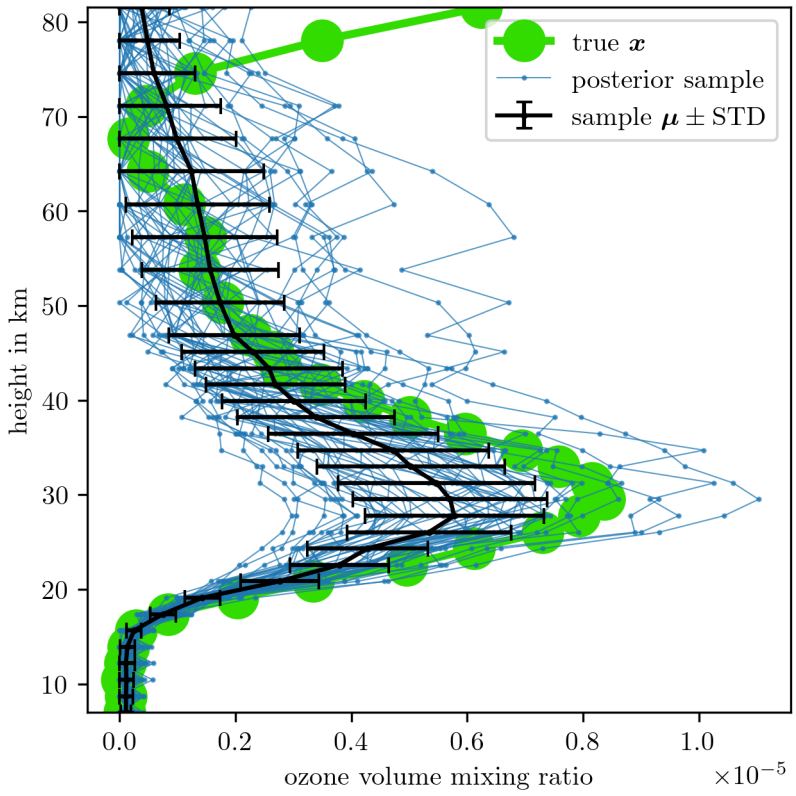


Figure 6.16: Conditioned on the hyper-parameter samples from the marginal posterior distribution $\pi(\boldsymbol{\theta}, \delta, \gamma | \boldsymbol{y})$ we plot the corresponding ozone sample from the full conditional posterior $\pi(\boldsymbol{x} | \boldsymbol{\theta}, \delta, \gamma, \boldsymbol{y})$ using the RTO method.

7 Summary and Outlook

In this chapter, we summarise the key results and conclusions of our work and provide an outlook for future research. We compare the Bayesian approach to a regularisation approach and elaborate on the differences between sampling-based methods and the TT approximation. Lastly, we situate our results within the broader context of atmospheric modelling and discuss the implications for the future development of an atmospheric limb sounder.

7.1 Regularisation Solution vs. Hierarchical Bayesian Approach

Using a regularisation approach, we need 200 solves of $\boldsymbol{x}_\lambda$ to obtain one solution of this inverse problem. In contrast, the hierarchical Bayesian approach involves 25 function evaluations of the marginal posterior (to find the mode) and then 10100 samples from the approximated marginal posterior followed by 20 evaluations of $\boldsymbol{x}_\lambda$ and $\boldsymbol{B}_\lambda^{-1}$ to characterise the full posterior in ≈ 0.5s. Utilising a TT approximation (including finding the mode) to compute the full posterior mean and covariance takes ≈ 0.025s, requiring only 400 function evaluations to approximate $\pi(\lambda, \gamma|\boldsymbol{x})$ and is almost as fast as the regularisation approach (≈ 0.015). Regardless, either method has a runtime of much less than a second on a basic laptop.

While regularisation yields a single optimal solution (point estimate), a Bayesian framework provides a distribution of ozone profiles, which are all feasible solutions to the inverse problem, and hence true errors. Moreover, within the hierarchical Bayesian approach, we can include prior knowledge about the noise, ozone profile and many more physical processes through hyper-parameters, offering an arbitrarily flexible and informative inference framework.

7.2 Sampling Methods vs. TT Approximation

Using the TT approximation involves far fewer function evaluations of the target distribution compared to sample-based methods, but requires a predefined grid and a normalisation constant, which, for now, we have to find iteratively. Relying solely on TT approximations may lead to a substantial amount of trial and error and dealing with numerical issues. Nevertheless, once properly configured, we have shown the potential and advantages of TT methods.

More specifically, the TT approximation of the 2-dimensional marginal posterior (≈ 0.02s) is more than 20 times faster than the MWG sampler (≈ 0.5s). Excluding the function evaluations for finding the mode of the marginal posterior, the MWG sampler takes 10100 steps while the TT approximation only needs 400 function evaluations; this is a factor of ≈ 25. Alternatively, for low-dimensional distributions, it may be preferable to approximate integrals directly using existing freely available quadrature libraries and packages such as `quadpy`.

In higher dimensions, such as the 18-dimensional marginal posterior considered in this thesis, TT methods (≈ 0.5min) outperform samplers like the t-walk (≈ 10min), once a grid and normalisation constant have been defined. Although the t-walk may not be the best sampler for this specific marginal posterior and the underlying correlation structure, it is robust and easy to implement. To illustrate the efficiency of TT approximations, we compare the number of function evaluations per 1000 independent samples. For 1000 independent samples with a maximum IACT of 1100 and a burn-in period of 100 independent samples, the t-walk needs 1210000 function evaluations. In contrast, 34080 function evaluations are enough to approximate the marginal posterior in the TT format. Then drawing 1000 independent samples via the SIRT-MH scheme requires another 2000 function evaluations with an IACT of ≈ 1.2. So the cost per independent sample for the t-walk is 1210 and for the TT approach is ≈ 36 function evaluations, including the burn-in period and the TT approximation via the `rect_cross.cross` Python function. Or, after the burn-in phase, the t-walk requires around 1100 function evaluations per independent sample, while with an approximation of a probability density in the TT format available, only two function evaluations per independent sample are needed.

For future application, we suggest improving the efficiency of the TT approximation by, e.g., reducing the correlation structure through a coordinate system rotation or using better interpolators in between grid points to reduce the approximation error. This may be particularly important when the CDF in the SIRT scheme is not smooth due to poor approximations of the target density at previous samples. Moreover, using a different reference measure for integration as in [8], such as a Gaussian measure instead

of the current Lebesgue measure, may increase numerical stability. Currently, we have to predefine a normalisation constant and lower ranks manually, bounding the ranks automatically would be helpful (see e.g. [46]).

7.3 Atmospheric Physics

Here we summarise results within the context of our simplified physical atmospheric limb-sounding model. We demonstrated that the underlying non-linear forward model can be approximated with an affine map and the linear model, making this a linear inverse problem. For future application, we wish to include more measurement device-specific hyper-parameters in the forward model. This could include e.g. uncertainty in pointing accuracy or an antenna response function.

In Sec. 3.2.1, we showed that we do not gain more information if we measure more frequently or collect more data in noise-dominated regions and that we need an SNR of $\approx 10^4$ to produce data, which is informative about ozone at higher altitudes.

Fig. 6.16 and Fig. 6.15 illustrate that pressure and ozone are highly correlated. One has to consider that when conditioning on pressure estimates from other measurements, as a slight change in pressure does skew the ozone VMR significantly. We could fix that by choosing a more restrictive prior for the pressure-related hyper-parameter b, but that would not be objective. By explanatory analysis, we found that data with an SNR of ≈ 1000 recovers an ozone (without a peak in higher altitudes) and pressure profile close to the ground truth. As previously mentioned in the prior analysis (see Fig. 6.4), the model as well as the data are uninformative about temperature and dominated by the exponentially decreasing pressure.

All the samples plotted in Fig. 5.4, Fig. 4.5, and Fig. 6.16 present valid solutions to the inverse problem, but consistently fail to capture the ozone peak at higher altitudes. This is due to noise-dominated data (see Fig. 3.7) and low signal strength in upper atmospheric regions, where the variability of the posterior ozone is large and primarily determined by the prior. We conclude that the main objective for future research is to develop a more accurate, potentially parametrised (prior) model, which captures physical properties and chemical processes of ozone in the atmosphere.

References

[1] Ambrosio, L., Brué, E., and Semola, D. *Lectures on Optimal Transport*. Cham: Springer Nature Switzerland, 2024.

[2] Bardsley, J. "MCMC-based image reconstruction with uncertainty quantification". In: *SIAM Journal on Scientific Computing* 34.3 (2012), A1316–A1332.

[3] Bardsley, J., Solonen, A., Haario, H., and Laine, M. "Randomize-then-optimize: A method for sampling from posterior distributions in nonlinear inverse problems". In: *SIAM Journal on Scientific Computing* 36.4 (2014), A1895–A1910.

[4] Bardsley, J. and Cui, T. "A Metropolis-Hastings-Within-Gibbs Sampler for Nonlinear Hierarchical-Bayesian Inverse Problems". In: *2017 MATRIX Annals*. Vol. 2. MATRIX Book Series. Switzerland: Springer, 2019, pp. 2–12.

[5] Champ, C. W. and Sills, A. V. "The Generalized Law of Total Covariance". In: *preprint* (2022).

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

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

[8] Cui, T. and Dolgov, S. "Deep composition of tensor-trains using squared inverse rosenblatt transports". In: *Foundations of Computational Mathematics* 22.6 (2022), pp. 1863–1922.

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

[10] Dick, J., Kuo, F. Y., and Sloan, I. H. "High-dimensional integration: The quasi-Monte Carlo way". In: *Acta Numerica* 22 (2013), 133–288.

[11] Dolgov, S., Anaya-Izquierdo, K., Fox, C., and Scheichl, R. "Approximation and sampling of multivariate probability distributions in the tensor train decomposition". In: *Statistics and Computing* 30 (2020), pp. 603–625.

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

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

[14] Facility, A. N. C. D. *CubeSat Microwave Radiometer Mission to Support Global Ozone Layer Monitoring. Concept Study - Summary Report*. unpublished, internal report. Canberra BC: UNSW Canberra Space, 2023.

[15] Feydy, J. "Analyse de données géométriques, au delà des convolutions". Ph.D. Thesis. Université Paris-Saclay, July 2020.

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

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

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

[19] Fox, C. "Conductance Imaging. Estimation of Isotropic Conductance Perturbations from Low-Frequency Boundary Measurements in Circular Geometries". Ph.D. Thesis. University of Cambridge, Oct. 1988.

[20] Fox, C., Dolgov, S., Morrison, M. E., and Molteno, T. C. "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.

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

[22] Geyer, C. J. "Practical markov chain monte carlo". In: *Statistical science* (1992), pp. 473–483.

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

[24] Hansen, P. C. *Discrete Inverse Problems: Insight and Algorithms.* Philadelphia: SIAM, 2010.

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

[26] Hansen, P. C. and O'Leary, D. P. "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.

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

[28] Kaipio, J. P. and Somersalo, E. *Statistical and Computational Inverse Problems.* New York: Springer-Verlag New York, 2005.

[29] Kessy, A., Lewin, A., and Strimmer, K. "Optimal Whitening and Decorrelation". In: *The American Statistician* 72.4 (Jan. 2018), 309–314. ISSN: 1537-2731.

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

[31] Li, B., Miao, L., Zhang, C., and Yang, W. "A Lagrange Multiplier-based Regularization Algorithm for Image Super-resolution". In: *2018 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM).* 2018, pp. 422–426.

[32] Livesey, N. J., Van Snyder, W, Read, W. G., and Wagner, P. A. "Retrieval algorithms for the EOS Microwave limb sounder (MLS)". In: *IEEE Transactions on Geoscience and Remote Sensing* 44.5 (2006), pp. 1144–1155.

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

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

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

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

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

[38] Oseledets, I., Bershatsky, D., and Saluev, T. *ttpy - a Python implementation of the TT-toolbox.* `https://github.com/oseledets/ttpy`. [Online; accessed 23/06/25]. 2018.

[39] Oseledets, I. and Tyrtyshnikov, E. "TT-cross approximation for multidimensional arrays". In: *Linear Algebra and its Applications* 432.1 (2010), pp. 70–88.

[40] Read, W., Shippony, Z., Schwartz, M., Livesey, N. J., and Van Snyder, W. "The clear-sky unpolarized forward model for the EOS aura microwave limb sounder (MLS)". In: *IEEE Transactions on Geoscience and Remote Sensing* 44.5 (2006), pp. 1367–1379.

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

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

[43] Roberts, G. O. and Rosenthal, J. S. "General state space Markov chains and MCMC algorithms". In: *Probability Surveys* 1 (2004), pp. 20–71.

[44] Roberts, G. O. and Rosenthal, J. S. "Harris recurrence of Metropolis-within-Gibbs and trans-dimensional Markov chains". In: *The Annals of Applied Probability* 16.4 (2006), 2123–2139.

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

[46] Rohrbach, P. B., Dolgov, S., Grasedyck, L., and Scheichl, R. "Rank Bounds for Approximating Gaussian Densities in the Tensor-Train Format". In: *SIAM/ASA Journal on Uncertainty Quantification* 10.3 (2022), pp. 1191–1224.

[47] Rue, H. and Held, L. *Gaussian Markov random fields: theory and applications*. London: CRC press, 2005.

[48] Rybicki, G. B. and Lightman, A. P. *Radiative processes in Astrophysics*. Weinheim: Wiley-VCH, 2004.

[49] Santosh, K., Das, N., and Ghosh, S. "Chapter 3 - Deep learning models". In: *Deep Learning Models for Medical Imaging*. Primers in Biomedical Imaging Devices and Systems. Academic Press, 2022, pp. 65–97.

[50] Satopää, V., Albrecht, J., Irwin, D., and Raghavan, B. "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.

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

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

[53] Šimečková, M., Jacquemart, D., Rothman, L. S., Gamache, R. R., and Goldman, A. "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.

[54] Simpson, D., Lindgren, F., and Rue, H. "Think continuous: Markovian Gaussian models in spatial statistics". In: *Spatial Statistics* 1 (2012), pp. 16–29.

[55] Sokal, A. "Monte Carlo Methods in Statistical Mechanics: Foundations and New Algorithms". In: *Functional Integration: Basics and Applications*. Boston, MA: Springer US, 1997, pp. 131–192.

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

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

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

[59] *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.

[60] Ustin, S. and Middleton, E. M. "Current and near-term Earth-observing environmental satellites, their missions, characteristics, instruments, and applications". In: *Sensors* 24.11 (2024), p. 3488.

[61] Vats, D., Acosta, F., Huber, M. L., and Jones, G. L. "Understanding Linchpin Variables in Markov Chain Monte Carlo". In: *preprint* (2022).

[62] Wang, Y.-X., Sharpnack, J., Smola, A. J., and Tibshirani, R. J. "Trend Filtering on Graphs". In: *Journal of Machine Learning Research* 17.105 (2016), pp. 1–41.

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

[64] Watzenig, D and Fox, C. "A review of statistical modelling and inference for electrical capacitance tomography". In: *Measurement Science and Technology* 20.5 (2009), p. 052002.

[65] Wolff, U. "Monte Carlo errors with less errors". In: *Computer Physics Communications* 156.2 (2004), pp. 143–153.

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

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

Appendices

A

Theoretical and Technical Background

A.1 Correlation Structure

In the book Gaussian Markov Random Fields [47], Rue and Held demonstrate that a strong correlation between the hyper-parameter μ and the latent field $\boldsymbol{x}$ can significantly slow down convergence particularly when using Gibbs samplers. They consider the hierarchical model

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

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

and apply a Gibbs sampler based on the full conditional distributions

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

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

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

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

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

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

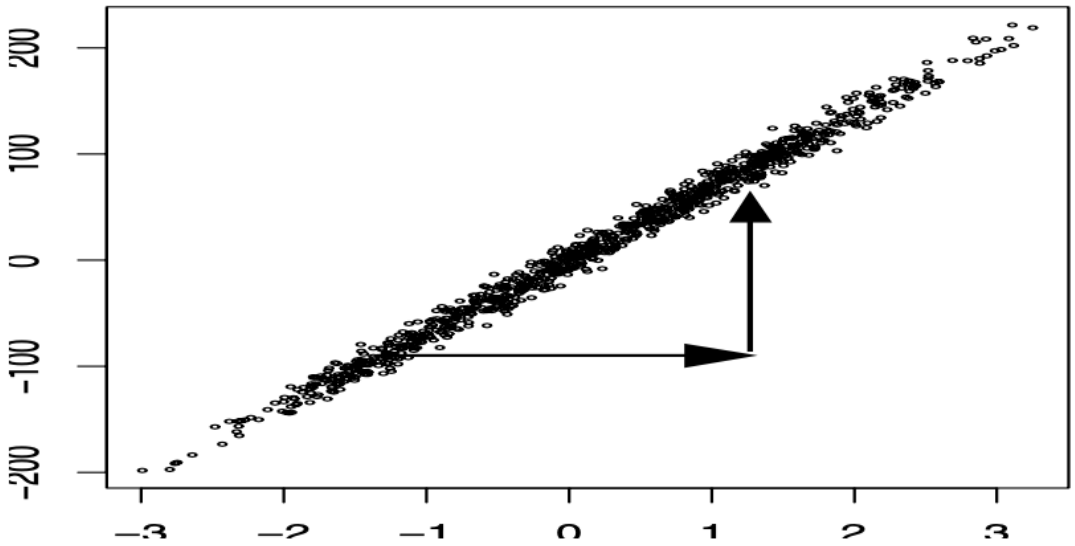


Figure A.1: The figure taken from [47, Figure 4.1 (b)] shows samples from the chain of μ (x-axis) and $\mathbf{1}^T \boldsymbol{Q}\boldsymbol{x}^{(k)}$ (y-axis) for over 1000 iterations, based on the hierarchical model in Eq. A.1, with an autoregressive process encoded in $\boldsymbol{Q}$. The algorithm updates μ and $\boldsymbol{x}$ successively from their conditional distributions (see Eq. A.2 and Eq. A.3). The plot displays $(\mu^{(k)}, \mathbf{1}^T \boldsymbol{Q}\boldsymbol{x}^{(k)})$, with $\mu^{(k)}$ on the horizontal axis and $\mathbf{1}^T \boldsymbol{Q}\boldsymbol{x}^{(k)}$ on the vertical axis. The slow mixing and convergence of μ result from its strong dependence on $\mathbf{1}^T \boldsymbol{Q}\boldsymbol{x}^{(k)}$, while the sampler permits only axis-aligned (horizontal and vertical) and does not allow diagonal moves, as illustrated by the arrows.

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

A.2 Monte-Carlo Error and Integrated Autocorrelation Time

To assess the error $(\sigma^{(i)})^2$ of a samples-based estimate

$$\bar{h}_N^{(i)} := \mathrm{E}_{\boldsymbol{x}|\boldsymbol{y}}[h(\boldsymbol{x})] = \frac{1}{N}\sum_{k=1}^{N} h(\boldsymbol{x}^{(k)}), \tag{A.6}$$

from the chain $\mathcal{M}^{(i)} = \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(k)}, \ldots, \boldsymbol{x}^{(s)}, \ldots, \boldsymbol{x}^{(N)}\} \sim \pi(\boldsymbol{x}|\boldsymbol{y})$, we ignore systematic error due to initialisation bias (burn-in period), but we have to take into account that samples produced by any system or algorithm are correlated. To derive the IACT, we follow Ulli Wolff's lecture notes [67] (or alternatively [65]).

In general, the error of a Monte-Carlo estimate is:

$$(\sigma^{(i)})^2 = \mathrm{Var}(\bar{h}_N^{(i)}) = \mathrm{Var}(\mathrm{E}_{\boldsymbol{x}|\boldsymbol{y}}[h(\boldsymbol{x})]) = \left(\frac{1}{N}\sum_{k=1}^{N} h(\boldsymbol{x}^{(k)}) - \bar{h}_N^{(i)}\right)^2. \tag{A.7}$$

Expanding this summation, we see that

$$(\sigma^{(i)})^2 = \frac{1}{N^2} \sum_{k,s=1}^{N} \Gamma(k-s) \tag{A.8}$$

with the autocorrelation coefficient $\Gamma(k-s) = \big(h(\boldsymbol{x}^{(k)}) - \bar{h}_N^{(i)}\big)\big(h(\boldsymbol{x}^{(s)}) - \bar{h}_N^{(i)}\big)$. Next we rewrite

$$\sum_{k,s=1}^{N} \Gamma(k-s) = \mathrm{Var}(h(\boldsymbol{x})) \sum_{k,s=1}^{N} \frac{\Gamma(k-s)}{\Gamma(0)} = \mathrm{Var}(h(\boldsymbol{x})) \sum_{k,s=1}^{N} \rho(k-s)\,, \tag{A.9}$$

with the normalised autocorrelation coefficient $\rho(k-s) = \Gamma(k-s)/\Gamma(0)$ at lag $k-s$ and $\Gamma(0) = \mathrm{Var}(h(\boldsymbol{x}))$ for $k = s$. Typically $\Gamma(t)$ decays exponentially so that, for $N \gg \tau$, $\Gamma(t) \overset{t\to\infty}{\propto} \exp\{-|t|/\tau\}$ and we can approximate

$$\sum_{k,s=1}^{N} \rho(k-s) = N \sum_{t=-(N-1)}^{N-1} \left(1 - \frac{t}{N}\right)\rho(t) \approx N \sum_{t=-\infty}^{\infty} \rho(t)\,, \tag{A.10}$$

see [55, p. 137]. If $\tau \gg 1$

$$\sum_{t=-\infty}^{\infty} \rho(t) = 1 + 2\sum_{t=1}^{\infty} \left(e^{-1/\tau}\right)^t = 1 + 2\frac{e^{-1/\tau}}{1 - e^{-1/\tau}} \approx 1 + 2\frac{1 - 1/\tau}{1/\tau} = 2\tau - 1\,. \tag{A.11}$$

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

$$(\sigma^{(i)})^2 \approx \frac{\mathrm{Var}(h(\boldsymbol{x}))}{N} \sum_{t=-\infty}^{\infty} \rho(t) \approx \frac{\mathrm{Var}(h(\boldsymbol{x}))}{N} \underbrace{\left(1 + 2\sum_{t=1}^{W} \rho(t)\right)}_{:=\tau_{\mathrm{int}}} = \mathrm{Var}(h(\boldsymbol{x}))\frac{\tau_{\mathrm{int}}}{N}\,, \tag{A.12}$$

where W is the summation window and we define the IACT as twice the value as in [67, pp. 103-105]. The IACT provides a good estimate of how many steps the sampling algorithm needs to take to produce one independent sample. More specifically, the effective sample size $\frac{\tau_{\mathrm{int}}}{N}$ gives an estimate of how efficient a sampler is.

A.3 Python Code

```
def MargBack(TTCore, univarGrid):
  ''' Backward marginalisation (see Prop.  1) as in SIRT from Cui and Dolgov [8] '''

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

  for k in range(dim - 1, 0, -1):
    r_kmin1, n, r_k = np.shape(TTCore[k])
    # Eq. 2.28, [8, Eq. 22] !! we set Lebesgue Measure to const = one
    M = np.identity(n) * (univarGrid[k][-1] - univarGrid[k][0])  # Mass matrix
    L = scy.linalg.cholesky(M)

    # construct Tensor C  Eq. 2.33, [8, Eq. 27]
    C[k] = np.zeros((r_kmin1, n, r_k))
    for alpha in range(0, r_kmin1):
      for l in range(0, r_k):
        C[k][alpha, :, l] = B[k][alpha, :, l] @ L[:, :]

    # unfold along first coordinate and compute thin QR decomposition of C^T
    #  Eq. 2.34, [8, Eq. 28]
    Q, R[k] = np.linalg.qr(C[k].reshape((r_kmin1, n * r_k), order='C').transpose(), mode='reduced')

    # compute next coefficient tensor Eq. 2.35, [8, Eq. 29]
    r_kmin2, n, r_kmin1 = np.shape(TTCore[k - 1])
    B[k - 1] = np.zeros(np.shape(TTCore[k - 1]))
    for alpha_2 in range(0, r_kmin2):
      for l_1 in range(0, r_kmin1):
        B[k - 1][alpha_2, :, l_1] = TTCore[k - 1][alpha_2, :, :] @ R[k][l_1, :]

return B
```

Listing A.1: Python code to calculate Backward marginals, as in Prop. 1 and [8].

```python
def MargForw(TTCore, univarGrid):
''' Forward marginalisation (see Prop. 2)
  similar to backward marginalsiation as in  Cui and Dolgov [8] '''

# compute pre marginal coefficients sarting at dim = 1, k = 0
BPre = dim * [None]  # coeffTensor
LebLam = 1  # !! Lebesgue Measure
BPre[0] = TTCore[0]
RPre = [None] * dim
CPre = [None] * dim

for k in range(0, dim-1):
  r_kmin1, n, r_k = np.shape(TTCore[k])
  #  Eq. 2.28, [8, Eq. 22] !! we set Lebesgue Measure to const = one
  M = np.identity(n) * (univarGrid[k][-1] - univarGrid[k][0])  # Mass matrix
  L = scy.linalg.cholesky(M)

  # construct Tensor C  Eq. 2.36
  CPre[k] = np.zeros((r_kmin1, n, r_k))
  for alpha in range(0, r_kmin1):
    for l in range(0, r_k):
      CPre[k][alpha, :, l] = BPre[k][alpha, :, l] @ L[:, :]

  # unfold along first coordinate and compute thin QR decomposition of C
  # Eq. 2.37
  Q, RPre[k] = np.linalg.qr(CPre[k].reshape((r_kmin1 * n, r_k), order='C'), mode='reduced')

  # compute next coefficient tensor  Eq. 2.38
  r_k, n, r_kpls1 = np.shape(TTCore[k + 1])
  BPre[k + 1] = np.zeros(np.shape(TTCore[k + 1]))
  for alpha_1 in range(0, r_kpls1):
    for l_1 in range(0, r_k):
      BPre[k + 1][l_1, :, alpha_1] = RPre[k][l_1,:]
                                          @ TTCore[k + 1][:, :, alpha_1]

return BPre
```

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

```python
def SIRT(seeds, SQTT, univarGrid, BackMarg, absError):
  ''' do squared inverse rosenblatt transform (SIRT) as in Cui et al. [8] '''

  dim, numbSampl = seeds.shape
  sampls = np.zeros(seeds.shape) # samples from approximated PDF
  probVal = np.zeros(seeds.shape) # PDF values, for MH-correction step
  Approx = np.zeros(seeds.shape[1]) # TT-Approx., to compare to true function

  # Lebesgue measure for quadrature Eq. 2.20
  WholeLebLam = np.zeros(dim)
  for k in range(0, dim):
    WholeLebLam[k] = (univarGrid[k][-1] - univarGrid[k][0])
  lamX = np.ones(dim)
  for k in range(1, dim):
    lamX[k - 1] = np.prod(WholeLebLam[k:])


  gamError = absError / np.prod(WholeLebLam) # error as in Eq. 2.25 [8, Eq. 35]

  # sample from first dimension [8, Eq. 30]
  firstMarg = gamError * lamX[0] + np.sum(BackMarg[0][0, :, :] ** 2, 1)
  # cumulative distribution function, normalised numerically Eq. 2.39 [8, Eq. 17]
  firstCDF = np.cumsum(firstMarg / np.sum(firstMarg))
  # draw samples as 'inverse transform'
  sampls[0] = np.interp(seeds[0], firstCDF, univarGrid[0])
  probVal[0] = np.interp(sampls[0], univarGrid[0], firstMarg / np.sum(firstMarg))

  # sample from other dimensions
  for n in range(0, numbSampl):
    # piecew. poly. interpol in first dimension Eq. 2.41 [11]
    CurrApprCore = LinInterPolTT(SQTT[0], univarGrid[0], sampls[0][n])
    for d in range(1, dim):
      # marginal function, conditioned on previous samples
      rank_min, gridSize, rank_pls = BackMarg[d].shape
      MargDep = np.zeros((BackMarg[d].shape))
      for r in range(0, rank_min):
        # condition on previous samples
        MargDep[r, :, :] = CurrApprCore[0, r] * BackMarg[d][r, :, :]

      currMarg = gamError * lamX[d] + np.sum(np.sum(np.copy(MargDep), axis=0)** 2,
        axis=1) # Eq. 2.40 [8, Eq. 31]

      currCDF = np.cumsum(currMarg / np.sum(currMarg)) # Eq. 2.39 [8, Eq. 17]

      # draw sample as 'inverse transform'
      sampls[d][n] = np.interp(seeds[d][n], currCDF, univarGrid[d])
      probVal[d][n] = np.interp(sampls[d][n], univarGrid[d],
        currMarg / np.sum(currMarg))
      # piecew. poly. interpol., Eq. 2.41 [11], cond. on sampl. for next PDF
      CurrApprCore = np.copy(CurrApprCore) @ LinInterPolTT(SQTT[d], univarGrid[d],
        sampls[d][n])

  Approx[n] = gamError + CurrApprCore ** 2

  return sampls, probVal, Approx
```

Listing A.3: Python code to draw samples via SIRT, as in Alg. Box 1.

B
Additional Figures

Some of the additional figures shown are repetitive and omitted from the main document; others provide details that are not necessary for understanding the main results but may be interesting and offer a more visual understanding for the curious reader.

B.1 Ozone

B.1.1 Ozone Prior

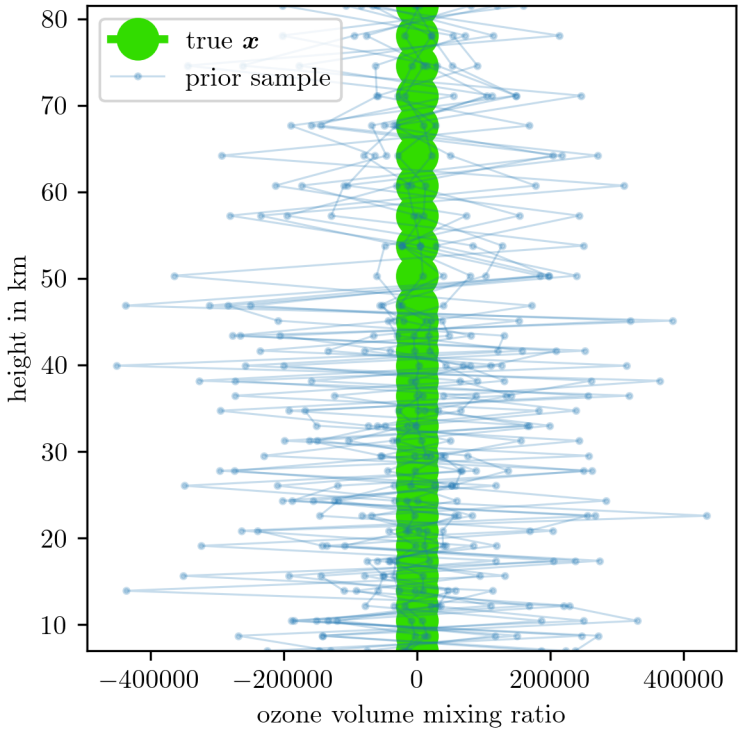


Figure B.1: We draw samples from ozone prior distribution $\boldsymbol{x} \sim \mathcal{N}(0, \delta \boldsymbol{L})$ after generating a sample from the hyper-prior distribution $\delta \sim \mathcal{T}(1, 10^{-10})$. Note that since the variance of prior samples is very large compared to the ozone volume mixing ratios, the ozone profile appears to be constant, which is not the case, see e.g. Fig. 4.5.

B.1.2 Integrated Autocorrelation Time

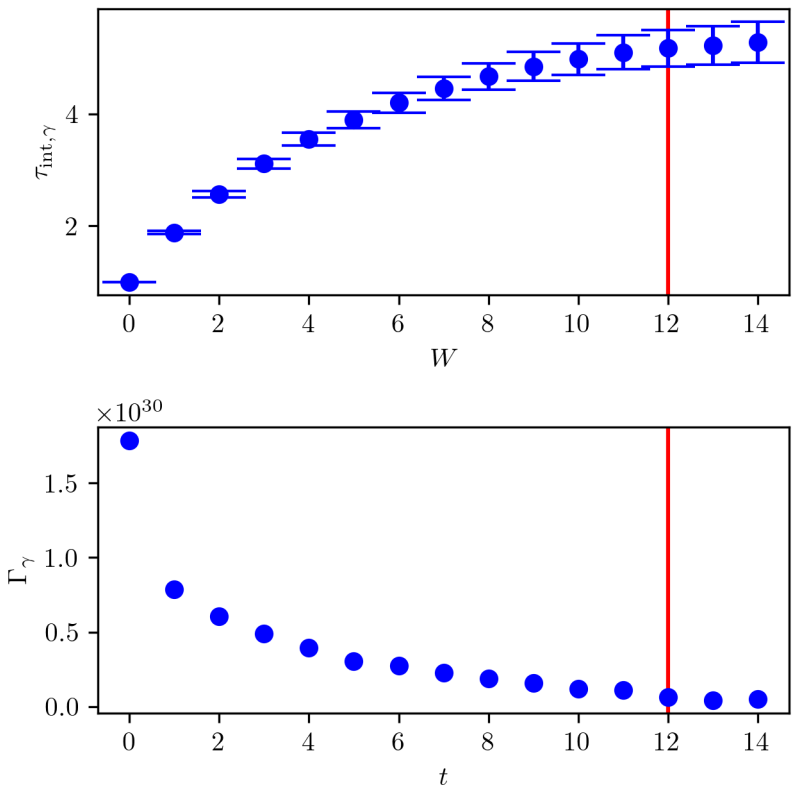


Figure B.2: Provided by [27], the IACT $\tau_{\text{int},\gamma}$ at summation windows W as well as the estimated autocorrelation function Γ_γ at lag t of the samples $\gamma \sim \pi(\cdot|\boldsymbol{y})$ based on the linear forward model.

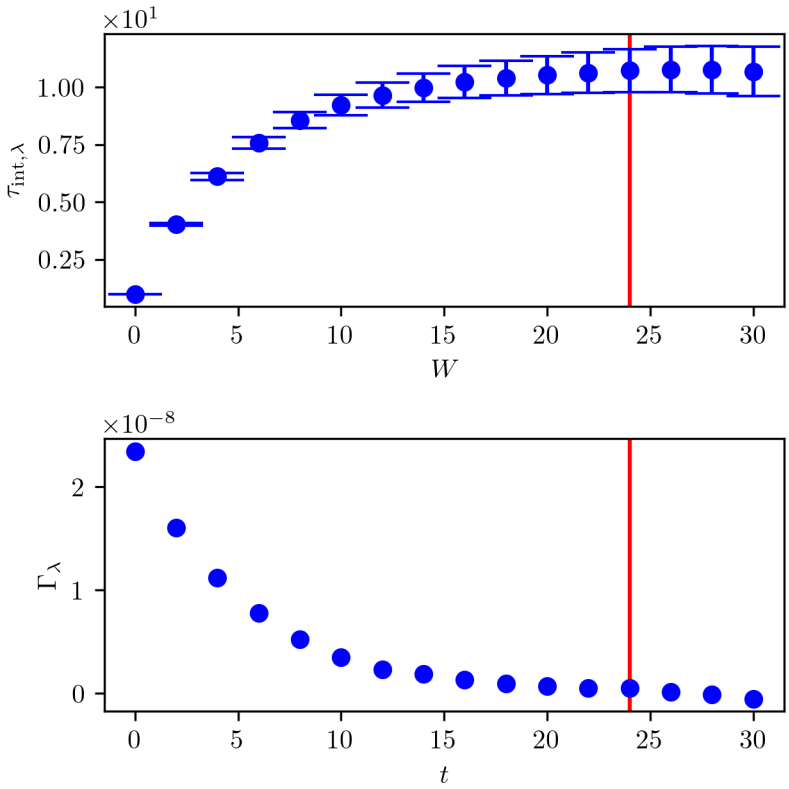


Figure B.3: Provided by [27], the IACT $\tau_{\text{int},\lambda}$ at summation windows W as well as the estimated autocorrelation function Γ_λ at lag t of the samples $\lambda \sim \pi(\cdot|\boldsymbol{y})$ based on the approximated forward model.

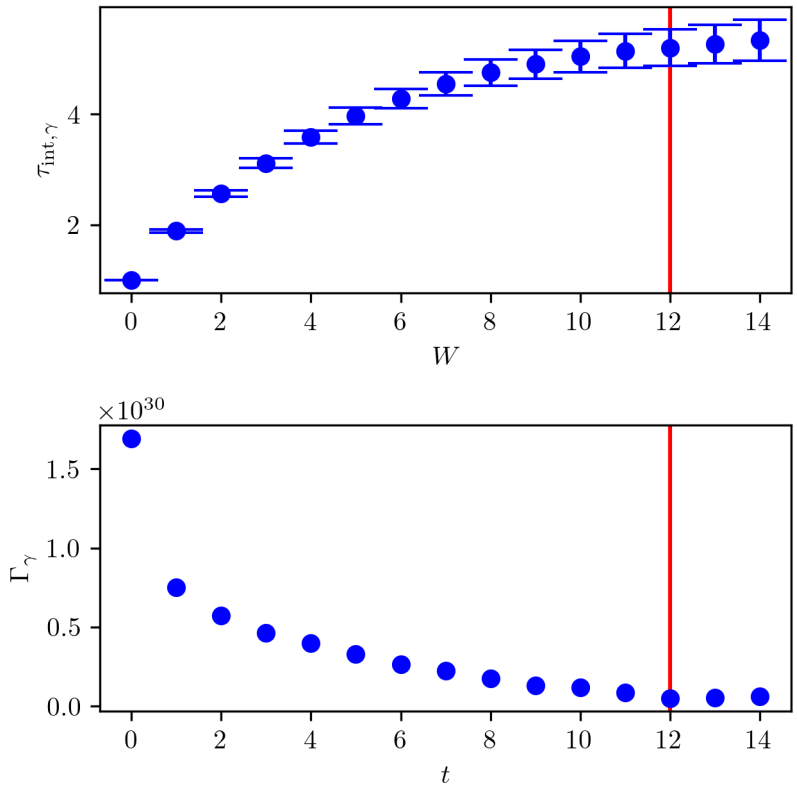


Figure B.4: Provided by [27], the IACT $\tau_{\mathrm{int},\gamma}$ at summation windows W as well as the estimated autocorrelation function Γ_γ at lag t of the samples $\gamma \sim \pi(\cdot|\boldsymbol{y})$ based on the approximated forward model.

B.1.3 Eigenvectors of Full Conditional Posterior Precision Matrix

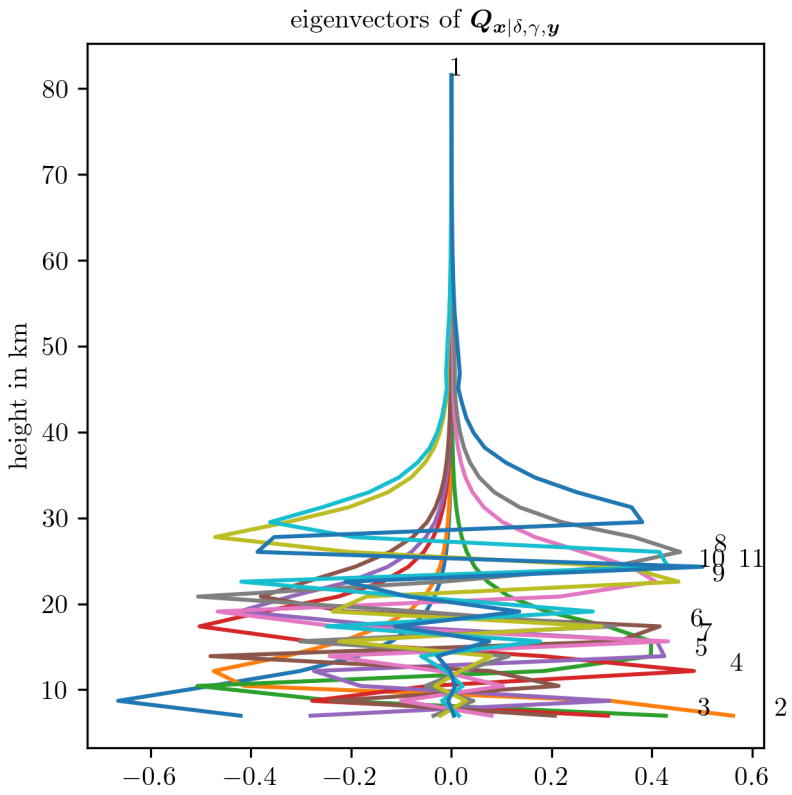


Figure B.5: First 11 eigenvectors corresponding to in size ordered eigenvalues of conditional precision matrix $\boldsymbol{Q}_{\boldsymbol{x}|\delta,\gamma,\boldsymbol{y}}$. We see that the eigenvectors span structures for heights ≤ 40.

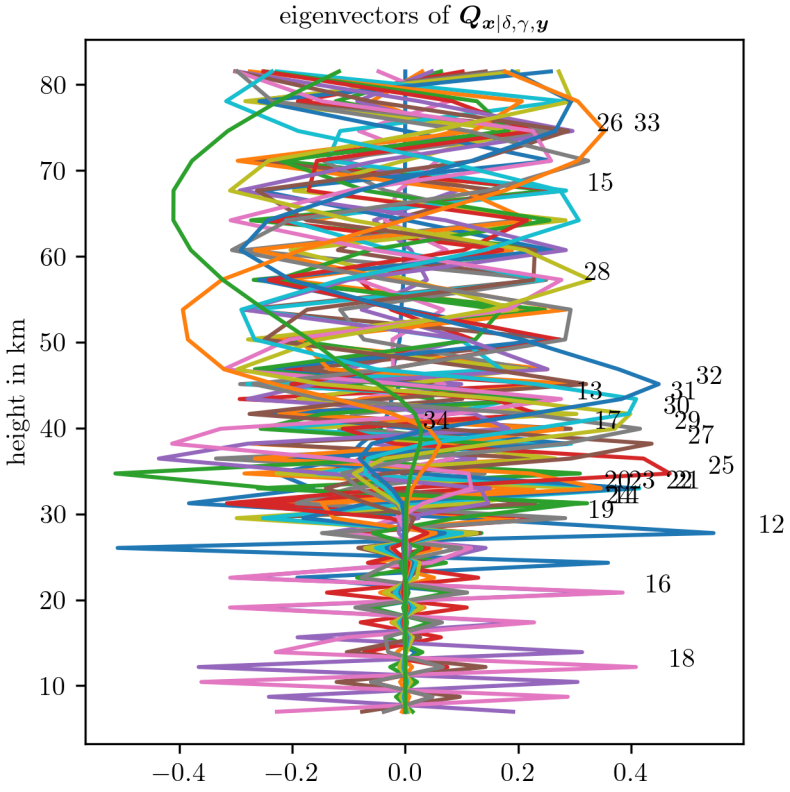


Figure B.6: Last 23 eigenvectors corresponding to in size ordered eigenvalues of conditional precision matrix $\boldsymbol{Q}_{\boldsymbol{x}|\delta,\gamma,\boldsymbol{y}}$. The eigenvectors represent structures according to the prior.

B.2 Pressure and Temperature

B.2.1 Priors

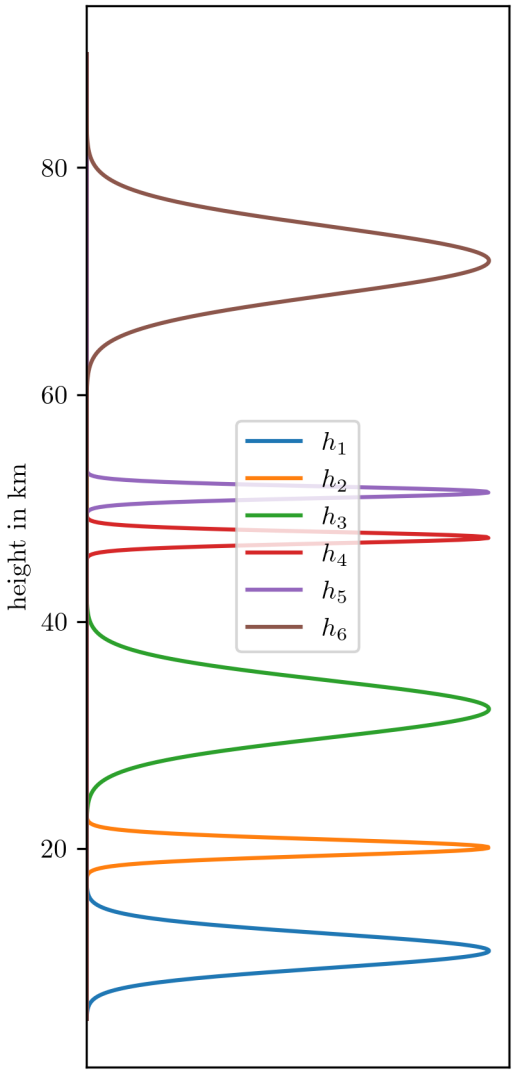


Figure B.7: Prior distributions $\pi(\boldsymbol{h_T})$, which we choose so that they do not overlap and not conflict with the temperature function in Eq. 3.9.

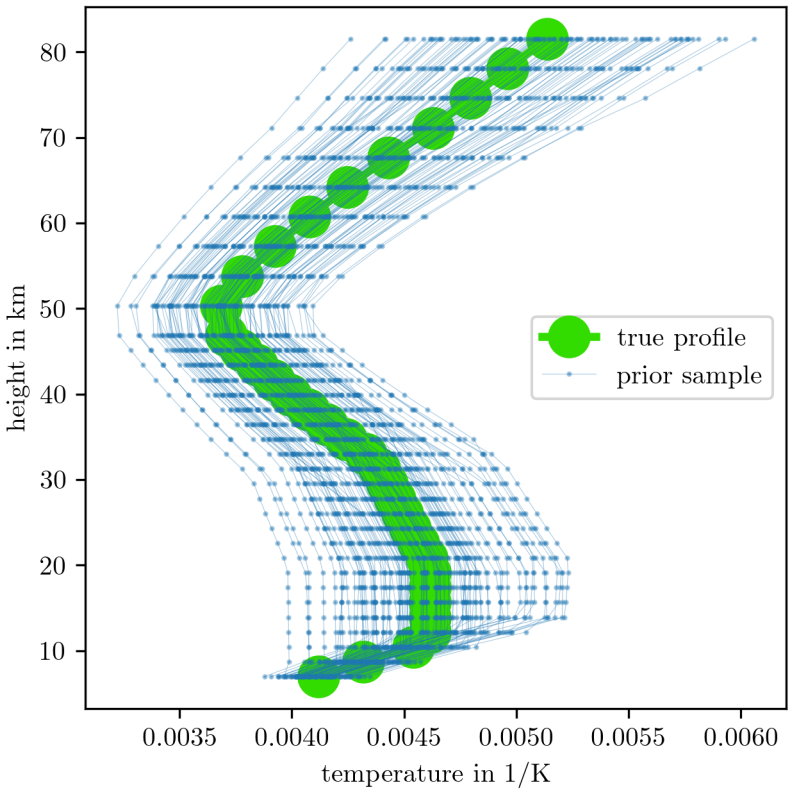


Figure B.8: Prior samples of the inverted temperature profile.

B.2.2 Integrated Autocorrelation Time

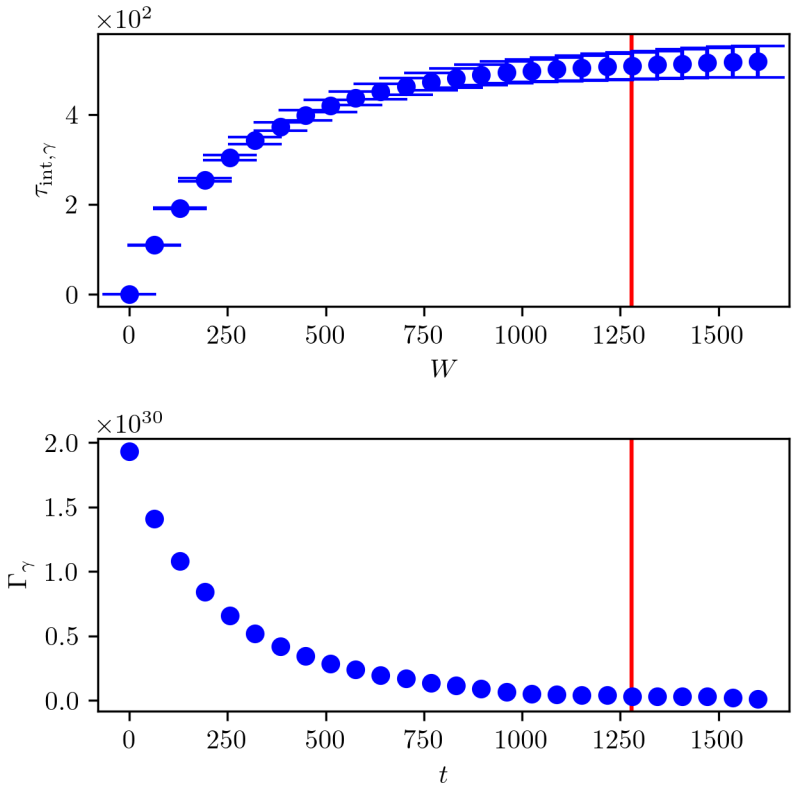


Figure B.9: Provided by [27], the IACT $\tau_{\text{int},\gamma}$ at summation windows W and the estimated autocorrelation function Γ_γ at lag t of samples $\gamma \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

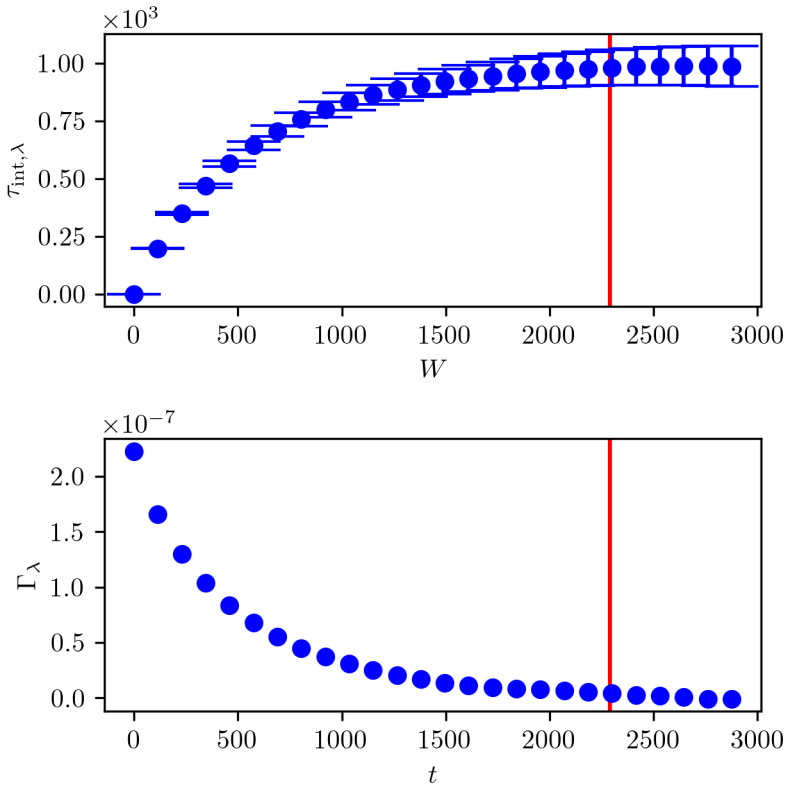


Figure B.10: Provided by [27], the IACT $\tau_{\text{int},\lambda}$ at summation windows W and the estimated autocorrelation function Γ_λ at lag t of samples $\lambda \sim \pi(\cdots|\boldsymbol{y})$ from the t-walkfor the approximated forward model.

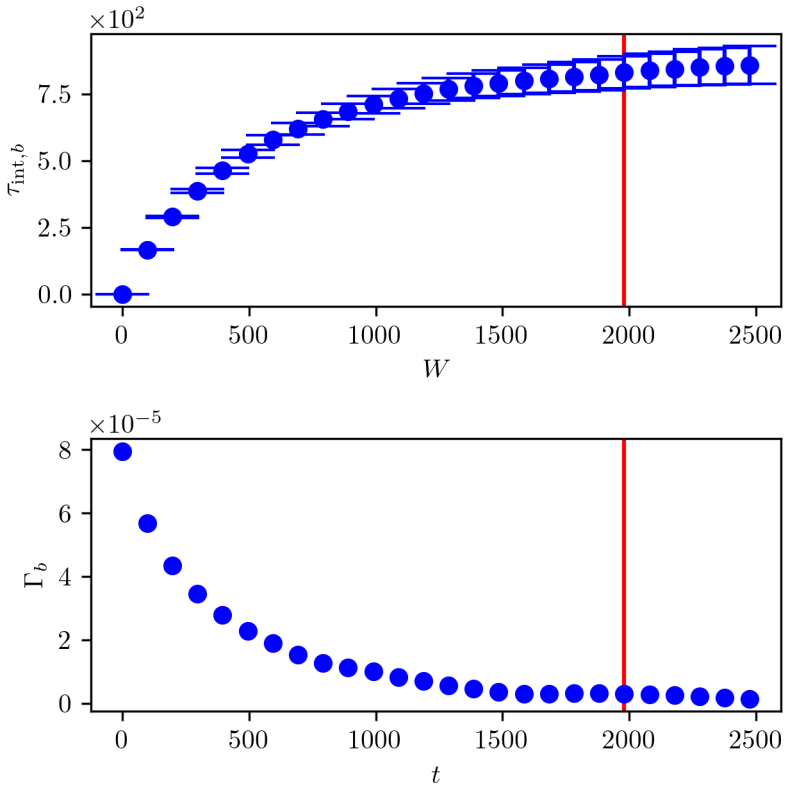


Figure B.11: Provided by [27], the IACT $\tau_{\text{int},b}$ at summation windows W and the estimated autocorrelation function Γ_b at lag t of samples $b \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

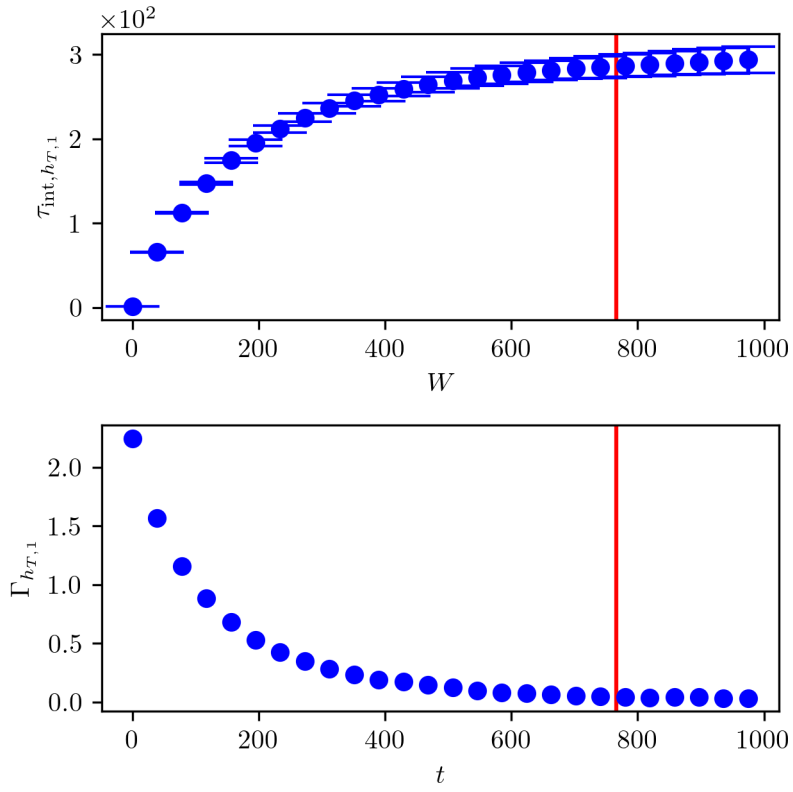


Figure B.12: Provided by [27], the IACT $\tau_{\text{int},h_{T,1}}$ at summation windows W and the estimated autocorrelation function $\Gamma_{h_{T,1}}$ at lag t of samples $h_{T,1} \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

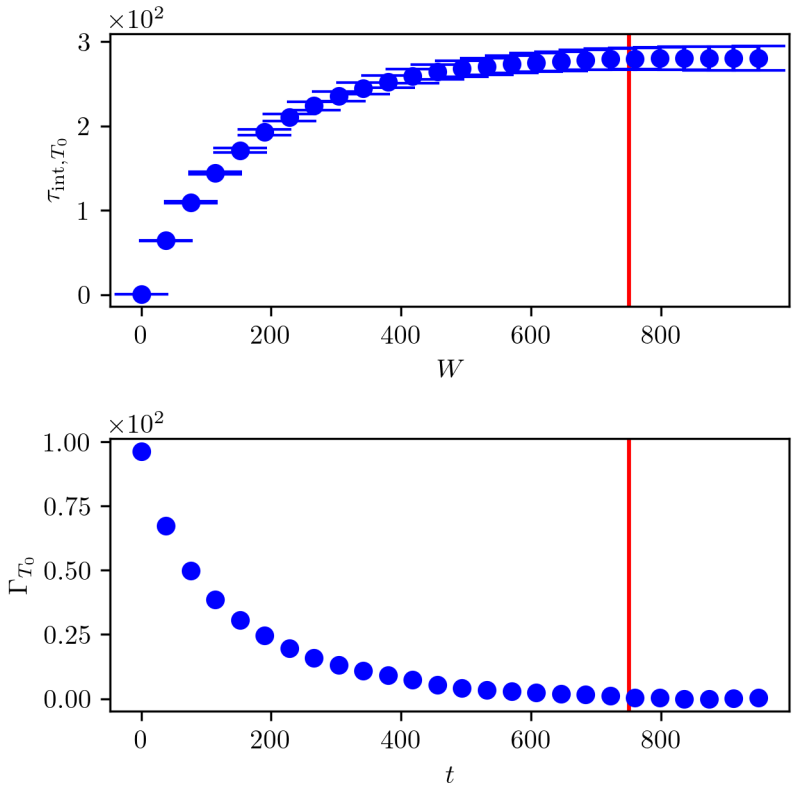


Figure B.13: Provided by [27], the IACT τ_{int,T_0} at summation windows W and the estimated autocorrelation function Γ_{T_0} at lag t of samples $T_0 \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

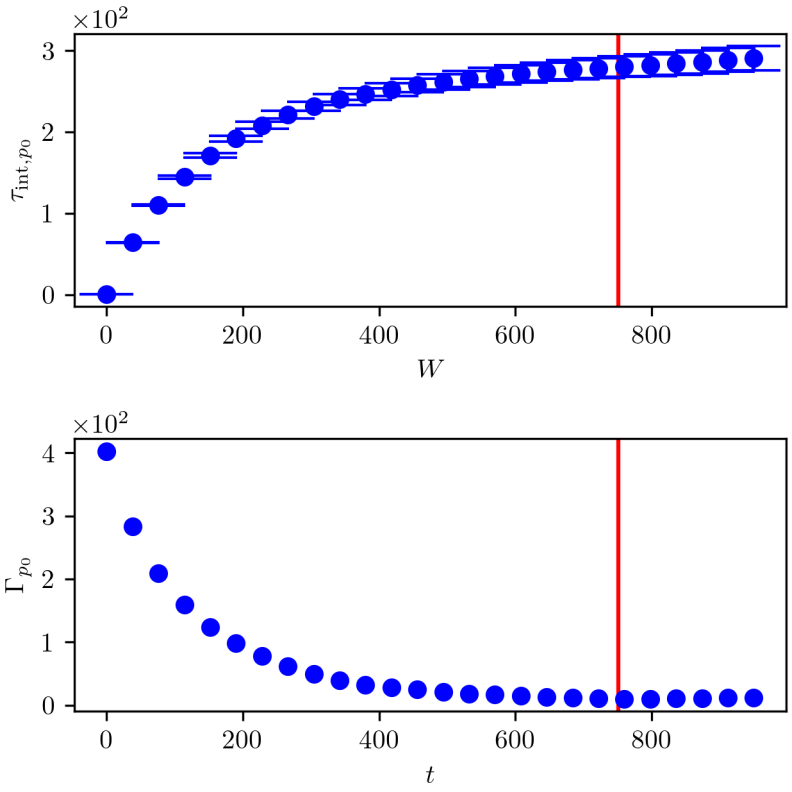


Figure B.14: Provided by [27], the IACT τ_{int,p_0} at summation windows W and the estimated autocorrelation function Γ_{p_0} at lag t of samples $p_0 \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

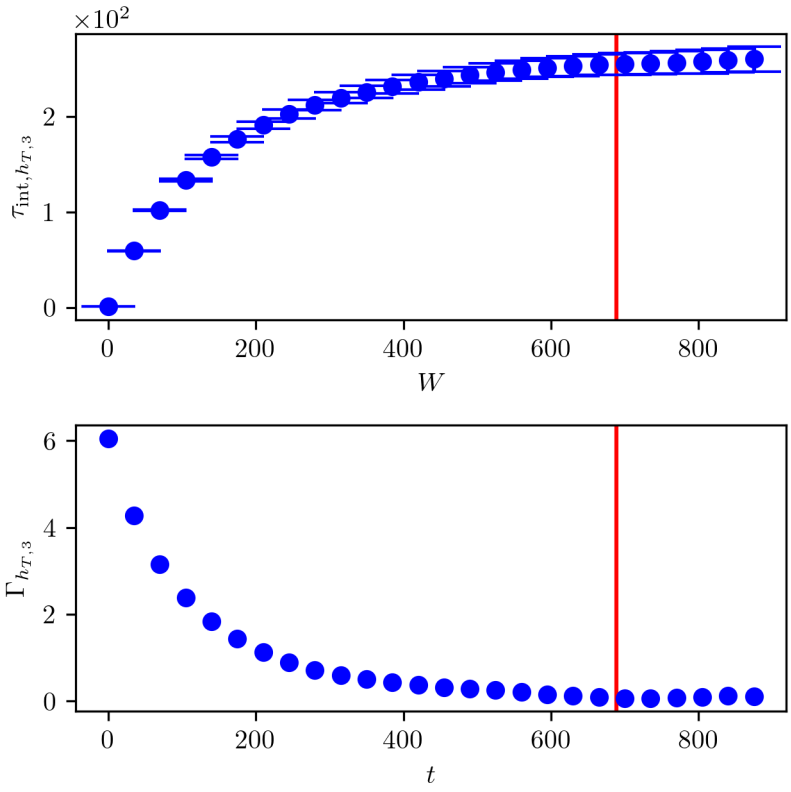


Figure B.15: Provided by [27], the IACT $\tau_{\text{int},h_{T,3}}$ at summation windows W and the estimated autocorrelation function $\Gamma_{h_{T,3}}$ at lag t of samples $h_{T,3} \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

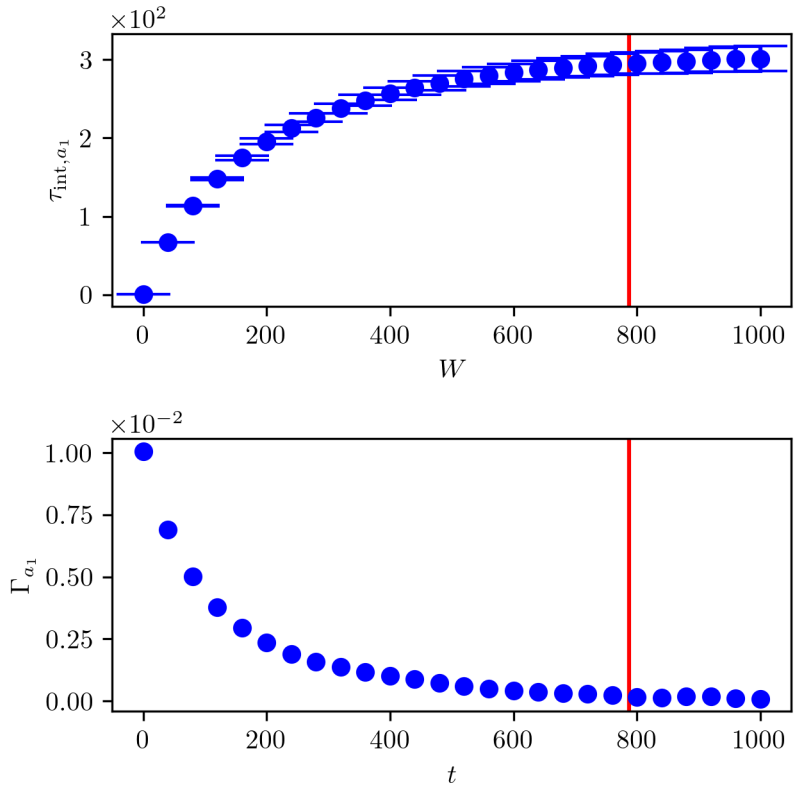


Figure B.16: Provided by [27], the IACT τ_{int,a_1} at summation windows W and the estimated autocorrelation function Γ_{a_1} at lag t of samples $a_1 \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

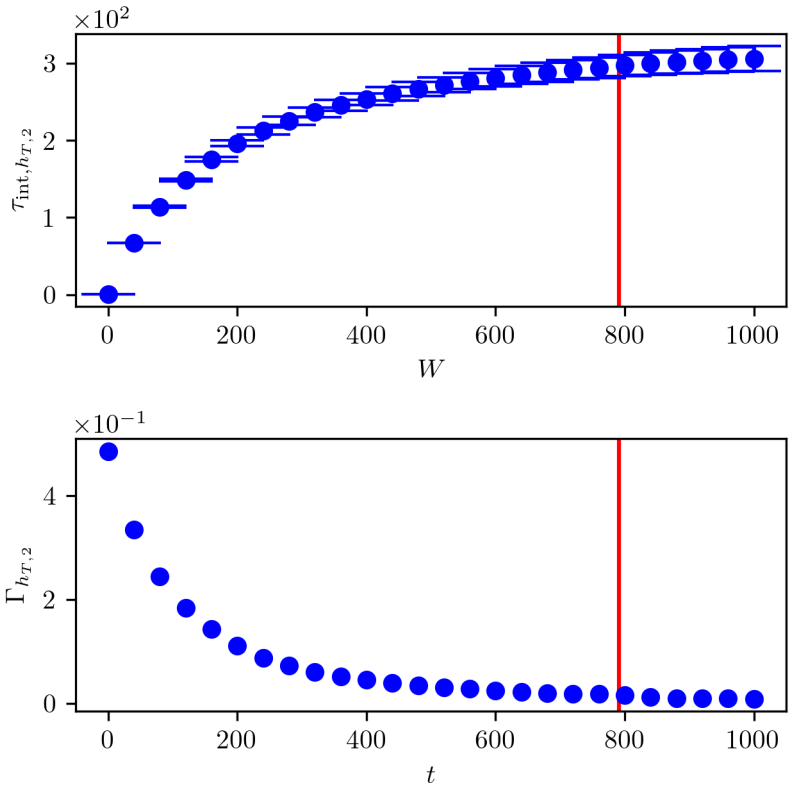


Figure B.17: Provided by [27], the IACT $\tau_{\text{int},h_{T,2}}$ at summation windows W and the estimated autocorrelation function $\Gamma_{h_{T,2}}$ at lag t of samples $h_{T,2} \sim \pi(\cdots|\boldsymbol{y})$ from the t-walk for the approximated forward model.

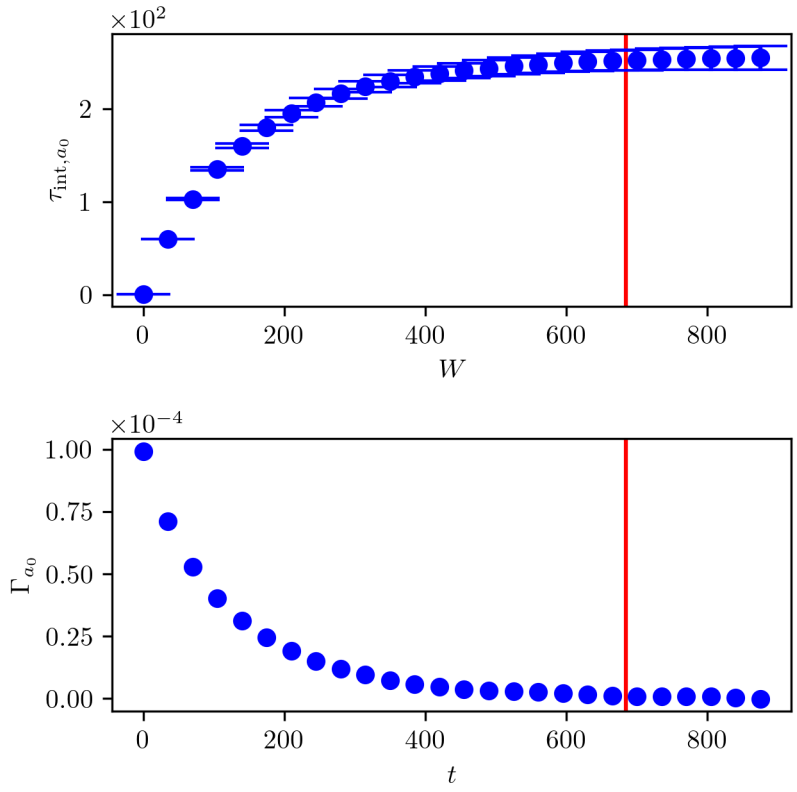


Figure B.18: Provided by [27], the IACT τ_{int,a_0} at summation windows W and the estimated autocorrelation function Γ_{a_0} at lag t of samples $a_0 \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

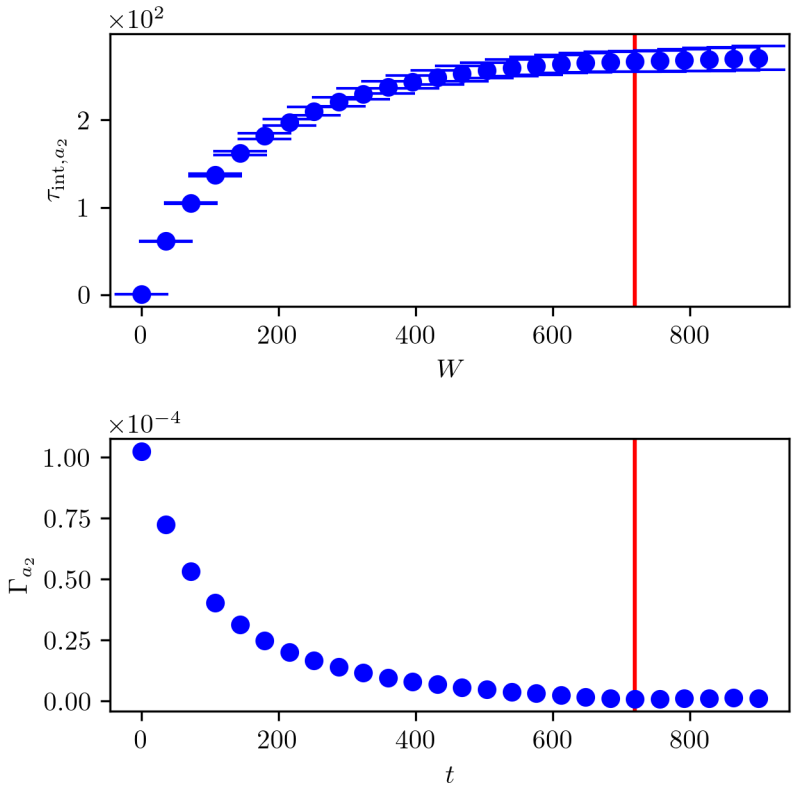


Figure B.19: Provided by [27], the IACT τ_{int,a_2} at summation windows W and the estimated autocorrelation function Γ_{a_2} at lag t of samples $a_2 \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

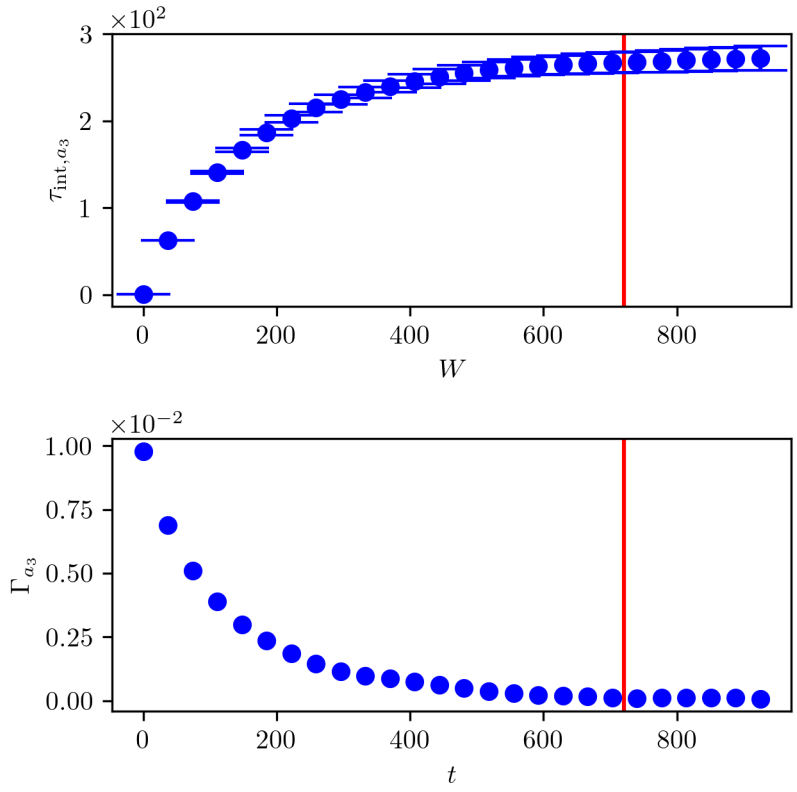


Figure B.20: Provided by [27], the IACT τ_{int,a_3} at summation windows W and the estimated autocorrelation function Γ_{a_3} at lag t of samples $a_3 \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

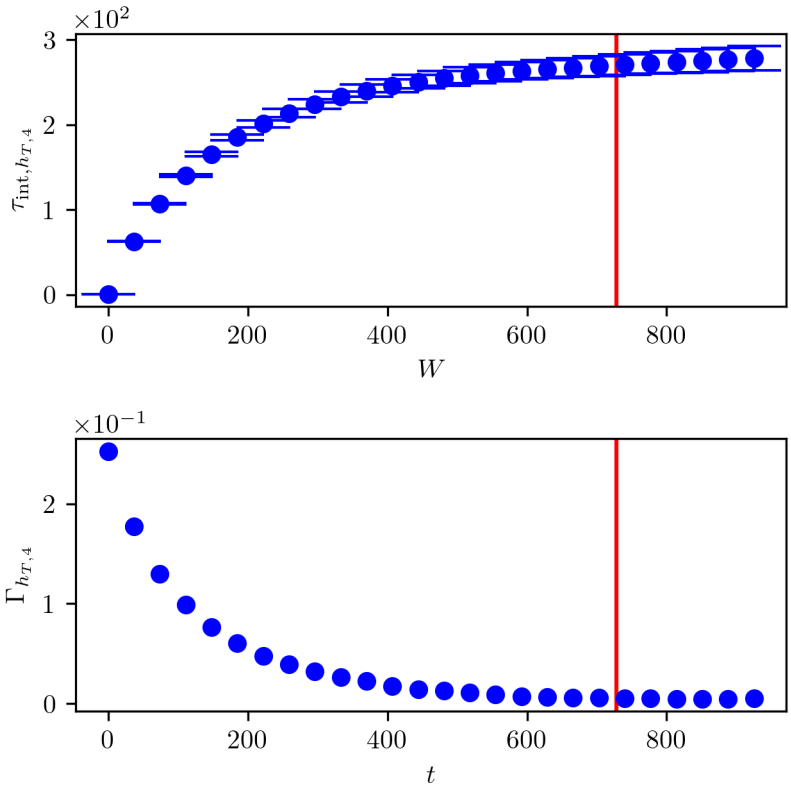


Figure B.21: Provided by [27], the IACT $\tau_{\text{int},h_{T,4}}$ at summation windows W and the estimated autocorrelation function $\Gamma_{h_{T,4}}$ at lag t of samples $h_{T,4} \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

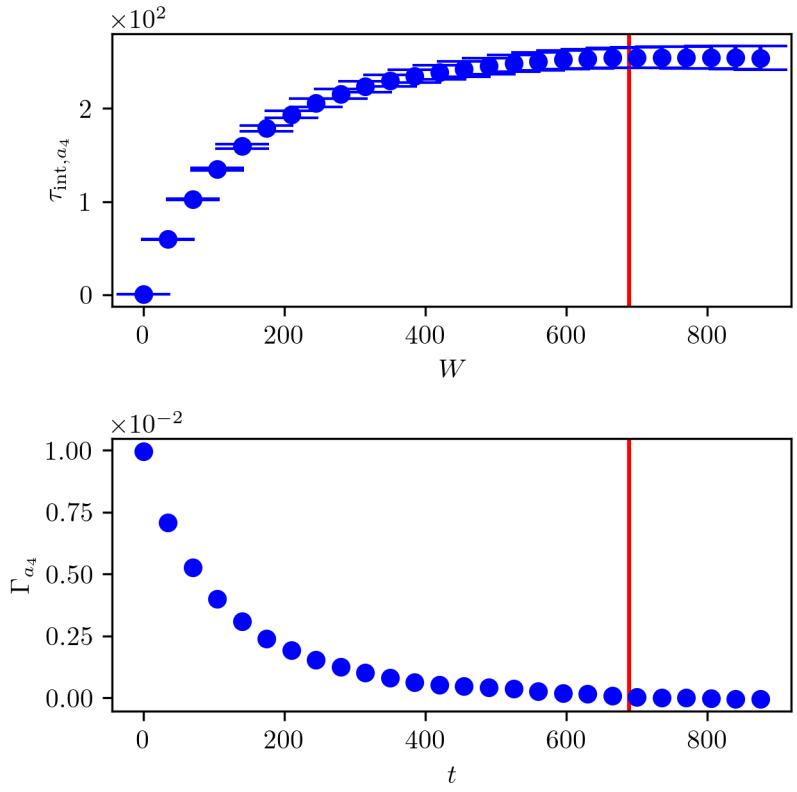


Figure B.22: Provided by [27], the IACT τ_{int,a_4} at summation windows W and the estimated autocorrelation function Γ_{a_4} at lag t of samples $a_4 \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

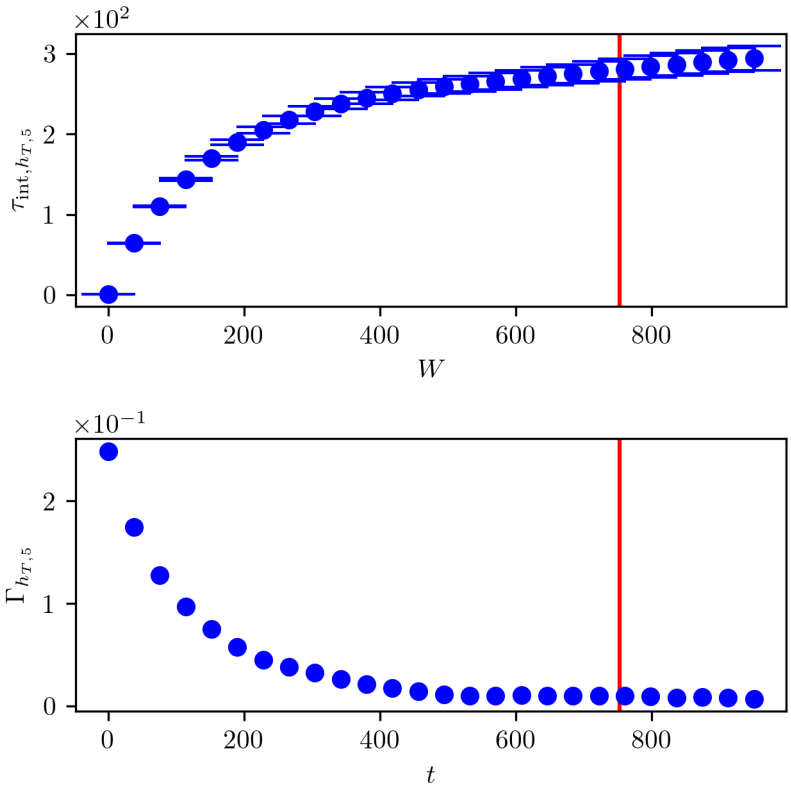


Figure B.23: Provided by [27], the IACT $\tau_{\text{int},h_{T,5}}$ at summation windows W and the estimated autocorrelation function $\Gamma_{h_{T,5}}$ at lag t of samples $h_{T,5} \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

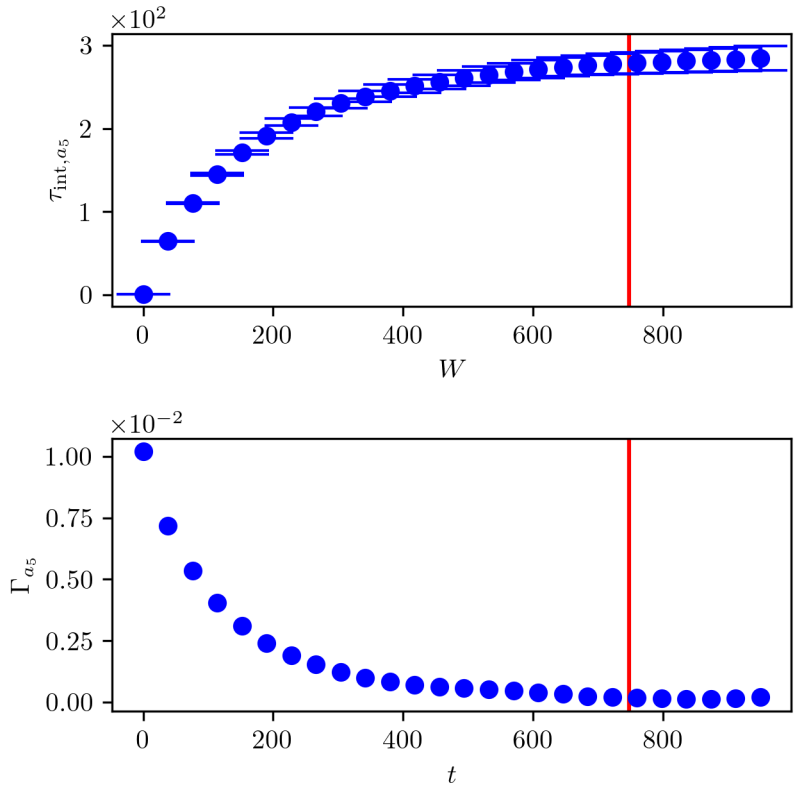


Figure B.24: Provided by [27], the IACT τ_{int,a_5} at summation windows W and the estimated autocorrelation function Γ_{a_5} at lag t of samples $a_5 \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

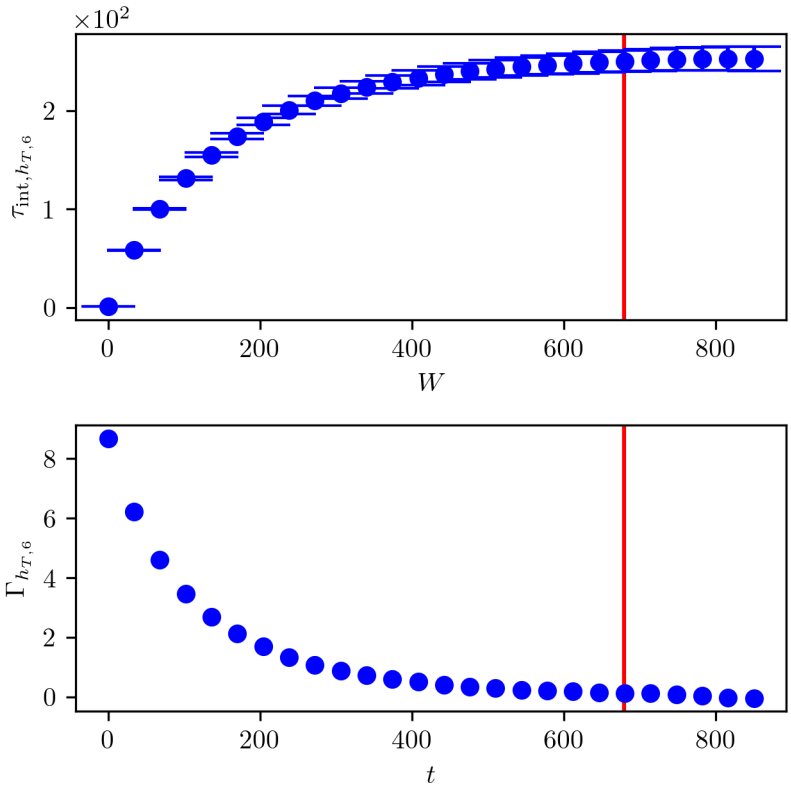


Figure B.25: Provided by [27], the IACT $\tau_{\text{int},h_{T,6}}$ at summation windows W and the estimated autocorrelation function $\Gamma_{h_{T,6}}$ at lag t of samples $h_{T,6} \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

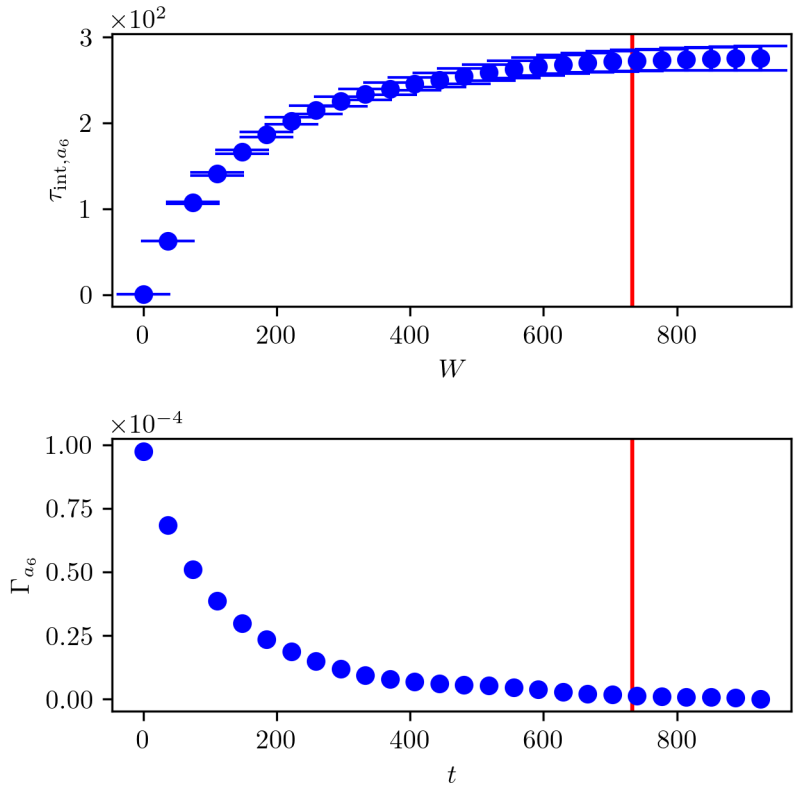


Figure B.26: Provided by [27], the IACT τ_{int,a_6} at summation windows W and the estimated autocorrelation function Γ_{a_6} at lag t of samples $a_6 \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.