# Stochastic Analytic Continuation: A Bayesian Approach

Von der Fakultät für Mathematik, Informatik und Naturwissenschaften der RWTH Aachen University zur Erlangung des akademischen Grades eines Doktors der Naturwissenschaften genehmigte Dissertation

vorgelegt von

*Khaldoon Ghanem, M.Sc.*

aus

*Damaskus, Syrien*

Berichter: Apl. Prof. Dr. Erik Koch
Univ.-Prof. Stefan Wessel, Ph.D.

Tag der mündlichen Prüfung: 26. Juni 2017

*To my dear parents*

إلى أمي وأبي الغاليين

# Acknowledgments

First of all, I would like to thank my supervisor, Prof. Dr. Erik Koch, for his guidance, patience and understanding. You gave me space to work independently, yet you have always been there for inspiring discussions and suggestions. You helped me build my academic skills and opened new opportunities. I will always be indebted to you.

Next, I would like to thank Prof. Dr. Stefan Wessel for being the second referee of this thesis and for his helpful comments. I also want to thank my colleagues, Hunter, Michael, Julian, Esmaeel and Amin, for the time we spent together.

Special thanks to my colleague and dear friend Qian for helping with the small things. I will miss our after-work dinners at Pontstraße and the game nights at your place.

I would like to express my gratitude to my parents, Ahmad and Sawsan, for their love and encouragement. You have always valued my passion for science, and believed in me and pursuing my goals. You are the reason I am where I am today.

I am also grateful to my sister, Khoulod, and my best friends, Hany and Zaher. Despite the long distances, your support and calls gave me perspective, and helped me overcome the hard times in my personal life.

Finally, I would like to thank German Research School for Simulation Sciences for funding the first part of my work, and Jülich Supercomputing Center for funding the second part.

# Abstract

The stochastic sampling method (StochS) is used for the analytic continuation of quantum Monte Carlo data from the imaginary axis to the real axis. Compared to the maximum entropy method, StochS does not have explicit parameters, and one would expect the results to be unbiased. We present a very efficient algorithm for performing StochS and use it to study the effect of the discretization grid. Surprisingly, we find that the grid affects the results of StochS acting as an implicit default model. We provide a recipe for choosing a reliable StochS grid.

To reduce the effect of the grid, we extend StochS into a gridless method (gStochS) by sampling the grid points from a default model instead of having them fixed. The effect of the default model is much reduced in gStochS compared to StochS and depends mainly on its width rather than its shape. The proper width can then be chosen using a simple recipe like we did in StochS.

Finally, to avoid fixing the width, we go one step further and extend gStochS to sample over a whole class of default models with different widths. The extended method (eStochS) is then able to automatically relocate the grid points and concentrate them in the important region. Test cases show that eStochS gives good results resolving sharp features in the spectrum without the need for fine tuning a default model.

# Zusammenfassung

Die Stochastische-Mittelungs-Methode (StochS) wird für die analytische Fortsetzung der Quanten-Monte-Carlo-Daten von der imaginären Achse zur reellen Achse verwendet. Im Vergleich zu Maximum-Entropie-Methode hat StochS keine expliziten Parameter, so dass man unbeeinflusste Ergebnisse erwarten würde. Wir stellen einen hocheffizienten Algorithmus für StochS vor und benutzen ihn, um der Einfluss des Gitters zu analysieren. Überraschenderweise finden wir, dass das Gitter die Ergebnisse von StochS wie ein Default-Modell beeinflusst. Wir geben ein Rezept für die Wahl eines verlässlichen StochS-Gitter an.

Um der Einfluss des Gitters zu reduzieren, erweitern wir StochS zu einer gitterlosen Methode (gStochS), indem die Gitterpunkte gemäß einem Default-Modell gezogen werden, anstatt sie a priori festzulegen. Der Einfluss des Default-Modells in gStochS ist im Vergleich zu StochS stark reduziert und hängt hauptsächlich von seiner Breite, nicht der Form ab. Die passende Breite kann dann mit einem einfachen Rezept gefunden werden, ähnlich wie wir es für StochS entwickelt haben.

Schließlich gehen wir, um die Festlegung der Breite zu vermeiden, einen Schritt weiter und erweitern gStochS durch die Mittelung über eine ganze Klasse von Default-Modellen mit unterschiedlichen Breiten. Diese erweiterte Methode (eStochS) kann dann die Gitterpunkte automatisch versetzen und in den wichtigen Bereichen konzentrieren. Testfälle zeigen, dass eStochS gute Ergebnisse liefert, die scharfe Struktur des Spektrums reproduzieren kann, ohne dass eine Feinabstimmung eines Default-Modells nötig wäre.

# Contents

# Introduction

Quantum Monte Carlo (QMC) methods often compute Green or correlation functions for imaginary times or Matsubara frequencies. This data need to be analytically continued to the real axis in order to extract the dynamical properties of the physical system of interest. One example of analytic continuation is retrieving the spectral function $A(\omega)$ at real frequencies from the imaginary time Green function $\mathcal{G}(\tau)$ coming from continuous-time QMC for DMFT. The Green and spectral functions are related by the following relation

$$\mathcal{G}(\tau) = \int \frac{d\omega}{2\pi} \frac{-e^{-\tau\omega}}{1 \pm e^{-\beta\omega}} A(\omega), \quad \tau \in [0, \beta] \tag{0.1}$$

where the upper (lower) sign is for the fermionic (bosonic) case and $\beta = 1/T$ is the inverse temperature. In general, the analytic continuation problem can be formulated as a Fredholm integral equation of first kind

$$g(y) = \int dx \; K(y,x) f(x) \; , \tag{0.2}$$

where the left-hand side $g(y)$ represents QMC data known numerically, while the integral kernel $K(y,x)$ is a continuous function known analytically. The goal is estimating the function $f(x)$, called the the model. Based on physical arguments, the model is always a density-like function i.e. it should be non-negative and integrable. It even sometimes satisfies few sum rules; e.g. the fermionic spectral function is normalized. These properties, especially the non-negativity, provide important prior information that help making the analytic continuation easier.

In the presence of noise, solving the above integral equation is an ill-posed problem with no unique solution. When computing the data $g(y)$, oscillations and sharp features in the model $f(x)$ get smoothed and noise get damped due to the integration. The inverse problem of reconstructing the model with its details, however, is difficult. Without regularization, small noise on the data gets extremely amplified leading to models dominated by noise.

There are different approaches to tackle this problem; they differ by their assumptions, quality and computational cost. The de facto standard is the maximum entropy method (MaxEnt) which tries to find a model that balances between two terms: the fit to the data and the entropy relative to some default model. MaxEnt is efficient and produces good results in general [1]. However, it suffers from a smoothing effect, and it has parameters that need tuning [2]. Recently, another promising approach, the stochastic sampling method (StochS),[1] has gained momentum due to the increase in computational

[1] This method has different names in different papers: the stochastic method [3], statistical sampling method [2] and average spectrum method [4].

power. StochS averages over all non-negative models weighted by how well they fit the data. It is more computationally demanding than MaxEnt, but it has the potential of resolving sharp features and has no explicit parameters [2, 3, 4]. Stochastic sampling approaches are the main subject of this thesis.

In chapter 1, we study the analytic structure of Green and correlation functions explaining the origin of the analytic continuation problem. We derive the relations between the different Green and correlation functions and their spectral densities, and discuss many of their properties.

In chapter 2, we use the singular value decomposition to characterize the ill-posedness of the analytic continuation and why it is a challenging problem. We then present several regularization methods and compare them showing that non-negativity constraints are very helpful in solving the problem. We use non-negativity to develop a new regularization method: perturbed data sampling (PDS). We have implemented these algorithms as a toolbox of quick analytic continuation methods.

In chapter 3, we introduce Bayesian inference and formulate StochS and other methods in Bayesian terms. This provides a unified approach to the analytic continuation problem and sheds light on the assumptions employed by each method. Then, we present an efficient new algorithm for performing StochS: blocked modes sampling (BMS). In comparison to earlier sampling algorithms, BMS reduces the computational times by orders of magnitude. Using our fast algorithm, we find that StochS results depend on the discretization grid which acts as an implicit default model. This effect has not been discussed before in the literature. We study and explain the grid dependence and develop a procedure for the proper choice of the grid. To minimize the grid effect, we extended StochS into a gridless method (gStochS) by sampling the grid points explicitly from a default model instead of being fixed. This allows the grid points to move to best fit the data. We then extend gStochS further into eStochS to sample over a whole class of default models.

In chapter 4, we apply StochS and its extensions to realistic test cases demonstrating how to use them in practice. In appendix A, we give a general criterion for consistent Bayesian analytic continuation and reformulate the different stochastic sampling methods and MaxEnt in terms of stochastic processes to put them on an equal footing. In appendix B, we comment on the open problem of analytic continuation of non-diagonal spectral functions.

# 1. Analytic Structure of Green and Correlation Functions

In this chapter, we introduce the analytic continuation as a problem arising in condensed matter physics. We start with a brief introduction to the mathematical ideas of analytic functions and analytic continuation. The experiments of photoemission spectroscopy are then used to motivate the need for Green functions. We study the analytic structure of Green functions in both the time and frequency domains. Similarly, correlation functions are motivated by linear response theory, which relates them to the responses of physical systems to external perturbations. The analytic structure of correlation functions is identical to that of bosonic Green functions, so we provide the mapping between the two.

The analytic structure of both Green and correlation functions in the frequency domain shows that they are completely determined by their spectral densities. On the other hand, quantum Monte Carlo (QMC) simulations compute Green and correlation function values on the imaginary axis only. Determining the spectral function using QMC data is the problem of analytic continuation. We derive the relation between the two and show some useful properties of the spectral functions which provide important prior information about the solution. Finally, we formulate the analytic continuation problem as the mathematical problem of solving an integral equation whose unknown function is density-like i.e. both non-negative and integrable.

## 1.1. Mathematical preliminaries

We call a complex function, $f(z)$, **analytic** on an open set $\mathcal{D}$ if it can be written as a convergent power series in the neighborhood of any point $z_0 \in \mathcal{D}$

$$f(z) = \sum_{n=0}^{\infty} a_n (z - z_0)^n \; .$$

Analytic functions are special because their values in a large domain are completely determined by their values in any sub-domain. This follows directly from the identity theorem

**Identity Theorem.** *If two functions are analytic on some domain $\mathcal{V}$, and they agree on all points of an open subset[1] $\mathcal{U} \subset \mathcal{V}$, then they must agree on all points of $\mathcal{V}$.*

[1] This theorem can be further generalized to the case where $\mathcal{U}$ is any subset of $\mathcal{V}$ with an accumulation point in $\mathcal{V}$.

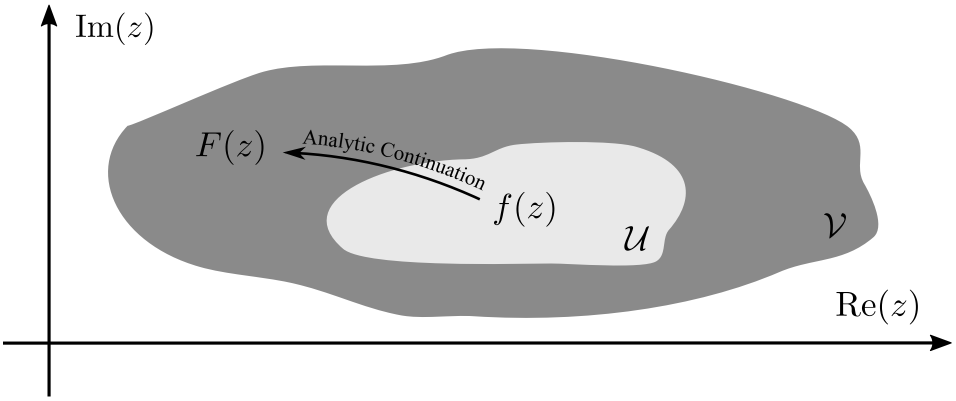


Figure 1.1.: $F(z)$ is the analytic continuation of $f(z)$ to the larger domain $\mathcal{V}$.

Therefore, if $f$ is some analytic function on $\mathcal{U}$, and $F$ is another analytic function on $\mathcal{V}$ such that $f(z) = F(z)$ for all $z \in \mathcal{U}$, then $F$ is unique and is called the **analytic continuation** of $f$ to $\mathcal{V}$.

An important consequence of the analytic continuation is the so-called *permanence of algebraic form* [5]. This means that if an analytic function on the real axis $f(x)$ is represented by some algebraic expression (e.g. a power series), then by replacing the real variable in this expression by a complex variable $x \to z$, the function $f(z)$ is the unique analytic continuation of $f(x)$ to the regions of the complex plane where that expression is still well-defined. More generally, if two functions $f_1(x)$ and $f_2(x)$ satisfy some algebraic relation, then their analytic continuations $f_1(z)$ and $f_2(z)$ satisfy this relation in their common domain of analyticity.

## 1.2. Green functions

### 1.2.1. Motivation: photoemission spectroscopy

The well-known photoelectric effect refers to the phenomenon that materials emit electrons when exposed to light of sufficiently high frequency. Photoemission spectroscopy (PES) experiments exploit this effect to infer the electronic structure of materials by measuring the energies of these emitted electrons. In such an experiment, a sample is exposed to light of specific frequency. If the energy of a photon is larger than a minimum threshold, it kicks an electron out of the sample, then a detector collects the ejected electrons and measures their kinetic energy. The binding energy of the emitted electrons can then be computed using conservation of energy. In angle-resolved PES, one also measures the direction of the ejected electron which determines its final momentum. Using conservation of momentum and certain assumptions,[2] the crystal momentum of the electron (its momentum when it was in the sample) can then be estimated [6].

[2] Only the component of the momentum parallel to the sample surface is conserved, so other assumptions are needed to determine the perpendicular component.

We are interested in the photocurrent $I_\kappa(\varepsilon)$ measured as a function of both energy $\varepsilon$ and momentum $\kappa$,[3] because it gives us a distribution reflecting the density of states in the sample (modified by some matrix elements, as we will see in Eq. 1.4). This current equals the probability per unit time of detecting an electron with momentum $\kappa$ and binding energy $\varepsilon$ upon exposing the material to a monochromatic light of frequency $\omega$. The electromagnetic field of the light is varying sinusoidally in time

$$E(t) = E_0 \left[\exp(-i\omega t) + \exp(i\omega\ t)\right] = 2E_0 \cos(\omega\ t)\ .$$

Taking the direction of the field in the x direction, the electrostatic energy of electrons in this field leads to the perturbing Hamiltonian

$$\hat{H}_\mathrm{p}(t) = E(t) \sum_i \hat{x}_i = \hat{V} \left[\exp(-i\omega\ t) + \exp(i\omega\ t)\right]\ ,$$

where the operator $\hat{V} = E_0 \sum_i \hat{x}_i$ is time-independent and called the *electric dipole approximation.* Applying first-order perturbation theory [7], we get Fermi's golden rule for calculating the transition rate (probability per unit time) from some initial state of the system $|\psi_i\rangle$ to some final state $|\psi_f\rangle$

$$P_{f\to i} = 2\pi \left|\langle\psi_f|\, \hat{V}\, |\psi_i\rangle\right|^2 \delta(E_f - E_i - \omega)\ ,$$

where $E_i$ is the initial energy of the system and $E_f$ is its final energy.

Assuming the system is initially in its ground state with $N$ electrons, the desired probability of detecting an electron with momentum $\kappa$ is the sum of transition rates to all possible final states with $N$ electrons, where one of them is the ejected electron with momentum $\kappa$

$$I_\kappa(\varepsilon) = 2\pi \sum_n |\,\langle\psi^N_{\kappa,n}|\hat{V}|\psi^N_0\rangle\,|^2 \delta(E^N_{\kappa,n} - E^N_0 - \omega)\ .$$

Using the sudden approximation, which assumes that the electron is ejected instantly without further interactions with the rest of the system, each final state can be written as a product of the ejected electron state and the state of the system with $N-1$ remaining electrons

$$|\psi^N_{\kappa,n}\rangle = \hat{c}^\dagger_\kappa\, |\psi^{N-1}_n\rangle\ ,$$

and the final energy equals the sum of the kinetic energy of the ejected electron and the energy of the rest of the system

$$E^N_{\kappa,n} = E^{N-1}_n + \varepsilon_\mathrm{kinetic}\ .$$

[3] We are using atomic units where the numerical values of the four fundamental constants: electron mass $m_e$, elementary charge $e$, reduced Planck's constant $\hbar$ and Coulomb's constant $\frac{1}{4\pi\epsilon_0}$ are unity. In atomic units, momentum and wavevector have the same numerical value, while energy and frequency have the same numerical value.

The kinetic energy of the electron is given by

$$\varepsilon_{\text{kinetic}} = \omega - \varepsilon - \phi \ ,$$

where $\phi$ is a threshold known as the work function.

Besides, the single-particle operator $\hat{V}$ can be written in second quantization as

$$\hat{V} = \sum_{i,j} V_{i,j}\, \hat{c}_i^\dagger \hat{c}_j \ .$$

Then we can rewrite the following matrix elements

$$\begin{aligned}\langle \psi_{\kappa,n}^N | \hat{V} | \psi_0^N \rangle &= \sum_{i,j} V_{i,j}\ \langle \psi_n^{N-1} | \hat{c}_\kappa \hat{c}_i^\dagger \hat{c}_j | \psi_0^N \rangle = \sum_{i,j} V_{i,j}\ \langle \psi_n^{N-1} | (\delta_{\kappa,i} - \hat{c}_i^\dagger \hat{c}_\kappa) \hat{c}_j | \psi_0^N \rangle \\ &= \sum_j V_{\kappa,j}\ \langle \psi_n^{N-1} | \hat{c}_j | \psi_0^N \rangle + \sum_{i,j} V_{i,j}\ \langle \psi_n^{N-1} | \hat{c}_i^\dagger \hat{c}_j \hat{c}_\kappa | \psi_0^N \rangle \ .\end{aligned}$$

Since the ground state typically has an extremely small contribution from high energy electrons (like the ejected one with momentum $\kappa$), the norm of the state $\hat{c}_\kappa \, |\psi_0^N\rangle$ is almost zero and the second term above can be neglected. This approximation gets better, the higher the frequency of the light source i.e. the higher the kinetic energy.

The photocurrent thus reads

$$I_\kappa(\varepsilon) = \sum_j |V_{\kappa,j}|^2 \left[ 2\pi \sum_n |\, \langle \psi_n^{N-1} | \hat{c}_j | \psi_0^N \rangle \,|^2 \delta(E_n^{N-1} - E_0^N - \varepsilon - \phi) \right] \ .$$

Using the integral representation of the delta function and the completeness of the eigenstates $|\psi_n\rangle$, we can rewrite the above expression as

$$\begin{aligned} I_\kappa(\varepsilon) &= \sum_j |V_{\kappa,j}|^2 \sum_n |\, \langle \psi_n^{N-1} | \hat{c}_j | \psi_0^{N-1} \rangle \,|^2 \int dt\, e^{-i(E_n^{N-1} - E_0^N - \varepsilon - \phi)t} && (1.1) \\ &= \sum_j |V_{\kappa,j}|^2 \int dt\, e^{i(\varepsilon+\phi)t} \sum_n \langle \psi_0 | e^{i\hat{H}t} \hat{c}_j^\dagger e^{-i\hat{H}t} | \psi_n \rangle \, \langle \psi_n | \hat{c}_j | \psi_0 \rangle && (1.2) \\ &= \sum_j |V_{\kappa,j}|^2 \int dt\, e^{i(\varepsilon+\phi)t} \, \langle \psi_0 | e^{i\hat{H}t} \hat{c}_j^\dagger e^{-i\hat{H}t} \hat{c}_j | \psi_0 \rangle && (1.3) \\ &= \sum_j |V_{\kappa,j}|^2 \int dt\, e^{i(\varepsilon+\phi)t} \, \langle \psi_0 | \hat{c}_j^\dagger(t) \hat{c}_j | \psi_0 \rangle \ . && (1.4) \end{aligned}$$

So, computing the photocurrent boils down to the Fourier transform of the quantity $\langle \psi_0 | \hat{c}_j^\dagger(t) \hat{c}_j | \psi_0 \rangle$. This quantity is called the **lesser Green function**, and there are many other types of Green functions which we will encounter later in this chapter. The utility of Green functions is not restricted to photoemission spectroscopy. They can also be used to calculate various interesting properties like the ground state of the system, its excitation spectrum and the expectation values of single-particle observables [8].

### 1.2.2. Green functions in the time domain

In the following, we explore the analytic structure of Green functions in the time domain. The basic building blocks are the greater and lesser Green functions, which are defined on the real axis and then analytically continued to the complex plane. We use these analytic continuations to build a single Green function of complex time, in terms of which we express all other Green functions.

#### Greater and lesser Green functions

The greater and lesser Green functions are the building blocks for other Green functions. They are defined in configuration space $x := (\mathbf{r}, \sigma)$ as following:

$$G^{>}(x, x', t, t') := -i \left\langle \hat{\Psi}_{\sigma}(\mathbf{r}, t) \hat{\Psi}^{\dagger}_{\sigma'}(\mathbf{r}', t') \right\rangle \tag{1.5}$$

$$G^{<}(x, x', t, t') := \pm i \left\langle \hat{\Psi}^{\dagger}_{\sigma'}(\mathbf{r}', t') \hat{\Psi}_{\sigma}(\mathbf{r}, t) \right\rangle \tag{1.6}$$

where $\hat{\Psi}$ and $\hat{\Psi}^{\dagger}$ are the field operators in the Heisenberg picture and $\langle ... \rangle$ is an expectation value to be defined next. The upper sign in the second equation is used for fermions while the lower one is used for bosons.[4] Note that the second equation follows from the first one by exchanging the field operators and the different signs for fermions and bosons arise naturally from their statistics.

There are two classes of Green functions, zero temperature and finite temperature. The previous definitions are valid for both and which is which depends on the interpretation of the expectation value.
At zero temperature, the system stays in its ground state $|\psi_0\rangle$ and all the information we need about an operator $\hat{O}$ at zero temperature is contained in the following expectation value

$$\left\langle \hat{O} \right\rangle = \frac{\langle \psi_0 | \hat{O} | \psi_0 \rangle}{\langle \psi_0 | \psi_0 \rangle}$$

On the other hand, at finite temperature the system is in a mixed state and the probability of finding the system at a specific energy level depends on the external constrains and determined by means of statistical mechanics. In this case, the expectation value should take into account both quantum and statistical averages.

We are interested in the so called *Grand Canonical Ensemble*, where the system is allowed to exchange not only energy, but also particles with the surrounding while being kept at fixed temperature $T$ and chemical potential $\mu$. With these constraints, the probability of finding the system at energy level $E$ with $N$ particles is proportional to $e^{-(E-\mu N)/(kT)}$.

[4] We will address both the fermionic and the bosonic case in most formulas simultaneously using the same convention.

In this ensemble the finite-temperature expectation value of the operator $\hat{O}$ is defined as

$$\left\langle \hat{O} \right\rangle = Z^{-1}\,\mathrm{Tr}\left[e^{-\beta(\hat{H}-\mu\hat{N})}\hat{O}\right] = Z^{-1}\sum_n \langle n|\, e^{-\beta(\hat{H}-\mu\hat{N})}\hat{O}\,|n\rangle$$
$$\text{where } Z = \sum_n \langle n|\, e^{-\beta(\hat{H}-\mu\hat{N})}\,|n\rangle \;\; .$$

The sum is over an orthonormal basis in Fock space (i.e. the sum runs over states with different particle numbers), $\hat{N}$ is the particle number operator and $\beta = 1/kT$ is the inverse temperature. In order for this expectation value to exist, the trace must converge absolutely.[5] Therefore, the operator $\hat{H}$ must be bounded from below (i.e. there exists a ground state, which is the case for physical Hamiltonians) and we assume that this is sufficient to guarantee the necessary absolute convergence.

Note that we can retrieve the zero temperature definition by taking the limit $\beta \to \infty$. In this limit, only the state with the lowest energy survives the exponentially damping factor and we are left with the ground state. This assumes, that the ground state is non-degenerate, otherwise we end up with an equally weighted average of the degenerate ground states.

**Remark.** *In the grand-canonical ensemble, it is convenient to modify the definition of the Heisenberg operators as follows*

$$\hat{O}(t) \equiv e^{i(\hat{H}-\mu\hat{N})t}\hat{O}e^{-i(\hat{H}-\mu\hat{N})t} \;.$$

*This has the advantage of using the same operator $\hat{H} - \mu\hat{N}$ for both time evolution and thermal averaging. It will produce the same results as if we had used $\hat{H}$ for time evolution instead, as long as the Hamiltonian $\hat{H}$ preserves the number of particles (which is assumed to be the case) and the operator $\hat{O}$ also does not change the number of particles (which is the case for combinations of paired $\hat{\psi}$ and $\hat{\psi}^\dagger$). The first property means that $\hat{H}$ commutes with $\hat{N}$, so the exponentials can be factorized into two terms $e^{\pm i\hat{H}t}$ and $e^{\mp i\mu\hat{N}t}$. The second property means that $e^{\mp i\mu\hat{N}t}$ commute with $\hat{O}$, so we can combine them getting the unity operator and we are back to the original definition of Heisenberg operators. However, for Green functions, the operator $\hat{O}$ is a field operator so it does not commute with $\hat{N}$. Had we used only $\hat{H}$ for the time evolution instead, we would have had a different version of Green functions $\tilde{G}$ which can be related to the our version by a phase factor*

$$\tilde{G}^{>}(x,x',t,t') = e^{-i\mu(t-t')}\;G^{>}(x,x',t,t')$$
$$\tilde{G}^{<}(x,x',t,t') = e^{\;i\mu(t-t')}\;G^{<}(x,x',t,t')\;.$$

*Keep in mind that this simply corresponds to a shift by $\mu$ in the frequency domain i.e. this modification corresponds to measuring single-particle energies with respect to the chemical potential.*

[5] Remember that the expectation value of a random variable $x$ exists and equals to $\mathrm{E}[x] = \sum_{i=1}^{\infty} p_i x_i$ if and only if this series converges absolutely i.e. the sum of the absolute value of the summand is finite. Otherwise, the series can be rearranged to have different limits!

*Using this modification, only $\hat{H} - \mu\hat{N}$ appears in Heisenberg operators. Therefore, it is convenient shorten the notation in the grand-canonical ensemble and use $\hat{H}$ to actually denote $\hat{H} - \mu\hat{N}$ and $E_n$ to denote $E_n - \mu N$. This allows us to handle both the zero and finite temperature cases with the same notation.*

**Physical interpretation** For $t > t'$, the greater Green function can be interpreted as the probability amplitude for detecting a particle with spin $\sigma$ at position $\mathbf{r}$ and time $t$ after adding a particle with spin $\sigma'$ at position $\mathbf{r}'$ and time $t'$. The lesser Green function has no physical interpretation in this case. On the other hand, for $t' > t$, the lesser Green function can be interpreted as the probability amplitude for detecting a hole with spin $\sigma'$ at position $\mathbf{r}'$ and time $t'$ after removing a particle with spin $\sigma$ from position $\mathbf{r}$ and time $t$. The greater Green function does not have a physical interpretation in this case. By using the greater Green function in the earlier case and the lesser one in the later case, we get the causal Green function (see Eq. 1.23).

**Different basis** The field operators can be used to define creation and annihilation operators in any single-particle basis $\psi_\kappa(x)$

$$\begin{aligned} \hat{c}^\dagger_\kappa &= \int dx\ \psi_\kappa(x)\ \hat{\Psi}^\dagger(x) \\ \hat{c}_\kappa &= \int dx\ \psi^*_\kappa(x)\ \hat{\Psi}(x)\ , \end{aligned}$$

This allows us to rewrite the Green functions in configuration space as

$$G^>(x,x',t,t') = \sum_\kappa \sum_{\kappa'} \psi_\kappa(x)\psi^*_{\kappa'}(x')G^>_{\kappa,\kappa'}(t,t') \tag{1.7}$$

$$G^<(x,x',t,t') = \sum_\kappa \sum_{\kappa'} \psi_\kappa(x)\psi^*_{\kappa'}(x')G^<_{\kappa,\kappa'}(t,t')\ , \tag{1.8}$$

where the Green functions in this single-particle basis are defined as following

$$\begin{aligned} G^>_{\kappa,\kappa'}(t,t') &:= -i\left\langle \hat{c}_\kappa(t)\hat{c}^\dagger_{\kappa'}(t')\right\rangle \\ G^<_{\kappa,\kappa'}(t,t') &:= \pm i\left\langle \hat{c}^\dagger_{\kappa'}(t')\hat{c}_\kappa(t)\right\rangle\ . \end{aligned}$$

The physical interpretation of these Green functions is the same as above except that they are talking about particles in states $\kappa, \kappa'$ instead of $x, x'$.

**Time dependence** Assuming the Hamiltonian is time-independent, only time differences matter and the Green functions are functions of one time variable only

$$G^>_{\kappa,\kappa'}(t) := -i\left\langle \hat{c}_\kappa(t)\hat{c}^\dagger_{\kappa'}(0)\right\rangle \tag{1.9}$$

$$G^<_{\kappa,\kappa'}(t) := \pm i\left\langle \hat{c}^\dagger_{\kappa'}(0)\hat{c}_\kappa(t)\right\rangle\ . \tag{1.10}$$

In the rest of this chapter, we will use this version of Green functions which are written in a basis $\psi_\kappa$ and are functions of one time variable. Sometimes, we will drop the basis indices $\kappa, \kappa'$ to make the formulas less cluttered. In this case, such a basis is implicitly assumed.

### Analytic continuation of $G^>(t)$ and $G^<(t)$

Assuming the greater and lesser Green functions are analytic on the real axis, it is possible to analytically continue them to the imaginary axis using the replacement $it \to \tau$ (this is known as Wick rotation), where $\tau$ is called the imaginary time. More generally, we can analytically continue these functions to other regions of the complex plane by naively replacing the real time $t$ in Eqs. (1.9) and (1.10) with the complex time $\zeta := t - i\tau$, where $t$ represents real and $\tau$ imaginary time (the minus sign is introduced to agree with Wick rotation):

$$G^>_{\kappa,\kappa'}(\zeta) := -i \left\langle \hat{c}_\kappa(\zeta)\hat{c}^\dagger_{\kappa'}(0) \right\rangle \tag{1.11}$$

$$G^<_{\kappa,\kappa'}(\zeta) := \pm i \left\langle \hat{c}^\dagger_{\kappa'}(0)\hat{c}_\kappa(\zeta) \right\rangle \ . \tag{1.12}$$

In order for this simple substitution (replacing the real time by a complex one) to give the correct answer at some point $\zeta_0$, it should be possible to connect $\zeta_0$ to the real axis by some arc that do not cross any singularity. Fortunately, as we will show below, $G^>(\zeta)$ is bounded on the strip

$$\mathcal{D}^> = \{\zeta : -\beta < \mathrm{Im}(\zeta) < 0\} = \{t - i\tau : 0 < \tau < +\beta,\ t \in \mathbb{R}\}\ , \tag{1.13}$$

and $G^<(\zeta)$ is bounded on the strip

$$\mathcal{D}^< = \{\zeta : 0 < \mathrm{Im}(\zeta) < +\beta\} = \{t - i\tau : -\beta < \tau < 0,\ t \in \mathbb{R}\}\ . \tag{1.14}$$

This guarantees that the greater and lesser Green functions do not have singularities in the corresponding strips and thus $G^>(\zeta)$ and $G^<(\zeta)$ are the unique analytic continuation of $G^>(t)$ and $G^<(t)$ from the real axis to $\mathcal{D}^>$ and $\mathcal{D}^<$, respectively.

In the definition of the greater Green function for real time, we implicitly assumed that the factor $e^{-\beta\hat{H}}$ guarantees the absolute convergence of the expectation value

$$\left\langle e^{-\beta\hat{H}} e^{it\hat{H}} \hat{c}_\kappa e^{-it\hat{H}} \hat{c}^\dagger_{\kappa'} \right\rangle \ .$$

This implies that the expectation value with complex time $\zeta = t - i\tau$

$$\left\langle e^{-\beta\hat{H}} e^{i\zeta\hat{H}} \hat{c}_\kappa e^{-i\zeta\hat{H}} \hat{c}^\dagger_{\kappa'} \right\rangle = \left\langle e^{-(\beta-\tau)\hat{H}} e^{it\hat{H}} \hat{c}_\kappa e^{-it\hat{H}} e^{-\tau\hat{H}} \hat{c}^\dagger_{\kappa'} \right\rangle \tag{1.15}$$

must also converge absolutely when both the exponential factors $e^{-(\beta-\tau)\hat{H}}$ and $e^{-\tau\hat{H}}$ have negative exponents. This is satisfied for $\tau \in ]0, \beta[$, and therefore $G^>(\zeta)$ is bounded and analytic on the strip $\mathcal{D}^>$. Similarly, we can show that the mere existence of the lesser Green function on the real axis implies that $G^<(\zeta)$ is bounded and analytic on the strip $\mathcal{D}^<$. The analytic structure of $G^>(\zeta)$ and $G^<(\zeta)$ is depicted schematically in Fig. 1.2.

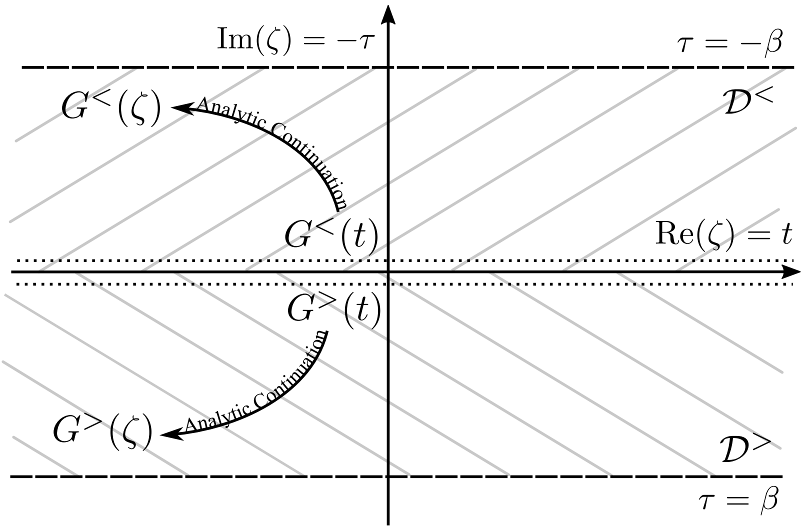


Figure 1.2.: Analytic structure of $G^{>}(\zeta)$ and $G^{<}(\zeta)$ in the complex time domain $\zeta = t - i\tau$. The greater Green function is analytic on the strip $\mathcal{D}^{>}$ while the lesser Green function is analytic on the strip $\mathcal{D}^{<}$. Note that counter-intuitively, the greater and lesser Green functions live in the lower and upper half-planes, respectively. This is because we introduced the minus sign in $\zeta := t - i\tau$ in order to agree with Wick rotation and the convention used in the literature.

**Remark.** *Note that the analytic continuation of $G^{>}$ and $G^{<}$ to strips $\mathcal{D}^{>}$ and $\mathcal{D}^{<}$, respectively, is all we can say in the general case. In certain cases, those functions can be analytically continued further outside those regions. For example, for systems with a finite basis set (like those arising in the linear combination of atomic orbitals approximation), the expectation value in Eq.* (1.15) *is finite and exists for any $\tau$, thus $G^{>}(\zeta)$ is defined in the entire complex plane of $\zeta$. However, as we will see later, knowing the Green functions in the aforementioned strips (or even smaller parts of them like the imaginary axis) is all we need.*

**Relating $G^{>}(\zeta)$ and $G^{<}(\zeta)$** The analytically continued greater and lesser Green functions satisfy an important (anti)periodicity relation along imaginary times

$$\begin{aligned} G^{>}_{\kappa,\kappa'}(\zeta - i\beta) &= \frac{-i}{Z}\,\mathrm{Tr}\left[e^{-\beta\hat{H}} e^{i(\zeta-i\beta)\hat{H}} \hat{c}_{\kappa} e^{-i(\zeta-i\beta)\hat{H}} \hat{c}^{\dagger}_{\kappa'}\right] \\ &= \frac{-i}{Z}\,\mathrm{Tr}\left[e^{i\zeta\hat{H}} \hat{c}_{\kappa} e^{-i\zeta\hat{H}} e^{-\beta\hat{H}} \hat{c}^{\dagger}_{\kappa'}\right] \\ &= \frac{-i}{Z}\,\mathrm{Tr}\left[e^{-\beta\hat{H}} \hat{c}^{\dagger}_{\kappa'} e^{i\zeta\hat{H}} \hat{c}_{\kappa} e^{-i\zeta\hat{H}}\right] \\ &= \mp G^{<}_{\kappa,\kappa'}(\zeta)\,, \end{aligned} \tag{1.16}$$

where the cyclic property of trace was employed in the third line.

**Zero limits of $G^{>}(\zeta)$ and $G^{<}(\zeta)$** Due to analyticity, the limit $\zeta \to 0$ of greater and lesser Green functions is independent of the way zero is approached, as long as $\zeta$ stays in the proper region of analyticity

$$\lim_{\zeta\to 0} G^{<}_{\kappa,\kappa'}(\zeta) = \pm i \left\langle \hat{c}^{\dagger}_{\kappa'} \hat{c}_{\kappa} \right\rangle$$
$$\lim_{\zeta\to 0} G^{>}_{\kappa,\kappa'}(\zeta) = -i \left(1 \mp \left\langle \hat{c}^{\dagger}_{\kappa'} \hat{c}_{\kappa} \right\rangle\right) \ .$$

When $\kappa' = \kappa$, the limits are related to $n_\kappa = \left\langle \hat{c}^{\dagger}_{\kappa'} \hat{c}_{\kappa} \right\rangle$, the density of particles in state $\kappa$, as following

$$\lim_{\zeta\to 0} G^{<}_{\kappa,\kappa}(\zeta) = -i \left(\mp n_\kappa\right) \tag{1.17}$$
$$\lim_{\zeta\to 0} G^{>}_{\kappa,\kappa}(\zeta) = -i \left(1 \mp n_\kappa\right) \ . \tag{1.18}$$

**Green function of complex time $G(\zeta)$**

The analytic structure of $G^{>}(\zeta)$ and $G^{<}(\zeta)$ and their relation motivates us to combine them into one function:

$$G(\zeta) := \begin{cases} G^{>}(\zeta), & \text{for } \zeta \in \mathcal{D}^{>} \\ G^{<}(\zeta), & \text{for } \zeta \in \mathcal{D}^{<} \end{cases} \ . \tag{1.19}$$

This function has some interesting properties:

**Analyticity** It is only analytic on each of the strips $\mathcal{D}^{>}$ and $\mathcal{D}^{<}$ *separately* because it has a jump $[G^{>}(t) - G^{<}(t)]$ over the real axis. The jump over the real time axis is called the **spectral function** and defined with an extra $i$ factor as

$$A(t) := i \left[G^{>}(t) - G^{<}(t)\right] \ , \tag{1.20}$$

which is nothing but the expectation value of the (anti)commutator of annihilation and creation operators at different times

$$A_{\kappa,\kappa'}(t) = \left\langle \left[\hat{c}_\kappa(t), \hat{c}^{\dagger}_{\kappa'}(0)\right]_{\pm} \right\rangle = \left\langle \hat{c}_\kappa(t)\hat{c}^{\dagger}_{\kappa'}(0) \pm \hat{c}^{\dagger}_{\kappa'}(0)\hat{c}_\kappa(t) \right\rangle \ .$$

The spectral function will be important later when we move to the frequency domain.

**Boundedness** Since $G^{>}(\zeta)$ and $G^{<}(\zeta)$ are bounded on strips $\mathcal{D}^{>}$ and $\mathcal{D}^{<}$, respectively, the function $G(\zeta)$ is bounded in on its domain.

**(Anti)periodicity** It is (anti)periodic with period $\beta$ along any line parallel to the imaginary axis. This follows directly from Eq. (1.16) relating the greater and lesser Green functions. Therefore, It is possible to extend the definition of $G(\zeta)$ beyond

$\mathcal{D}^> \cup \mathcal{D}^<$ by repeating $G^>$ and $G^<$ alternatively over strips of width $\beta$ . However, it is important to note that this (anti)periodicity exists "naturally" only in the region $\mathcal{D}^> \cup \mathcal{D}^<$ and is only imposed by definition outside it. The reason is that, when the analytic continuation of $G(\zeta)$ is possible outside this region, then the analytically continued function, in general, won't be (anti)periodic.

Despite these nice properties, one may still ask: why do we bother with defining this extra function? The answer, as we will see in the next section, is that the values of the commonly used Green functions can be related to the values of $G(\zeta)$ over different contours. Furthermore, the Fourier transform of each of the previously defined Green functions can be computed as the Fourier transform of $G(\zeta)$ along some contour.

*Therefore, we can say that the function $G(\zeta)$ is the fundamental function in time domain and all other Green functions are just different faces of this function.*

### Relating $G(\zeta)$ to other Green functions

Besides the greater and lesser Green functions, there are many other ones. The retarded and advanced Green functions are defined as

$$G^R_{\kappa,\kappa'}(t) := -i\theta(t)\left\langle \left[\hat{c}_\kappa(t), \hat{c}^\dagger_{\kappa'}(0)\right]_\pm \right\rangle \tag{1.21}$$

$$G^A_{\kappa,\kappa'}(t) := i\theta(-t)\left\langle \left[\hat{c}_\kappa(t), \hat{c}^\dagger_{\kappa'}(0)\right]_\pm \right\rangle \ . \tag{1.22}$$

As we will see later, the Fourier transforms of these functions turn out to be very important in the frequency domain.

The causal and anti-causal Green functions are defined as

$$G^C_{\kappa,\kappa'}(t) := -i\left\langle T\left(\hat{c}_\kappa(t), \hat{c}^\dagger_{\kappa'}(0)\right)\right\rangle \tag{1.23}$$

$$G^{AC}_{\kappa,\kappa'}(t) := -i\left\langle \tilde{T}\left(\hat{c}_\kappa(t), \hat{c}^\dagger_{\kappa'}(0)\right)\right\rangle \ , \tag{1.24}$$

where $T$ is the time-ordering operator which orders its arguments in chronological order from right to left

$$T\left(\hat{A}(t), \hat{B}(t')\right) := \begin{cases} \hat{A}(t)\,\hat{B}(t') & \text{if } t' < t \\ \mp\hat{B}(t')\,\hat{A}(t) & \text{if } t' > t \end{cases} \tag{1.25}$$

and $\tilde{T}$ is the anti-time-ordering operator which orders its arguments in chronological order from left to right. The causal Green function has a physical interpretation as it describes the propagation of an additional particle for $t > 0$, and the propagation of an additional hole for $t < 0$. The anti-causal Green function is mentioned here for completeness as the opposite of the causal one.

Finally, the Matsubara Green function is defined as

$$\mathcal{G}_{\kappa,\kappa'}(\tau) := -\left\langle T_\tau\left(\hat{c}_\kappa(\tau), \hat{c}^\dagger_{\kappa'}(0)\right)\right\rangle \tag{1.26}$$

where $\tau$ is the imaginary time with $\tau \in ]-\beta, 0[\cup]0, \beta[$, $T_\tau$ is the imaginary-time-ordering operator which orders its arguments in increasing order of $\tau$ from right to left, and the imaginary-time operator $\hat{c}_\kappa(\tau)$ is defined as $e^{\hat{H}\tau}\hat{c}_\kappa e^{-\hat{H}\tau}$. Matsubara Green function is the one which is often calculated using Monte Carlo simulations, because it is generally smooth and has nicer properties than Green functions of real time (remember that under Wick rotation $it \to \tau$, the oscillatory factor $e^{-it\hat{H}}$ becomes convergent $e^{-\tau\hat{H}}$).

These Green functions can be related to the greater and lesser ones as following

$$\begin{aligned} G^R(t) = &\quad \theta(+t)\ [G^>(t) - G^<(t)] && (1.27)\\ G^A(t) = &\quad \theta(-t)\ [G^<(t) - G^>(t)] && (1.28)\\ G^C(t) = &\quad \theta(-t)\, G^<(t) + \theta(+t)\, G^>(t) && (1.29)\\ G^{AC}(t) = &\quad \theta(-t)\, G^>(t) + \theta(+t)\, G^<(t) && (1.30)\\ \mathcal{G}(\tau) = -i\, [&\ \theta(-\tau)\, G^<(-i\tau) + \theta(+\tau)\, G^>(-i\tau)]\ , && (1.31) \end{aligned}$$

from which we see that each one of them can be related to $G(\zeta)$ along some contour. Take for example the Matsubara Green function whose values equal the values of $G(\zeta)$ along the imaginary axis (up to a factor $-i$).[6] Other Green functions can be identified as $G(\zeta)$ along contours that are infinitesimally close to the real axis. These contours differ by how they approach the real axis for positive and negative times. For example, the contour of the causal Green function is above the real axis for negative times and below it for positive ones. Fig. 1.3 shows the different contours corresponding to the different Green functions.

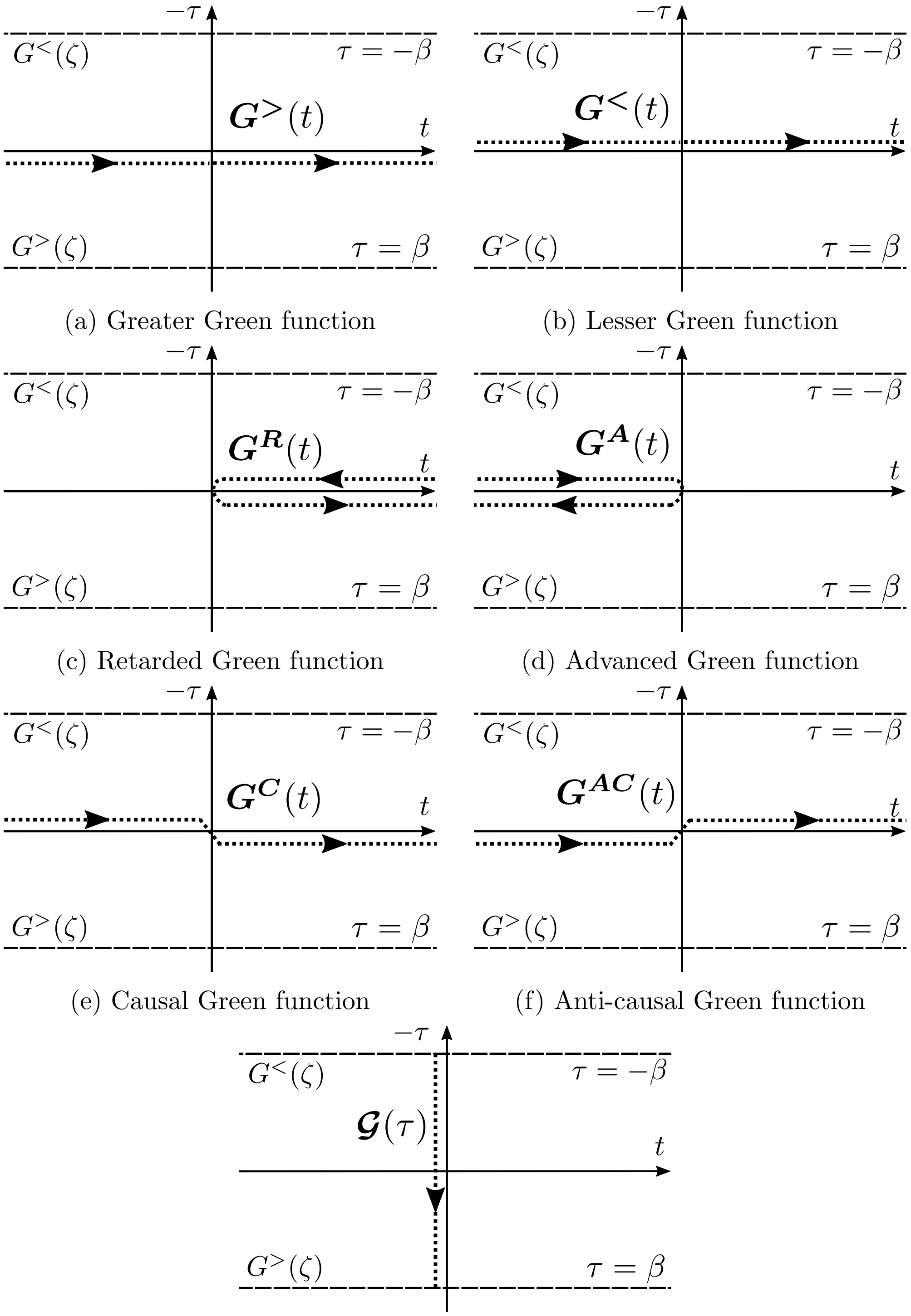
This unified view of the different Green functions as different faces of the fundamental Green function $G(\zeta)$ is quit useful. It allows us to infer easily many properties of the different Green functions and their relations. For example, comparing Figs. 1.3e and 1.3g shows that rotating the contour of the causal Green function by 90 degrees clockwise gives us the Matsubara function; This is nothing but the Wick rotation! Notice also how the imaginary time ordering, which may seem quite arbitrary in the definition of the Matsubara Green function, arises naturally as the result of moving along the imaginary axis. Moreover, the zero limits of Matsubara Green function follow directly from Eqs. (1.17) and (1.18)

$$\begin{aligned} \lim_{\tau\to 0^-} \mathcal{G}_{\kappa,\kappa}(\tau) &= \pm n_\kappa && (1.32)\\ \lim_{\tau\to 0^+} \mathcal{G}_{\kappa,\kappa}(\tau) &= -1 \pm n_\kappa\ , && (1.33) \end{aligned}$$

and the $\beta$ limits follow directly from the (anti)periodicity relation (see Eq. 1.16)

$$\begin{aligned} \lim_{\tau\to \beta} \mathcal{G}_{\kappa,\kappa}(\tau) &= \mp \lim_{\tau\to 0^-} \mathcal{G}_{\kappa,\kappa}(\tau) = -n_\kappa && (1.34)\\ \lim_{\tau\to -\beta} \mathcal{G}_{\kappa,\kappa}(\tau) &= \mp \lim_{\tau\to 0^+} \mathcal{G}_{\kappa,\kappa}(\tau) = \pm 1 - n_\kappa\ . && (1.35) \end{aligned}$$

[6] This factor comes back automatically when computing contour integrals because $d\zeta$ equals $-id\tau$ along the imaginary axis.



(a) Greater Green function

(b) Lesser Green function

(c) Retarded Green function

(d) Advanced Green function

(e) Causal Green function

(f) Anti-causal Green function

(g) Matsubara Green function

Figure 1.3.: The different Green functions of time can be seen as contours of $G(\zeta)$ in the complex time plane $\zeta = t - i\tau$. Notice that the contours (a)-(f) are shifted infinitesimally above or below the real axis because $G(\zeta)$ is undefined on the real axis. However, the contour (g) of the Matsubara Green function is actually *on* the imaginary axis and the apparent shift is for visual clarity only.

### 1.2.3. Green functions in the frequency domain

Now that we are done with the analytic structure in the time domain, we move to the frequency domain. We take the Fourier transforms of the different Green functions and see how they are related. The Fourier transforms of retarded and advanced Green functions turn out to be the basic building blocks in the frequency domain. We analytically continue them to the complex frequency plane, and use that to build a single Green function of complex frequency, in terms of which we express all other Fourier transforms.

#### Fourier Transform of $G^>$ and $G^<$

The greater Green function is bounded over the real time axis, so its Fourier transform exists and equals

$$G^>(\omega) = \int_{-\infty}^{\infty} dt\, e^{i\omega t}\, G^>(t)\ .$$

Its analytic continuation $G^>(\zeta)$ is also bounded and has a Fourier transform along any line parallel to the real axis within the strip $\mathcal{D}^>$. Interestingly, this transform is the same for every line $\mathcal{L}_1$ regardless of its shift i.e.

$$\int_{\mathcal{L}_1} d\zeta\, e^{iw\zeta}\, G^>(\zeta) = \int_{-\infty}^{\infty} dt\, e^{i\omega(t-i\tau)}\, G^>(t-i\tau) = \int_{-\infty}^{\infty} dt\, e^{i\omega t}\, G^>(t) = G^>(\omega)\ . \quad (1.36)$$

This can be readily seen by performing the integral along the contour shown in Fig. 1.4. The total contour integral equals zero because the function is analytic in the enclosed region. By making the left and right limits go to infinity, contributions from side integrals vanish and thus the integrals over the real axis and line $\mathcal{L}_1$ are equal.

Similarly, we can find the Fourier transform of the lesser Green function over real time axis

$$G^<(\omega) = \int_{-\infty}^{\infty} dt\, e^{i\omega t}\, G^<(t)\ ,$$

and it equals its transform along any line $\mathcal{L}_2$ within the strip $\mathcal{D}^<$ and parallel to the real axis

$$\int_{\mathcal{L}_2} d\zeta\, e^{iw\zeta}\, G^<(\zeta) = \int_{-\infty}^{\infty} dt\, e^{i\omega t}\, G^<(t) = G^<(\omega)\ . \quad (1.37)$$

**Relating $G^>(\omega)$ and $G^<(\omega)$** The (anti)periodicity relation of greater and lesser Green functions in the time domain (see Eq. 1.16) leads to the following relation in the frequency domain

$$\begin{aligned} G^>(\omega) &= \int\limits_{\mathcal{L}_1} d\zeta\, e^{iw\zeta}\, G^>(\zeta) = \mp \int\limits_{\mathcal{L}_1} d\zeta\, e^{iw\zeta}\, G^<(\zeta + i\beta) \\ &= \mp \int\limits_{\mathcal{L}_1 - i\beta} d\zeta\, e^{iw(\zeta - i\beta)}\, G^<(\zeta) = \mp e^{\beta\omega} \int\limits_{\mathcal{L}_1 - i\beta} d\zeta\, e^{iw\zeta}\, G^<(\zeta) \\ &= \mp e^{\beta\omega}\, G^<(\omega)\ . \end{aligned} \quad (1.38)$$

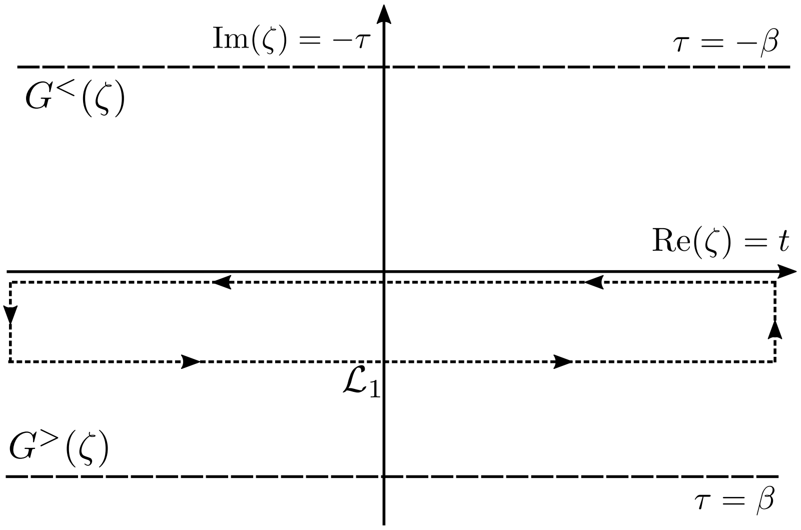


Figure 1.4.: The contour used to compute the Fourier transform of $G^>(\zeta)$ along the line $\mathcal{L}_1$.

### Fourier transforms of other Green functions

From the previous section, we see that $G^>(\omega)$ is the Fourier transform of $G(\zeta)$ along any line parallel to the real axis in $\mathcal{D}^>$ and $G^<(\omega)$ is its Fourier transform along any line parallel to the real axis in $\mathcal{D}^<$. But why to stop at such paths?
Since $G(\zeta)$ is bounded, it has a Fourier transform along any path in its domain $\mathcal{D}^> \cup \mathcal{D}^<$. Different paths lead to different Fourier transforms, and the most interesting paths are the ones corresponding to the previously defined Green functions (see Fig. 1.3). The plan now is to compute the Fourier transforms along these paths and see how they are connected in the frequency domain.

The first connection point is the Fourier transform of the spectral function $A(\omega)$, in terms of which we will express the different Fourier transforms. As we will see later, $A(\omega)$ is the central quantity in the frequency domain and computing it from other quantities is *the analytic continuation problem* in condensed matter physics.

**Greater and lesser functions** The Fourier transforms of $G(\zeta)$ along paths (a) and (b) of Fig. 1.3 give the Fourier transforms of the greater and lesser functions, respectively

$$\int_{\mathcal{C}_a} d\zeta\; e^{iw\zeta}\, G(\zeta) = \int_{-\infty}^{\infty} dt\; e^{iwt}\, G^>(t) = G^>(\omega)$$
$$\int_{\mathcal{C}_b} d\zeta\; e^{iw\zeta}\, G(\zeta) = \int_{-\infty}^{\infty} dt\; e^{iwt}\, G^<(t) = G^<(\omega)\ .$$

We can relate them to the Fourier transform of the spectral function by taking the Fourier transform of both sides of Eq. (1.20):

$$A(\omega) = i\left[G^>(\omega) - G^<(\omega)\right]\ .$$

But $G^{>}(\omega)$ and $G^{<}(\omega)$ are related by Eq. (1.38), therefore

$$\begin{aligned} G^{>}(\omega) &= -i\frac{A(\omega)}{1 \pm e^{-\omega\beta}} \\ G^{<}(\omega) &= \quad i\frac{A(\omega)}{1 \pm e^{\omega\beta}} \ . \end{aligned}$$

Defining the weight function

$$f(\omega) \coloneqq \frac{1}{1 \pm e^{-\omega\beta}} \ , \tag{1.39}$$

the Fourier transforms of greater and lesser Green functions read

$$G^{>}(\omega) = -iA(\omega)f(\omega) \tag{1.40}$$

$$G^{<}(\omega) = \quad iA(\omega)f(-\omega) \ . \tag{1.41}$$

**Remark.** *The weight function $f(\omega)$ is closely related to the particle statistics*

$$f(\omega) = \pm\frac{1}{e^{-\omega\beta} \pm 1} = \pm n(\omega) \ .$$

*The upper sign gives the Fermi-Dirac distribution $n_{\text{F-D}}$ while the lower sign gives minus the Bose-Einstein distribution $-n_{\text{B-E}}$. This function satisfies the following relations*

$$f(-\omega) = \frac{1}{1 \pm e^{\omega\beta}} = \frac{e^{-\omega\beta}}{e^{-\omega\beta} \pm 1} = \pm e^{-\omega\beta} f(-\omega) \tag{1.42}$$

$$f(\omega) + f(-\omega) = f(\omega) \pm e^{-\omega\beta} f(-\omega) = f(\omega)[1 \pm e^{\omega\beta}] = 1. \tag{1.43}$$

**Retarded and advanced functions** The Fourier transform along paths (c) and (d) of Fig. 1.3 give the Fourier transforms of the retarded and advanced functions, respectively

$$\begin{aligned} \int_{\mathcal{C}_c} d\zeta \, e^{iw\zeta}\, G(\zeta) &= \int_{-\infty}^{0} dt \, e^{iwt}\, G^{<}(t) + \int_{0}^{\infty} dt \, e^{iwt}\, G^{>}(t) \\ &= \int_{0}^{\infty} dt \, e^{iwt} \, [G^{>}(t) - G^{<}(t)] = G^{R}(\omega) \\ \int_{\mathcal{C}_d} d\zeta \, e^{iw\zeta}\, G(\zeta) &= \int_{-\infty}^{0} dt \, e^{iwt}\, G^{<}(t) + \int_{0}^{-\infty} dt \, e^{iwt}\, G^{>}(t) \\ &= \int_{-\infty}^{0} dt \, e^{iwt} \, [G^{<}(t) - G^{>}(t)] = G^{A}(\omega) \end{aligned}$$

We can relate $G^{R}(\omega)$ to the Fourier transform of the spectral function by taking the Fourier transform of both sides of the following equation (see Eq. 1.27)

$$G^{R}(t) = -i\theta(t)A(t) \ .$$

and writing $A(t)$ as the inverse Fourier transform of $A(\omega)$

$$\begin{aligned}
G^R(\omega) &= -i \int_0^\infty dt\, e^{i\omega t} A(t) \\
&= -i \int_0^\infty dt\, e^{i\omega t} \int \frac{d\omega'}{2\pi}\, e^{-i\omega' t} A(\omega') \\
&= -i \int \frac{d\omega'}{2\pi}\, A(\omega') \int_0^\infty dt\, e^{i(\omega-\omega')t} \\
&= - \int \frac{d\omega'}{2\pi}\, A(\omega') \left[ \frac{e^{i(\omega-\omega')t}}{\omega-\omega'} \right]_{t=0}^{t\to\infty} .
\end{aligned}$$

The upper limit does not exist for real frequencies, but it goes to zero if the frequency has an infinitesimally small positive imaginary part $+i\eta$. So we replace $\omega$ by $\omega + i\eta$ and get the Fourier transform of the retarded Green function

$$G^R(\omega) = \int \frac{d\omega'}{2\pi} \frac{A(\omega')}{\omega + i\eta - \omega'} . \tag{1.44}$$

Similarly, we can relate the Fourier transform of the advanced Green function to the Fourier transform of the spectral function

$$G^A(\omega) = \int \frac{d\omega'}{2\pi} \frac{A(\omega')}{\omega - i\eta - \omega'} . \tag{1.45}$$

**Remark.** *Since multiplication in the time domain becomes convolution in the frequency domain, we can also derive the previous results as the convolution of the Fourier transforms of the spectral function and the step function. The Fourier transform of the step function reads*

$$\int dt\; e^{i\omega t}\, \theta(t) = \frac{i}{\omega + i\eta} , \tag{1.46}$$

*where $\eta$ is an infinitesimally small positive number. This can be verified by evaluating the inverse Fourier transform*

$$\int \frac{d\omega}{2\pi}\, e^{-i\omega t}\, \frac{i}{\omega + i\eta} .$$

*When $t > 0$, the integral can be closed in the lower half-plane of complex $\omega$ enclosing a pole of residue $i$ and the expression evaluates to one using the residue theorem. When $t < 0$, the integral can be closed in the upper half-plane where no pole exists and the expression evaluates to zero. The above expression is also known as the integral representation of the step function.*

**Causal and anti-causal functions** The Fourier transforms of $G(\zeta)$ along paths (e) and (f) of Fig. 1.3 give the Fourier transforms of the causal and anti-causal functions, respectively

$$\int_{\mathcal{C}_e} d\zeta\; e^{iw\zeta}\, G(\zeta) = \int_{-\infty}^{0} dt\, e^{i\omega t} G^{<}(t) + \int_{0}^{\infty} dt\, e^{i\omega t} G^{>}(t) = G^{C}(\omega)$$
$$\int_{\mathcal{C}_f} d\zeta\; e^{iw\zeta}\, G(\zeta) = \int_{-\infty}^{0} dt\, e^{i\omega t} G^{>}(t) + \int_{0}^{\infty} dt\, e^{i\omega t} G^{<}(t) = G^{AC}(\omega)\ .$$

We can relate $G^C(\omega)$ to the Fourier transform of the spectral function by writing $G^>(t)$ and $G^<(t)$ as the inverse Fourier transform of $G^>(\omega)$ and $G^<(\omega)$ and using Eqs. (1.40) and (1.41)

$$\begin{aligned} G^{C}(\omega) &= \int_{-\infty}^{0} dt\, e^{i\omega t} G^{<}(t) + \int_{0}^{\infty} dt\, e^{i\omega t} G^{>}(t) \\ &= i \int_{-\infty}^{0} dt\, e^{i\omega t} \int \frac{d\omega'}{2\pi} e^{-i\omega' t} A(\omega') f(-\omega') \\ &\quad - i \int_{0}^{\infty} dt e^{i\omega t} \int \frac{d\omega'}{2\pi}\, e^{-i\omega' t} A(\omega') f(\omega')\ . \end{aligned}$$

Applying the same trick used for retarded and advanced functions, adding $i\eta$ to the frequency for positive times and $-i\eta$ for negative times, we get

$$G^{C}(\omega) = \int \frac{d\omega'}{2\pi} \frac{A(\omega')}{\omega - i\eta - \omega'}\; f(-\omega') + \int \frac{d\omega'}{2\pi} \frac{A(\omega')}{\omega + i\eta - \omega'}\; f(\omega')\ . \tag{1.47}$$

Similarly, the Fourier transform of the anti-causal Green function can be related to the Fourier transform of the spectral function

$$G^{AC}(\omega) = \int \frac{d\omega'}{2\pi} \frac{A(\omega')}{\omega - i\eta - \omega'}\; f(\omega') + \int \frac{d\omega'}{2\pi} \frac{A(\omega')}{\omega + i\eta - \omega'}\; f(-\omega')\ . \tag{1.48}$$

**Matsubara Green function** Since $G(\zeta)$ is (anti)periodic along the imaginary axis, the Matsubara Green function is also (anti)periodic with period $\beta$. This means that it can be written as a Fourier series

$$\mathcal{G}(\tau) = \frac{1}{\beta} \sum_{\omega_n} e^{-i\omega_n \tau} \mathcal{G}_n$$

whose coefficients $\mathcal{G}_n$ can be computed as the Fourier transforms

$$\mathcal{G}_n = \int_0^{\beta} d\tau\, e^{i\omega_n \tau} \mathcal{G}(\tau)\ ,$$

at the so-called **the Matsubara frequencies** $\omega_n$ given by

$$\begin{aligned} \omega_n :=& \frac{(2n+1)\pi}{\beta}\ \text{(Fermions)} \\ \omega_n :=& \frac{2n\pi}{\beta}\ \text{(Bosons)}\ . \end{aligned} \tag{1.49}$$

Those coefficients are related to the Fourier transform of $G(\zeta)$ along the imaginary axis (path (g) of Fig. 1.3) for imaginary frequencies $i\omega_n$

$$\begin{aligned}
\mathcal{G}_n &= \int_0^\beta d\tau\, e^{i\omega_n\tau}\mathcal{G}(\tau) = \frac{1}{2}\int_{-\beta}^{\beta} d\tau\, e^{i\omega_n\tau}\mathcal{G}(\tau) \\
&= -\frac{i}{2}\int_{-\beta}^{0} d\tau\, e^{i(i\omega_n)(-i\tau)}G^<(-i\tau) - \frac{i}{2}\int_0^\beta d\tau\, e^{i(i\omega_n)(-i\tau)}G^>(-i\tau) \\
&= \frac{1}{2}\int_{\mathcal{C}_g} d\zeta\, e^{i(i\omega_n)\zeta}\, G(\zeta)
\end{aligned}$$

We can relate the coefficient $\mathcal{G}_n$ to the Fourier transform of the spectral function by employing the inverse Fourier transform of $G^>(-i\tau)$

$$\begin{aligned}
\mathcal{G}_n &= -i\int_0^\beta d\tau\, e^{i\omega_n\tau}G^>(-i\tau) \\
&= -i\int_0^\beta d\tau\, e^{i\omega_n\tau}\int\frac{d\omega'}{2\pi}e^{-i\omega'(-i\tau)}G^>(\omega') \\
&= -\int\frac{d\omega'}{2\pi}A(\omega')f(\omega')\int_0^\beta d\tau\, e^{(i\omega_n-\omega')\tau} \\
&= -\int\frac{d\omega'}{2\pi}A(\omega')f(\omega')\left[\frac{e^{(i\omega_n-\omega)\tau}}{i\omega_n-\omega'}\right]_{\tau=0}^{\tau=\beta} \\
&= \int\frac{d\omega'}{2\pi}A(\omega')\,\frac{1}{1\pm e^{-\omega'\beta}}\,\frac{1-e^{i\omega_n\beta}e^{-\omega'\beta}}{i\omega_n-\omega'}\ .
\end{aligned}$$

Since $\beta\omega_n$ is an odd(even) multiple of $\pi$ for fermions(bosons), $e^{i\omega_n\beta} = \mp 1$ and we get

$$\mathcal{G}_n = \int\frac{d\omega'}{2\pi}\frac{A(\omega')}{i\omega_n-\omega'} \tag{1.50}$$

**Remark.** *For bosonic systems, the Matsubara frequency $\omega_0 = 0$ lies on the real axis. In this case, $A(\omega)$ should vanish at $\omega = 0$ at least as fast as $\omega$, in order for $\mathcal{G}_0$ to be defined, and we have $\mathcal{G}_0 = G^R(0) = G^A(0) = \int\frac{d\omega}{2\pi}\frac{-A(\omega)}{\omega}$.*

### Analytic continuation of $G^R(\omega)$ and $G^A(\omega)$

When computing the Fourier transform of the retarded Green function, the transform didn't exist for real frequencies, and we had to add a positive infinitesimal imaginary part to the frequency to obtain a meaningful result. Therefore, a different derivation would have been to compute the Fourier transform directly for complex frequencies[7] $z$

[7] Typically Fourier transform is only defined for real frequencies. Its analytic continuation to complex frequencies is rather called the **Laplace transform**. However, since the name "Laplace transform" is not very common in this context, we will use the name "Fourier transform" whether we have real or complex frequencies.

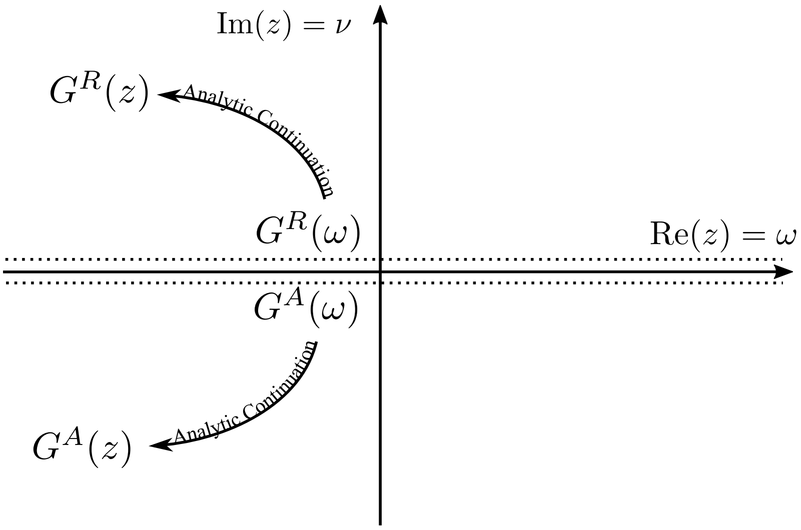


Figure 1.5.: The analytic structure of $G^R(z)$ and $G^A(z)$ in the complex frequency domain $z = \omega + i\nu$. The retarded Green function is analytic in the upper half-plane while the advanced Green function is analytic in the lower half-plane.

and then take the limit to real frequencies. This "extended" Fourier transform reads

$$G^R(z) = \int dt\, e^{izt} G^R(t) = -i \int_0^\infty dt\, e^{izt} A(t) \ .$$

Since $A(t)$ is bounded, the last integral exists when $\mathrm{Im}(z) > 0$. Therefore, the Fourier transform of the retarded Green function exits for frequencies in the upper half-plane and it equals

$$G^R(z) = \int \frac{d\omega}{2\pi} \frac{A(\omega')}{z - \omega} \qquad : \mathrm{Im}(z) > 0 \ .$$

$G^R(\omega)$ is clearly the limit of $G^R(z)$ to real frequencies. Conversely, since $G^R(z)$ is analytic in the upper half-plane and it equals $G^R(\omega)$ on a line in that region, $G^R(z)$ is the unique analytic continuation of $G^R(\omega)$ to the upper-half plane.

Similarly, the Fourier transform of the advanced Green function exits for frequencies in the lower half-plane and it has the same functional form

$$G^A(z) = \int \frac{d\omega}{2\pi} \frac{A(\omega')}{z - \omega} \qquad : \mathrm{Im}(z) < 0 \ .$$

So $G^A(\omega)$ is the limit of $G^A(z)$ to real frequencies, and $G^A(z)$ is the analytic continuation of $G^A(\omega)$ to the lower-half plane. The analytic structure of $G^R(z)$ and $G^A(z)$ is depicted schematically in Fig 1.5.

**Green function of complex frequency** $G(z)$

The identical functional form of $G^R(z)$ and $G^A(z)$ motivates us to combine them into one function

$$G(z) := \int \frac{d\omega}{2\pi} \frac{A(\omega)}{z-\omega} \, , \tag{1.51}$$

which has the following properties

**Analyticity** It is analytic in the upper half-plane and the lower half-plane *separately* because it has a jump $G^R(\omega)-G^A(\omega)$ over the real axis. To compute this difference

$$G^R(\omega) - G^A(\omega) = \int \frac{d\omega'}{2\pi} \frac{A(\omega')}{\omega + i\eta - \omega'} - \int \frac{d\omega'}{2\pi} \frac{A(\omega')}{\omega - i\eta - \omega'}$$

we use the identity

$$\frac{1}{\omega \pm i\eta} = \mathcal{P}\frac{1}{\omega} \mp \pi i \delta(\omega) \, , \tag{1.52}$$

where $\mathcal{P}$ denotes the principal value.

$$\begin{aligned} G^R(\omega) - G^A(\omega) = {} & \mathcal{P}\int \frac{d\omega'}{2\pi} \frac{A(\omega')}{\omega - \omega'} - \frac{\pi i}{2\pi} A(\omega) \\ & - \mathcal{P}\int \frac{d\omega'}{2\pi} \frac{A(\omega')}{\omega - \omega'} - \frac{\pi i}{2\pi} A(\omega) \\ & = -iA(\omega) \, . \end{aligned}$$

Therefore, the jump over the real axis equals (up to a $-i$ factor) the spectral function

$$A(\omega) = i\left[G^R(\omega) - G^A(\omega)\right] \, . \tag{1.53}$$

**Decay Behavior** It goes to zero, as $z$ goes to infinity in any line in the upper or lower half of the complex plane. More specifically, it decays as $1/z$. This follows from the boundedness of $A(t)$ because

$$\lim_{|z|\to\infty} G(z) = \lim_{|z|\to\infty} \int \frac{d\omega}{2\pi} \frac{A(\omega)}{z-\omega} = \frac{\int \frac{d\omega}{2\pi} A(\omega)}{z} = \frac{A(t=0)}{z}$$

Like in the time domain, one may ask: why do we bother with defining this extra function? The answer again is that we can express the different Fourier transforms in terms of this function as we will do in the next section.

*Therefore, we can say that the function $G(z)$ is the fundamental function in frequency domain and the Fourier transforms of all the other Green functions are just different faces of this function.*

### Relating $G(z)$ to other Fourier transforms

Let us start with the advanced and retarded functions $G^R(\omega)$ and $G^A(\omega)$, which can be seen as the limits of $G(z)$ when approaching the real axis from above and below, respectively

$$G^R(\omega) = \lim_{\eta\to 0} G(\omega + i\eta), \quad G^A(\omega) = \lim_{\eta\to 0} G(\omega - i\eta) \ .$$

Other Fourier transforms can be expressed in terms of $G^R(\omega)$ and $G^A(\omega)$ or their analytic continuation, and so they can be written in terms of $G(z)$.

**Greater and lesser functions** Substituting Eq. (1.53) in Eqs. (1.40)-(1.41), we get

$$G^>(\omega) = f(\omega)\left[G^R(\omega) - G^A(\omega)\right] \tag{1.54}$$

$$G^<(\omega) = -f(-\omega)\left[G^R(\omega) - G^A(\omega)\right] \ . \tag{1.55}$$

**Causal and anti-causal functions** Applying identity (1.52) to Eq. (1.47), we can write

$$\begin{aligned} G^C(\omega) &= \mathcal{P}\int \frac{d\omega'}{2\pi}\frac{A(\omega')}{\omega-\omega'}\left[f(-\omega')+f(\omega')\right] + \frac{\pi i}{2\pi}\left[f(-\omega)-f(\omega)\right]A(\omega) \\ &= f(-\omega)\left[\mathcal{P}\int \frac{d\omega'}{2\pi}\frac{A(\omega')}{\omega-\omega'} + \frac{\pi i}{2\pi}A(\omega)\right] \\ &\quad + f(\omega)\left[\mathcal{P}\int \frac{d\omega'}{2\pi}\frac{A(\omega')}{\omega-\omega'} - \frac{\pi i}{2\pi}A(\omega)\right] \ , \end{aligned}$$

where the trivial result $f(\omega) + f(-\omega) = 1 = f(\omega') + f(-\omega')$ was used (see Eq. 1.43). Applying the aforementioned identity in reverse, we get

$$G^C(\omega) = f(-\omega)\int \frac{d\omega'}{2\pi}\frac{A(\omega')}{\omega - i\eta - \omega'} + f(\omega)\int \frac{d\omega'}{2\pi}\frac{A(\omega')}{\omega + i\eta - \omega'} \ , \tag{1.56}$$

which can be expressed in terms of retarded and advanced functions as

$$G^C(\omega) = f(\omega)G^R(\omega) + f(-\omega)G^A(\omega) \ . \tag{1.57}$$

Similarly, the Fourier transform of the anti-causal Green function can be written as

$$G^{AC}(\omega) = f(\omega)G^A(\omega) + f(-\omega)G^R(\omega) \ . \tag{1.58}$$

**Matsubara Green function** Comparing the Fourier coefficients of the Matsubara Green function Eq. (1.50) with the definition of $G(z)$, we find that those coefficients are just $G(z)$ evaluated on the imaginary axis at Matsubara frequencies

$$\mathcal{G}_n = G(i\omega_n) \ . \tag{1.59}$$

The fact that $G(z)$ and $\mathcal{G}_n$ have similar analytical formulas may seem puzzling at first glance, but it is actually a direct result of the analyticity and (anti)periodicity of $G(\zeta)$!

It would not have been surprising, had we calculated coefficients $\mathcal{G}_n$ in a different way, which we will do next. The coefficients $\mathcal{G}_n$ can be calculated as the Fourier transform of $G(\zeta)$ along the imaginary axis for imaginary frequency $i\omega_n$

$$\begin{aligned}\mathcal{G}_n &= -i\int_0^\beta d\tau\, e^{i\omega_n\tau}G^>(-i\tau) = \int_0^{-i\beta} d(-i\tau)\, e^{i(i\omega_n)(-i\tau)}G^>(-i\tau)\\ &= \int_{\mathcal{C}_1} d\zeta\, e^{i(i\omega_n)\zeta}G^>(\zeta)\end{aligned}$$

For positive $\omega_n$, we can close this contour as shown in Fig. 1.6 . The total contour integral is zero because the enclosed region has no poles, while the contribution from the right contour $\mathcal{C}_3$ vanishes for large times, and we have

$$\int_{\mathcal{C}_1} d\zeta\, e^{i(i\omega_n)\zeta}G^>(\zeta) = -\int_{\mathcal{C}_2} d\zeta\, e^{i(i\omega_n)\zeta}G^>(\zeta) - \int_{\mathcal{C}_4} d\zeta\, e^{i(i\omega_n)\zeta}G^>(\zeta)\ .$$

But since $e^{i\omega_n\beta} = \mp 1$ and $G^>(\zeta - i\beta) = \mp G^<(\zeta)$, we can calculate the $\mathcal{C}_2$ integral as a contour integral just above the real axis, and we have

$$\mathcal{G}_n = -\int_{\mathcal{C}_5} d\zeta\, e^{i(i\omega_n)\zeta}G^<(\zeta) - \int_{\mathcal{C}_4} d\zeta\, e^{i(i\omega_n)\zeta}G^>(\zeta)\ ,$$

which is nothing but the contour integral for the retarded Green function (compare to Fig. 1.3c). As a result, the Matsubara Fourier coefficients for positive Matsubara frequencies equal the Fourier transform of the retarded Green function at the corresponding imaginary frequencies. Similarly for negative $\omega_n$, we can close the contour in the third quadrant and relate $\mathcal{G}_n$ to the Fourier transform of the advanced Green function.

### Uniqueness of $G(z)$

Knowing the retarded Green function $G^R(\omega)$ is sufficient to determine $G(z)$ uniquely in the upper half-plane. This follows immediately from the identity theorem because $G^R(\omega)$ provides $G(z)$ values on a line infinitesimally above the real axis which qualifies as a set with an accumulation point in the upper half plane. Similarly, knowing the advanced one $G^A(\omega)$ is sufficient to determine $G(z)$ uniquely in the lower half-plane.

Interestingly, knowing $\mathcal{G}_n$ on the Matsubara frequencies is also sufficient to determine $G(z)$ uniquely in both the upper and lower half-planes. This is less obvious than the previous case because Matsubara frequencies form a discrete set with no accumulation point. The key features in spelling out $G(z)$ as the unique analytic function, that matches $\mathcal{G}_n$ at Matsubara frequencies, are its two aforementioned properties: analyticity in the upper and lower half-planes, and decay to zero at infinity. Ref. [9] provides a proof of the uniqueness of $G(z)$ given its values at Matsubara frequencies. In the following, we provide an alternative proof of the same result.

Suppose that there are two functions $G_1(z)$ and $G_2(z)$ which are analytic on the upper and lower half-planes, separately. Suppose also that they decay to zero when $z$ approaches infinity and that their values are equal at Matsubara frequencies $z_n = i\omega_n$.

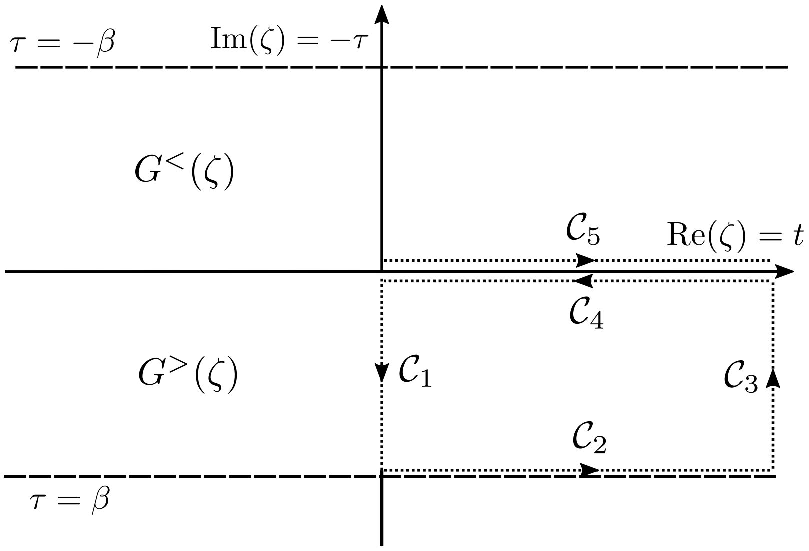


Figure 1.6.: Matsubara Fourier coefficients $\mathcal{G}_n$ can be computed as the Fourier transform of $G(\zeta)$ along $\mathcal{C}_1$ for complex frequencies $i\omega_n$. Due to analyticity and (anti)periodicity of $G(\zeta)$, this equals to the Fourier transform along contours $-\mathcal{C}_4$ and $-\mathcal{C}_5$ (when $\omega_n$ is positive). These contours are equivalent to the contour of the retarded Green function.

Then, their difference $F(z) := G_1(z) - G_2(z)$ is also analytic on the two half-planes and has zeros at $z_n$. Due to the decay of $G_1$ and $G_2$, this function is bounded in any region away from the real axis[8], so the shifted function $E(z) := F(z + 2\pi/\beta)$ is bounded on the upper half plane.[9] The next step is to prove that $E(z)$ is identically zero in the upper half plane.

Consider the following transformation from the open unit disk to the upper half-plane

$$\phi : y \to z = i\,\frac{1+y}{1-y}\ .$$

Using this transformation, we compose the following function

$$\tilde{E}(y) := E(\phi(y))\ ,$$

which is analytic and bounded on the unit disk and has zeros at points $y_n$ inside the unit disk. The set of zeros $y_n$ is the preimage of positive Matsubara frequencies under the above transformation

$$y_n = \phi^{-1}(i\omega_n) = \frac{i\omega_n - i}{i\omega_n + i} = \frac{\omega_n - 1}{\omega_n + 1}\ ,$$

[8] The functions $G_1$ and $G_2$ are analytic on the upper and lower half-planes separately, so they may diverge near the real axis.

[9] The shift is chosen such that new function does not not diverge in the upper half plan and has zeros at the Matsubara frequencies.

and it has 1 as an accumulation point. If this accumulation point were inside the unit disk, we would be able to invoke the identity theorem and the function $\tilde{E}$ would be zero as desired. Unfortunately, the accumulation point lies on the unit circle, so we need another theorem that makes use also of the boundedness of $\tilde{E}$. Theorem 15.23 in Ref. [10] states that if a function is analytic and bounded on the open unit disk, and is not identically zero, then $(\sum_n 1 - |y_n|) < \infty$. But this sum diverges for our function $\tilde{E}(y)$

$$\sum_{n=1}^{\infty}(1 - |y_n|) = \sum_{n=1}^{\infty} \frac{2}{\omega_n + 1} = \infty ,$$

so this function must be identically zero in the unit disk. Then, the original function $E(z)$, and consequently the difference $F(z)$, must also be identically zero in upper half-plane, implying that $G_1$ and $G_2$ are identical in the upper half-plane. Similarly, we can prove that $G_1$ and $G_2$ are identical in the lower half-plane. This completes the proof that the function $G(z)$ which is analytic off the real axis and decays to zero at infinity is uniquely determined by its values at the Matsubara frequencies.

We should emphasize that the decay behavior at infinity is an essential ingredient for the uniqueness of $G(z)$. For example, $G(ze^{\beta z})$ agrees with $G(z)$ on all bosonic Matsubara frequencies but the two are completely different functions. As another example, take the Fourier transform of the greater Green function $G^{>}(\zeta)$ along the imaginary axis, and extend it to complex frequencies $z$. Working out the calculation, we get the function

$$\mathcal{G}(z) = \int \frac{d\omega}{2\pi} \frac{A(\omega)}{z - \omega} \frac{1 - e^{z\beta} e^{-\omega\beta}}{1 \pm e^{-\omega\beta}} ,$$

which has exactly the same values as $G(z)$ at Matsubara frequencies, but which is otherwise different from $G(z)$. These examples, however, do not contradict the above result, because those other functions do not decay to zero when $z$ approaches infinity.

### 1.2.4. Back to the time domain

In this section, we want to close the loop and go back to the time domain by computing the inverse Fourier transforms of the different Green functions. Remember that we were able to express all the different Green functions of frequency as Fourier transforms of $G(\zeta)$ along different contours in the complex time domain (see Fig. 1.3). The question now is whether we can do the same with $G(z)$ i.e. can we express each Green function of time as an inverse Fourier transform of $G(z)$ along some path in the complex frequency domain?

Clearly, the retarded $G^R(t)$ and advanced $G^A(t)$ Green functions can be computed respectively as the inverse Fourier transforms of $G(z)$ along lines above and below the real axis and infinitesimally close to it (see Fig. 1.5). Apart from these two functions, the answer is *no*![10] Nevertheless, we can express other Green functions of time as the inverse Fourier transforms of functions related to $G(z)$, namely $f(z)G(z)$ and $-f(-z)G(z)$. A

[10] It is possible though in the zero temperature limit, which is discussed later (see Fig. 1.9).

key component in this formulation is the function $f(z)$, which is the analytic continuation of the weight function $f(\omega)$ (see Eq. 1.39)

$$f(z) := \frac{1}{1 \pm e^{-z\beta}} . \tag{1.60}$$

Both $f(z)$ and $-f(-z)$ have poles on the imaginary axis at Matsubara frequencies $\omega_n$ with residues $1/\beta$.

In Fig. 1.7 and Fig. 1.8, we present the different functions whose inverse Fourier transforms along the depicted contours give the different Green functions of time. By noting that the contour $\mathcal{C}_2$ in subfigures (c)-(d) in nothing but a deformation of the contour $\mathcal{C}_1$ in regions of analyticity, we can say the following: While the different Green functions of frequency can be expressed as the Fourier transforms of the *same function* $G(\zeta)$ along *different contours*, the different Green functions of time can be expressed as the inverse Fourier transforms of *different functions* along the *same contour* $\mathcal{C}_1$ .

**Remark.** *Fig. 1.7 shows the contours for fermionic systems. For bosonic systems, these contours would catch (or miss) the pole of $f(z)$ at Matsubara frequency $i\omega_n = 0$. Therefore, we modify those contours slightly in the bosonic case as shown in Fig. 1.8. One may then object, that the bosonic contours miss (or catch) the pole of $G(z)$ at $\omega = 0$. This not a problem, however, when the spectral function of bosonic systems $A(\omega)$ vanishes at zero (see the remark following Eq.* (1.50)*).*

In the following, we will prove that the different Green functions of time are the suggested inverse Fourier transforms in the complex frequency plane.

**Greater and lesser functions** Let us start with the greater Green function $G^>(t)$ which can be computed as the inverse Fourier transform of $f(z)G(z)$ along the contour $\mathcal{C}_1$ around the real axis (see Fig. 1.7a)

$$\mathcal{I}_1(t) := \int_{\mathcal{C}_1} \frac{dz}{2\pi} e^{-izt} f(z)G(z) = \int \frac{d\omega}{2\pi} A(\omega) \int_{\mathcal{C}_1} \frac{dz}{2\pi} e^{-izt} \left[\frac{f(z)}{z-\omega}\right] .$$

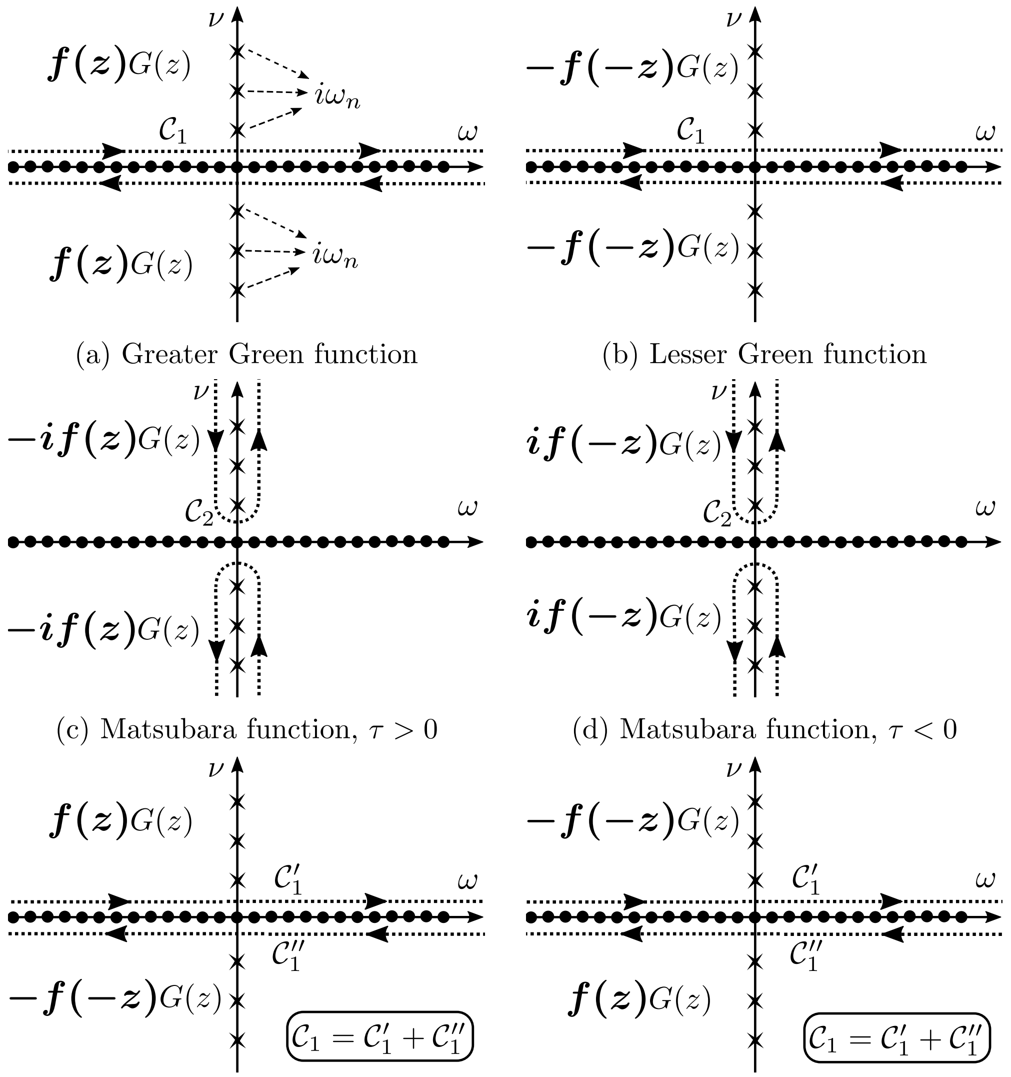
The contour $\mathcal{C}_1$ catches the pole at $\omega$ and we have the expected result

$$\mathcal{I}_1(t) = \int \frac{d\omega}{2\pi} A(\omega) \left[\frac{-2\pi i}{2\pi} e^{i\omega t} f(\omega)\right] = -i \int \frac{d\omega}{2\pi} e^{i\omega t} f(\omega) A(\omega) = G^>(t) .$$

Similarly, the lesser Green function $G^<(t)$ can be computed as the inverse Fourier transform of $-f(-z)G(z)$ along the same contour (see Fig. 1.7b).

**Matsubara function** For positive $\tau$, the Matsubara Green function $\mathcal{G}(\tau)$ can be computed as the inverse Fourier transform of $-if(z)G(z)$ along the contour $\mathcal{C}_2$ around the imaginary axis (see Fig. 1.7c)

$$\mathcal{I}_2(\tau) := -i \int_{\mathcal{C}_2} \frac{dz}{2\pi} e^{-iz(-i\tau)} f(z)G(z) \ : \tau > 0 .$$



(a) Greater Green function

(b) Lesser Green function

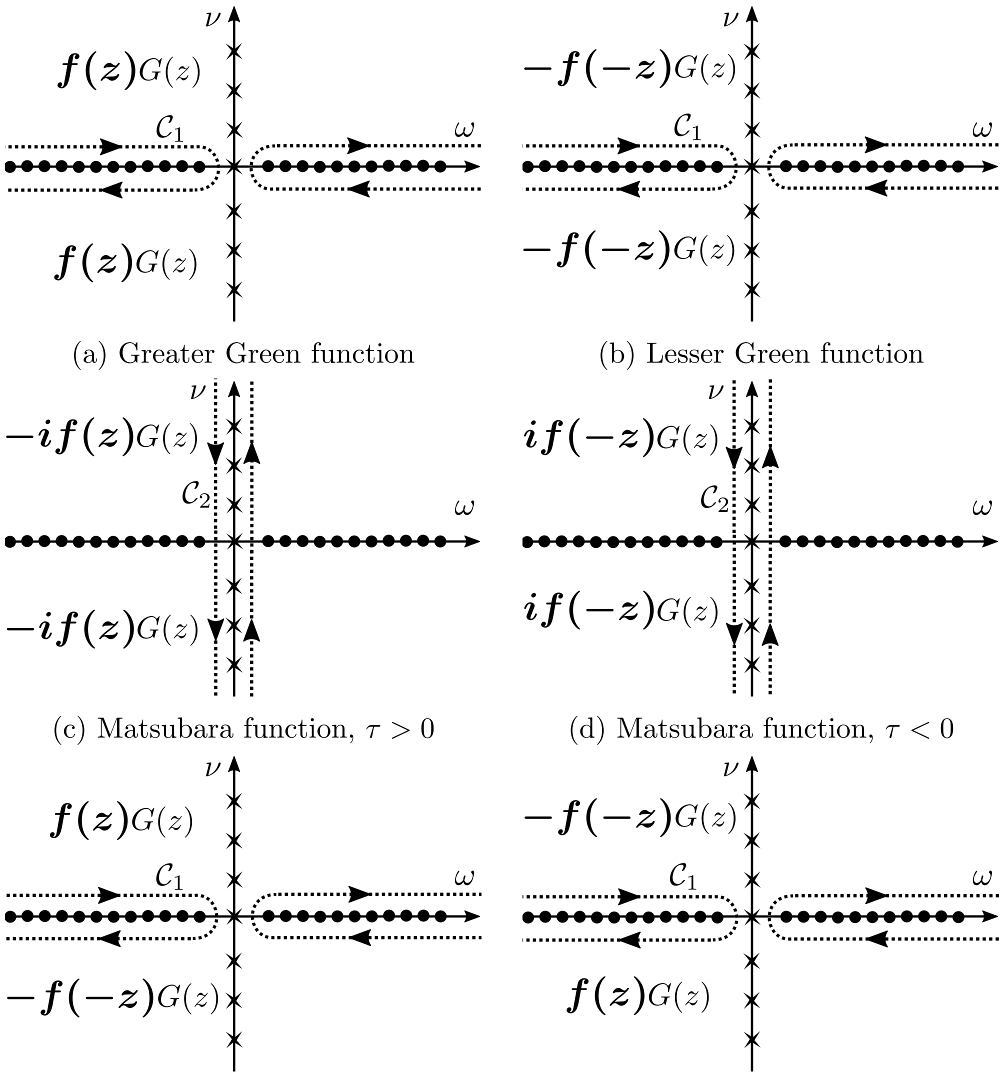
(c) Matsubara function, $\tau > 0$

(d) Matsubara function, $\tau < 0$

(e) Causal Green function

(f) Anti-causal Green function

Figure 1.7.: *Fermionic* contours for calculating different Green functions as inverse Fourier transforms of functions of $G(z)$ in the complex frequency plane $z = \omega + i\nu$. The contour $\mathcal{C}_1$ is shifted infinitesimally above and below the real axis while the contour $\mathcal{C}_2$ is shifted infinitesimally to the left and right of the imaginary axis. Notice that the contour $\mathcal{C}_2$ can be thought of as a deformation of the contour $\mathcal{C}_1$ in regions of analyticity.



(a) Greater Green function

(b) Lesser Green function

(c) Matsubara function, $\tau > 0$

(d) Matsubara function, $\tau < 0$

(e) Causal Green function

(f) Anti-causal Green function

Figure 1.8.: *Bosonic* contours for calculating different Green functions as inverse Fourier transforms of functions of $G(z)$ in the complex frequency plane $z = \omega + i\nu$. Those contours are slightly modified from the fermion contours (Fig. 1.7) in order to avoid (or include) the Matsubara frequency at zero. The gap of the contour $\mathcal{C}_1$ around zero is infinitesimally small.

The contour $\mathcal{C}_2$ catches the poles of $f(z)$ at Matsubara frequencies and we have the desired result

$$\mathcal{I}_2(\tau) = -i \sum_{\omega_n} \frac{2\pi i}{2\pi\beta} e^{-i\omega_n\tau} G(i\omega_n) = \frac{1}{\beta} \sum_{\omega_n} e^{-i\omega_n\tau} G(i\omega_n) = \mathcal{G}(\tau) \; .$$

Similarly, for negative $\tau$, the Matsubara Green function can be computed as the inverse Fourier transform of $if(-z)G(z)$ along the same contour (see Fig. 1.7d).

**Remark.** *It is interesting to see the analyticity interplay between time and frequency domains. Let us reevaluate the contour integral $\mathcal{I}_2$. Since the integrand is analytic inside each quadrant and vanishes for large values, the contour $\mathcal{C}_2$ around the imaginary axis can be deformed into contour $\mathcal{C}_1$ around the real axis giving*

$$\begin{aligned} &\mathcal{I}_2(\tau) = -i \int_{\mathcal{C}_1} \frac{dz}{2\pi} e^{-iz(-i\tau)} f(z)G(z) = -i\mathcal{I}_1(-i\tau) \Rightarrow \\ &\mathcal{G}(\tau) = -iG^{>}(-i\tau) : \; \tau > 0 \; . \end{aligned}$$

*So the analyticity and decay of $f(z)G(z)$ leads directly to the analyticity of $G^{>}(\zeta)$. Similarly, the analyticity and decay of $-f(-z)G(z)$ leads directly to the analyticity of $G^{<}(\zeta)$.*

**Causal and anti-causal functions** By multiplying $G(z)$ with $f(z)$ in the upper half-plane and $-f(-z)$ in the lower half-plane, we can get the causal Green function $G^C(t)$ as the inverse Fourier transform along contour $\mathcal{C}_1$ around the real axis (see Fig. 1.7e)

$$\begin{aligned} \mathcal{I}_3(t) &:= \int_{\mathcal{C}_1'} \frac{dz}{2\pi} e^{-izt} f(z)G(z) - \int_{\mathcal{C}_1''} \frac{dz}{2\pi} e^{-izt} f(-z)G(z) \\ &= \int \frac{d\omega}{2\pi} A(\omega) \left[ \int_{\mathcal{C}_1'} \frac{dz}{2\pi} e^{-izt} \frac{f(z)}{z-\omega} - \int_{\mathcal{C}_1''} \frac{dz}{2\pi} e^{-izt} \frac{f(-z)}{z-\omega} \right] \; . \end{aligned}$$

When $t > 0$, both contours can be closed in the lower half-plane. The first contour encloses the pole at $\omega$ and poles of $f(z)$ at negative Matsubara frequencies while the second contour encloses only poles of $-f(-z)$ at negative Matsubara frequencies. The residuals at Matsubara frequencies resulting from the first term cancel with those resulting from the second term, and we are left with

$$\mathcal{I}_3(t) = \int \frac{d\omega}{2\pi} A(\omega) \frac{-2\pi i}{2\pi} e^{i\omega t} f(\omega) = G^{>}(t) \;\; : t > 0 \; .$$

When $t < 0$, both contours can be closed in the upper half-plane. The second contour encloses the pole at $\omega$ and poles of $-f(-z)$ at positive Matsubara frequencies while the second contour encloses only poles of $f(z)$ at positive Matsubara frequencies. The residuals at Matsubara frequencies resulting from the first term cancel with those resulting from the second term, and we are left with

$$\mathcal{I}_3(t) = \int \frac{d\omega}{2\pi} A(\omega) \frac{2\pi i}{2\pi} e^{i\omega t} f(-\omega) = G^{<}(t) \;\; : t < 0 \; .$$

Therefore, we have the desired result

$$\mathcal{I}_3(t) = \theta(t)G^>(t) + \theta(-t)G^<(t) = G^C(t) \ .$$

Similarly, the anti-causal Green function can be written as the inverse Fourier transform of a function whose values in the upper half-plane equal $-f(-z)G(z)$, and in the lower half-plane equal $f(z)G(z)$ (see Fig. 1.7f).

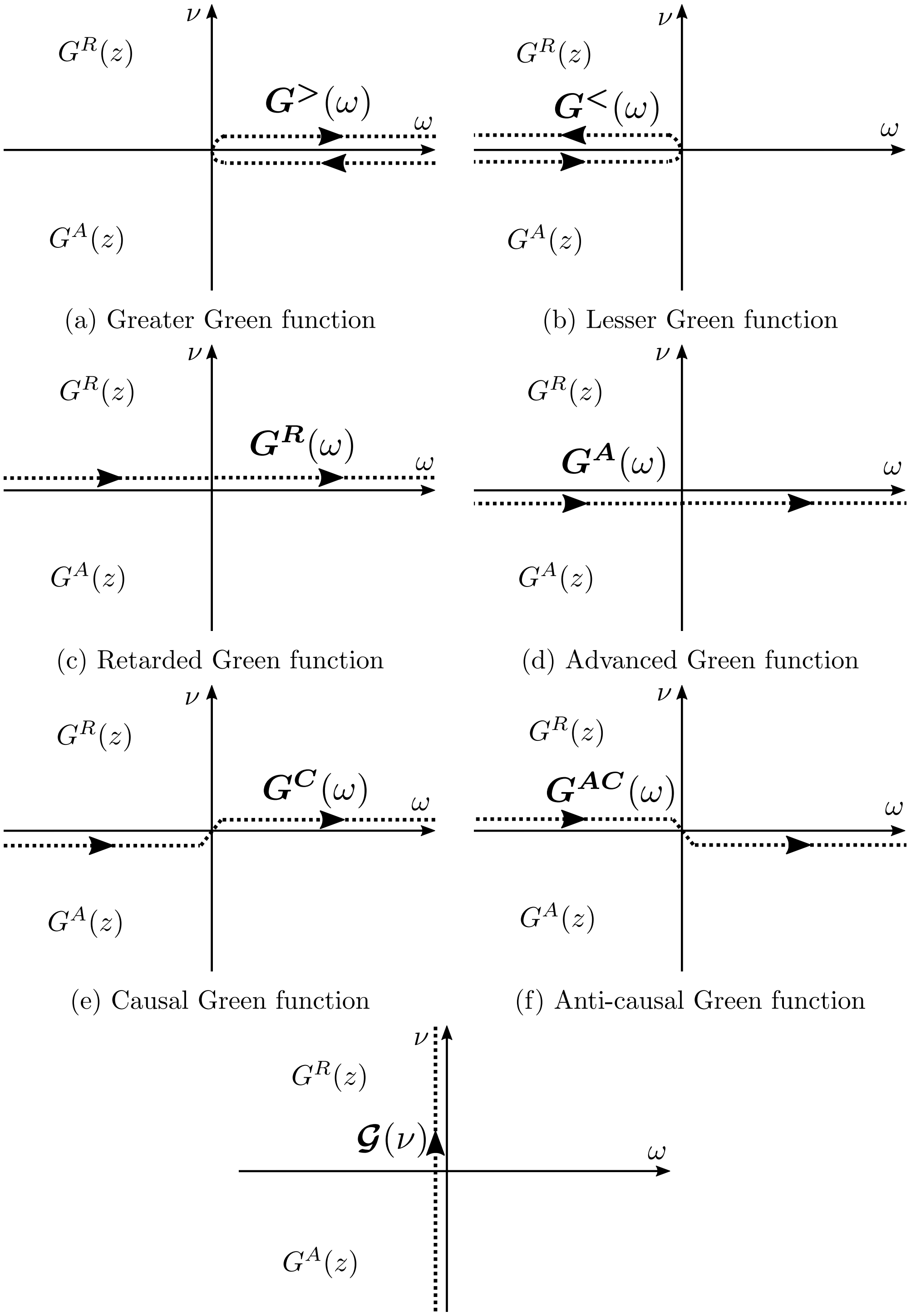
### 1.2.5. Duality of time and frequency

There is a nice duality between the time and frequency domains. We can think of $G^>(\zeta)$ and $G^<(\zeta)$ as the duals of $G^R(z)$ and $G^A(z)$, respectively, and the step function $\theta(t)$ as the dual of the weight function $f(\omega)$. Table. 1.1 shows other aspects of this duality.

In *the zero temperature limit* $\beta \to \infty$, the duality takes its ultimate form. In the frequency domain, the weight function tends to the Heaviside step function $f(\omega) \to \theta(\omega)$ and the Matsubara frequencies get denser till they cover the whole imaginary axis $i\omega_n \to i\nu$. In the time domain, the strips $\mathcal{D}^>$ and $\mathcal{D}^<$ extend till they cover the whole lower and upper half-planes respectively. Most interestingly in this limit, the values of each Green function of frequency can be related to the values of $G(z)$ along some path. Therefore, the inverse Fourier transforms of the different Green functions can be computed as the inverse Fourier transforms of a single function $G(z)$ along different contours in the complex frequency domain (see Fig. 1.9 and compare it to Fig. 1.3).

| Time Domain $\zeta = t - i\tau$ | Frequency Domain $z = \omega + i\nu$ |
|---|---|
| $G^>(\zeta)$ and $G^<(\zeta)$ are the basic building blocks of Green functions of time. | $G^R(z)$ and $G^A(z)$ are the basic building blocks of Green functions of frequency. |
| $G^>(\zeta)$ and $G^<(\zeta)$ are analytic in the strips $\mathcal{D}^>$ and $\mathcal{D}^<$, respectively. | $G^R(z)$ and $G^A(z)$ are analytic in upper and lower half-planes, respectively. |
| $G^>(\zeta)$, $G^<(\zeta)$ are combined into a single function $G(\zeta)$. | $G^R(z)$ and $G^A(z)$ are combined into a single function $G(z)$. |
| $G(\zeta)$ has a jump over the real axis which eqauls $-iA(t)$. | $G(z)$ has a jump over the real axis which equals $-iA(\omega)$. |
| $A(t) = i\,[G^>(t) - G^<(t)]$<br>$G^R(t) = \theta(+t)\ [G^>(t) - G^<(t)]$<br>$G^A(t) = \theta(-t)\ [G^<(t) - G^>(t)]$ | $A(\omega) = i\left[G^R(\omega) - G^A(\omega)\right]$<br>$G^>(\omega) = f(+\omega)\ \left[G^R(\omega) - G^A(\omega)\right]$<br>$G^<(\omega) = f(-\omega)\ \left[G^A(\omega) - G^R(\omega)\right]$ |
| $G^C(t) = \theta(t)G^>(t) + \theta(-t)G^<(t)$<br>$G^{AC}(t) = \theta(-t)G^>(t) + \theta(t)G^<(t)$ | $G^C(\omega) = f(\omega)G^R(\omega) + f(-\omega)G^A(\omega)$<br>$G^{AC}(\omega) = f(-\omega)G^R(\omega) + f(\omega)G^A(\omega)$ |
| $\mathcal{G}(\tau) = -iG(-i\tau)$ on the imaginary axis inside the interval $]-\beta, \beta[$. | $\mathcal{G}_n = G(i\omega_n)$ on the imaginary axis at Matsubara frequencies $\omega_n$. |

Table 1.1.: Duality between Green functions of time and Green functions of frequency.



(a) Greater Green function

(b) Lesser Green function

(c) Retarded Green function

(d) Advanced Green function

(e) Causal Green function

(f) Anti-causal Green function

(g) Matsubara Green function

Figure 1.9.: The different Green functions of frequency in *the zero temperature limit* can be seen as contours of $G(z)$ in the complex frequency plane $z = \omega + i\nu$. Notice that the contours (a)-(f) are shifted infinitesimally above or below the real axis because $G(z)$ is undefined on the real axis. However, the contour (g) of Matsubara Green function is actually *on* the imaginary axis and the apparent shift is for visual clarity only.

## 1.3. Correlation Functions

### 1.3.1. Motivation: linear response theory

The properties of a material are described by how it changes in response to an external perturbation like an electromagnetic field. When the perturbation is sufficiently weak, we can neglect higher-order terms and focus on the linear response of the system [11]. This is given by the *Kubo formula* which we will derive next.

Let us have a quantum mechanical system whose unperturbed Hamiltonian is $\hat{H}_0$. Assuming the system is in thermodynamic equilibrium, we switch on at time $t_0$ a time-dependent external field $g(t)$ that couples to an observable of the system $\hat{B}$. This perturbation leads to an additional term in the Hamiltonian

$$\delta\hat{H}(t) = \hat{B}g(t) \; , \tag{1.61}$$

and the total Hamiltonian of the system reads

$$\hat{H}(t) = \hat{H}_0 + \theta(t - t_0)\delta\hat{H}(t) \; .$$

Now we are interested in the time dependence of the expectation value of some observable $\hat{A}$

$$\left\langle \hat{A}(t) \right\rangle = \left\langle e^{i\hat{H}t}\hat{A}e^{-i\hat{H}t} \right\rangle = \left\langle \underbrace{e^{i\hat{H}t}e^{-i\hat{H}_0 t}}_{\hat{U}_I^\dagger(t)} \underbrace{e^{i\hat{H}_0 t}\hat{A}e^{-i\hat{H}_0 t}}_{\hat{A}_I(t)} \underbrace{e^{i\hat{H}_0 t}e^{-i\hat{H}t}}_{\hat{U}_I(t)} \right\rangle \; .$$

The subscript $I$ stands for the interaction picture where the operator evolves according to the unperturbed Hamiltonian $\hat{H}_0$. The evolution operator $\hat{U}_I(t)$ satisfies the the following differential equation

$$\partial_t \hat{U}_I(t) = e^{i\hat{H}_0 t}(i\hat{H}_0 - i\hat{H})e^{-i\hat{H}t} = -i\delta\hat{H}_I(t)\hat{U}_I(t) \; ,$$

with the initial condition

$$\hat{U}_I(t_0) = 1 \; .$$

We can integrate this differential equation to get a first-order approximation of the evolution operator

$$\hat{U}_I(t) = 1 - i\int_{t_0}^{t} dt'\delta\hat{H}_I(t')\hat{U}_I(t') \approx 1 - i\int_{t_0}^{t} dt'\delta\hat{H}_I(t') + \mathcal{O}((\delta\hat{H})^2)$$

Substituting back in the expectation value and neglecting high-order terms, we get

$$\begin{aligned}\left\langle \hat{A}(t) \right\rangle &= \left\langle \hat{A}_I(t) - i\int_{t_0}^{t} dt'\hat{A}_I(t)\delta\hat{H}_I(t') + i\int_{t_0}^{t} dt'\delta\hat{H}_I(t')\hat{A}_I(t) \right\rangle \\ &= \left\langle \hat{A}_I(t) \right\rangle - i\int_{t_0}^{t} dt' \left\langle \left[\hat{A}_I(t), \delta\hat{H}_I(t')\right] \right\rangle \; .\end{aligned}$$

Since the system, in the absence of a perturbation, is in thermal equilibrium, the first term is time-independent and equals to the expectation value before switching on the perturbation. As a result, the last equation can be rearranged to give the linear response of observable $\hat{A}$ as

$$\delta A(t) := \left\langle \hat{A}(t) \right\rangle - \left\langle \hat{A}_I(t) \right\rangle = -i \int_{t_0}^{t} dt' \left\langle \left[ \hat{A}_I(t), \delta\hat{H}_I(t') \right] \right\rangle \ .$$

Using the relation between the perturbation term $\delta\hat{H}$ and observable $\hat{B}$ (see Eq. 1.61), the linear response can rewritten in the following form, called **Kubo formula**

$$\delta A(t) = \int_{t_0}^{\infty} dt' \chi^R_{\hat{A}\hat{B}}(t, t') g(t') \ ,$$

where we have defined the **retarded correlation function** between observables $\hat{A}$ and $\hat{B}$, also called the **response function**, as

$$\chi^R_{AB}(t, t') := -i\theta(t - t') \left\langle \left[ \hat{A}_I(t), \hat{B}_I(t') \right] \right\rangle \ .$$

The operators of the correlation function are in the interaction picture which is equivalent to the Heisenberg picture of the system with *perturbation turned off*. Therefore, the Kubo formula expresses the linear response of the system to an external perturbation in terms of a correlation function of the unperturbed system!

Since the unperturbed Hamiltonian is time-independent, the correlation function depends on time differences only

$$\chi^R_{AB}(t, t') = \chi^R_{AB}(t - t') \ .$$

Besides, we are usually interested in the steady-state response of the system (as opposed to transient response), so it makes sense to take the limit $t_0 \to \infty$. Then, the linear response becomes a convolution between the correlation function and the field

$$\delta A(t) = \int_{-\infty}^{\infty} dt' \chi^R_{AB}(t - t') g(t') \ . \tag{1.62}$$

Taking the Fourier transform, the convolution becomes a product and we have Kubo formula in the frequency domain

$$\delta A(\omega) = \chi^R_{AB}(\omega) g(\omega) \ . \tag{1.63}$$

In the following, we discuss two important cases: The linear response of magnetization to an external magnetic field which gives rise to the magnetic susceptibility, and the linear response of the electric current to an external electric field which gives rise to the optical conductivity.

**Magnetic Susceptibility** Let us apply a magnetic field $\mathbf{h}(t)$ to a paramagnetic material. The field couples to the magnetic moments $\hat{\boldsymbol{\mu}}_i$ leading to an interaction term

$$\delta\hat{H}(t) = -\sum_{i=1}^{n} \hat{\boldsymbol{\mu}}_i \cdot \mathbf{h}_i(t) = g_s\mu_B \sum_{i=1}^{n} \hat{\mathbf{S}}_i \cdot \mathbf{h}_i(t) = g_s\mu_B \sum_{i=1}^{n} \sum_{\alpha=x,y,z} \hat{\mathbf{S}}_i^\alpha h_i^\alpha(t) \ ,$$

where $\sum_i$ is a sum over all sites and $\sum_\alpha$ is a sum over different directions. We are interested in the magnetization of the system in response to the magnetic field. The magnetization is defined as the density of the magnetic moment

$$\mathbf{M}_i = \frac{\langle \hat{\boldsymbol{\mu}}_i \rangle}{V} = -\frac{g_s\mu_B}{V} \left\langle \hat{\mathbf{S}}_i \right\rangle \ .$$

Identifying operators $\hat{A}$, $\hat{B}$ in the Kubo formula as spin operators, we can write the induced magnetization at site $i$ in direction $\alpha$

$$\delta M_i^\alpha(t) = \int_{-\infty}^{\infty} dt' \sum_{j=1}^{n} \sum_{\beta} \chi^{M}{}_{ij}^{\alpha\beta}(t-t')\ h_j^\beta(t') \ , \tag{1.64}$$

where the **magnetic susceptibility tensor** is defined in terms of the spin-spin retarded correlation function as

$$\chi^{M}{}_{ij}^{\alpha\beta}(t-t') := -\frac{1}{V} \left(g_s\mu_B\right)^2 \chi^R_{S_i^\alpha S_j^\beta}(t-t') \ . \tag{1.65}$$

**Optical Conductivity** Let us apply an electric field $\mathbf{E}(\mathbf{r},t)$ to a system of electrons which can be written in terms of the vector potential $\mathbf{A}_{\text{Ext}}(\mathbf{r},t)$ as

$$\mathbf{E}(\mathbf{r},t) = -\partial_t \mathbf{A}_{\text{Ext}}(\mathbf{r},t) \ ,$$

where Coulomb gauge has been chosen i.e. the external electric potential $\phi_{\text{ext}}$ vanishes. This field couples to the current operator $\mathbf{J}$ leading to the interaction term

$$\delta\hat{H}(t) = e \int d\mathbf{r}\, \hat{\mathbf{J}}(\mathbf{r}) \cdot \mathbf{A}_{\text{Ext}}(\mathbf{r},t) \ .$$

To simplify matters, we go to the frequency domain where the Fourier transforms of the previous relations read

$$\mathbf{E}(\mathbf{r},\omega) = i\omega \mathbf{A}_{\text{Ext}}(\mathbf{r},\omega)$$

$$\delta\hat{H}(\omega) = e \int d\mathbf{r}\, \hat{\mathbf{J}}(\mathbf{r}) \cdot \mathbf{A}_{\text{Ext}}(\mathbf{r},\omega) = \frac{e}{i\omega} \int d\mathbf{r}\, \hat{\mathbf{J}}(\mathbf{r}) \cdot \mathbf{E}(\mathbf{r},\omega) \ .$$

We are interested in the response of the electric current $\mathbf{J}_e = -e \left\langle \hat{\mathbf{J}} \right\rangle$ which requires evaluating the expectation value of the current operator

$$\hat{\mathbf{J}}(\mathbf{r},\omega) = \hat{\mathbf{J}}^0(\mathbf{r},\omega) + \frac{e}{m} \mathbf{A}_{\text{ext}}(\mathbf{r},\omega) \hat{\rho}(\mathbf{r}) \ .$$

The first term represents the current in equilibrium, while the second represents an additional current due to the external field. Since there is no net current in equilibrium, the expectation value of the equilibrium current vanishes in *the interaction picture* i.e. $\left\langle \hat{\mathbf{J}}^0_I(\mathbf{r}) \right\rangle = 0$. Consequently, we can use Kubo formula with operators $\hat{A} = \hat{\mathbf{J}}_0$ and $\hat{B} = \hat{\mathbf{J}}_0$, to get the desired expectation value of the first term to a linear order in $\mathbf{E}$ as

$$\left\langle \hat{\mathbf{J}}^0(\mathbf{r}, \omega) \right\rangle = \delta \left\langle \hat{\mathbf{J}}^0(\mathbf{r}, \omega) \right\rangle = \frac{e}{i\omega} \int d\mathbf{r}' \, \boldsymbol{\chi}^R_{\mathbf{J}_0(\mathbf{r})\mathbf{J}_0(\mathbf{r}')}(\omega) \cdot \mathbf{E}(\mathbf{r}', \omega) \; .$$

The expectation value of the second term can be also evaluated to a linear order in $\mathbf{E}$ in terms of the expectation value of the density operator in *the interaction picture*

$$\left\langle \frac{e}{m} \mathbf{A}_{\text{ext}}(\mathbf{r}, \omega) \hat{\rho}(\mathbf{r}) \right\rangle \approx \frac{e}{m} \mathbf{A}_{\text{ext}}(\mathbf{r}, \omega) \left\langle \hat{\rho}_I(\mathbf{r}) \right\rangle = \frac{e}{i\omega m} \mathbf{E}(\mathbf{r}, \omega) \left\langle \hat{\rho}_I(\mathbf{r}) \right\rangle$$

The two terms can be summed to give the electric current at position $\mathbf{r}$ in direction $\alpha$

$$J^\alpha_e(\mathbf{r}, \omega) = \int d\mathbf{r}' \sum_\beta \sigma^{\alpha\beta}(\mathbf{r}, \mathbf{r}', \omega) E^\beta(\mathbf{r}', \omega) \; , \tag{1.66}$$

where the optical conductivity tensor is defined in terms of the current-current retarded correlation function $\Pi^R_{\alpha\beta}(\mathbf{r}, \mathbf{r}') := \chi^R_{\hat{J}^\alpha_0(\mathbf{r})\hat{J}^\beta_0(\mathbf{r}')}$ and the electronic density $n(\mathbf{r}) := \langle \hat{\rho}_I(\mathbf{r}) \rangle$ as

$$\sigma^{\alpha\beta}(\mathbf{r}, \mathbf{r}', \omega) := \frac{ie^2}{\omega} \Pi^R_{\alpha\beta}(\mathbf{r}, \mathbf{r}', \omega) + \frac{ie^2}{\omega m} n(\mathbf{r}) \delta(\mathbf{r} - \mathbf{r}') \delta_{\alpha\beta} \; . \tag{1.67}$$

### 1.3.2. Analytic structure of correlation functions

The Kubo formula served as a strong motivation for studying correlation functions between observables. The central correlation function is the retarded one [11,12]

$$\chi^R_{AB}(t) := -i\theta(t) \left\langle \left[ \hat{A}(t), \hat{B}(0) \right] \right\rangle \; . \tag{1.68}$$

However, this function does not have nice analytical properties due to the step function, thus we define the following more basic correlation function[13]

$$S_{AB}(t) := \left\langle \hat{A}(t) \hat{B}(0) \right\rangle \; . \tag{1.69}$$

We can also swap the operators in the above expression to get a different correlation function

$$\tilde{S}_{AB}(t) := S_{BA}(-t) = \left\langle \hat{B}(0) \hat{A}(t) \right\rangle \; . \tag{1.70}$$

[11] To derive the Kubo formula, we had to use the total Hamiltonian and the interaction picture. From now on, we will reuse $H$ to denote the unperturbed Hamiltonian and go back to the Heisenberg picture where all operators evolve according to the unperturbed $H$.

[12] We have dropped the second time variable because the function depends on time differences only.

[13] The term **correlation function**, without any prefix, is often used to refer to this particular version of correlation functions.

These correlation functions look similar to the bosonic greater and lesser Green functions (Eqs. (1.5), (1.6)) where the annihilation operator is replaced by observable $\hat{A}$ and the creation operator is replaced by observable $\hat{B}$. Noticing that in deriving the analytic properties of Green functions, we have not used any particular property of creation and annihilation operators, it should come as no surprise that correlation functions have exactly the same analytical structure as the bosonic Green functions. In particular, we can analytically continue $S_{AB}(t)$ and $\tilde{S}_{AB}(t)$ to get the imaginary-time correlation function

$$\mathcal{X}(\tau) := \theta(\tau)\left\langle \hat{A}(-i\tau)\hat{B}(0)\right\rangle + \theta(-\tau)\left\langle \hat{B}(0)\hat{A}(-i\tau)\right\rangle \ ,$$

and the correlation function of complex time

$$\chi_{AB}(\zeta) := \begin{cases} S_{AB}(\zeta), & \text{for } \zeta \in \mathcal{D}^{>} \\ \tilde{S}_{AB}(\zeta), & \text{for } \zeta \in \mathcal{D}^{<} \end{cases} . \tag{1.71}$$

Moreover, each correlation function has a spectral function defined as

$$\chi''(t) := 2\left\langle \hat{A}(t)\hat{B}(0) - \hat{B}(0)\hat{A}(t)\right\rangle \ , \tag{1.72}$$

whose Fourier transform $\chi''(\omega)$ can be used to get the correlation function of complex frequency

$$\chi(z) := \int \frac{d\omega}{\pi}\frac{\chi''(\omega)}{z-\omega} \ . \tag{1.73}$$

By using the following basic mapping, all the previous relations derived for bosonic Green functions apply directly to correlation functions:

| | |
|---|---|
| $\hat{c}_\kappa \leftrightarrow \hat{A}$ | observable under study |
| $\hat{c}^\dagger_{\kappa'} \leftrightarrow \hat{B}$ | observable coupled to external field |
| $iG(\zeta) \leftrightarrow \chi(\zeta)$ | correlation function of complex time |
| $G(z) \leftrightarrow \chi(z)$ | correlation function of complex frequency |

Mapping of other quantities follows immediately

| | |
|---|---|
| $G^R \leftrightarrow \chi^R$ | response function |
| $iG^{>} \leftrightarrow S$ | correlation function |
| $-\mathcal{G} \leftrightarrow \mathcal{X}$ | imaginary-time correlation function |
| $1/2\,A \leftrightarrow \chi''$ | spectral function |

There are two particularly important relations. The first is the spectral representation of the response function which is the analog of Eq. (1.44)

$$\chi^R_{AB}(\omega) = \int \frac{d\omega'}{\pi}\,\frac{\chi''_{AB}(\omega')}{\omega' - \omega - i\eta} \ . \tag{1.74}$$

The second is the fluctuation-dissipation theorem which is the analog of Eq. (1.40)

$$S_{AB}(\omega) = \frac{2}{1-e^{-\beta\omega}}\,\chi''_{AB}(\omega) \tag{1.75}$$

## 1.4. The problem of analytic continuation

Green and correlation functions are analytic functions in the complex planes of time and frequency and their analytic structure means that they are completely determined by their values in any sub-domain of either complex time or complex frequency. It turned out that Green or correlation function values are most efficiently computed on the imaginary axis using quantum Monte Carlo (QMC) simulations. When QMC is done in the time domain, the values are calculated for positive imaginary times $i\tau : 0 < \tau < \beta$, while they are calculated on Matsubara frequencies $i\omega_n$ when QMC is done in the frequency domain. However, examining the definition of Green or correlation function

$$G(z) = \int \frac{d\omega}{2\pi} \frac{A(\omega)}{z - \omega}$$

$$\chi(z) = \int \frac{d\omega}{\pi} \frac{\chi''(\omega)}{z - \omega}$$

makes it clear that we need the spectral function $A(\omega)$ or $\chi''(\omega)$ in order to determine $G(z)$ or $\chi(z)$ in any other region of the complex frequency plane.

Calculating spectral functions $A(\omega)$ or $\chi''(\omega)$ from QMC data on either imaginary time or Matsubara frequencies is known as **the analytic continuation problem**. This is an important problem because the dynamical properties of a system are determined by functions like $G^R(\omega)$ and $\chi^R(\omega)$ which correspond to Green and correlation function values on the real axis.

### 1.4.1. Analytic continuation relations

In the following, we will relate the spectral functions to both imaginary-time and imaginary-frequency quantities. The relation to Green function values at Matsubara frequencies is obtained easily from equations (1.51) and (1.59)

$$\mathcal{G}_n = G(i\omega_n) = \int \frac{d\omega}{2\pi} \frac{A(\omega)}{i\omega_n - \omega} \; . \tag{1.76}$$

The relation to imaginary-time Green function of positive $\tau$ can be obtained from the inverse Fourier transform of the greater Green function

$$\mathcal{G}(\tau) = -iG^{>}(-i\tau) = -i \int \frac{d\omega}{2\pi} e^{-i\omega(-i\tau)} G^{>}(\omega) \Rightarrow$$

$$\mathcal{G}(\tau) = \int \frac{d\omega}{2\pi} \frac{-e^{-\omega\tau}}{1 \pm e^{-\omega\beta}} A(\omega) \quad : 0 < \tau < \beta \; . \tag{1.77}$$

For negative $\tau$, we use the inverse Fourier transform of the lesser Green function

$$\mathcal{G}(\tau) = -iG^{<}(-i\tau) = -i \int \frac{d\omega}{2\pi} e^{-i\omega(-i\tau)} G^{<}(\omega) \Rightarrow$$

$$\mathcal{G}(\tau) = \int \frac{d\omega}{2\pi} \frac{e^{-\omega\tau}}{1 \pm e^{\omega\beta}} A(\omega) \quad : -\beta < \tau < 0 \; . \tag{1.78}$$

Similarly, the correlation function values at Matsubara frequencies read

$$\mathcal{X}_n = -\chi(i\omega_n) = \int \frac{d\omega}{\pi} \frac{\chi''(\omega)}{\omega - i\omega_n} \,, \tag{1.79}$$

while the relations between imaginary-time correlation function and the spectral function read

$$\mathcal{X}(\tau) = \int \frac{d\omega}{\pi} \frac{e^{-\omega\tau}}{1 - e^{-\omega\beta}}\, \chi''(\omega) \qquad : 0 < \tau < \beta \tag{1.80}$$

$$\mathcal{X}(\tau) = \int \frac{d\omega}{\pi} \frac{-e^{-\omega\tau}}{1 - e^{\omega\beta}}\, \chi''(\omega) \qquad : -\beta < \tau < 0 \tag{1.81}$$

### 1.4.2. Matrix structure of spectral functions

In most of the previous relations, we omitted the basis indices $\kappa, \kappa'$ of $A_{\kappa,\kappa'}$ and operator indices $A, B$ of correlation functions $\chi''_{A,B}$. This was intentional because all previous relations hold for all indices, and we did not want to clutter the equations. Now we examine this matrix structure more closely.

The transformation of creation and annihilation operators between any two single-particle orthonormal basis sets $\{|\psi_\kappa\rangle\}$ and $\{|\tilde{\psi}_\lambda\rangle\}$ reads

$$\hat{c}_\lambda = \sum_\kappa \langle\tilde{\psi}_\lambda|\psi_\kappa\rangle\, \hat{c}_\kappa \tag{1.82}$$

$$\hat{c}^\dagger_\lambda = \sum_\kappa \langle\psi_\kappa|\tilde{\psi}_\lambda\rangle\, \hat{c}^\dagger_\kappa \tag{1.83}$$

Therefore, we can transform the Green spectral functions between the two basis as following

$$\tilde{A}_{\lambda,\lambda'} = \sum_\kappa \sum_{\kappa'} \langle\tilde{\psi}_\lambda|\psi_\kappa\rangle\, A_{\kappa,\kappa'}\, \langle\psi_{\kappa'}|\tilde{\psi}_{\lambda'}\rangle \;, \tag{1.84}$$

which holds for spectral functions of both time and frequency. Arranging the spectral functions $\tilde{A}_{\lambda,\lambda'}$ and $A_{\kappa,\kappa'}$ and the transformation coefficients $U_{\kappa,\lambda} := \langle\psi_\kappa, \tilde{\psi}_\lambda\rangle$ in matrices, we can rewrite the above equation compactly as

$$\tilde{\mathbf{A}} = \mathbf{U}^\dagger \mathbf{A} \mathbf{U} \,. \tag{1.85}$$

We can also, in some special cases, derive a similar result for correlation functions. For example, let us take the density operator in the configuration space $\hat{\rho}_\sigma(\mathbf{r}) := \hat{\Psi}^\dagger_\sigma(\mathbf{r})\hat{\Psi}_\sigma(\mathbf{r})$. This can be related to the density operator in momentum space $\hat{\rho}_{\mathbf{q}\sigma} := \sum_\mathbf{k} \hat{c}^\dagger_{\mathbf{k}\sigma}\hat{c}_{\mathbf{k}+\mathbf{q}\sigma}$ as

$$\begin{aligned}\hat{\rho}_\sigma(\mathbf{r}) &= \frac{1}{\mathcal{V}} \sum_{\mathbf{k},\mathbf{k}'} e^{i(\mathbf{k}-\mathbf{k}')\cdot\mathbf{r}} \hat{c}^\dagger_{\mathbf{k}'\sigma}\hat{c}_{\mathbf{k}\sigma} = \frac{1}{\mathcal{V}} \sum_{\mathbf{k},\mathbf{q}} e^{-i\mathbf{q}\cdot\mathbf{r}} \hat{c}^\dagger_{\mathbf{k}+\mathbf{q}\sigma}\hat{c}_{\mathbf{k}\sigma} \\ &= \frac{1}{\mathcal{V}} \sum_\mathbf{q} e^{i\mathbf{q}\cdot\mathbf{r}} \left( \sum_\mathbf{k} \hat{c}^\dagger_{\mathbf{k}\sigma}\hat{c}_{\mathbf{k}+\mathbf{q}\sigma} \right) = \frac{1}{\mathcal{V}} \sum_\mathbf{q} e^{i\mathbf{q}\cdot\mathbf{r}} \hat{\rho}_{\mathbf{q}\sigma} \,,\end{aligned}$$

where $\mathbf{q} = \mathbf{k}' - \mathbf{k}$. In such cases,[14] the relation between observable operators $\hat{O}_\kappa$ in the single-particle orthonormal basis $\{|\psi_\kappa\rangle\}$ and the observable operators $\hat{O}_\lambda$ in the basis $\{|\tilde{\psi}_\lambda\rangle\}$ reads

$$\hat{O}_\lambda = \sum_\kappa \langle\tilde{\psi}_\lambda|\psi_\kappa\rangle\, \hat{O}_\kappa \ . \tag{1.86}$$

Therefore, we can transform the correlation spectral functions between the two basis as following

$$\tilde{\chi}''_{O_\lambda,O_{\lambda'}} = \sum_\kappa \sum_{\kappa'} \langle\tilde{\psi}_\lambda|\psi_\kappa\rangle\, \langle\psi_{\kappa'}|\tilde{\psi}_{\lambda'}\rangle\, \chi''_{O_\kappa,O_{\kappa'}} \ , \tag{1.87}$$

which can also be written compactly as

$$\tilde{\boldsymbol{\chi}}'' = \mathbf{U}^\dagger \boldsymbol{\chi}'' \mathbf{U} \ . \tag{1.88}$$

Now we will show that the spectral matrices $\mathbf{A}(\omega)$ and $\boldsymbol{\chi}''(\omega)$ are Hermitian matrices for each $\omega$. Using the definition of Green spectral function in the time domain, we can write

$$\begin{aligned} A_{\kappa,\kappa'}(t) &= \left\langle \hat{c}_\kappa(t)\hat{c}^\dagger_{\kappa'}(0) \pm \hat{c}^\dagger_{\kappa'}(0)\hat{c}_\kappa(t) \right\rangle \\ &= \left\langle \left(\hat{c}_{\kappa'}(0)\hat{c}^\dagger_\kappa(t)\right)^\dagger \pm \left(\hat{c}^\dagger_\kappa(t)\hat{c}_{\kappa'}(0)\right)^\dagger \right\rangle \\ &= \left[\left\langle \hat{c}_{\kappa'}(0)\hat{c}^\dagger_\kappa(t) \pm \hat{c}^\dagger_\kappa(t)\hat{c}_{\kappa'}(0) \right\rangle\right]^* \\ &= \left[\left\langle \hat{c}_{\kappa'}(-t)\hat{c}^\dagger_\kappa(0) \pm \hat{c}^\dagger_\kappa(0)\hat{c}_{\kappa'}(-t) \right\rangle\right]^* \\ &= \left[A_{\kappa',\kappa}(-t)\right]^* \ . \end{aligned} \tag{1.89}$$

In the frequency domain, this relation reads

$$A_{\kappa,\kappa'}(\omega) = \left[A_{\kappa',\kappa}(\omega)\right]^* \ . \tag{1.90}$$

Similarly, we find for correlation spectral functions

$$\chi''_{O_\kappa,O_{\kappa'}}(\omega) = \left[\chi''_{O_{\kappa'},O_\kappa}(\omega)\right]^* \ . \tag{1.91}$$

So indeed the spectral matrices are Hermitian.

## 1.4.3. Diagonal spectral functions

From the Hermiticity of the spectral matrices derived above, it follows immediately that diagonal spectral functions are real. These diagonal functions have also other properties that make their analytic continuation easier than non-diagonal ones. We start by deriving the so-called **Lehmann representation** where a complete set of eigenstates (complete in Fock space, so no restriction on particle number) is inserted in

[14] This is not general because observable operators transform normally as matrices not as vectors.

the definition of the spectral function. The spectral function of a Green function in the time domain reads

$$\begin{aligned}
A_\kappa(t) &= \left\langle \hat{c}_\kappa(t)\hat{c}_\kappa^\dagger(0) \pm \hat{c}_\kappa^\dagger(0)\hat{c}_\kappa(t) \right\rangle = \left\langle e^{i\hat{H}t}\hat{c}_\kappa e^{-i\hat{H}t}\hat{c}_\kappa^\dagger \pm \hat{c}_\kappa^\dagger e^{i\hat{H}t}\hat{c}_\kappa e^{-i\hat{H}t} \right\rangle \\
&= Z^{-1}\sum_n \left\{ \langle n| e^{-\beta\hat{H}} e^{i\hat{H}t}\hat{c}_\kappa e^{-i\hat{H}t}\hat{c}_\kappa^\dagger |n\rangle \pm \langle n| e^{-\beta\hat{H}}\hat{c}_\kappa^\dagger e^{i\hat{H}t}\hat{c}_\kappa e^{-i\hat{H}t} |n\rangle \right\} \\
&= Z^{-1}\sum_{n,m} e^{-\beta E_n} \Big\{ \langle n| e^{i\hat{H}t}\hat{c}_\kappa e^{-i\hat{H}t} |m\rangle \langle m| \hat{c}_\kappa^\dagger |n\rangle \\
&\qquad\qquad \pm \langle n| \hat{c}_\kappa^\dagger |m\rangle \langle m| e^{i\hat{H}t}\hat{c}_\kappa e^{-i\hat{H}t} |n\rangle \Big\} \\
&= Z^{-1}\sum_{n,m} e^{-\beta E_n} \left\{ e^{i(E_n-E_m)t} |\langle n| \hat{c}_\kappa |m\rangle|^2 \pm e^{i(E_m-E_n)t} |\langle m| \hat{c}_\kappa |n\rangle|^2 \right\} \\
&= Z^{-1}\sum_{n,m} \left( e^{-\beta E_n} \pm e^{-\beta E_m} \right) |\langle n| \hat{c}_\kappa |m\rangle|^2 e^{i(E_n-E_m)t}
\end{aligned}$$

Taking the Fourier transform, we get the desired Lehmann representation in the frequency domain

$$A_\kappa(\omega) = Z^{-1}\sum_{n,m} \left( e^{-\beta E_n} \pm e^{-\beta E_m} \right) |\langle n| \hat{c}_\kappa |m\rangle|^2 \, 2\pi\delta(\omega + E_n - E_m) \; . \tag{1.92}$$

Similarly, we find the Lehmann representation of the spectral function of a correlation function

$$\chi_O''(\omega) = Z^{-1}\sum_{n,m} \left( e^{-\beta E_n} - e^{-\beta E_m} \right) \left| \langle n| \hat{O} |m\rangle \right|^2 \pi\delta(\omega + E_n - E_m) \; . \tag{1.93}$$

**Inequalities** We see easily that all terms in Eq. (1.92) (upper sign) are non-negative. Therefore, we have

$$A(\omega) \geq 0 \quad : \text{for the fermionic Green function case} \; .$$

On the other hand, the terms in Eq. (1.92) (lower sign) corresponding to positive frequencies are positive because then $E_m > E_n$ so $(e^{-\beta E_n} - e^{-\beta E_m}) > 0$, while terms corresponding to negative frequencies are negative because then $E_m < E_n$ so $(e^{-\beta E_n} - e^{-\beta E_m}) < 0$. Therefore, we have

$$\text{sign}(\omega)A(\omega) \geq 0 \quad : \text{for the bosonic Green function case} \; .$$

Similarly, using Eq. (1.93) we get

$$\text{sign}(\omega)\chi''(\omega) \geq 0 \quad : \text{for the correlation function case} \; .$$

**Parity** The spectral functions of correlation functions are odd

$$\chi''(\omega) = -\chi''(-\omega) \; . \tag{1.94}$$

This can be seen from Eq. (1.93) by exchanging the dummy variables $n, m$ and using the fact that $\hat{O}$ is hermitian (because it corresponds to an observable). However, the spectral functions of Green functions do not possess any parity in the general because the creation and annihilation operators are not Hermitian. In correlation functions case, the parity can be used to rewrite Eq. (1.79) as an integral over positive frequencies only. We start by writing Eq. (1.79) as

$$\mathcal{X}_n = \int_{-\infty}^{\infty} \frac{d\omega}{\pi} \frac{\chi''(\omega)}{\omega - i\omega_n} = \int_{-\infty}^{\infty} \frac{d\omega}{\pi} \frac{\omega + i\omega_n}{\omega^2 + \omega_n^2} \chi''(\omega) \; .$$

The imaginary part vanishes because the integrand is odd while the real part of the integrand is even and thus we can restrict the integral to positive frequencies

$$\mathcal{X}_n = \int_{0}^{\infty} \frac{d\omega}{\pi} \frac{2\omega}{\omega^2 + \omega_n^2} \chi''(\omega) \; .$$

This can be written in the following suggestive form, which highlights the non-negativity of $\chi''(\omega)/\omega$ and removes the singularity at $\omega = 0$ from the integral kernel

$$\mathcal{X}_n = \int_{0}^{\infty} \frac{d\omega}{\pi} \frac{2\omega^2}{\omega^2 + \omega_n^2} \frac{\chi''(\omega)}{\omega} \; . \tag{1.95}$$

We can also rewrite Eq. (1.80) as an integral over positive frequencies only. First we need to employ the periodicity of the imaginary-time correlation function

$$\begin{aligned}
\mathcal{X}(\tau) &= \frac{1}{2}(\mathcal{X}(\tau) + \mathcal{X}(\tau - \beta)) \\
&= \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \frac{e^{-\omega\tau}}{1 - e^{-\omega\beta}} \chi''(\omega) - \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \frac{e^{-\omega(\tau-\beta)}}{1 - e^{\omega\beta}} \chi''(\omega) \\
&= \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \frac{e^{-\omega\tau}}{1 - e^{-\omega\beta}} \chi''(\omega) - \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \frac{e^{\omega(\tau-\beta)}}{1 - e^{-\omega\beta}} \chi''(-\omega) \\
&= \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \frac{e^{-\omega\tau} + e^{-\omega(\beta-\tau)}}{1 - e^{-\omega\beta}} \chi''(\omega)
\end{aligned}$$

Now the integrand is even, so we can restrict the integral to positive frequencies

$$\mathcal{X}(\tau) = \int_{0}^{\infty} \frac{d\omega}{\pi} \frac{e^{-\omega\tau} + e^{-\omega(\beta-\tau)}}{1 - e^{-\omega\beta}} \chi''(\omega) \; : \; 0 < \tau < \beta$$

Again, this equation can be written in a suggestive form that highlights the non-negativity of $\chi''(\omega)/\omega$ and removes the singularity at $\omega = 0$ from the kernel of the integral

$$\mathcal{X}(\tau) = \int_{0}^{\infty} \frac{d\omega}{\pi} \frac{\omega \left[e^{-\omega\tau} + e^{-\omega(\beta-\tau)}\right]}{1 - e^{-\omega\beta}} \frac{\chi''(\omega)}{\omega} \; : \; 0 < \tau < \beta \tag{1.96}$$

**Sum rules** The spectral function of a a Green function satisfies the following sum rule as a direct result of (anti)commutation relations

$$A_\kappa(t=0) = \left\langle \hat{c}_\kappa \hat{c}^\dagger_\kappa \pm \hat{c}^\dagger_\kappa \hat{c}_\kappa \right\rangle = 1 \Rightarrow \int \frac{d\omega}{2\pi}\, A(\omega) = 1 \;. \tag{1.97}$$

On the other hand, the integral of a spectral function of a correlation function vanishes because it is odd. However, it satisfies the following sum rule which follows directly from Eq. (1.79) evaluated at zero Matsubara frequency

$$\int \frac{d\omega}{\pi} \frac{\chi''(\omega)}{\omega} = \mathcal{X}_0 \tag{1.98}$$

Similarly, the spectral functions of a bosonic Green function satisfy a similar sum rule

$$\int \frac{d\omega}{2\pi} \frac{A(\omega)}{\omega} = -\mathcal{G}_0 \quad \text{(bosons only)} \tag{1.99}$$

### 1.4.4. Non-diagonal spectral functions

Since the inequalities satisfied by diagonal spectral functions are valid in whatever basis we choose, and the diagonal elements in some basis can be expressed in terms of non-diagonal elements in another basis, the non-diagonal elements must also satisfy some inequalities. Naively, one would think that there are an infinite number of inequalities for each $\omega$ because there are an infinite number of possible basis transformations. However, we will show that those inequalities are limited and equal to the number of principle minors of the spectral matrix.

For each $\omega$, these is a basis that diagonalizes $\mathbf{A}(\omega)$. Such a basis need not simultaneously diagonalize all matrices of different $\omega$, but the important thing is that for each $\omega$ such a basis exists. Then, the diagonal elements of these diagonal matrices represent the eigenvalues of the spectral matrix, and the inequalities they satisfy imply the following

Fermionic Green functions: $\mathbf{A}(\omega)$ is a positive semidefinite matrix for all values of $\omega$.

Bosonic Green functions: $\mathbf{A}(\omega)$ is a positive semidefinite matrix for $\omega > 0$ and a negative semidefinite matrix for $\omega < 0$.

Correlation functions: $\boldsymbol{\chi}''(\omega)$ is a positive semidefinite matrix for $\omega > 0$ and a negative semidefinite matrix for $\omega < 0$.

where a matrix is negative/positive semidefinite if and only if all its eigenvalues are non-positive/non-negative. Sylvester's criterion gives a necessary and sufficient condition for a matrix to be positive semidefinite or negative semidefinite. For spectral matrices of fermionic Green functions, it says that all principal minors (i.e. the determinants of the upper left matrices) are non-negative for any $\omega$ . This also holds for spectral matrices of bosonic Green functions and correlation functions when $\omega > 0$. On the other hand, when $\omega < 0$, these matrices are negative semidefinite, so their principal minors of odd dimension are non-positive while principal minors of even dimension are non-negative [12, p.383]. Therefore, if the principal minors of spectral matrices satisfy these limited number of inequalities in some fixed basis, then their diagonal elements satisfy the necessary inequalities in any basis.

### 1.4.5. Analytic continuation as density estimation

In the thesis we are concerned with the analytic continuation of diagonal spectral functions, so we conclude this chapter with an observation that prevail our solution to the analytic continuation problem. The inequalities satisfied by the diagonal spectral functions and their sum rules allows us to interpret the following functions:

$$\begin{aligned}
&A(\omega) &&\text{(for fermionic Green functions case)}\\
&A(\omega)/\omega &&\text{(for bosinic Green functions case)}\\
&\chi''(\omega)/\omega &&\text{(for correlation functions case)}
\end{aligned}$$

as *density functions* because each one is non-negative and has a finite integral. Using this interpretation, the analytic continuation of diagonal Green and correlation functions boils down to estimating a density-like function $f(x)$ from an integral equation

$$g(y) = \int dx\, K(y,x) f(x) \ ,$$

where the left-hand side $g(y)$ represents QMC data known numerically, while the integral kernel $K(y,x)$ is a continuous function known analytically. The kernels of the different analytic continuation problems are summarized in Table. 1.2.

| Description | $g(y)$ | $=$ | $\int dx$ | $K(x,y)$ | $f(x)$ |
|---|---|---|---|---|---|
| Fermionic Green function/Time | $\mathcal{G}(\tau)$ | $=$ | $\int_{-\infty}^{\infty} \frac{d\omega}{2\pi}$ | $\frac{-e^{-\omega\tau}}{1+e^{-\omega\beta}}$ | $A(\omega)$ |
| Bosonic Green function/Time | $\mathcal{G}(\tau)$ | $=$ | $\int_{-\infty}^{\infty} \frac{d\omega}{2\pi}$ | $\frac{-\omega e^{-\omega\tau}}{1-e^{-\omega\beta}}$ | $\frac{A(\omega)}{\omega}$ |
| Correlation function/Time | $\mathcal{X}(\tau)$ | $=$ | $\int_{0}^{\infty} \frac{d\omega}{\pi}$ | $\frac{\omega\left[e^{-\omega\tau}+e^{-\omega(\beta-\tau)}\right]}{1-e^{-\omega\beta}}$ | $\frac{\chi''(\omega)}{\omega}$ |
| Fermionic Green function/Frequency | $\mathcal{G}_n$ | $=$ | $\int_{-\infty}^{\infty} \frac{d\omega}{2\pi}$ | $\frac{1}{i\omega_n-\omega}$ | $A(\omega)$ |
| Bosonic Green function/Frequency | $\mathcal{G}_n$ | $=$ | $\int_{-\infty}^{\infty} \frac{d\omega}{2\pi}$ | $\frac{\omega}{i\omega_n-\omega}$ | $\frac{A(\omega)}{\omega}$ |
| Correlation function/Frequency | $\mathcal{X}_n$ | $=$ | $\int_{0}^{\infty} \frac{d\omega}{\pi}$ | $\frac{2\omega^2}{\omega^2+\omega_n^2}$ | $\frac{\chi''(\omega)}{\omega}$ |

Table 1.2.: The integral equations of different analytic continuation problems. Time relations hold for positive imaginary time only $\tau \in \,]0,\beta[$ , for which QMC data is usually computed.

# 2. Regularization Methods

As we saw in the last chapter, the problem of analytic continuation can be formulated as a Fredholm integral equation of the first kind

$$g(y) = \int dx\ f(x)\ K(x,y)\ , \tag{2.1}$$

where the integral kernel $K(y,x)$ is known analytically, a finite number of noisy data values $g(y)$ are available, and we need to find the model $f(x)$ which is known to be non-negative.

Fredholm integral equations are well-known beyond the analytic continuation problem and have applications in many different fields. The difficulty in solving this type of equations is that they are inherently ill-posed [13]. When the data is computed, sharp features in the model get smoothed and errors get damped due to the integration. The inverse process, therefore, is problematic; small errors in the data may lead (depending on the used method) to very large errors in the reconstructed model.

## 2.1. Discretization

The first step in solving a Fredholm equation is discretizing it to obtain an approximate algebraic equation. Since QMC simulations provide a limited number of data values, the discretization of the $y$ coordinate is already determined by the available data values. We assume there are $m$ such values $g(y_j)$ and organize them in a column vector $\mathbf{g} \in \mathbb{R}^m$ (the case of complex data and complex kernels is handled below).

For discretizing the right-hand side, we introduce a grid in the variable $x$ and evaluate the integral by numerical quadrature

$$\int dx\ f(x)\ K(x,y_j) \approx \sum_{i=1}^{n} \Delta x_i\ f(x_i)\ K(x_i,y_j)\ . \tag{2.2}$$

We then build a column vector $\mathbf{f} \in \mathbb{R}^n$ whose elements are $\sqrt{\Delta x_i} f(x_i)$ and a matrix $\mathbf{K} \in \mathbb{R}^{m\times n}$ whose elements are $\sqrt{\Delta x_i} K(y_j, x_i)$. The quadrature factors $\Delta x_i$ could be removed from $\mathbf{f}$ and included entirely in the matrix $\mathbf{K}$. However, splitting them in the earlier way has the advantage of using the euclidean norm of $\mathbf{f}$ as an approximation of the $L^2$-norm of $f(x)$ i.e.

$$\|f\|_2^2 = \int |f(x)|^2 dx \approx \mathbf{f}^{\mathrm{T}}\mathbf{f} = \|\mathbf{f}\|^2\ . \tag{2.3}$$

This discretization gives the following system of linear equations:

$$\mathbf{K}\ \mathbf{f} = \mathbf{g}, \quad \text{where} \quad \mathbf{f} \in \mathbb{R}^n\ , \mathbf{g} \in \mathbb{R}^m,\ \mathbf{K} \in \mathbb{R}^{m \times n}\ , \tag{2.4}$$

and the problem is finding the vector $\mathbf{f}$.

**Tip** It is worth noting that when the integral extends from $-\infty$ to $+\infty$ (which is typically the case for analytic continuation) and the integrand is analytic in an open strip around the real axis, it is recommended to use the trapezoidal rule (or the sightly different, the rectangle rule) for discretizing the integral. Besides the simplicity of this rule, it converges exponentially with the grid spacing when the above mentioned conditions are satisfied (see Refs. [14, 15]). This discretization error is different from the cutoff error which depends on the decay of the integrand.

**Complex case** In case of a complex kernel and complex data, like Green function in the frequency domain, we can still represent the problem in real space. We split the real and imaginary part of both the kernel and the data

$$\int dx\ f(x)\ [K_1(x,y) + i\ K_2(x,y)] = g_1(y) + i\ g_2(y)\ . \tag{2.5}$$

Since the model is always real, the real part of the data is related only to the real part of the kernel and the imaginary part of the data is related only to the imaginary part of the kernel

$$\int dx\ f(x)\ K_1(x,y) = g_1(y) \tag{2.6}$$

$$\int dx\ f(x)\ K_2(x,y) = g_2(y)\ . \tag{2.7}$$

This way the original complex equation is equivalent to two real decoupled ones. These real equations can then be discretized resulting in a real system of linear equations

$$\mathbf{K}\ \mathbf{f} = \mathbf{g}, \quad \text{where} \quad \mathbf{f} \in \mathbb{R}^n,\ \mathbf{g} = \begin{bmatrix} \mathbf{g_1} \\ \mathbf{g_2} \end{bmatrix} \in \mathbb{R}^{2m},\ \mathbf{K} = \begin{bmatrix} \mathbf{K_1} \\ \mathbf{K_2} \end{bmatrix} \in \mathbb{R}^{2m \times n}\ . \tag{2.8}$$

## 2.2. Least squares method and ill-posedness

Now our goal is to estimate the model vector $\mathbf{f}$. The most naive and straightforward way is solving Eq. 2.4 as any linear system of equations using the least squares method. Least squares finds the model minimizing $\chi^2(\mathbf{f}) := \|\mathbf{K}\ \mathbf{f} - \mathbf{g}\|^2$ which represents how well the model fits the data

$$\mathbf{f}_{\text{LS}} = \underset{\mathbf{f} \in \mathbb{R}^n}{\arg\min}\ \chi^2(\mathbf{f})\ . \tag{2.9}$$

The least squares solution is found by setting the derivative of the fit[1] to zero

$$\left.\frac{d\chi^2}{d\mathbf{f}}\right|_{\mathbf{f}=\mathbf{f}_{\mathrm{LS}}} = 0 \Leftrightarrow \frac{d}{d\mathbf{f}^{\mathrm{T}}}(\mathbf{f}^{\mathrm{T}}\ \mathbf{K}^{\mathrm{T}} - \mathbf{g}^{\mathrm{T}})(\mathbf{K}\ \mathbf{f} - \mathbf{g}) = 0 \Leftrightarrow \mathbf{K}^{\mathrm{T}}\mathbf{K}\ \mathbf{f}_{\mathrm{LS}} = \mathbf{K}^{\mathrm{T}}\mathbf{g}\ .$$

The last linear system is called the system of *normal equations* and it has a unique solution when $\mathbf{K}$ has full column rank. However, when $\mathbf{K}$ is column rank deficient, which is typically the case in analytic continuation, there is an infinite number of solutions. This is because adding any vector from the null space of $\mathbf{K}$ to a solution does not change the corresponding data. Therefore, it is common to take the least squares solution with the minimum norm as the unique least squares solution

$$\mathbf{f}_{\mathrm{LS}} = \underset{\mathbf{f}\in\mathbb{R}^n}{\arg\min}\ \chi^2(\mathbf{f}) \ \text{ and } \|\mathbf{f}\|^2 \text{ is minimal}\ . \tag{2.10}$$

Chapter 5 of Ref. [16] discusses the least squares problem and several numerically stable algorithms for solving it.

**Noise covariance** Since QMC results are averages of many data samples, the central limit theorem says that the noise on the data is distributed as a Gaussian with zero mean. Let the covariance matrix of this Gaussian be $\mathbf{Cov}$, which can be estimated from multiple independent data samples, then it is better to define the fit as

$$\chi^2(\mathbf{f}) \coloneqq (\mathbf{K}\ \mathbf{f} - \mathbf{g})^{\mathrm{T}}\mathbf{Cov}^{-\mathbf{1}}(\mathbf{K}\ \mathbf{f} - \mathbf{g})\ .$$

Minimizing this fit function is called the *Generalized Least Squares* which is equivalent to the ordinary least squares but gives more weight to more accurate data components. The weight matrix $\mathbf{W}$ is obtained by taking the Cholesky decomposition of the inverse covariance matrix $\mathbf{Cov}^{-1} = \mathbf{W}^{\mathrm{T}}\mathbf{W}$ and the previous fit reads

$$\chi^2(\mathbf{f}) = (\mathbf{K}\ \mathbf{f} - \mathbf{g})^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{W}(\mathbf{K}\ \mathbf{f} - \mathbf{g}) = \|\mathbf{W}\mathbf{K}\ \mathbf{f} - \mathbf{W}\mathbf{g}\|^2\ . \tag{2.11}$$

So the generalized least squares can be obtained as the least squares of the modified matrix $\mathbf{WK}$ and modified data $\mathbf{Wg}$. Note that the modified data has now an identity covariance matrix, so its noise is uncorrelated and has a standard normal distribution.

Using the least squares solution for solving the analytic continuation problem gives typically an extremely bad solution with extremely large noise. The reason is that the matrix $\mathbf{K}$ has a very large condition number; a concept which will be discussed in the next section. For now, let us take an example and see ill-posedness in practice.

**Test case 1** Consider the analytic continuation of a fermionic imaginary-time Green function described by the following equation

$$\mathcal{G}(\tau) = \int d\omega\ \frac{-e^{-\tau\omega}}{1+\ e^{-\beta\omega}}\ A(\omega)\ . \tag{2.12}$$

[1] We will use the term ”fit” to refer to either $\chi^2$ or its square root $\chi$. The intended use should be clear from the context.

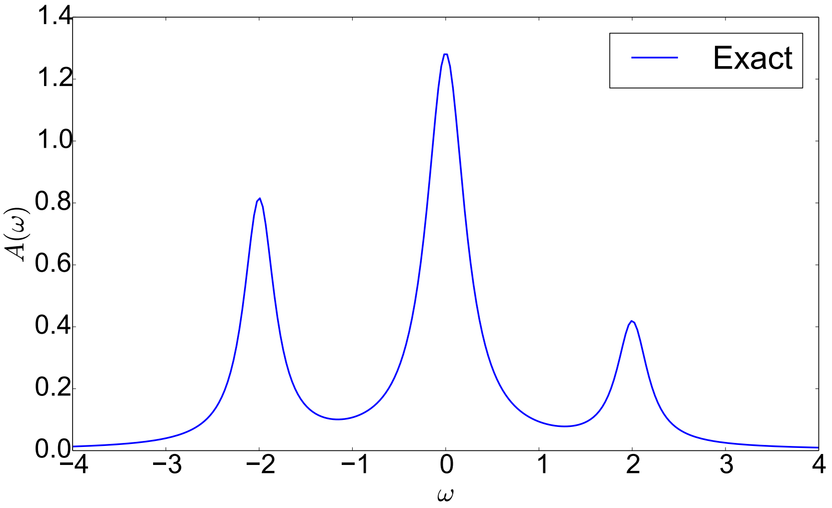


Figure 2.1.: Original spectral function of test case 1. It consists of three Lorentzian peaks with the following half-widths : 0.2, 0.25, 0.2, and the following weights: 0.5, 1.0, 0.25 (left to right).

We take a spectral function composed of three Lorentzian peaks as shown in Fig. 2.1. The integral is discretized using the trapezoidal rule on a uniform grid form $-4$ to $+4$ with 256 points; this same discretization is used for solving the inverse problem. Green function values are generated at 64 equally-spaced $\tau$ points in the interval $[0, \beta]$ with $\beta = 50$. To simulate the effect of computational errors existing in QMC data, we put white noise on the data $\tilde{\mathcal{G}}(\tau_j) = \mathcal{G}(\tau_j) + e_j$ where $e_j$ are normal random variables with zero mean and standard deviation $\sigma = 10^{-4}$. The least squares solution in shown Fig. 2.2 and it is totally dominated by high-frequency noise. This noise has an order of $10^9$ even though the data noise is only of the order of $10^{-4}$! To understand this huge amplification of noise, we turn to the singular value decomposition.

## 2.3. Singular value decomposition

An invaluable tool in studying ill-posed problems is the singular value decomposition (SVD) of a matrix. Every matrix $\mathbf{K} \in \mathbb{R}^{m\times n}$ can be decomposed as

$$\mathbf{K} = \mathbf{U}\ \mathbf{S}\ \mathbf{V}^{\mathrm{T}} \tag{2.13}$$

where $\mathbf{U} \in \mathbb{R}^{m\times m}$ and $\mathbf{V} \in \mathbb{R}^{n\times n}$ are orthogonal matrices [2], and $\mathbf{S} \in \mathbb{R}^{m\times n}$ is a diagonal matrix with diagonal elements $s_1 \geq ... \geq s_p \geq 0$ and $p = \min\{m, n\}$.

The diagonal elements of $\mathbf{S}$ are called the *singular values* of $\mathbf{K}$ and they are ordered descendingly. The number of non-zero values equals the rank of the matrix $r$ and the ratio of the largest singular value to the smallest non-zero one is the *condition number*

[2] A matrix $\mathbf{Q}$ is orthogonal if and only if $\mathbf{Q}^{\mathrm{T}}\mathbf{Q} = \mathbf{I}$.

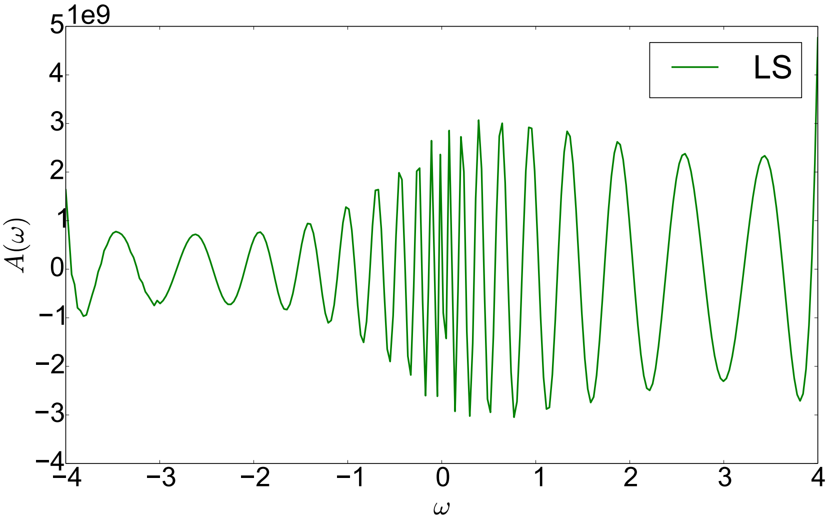


Figure 2.2.: The spectral function of test case 1, reconstructed using the least squares method (LS). High-frequency noise of the order $10^9$ is dominating the solution rendering it useless.

of the matrix

$$\kappa(\mathbf{K}) := \frac{s_1}{s_r} \; . \tag{2.14}$$

The columns of $\mathbf{V}$ (denoted by $\mathbf{v_i}$) form an orthonormal basis of the model space $\mathbb{R}^n$; we call them the *right singular vectors* or *modes* for short. The columns of $\mathbf{U}$ (denoted by $\mathbf{u_i}$) form an orthonormal basis of the data space $\mathbb{R}^m$; we call them the *left singular vectors*. The first $r$ modes are related to the first $r$ right singular vectors by the following relation

$$\mathbf{K} \; \mathbf{v_i} = s_i \; \mathbf{u_i} \quad : i = 1, ..., r \; , \tag{2.15}$$

which is simply a rewriting of the SVD relation (2.13). The remaining modes give zero when multiplied by the matrix

$$\mathbf{K} \; \mathbf{v_i} = 0 \quad : i = r + 1, ..., n \; , \tag{2.16}$$

and thus they form an orthonormal basis of the null space of $\mathbf{K}$. We call these modes *free modes*.

The singular values and vectors can be arbitrary for arbitrary matrices. However, for matrices arising from discretizing analytic continuation kernels, we can identify typical features of their SVD. First, the singular values decay exponentially to zero as shown in Fig. 2.3. The smoother the kernel function $K(y, x)$, the faster the decay [17, Sec. 2.3]. Second, the modes are similar to Fourier functions in the sense that large singular values correspond to smooth modes, and the smaller the singular value $\sigma_i$, the more oscillations in the corresponding mode $\mathbf{v_i}$ [17, Eq. 2.14]. This is evident by checking the modes themselves in Figs. 2.4 or their Fourier components in Fig. 2.5.

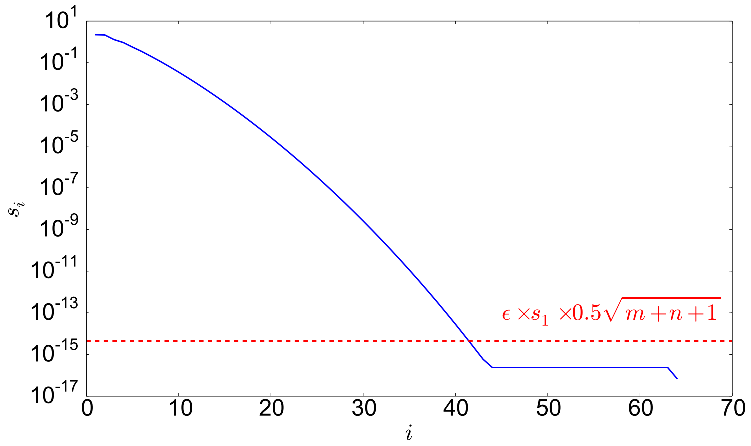


Figure 2.3.: A semi-log plot of the singular values of the matrix test case 1. The exponential decay of singular values is typical for the analytic continuation kernels. They level off at the machine epsilon $10^{-16}$. The dashed line represents the threshold under which we consider singular values to be numerically zero. Therefore, the numerical rank of this matrix is 41 and the condition number equals approximately $10^{14}$.

**Numerical precision** Due to the limited precision of floating-point operations, the singular values level off when their ratio to the largest singular value hits the machine epsilon $\epsilon$. These non-zero but very small singular values and their corresponding modes are probably not correct and their computation is corrupted by roundoff errors. Therefore, one usually sets to zero all such singular values whose ratio to the largest one is less than the machine epsilon times some constant.[3] The numerical rank and condition number are then computed accordingly.

## 2.4. Forward vs. inverse problem

Let us expand the model in the orthonormal basis of the modes

$$\mathbf{f} = \sum_{i=1}^{n} (\mathbf{v_i}^{\mathrm{T}} \mathbf{f})\ \mathbf{v_i}\ , \tag{2.17}$$

then, using SVD as in Eqs. (2.15) and (2.16), the corresponding data can be expressed as a linear combination of the right singular vectors

$$\mathbf{g} = \mathbf{K}\ \mathbf{f} = \sum_{i=1}^{r} s_i\ (\mathbf{v_i}^{\mathrm{T}} \mathbf{f})\ \mathbf{u_i}\ . \tag{2.18}$$

[3] The constant depends on the estimate of roundoff errors. One common choice is $\max(m, n)$. Another choice is $0.5\sqrt{m + n + 1}$ (see [18, p.795 and p.71]).

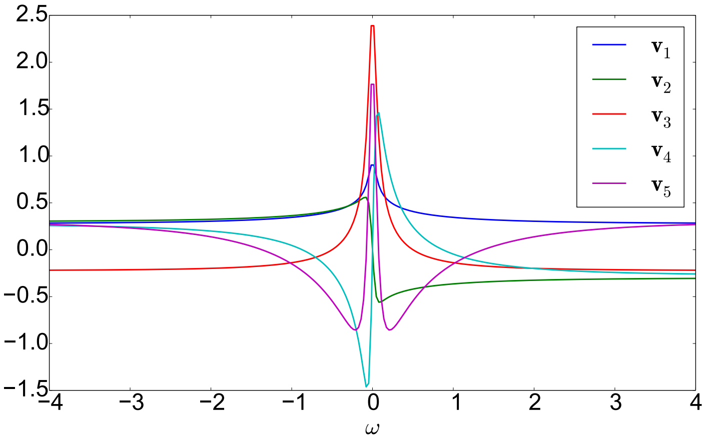


Figure 2.4.: The first five modes of the matrix of test case 1. The number of zero crossings increases as the index of the mode increases and the corresponding singular value decreases. Remember that we included the quadrature factors $\sqrt{\Delta x_i}$ in the matrix $\mathbf{K}$, so we had to divide its modes by these factors before plotting them here as a function of $\omega$.

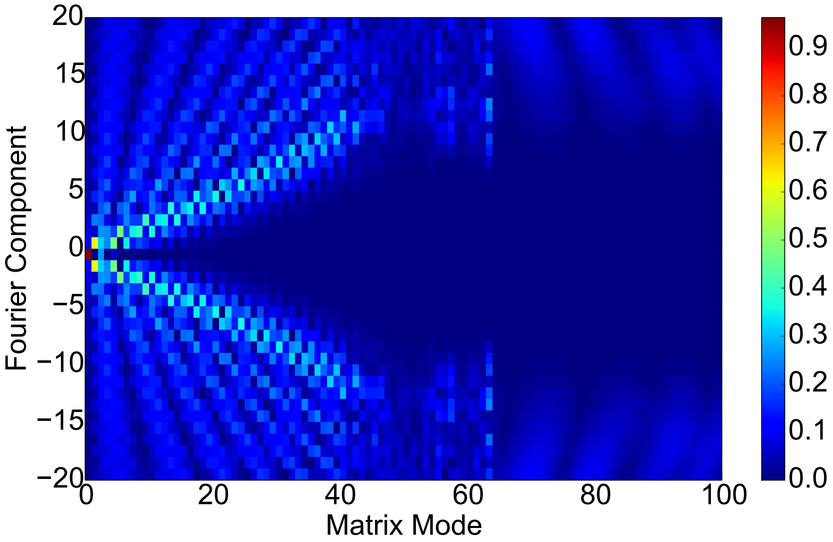


Figure 2.5.: Fourier components of the modes of the matrix of test case 1. Notice that the leading modes are characterized by low frequencies, while later ones are characterized by higher frequencies. Due to numerical degeneracy, free modes (modes > 41) which correspond to zero singular values cannot be characterized by Fourier functions, but they are rather arbitrary linear combinations of high frequencies.

The projection coefficients of the data on $\mathbf{U}$ (we call them *data coefficients*) are the same as the projection coefficients of the model on $\mathbf{V}$ (we call them *model coefficients*) weighted by the corresponding singular values. Since the singular values are decaying, the model coefficients $\mathbf{v_i^T f}$ of later modes and their associated noise get suppressed in comparison to the leading modes. The forward problem is, therefore, well-posed.

Let us look at the inverse problem. Given the data, we want to determine the model minimizing the fit $\chi^2 = \|\mathbf{Kf} - \mathbf{g}\|^2$. By expanding the model in the modes as in Eq. (2.17) and expanding the data in the orthonormal basis of the left singular vectors as following

$$\mathbf{g} = \sum_{j=1}^{m} (\mathbf{u_j}^{\mathrm{T}}\mathbf{g})\ \mathbf{u_j}\ , \tag{2.19}$$

the fit reads

$$\begin{aligned}
\chi^2 &= \| \sum_{i=1}^{r} s_i\ (\mathbf{v_i}^{\mathrm{T}}\mathbf{f})\ \mathbf{u_i} - \sum_{j=1}^{m} (\mathbf{u_j}^{\mathrm{T}}\mathbf{g})\ \mathbf{u_j}\|^2 \\
&= \| \sum_{i=1}^{r}(s_i\mathbf{v_i}^{\mathrm{T}}\mathbf{f} - \mathbf{u_i}^{\mathrm{T}}\mathbf{g})\ \mathbf{u_i} - \sum_{j=r}^{m} (\mathbf{u_j}^{\mathrm{T}}\mathbf{g})\ \mathbf{u_j}\|^2 \\
&= \sum_{i=1}^{r}(s_i\mathbf{v_i}^{\mathrm{T}}\mathbf{f} - \mathbf{u_i}^{\mathrm{T}}\mathbf{g})^2 + \sum_{j=r}^{m} (\mathbf{u_j}^{\mathrm{T}}\mathbf{g})^2\ .
\end{aligned}$$

The first sum can be set to zero by choosing the first $r$ model coefficients as

$$\mathbf{v_i}^{\mathrm{T}}\mathbf{f}_{\mathrm{LS}} = (\mathbf{u_i}^{\mathrm{T}}\mathbf{g})/s_i\ . \tag{2.20}$$

The second sum corresponds to the part of the data laying outside the range of $\mathbf{K}$ and is independent of the model. This sum should ideally be zero, however, due to noise and discretization error, the data lies outside the range of the matrix and this sum determines the minimum fit. Hence, the least squares model minimizing the fit reads

$$\mathbf{f}_{\mathrm{LS}} = \sum_{i=1}^{r} s_i^{-1}\ (\mathbf{u_i}^{\mathrm{T}}\mathbf{g})\ \mathbf{v_i}\ . \tag{2.21}$$

The model coefficients of free modes has no effect on the fit, so we choose to set them to zero to get the unique solution with the minimum norm $\|\mathbf{f}\|$ as requried by Eq. (2.10).

Now we can understand the source of the ill-posedness of the inverse problem using Eq. (2.21). The singular values are exponentially decaying, so their inverse is exponentially increasing and any small noise on the later data coefficients $\mathbf{u_i}^{\mathrm{T}}\mathbf{g}$ gets extremely amplified in computing the corresponding model coefficients $\mathbf{v_i}^{\mathrm{T}}\mathbf{f}_{\mathrm{LS}}$. Since the later modes have higher frequencies than the leading ones, high-frequency noise dominates the least squares solution. As the condition number of the matrix gets larger, the gap between leading and later singular values increases and the problem becomes more ill-conditioned. This amplification of noise on the later modes is clear when comparing the original coefficients $\mathbf{v_i}^{\mathrm{T}}\mathbf{f}$ of test case 1 with the ones obtained using the least squares method (see Fig. 2.6)

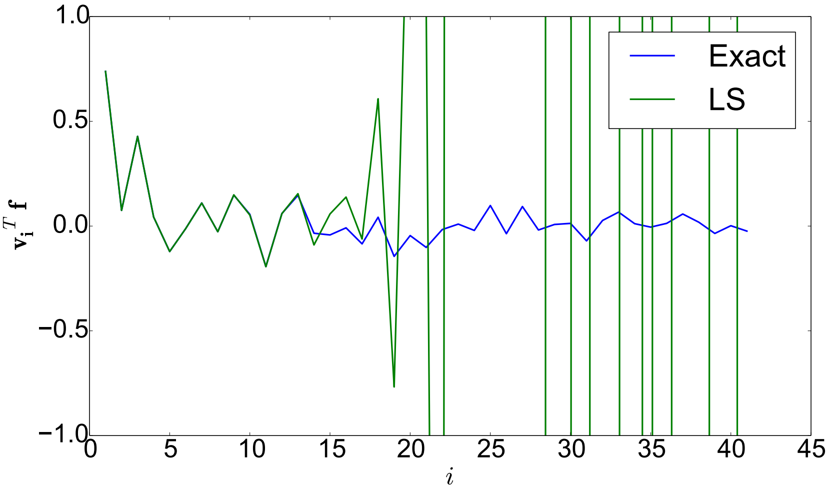


Figure 2.6.: The mode expansion of the original model of test case 1 and its least squares solution (LS).

**Decay of singular values** The singular values of analytic continuation problems decay exponentially, therefore, they hit the machine epsilon very fast. As a result, the condition number is almost always of the order of the inverse of the machine epsilon and the problem is extremely ill-conditioned. Since the condition number in these cases is determined by the numerical precision rather than the matrix under study, we cannot use it as a measure of how ill-conditioned an analytic continuation problem is in comparison to other problems. Therefore, we suggest using the decay of singular values as an alternative measure. Assuming that the singular values decay asymptotically as $O(e^{-\alpha i})$, we use the factor $\alpha$ as a measure of ill-conditioning. It can be estimated from the log difference of the last two non-zero singular values $\alpha \approx \log s_{r-1} - \log s_r$. The larger the difference, the more ill-conditioned the problem. In Fig. 2.7, we plot the singular values of three different matrices. All matrices have roughly the same condition number, but the one with the fastest decaying singular values is the most ill-conditioned one.

**Pseudoinverse** The least squares solution can be expressed concisely in terms of $\mathbf{K}^+$, the pseudoinverse of $\mathbf{K}$

$$\mathbf{f}_{\mathrm{LS}} = \sum_{i=1}^{r} s_i^{-1}\,(\mathbf{u_i}^{\mathrm{T}}\mathbf{g})\ \mathbf{v_i} = \mathbf{K}^+\mathbf{g}\ , \tag{2.22}$$

where the pseudoinverse is defined as

$$\mathbf{K}^+ := \mathbf{V}\ \mathbf{S}^+\ \mathbf{U}^{\mathrm{T}} \text{ with } \mathbf{S}^+ := \mathrm{diag}(s_1^{-1}, ..., s_r^{-1}, 0, ..., 0) \in \mathbb{R}^{n\times m}\ . \tag{2.23}$$

It can be easily checked that the pseudoinverse equals the usual inverse for full rank matrices.

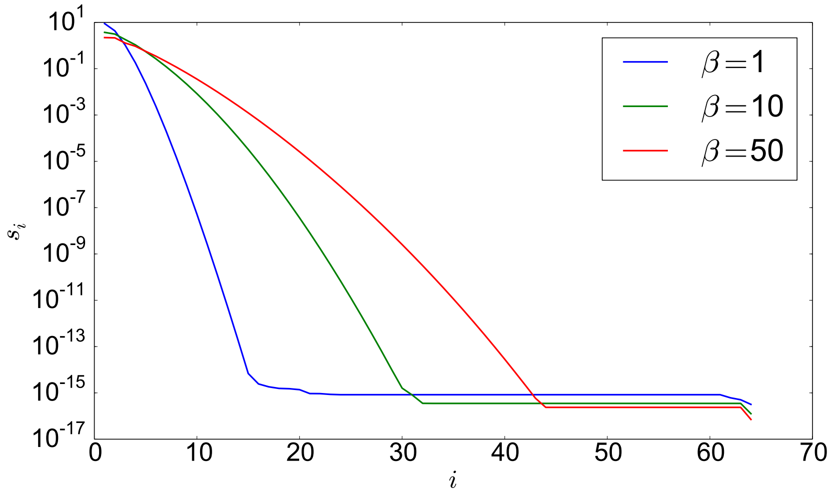


Figure 2.7.: The singular values of matrices resulting from discretizing Eq. 2.12 for different values of the inverse temperature $\beta$. As the temperature increases (or $\beta$ decreases), the singular values decay faster and the matrix becomes more ill-conditioned.

## 2.5. Truncated SVD

In the last section, we saw that the problem of ill-posedness lies in the later modes which are sensitive to noise. An obvious way to regularize this problem is to truncate these modes from the least squares solution i.e. restrict the sum in Eq. (2.21) to the first $k$ modes

$$\mathbf{f}_{\mathrm{TSVD}} = \sum_{i=1}^{k} s_i^{-1} \left(\mathbf{u_i}^{\mathrm{T}}\mathbf{g}\right) \mathbf{v_i} \; . \tag{2.24}$$

This is known as the *truncated SVD method (TSVD)*. The truncation removes the noise associated with the truncated part but it also loses the associated information. Since the modes with higher indices have higher frequencies, the lost information is typically about sharp features. The balance between the noise reduction and information loss can be tuned by the truncation parameter $k$. Smaller $k$ leads to more truncation, less noise and smoother solution and vice versa.

In Fig. 2.6, we see that the model coefficients of the least squares solution deviate noticeably from the original ones starting from the 14th mode. Accordingly, it is best to truncate them and set $k = 13$. Choosing a higher value of the truncation parameter leads to including noisy coefficients in the solution (overfitting) while choosing a lower value unnecessarily discards good coefficients (oversmoothing) (see Fig. 2.24). In this case, we utilized our knowledge about the original model to determine the optimal value of the truncation parameter. However, this information is not available in practice and we have to resort to some other criterion.

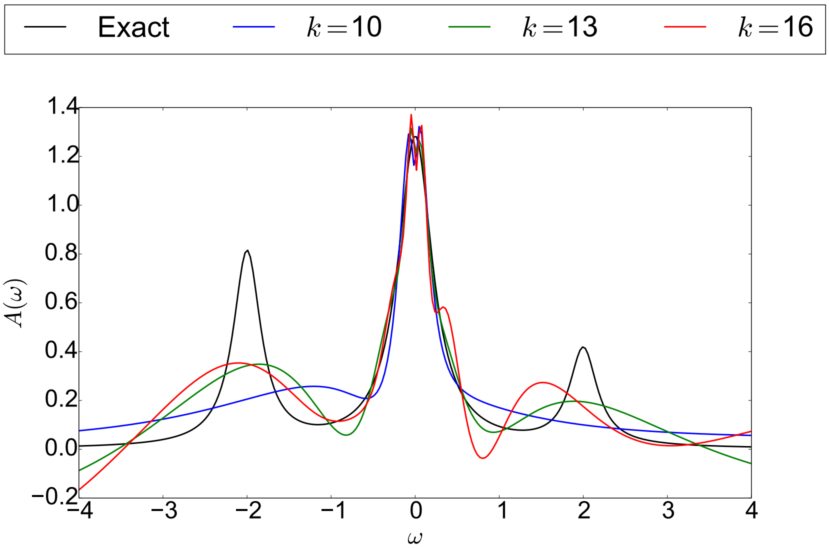


Figure 2.8.: TSVD solutions of test case 1 for different values of the truncation parameter $k$. Based on our knowledge about the original model, the best value is $k = 13$. Lower values lead to oversmoothing while higher values lead to overfitting of noise.

### 2.5.1. Truncation criterion

We argued before that the noise on analytic continuation data is Gaussian with zero mean. Without loss of generality,[4] we further assume that the noise on different components is uncorrelated and has the same standard deviation $\sigma$

$$\mathbf{g}_{\text{noisy}} = \mathbf{g}_{\text{exact}} + \mathbf{e} \qquad \text{where} \;\; \mathbf{e} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}) \; . \tag{2.25}$$

Since the matrix $\mathbf{U}$ is an orthogonal matrix, the noise on the data coefficients $\mathbf{u_j}^{\mathrm{T}}\mathbf{g}$ is also Gaussian with with zero mean and standard deviation $\sigma$

$$\mathbf{u_i}^{\mathrm{T}}\mathbf{g}_{\text{noisy}} = \mathbf{u_i}^{\mathrm{T}}\mathbf{g}_{\text{exact}} + \epsilon_i \qquad \text{where} \;\; \epsilon_i := (\mathbf{u_i}^{\mathrm{T}}\mathbf{e}) \sim \mathcal{N}(0, \sigma^2) \; . \tag{2.26}$$

It is clear then that when the absolute value of a coefficient is large enough i.e. $|\mathbf{u_i}^{\mathrm{T}}\mathbf{g}| > C\sigma$ for some constant $C$, this coefficient is reliable and its relative error is small. Therefore, we choose the truncation parameter $k$, such that these reliable coefficients are retained while noise-dominated ones are discarded.

For example in Fig. 2.9, we show the noisy data coefficients of test case 1. Using $5\sigma$ as a threshold for trustworthy coefficients, we find that the optimal truncation parameter is $k = 13$. This is the same value we got from our knowledge about the original model showing that this criterion is indeed a good heuristic for choosing the truncation parameter. Note that the resulting truncation parameter $k$ is not very sensitive to the choice of the threshold $C$ (here $C = 5$) due to the exponential decay of the coefficients.

[4] Remember that when the covariance matrix is not the identity, we can always take the Cholesky decomposition of the inverse covariance matrix $\mathbf{Cov}^{-1} = \mathbf{W}^{\mathrm{T}}\mathbf{T}$, and solve the modified problem with matrix $\mathbf{WK}$ and data $\mathbf{Wg}$ whose noise is uncorrelated and has a standard normal distribution.

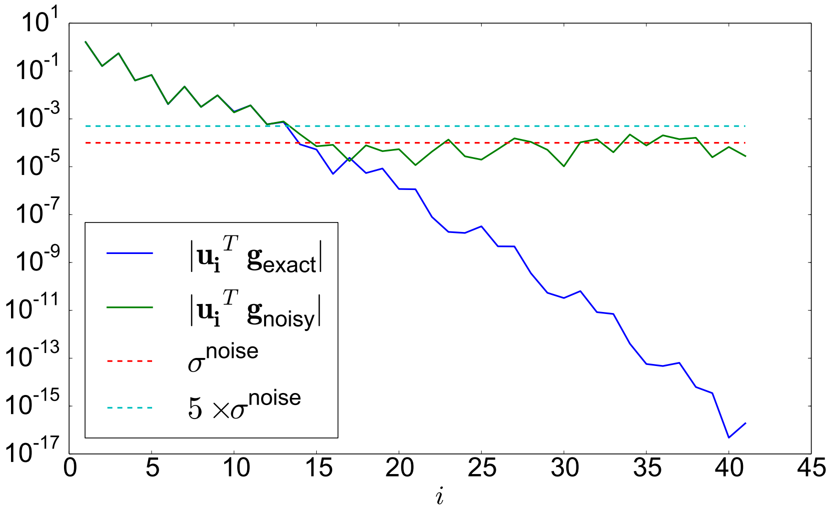


Figure 2.9.: Absolute value of exact and noisy data coefficients of test case 1. The noise is Gaussian with zero mean and $\sigma$ standard deviation. The exact coefficients decay to zero, while the noisy ones decay till they reach noise level $\sigma$ and then fluctuate around it. Notice how large noisy coefficients are very close to the exact ones and the deviation becomes significant only when their values drop to near the noise level.

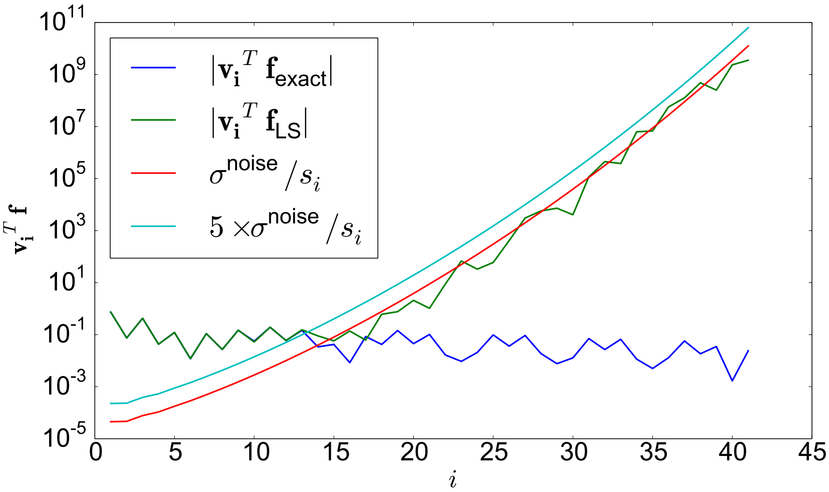


Figure 2.10.: Absolute value of exact and least squares model coefficients of test case 1. The exact coefficients decay slowly to zero while the ones from least squares grow exponentially as fast as the singular values. The reason is that the error on these coefficients is inversely proportional to the singular values.

Since ill-posedness is caused by the exponential decay of the singular values, one may then ask at this point: should not the singular values enter somewhere in the above criterion? They do but rather implicitly. To see this, imagine we generate the data corresponding to some model using two different matrices; one is more ill-conditioned than the other. Assuming the same level of noise in the two cases, the data coefficients produced by the ill-conditioned matrix would decay much faster to the noise level than the other one. Therefore, we have to truncate more coefficients in the ill-conditioned case, and the more ill-conditioned the problem is, the more we have to truncate.

Another way of illustrating the role of the singular values is reformulating the truncation criterion in terms of the model coefficients of the least squares solution. These coefficients are related to the data coefficients by the inverse of the corresponding singular values (see Eq. 2.20) and thus their errors are inversely proportional to the singular values

$$\mathbf{v_i}^{\mathrm{T}}\mathbf{f}_{\mathrm{LS}} = \mathbf{v_i}^{\mathrm{T}}\mathbf{f}_{\mathrm{exact}} + \tilde{\epsilon_i} \qquad \text{where } \tilde{\epsilon_i} := (\mathbf{u_i}^{\mathrm{T}}\mathbf{e}/s_i) \sim \mathcal{N}(0, \sigma^2/s_i^2)\ . \tag{2.27}$$

Therefore, the error dominates those model coefficients whose absolute values are less than a threshold $|\mathbf{v_i}^{\mathrm{T}}\mathbf{f}| < C\sigma/s_i$. The more ill-conditioned the problem is, the faster this threshold grows, and the smaller the number of reliable coefficients is (see Fig. 2.10).

### 2.5.2. Noise estimation

The previous discussion about the truncation criterion provides some insight into estimating the noise level $\sigma$ from the data vector itself. The exact data coefficients are related to model coefficients by the singular values

$$\mathbf{u_i}^{\mathrm{T}}\mathbf{g}_{\mathrm{exact}} = \mathbf{v_i}^{\mathrm{T}}\mathbf{f}_{\mathrm{exact}} s_i\ . \tag{2.28}$$

Assuming that the exact model has a reasonable norm and knowing that singular values are practically zero for $i > r$, the corresponding exact data coefficients are also practically zero and the noisy data coefficients are plain noise

$$\mathbf{u_i}^{\mathrm{T}}\mathbf{g}_{\mathrm{noisy}} \approx \epsilon_i \qquad : r < i \leq m\ . \tag{2.29}$$

Therefore, we can use the variance of these coefficients to estimate the noise variance as

$$\sigma^2 \approx \frac{1}{m-r} \sum_{i=r+1}^{m} (\mathbf{u_i}^{\mathrm{T}}\mathbf{g}_{\mathrm{noisy}})^2\ , \tag{2.30}$$

where we have used the formula for estimating population variance with a known mean 0. This provides a very good estimation of the noise in practice and can also be used to cross-check other estimates of the noise. However, it is very important to remember our assumption that the noise on different components is uncorrelated and has the same standard deviation i.e.

$$\mathbf{e} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})\ . \tag{2.31}$$

If this assumption is not satisfied, then the above estimation is not reliable.

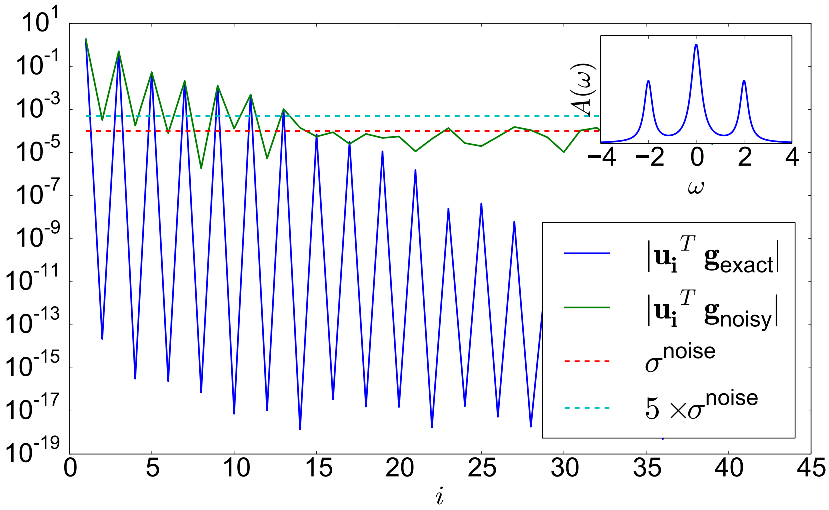


Figure 2.11.: Absolute value of data coefficients of the symmetric model shown in the inset (the kernel and noise level are as in test case 1). Note that the exact value of every other coefficient is practically zero. These coefficients correspond to anti-symmetric modes and are very susceptible to noise. By an appropriate choice of a threshold, they are set to zero in selective SVD method.

### 2.5.3. Selective SVD

Truncated SVD assumes that the leading coefficients are the reliable ones and that they get worse as the index increases. Then it is sufficient to find the optimal truncation position to discard the noise-dominated ones. This is usually a reasonable assumption but we can construct cases where reliable and noise-dominated coefficients are mixed. For example, let us make the model of test case 1 symmetric around zero. Then its projection coefficients on *anti-symmetric* modes are vanishing. Due to their small values, these coefficients are sensitive to noise. So here we have a case of alternating good and bad coefficients where a simple truncation does not discard all noise-dominated coefficients (see Fig. 2.11). A better suited approach here is the so-called *selective SVD* (SSVD) where all noise-dominated coefficients are set to zero [17]

$$\mathbf{f}_{\text{SSVD}} = \sum_{|\mathbf{u_i}^{\text{T}}\mathbf{g}| > C\sigma} s_i^{-1} \left(\mathbf{u_i}^{\text{T}}\mathbf{g}\right) \mathbf{v_i} \,. \tag{2.32}$$

Nevertheless, results for the truncated SVD is usually very close to those for the selective SVD because the noisy model coefficients that are set to zero by SSVD are already small compared to the other coefficients, and thus it does not make a big difference whether they are actually zero or have a very small noisy value.

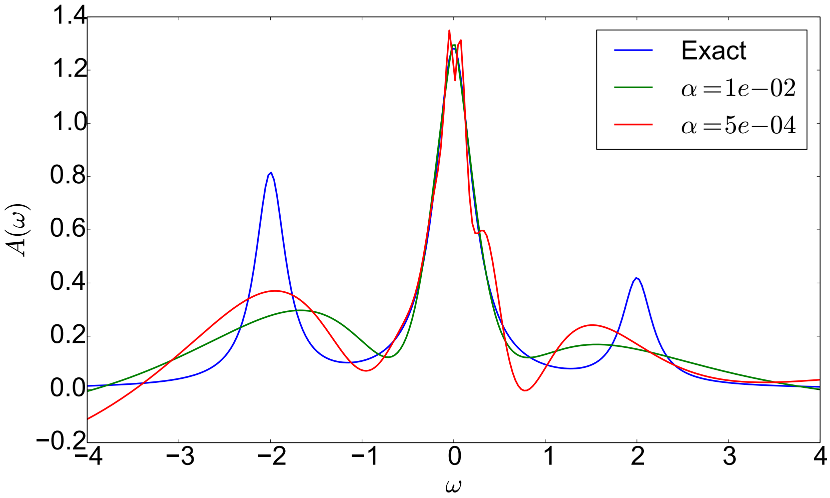


Figure 2.12.: Tikhonov solutions of test case 1 for two different values of $\alpha$. The first value is obtained using the discrepancy principle while the second one is obtained from the L-curve.

## 2.6. Tikhonov regularization

Truncated SVD belongs to a class of regularization methods known as *spectral filtering methods* where the solution is similar to the least squares one but with filtered terms

$$\mathbf{f}_{\text{filtered}} = \sum_{i=1}^{r} \phi_i s_i^{-1} \left(\mathbf{u_i}^{\mathrm{T}}\mathbf{g}\right) \mathbf{v_i} \; . \tag{2.33}$$

The filter function of truncated SVD has a sharp cutoff and is defined as

$$\phi_i^{\text{TSVD}} = \begin{cases} 0 & \text{for } i \leq k \\ 1 & \text{otherwise} \end{cases} . \tag{2.34}$$

Alternatively, we can use a smoother filter that depends on the singular values as in *Tikhonov regularization* which applies the following filter

$$\phi_i^{\text{Thikh}} = \frac{s_i^2}{s_i^2 + \alpha^2} \; , \tag{2.35}$$

where $\alpha$ is an adjustable parameter. Given this filter function, terms corresponding to very small singular values are damped significantly ($\lim_{s_i \to 0} \phi_i = 0$), while the ones corresponding to large singular values are hardly modified ($\lim_{s_i \to \infty} \phi_i = 1$). Substituting in Eq. (2.33), we get the Tikhonov solution

$$\mathbf{f}_{\text{Tikh}} = \sum_{i=1}^{r} \frac{s_i}{s_i^2 + \alpha^2} \left(\mathbf{u_i}^{\mathrm{T}}\mathbf{g}\right) \mathbf{v_i} \; . \tag{2.36}$$

The Tikhonov solution can also be obtained as the unique solution of the following equation

$$(\mathbf{K}^{\mathrm{T}}\mathbf{K} + \alpha^2\mathbf{I})\ \mathbf{f}_{\mathrm{Tikh}} = \mathbf{K}^{\mathrm{T}}\mathbf{g}\ . \tag{2.37}$$

To prove it, we first note that this system has actually a unique solution because the matrix $\mathbf{K}^{\mathrm{T}}\mathbf{K} + \alpha^2\mathbf{I}$ is positive definite and thus non-singular. Using the singular value decomposition of the matrix $\mathbf{K} = \mathbf{USV}^{\mathrm{T}}$ and utilizing that $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices, we write

$$\begin{aligned} \mathbf{V}(\mathbf{S}^{\mathrm{T}}\mathbf{S} + \alpha^2\mathbf{I})\mathbf{V}^{\mathrm{T}}\ \mathbf{f}_{\mathrm{Tikh}} &= \mathbf{VS}^{\mathrm{T}}\mathbf{U}^{\mathrm{T}}\mathbf{g} \\ \Rightarrow (\mathbf{S}^{\mathrm{T}}\mathbf{S} + \alpha^2\mathbf{I})\mathbf{V}^{\mathrm{T}}\ \mathbf{f}_{\mathrm{Tikh}} &= \mathbf{S}^{\mathrm{T}}\mathbf{U}^{\mathrm{T}}\mathbf{g} \end{aligned}$$

Since the matrix $\mathbf{S}^{\mathrm{T}}\mathbf{S} + \alpha^2\mathbf{I}$ is diagonal with diagonal elements $s_i^2 + \alpha^2$, we have

$$(s_i^2 + \alpha^2)\mathbf{v}_i\mathbf{f}_{\mathrm{Tikh}} = s_i(\mathbf{u_i}^{\mathrm{T}}\mathbf{g})\ ,$$

and the solution is indeed the Tikhonov solution (compare to Eq. 2.36).

It is worth noting that Eq. (2.37) is the normal equation of a least squares problem with a modified matrix and modified data

$$\mathbf{f}_{\mathrm{Tikh}} = \underset{\mathbf{f}\in\mathbb{R}^n}{\arg\min} \left\| \begin{pmatrix} \mathbf{K} \\ \alpha\mathbf{I} \end{pmatrix} \mathbf{f} - \begin{pmatrix} \mathbf{g} \\ \emptyset \end{pmatrix} \right\|^2 . \tag{2.38}$$

This formulation has the advantage of getting Tikhonov solution without an explicit computation of the singular value decomposition which can be a huge computational effort for large scale problems.

We can also easily rewrite Eq. (2.38) as the following minimization problem

$$\mathbf{f}_{\mathrm{Tikh}} = \underset{\mathbf{f}\in\mathbb{R}^n}{\arg\min}\ \chi^2(\mathbf{f}) + \alpha^2 \left\|\mathbf{f}\right\|^2\ , \tag{2.39}$$

This allows us to interpret the Tikhonov solution as the one that balances between the fit to the data and the model norm. The balance is controlled by the regularization parameter $\alpha$. When $\alpha$ is very small, we approach the least squares solution which fits the data very well but has a very large norm. As $\alpha$ increases, the solution becomes smoother with smaller norm but worse fit. We will discuss below two commonly-used heuristics for choosing the optimal $\alpha$. It is worth noting that Tikhonov regularization can generalized by replacing the model norm in the last equation with some other bilinear function of the model vector.

**Discrepancy principle** Assuming that the noise on the data is uncorrelated and has standard deviation $\sigma$ as in Eq. (2.25), the expected norm of the noise is $\|\mathbf{e}\| = \sqrt{m}\sigma$, where $m$ is the size of data/noise vector. The discrepancy principle simply says that a good model would produce data such that the residual vector $\mathbf{K}\ \mathbf{f} - \mathbf{g}$ is dominated by noise. Therefore, we choose $\alpha$ such that the norm of the residual (i.e. the fit) roughly equals the expected norm of the noise vector. For safety, it is better to be slightly larger

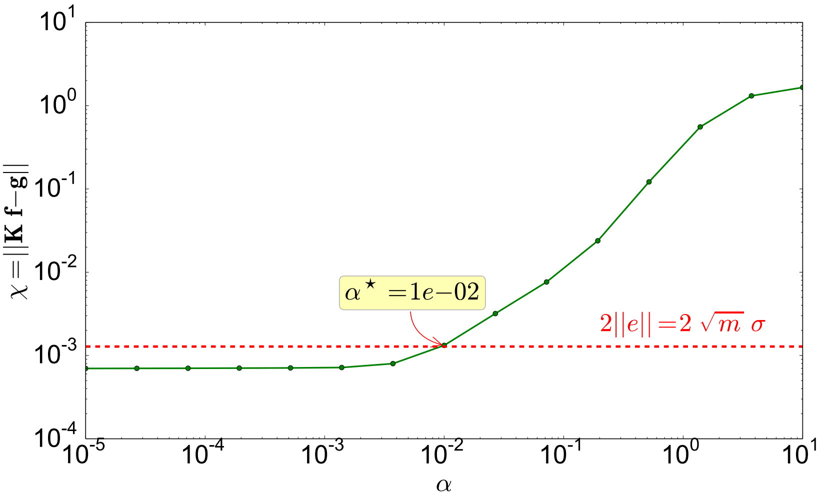


Figure 2.13.: Fits of Tikhonov solutions versus $\alpha$ for test case 1. Notice how the fit increases monotonically with $\alpha$. According to the discrepancy principle, the best value of $\alpha$ is the one where the fit equals a multiple (here twice) of the expected norm of the noise (dashed line).

by some factor $C$. In other words, find $\alpha$ such that $\|\mathbf{K}\ \mathbf{f} - \mathbf{g}\| = C\sqrt{m}\sigma$ which has a unique solution because the residual norm increases monotonically with $\alpha$. In Fig. 2.13, we plot the fits of Tikhonov solutions of test case 1 . Setting $C = 2$, we get the optimal value $\alpha = 1\mathrm{e}{-2}$ and the corresponding solution to is shown in Fig. 2.12. Ref. [19] uses a similar approach for determining the regularization parameter of MaxEnt.

**L-curve** The formulation of Tikhonov solution as a balance between the data fit and the model norm (see Eq. 2.39) motivates the L-curve method [20]. This method suggests plotting the model norm $\|\mathbf{f}\|$ versus the fit $\chi(\mathbf{f})$ on a log-log scale for different values of $\alpha$. The curve will have an L-shape and the value of $\alpha$ at the corner of the L-curve is the best value balancing between the data fit and the model norm. In Fig. 2.14, we show the L-curve for test case 1. For small $\alpha$, the solution fits the data very well but is dominated by very large noise. Therefore, the fit saturates around the best possible fit while the model norm explodes. For large $\alpha$, the fit gets worse as more of the leading modes get filtered out while the model norm plateaus around the norm of the exact model before it drops to zero. The optimal value is the one at the corner $\alpha = 5\mathrm{e}{-4}$ and its corresponding solution is plotted in Fig. 2.12.

**L-curve Vs. discrepancy principle** The discrepancy principle is more conservative (especially with a large constant $C$), while the L-curve tries to get the most out of the data. This means that the L-curve will outperform the discrepancy principle when the actual noise happens to be smaller than expected. On the other hand, it may as well over-fit noise that does not affect the model norm dramatically. For example, in Fig. 2.12, we plot the Tikhonov solutions of test case 1 obtained by the two methods. The

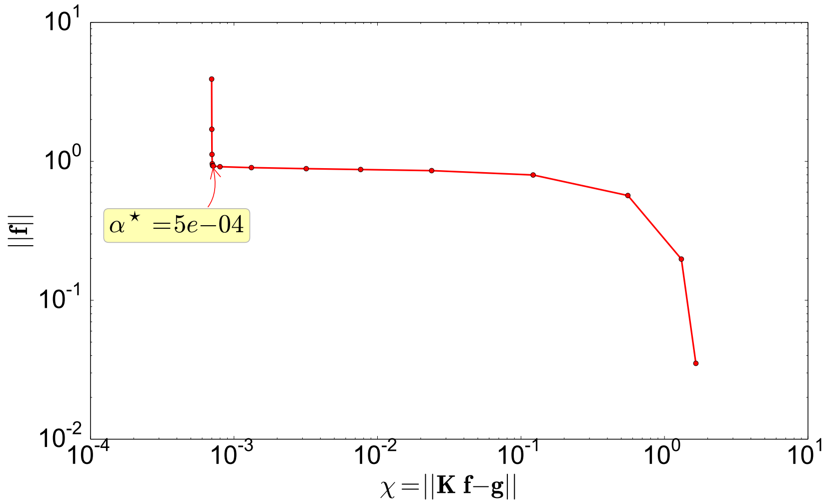


Figure 2.14.: L-curve for test case 1; a log-log plot of the model norm versus the data fit of Tikhonov solutions for different values of $\alpha$. According to the L-curve heuristic, the best value of $\alpha$ is the one corresponding to the corner.

L-curve gives a lower value of $\alpha$ which leads to some extra noise-related features in the solution, while the discrepancy principle gives a smoother solution with no overfitting. Both methods are heuristics and which works better depends on the test case. As a general strategy, we prefer using the discrepancy principle when a good estimate of the standard deviation of the noise is available, while we resort to the L-curve when no such estimate exits.

### 2.6.1. Differential formulation of Tikhonov regularization

Interestingly, we can formulate the Tikhonov solution as an approximation to the following differential equation

$$\frac{d\mathbf{f}}{dt} = -\left(\mathbf{K}^{\mathrm{T}}\mathbf{K}\right)^{+}\mathbf{f}(t)\ , \tag{2.40}$$

with the least squares solution as an initial condition

$$\mathbf{f}(0) = \mathbf{f}_{\mathrm{LS}} = \mathbf{K}^{+}\mathbf{g}\ , \tag{2.41}$$

where the symbol $^+$ denotes the pseudoinverse defined in (2.23) and $t$ is a fictitious time parameter.

Let us apply the implicit Eular scheme with time step $\Delta t$ to this differential equation

$$\Delta t^{-1}\left(\mathbf{f}^{t_{i+1}} - \mathbf{f}^{t_i}\right) = -\left(\mathbf{K}^{\mathrm{T}}\mathbf{K}\right)^{+}\mathbf{f}^{t_{i+1}} \tag{2.42}$$

$$\Rightarrow\quad \mathbf{K}^{\mathrm{T}}\mathbf{K}\left(\mathbf{f}^{t_{i+1}} - \mathbf{f}^{t_i}\right) = -\Delta t\ \mathbf{f}^{t_{i+1}} \tag{2.43}$$

$$\Rightarrow\quad (\mathbf{K}^{\mathrm{T}}\mathbf{K} + \underbrace{\Delta t}_{\alpha^2}\ \mathbf{I})\ \mathbf{f}^{t_{i+1}} = \mathbf{K}^{\mathrm{T}}\ \underbrace{\mathbf{K}\ \mathbf{f}^{t_i}}_{\mathbf{g}^{t_i}} \tag{2.44}$$

Comparing this relation to Eq. (2.37), we see that taking a single time step $\Delta t$ is equivalent to applying Tikhonov regularization to the model of the previous time step with a regularization parameter $\alpha = \sqrt{\Delta t}$. We discovered this relation independently and found out later that a closely related connection has already been established in the context of image deblurring [21].

The differential equation (2.40) can even be solved exactly with the help of the singular value decomposition and it leads to yet another spectral filtering method. Using SVD, we can write

$$\mathbf{K} = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathrm{T}} \Rightarrow \mathbf{K}^{\mathrm{T}}\mathbf{K} = \mathbf{V}\mathbf{S}^{2}\mathbf{V}^{\mathrm{T}} \Rightarrow \left(\mathbf{K}^{\mathrm{T}}\mathbf{K}\right)^{+} = \mathbf{V}\mathbf{S}^{+2}\mathbf{V}^{\mathrm{T}} \tag{2.45}$$

Substituting in the Eq. 2.40, we get

$$\frac{d\mathbf{f}}{dt} = -\mathbf{V}\mathbf{S}^{+2}\mathbf{V}^{\mathrm{T}}\ \mathbf{f}(t) \tag{2.46}$$

$$\Rightarrow \mathbf{V}^{\mathrm{T}}\frac{d\mathbf{f}}{dt} = -\mathbf{S}^{+2}\mathbf{V}^{\mathrm{T}}\ \mathbf{f}(t) \tag{2.47}$$

$$\Rightarrow \frac{d\mathbf{c}}{dt} = -\mathbf{S}^{+2}\ \mathbf{c}(t)\ , \tag{2.48}$$

where we denoted model coefficients as $\mathbf{c} := \mathbf{V}\mathbf{f}$. Since the matrix $\mathbf{S}^{+2}$ is diagonal with diagonal elements $s_i^{-2}$, the differential equations of different coefficients are decoupled and can be easily solved

$$\frac{dc_i}{d_t} = -s_i^{-2}c_i(t) \tag{2.49}$$

$$\Rightarrow \quad c_i(t) = c_i(0)\ e^{-t/s_i^2} \tag{2.50}$$

Using the initial condition (2.41), we see that $c_i(0)$ are nothing but the least squares coefficients and the solution is a filtered least squares solution

$$\mathbf{f}(t) = \sum_{i=1}^{r} e^{-t/s_i^2}\ \left[s_i^{-1}\ (\mathbf{u_i}^{\mathrm{T}}\mathbf{g})\ \mathbf{v_i}\right]\ . \tag{2.51}$$

So each model coefficient starts with its least squares value and then decays exponentially with a lifetime that equals the square of the singular value. Therefore, modes corresponding to small singular values die out quickly while the ones with larger singular values survive longer.

## 2.7. Non-negative least squares

The previous methods are general and apply to all inverse problems. Now we utilize a simple, yet important, piece of knowledge about the analytic continuation problem: *the non-negativity of the model.* The first step is restricting the least squares method to non-negative models

$$\mathbf{f}_{\mathrm{NNLS}} = \underset{\mathbf{f}\in\mathbb{R}^n,\mathbf{f}\geq 0}{\arg\min}\ \chi^2(\mathbf{f})\ , \tag{2.52}$$

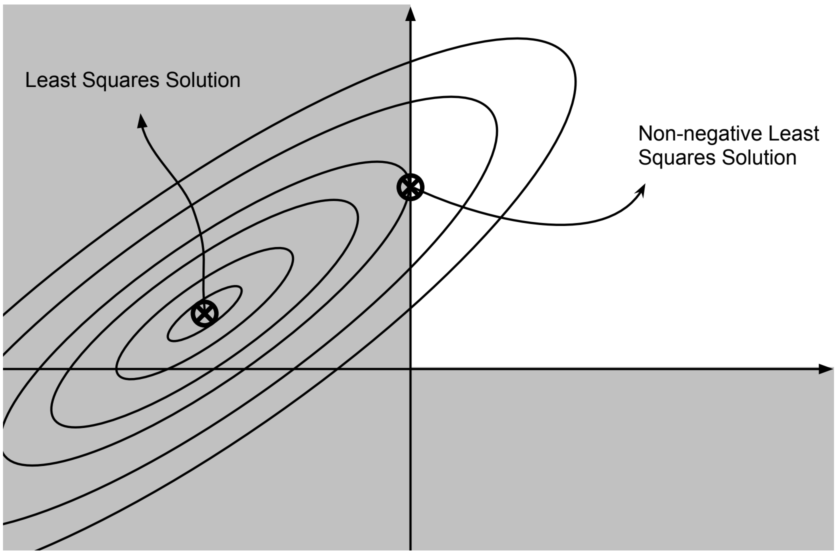


Figure 2.15.: An illustration of the difference between the least squares solution and the non-negative least squares solution for a two-dimensional case. The ellipses represent the contour of the fit function that least squares methods try to minimize. Least squares searches the whole plane for the minimum fit, while non-negative least squares restricts it search to the non-negative quadrant.

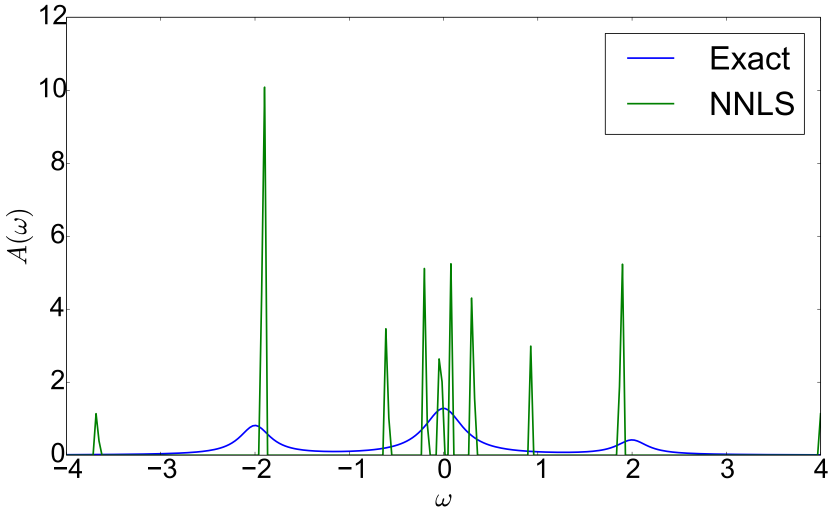


Figure 2.16.: The spectral function of test case 1, reconstructed using the non-negative least squares method (NNLS). The solution has roughly the same zeroth, first and second moments of the exact spectral function.

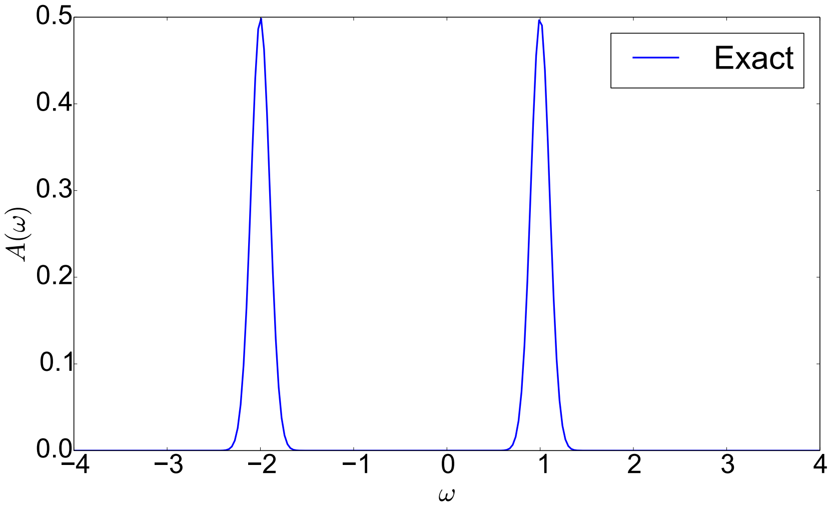


Figure 2.17.: Original spectral function of test case 2. It consists of two Gaussian peaks; both have width 0.1 and weight 0.5.

which is known as the *non-negative least squares solution (NNLS)*. Fig. 2.15 shows a simple illustration of the difference between the least squares solution and the non-negative least squares solution for a two-dimensional case. The two solutions can, in principle, be the same but it is highly unlikely because the noise typically throws the least squares solution outside the allowed region.

In Fig. 2.16, we show the non-negative least squares solution for test case 1. This solution is clearly a huge improvement over the least squares (LS) one shown in Fig. 2.2. While the LS solution is completely useless with huge oscillations of the order $10^9$, the NNLS solution, although still pretty bad, captures at least some information about the model. NNLS solution has the same order of magnitude as the original model (similar zeroth moment) and is concentrated in roughly the same region where the original is concentrated (similar first and second moments). So simply taking the non-negativity constraints into account already provides some kind of regularization. In other cases, it can even provide more information as in the following test case.

**Test case 2** The setting of this test case is exactly the same as test case 1 except that the exact spectral function is composed of two sharp Gaussian peaks separated by a large gap (see Fig. 2.17). The non-negative least squares solution is shown in Fig. 2.18 and it has several sharp peaks; two of them correspond to the original peaks. The positions of these are quite accurate but their widths are shrunk to the grid spacing. This sharp structure is typical in NNLS solutions and can be understood intuitively from Fig. 2.15, where the NNLS solution usually lies on the boundary of the non-negative region and thus has many zeros. Mathematically, it is the result of KKT conditions explained in the next section (see Eqs. 2.54 and 2.55).

We can see the regularizing effect of non-negativity more clearly by checking the data produced by the NNLS solution $\mathbf{g}_{\text{NNLS}} := \mathbf{K}\mathbf{f}_{\text{NNLS}}$ and comparing it to the exact and

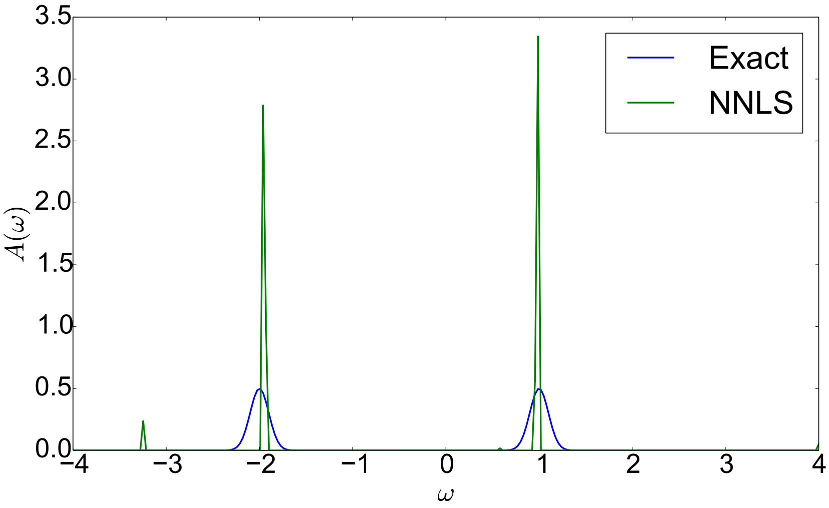


Figure 2.18.: The spectral function of test case 2 reconstructed using the non-negative least squares method (NNLS). The large two peaks in the NNLS solution correspond to the original peaks and are located roughly where they should be. However, no information about their widths is available in the NNLS solution.

noisy data. In Fig. 2.19, we plot the the projection coefficients of these data vectors on the right singular vectors $\mathbf{U}$ for test case 1. While the noisy data coefficients level off around the noise level, the NNLS data coefficients have a similar decay to the exact data, and despite the differences between the two, NNLS gives us the right asymptotic behavior of the exact coefficients. The non-negativity is even more informative for test case 2 whose plot is shown in Fig. 2.20. In this case, the constraints give the right values of extra 8 coefficients below the noise level.

There are different algorithms for solving the non-negative least squares problem. The algorithm of Lawson and Hanson [22] was the first to appear in the literature. We explain this algorithm and why it works, and suggest a modification that improves its convergence. Numerical stability and issues caused by round-off errors are also discussed. For an overview of other non-negative least squares algorithms, refer to [23].

### 2.7.1. Karush-Kuhn-Tucker conditions

We can see NNLS as a minimization problem of the following quadratic objective function subject to inequality constraints[5]

$$\begin{aligned} \arg\min_{\mathbf{f}} \quad & \psi(\mathbf{f}) = \frac{1}{2}\mathbf{f}^{\mathrm{T}}\mathbf{K}^{\mathrm{T}}\mathbf{K}\mathbf{f} - \mathbf{f}^{\mathrm{T}}\mathbf{K}^{\mathrm{T}}\mathbf{g} \\ \text{s.t.} \quad & \mathbf{f} \geq 0 \end{aligned} \tag{2.53}$$

[5] You can easily check that this is equivalent to minimizing $\chi^2(\mathbf{f}) = 2\psi(\mathbf{f}) + \mathbf{g}^{\mathrm{T}}\mathbf{g}$. Also recall that all quantities are real and that a complex case can be handled as a real one of twice the size (see Eq. 2.8).

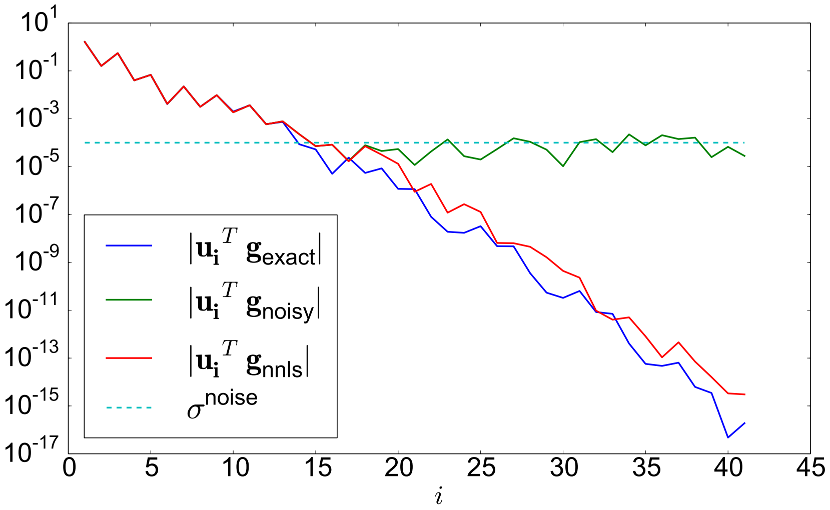


Figure 2.19.: Absolute value of exact, noisy and NNLS data coefficients of test case 1. Using only the non-negativity constraints, NNLS method helps us getting the right asymptotic decay of the exact coefficients.

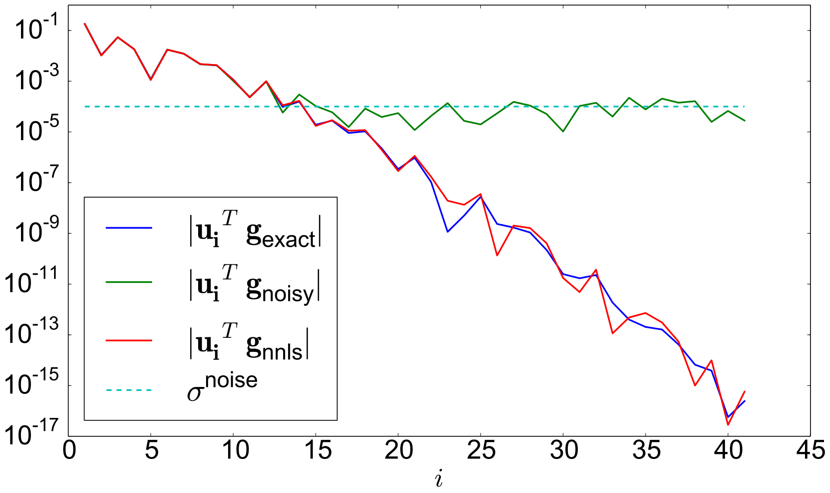


Figure 2.20.: Absolute value of exact, noisy and NNLS data coefficients of test case 2. Notice how NNLS method retrieves the exact values of coefficients 15-22, although the corresponding noisy data coefficients are dominated by noise.

The Karush-Kuhn-Tucker (KKT) conditions provide necessary conditions for the solution of non-linear optimization problems [25]. These conditions are also sufficient [24] when the objective function and the inequality constraints are convex, which is the case here.

Applying these conditions to NNLS, we find that an $n$-vector $\mathbf{f}$ is a solution of Eq. 2.53 (or equivalently Eq. 2.52) if and only if for each index $i$ either:

$$f_i = 0, \quad \Delta_i > 0 \quad \text{(active)} \tag{2.54}$$

$$f_i > 0, \quad \Delta_i = 0 \quad \text{(passive)} \tag{2.55}$$

where $\mathbf{\Delta} := \mathbf{K}^{\mathrm{T}}(\mathbf{g} - \mathbf{Kf})$ is the negative gradient vector of $\psi(\mathbf{f})$.

These conditions allow us to divide the indices of the solution into two sets: an *active set* (Eq. 2.54) and a *passive set* (Eq. 2.55). The active set refers to the indices where the constraints are active, and consequently the solution is zero. The passive set refers to the indices where constraints are passive, and hence the solution is strictly positive. The negative gradient is positive for the active set which means that trying to minimize the residual in any of these directions will violate the constraints. On the other hand, the gradient is zero for the passive set, so the solution has its optimal values along the passive directions. **Consequently, the NNLS solution is nothing but the unconstrained least squares solution using the passive set components only.**

### 2.7.2. The algorithm

Using the observation from last section, a naive algorithm is to compute the unconstrained least squares solutions for all possible passive sets (all subsets of the index set $\{1, ..., n\}$). Some solutions will have some negative components while others are strictly positive. Among these strictly positive solutions, the NNLS solution is the one with the minimum residual. This naive algorithm takes a finite time but it has an exponential complexity because the number of possible passive sets is $2^n$.

The Lawson-Hanson algorithm does much better by starting from an empty passive set and updating this set to get a lower residual at each step. The algorithm stops when KKT conditions are satisfied i.e. the gradient is negative for all indices in the active set, and any further reduction in the residual would lead to violation of the constraints. In case of degeneracy, this algorithm converges to one of the multiple solutions. Here is a pseudocode of the algorithm:

```
function NNLS($\mathbf{K}, \mathbf{g}$)
    $\mathbf{f} \leftarrow 0$
    $\mathcal{P} \leftarrow \{\}$                                         ▷ passive set
    $\mathcal{Z} \leftarrow \{1, ..., n\}$                                ▷ active set
    $\mathbf{\Delta} \leftarrow \mathbf{K}^{\mathrm{T}}\mathbf{g}$            ▷ initial negative gradient vector
    while $\mathcal{Z} \neq \phi$ and $\max_{i \in \mathcal{Z}} \Delta_i > 0$ do      ▷ loop till KKT condition satisfied
        $t \leftarrow \arg\max_{i \in Z} \Delta_i$          ▷ choose an index to move to the passive set
        $\mathcal{Z} \leftarrow \mathcal{Z} \setminus t$
        $\mathcal{P} \leftarrow \mathcal{P} \cup t$
        $\mathbf{f}' \leftarrow LS(\mathbf{K}^{\mathcal{P}}, \mathbf{g})$    ▷ least squares with the passive set
        while $\min_{i \in \mathcal{P}} f'_i \leq 0$ do    ▷ loop till constraints are satisfied
            $\alpha \leftarrow \min_i f_i/(f_i - f'_i) : f'_i \leq 0, i \in \mathcal{P}$    ▷ interpolation factor
            $\mathbf{f} \leftarrow \mathbf{f} + \alpha(\mathbf{f}' - \mathbf{f})$    ▷ interpolate
            Update $\mathcal{P}$ and $\mathcal{Z}$    ▷ move indices of zero values form $\mathcal{P}$ to $\mathcal{Z}$
            $\mathbf{f}' \leftarrow LS(\mathbf{K}^{\mathcal{P}}, \mathbf{g})$    ▷ least squares with the passive set
        end while
        $\mathbf{f} \leftarrow \mathbf{f}'$    ▷ update solution
        $\boldsymbol{\Delta} \leftarrow \mathbf{K}^{\mathrm{T}}(\mathbf{g} - \mathbf{K}\mathbf{f})$    ▷ new negative gradient vector
    end while
    return f
end function
```

The algorithm has two loops: an outer one and an inner one. The outer loop keeps iterating till the KKT conditions are satisfied. Each iteration reduces the residual of the solution vector **f** by moving the index of the maximum negative gradient to the passive set. The unconstrained least squares solution $\mathbf{f}'$ using the new passive set has definitely a lower residual because there are more degrees of freedom and the gradient was non-zero. At the end of each iteration, the vector $\mathbf{f}^{\mathcal{P}}$ is the unconstrained least squares solution using the passive set and it should be strictly positive (outer loop invariant). If the candidate solution $\mathbf{f}'$ is non-negative, this iteration of the outer loop has achieved its goal and the candidate solution is used as the current solution. Otherwise, the candidate solution $\mathbf{f}'$ lies outside the allowed region and it is used by the inner loop to get a new allowed solution with lower residual than the current one **f**. This is done by finding the intersection point of the line connecting **f** and $\mathbf{f}'$ with the boundary of the allowed region (see Fig. 2.21). The intersection point has a lower residual than the current solution **f** because it is closer to $\mathbf{f}'$. It also has fewer passive components because it lies on the boundary. This intersection point is then used as the current solution **f** for the next iteration of the inner loop. Hence, each iteration of the inner loop moves some components to the active set and reduces the residual. The inner loop keeps iterating till the unconstrained least squares solution is positive.

**Convergence and computational cost** The convergence of the algorithm is guaranteed by the finiteness of both the inner and outer loops. The inner loop will have at worse $k-1$ iterations,[6] where $k$ is the size of the passive set upon entering the loop. Regarding the outer loop, the residual gets lower each iteration, and thus the solution **f** and its corresponding passive set $\mathcal{P}$ are distinct from all their previous values. This proves the finiteness of the outer loop because there are only a finite number of values for $\mathcal{P}$, namely $2^n$, the number of the subsets of $\{1, ..., n\}$. Like the naive algorithm, this algorithm also has an exponential worst case complexity, but in practice it is quite fast with a few

[6] It is $k-1$ rather than $k$, because the recently added index to $\mathcal{P}$ is guaranteed to be positive by Lemma [22, 23.17].

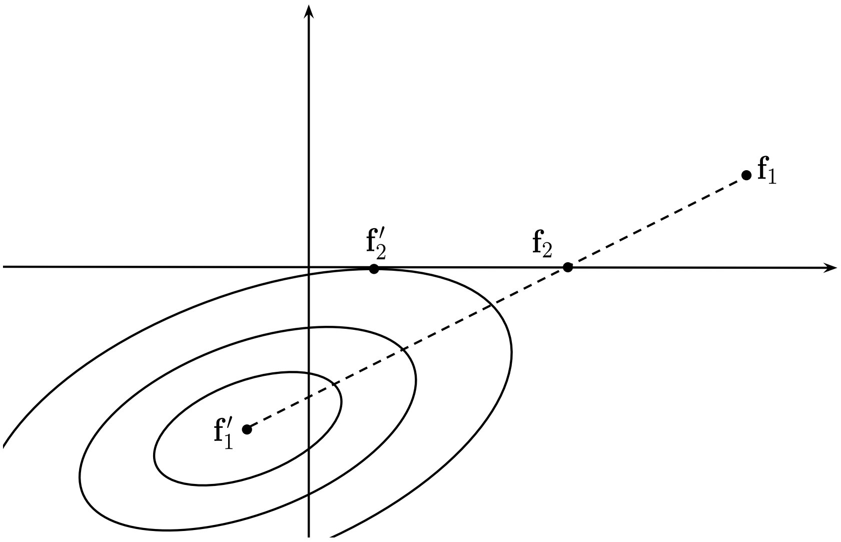


Figure 2.21.: 2D illustration of the inner loop of the NNLS algorithm. Upon entering the first iteration, $\mathbf{f_1}$ is the current solution and $\mathbf{f'_1}$ is the candidate solution (the unconstrained least squares solution). The point $\mathbf{f_2}$ is the interpolation between $\mathbf{f_1}$ and $\mathbf{f'_1}$ that is closest to $\mathbf{f'_1}$ and still non-negative. It is used as the current solution for the second iteration. Solving the unconstrained least squares solution in the second iteration gives $\mathbf{f'_2}$ which is already non-negative, so the loop terminates with $\mathbf{f'_2}$ as the current solution.

hundred iterations for typical test cases. We observed that reducing the noise level $\sigma$ to very small values like $10^{-8}$ increases the number of iterations considerably. Also note that the cost of each iteration increases with the size of data $m$ because it affects the cost of least squares solution at steps 10 and 15.

### 2.7.3. Modified algorithm

At step 7 of the NNLS algorithm, the index with the maximum value for the negative gradient is chosen to be moved from the active to the passive set. This is inspired by the gradient descent algorithm which takes successive steps in the directions of the negative gradient to reach a local minimum. NNLS, however, does not take the direction of the negative gradient but the direction of the component on which the negative gradient has the maximum projection.

This choice is not only non-optimal, but even arbitrary in some sense! Let us see how. Solving Eq. 2.52 can be done equivalently by solving a scaled problem and then rescaling the solution back, i.e.

$$\mathbf{f}_{\text{NNLS}} = \mathbf{D}^{-1} \underset{\mathbf{f}\in\mathbb{R}^n,\mathbf{f}\geq 0}{\arg\min} \, \|\mathbf{KDf}' - \mathbf{g}\|^2 \tag{2.56}$$

where $\mathbf{D}$ is a strictly positive diagonal matrix. By making the $i$-th diagonal element of $\mathbf{D}$ arbitrary large, we can force the algorithm to choose the $i$-th component to be moved to the passive set regardless of the current solution or the matrix $\mathbf{K}$ (as long as $\Delta_i$ is positive, of course).

We propose a minor modification that leads to a more reasonable choice of the index. Rescale the components of negative gradient vector by the norms of the corresponding columns of the matrix $\mathbf{K}$. Then choose the index with largest value of the new rescaled vector. This choice of the index gives the lowest fit conditional on the values of the current passive variables, which can be proven easily as following.

Lowering the fit conditionally using an additional passive variable$f_i$ means fitting the residual vector $\mathbf{r} := \mathbf{g} - \mathbf{K}\mathbf{f}$ using $\mathbf{K}_i$, the $i$-th column of the matrix $\mathbf{K}$. This is a least squares problem with one variable

$$\mathbf{K}_i f_i = \mathbf{r} \Rightarrow \mathbf{K}_i^{\mathrm{T}} \mathbf{K}_i f_i = \mathbf{K}_i^{\mathrm{T}} \mathbf{r} \Rightarrow f_i^\star = \frac{\mathbf{K}_i^{\mathrm{T}} \mathbf{r}}{\mathbf{K}_i^{\mathrm{T}} \mathbf{K}_i} \; , \tag{2.57}$$

and its fit reads

$$\|\mathbf{r} - \mathbf{K}_i f_i^\star\|^2 = \mathbf{r}^{\mathrm{T}} \mathbf{r} - \frac{(\mathbf{K}_i^{\mathrm{T}} \mathbf{r})^2}{\mathbf{K}_i^{\mathrm{T}} \mathbf{K}_i} \; . \tag{2.58}$$

Therefore, we can minimize this fit by choosing the index that maximizes $\mathbf{K}_i^{\mathrm{T}} \mathbf{r} / \sqrt{\mathbf{K}_i^{\mathrm{T}} \mathbf{K}_i}$ which is nothing but the aforementioned *rescaled* negative gradient.

This modified algorithm chooses the same index even for a rescaled matrix, so it does not suffer from the same arbitrariness as the original algorithm. Moreover, we find that in practice, the modified algorithm converges in a smaller number of iterations than the original one. Sometimes the reduction is just a few iterations, while in other cases it can be an order of magnitude. Therefore, we recommend to always use this modification which can be applied easily to existing implementations of the original algorithm. Simply normalize the columns of the matrix before passing it to the algorithm, and then rescale the returned solution by the norms of the columns.

### 2.7.4. Numerical stability and round-off errors

- Comparison of the maximum negative gradient (or the rescaled one for the modified algorithm) with zero at step 6, should be replaced by a comparison with some tolerance value.

- When updating the passive and active sets at step 11, any component of $\mathbf{f}$ that is less than some tolerance, should be set to zero and moved to the active set.

- Rescaling the columns of the matrix $\mathbf{K}^{\mathcal{P}}$ by their norm and ordering them such that the recently added column is at the end, provides more numerical stability to the least squares solutions at steps 10 and 15.

- Computing the negative gradient at step 17 involves computing the residual vector $\mathbf{g} - \mathbf{K}\mathbf{f}$. It is more numerically stable to compute this vector by projecting $\mathbf{g}$ onto the range of the matrix $\mathbf{K}^{\mathcal{P}}$. QR decomposition provides a vector basis set for this range. This does not lead to any computational overhead because this decomposition is already needed for the least squares solution at steps 10 and 15.

- According to Lemma [22, 23.17], $f_t$ should theoretically be greater than zero at step 11. In practice, it may be zero or negative which indicates numerical errors in computing $\Delta_t$. Therefore, we should check that $f_t$ is greater than zero before entering the inner loop at step 11. If it is not, we set $\Delta_t$ to zero and go back to step 7 to find another index t.

- If the matrix $\mathbf{K}$ is numerically deficient, adding an index $t$ to the passive set may lead to a higher residual. Inasmuch as the negative gradient component $\Delta_t$ is greater than zero, the fit should decrease in exact arithmetic. However, it may increase slightly due to round-off errors. Since the fit is not guaranteed to decrease, the algorithm may get stuck by moving a set of indices between the passive and active sets back and forth without converging. One idea is to enforce the reduction of the fit by rejecting any step that does not reduce it. In practice, we found that this may lead to sub-optimal solutions. Instead we allow the fit to increase temporarily but prevent getting stuck in an infinite loop by the following trick. Whenever an index reenters the passive set, the fit should be different from the one calculated upon the previous entrance of the same index. This trick guarantees convergence even when the fit increases due to numerical errors.

- The least squares solution at steps 10 and 15 requires a QR decomposition of the matrix composed of the passive set columns. Updating the QR decomposition when adding or removing columns is more efficient than computing the decomposition from scratch [22, Ch.24]. Surprisingly, we found that updating also solves the above problem with numerically deficient matrices and the numerically-computed fit is always decreasing. We do not have an explanation for this ”numerical regularization” effect, but we suspect that it is similar to the regularization that comes implicitly with iterative methods [17, Ch.6].

## 2.8. Non-negative Tikhonov

We can impose the non-negativity constraints on Tikhonov regularization to get what we call *non-negative Tikhonov solution (NNT)*

$$\mathbf{f}_{\text{NNT}} = \underset{\mathbf{f}\in\mathbb{R}^n, \mathbf{f}\geq 0}{\arg\min}\, \chi^2(\mathbf{f}) + \alpha^2 \left\|\mathbf{f}\right\|^2 \ . \tag{2.59}$$

Obtaining this solution does not require any new algorithm. Remembering from Eq. (2.38) that the Tikhonov solution is a least squares solution of a modified problem, we can simply use any NNLS algorithm to solve the modified problem and get the non-negative Tikhnonv solution. The modification is straightforward: the data is padded with zeros and the matrix is padded with an $\alpha$ multiple of unity. Also similar to Tikhonov regularization, the optimal $\alpha$ can be determined using some heuristic like the discrepancy principle or the L-curve. It is worth noting that as $\alpha \to 0$, the NNT solutions

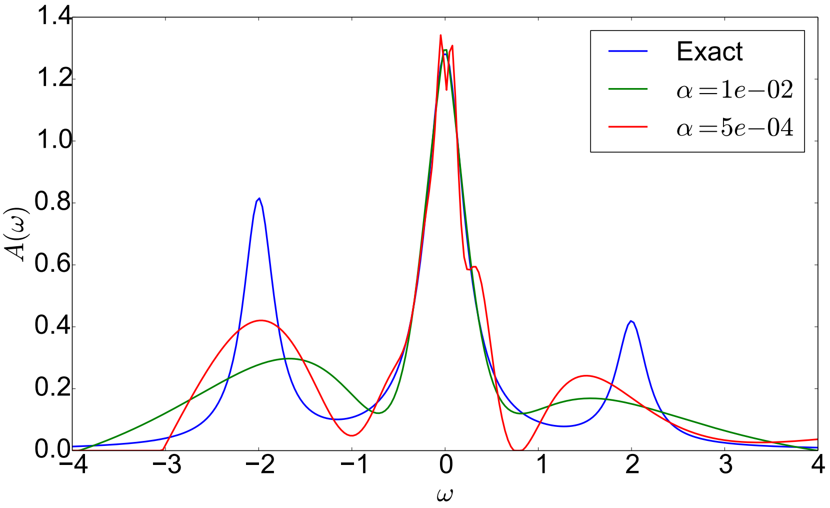


Figure 2.22.: Non-negative Tikhonov solutions of test case 1 for two different values of $\alpha$. The first value is obtained using the discrepancy principle while the second one is obtained from the L-curve.

approaches the NNLS solution. Therefore, the solution norm in the L-curve reaches a moderate value (the norm of NNLS) instead of getting very large as in usual Tikhonov.

In Fig. 2.22 we show the NNT solutions of test case 1 for the two values of $\alpha$ obtained by the discrepancy principle and the L-curve. Comparing these with the earlier Tikhonov solutions (see Fig. 2.12), we see that the differences are minuscule. Basically, the negative parts of Tikhnonv solutions are set to zero. NNT does not give us more information here because the Tikhonov solutions for this test case are mainly positive to start with.

Nevertheless, there are other cases where the non-negativity plays a greater role and NNT clearly outperforms Tikhnonv. For example, we show in Fig. 2.23 non-negative Tikhonov solutions of test case 2. NNT successfully captures the two peaks and locates them at roughly the right positions with widths depending on the regularization parameter $\alpha$. In contrast, usual Tikhonov gives oscillatory solutions around zero that do not show clearly the original two-peaks structure (see Fig. 2.24).

Like we did with NNLS, we can assess the role of non-negativity constraints by comparing the data coefficients $\mathbf{u}_i^{\mathrm{T}}\mathbf{g}$ of the different methods. In Fig. 2.25, we plot the coefficients for test case 1. Although the NNT coefficients matches the Tikhnonv ones initially, the non-negativity constraints prevent the later ones in NNT from being suppressed as much as in Tikhnonov. This effect is even stronger for Test case 2 whose plot is shown in Fig. 2.26. In this case, only robust coefficients (the ones above noise level) are the same for NNT and Tikhnonv while the rest are completely different. We can actually use this difference between usual Tikhnonv and non-negative Tikhnonv solutions as a measure of the importance of non-negativity constraints in any specific case under study. This may help us predict a priori whether stochastic sampling, a computationally-heavy method that relies strongly on non-negativity, would provide a better solution than simple methods before performing the actual calculation.

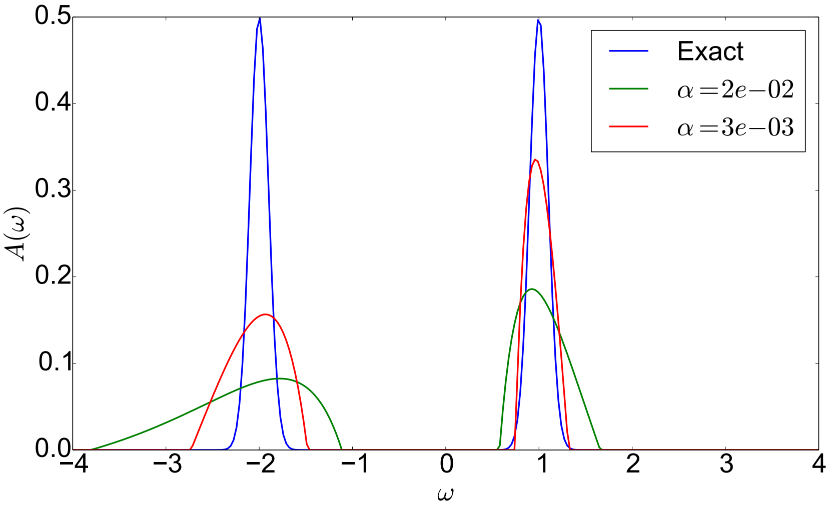


Figure 2.23.: Non-negative Tikhonov solutions of test case 2 for two different values of $\alpha$. The first value is obtained using the discrepancy principle while the second one is obtained from the L-curve.

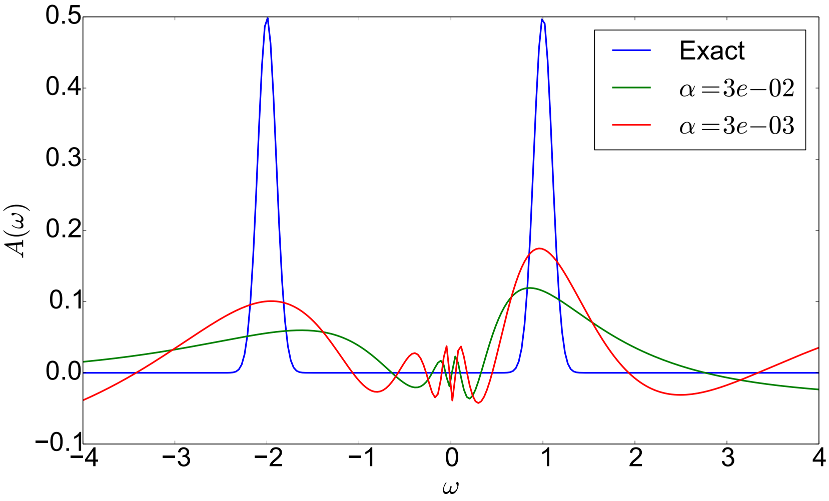


Figure 2.24.: Tikhonov solutions of test case 2 for two different values of $\alpha$. The first value is obtained using the discrepancy principle while the second one is obtained from the L-curve.

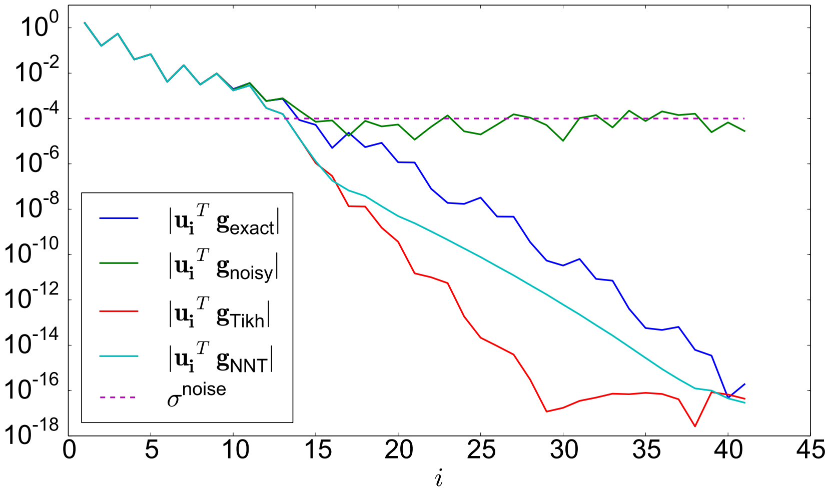


Figure 2.25.: Absolute value of data coefficients of test case 1. The leading coefficients of NNT data match the ones from Tikhnonov, but the non-negativity prevents the later ones from being suppressed as much. For both NNT and Tikhnonv, we used $\alpha = 1\text{e}{-2}$.

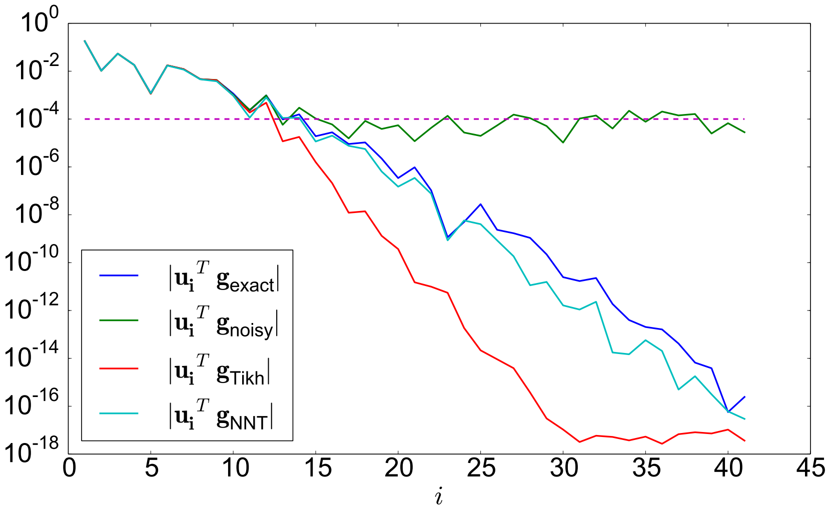


Figure 2.26.: Absolute value of data coefficients of test case 2. Except for the leading robust coefficients, the NNT coefficients are completely different from the Tikhnonov ones. This suggests a greater role of non-negativity constraints in this test case. For both NNT and Tikhnonv, we used $\alpha = 1\text{e}{-2}$.

## 2.9. Perturbed data sampling

We saw that NNLS overfits the data leading to very sharp peaks. This is overcome in NNT by minimizing the norm alongside the data fit. We propose here a new method that avoids overfiting by averaging over different NNLS solutions using different perturbations of the data. The perturbed data samples are obtained by adding extra noise to the originally noisy data. Then we solve the non-negative least squares problem with each perturbed data sample and average their results. Each of these non-negative least squares solutions overfits its corresponding perturbed data sample, but it fits the original data only to the extent of how close the perturbed sample is to the original data. Although, each NNLS solution has very sharp structure, we hope that with an appropriate choice of the perturbing noise, spurious structure would average out and only the "real" structure would survive. We call this method *perturbed data sampling (PDS)* and it can be expressed mathematically as as a weighted integral over perturbing noise

$$\mathbf{f}_{\mathrm{PDS}} = \int d\epsilon \; P(\epsilon) \; \mathbf{f}_{\mathrm{NNLS}} \left( \mathbf{g} + \epsilon \right) \ , \tag{2.60}$$

where $P(\epsilon)$ is the probability of obtaining a noise vector $\epsilon$.

Since the original noise is usually assumed to be Gaussian, uncorrelated and has standard deviation $\sigma$ (see Eq. 2.25), we tried first drawing the perturbing noise from this Gaussian and found that the resulting PDS solution was still overfitting the data. We hoped then that by increasing $\sigma$ of the perturbing noise, we could sample a larger set of perturbed data and get less overfitting. PDS solutions get indeed smoother as $\sigma$ increased, but they also have extra structure that did not exist in the exact model! The reason is that PDS with larger $\sigma$ is averaging over worse set of solutions because it becomes more likely to obtain noise vectors with large norms than ones with smaller norms. This may seem counter-intuitive at first sight, since the Gaussian distribution always gives higher probability for smaller norms. However, one should remember that there are many more vectors with large norms than ones with smaller norms giving an overall larger probability for large norms. More precisely, the probability of obtaining an m-dimensional Gaussian random vector with norm $r$ reads

$$P(\|\epsilon\| = r) \propto r^{m-1} e^{-\frac{r^2}{2\sigma^2}} \ , \tag{2.61}$$

where the factor $r^{m-1}$ comes from the surface area of a hypersphere with radius $r$. This distribution has a maximum at $\sqrt{m-1}\sigma$ (see Fig. 2.27). As $\sigma$ increases, the distribution gets wider but shifted to the right, so most of the perturbing noise in PDS gets larger. Therefore, PDS is systematically averaging over a broader but worse set of solutions and the overall average is bad.

In order to sample over a broad set of solutions while still favoring good ones, we draw the perturbing noise in two stages. First, we draw a normalized random vector $\tilde{\epsilon}$. Then, we draw the norm value $r$ from an exponential distribution with mean $\alpha$ and use it to rescale the noise vector as $\epsilon := r\tilde{\epsilon}$. This leads to more solutions with good fits than bad ones as shown in Fig. 2.28. The mean $\alpha$ is a parameter of the method to be

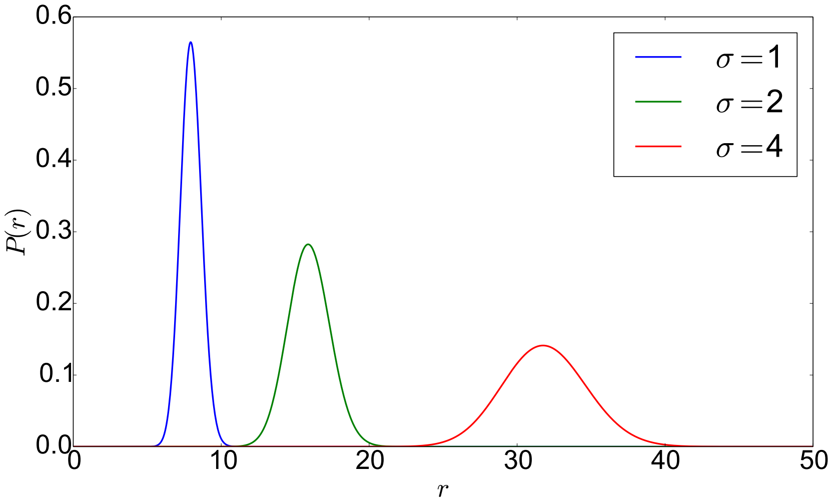


Figure 2.27.: The probability of obtaining a 64-dimensional Gaussian noise vector with norm $r$ for different values of $\sigma$. Increasing $\sigma$ leads to broadening the distribution and shifting it to the right. This means that by increasing $\sigma$, PDS averages over a larger but predominantly worse set of solutions.

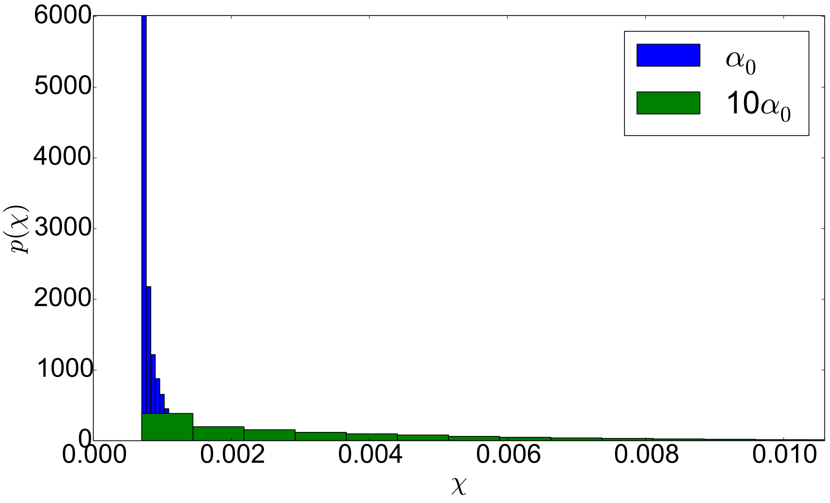


Figure 2.28.: Histogram of the fits of solutions averaged by PDS using perturbing noise whose norm is drawn from an exponential $r \sim \text{Exp}(\alpha)$. Increasing $\alpha$ leads to broadening the distribution which means averaging over a larger set of solutions. This way of perturbing the data always produces more good solutions than bad ones and thus provides a better average than the perturbation with Gaussian noise of fixed $\sigma$.

determined heuristically. Larger $\alpha$ leads to a broader set of averaged solutions and a smoother average.[7]

It is worth noting that perturbed data sampling works *because of the non-negativity constraints*. Suppose that we replace NNLS by just LS. Due to the linearly of the problem, the average model would be the least squares solution of the average of the perturbed data. Since the perturbation has zero mean, this is simply the unperturbed data, and the PDS solution without constraints reduces to the trivial least squares solution as following

$$\int d\epsilon\ P(\epsilon)\ \mathbf{f}_{\mathrm{LS}}\left(\mathbf{g}+\epsilon\right) = \int d\epsilon\ P(\epsilon)\ \left[\mathbf{K}^{+}\mathbf{g}+\mathbf{K}^{+}\epsilon\right] = \left[\mathbf{K}^{+}\mathbf{g}\right] + \mathbf{K}^{+}\cancelto{0}{\int d\epsilon\ P(\epsilon)\ \epsilon} = \mathbf{f}_{\mathrm{LS}}\ ,$$

where LS solutions are expressed in terms of the pseudoinverse $\mathbf{K}^{+}$ (see Eq. 2.23). On the contrary, non-negativity constraints introduce non-linearity into the problem and the NNLS solution of averaged data is not the same as the average of its NNLS solutions.

### 2.9.1. The algorithm

Here is a pseudocode of the PDS algorithm which is basically a repeated non-negative least squares:

```
function PDS($\mathbf{K}, \mathbf{g}, \alpha, N_{\mathrm{samples}}$)
    $\mathbf{f}_{\mathrm{sum}} \leftarrow 0$
    for $1 \le i \le N_{\mathrm{samples}}$ do
        $\mathbf{t} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
        $\tilde{\epsilon} \leftarrow \mathbf{t}/\|\mathbf{t}\|$
        $r \sim \mathrm{Exp}(\alpha)$
        $\epsilon \leftarrow r\tilde{\epsilon}$
        $\mathbf{g}' \leftarrow \mathbf{g} + \epsilon$
        $\mathbf{f}' \leftarrow NNLS(\mathbf{K}, \mathbf{g}')$
        $\mathbf{f}_{\mathrm{sum}} \leftarrow \mathbf{f}_{\mathrm{sum}} + \mathbf{f}'$
    end for
    return $\mathbf{f}_{\mathrm{sum}}/N_{\mathrm{samples}}$
end function
```

The number of samples needed to get a good average depends on the test case and the resolution of the grid $n$. Usually 1000 to 10000 samples are enough to get a decent average and can be increased to smooth out undesired statistical errors. The more important parameter is $\alpha$ which may be determined using the discrepancy principle as in Tikhonov regularization (see Sec. 2.6). The computational cost of the algorithm depends on the cost of NNLS. In practice, we found that the deciding factor is the size of data $m$. Therefore, when $m$ is really large, it may worth first computing the SVD of the full matrix $\mathbf{K} = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathrm{T}}$ and projecting both the data and the matrix on the space spanned by the first $r$ right singular vectors $\mathbf{U}$, where $r$ is the rank of the matrix. This

[7] We used an exponential distribution instead of a Gaussian one for drawing the norm $r$ because the earlier is less concentrated than the later and gives better results.

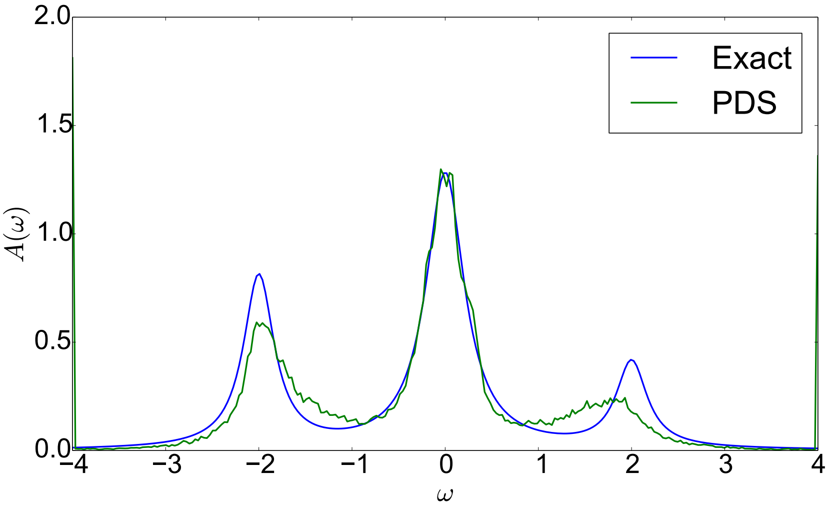


Figure 2.29.: Perturbed data sampling solution of test case 1. The parameter $\alpha = 5\text{e}{-2}$ is determined using the discrepancy principle. The number of samples is 10000 and the running time is 18 seconds.

removes the part of the data lying in the null space of $\mathbf{K}$ and reduces the size of the data to $r$, which is typically much lower than $m$.

### 2.9.2. Results

In Fig. 2.29 and Fig. 2.30, we show PDS solutions of test cases 1 and 2, respectively. For test case 1, the outer peaks are oversmoothed but their shape and decay are clearly better than NNT solution (see Fig. 2.22). For test case 2, the position and width of the left peak are estimated accurately but the right peak is undersmoothed. More importantly, while the peaks in the exact model are Gaussian, the ones in PDS solution have a Lorentzian shape. This suggests that PDS is better at reconstructing Lorentzian peaks which is further confirmed in Fig. 2.31. In this plot, we replace the Gaussian peaks of the exact model by Lorentzian ones and apply the PDS method. Except for the slight broadening of the left peak, PDS is able to reconstruct the two Lorentzian peaks to a surprising accuracy.

A main drawback of this method is its behavior at the grid boundaries. For example, notice the two large values located exactly on the first and last grid points in Fig. 2.29. They represent the weight that leaks outside the grid boundaries. We can understand this behavior intuitively as following. Perturbing the data will move the corresponding structure in NNLS solution around. However, when this structure gets outside the grid, NNLS, due to its overfitting nature, will put the weight of this structure on the closest possible point which is the boundary. In the previous test cases, this effect is easily recognizable and can be safely ignored. Nevertheless, it can be a big problem for analytic continuation of optical conductivity whose grid start at zero. Then it becomes hard to distinguish the boundary effect from the real structure near zero.

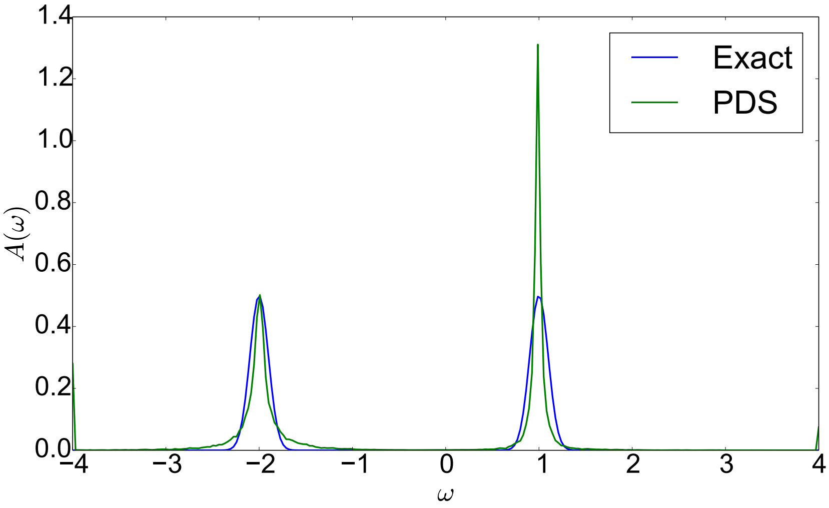


Figure 2.30.: Perturbed data sampling solution of test case 2. The parameter $\alpha = 15\text{e}{-3}$ is determined by the discrepancy principle. The number of smaples is 10000 and the running time is 14 seconds.

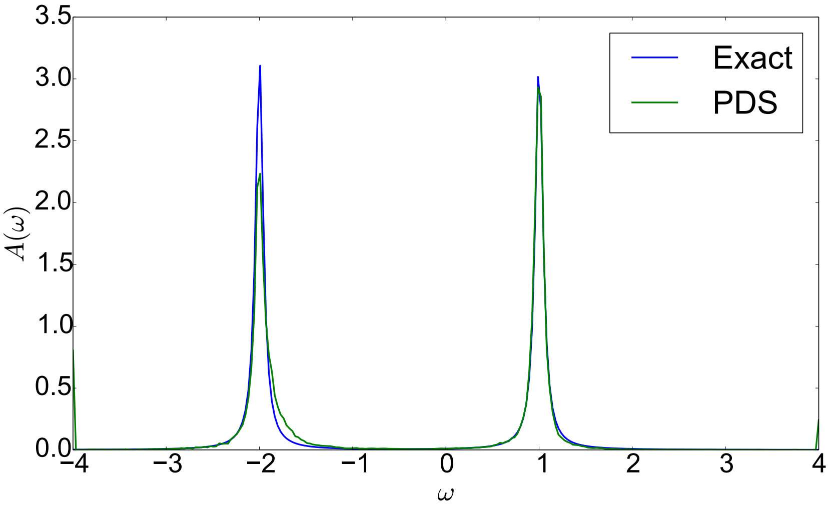


Figure 2.31.: The Gaussian peaks of test case 2 are replaced by Lorentzian peaks; both have half-width 0.05 and weight 0.5. Comparing with Fig. 2.30 shows that PDS is better at retrieving Lorentzian peaks than Gaussian peaks.

# 3. Stochastic Sampling Methods

In this chapter, we apply Bayesian inference to the analytic continuation problem and formulate the stochastic sampling method (StochS) and other methods in Bayesian terms, clarifying the assumptions employed by each method. We present a new efficient algorithm for performing StochS called: *blocked modes sampling* (BMS). In comparison to earlier sampling algorithms, BMS reduces the computational times by orders of magnitude. We then find that StochS results depend on the discretization grid, an effect which has not been discussed before in the literature. We provide the theoretical explanation for the effect, show that StochS has a default model implicitly determined by the grid and give a recipe for choosing a reliable StochS grid.

To make the method more robust, we extend StochS into a gridless method (gStochS) by sampling the grid points from a default model instead of keeping them fixed. The effect of the default model is much reduced in gStochS compared to StochS and depends mainly on its width rather than its shape. The proper width can then be chosen using a simple recipe like we did in StochS.

Finally, to avoid having to fix the width of the default model, we go one step further and extend gStochS to sample over a class of default models with different widths. This extended method (eStochS) is then able to automatically relocate the grid points and concentrate them in the important region. Results show that eStochS can give good results and resolves sharp features in the spectrum without the need for fine tuning a default model.

## 3.1. Introduction to Bayesian inference

Bayesian inference is a statistical method based on Bayes' rule which is used to update the beliefs about a hypothesis in the light of new evidence. Bayesian inference has several advantages. It can be derived from a set of minimal rules for consistent reasoning known as Cox's Axioms (see Ref. [26]). As a statistical method, it does not only allow the estimation of a solution, but also of the uncertainty in that estimate. Additionally, it allows using prior knowledge in the inference about the current situation. As we will see, different methods mainly differ by their prior assumptions which, naturally, affect the results.

Let $H$ be some hypothesis and $E$ some observed evidence, using the definition of conditional probabilities $P(H|E)$ and $P(E|H)$, the joint probability of both $H$ and $E$ reads

$$P(H \cap E) = P(H|E)P(E) = P(E|H)P(H)\ . \tag{3.1}$$

Rearranging the terms gives us readily the *Bayes' rule* for computing the *posterior probability* of the hypothesis $H$ given the evidence $E$

$$P(H|E) = \frac{P(E|H)P(H)}{P(E)} \; , \tag{3.2}$$

where $P(H)$ is the *prior probability* of the hypothesis independent of the evidence, $P(E|H)$ is the probability of the evidence given that the hypothesis is true (known as the *likelihood*) and $P(E)$ is the overall probability of the evidence (known as the *marginal likelihood*).

Assuming a set of disjoint and complete hypotheses $H_i$, the sum of their probabilities add up to one. This holds true not only a priori, but also after observing the evidence

$$\sum_i P(H_i) = 1 \; , \tag{3.3}$$

$$\sum_i P(H_i|E) = 1 \; . \tag{3.4}$$

Substituting the Bayes' rule (Eq. 3.2) in the above relation, the marginal likelihood of the evidence reads

$$P(E) = \sum_i P(E|H_i)P(H_i) \; , \tag{3.5}$$

which can be seen as a normalization of $P(H|E)$ in Eq. 3.2.

## 3.2. Bayesian analytic continuation

Let us apply Bayes' rule to the analytic continuation problem

$$\mathbf{g} = \mathbf{K}\mathbf{f} \; . \tag{3.6}$$

The hypothesis is that some vector $\mathbf{f}$ represents the exact model over some grid intervals, and the evidence is observing the noisy data vector $\mathbf{g}$:

$$P(\mathbf{f}|\mathbf{g}) = \frac{P(\mathbf{g}|\mathbf{f})P(\mathbf{f})}{P(\mathbf{g})} \; . \tag{3.7}$$

The left hand side gives us the posterior probability distribution $P(\mathbf{f}|\mathbf{g})$: the probability of $\mathbf{f}$ being the true model given the data $\mathbf{g}$. Next we explain the terms on the right hand side of the previous relation, then we discuss how to use the posterior probability in estimating the exact model.

### 3.2.1. Likelihood

The likelihood, $P(\mathbf{g}|\mathbf{f})$, is the probability of measuring the data vector $\mathbf{g}$ given that $\mathbf{f}$ is the exact model. If the model is $\mathbf{f}$, then the exact data is $\mathbf{g}^\star = \mathbf{K}\;\mathbf{f}$, and the likelihood would ideally be a delta function at the actual data $P(\mathbf{g}|\mathbf{f}) = \delta(\mathbf{g} - \mathbf{g}^\star) = \delta(\mathbf{g} - \mathbf{K}\mathbf{f})$. However, in reality, the measured data $\mathbf{g}$ differs from the exact one $\mathbf{g}^\star$ due to noise and computational errors. Since QMC results are averages of many data samples, it follows from the central limit theorem that the noise is distributed as a Gaussian. Let the covariance matrix of this Gaussian be $\mathbf{Cov}$, which can be estimated from multiple independent data samples. Then the measured data is distributed as a Gaussian around the exact one[1]

$$P(\mathbf{g}|\mathbf{g}^\star) \propto \exp\left[-\frac{1}{2}(\mathbf{g} - \mathbf{g}^\star)^{\mathrm{T}}\mathbf{Cov}^{-1}(\mathbf{g} - \mathbf{g}^\star)\right] . \tag{3.8}$$

By integrating over hypothetical exact data, we calculate the likelihood of a model given the measured data

$$P(\mathbf{g}|\mathbf{f}) = \int d\mathbf{g}^\star\ P(\mathbf{g}|\mathbf{g}^\star)\ P(\mathbf{g}^\star|\mathbf{f}) = P(\mathbf{g}|\mathbf{g}^\star = \mathbf{K}\mathbf{f}) \propto \exp\left(-\frac{1}{2}\chi^2\right) , \tag{3.9}$$

where $\chi^2 = (\mathbf{g} - \mathbf{K}\ \mathbf{f})^{\mathrm{T}}\ \mathbf{Cov}^{-1}\ (\mathbf{g} - \mathbf{K}\ \mathbf{f})$ is the usual fit of a model $\mathbf{f}$ to the data $\mathbf{g}$.

This is a Gaussian function of the data with mean $\mathbf{g}$ and covaraince matrix $\mathbf{Cov}$. Due to the linearity of the relation between the model and data, the likelihood can also be seen as a Gaussian in the model space whose mean is the least squares solution $\mathbf{K}^\dagger\mathbf{g}$ (see Eq. 2.22) and whose covariance matrix is the inverse of $\mathbf{K}^{\mathrm{T}}\mathbf{Cov}^{-1}\mathbf{K}$. This is done by separating the data into two parts $\mathbf{g} = \mathbf{K}\mathbf{K}^\dagger\mathbf{g} + \mathbf{g}_\perp$ where $\mathbf{K}^\dagger$ is the pseudoinverse of the matrix $\mathbf{K}$ (see Eq. 2.23) and $\mathbf{g}_\perp$ is the projection of the data on the null space of $\mathbf{K}$. Then by completing the squares of the fit as a function of $\mathbf{f}$, we can write it in the following suggestive form

$$\chi^2(\mathbf{f}) = \left(\mathbf{K}^\dagger\mathbf{g} - \mathbf{f}\right)^{\mathrm{T}} \mathbf{K}^{\mathrm{T}}\mathbf{Cov}^{-1}\mathbf{K}\left(\mathbf{K}^\dagger\mathbf{g} - \mathbf{f}\right) + \text{const.} \tag{3.10}$$

The matrix $\mathbf{K}^{\mathrm{T}}\mathbf{Cov}^{-1}\mathbf{K}$ is most likely to be rank-deficient because the kernel matrix $\mathbf{K}$ is. Consequently, the variance of this Gaussian would be infinite in the null space of $\mathbf{K}$, which can be seen as a uniform distribution on the free modes of $\mathbf{K}$.

Remember, as explained in the previous chapter, that by taking the Cholesky decomposition of the inverse covariance matrix $\mathbf{Cov}^{-1}$, the fit can be written in terms of a modified data vector and a modified kernel matrix such that the data values are independent and have unit variance (see Eq. 2.11). Therefore, we will always assume that the covariance matrix is the identity to simplify later manipulations.

### 3.2.2. Prior probability

The prior probability, $P(\mathbf{f})$, represents our prior knowledge and assumptions about the exact model which can be heuristic or exact. An example of heuristic information is

[1] Pay attention not to confuse this with the seemingly-equivalent but completely different assumption that the exact data is distributed as a Gaussian around the measured one i.e. $P(\mathbf{g}^\star|\mathbf{g})$ is a Gaussian. Such an assumption has no justification whatsoever and leads to wrong conclusions because it neglects the prior distribution of the exact model and thus that of the exact data.

expecting the model to be smooth. Using the $L_2$-norm as a measure of smoothness, we can express this by assigning higher prior probabilities to models with lower norms. This leads naturally to the prior used by the Tikhonov method

$$P_{\mathrm{Tikh}}(\mathbf{f}) \propto \exp\left[-\frac{1}{2}\alpha^2\|\mathbf{f}\|^2\right] , \tag{3.11}$$

which is a Gaussian centered around zero and parameterized by its variance $1/\alpha^2$. More generally, the Tikhonov method can use other bilinear functions of the model (see Sec. 2.6) and still have a Gaussian prior.

Another heuristic is the resemblance to another model $\mathbf{m}$ called the default model. The resemblance between two non-normalized positive models is measured by their relative entropy

$$S(\mathbf{f}) = \sum_i f_i - \sum_i m_i - \sum_i f_i \ln\left(\frac{f_i}{m_i}\right) . \tag{3.12}$$

The maximum entropy method (MaxEnt) uses an exponential decaying function of the model's entropy as a prior

$$P_{\mathrm{MaxEnt}}(\mathbf{f}) \propto \exp\left[\alpha S(\mathbf{f})\right] , \tag{3.13}$$

where the parameter $\alpha$ controls the strength of the prior. Notice how the entropy implicitly enforces the positivity of the model due to the presence of the logarithm function, which is undefined for negative values. Also, the entropy cannot be expressed as a bilinear function of the model, so MaxEnt cannot be reduced to a Tikhonov method.

Exact prior information includes the non-negativity of the model and the sum rules it may satisfy. These linear constraints restrict the admissible models to a convex set $\mathcal{F}$, i.e. every linear combination of admissible models whose coefficients are positive and add up to one, is also an admissible model. In the absence of any other information, the simplest and most intuitive prior is to assign the same probability to all admissible models and zero probability to models outside the allowed region. For example, the prior used by the non-negative least squares method (NNLS) reads

$$P_{\mathrm{NNLS}}(\mathbf{f}) \propto \begin{cases} 1 \text{ for } \mathbf{f} \geq 0 \\ 0 \text{ otherwise} \end{cases} . \tag{3.14}$$

### 3.2.3. Marginal likelihood

The marginal likelihood, $P(\mathbf{g})$, is the probability of observing the data regardless of what the actual model is. Using Eq. 3.5, it can be computed as the integral of the likelihood over all possible models weighted by their prior probabilities

$$P(\mathbf{g}) = \int d\mathbf{f} P(\mathbf{f}) P(\mathbf{g}|\mathbf{f}) . \tag{3.15}$$

Since it is independent of the model, it is just a normalization constant that will not affect the estimation of $\mathbf{f}$, and thus we need not care about it.

### 3.2.4. Estimator

After fixing the prior and the likelihood, we get a posterior probability $P(\mathbf{f}|\mathbf{g})$ (up to a normalization constant) for each model $\mathbf{f}$. Now we want to choose from all the infinitely-many possible models just one model $\mathbf{f}^\star$ as our estimate. But which one?

Bayesian decision theory answers this question by asking another one: what is the cost of being wrong? Given the cost (called *Loss function*), it makes sense to choose the model with the minimum expected cost.

Formally, let the loss function $L(\mathbf{f}',\mathbf{f})$, represent the price we pay when using $\mathbf{f}'$ as an estimate when $\mathbf{f}$ is really the exact model. The best estimation $\mathbf{f}^\star$ is then the one that minimizes the expected value of the loss function:

$$\mathbf{f}^\star = \arg\min_{\mathbf{f}} \int d\mathbf{f}'\ P(\mathbf{f}'|\mathbf{g})\ L(\mathbf{f}',\mathbf{f}) \tag{3.16}$$

A commonly-used loss function is the 0-1 loss function when $\mathbf{f}$ has discrete values

$$L(\mathbf{f}',\mathbf{f}) = \begin{cases} 0 \text{ when } \mathbf{f}' = \mathbf{f} \\ 1 \text{ when } \mathbf{f}' \neq \mathbf{f} \end{cases} , \tag{3.17}$$

or the functional Dirac when it has continuous values

$$L(\mathbf{f}',\mathbf{f}) = 1 - \delta(\mathbf{f}' - \mathbf{f}) , \tag{3.18}$$

which both lead to the maximum of the posterior probability as an estimator

$$\mathbf{f}^\star = \arg\min_{\mathbf{f}} \int d\mathbf{f}' P(\mathbf{f}'|\mathbf{g}) - P(\mathbf{f}|\mathbf{g}) = \arg\min_{\mathbf{f}} [1 - P(\mathbf{f}|\mathbf{g})] \tag{3.19}$$

$$\Rightarrow \mathbf{f}^\star = \arg\max_{\mathbf{f}} P(\mathbf{f}|\mathbf{g}) . \tag{3.20}$$

Another reasonable loss function is the $L_2$-norm squared of the difference between a candidate model and the exact one

$$L(\mathbf{f}',\mathbf{f}) = \|\mathbf{f}' - \mathbf{f}\|_2^2 , \tag{3.21}$$

which leads to the mean of the posterior probability as an estimator

$$\mathbf{f}^\star = \arg\min_{\mathbf{f}} \int d\mathbf{f}'\ P(\mathbf{f}'|\mathbf{g})\ \|\mathbf{f}' - \mathbf{f}\|_2^2 \Rightarrow \int d\mathbf{f}'\ P(\mathbf{f}'|\mathbf{g})\ 2(\mathbf{f}' - \mathbf{f}) = 0 \tag{3.22}$$

$$\Rightarrow \mathbf{f}^\star = \int d\mathbf{f}\ \mathbf{f}\ P(\mathbf{f}|\mathbf{g}). \tag{3.23}$$

The maximum estimator is used by many regularization methods including: Tikhonov, MaxEnt, and NNLS which are consequently formulated as optimization problems with different objective functions:

$$\mathbf{f}_{\text{Tikh}} = \arg\min_{\mathbf{f}} \quad \chi^2(\mathbf{f}) + \alpha^2 \|\mathbf{f}\|^2 \tag{3.24}$$

$$\mathbf{f}_{\text{MaxEnt}} = \arg\max_{\mathbf{f}} \ -\frac{1}{2}\ \chi^2(\mathbf{f}) + \alpha\ S(\mathbf{f}) \tag{3.25}$$

$$\mathbf{f}_{\text{NNLS}} = \arg\min_{\mathbf{f}\geq 0} \quad \chi^2(\mathbf{f}) \tag{3.26}$$

The use of the maximum in MaxEnt is justified by the argument that the maximum is a good representative of the resulting posterior which is unimodal (i.e. it has only one local maximum) and concentrated around this maximum. Note that for the Tikhnonv method, the use of the maximum would give the same result as the mean because the posterior is a Gaussian.

However, the maximum estimator does not seem to work very well for non-negative least squares. The results of this method are generally models with very sharp peaks over-fitting the data (see Fig. 2.16 and Fig. 2.18). This failure of NNLS may be attributed at first to the "non-informative" flat prior. But a closer examination shows that it is also the result of a poor choice of the estimator. Since the prior is flat over all non-negative models and the likelihood is a Gaussian, the posterior is a Gaussian truncated to the non-negative region (Fig. 2.15 shows the contours of such a posterior) and its maximum usually lies on the boundary of this region. Clearly, such an estimate does not reflect the posterior distribution well enough. Using the mean, on the other hand, would take into account every possible model weighted by its posterior probability. In other words, NNLS does not utilize the information about the noise that is encoded in the Gaussian likelihood. As we will see in the next section, stochastic sampling uses this information and gets a huge improvement in the quality of the estimation.

## 3.3. Stochastic sampling (StochS)

Replacing the maximum estimator of NNLS with the mean estimator gives us the *stochastic sampling method* (StochS)

$$\mathbf{f}_{\mathrm{StochS}} = \frac{1}{C} \int_{\mathcal{F}} d\mathbf{f}\ \mathbf{f}\ \exp\left[-\frac{1}{2}\chi^2(\mathbf{f})\right] \ , \tag{3.27}$$

where $C$ is a normalization constant. Intuitively, this method averages over all allowed models weighted by how well they fit the data. The weight factor is a Gaussian given by the noise on the data. We expect that this averaging will lead to smoothing the details not supported by the data. The larger the noise, the larger the smoothing. Note that the average is guaranteed to be an allowed model because the set of allowed models $\mathcal{F}$ is convex.

From a Bayesian point of view, StochS has a flat prior over non-negative models and a Gaussian likelihood just like NNLS, but it has a mean estimator. Computing this mean is discussed in the next section, but first let us consider the following test case and see how StochS performs in comparison to NNLS.

**Test case 3** This test case is taken from Ref. [2]. It is about the analytic continuation of the optical conductivity $\sigma(\omega)$ using the current-current correlation function $\Pi(\nu)$

$$\Pi(\nu) = \frac{2}{\pi} \int_0^{+\infty} d\omega\ \frac{\omega^2}{\nu^2 + \omega^2}\ \sigma(\omega) \ . \tag{3.28}$$

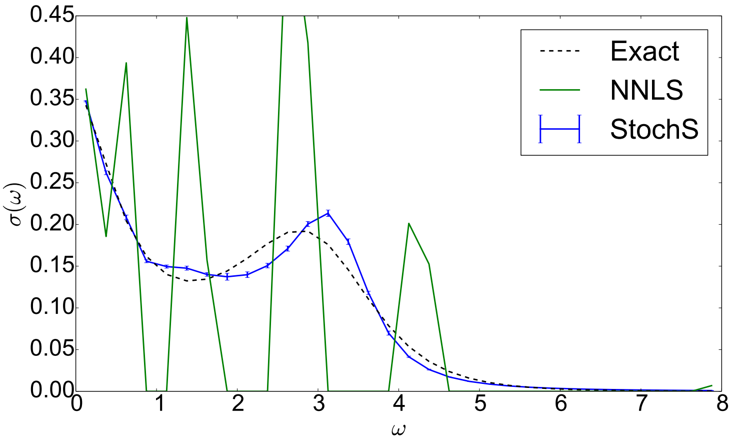


Figure 3.1.: StochS solution vs. NNLS solution for test case 3. Both methods use the same prior (flat over non-negative models) and likelihood (Gaussian) but different estimators. Using the mean estimator in StochS leads to huge improvement in the results in comparison to the maximum in NNLS. The exact model is shown in dashed black.

Ref. [2] uses the following optical conductivity model for its tests

$$\sigma(\omega) = \left\{ \frac{W_1}{1+(\omega/\Gamma_1)^2} + \frac{W_2}{1+\left[(\omega-\epsilon)/\Gamma_2\right]^2} + \frac{W_2}{1+\left[(\omega+\epsilon)/\Gamma_2\right]^2} \right\} \frac{1}{1+(\omega/\Gamma_3)^6} \,, \tag{3.29}$$

with two different sets of parameters. Here we use one of them: $\Gamma_1 = 0.6, \Gamma_2 = 1.2, \Gamma_3 = 4, \epsilon = 3, W_1 = 0.3, W_2 = 0.2$. The data values $\Pi(\nu_j)$ are computed analytically for the 60 smallest Matsubara frequencies $\nu_j = j\ 2\pi T$ where temperature is set to $T = 1/15$. Relative normally-distributed noise with standard deviation $10^{-3}$ is then added to the data and the model $\sigma(\omega)$ is reconstructed using the noisy data. One thing that is not discussed in Ref. [2] is how the data integrals are discretized. As a starting point, we take a uniform grid of $\omega$ in the range $[0, 8]$ with 32 points and use the rectangle rule. The discretization and cutoff errors using this grid are below the noise level (check Sec. 2.1 for a comment on the convergence of the rectangle rule). In Fig. 3.1, we show the results using the NNLS and StochS methods. As expected the NNLS solution is composed of a few sharp peaks roughly located where the bulk of the model is. In contrast, the StochS solution is a smooth function resolving the main features of the exact model. Notice that the StochS prior makes no assumptions about the smoothness of the model and the resulting smoothness comes from averaging only. For example, Fig. 3.2 shows some of the high-probability models in StochS. These models fit the data well enough but lack any smoothness, and therefore StochS has the potential of resolving sharp features when they are supported by the data.

*Note on Error Bars:* The error bars shown in stochastic sampling plots represent the statistical error in computing the average. They are estimated from independent runs of the sampling algorithm as the standard deviation of the average. This should not

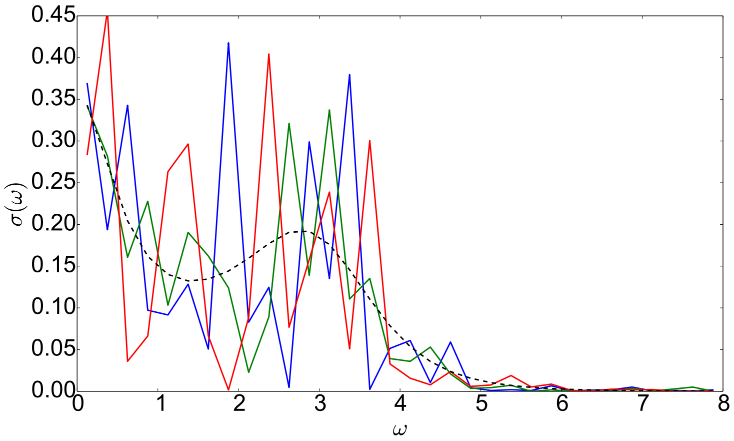


Figure 3.2.: Three different models that have high posterior probability in StochS. Since the prior is flat, these models also have a high likelihood and thus they fit the data well. StochS smooth solution is mainly the average of such sharp models.

be confused with the standard deviation of the posterior probability distribution. The earlier goes down to zero as the number of samples increases, while the latter is an intrinsic property of the distribution and not related to the averaging itself.

### 3.3.1. Sampling algorithm

Computing the StochS solution requires evaluating a multidimensional integral. Had we not had any constraints, the set of allowed models would have been the entire space. In this case, the posterior distribution of the stochastic sampling would be improper because the kernel matrix $\mathbf{K}$ is rank-deficient which leads to an infinite normalization factor. The problem comes from the null space of the matrix $\mathbf{K}$, in which the Gaussian becomes uniform (i.e. infinite variance), and thus it is unnormalizable in that infinite subspace. Nevertheless, if we exclude the null space, the posterior is a normalizable Gaussian centered around $\mathbf{K}^{\dagger}\mathbf{g}$ (see Eq. 3.10) and the stochastic sampling solution can be well-defined. Since the mean of a Gaussian matches its maximum, the StochS solution in the absence of the non-negativity constraints equals the least squares ones.

The non-negativity constraints that make computing the mean more laborious, improve the quality of estimation considerably. Due to truncation to the allowed region $\mathcal{F}$, the desired mean is not $\mathbf{K}^{\dagger}\mathbf{g}$ anymore, but has to be computed numerically by sampling the models using, e.g., the Monte Carlo method and then averaging the samples.

Refs. [4, 27, 3] use the Metropolis algorithm for the sampling. It starts from an initial admissible model and suggests constraint-satisfying random changes on its components to obtain a candidate sample. If the candidate sample has a higher probability than the old one, it is accepted directly; otherwise it is accepted according to the ratio of the probability of the candidate sample to the probability of the old sample. When a candidate is rejected, the old sample is kept. This is repeated until a sufficient number of samples is generated. Refs. [4, 27, 3] do not report quantitative information about the running time of this method. However, they describe the generated samples as highly-correlated. Actually, the correlation time is so high that they use simulated annealing procedures to sample efficiently and avoid being stuck around a specific model of high probability.

We propose a new sampling algorithm, *blocked modes sampling (BMS)*, that has small correlation time and can cover the admissible space of models without the need of simulated annealing. BMS takes advantage of two properties of the truncated multivariate Gaussian distribution. First, its conditional distributions are truncated univariate Gaussians, for which fast sampling algorithm exists. Second, any linear transformation of the variables is itself distributed as a truncated multivariate Gaussian. The first property allows us to compute conditional probabilities and use Gibbs sampling, a special case of the Metropolis-Hastings algorithm to be explained below, which has an acceptance probability of one. The second property allows us to do variable transformations such that the steps taken by the sampling are large. Taking large steps with an acceptance ratio of one makes for an efficient Monte Carlo sampling!

In the following, we explain BMS by developing it gradually. We start from sampling the model's components directly, which is easy to implement but does not account for the correlation between the components. Then, using modes sampling, we go to the other extreme, where correlations are fully accounted for. However, the inequality constraints couple those modes together reducing the overall efficiency. Finally, we describe blocked modes sampling as a solution balancing the correlation of the components and the coupling of the modes, leading to a much higher efficiency.

**Note** To keep things simple, we will assume in the following discussion that we only have non-negativity constraints. Imposing sum rules will be explained in Sec.3.3.1.

### Components sampling

Gibbs sampling is useful for sampling a joint probability distribution $p(\mathbf{f})$ when its conditional distributions are known and easy to sample. It starts from some initial sample $\mathbf{f}^{(0)}$. Then a new sample $\mathbf{f}^{(t+1)}$ is generated from the current one $\mathbf{f}^{(t)}$ by sampling each component conditional on the values of all other components. It is easy to see that Gibbs sampling is a special case of Metropolis-Hasting algorithm. In Metropolis-Hasting, if we have a sample $\mathbf{f}^{(t)}$, we propose a new sample $\mathbf{f}'$ with probability $q(\mathbf{f}^{(t)} \to \mathbf{f}')$ and accept it with probability

$$r = \min \left[ 1, \frac{p(\mathbf{f}')}{p(\mathbf{f}^{(t)})} \frac{q(\mathbf{f}' \to \mathbf{f}^{(t)})}{q(\mathbf{f}^{(t)} \to \mathbf{f}')} \right] . \tag{3.30}$$

In Gibbs sampling, the current and proposed samples differ in the value of component $f_i$ which is updated according to its conditional probability. Therefore, the proposal probability $q(\mathbf{f}^{(t)} \to \mathbf{f}')$ equals $p(f_i' | \mathbf{f}^{(t)}_{(-i)})$, where $(-i)$ in the subscript of a vector indicates that component $i$ is removed. The acceptance probability then reads

$$r = \frac{p(\mathbf{f}')}{p(\mathbf{f}^{(t)})} \frac{q(\mathbf{f}' \to \mathbf{f}^{(t)})}{q(\mathbf{f}^{(t)} \to \mathbf{f}')} = \frac{p(f_i'|\mathbf{f}_{(-i)}^{(t)})\, p(\mathbf{f}_{(-i)}^{(t)})}{p(f_i^{(t)}|\mathbf{f}_{(-i)}^{(t)})\, p(\mathbf{f}_{(-i)}^{(t)})} \frac{p(f_i^{(t)}|\mathbf{f}_{(-i)}^{(t)})}{p(f_i'|\mathbf{f}_{(-i)}^{(t)})} = 1 \; . \tag{3.31}$$

Therefore, each proposed step in Gibbs sampling is accepted.

To apply Gibbs sampling to our case, we need to find the conditional distributions $p(f_i|\mathbf{f}_{(-i)})$. We rewrite the exponent $\chi^2$ from Eq. (3.10) as a function of $f_i$ alone, while considering $\mathbf{f}_{(-i)}$ as parameters

$$\begin{aligned}
\chi^2(f_i; \mathbf{f}_{(-i)}) &= (\mathbf{g} - \mathbf{K}\ \mathbf{f})^{\mathrm{T}} (\mathbf{g} - \mathbf{K}\ \mathbf{f}) && (3.32)\\
&= \left[\mathbf{g} - \begin{pmatrix}\mathbf{K}_i & \mathbf{K}_{(-i)}\end{pmatrix} \begin{pmatrix} f_i \\ \mathbf{f}_{(-i)} \end{pmatrix}\right]^{\mathrm{T}} \left[\mathbf{g} - \begin{pmatrix}\mathbf{K}_i & \mathbf{K}_{(-i)}\end{pmatrix} \begin{pmatrix} f_i \\ \mathbf{f}_{(-i)} \end{pmatrix}\right] && (3.33)\\
&= \left(\mathbf{g} - \mathbf{K}_i f_i - \mathbf{K}_{(-i)}\ \mathbf{f}_{(-i)}\right)^{\mathrm{T}} \left(\mathbf{g} - \mathbf{K}_i f_i - \mathbf{K}_{(-i)}\ \mathbf{f}_{(-i)}\right) \; , && (3.34)
\end{aligned}$$

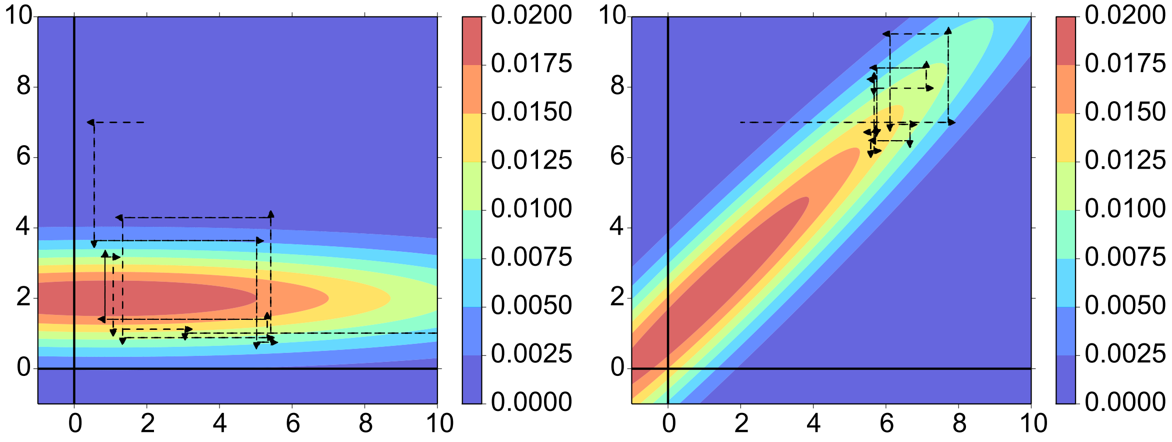
where $\mathbf{K}_i$ is the i-th column of $\mathbf{K}$, and $\mathbf{K}_{(-i)}$ is $\mathbf{K}$ with the i-th column dropped. Denoting $\mathbf{g}_{(-i)} \coloneqq \mathbf{g} - \mathbf{K}_{(-i)}\ \mathbf{f}_{(-i)}$, which is the data vector after subtracting the contribution of the conditioned variables, we have

$$\begin{aligned}
\chi^2(f_i; \mathbf{f}_{(-i)}) &= \left(\mathbf{g}_{(-i)} - \mathbf{K}_i f_i\right)^{\mathrm{T}} \left(\mathbf{g}_{(-\mathbf{i})} - \mathbf{K}_i f_i\right) && (3.35)\\
&= \mathbf{K}_i^{\mathrm{T}} \mathbf{K}_i f_i^2 - 2\ \mathbf{g}_{(-i)}^{\mathrm{T}} \mathbf{K}_i f_i + \mathbf{g}_{(-i)}^{\mathrm{T}} \mathbf{g}_{(-i)} && (3.36)\\
&= \frac{\left(f_i - \mathbf{g}_{(-\mathbf{i})}{}^{\mathrm{T}} \mathbf{K}_i / \mathbf{K}_i^{\mathrm{T}} \mathbf{K}_i\right)^2}{(\mathbf{K}_i^{\mathrm{T}} \mathbf{K}_i)^{-1}} + \text{const.} && (3.37)
\end{aligned}$$

where the last relation comes from completing the square. From this relation, we see that in the absence of constraints, the conditional $p(f_i|\mathbf{f}_{(-i)})$ is a univariate Gaussian with mean $\mu = \mathbf{K}_i^{\mathrm{T}} \mathbf{g}_{(-i)} / \mathbf{K}_i^{\mathrm{T}} \mathbf{K}_i$ and variance $\sigma^2 = (\mathbf{K}_i^{\mathrm{T}} \mathbf{K}_i)^{-1}$. Non-negativity constraints truncate this Gaussian to the interval $[0, \infty[$.

Let us put things together. We start from some initial non-negative model. We pick one component of the model and update its value while keeping the values of all other components fixed. The update is done according to a Gaussain distribution truncated to the positive region. The parameters of this distribution are computed by subtracting the contribution of the conditional components from the data and considering the kernel matrix column corresponding to the updated component. The mean is the projection of the residual data on the column matrix, and the variance is the squared inverse of the column matrix norm. Sampling a truncated univariate Gaussian can be done efficiently using, e.g., the algorithm provided by Ref. [28]. After drawing the value of this component, we have our first sample. Then we pick another component and update its value in the same way to get another sample and so on.

Picking components for updating can be done randomly according to whatever distribution we wish as long as all components have non-zero probabilities; otherwise the sampling would not cover the whole space. Whatever the picking distribution is, the sampling is still ergodic, because such a probability can be considered as part of the



(a) Case of Diagonal matrix **K**

(b) Case of non-diagonal matrix **K**

Figure 3.3.: *Components sampling* for the two dimensional case. The ellipses represent the contours of the sampled bivariate Guassian distribution. The arrows represent steps taken by component sampling. Notice that when the matrix **K** is diagonal, component sampling goes directly to the region of high probability. On the other hand, when the matrix in non-diagonal, sampling is less efficient.

proposal probability $q(\mathbf{f}^{(t)} \rightarrow \mathbf{f}')$. The simplest choice is a uniform distribution. Another option is to go through all the components one after another, and a new sample is considered after updating all the components. The latter way of sampling can no longer be considered as a special case of Metropolis-Hasting algorithm, but it can still be shown to satisfy the detailed balance condition [29]; actually the latter way of the sampling is what is typically referred to as Gibbs sampling.

The convergence of Gibbs sampling is proved in Ref. [30] under mild conditions. One condition that deserves our attention is the lower semi-continuity of the sampled distribution at zero. This means that models on the boundary of the admissible region should be assigned zero probability and only strictly positive models are admissible (i.e. the constraints should be $\mathbf{f} > 0$ rather than $\mathbf{f} \geq 0$). The exclusion of the boundary, however, does not affect the calculation of the stochastic sampling solution because the boundary has zero measure in the integral of Eq. (3.27).

Regarding computational complexity, computing $\mathbf{g}_{(-i)} \coloneqq \mathbf{g} - \mathbf{K}_{(-i)}\,\mathbf{f}_{(-i)}$ takes $O(m \times n)$ operations, and computing $\mu$ and $\sigma$ takes $O(m)$ operations. Since sampling a truncated univariate Gaussian is independent of the problem size and typically $m < n$, the total cost of sampling one component using Gibbs sampling is $O(n)$. Going through all components takes $O(n^2)$.

The efficiency of components sampling depends on the correlation between the different components which is determined by the matrix **K**. A diagonal matrix implies no correlation and Gibbs sampling becomes equivalent to direct sampling. On the other hand, large non-diagonal elements make Gibbs sampling extremely inefficient (see Fig. 3.3).

### Modes sampling

As we saw in the last section, components sampling is efficient when the matrix $\mathbf{K}$ is diagonal because it implies no correlation between the components. But what do we do when there are correlations? We go to a basis where there isn't such a correlation! This is done by rewriting the fit $\chi^2$ in terms of the singular value decomposition of the matrix $\mathbf{K} = \mathbf{USV}^{\mathrm{T}}$

$$\begin{aligned}\chi^2 &= (\mathbf{g} - \mathbf{K}\ \mathbf{f})^{\mathrm{T}} (\mathbf{g} - \mathbf{K}\ \mathbf{f}) && (3.38)\\ &= \left(\mathbf{g} - \mathbf{USV}^{\mathrm{T}}\ \mathbf{f}\right)^{\mathrm{T}} \left(\mathbf{g} - \mathbf{USV}^{\mathrm{T}}\ \mathbf{f}\right) && (3.39)\\ &= \left(\mathbf{U}^{\mathrm{T}}\mathbf{g} - \mathbf{SV}^{\mathrm{T}}\ \mathbf{f}\right)^{\mathrm{T}} \left(\mathbf{U}^{\mathrm{T}}\mathbf{g} - \mathbf{SV}^{\mathrm{T}}\ \mathbf{f}\right)\ , && (3.40)\end{aligned}$$

where $\mathbf{U}^{\mathrm{T}}\mathbf{U} = \mathbf{I}$ is used (see Sec. 2.2). Denoting the projection of the model on the modes as $\mathbf{e} := \mathbf{V}^{\mathrm{T}}\mathbf{f}$ and the projection of the modified data on the left singular vectors as $\mathbf{h} := \mathbf{U}^{\mathrm{T}}\mathbf{g}$, we have

$$\chi^2(\mathbf{e}) = (\mathbf{h} - \mathbf{Se})^{\mathrm{T}}(\mathbf{h} - \mathbf{Se})\ . \tag{3.41}$$

Since the projection coefficients $\mathbf{e}$ are related to the model $\mathbf{f}$ by a an orthogonal transformation, then $d\mathbf{f} = d\mathbf{e}$. As a result, we can directly change the integration variable in Eq. (3.27) from $\mathbf{f}$ to $\mathbf{e}$, and StochS solution reads

$$\mathbf{f}_{\mathrm{StochS}} = \frac{1}{C}\int_{\mathcal{F}} d\mathbf{e}\ \mathbf{Ve}\ \exp\left[-\frac{1}{2}\chi^2(\mathbf{e})\right] = \frac{\mathbf{V}}{C}\int_{\mathcal{F}} d\mathbf{e}\ \mathbf{e}\ \exp\left[-\frac{1}{2}\chi^2(\mathbf{e})\right]\ , \tag{3.42}$$

where the last equality holds due to the linearity of integration. The advantage of this transformation is that the multidimensional exponential is now factorized into $r$ one-dimensional exponentials, where $r$ is the number of non-zero singular values. This can be made clearer by rewriting the exponent as

$$\chi^2(\mathbf{e}) = \sum_{i=1}^{m}(h_i - s_i e_i)^2 = \sum_{i=1}^{r} s_i^2(h_i/s_i -\ e_i)^2 + \underbrace{\sum_{i=r}^{m} h_i^2}_{\text{Residual}}\ . \tag{3.43}$$

Then Eq. (3.42) reads

$$\mathbf{f}_{\mathrm{StochS}} = \frac{\mathbf{V}}{\tilde{C}}\int_{\mathcal{F}} de_1\ ...\ de_n\ \exp\left[-\frac{(e_1 - h_1/s_1)^2}{2s_1^{-2}}\right]\ ...\ \exp\left[-\frac{(e_r - h_r/s_r)^2}{2s_r^{-2}}\right]\ \mathbf{e}\ , \tag{3.44}$$

where $\exp(-0.5 \times \text{Residual})$ is absorbed in the normalization constant $\tilde{C}$. This equation (Eq. 3.44) and the original one (Eq. 3.27) express the same multidimensional integral in two different bases (see Fig. 3.4), which are related by the orthonormal matrix $\mathbf{V}$, whose columns $\mathbf{v_i}$ represent the modes.

Apart from truncation to the allowed region $\mathcal{F}$, it is obvious that the first $r$ projection coefficients $e_i$ are distributed as independent Gaussians, each of which has mean $h_i/s_i$

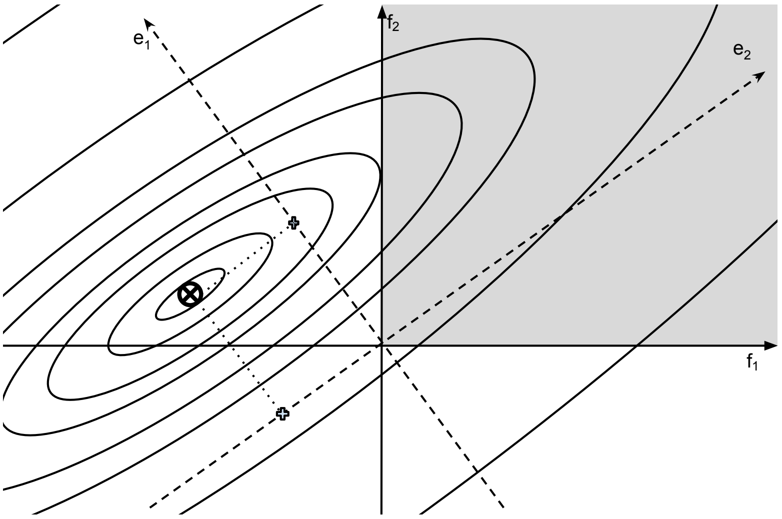


Figure 3.4.: 2D Illustration of the difference between components sampling and modes sampling. The ellipses represent the contours of the probability distribution in the space of models. Eq. (3.27) expresses this distribution in terms of $f_1, f_2$ while Eq. (3.44) expresses it in terms of $e_1, e_2$. The shaded region represents the region of models' space that should be sampled. Although this region is the same for both cases, it is more complicated to express in terms of $e_1, e_2$, while it is simply half open intervals in terms of $f_1, f_2$.

and variance $s_i^{-2}$. Restriction to the region $\mathcal{F}$ leads to a truncation to an interval $]a_i, b_i[$ (It is guaranteed to be one connected interval because the region $\mathcal{F}$ is convex). The truncation limits $a_i$ and $b_i$ of one coefficient depend on the values of all other coefficients. To determine these limits, let $e_i$ be the current value of the coefficient and $e'_i$ be the new one, then the new components of the model should satisfy the non-negativity constraints

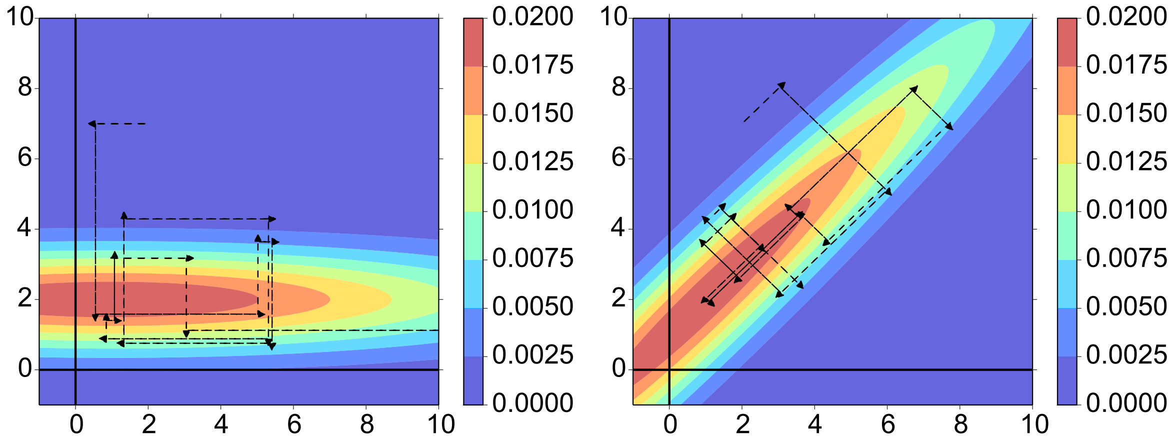
$$\mathbf{f}' > 0 \Rightarrow \mathbf{f} + (e'_i - e_i)\, \mathbf{V}_i > 0 \Rightarrow \forall k \in \{1, ..., n\} :\; f_i + (e'_i - e_i) V_{k,i} > 0 \Rightarrow \tag{3.45a}$$

$$\Rightarrow \begin{cases} e'_i > -f_i/V_{k,i} + e_i \;:\; V_{k,i} > 0 \\ e'_i < -f_i/V_{k,i} + e_i \;:\; V_{k,i} < 0 \end{cases} \tag{3.45b}$$

$$\Rightarrow \begin{cases} a_i = \max \quad \{-f_i/V_{k,i} + e_i : V_{k,i} > 0\} \cup \{-\infty\} \\ b_i = \min \quad \{-f_i/V_{k,i} + e_i : V_{k,i} < 0\} \cup \{+\infty\} \end{cases} . \tag{3.45c}$$

Coefficients corresponding to $s_i = 0$ can be thought of as Gaussians with infinite variances. Therefore, they are distributed uniformly in the interval $]a_i, b_i[$.

Modes sampling is using Gibbs sampling on the coefficients $e_i$ instead of the components $f_i$. The advantage of using the former rather than the latter is that the coefficients $e_i$ are uncorrelated. Comparing Fig. 3.3b to Fig. 3.5b, we see that, in contrast to components sampling, modes sampling is efficient even when the matrix $\mathbf{K}$ is non-diagonal. The disadvantage, however, is that the constraints are harder to express in terms of $e_i$ and they may lead to strong coupling between the modes.



(a) Case of Diagonal matrix $\mathbf{K}$

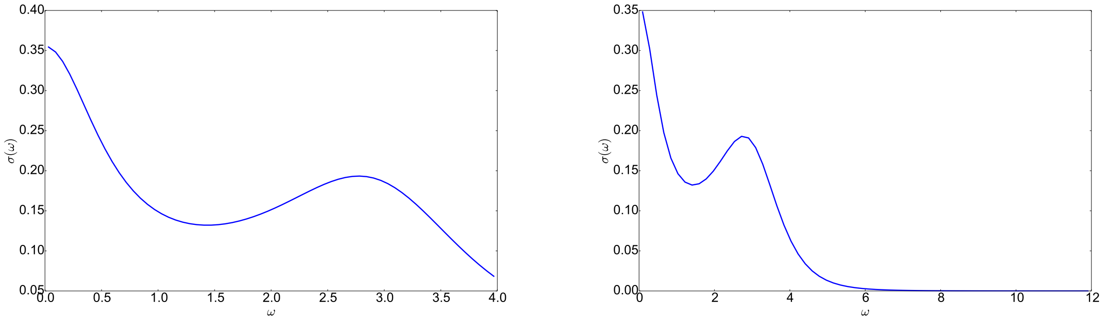
(b) Case of non-diagonal matrix $\mathbf{K}$

Figure 3.5.: *Modes sampling* for the two dimensional case. The ellipses represent the contours of the sampled bivariate Guassian distribution. The arrows represent steps taken by modes sampling. Notice that unlike components sampling (Fig. 3.3), modes sampling is efficient regardless of whether the matrix $\mathbf{K}$ is diagonal or not.

Regarding computational complexity, computing the singular value decomposition takes $O(n^3)$, assuming $m$ is of the same order of $n$ or less. But this cost is paid only once in the initialization phase, and the cost per sample is the important factor. The cost of updating one mode's coefficient $e_k$ is dominated by computing the limits $a_k, b_k$ which takes $O(n)$ operations. Therefore, the cost of sampling one mode is $O(n)$. Going through all modes takes $O(n^2)$, the same cost as component sampling.

So far so good! Having only the 2-dimensional picture in mind (Fig. 3.5), one may be led to think that modes sampling is the best way of sampling a truncated multivariate Gaussian as it gets rid of all the correlation in the matrix $\mathbf{K}$. However, in higher dimensions, it can also become quite inefficient depending on the shape of the exact model (see Fig. 3.6). This happens whenever the original model has both comparatively very large and very small values (Fig. 3.6b). Due to the positivity constraints (Eq. 3.45), the truncation interval of a coefficient is determined mainly by the lowest values of the model in the region where the corresponding mode is concentrated. Since most modes are non-local, their coefficients will be sampled from very small intervals leading to very small updates on the model and thus high correlation times.

**Kernel modification** A quick fix to this problem is to modify the model such that the discrepancy between its values is reduced. Of course, we do not know the exact shape of the model (otherwise, we would not need analytic continuation in the first place), but we may be able to guess its overall shape, i.e., where it has large values and where it has small values. For the model shown in Fig. (3.6b), a half-Lorentzian whose half-width is roughly 3 would be a good guess for example. Let such a guess be denoted as $m(x)$,



(a) Model 1

(b) Model 2

Figure 3.6.: Modes sampling is much more efficient for the original model shown on the left than for the one shown on the right. This is because the second model has many values near zero while the first one does not. The two models are actually the same; the left one is just truncated earlier than the right one.

then $f(x)/m(x)$ would have less discrepancy than $f(x)$ itself. So instead of solving the original problem with kernel $K(x, y)$

$$g(y) = \int dx \; f(x) \; K(x, y) \; , \tag{3.46}$$

we solve the equivalent problem using the modified kernel $K(x, y)m(x)$

$$g(y) = \int dx \; \left[ \frac{f(x)}{m(x)} \right] \; [K(x, y)m(x)] \; . \tag{3.47}$$

The result of stochastic sampling using the modified kernel will be $f(x)/m(x)$ instead of $f(x)$ which can be fixed by multiplying the result with the modification $m(x)$ afterwards. The two problems are completely equivalent as long as the modification is strictly positive.

We found that using kernel modification does indeed accelerate the convergence of modes sampling for the aforementioned cases, and it even makes some calculations possible that would be practically impossible without the modification. However, such calculations may still take very long time for pathological cases. More importantly, the usefulness of the kernel modification depends on our guessing of the shape of the function we are trying to recover; such information may not be available. Therefore, we need a better way of sampling that can handle such cases without the need of kernel modification. It is *blocked modes sampling.*

### Blocked modes sampling (BMS)

Modes sampling works best for models whose values vary little (see Fig. 3.6). Therefore, when the model has several regions with considerably different values, it makes sense to block components corresponding to the same region together and apply modes sampling

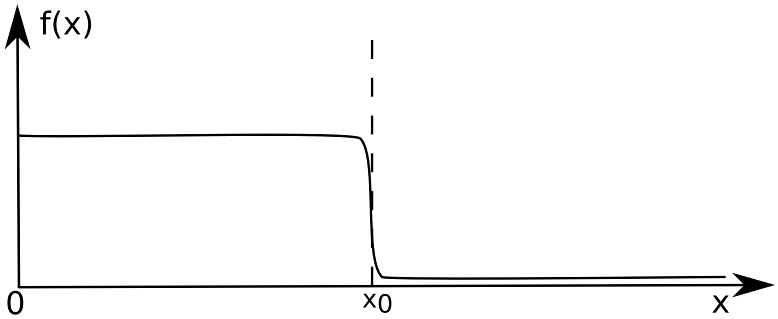


Figure 3.7.: Sampling around such a model is inefficient using modes sampling. However, applying modes sampling separately to the regions before and after $x_0$ leads to very efficient sampling.

to these blocks separately. This is the basic idea of blocked modes sampling. We will explain next the simple case of two regions.

Suppose we are sampling around the model shown in Fig. 3.7. This model has two regions; it has large values in the first region and comparatively small values in the second one. Let $\mathbf{f_1}, \mathbf{f_2}$ be the vectors composed of model components corresponding to the first and second regions, respectively. Similarly let $\mathbf{K_1}, \mathbf{K_2}$ be the matrices composed of the corresponding columns of $\mathbf{K}$. In terms of these, we rewrite the fit $\chi^2$ as following

$$\begin{aligned}\chi^2 &= (\mathbf{g} - \mathbf{K}\ \mathbf{f})^{\mathrm{T}} (\mathbf{g} - \mathbf{K}\ \mathbf{f}) && (3.48)\\ &= \left[\mathbf{g} - \begin{pmatrix}\mathbf{K_1} & \mathbf{K_2}\end{pmatrix} \begin{pmatrix}\mathbf{f_1}\\ \mathbf{f_2}\end{pmatrix}\right]^{\mathrm{T}} \left[\mathbf{g} - \begin{pmatrix}\mathbf{K_1} & \mathbf{K_2}\end{pmatrix} \begin{pmatrix}\mathbf{f_1}\\ \mathbf{f_2}\end{pmatrix}\right] && (3.49)\\ &= (\mathbf{g} - \mathbf{K_1 f_1} - \mathbf{K_2 f_2})^{\mathrm{T}} (\mathbf{g} - \mathbf{K_1 f_1} - \mathbf{K_2 f_2})\ . && (3.50)\end{aligned}$$

Similar to components sampling, we can sample $\mathbf{f_1}$ conditional on $\mathbf{f_2}$. Subtracting the contribution of $\mathbf{f_2}$ from the data and denoting $\mathbf{g_1} := \mathbf{g} - \mathbf{K_2}\ \mathbf{f_2}$, we have

$$\chi^2(\mathbf{f_1}; \mathbf{f_2}) = (\mathbf{g_1} - \mathbf{K_1 f_1})^{\mathrm{T}} (\mathbf{g_1} - \mathbf{K_1 f_1})\ . \qquad (3.51)$$

Clearly $p(\mathbf{f_1}|\mathbf{f_2})$ is a Gaussian with kernel matrix $\mathbf{K_1}$ and data $\mathbf{g_1}$ and it can be sampled using modes sampling. The same argument goes for $\mathbf{f_2}$ conditional on $\mathbf{f_1}$. It is also distributed as a Gaussian with kernel matrix $\mathbf{K_2}$ and data $\mathbf{g_2} := \mathbf{g} - \mathbf{K_1}\ \mathbf{f_1}$.

Blocked modes sampling starts from some initial sample $\mathbf{f} = (\mathbf{f_1}, \mathbf{f_2})^{\mathrm{T}}$. It then subtracts the contribution of $\mathbf{f_2}$ from the data and samples $\mathbf{f_1}$ using the modes of the matrix $\mathbf{K_1}$. After sampling all the modes of $\mathbf{f_1}$, it switches to $\mathbf{f_2}$, subtracts the contribution of the new $\mathbf{f_1}$ from the data and samples $\mathbf{f_2}$ using the modes of the matrix $\mathbf{K_2}$. After sampling the modes of $\mathbf{f_2}$, we have our first sample. This procedure is then repeated to obtain the desired number of samples. Generalization to an arbitrary number of blocks is straightforward: Always subtract the contribution of the other blocks from the data and sample using modes of the matrix corresponding to that block. Notice that components sampling and modes sampling are just special cases of blocked modes sampling. In components sampling, each block contains one component, while modes sampling has one big block containing all the components.

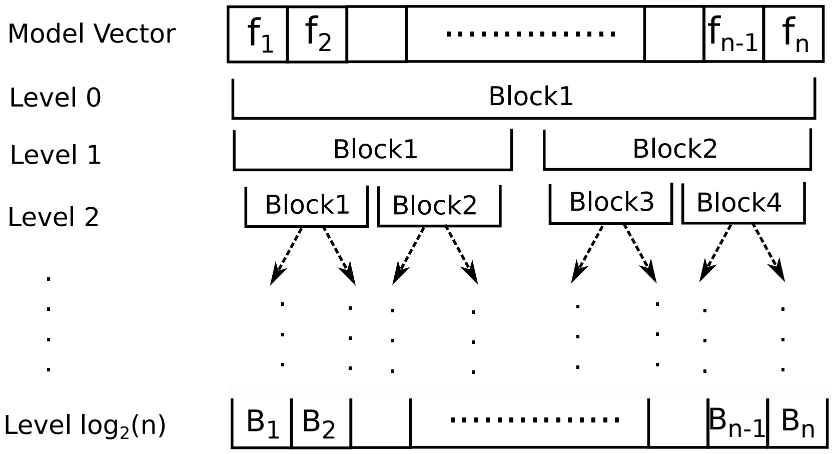


Figure 3.8.: An example of a hierarchy of partitions used by BMS.

The best partition for the case shown in Fig. 3.7 is two blocks touching around $x_0$. In practice, however, the best partitioning is not known beforehand. First, we probably do not know the shape of the model itself, so we may have no idea about the regions of similar values. Second, the transition between regions may not be as sharp as shown in Fig. 3.7 which makes choosing the boundaries of the regions fuzzy (see Fig. 3.6a). This is solved by switching between different partitions during the sampling. A systematic way of choosing partitions is using a hierarchy of partitions like a binary tree (see Fig. 3.8). BMS then switches randomly during sampling between the the different levels (partitions). Intuitively, higher levels of the hierarchy are responsible for global updates of the model while lower levels are responsible for updating the local details. We also shifted each other level of partitions by half the block size. This avoids the alignment of the boundaries of levels which may cause artifacts at these boundaries, requiring a lot of sampling to average out.

**Computational complexity** Let $b$ be the length of a block. Modes sampling of that block takes $O(b^2)$ while computing the contribution of the other blocks to the data takes $O(n-b)$. So in total we have $O(b^2+n)$ operations per block. Since we have $(n/b)$ blocks, the total cost is $O(n^2/b+nb)$ which equals $O(n^2)$ for $b$ between 1 and $n$ . Therefore, the cost of obtaining one sample is $O(n^2)$ regardless of the partition.

For sampling the modes we need to to compute the singular value decomposition of all blocks. This takes $O(b^3)$ operations per block and thus $O(nb^2)$ per level (partition). This is done only once in the initialization phase, and the results are cached and reused during sampling.

**Efficiency** In Fig. 3.9 and Fig. 3.10, we compare components sampling, modes sampling and BMS for the analytic continuation of the models shown in Fig. 3.6a and Fig 3.6b, respectively. All three methods start from the same initial sample which is chosen, for the sake of illustration, to have a very low probability.

Thermalization plots show how fast the sampling moves towards the high probability region. For model 1 (Fig. 3.9a), modes sampling and BMS thermalize almost instantly while components sampling takes many more iterations. For model 2 (Fig. 3.10a), the performance of modes sampling decreases significantly due to the discrepancy of the values of the model.

Step-size plots show the relative length of the random step taken by sampling at each iteration. It is defined as the ratio of the norm of the step vector to the norm of the new sample vector $\Delta_i = \parallel \mathbf{f}^{(i)} - \mathbf{f}^{(i-1)} \parallel / \parallel \mathbf{f}^{(i)} \parallel$. Regardless of the model, BMS has the largest step size while components sampling has relatively small step size. Modes sampling, on the other hand, takes large steps for model 1 and very tiny steps for model 2. These results confirm that whatever the model is, BMS is the method of choice for performing the sampling.

### Initialization

An ergodic Monte Carlo simulation converges eventually to the desired sampled distribution. However, depending on the starting point, the simulation may follow at the beginning a different distribution, and a certain number of samples at the beginning of the simulation (called thermalization period) should be discarded. If we could start immediately inside the high probability region, then the thermalization period can be drastically shortened.

In analytic continuation, we are sampling a multivariate distribution truncated to the non-negative region. The maximum of this distribution is nothing but the non-negative least squares solution. This solution could be a good starting point, but it has the disadvantage of lying on the boundary of the non-negative region (i.e. it has many zeros). Therefore, we use a fast regularization method that respects the constraints e.g. non-negative Tikhnonv or the perturbed data sampling method (see the previous chapter). Such an initial model would have a high probability (because it has good fit) and lies inside the allowed region (because it is regularized).

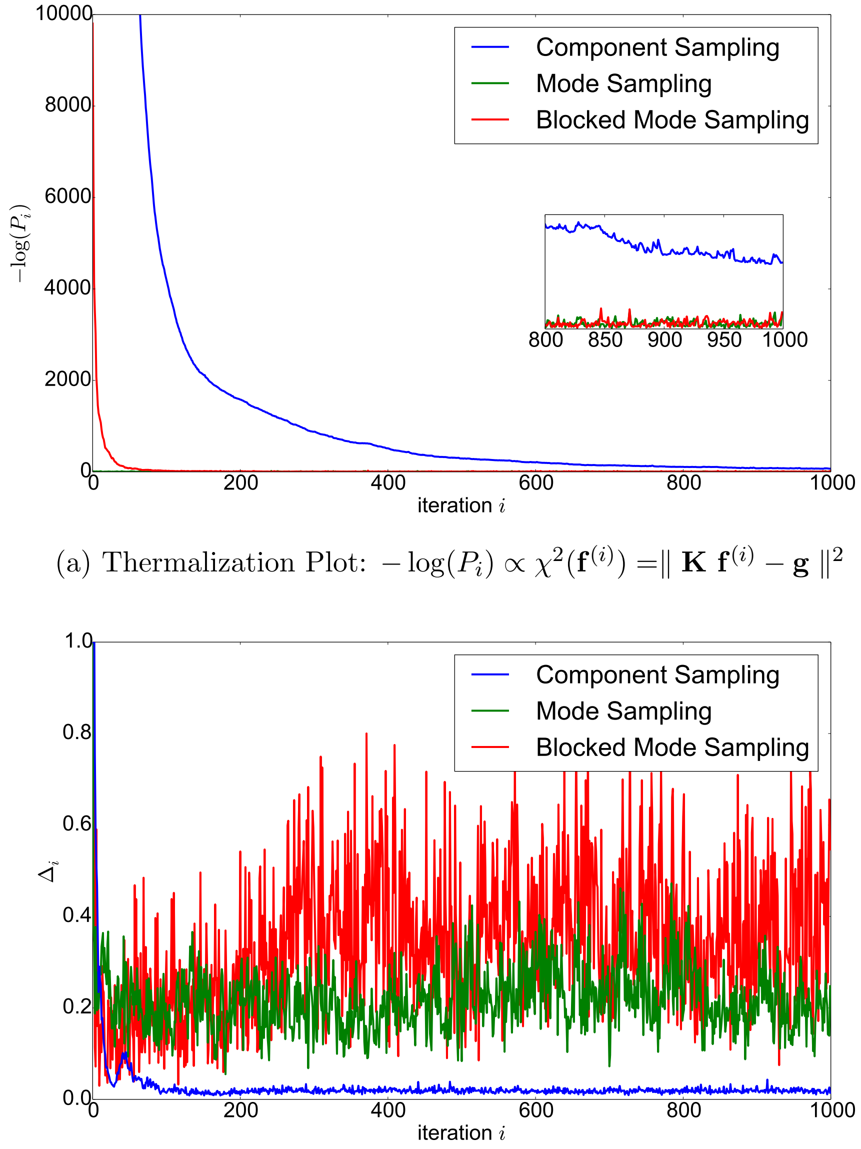
### Imposing sum rules

In the previous sections, we assumed only non-negativity constraints to simplify the arguments. Now we show how to impose $q$ sum rules. After discretization, the sum rules become a set of equality constraints written concisely in vector form

$$\mathbf{C}\ \mathbf{f} = \mathbf{d}\ , \tag{3.52}$$

where the matrix $\mathbf{C} \in \mathbb{R}^{q\times n}$. Using a QR decomposition or singular value decomposition, we can find a complete basis $\mathbf{Q} = [\mathbf{Q}_{\text{const}} \quad \mathbf{Q}_{\text{free}}] \in \mathbb{R}^{n\times n}$ where $\mathbf{Q}_{\text{const}} \in \mathbb{R}^{n\times q}$ is a basis spanning the row space of $\mathbf{C}$, while $\mathbf{Q}_{\text{free}} \in \mathbb{R}^{n\times(n-q)}$ is a basis spanning the null space of $\mathbf{C}$. Clearly the projection coefficients of the model on the row space are determined by the constraints, while the rest are free. The above equation then reads

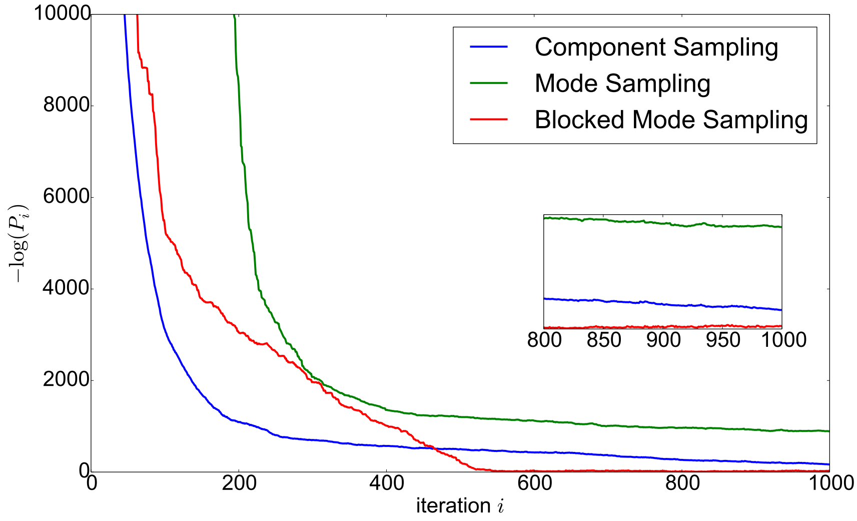
$$\underbrace{\mathbf{C}\ \mathbf{Q}_{\text{const}}}_{\mathbf{C}'}\underbrace{\mathbf{Q}_{\text{const}}^{\mathrm{T}}\ \mathbf{f}}_{\mathbf{f}_{\text{const}}} = \mathbf{d} \tag{3.53}$$



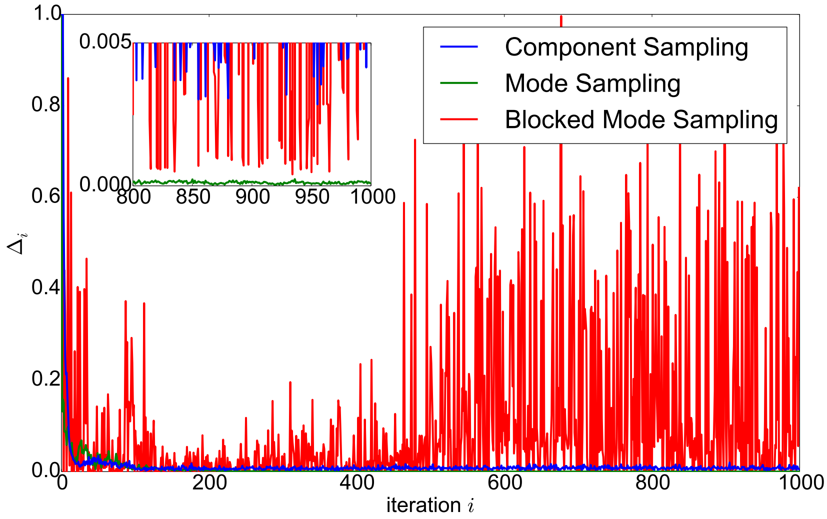
(a) Thermalization Plot: $-\log(P_i) \propto \chi^2(\mathbf{f}^{(i)}) = \parallel \mathbf{K}\ \mathbf{f}^{(i)} - \mathbf{g} \parallel^2$

(b) Step-Size Plot: $\Delta_i = \parallel \mathbf{f}^{(i)} - \mathbf{f}^{(i-1)} \parallel / \parallel \mathbf{f}^{(i)} \parallel$

Figure 3.9.: Comparison of the efficiency of different sampling algorithms for the analytic continuation of the model shown in Fig. 3.6a (a model with small discrepancy in its values). In the thermalization plot, we notice that modes sampling and BMS thermalize almost instantly while components sampling takes many more iterations. In the step-size plot, we notice that BMS and modes sampling take large steps while components sampling has a relatively small step size.



(a) Thermalization Plot: $-\log(P_i) \propto \chi^2(\mathbf{f}^{(i)}) = \parallel \mathbf{K}\ \mathbf{f}^{(i)} - \mathbf{g} \parallel^2$



(b) Step-Size Plot: $\Delta_i = \parallel \mathbf{f}^{(i)} - \mathbf{f}^{(i-1)} \parallel / \parallel \mathbf{f}^{(i)} \parallel$

Figure 3.10.: Comparison of the efficiency of different sampling algorithms for the analytic continuation of the model shown in Fig. 3.6b (a model with large discrepancy in its values). Notice that BMS has the best thermalization and takes the largest steps. Comparing with Fig. 3.9, we notice that the performance of components sampling is similar because this sampling is not affected by the shape of the model. On the other hand, modes sampling thermalizes much more slowly and its steps become very small because the model has both small and large values.

We can now split the model using the new basis $\mathbf{Q}$ into constrained and free parts

$$\mathbf{f} = \mathbf{Q}_{\text{const}}\mathbf{f}_{\text{const}} + \mathbf{Q}_{\text{free}}\mathbf{f}_{\text{free}} \ . \tag{3.54}$$

The averaging is then done only on the free part

$$\mathbf{f}_{\text{StochS}} = \mathbf{Q}_{\text{const}}\mathbf{f}_{\text{const}} + \frac{1}{C} \int\limits_{\mathbf{Q}_{\text{free}}\mathbf{f}_{\text{free}}+\mathbf{Q}_{\text{const}}\mathbf{f}_{\text{const}}>0} d\mathbf{f}_{\text{free}} \ \mathbf{f}_{\text{free}} \ \exp\left(-\frac{1}{2}\chi^2\right) \tag{3.55}$$

and the fit $\chi^2$ can be written in terms of $\mathbf{f}_{\text{free}}$ as

$$\begin{aligned}
\chi^2 &= (\mathbf{g} - \mathbf{K}\ \mathbf{f})^{\mathrm{T}} (\mathbf{g} - \mathbf{K}\ \mathbf{f}) &(3.56)\\
&= \left(\mathbf{g} - \mathbf{K}\ \mathbf{Q}\mathbf{Q}^{\mathrm{T}}\mathbf{f}\right)^{\mathrm{T}} \left(\mathbf{g} - \mathbf{K}\ \mathbf{Q}\mathbf{Q}^{\mathrm{T}}\mathbf{f}\right) &(3.57)\\
&= \left[\mathbf{g} - \begin{pmatrix}\mathbf{K}\mathbf{Q}_{\text{const}} & \mathbf{K}\mathbf{Q}_{\text{free}}\end{pmatrix} \begin{pmatrix}\mathbf{f}_{\text{const}} \\ \mathbf{f}_{\text{free}}\end{pmatrix}\right]^{\mathrm{T}} \left[\mathbf{g} - \begin{pmatrix}\mathbf{K}\mathbf{Q}_{\text{const}} & \mathbf{K}\mathbf{Q}_{\text{free}}\end{pmatrix} \begin{pmatrix}\mathbf{f}_{\text{const}} \\ \mathbf{f}_{\text{free}}\end{pmatrix}\right] &(3.58)\\
&= (\mathbf{g}_{\text{free}} - \mathbf{K}_{\text{free}}\mathbf{f}_{\text{free}})^{\mathrm{T}} (\mathbf{g}_{\text{free}} - \mathbf{K}_{\text{free}}\mathbf{f}_{\text{free}}) \ , &(3.59)
\end{aligned}$$

where $\mathbf{K}_{\text{free}} \coloneqq \mathbf{K}\mathbf{Q}_{\text{free}}$ is the projection of modified kernel on the null space, and $\mathbf{g}_{\text{free}} \coloneqq \mathbf{g} - \mathbf{K}\mathbf{Q}_{\text{const}}\mathbf{f}_{\text{const}}$ is the modified data after subtracting the contribution of the constrained part of the model. The last equation (3.59) shows that $\mathbf{f}_{\text{free}}$ is distributed as a multivariate Gaussian with matrix $\mathbf{K}_{\text{free}}$ and data $\mathbf{g}_{\text{free}}$, and it can be sampled using either modes sampling or more generally BMS.

Applying modes sampling to $\mathbf{f}_{\text{free}}$ is straightforward. Instead of using the modes of $\mathbf{K}$, we use the modes of $\mathbf{K}_{\text{free}}$. The main modification needed is computing the truncation limits in Eq. (3.45), where instead of using the positivity constraints, we use the condition $\mathbf{Q}_{\text{free}}\mathbf{f}_{\text{free}} > -\mathbf{Q}_{\text{const}}\mathbf{f}_{\text{const}}$.

Generalizing to BMS is also easy. For example, if we have two blocks $\mathbf{f_1}, \mathbf{f_2}$, we split Eq. (3.52) accordingly

$$\mathbf{C_1}\mathbf{f_1} + \mathbf{C_2}\mathbf{f_2} = \mathbf{d} \ , \tag{3.60}$$

where $\mathbf{C_1}, \mathbf{C_2}$ are the parts of the matrix $\mathbf{C}$ corresponding to the first and second blocks, respectively. Then we sample the first block $\mathbf{f_1}$ using modes sampling with constraints matrix $\mathbf{C_1}$ and right hand side $\mathbf{d} - \mathbf{C_2}\mathbf{f_2}$. Similarly, the second block is sampled with constraints matrix $\mathbf{C_2}$ and right hand side $\mathbf{d} - \mathbf{C_1}\mathbf{f_1}$.

Pay attention that the block size should always be larger than the number of linearly independent constraints on that block. Otherwise, the null space corresponding to the blocked constraints matrix is empty and the model values in that block are completely determined by the constraints and the rest of the model. We detect such blocks in the initialization and avoid sampling using them. For example, when a normalization constraint exists, component sampling cannot be used because we cannot move components individually without violating the constraint.

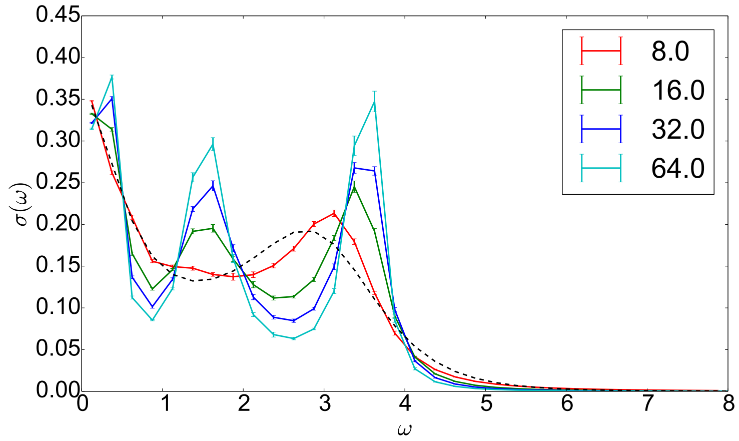


Figure 3.11.: StochS results of test case 3 using a uniform grid with spacing 0.25 and increasing cutoffs.

### 3.3.2. Grid dependence

The StochS result shown in Fig. 3.1 was obtained using a uniform grid with 32 points and cutoff 8. The result differs somehow from the exact model, so one would expect that by improving the grid, the results should get better because the discretization and cutoff errors would get smaller. Surprisingly, increasing the cutoff makes the results worse as shown in Fig. 3.11. As the cutoff increases, spurious peaks develop, and they become more and more pronounced for larger cutoffs! Moreover, the cutoff dependence cannot be attributed to the discretization error because we get the same results using numerical data i.e. data generated using the same grid used in StochS. This suggests that the grid dependence is an inherent property of the method. In order to study this dependence, let us first show how to systematically build different grids.

**Building a grid** An arbitrary grid of a variable $x \in [a, b]$, where $a$ is possibly $-\infty$ and $b$ is possibly $+\infty$, can be specified by the density of its points $\rho(x)$ and the total number of points $n$ which we call the grid size. The grid density function $\rho(x)$ is a normalized positive function whose integral over any interval fixes the fraction of grid points in that interval. Given $\rho(x)$ and $n$, the grid points $x_i$ and weights $\Delta x_i$ can be determined as following. Define the following variable transformation

$$P : x \to z : \int_a^x dx' \ \rho(x') \ . \tag{3.61}$$

Since $\rho(x)$ is strictly-positive and normalized, its antiderivative $P(x)$ is monotonic, and thus defines a one-to-one mapping between $x \in [a, b]$ and the new variable $z \in [0, 1]$ (see Fig. 3.12). Dividing the interval $[0, 1]$ of $z$ into $n$ sub-intervals of equal width

$$[0, z_1], [z_1, z_2], ..., [z_{n-1}, 1] \quad \text{where} \quad z_i = \frac{i}{n} \ , \tag{3.62}$$

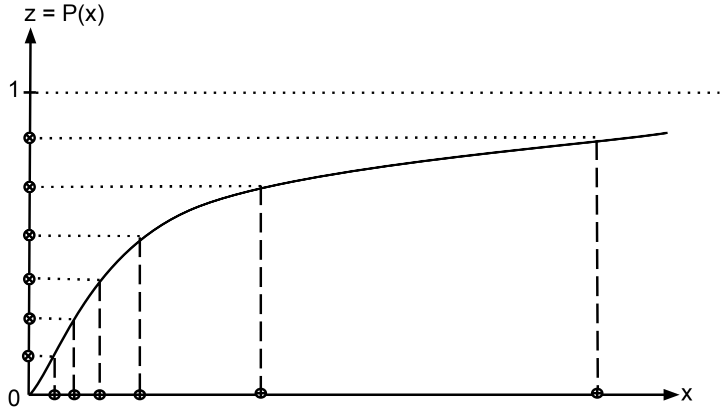


Figure 3.12.: Any non-uniform grid can be built using a uniform grid of a transformed variable. Therefore, this transformation (or its derivative, the grid density) characterizes the non-uniform grid completely (up to the total number of points). Remember that $z = 0$ maps into $x = a$ and that $z = 1$ maps into $x = b$, which may be infinites.

leads to dividing the interval $[a, b]$ of $x$ into $n$ generally non-equal sub-intervals

$$[a, x_1], [x_1, x_2], ...[x_{n-1}, b] \quad \text{where} \quad x_i = P^{-1}(z_i) \; . \tag{3.63}$$

In principle, we could now use this grid to numerically evaluate any integral of the variable $x$, but we would run into trouble when the first or last sub-interval extends to infinity. We can avoid this problem nicely by rewriting the integral in terms of $z$ and evaluating it on the uniform grid of $z$ using the mid-point rule

$$\int_a^b dx\ \phi(x) = \int_0^1 dz\ \frac{dx}{dz}\ \phi(P^{-1}(z)) \approx \frac{1}{n}\sum_{i=0}^{n-1}\ \frac{dx}{dz}\bigg|_{z=z_i^\star}\ \phi(P^{-1}(z_i^\star)) \tag{3.64}$$

$$\text{where} \quad z_i^\star = \frac{z_i + z_{i+1}}{2} = \frac{2i+1}{2n} \quad . \tag{3.65}$$

This gives following evaluation points in the $x$ variable

$$x_i^\star := P^{-1}(z_i^\star); , \tag{3.66}$$

with the following weights

$$\Delta x_i := \frac{1}{n}\ \frac{dx}{dz}\bigg|_{z=z_i^\star} = \frac{1}{n\rho(x_i^\star)} \; . \tag{3.67}$$

Using these points and weights instead of the midpoints and lengths of the non-uniform intervals of $x$, avoids the problem with infinite intervals.

**Grid examples** Obviously taking a constant density in the interval $[a, b]$ leads to a uniform grid on that interval

$$\rho(x) = \frac{dz}{dx} = \frac{1}{b-a} \Rightarrow z = P(x) = \frac{x-a}{b-a} \Rightarrow$$
$$x = P^{-1}(z) = a + (b-a)z \Rightarrow \frac{dx}{dz} = (b-a)\ . \tag{3.68}$$

We can get a highly-compressed grid on the interval $[0, \infty[$ by using an exponential density function with scale $\beta$

$$\rho(x) = \frac{dz}{dx} = \frac{1}{\beta} e^{-x/\beta} \Rightarrow z = P(x) = 1 - e^{-x/\beta} \Rightarrow$$
$$x = P^{-1}(z) = -\beta \ln(1-z) \Rightarrow \frac{dx}{dz} = \frac{\beta}{1-z}\ . \tag{3.69}$$

We can easily reflect this grid about zero to get a grid on the whole real line or we can use a Gaussian density function with variance $\sigma^2$

$$\rho(x) = \frac{dz}{dx} = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{x^2}{2\sigma^2}\right) \Rightarrow z = P(x) = \frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{x}{\sigma\sqrt{2}}\right)\right] \Rightarrow$$
$$x = P^{-1}(z) = \sqrt{2}\sigma\ \mathrm{erf}^{-1}(2z-1) \Rightarrow \frac{dx}{dz} = \sigma\sqrt{2\pi}\ \exp\left[2\ \mathrm{erf}^{-1}(2z-1)\right] \tag{3.70}$$

Another useful density that is also concentrated near zero but less compressed than the Gaussian grid, is a Lorentzian grid with half-width $\gamma$

$$\rho(x) = \frac{dz}{dx} = \frac{1}{\pi\gamma\left[1 + (x/\gamma)^2\right]} \Rightarrow z = P(x) = \frac{1}{\pi}\tan^{-1}\left(\frac{x}{\gamma}\right) + \frac{1}{2} \Rightarrow$$
$$x = P^{-1}(z) = \gamma\ \tan\left(\pi z - \frac{\pi}{2}\right) \Rightarrow \frac{dx}{dz} = \frac{2\gamma\pi}{1 - \cos(2\pi z)} \tag{3.71}$$

When needed, both the Gaussian and the Lorentzian grids can be easily truncated to the positive axis.

**Grid as a default model** In Fig. 3.13, we compare the results of a uniform grid with a Lorentzian grid and an exponential grid. Again the results are slightly different for different grids. To see the grid dependence more clearly, we look at the solutions for large $\omega$, where the information provided by the data is weak. Plotting those solutions on a logarithmic scale, we notice that the solution of the Lorentzian grid has a second-order decay (Fig. 3.14a) while the solution of the exponential grid has an exponential decay (Fig. 3.14b). In both cases, the decay of the solution is the same as the decay of the grid density! Furthermore, if we repeat the calculations using only the first data point $\Pi(0)$, which determines the normalization, then the solution on a grid is, up to a normalization factor, the density function of the grid (see Fig. 3.15). These results show that the grid density plays the role of a default model! In the next section, we will explain the grid dependence and show how to modify the flat prior used in StochS to simulate the results of one grid using another.

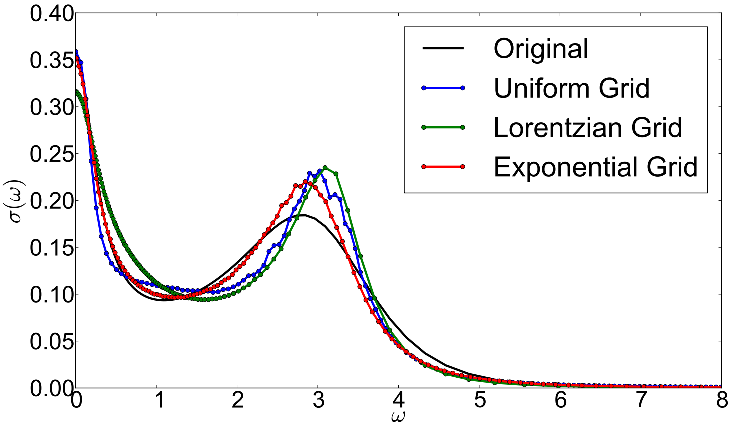
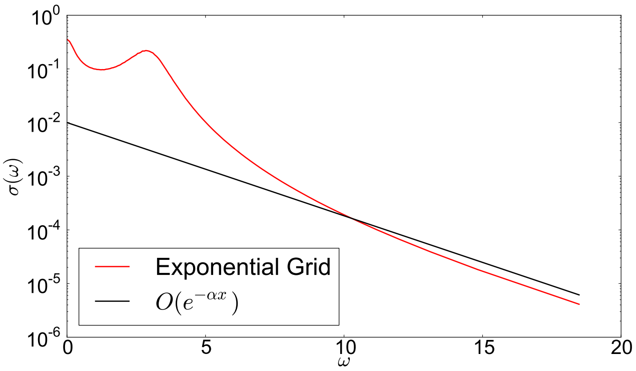
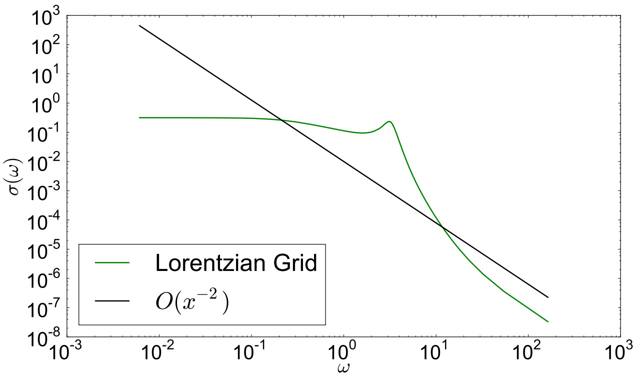


Figure 3.13.: StochS results of test case 3 using three different grids: a uniform grid with cutoff 8, a Lorentzian grid with half-width $\gamma = 2.5$ and an exponential grid with scale $\beta = 3$.



(a) Lorentzian grid results of Fig. 3.13 plotted on a semi-log scale.



(b) Exponential grid results of Fig. 3.13 plotted on a log-log scale.

Figure 3.14.: The decay of a solution matches the decay of the grid density.

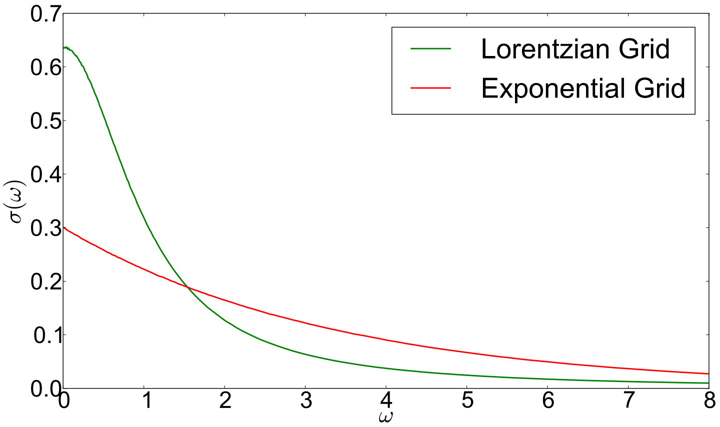


Figure 3.15.: Results using only information about the normalization. Notice that we obtain a Lorentzian function on a Lorentzian grid and an exponential function on an exponential grid. In general, the result is always proportional to the grid density.

### 3.3.3. Grid simulation

Although the grid dependence seems perplexing at first glance, we can actually understand it in a straightforward way. Starting from a fine grid, StochS assumes a flat prior on the components of the fine model. Forming a coarser grid model from the fine one corresponds to averaging consecutive values, but the average of non-negative uniform variables is not itself a uniform variable, and therefore the components of the resulting coarse model are not distributed uniformly. On the contrary, had we started directly from a coarse grid, StochS would have assumed again a flat prior on the components of the coarse model which contradicts the non-uniform distribution implied by the fine grid! This shows that the grid dependence is the result of having different assumptions on different grids. In order to get the same StochS results using two different grids, we should *assign the same prior to the same quantity*. In this section, we show how to modify the priors such that different grids can have the same assumptions about the model, and thus give the same results.

To simplify the following discussion, we will work with the model integrals $\mathbf{F}$ over grid intervals instead of the average values $\mathbf{f}$. This does not affect the results of StochS because a flat prior on $f_i$ implies a flat prior on $F_i$ and vice versa. The advantage is that moving from one grid to another would then correspond to summing consecutive components of the vector $\mathbf{F}$ instead of averaging the components of $\mathbf{f}$.

Before we start working on the prior, we need to be able to manipulate the uniform distributions assumed by StochS in a mathematically proper way. Assuming only non-negativity constraints, each model component is distributed uniformly over the positive values. A simple and straightforward trick is to using an exponential distribution

$$x \sim \mathrm{Exp}(\lambda) \Leftrightarrow p(x) = \lambda e^{-\lambda x} \; . \qquad (3.72)$$

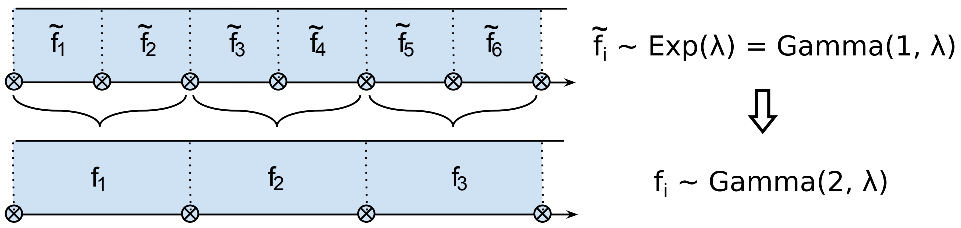


Figure 3.16.: Going from a fine grid to a coarse one. Assuming exponential distributions on the fine grid leads to gamma distributions on the coarse one.

As the rate parameter $\lambda$ goes to zero, this distribution approaches a uniform distribution over non-negative values. We can thus replace the uniform distributions with exponential distributions all having the same rate parameter, manipulate these exponential distributions, and then take the limit at the end.

Let us discuss the simplest possible case, going from a fine grid model $\tilde{\mathbf{F}} \in \mathbb{R}^{kn}$ to a $k$-times coarser grid model $\mathbf{F} \in \mathbb{R}^n$ (see Fig. 3.16). The components of the fine model are distributed as

$$\tilde{F}_i \sim \mathrm{Exp}(\lambda), \qquad \text{(using fine grid)} . \tag{3.73}$$

The components of the corresponding coarse model can be be obtained as the sum of every $k$ consecutive components of the fine model

$$F_i = \sum_{j=0}^{k-1} \tilde{F}_{ki+j}. \tag{3.74}$$

When $k = 2$, we can easily compute the distribution of the sum as following

$$p(F_i) = \int_0^\infty d\tilde{F}_{2i}\ \lambda e^{-\lambda \tilde{F}_{2i}} \int_0^\infty d\tilde{F}_{2i+1}\ \lambda e^{-\lambda \tilde{F}_{2i+1}}\ \delta(\tilde{F}_{2i} + \tilde{F}_{2i+1} - F_i) \tag{3.75}$$

$$= \int_0^{F_i} d\tilde{F}_{2i}\ \lambda e^{-\lambda \tilde{F}_{2i}}\ \lambda e^{-\lambda (F_i - \tilde{F}_{2i})} = \lambda^2 e^{-\lambda F_i} \int_0^{F_i} d\tilde{F}_{2i} \tag{3.76}$$

$$= \lambda^2 F_i\ e^{-\lambda F_i} \tag{3.77}$$

For larger values of $k$, the distribution can be computed similarly

$$p(F_i) = \frac{\lambda^k}{(k-1)!}\ F_i^{k-1}\ e^{-\lambda F_i}\ . \tag{3.78}$$

This distribution is known as the gamma distribution with shape parameter $k$ and rate parameter $\lambda$, and it is denoted as Gamma$(k, \lambda)$. In other words, the sum of $k$ exponential random variables Exp$(\lambda)$ is a gamma random variable Gamma$(k, \lambda)$ (Notice that the exponential distribution itself is a gamma distribution with shape parameter $k = 1$). Hence, applying StochS to the fine model $\tilde{\mathbf{F}}$ implies a gamma distribution on the corresponding coarse components

$$F_i \sim \mathrm{Gamma}(k, \lambda), \qquad \text{(using fine grid)} . \tag{3.79}$$

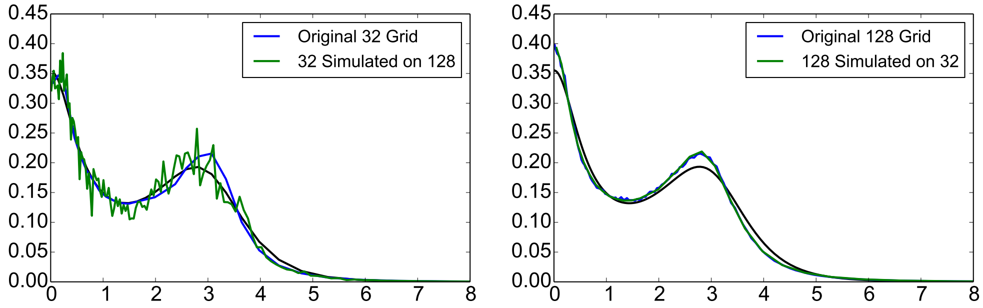


Figure 3.17.: Simulating an exponential grids with 32 points on 128 points grid and vice versa. The large fluctuations in the left figure are the result of a low acceptance ratio. It is hard to simulate a coarse grid on a fine one because of the low acceptance ratio.

If we, however, apply StochS directly to the coarse model $\mathbf{F}$, it will assume an exponential prior

$$F_i \sim \mathrm{Exp}(\lambda), \qquad \text{(using coarse grid)} \; . \tag{3.80}$$

Therefore, we can simulate the results of the fine grid using a coarse grid by multiplying the probability of a model by the distribution of Eq. (3.79) and dividing it by the distribution of Eq. (3.80). This modification reads

$$P'(\mathbf{F}) = \frac{p^{\text{fine}}(F_1, F_2, ..., F_n)}{p^{\text{coasre}}(F_1, F_2, ..., F_n)} = \frac{p^{\text{Gamma}(k,\lambda)}(F_1, F_2, ..., F_n)}{p^{\text{Exp}(\lambda)}(F_1, F_2, ..., F_n)} \tag{3.81}$$

$$\propto \frac{\prod_{i=1}^{n} F_i^{k-1} e^{-\lambda F_i}}{\prod_{i=1}^{n} e^{-\lambda F_i}} = \prod_{i=1}^{n} F_i^{k-1} \; . \tag{3.82}$$

It is independent of $\lambda$, so it remains valid in the limit $\lambda \to 0$.

For the previous result, we assumed that the number of fine grid points is an integer multiple $k$ of the number of coarse grid points. Since the gamma distribution accepts real shape parameters, it is compelling to extend the result to any real $k > 0$. This means that we can simulate a grid of $n_1$ points on a grid of the same type and $n_2$ points by modifying the prior by the factor

$$P'(\mathbf{F}) \propto \prod_{i=1}^{n} F_i^{n_1/n_2 - 1} \; . \tag{3.83}$$

Note that even when $n_2 > n_1$, the argument is still valid. This is because an exponential random variable $x \sim \mathrm{Exp}(\lambda)$ can be written as the sum of $n$ independent and identically distributed gamma random variables $x_i \sim \mathrm{Gamma}(1/n, \lambda)$. Fig. 3.17 shows the results of simulating grids with different number of points. These results are obtained by replacing the flat prior with the modified one in StochS average

$$\mathbf{F}_{\text{Simulated StochS}} = \frac{1}{C} \int_{\mathcal{F}} d\mathbf{F} \; P'(\mathbf{F}) \; \exp\left[-\frac{1}{2}\chi^2(\mathbf{F})\right] \; \mathbf{F} \; . \tag{3.84}$$

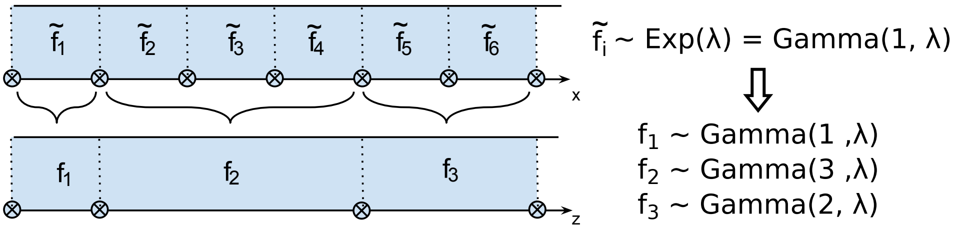


Figure 3.18.: Going from a uniform grid to a non-uniform one. Assuming exponential distributions on the uniform grid leads to gamma distributions on the non-uniform one.

BMS is used to perform StochS with this modified prior. Remember that the original BMS accepts all updates with probability one (Eq. 3.31). The only change needed is accepting the updates instead with probability $P'(\mathbf{F})$. We note that simulating a coarse grid on a fine one is much less efficient than simulating the other way around because the acceptance ratio is much smaller than one.

The modified prior can be generalized further to the case of simulating a uniform grid on a non-uniform one (see Fig. 3.18). A non-uniform grid can be formed from a uniform one by merging consecutive intervals. When the non-uniform grid has $n$ intervals and its $i$th interval is the result of merging $k_i$ intervals of the uniform gird, following the same argument as above, the prior probability should be a gamma variable with shape parameter $k_i$

$$P'(\mathbf{F}) \propto \prod_{i=1}^{n} F_i^{k_i-1} \,. \tag{3.85}$$

This can be used even when the non-uniform interval is not an integer multiple of the uniform one. In this case, we can use the ratio of the weights of the non-uniform interval to the uniform one as the value of parameter $k_i$ (see Eq. 3.67)

$$k_i = \frac{\Delta x_i}{\Delta x'} = \frac{n'}{n}\frac{1}{\rho(x_i)} \,, \tag{3.86}$$

where $\rho(x)$ is the density of the non-uniform grid and $n'$ is the size of the uniform one.

Finally using a very fine intermediate uniform grid, we can relate any grid with $n_1$ points and density $\rho_1(x)$ to any other grid with $n_2$ points and density $\rho_2(x)$. The probability modification when simulating the second one using the first one is

$$P'(\mathbf{F}) \propto \prod_{i=1}^{n_1} F_i^{k_i-1} \quad \text{where} \quad k_i = \frac{n_2}{n_1}\frac{\rho_2(x_i)}{\rho_1(x_i)}. \tag{3.87}$$

Fig. 3.19 shows the results of simulating two grids of different densities.

**Sum rules** The previous arguments are still valid in the presence of sum rules. Imposing a sum rule means multiplying the prior by a delta function of the form $\delta(\sum c_i F_i - d)$ which is identical for both grids (up to the discretization errors in the sum rule). Therefore, the two factors cancel out, and we get the same probability modification.

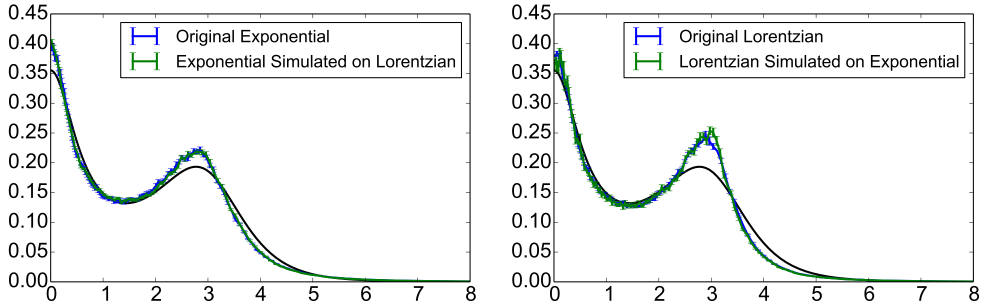


Figure 3.19.: Simulating an exponential grid using a Lorentzian one and vice versa.

### 3.3.4. Functional formulation

In the last two sections, we saw that the StochS results depend on the grid and that we can simulate the results of one grid using another by changing the prior distribution. This hassle came about because we were hasty in discretizing what is originally a continuous problem. What we should have done to avoid this, is applying Bayesian inference to the continuous problem and postponing the discretization to a later stage. This way the resulting discertized priors would be different for different discretizations, but they should give matching results because they all come from the same assumptions about the continuous case, a prior on function space.

Defining priors on function spaces is a complicated subject that requires measure-theoretic treatment of probability. On such infinite-dimensional spaces, one cannot talk about probability densities anymore but probability measures also known as stochastic processes. To keep things simple, one can think of such a prior measure as an integration measure over the function space. Then StochS becomes an average over functions with this integration measure weighted by the Gaussian factor of the data

$$\frac{1}{C}\int \mathcal{D}f\ f(x)\exp\left[-\frac{1}{2}\chi^2(f)\right] . \tag{3.88}$$

Discretizing this infinite-dimensional integration measure of functions using a grid $\mathbf{x}$ gives us a prior over the finite dimensional representations of these functions on that grid

$$\int \mathcal{D}f \xRightarrow{\text{discretization}} \int d\mathbf{F}P(\mathbf{F}) \tag{3.89}$$

In Appendix A, we explain how to build admissible prior processes (integration measures) for analytic continuation problem and give several examples.

The most important process for us here is the Gamma process $\mathrm{GP}(\lambda, \alpha, D)$. This process has three parameters: (1) a positive real number $\lambda > 0$, called rate parameter, (2) a positive real number $\alpha > 0$, called concentration parameter, and (3) a normalized positive function $D$ called the default model. This process produces non-negative random functions with mean

$$\mu(x) = \int [\mathcal{D}f]_{\mathrm{GP}(\lambda,\alpha,D)} f(x) = \frac{1}{\lambda}D(x) \tag{3.90}$$

and variance

$$\sigma^2(x) = \int [\mathcal{D}f]_{\mathrm{GP}(\lambda,\alpha,D)} \, [f(x) - \mu(x)]^2 = \frac{1}{\alpha\lambda^2} D(x) \tag{3.91}$$

Notice that the mean is proportional to $D(x)$ which justifies calling it a default model. The variance is inversely proportional to $\alpha$ which justifies calling it the concentration parameter.

Discretizing this process using a grid whose density function is $\rho(x)$ and size is $n$ gives a gamma distribution, the distribution used to simulate one grid of StochS over another

$$\int [\mathcal{D}f]_{\mathrm{GP}(\lambda,\alpha,D)} \xrightarrow[\text{grid density: } \rho(x), \text{ grid size: } n]{\text{discretization}} \frac{1}{C} \int_{\mathbf{F}>0} d\mathbf{F} \prod_i F_i^{\frac{a}{n}\frac{D(x_i)}{\rho(x_i)}-1} e^{-\lambda F_i} \; . \tag{3.92}$$

Moreover, by setting the default model to the grid density ($\rho = D$), setting the concentration parameter to the grid size ($n = \alpha$) and taking the limit of rate parameter to zero ($\lambda \to 0$), we get a flat distribution over all non-negative models on the grid

$$\int [\mathcal{D}f]_{\mathrm{GP}(0,n,\rho)} \xrightarrow[\text{grid density: } \rho(x), \text{ grid size: } n]{\text{discretization}} \frac{1}{C} \int_{\mathbf{F}>0} d\mathbf{F} \tag{3.93}$$

This is StochS prior distribution plain and simple! Therefore, we can establish a one-to-one mapping between StochS grids and gamma processes (with $\lambda \to 0$). StochS can be formulated as a Bayesian method on a function space with Gamma prior process whose default model equals the StochS grid density and whose concentration parameter equals the grid size.

$$f_{\mathrm{StochS}}(x; D, \alpha) = \frac{1}{C} \int [\mathcal{D}f]_{\mathrm{GP}(0,D,\alpha)} \, \exp\left[-\frac{1}{2}\chi^2(f)\right] \, f(x) \; . \tag{3.94}$$

This functional reformulation puts things in perspective mathematically. It explains why the grid density plays the role of the default model. It also shows that the grid size plays the role of the strength of the prior because it controls its variance: the more grid points, the stronger the bias towards the default model. Additionally, this formulation defines precisely the integration measure used by StochS (a gamma process). It shows that the result obtained by Beach [31], that MaxEnt is an approximation of StochS, is misleading because that derivation assumes a different integration measure (a multinomial process) as explained in the appendix.

Using this formulation, the StochS results become grid independent. However, its finite dimensional priors are not flat anymore. They are grid dependent and parameterized by a default model and concentration parameter.

### 3.3.5. Parametric formulation

Although the functional reformulation of StochS explained its grid dependence, it did not remove it. It just replaced this implicit dependence by an explicit dependence on the parameters of the gamma process. In the rest of this chapter, we will adopt a pragmatic approach to grid dependence and consider StochS to be averaging over models characterized not only by their integrals $\mathbf{F}$ but also the grid points $\mathbf{x}$ on which these integrals are defined. This way, we keep the problem finite dimensional and treat the grid points as parameters of the method to be selected (Sec. 3.3.6) or averaged over (Sec. 3.4).

Clearly, the model integrals are not enough to determine the model completely and some assumptions about its behavior inside the grid intervals are needed. We could assume a constant value over the interval or a delta function in the middle or any other non-negative integrable function. For a fine enough grid, however, this choice should not really affect the result. To stay as general as possible, we encode whatever assumptions in the object $f(x;\mathbf{F},\mathbf{x})$ which maps a set of gird points $\mathbf{x}$ and integrals $\mathbf{F}$ into a non-negative integrable function of the continuous variable $x$.

In contrast, the model integrals are enough to determine the data (up to an approximation error) without knowing the details of the model inside the grid intervals. Since the model is a non-negative integrable function, we can use the *First Mean Value Theorem for Integrals* to compute its data integrals as following

$$\begin{aligned} g(y) &= \int dx\ K(x,y)f(x) = \sum_i \int_{x_i}^{x_{i+1}} dx\ K(x,y)f(x) \\ &= \sum_i K(x_i^\star,y) \int_{x_i}^{x_{i+1}} dx\ f(x) = \sum_i K(x_i^\star,y)F_i\ , \end{aligned} \tag{3.95}$$

where each $x_i^\star$ is some specific point in the interval $[x_i, x_{i+1}]$. Till here no approximation is made! The approximation comes from not knowing the exact locations of the points $x_i^\star$, which depend on both the model $f(x)$ and the kernel $K(x,y)$. The approximation error is proportional to the difference between the maximum and minimum values of the kernel $K(x,y)$ inside each interval. Since the kernel is a continuous function of $x$, this error gets smaller, the smaller the intervals are and the smoother the kernel is. Using a fine enough grid, the error becomes so small that it is negligible in comparison to the noise that already exists on the data. The choice of the evaluation points inside the grid intervals is thus left to the convenience of the sampling algorithm.

StochS can now be defined as an average of the parameterized functions $f(x;\mathbf{F},\mathbf{x})$ given both the data $\mathbf{g}$ and the grid $\mathbf{x}$. The averaging is carried over all model integrals $\mathbf{F}$ weighted by their posterior probabilities

$$f_{\text{StochS}}(x;\mathbf{x}) = \int d\mathbf{F}\ P(\mathbf{F}|\mathbf{g},\mathbf{x})\ f(x;\mathbf{F},\mathbf{x})\ . \tag{3.96}$$

where the posterior probability reads

$$P(\mathbf{F}|\mathbf{g},\mathbf{x}) = \frac{P(\mathbf{g}|\mathbf{F},\mathbf{x})P(\mathbf{F}|\mathbf{x})}{P(\mathbf{g}|\mathbf{x})}\ . \tag{3.97}$$

This is the same as Bayes' rule of Eq. 3.7 but with probabilities written conditional on the grid points $\mathbf{x}$. The prior probability is as earlier: flat for all non-negative model integrals regardless of the grid points

$$P(\mathbf{F}|\mathbf{x}) \propto \begin{cases} 1 \text{ if } \mathbf{F} > 0 \\ 0 \text{ otherwise} \end{cases} . \tag{3.98}$$

The likelihood is still a Gaussian function

$$P(\mathbf{g}|\mathbf{F},\mathbf{x}) \propto \exp\left[-\frac{1}{2}\chi^2(\mathbf{F},\mathbf{x})\right] \tag{3.99}$$

where the data corresponding to the model $f(x;\mathbf{F},\mathbf{x})$ and its fit $\chi(\mathbf{F},\mathbf{x})$ can be approximated as discussed earlier. Finally, the marginal likelihood is the probability of the data $\mathbf{g}$ given grid $\mathbf{x}$ irrespective of the model integrals $\mathbf{F}$ (i.e. the likelihood marginalized over the model integrals)

$$P(\mathbf{g}|\mathbf{x}) = \int d\mathbf{F}\ P(\mathbf{g}|\mathbf{F},\mathbf{x})P(\mathbf{F}|\mathbf{x}) . \tag{3.100}$$

Since this is independent of the model integrals, it does not affect the averaging. Nevertheless, it is an important quantity in selecting the grid as discussed in the next section. Putting things together, the StochS solution is given by

$$f_{\text{StochS}}(x;\mathbf{x}) = \frac{1}{C}\int_{\mathbf{F}\geq 0} d\mathbf{F}\ e^{-\chi^2(\mathbf{F},\mathbf{x})/2}\ f(x;\mathbf{F},\mathbf{x}) , \tag{3.101}$$

which shows explicitly the grid points as parameters to be selected.

### 3.3.6. Grid selection

The Bayesian way of dealing with the grid dependence, as with any other inference problem, is to put a prior distribution over the grids and get a posterior probability for each grid. We can then either use the grid with the maximum posterior probability (maximum estimator)

$$f_{\text{StochS}}(x;\mathbf{x}^\star) \text{ where } \mathbf{x}^\star = \arg\max_{\mathbf{x}} P(\mathbf{x}|\mathbf{g}) \tag{3.102}$$

or average over different grids weighted by their posterior probability (mean estimator)

$$\int d\mathbf{x}\ P(\mathbf{x}|\mathbf{g})\ f_{\text{StochS}}(x;\mathbf{x}) . \tag{3.103}$$

The averaging solution is discussed in the next section and implemented as a gridless method. The maximum solution would require a non-linear optimization algorithm to find the "optimal" grid with the maximum posterior. Instead, we will use the posterior probability to derive a heuristic for comparing the quality of grids and use it in a recipe for selecting StochS grid.

The posterior probability of a grid using Bayes rule reads

$$P(\mathbf{x}|\mathbf{g}) = \frac{P(\mathbf{g}|\mathbf{x})P(\mathbf{x})}{P(\mathbf{g})} , \tag{3.104}$$

and the posterior odds of some gird $\mathbf{x}_1$ relative to another grid $\mathbf{x}_2$ are

$$\frac{P(\mathbf{x}_1|\mathbf{g})}{P(\mathbf{x}_2|\mathbf{g})} = \frac{P(\mathbf{x}_1)}{P(\mathbf{x}_2)} \frac{P(\mathbf{g}|\mathbf{x}_1)}{P(\mathbf{g}|\mathbf{x}_2)} \,. \tag{3.105}$$

Assuming no prior knowledge for favoring one grid over the other, we set $P(\mathbf{x}_1)/P(\mathbf{x}_2) = 1$. The following ratio of marginal likelihoods (see Eq. 3.100), known as *Bayes Factor* [32], can then be used to compare grids

$$B_{12} \coloneqq \frac{P(\mathbf{g}|\mathbf{x}_1)}{P(\mathbf{g}|\mathbf{x}_2)} = \frac{\int d\mathbf{F}\; P(\mathbf{g}|\mathbf{F},\mathbf{x}_1)P(\mathbf{F}|\mathbf{x}_1)}{\int d\mathbf{F}\; P(\mathbf{g}|\mathbf{F},\mathbf{x}_2)P(\mathbf{F}|\mathbf{x}_2)} \,, \tag{3.106}$$

where a value larger than one implies that grid $\mathbf{x}_1$ is more favorable than grid $\mathbf{x}_2$, and vice versa. To simplify the Bayes factor expression, we rewrite the marginal likelihood as the *posterior* harmonic mean of the likelihood i.e. the reciprocal of the expectation value of the reciprocal of the likelihood

$$\begin{aligned}
& \int d\mathbf{F} P(\mathbf{F}|\mathbf{x}) = 1 && \text{(using Eq. 3.4)} \\
\Rightarrow\quad & \int d\mathbf{F}\; \frac{P(\mathbf{g}|\mathbf{x})P(\mathbf{F}|\mathbf{g},\mathbf{x})}{P(\mathbf{g}|\mathbf{F},\mathbf{x})} = 1 && \text{(using Eq. 3.97)} \\
\Rightarrow\quad & P(\mathbf{g}|\mathbf{x}) \int d\mathbf{F}\; \frac{P(\mathbf{F}|\mathbf{g},\mathbf{x})}{P(\mathbf{g}|\mathbf{F},\mathbf{x})} = 1 \\
\Rightarrow\quad & P(\mathbf{g}|\mathbf{x}) = \mathrm{E}_{P(\mathbf{F}|\mathbf{g},\mathbf{x})}[1/P(\mathbf{g}|\mathbf{F},\mathbf{x})]^{-1} \,.
\end{aligned}$$

Substituting back in Bayes factor (Eq. 3.106) and using Eq. (3.99), we get

$$B_{12} = \frac{P(\mathbf{g}|\mathbf{x}_1)}{P(\mathbf{g}|\mathbf{x}_2)} = \frac{\mathrm{E}_{P(\mathbf{F}|\mathbf{g},\mathbf{x}_2)}[1/P(\mathbf{g}|\mathbf{F},\mathbf{x}_2)]}{\mathrm{E}_{P(\mathbf{F}|\mathbf{g},\mathbf{x}_1)}[1/P(\mathbf{g}|\mathbf{F},\mathbf{x}_1)]} = \frac{\mathrm{E}_{P(\mathbf{F}|\mathbf{g},\mathbf{x}_2)}[e^{+\chi_2^2/2}]}{\mathrm{E}_{P(\mathbf{F}|\mathbf{g},\mathbf{x}_1)}[e^{+\chi_1^2/2}]} \tag{3.107}$$

where $\chi_1$ and $\chi_2$ are the data fits using grids $\mathbf{x}_1$ and $\mathbf{x}_2$, respectively. Estimating these expectation values from StochS samples is possible in theory, but fails in practice [33]. The reason is that StochS produces few samples with large $\chi$ where $e^{\chi^2/2}$ is the largest. Therefore, we need a different criterion of judging the quality of StochS results on different girds.

Instead, we found that $P(\chi)$, the distribution of the fit of the models sampled by StochS, to be a reliable heuristic. Lower values of $\chi$ mean a better fit to the data, so the more this distribution is shifted to the left, the more the grid is supported by the data. For example in Fig. 3.20, we show the fit histograms of a Gaussian grid and a Lorentzian grid both with width 4 and 128 points. The fits for the Lorentzian grid are systematically worse than for the Gaussian one, so the data clearly supports the Gaussian grid in comparison with the Lorentzian one. This agrees with what one would choose by directly comparing the results with the original model.

In Fig. 3.21, we also show the fit histograms of the above grid densities with increasing grid size $n$. As the grid size increases above a certain threshold, the histograms in both

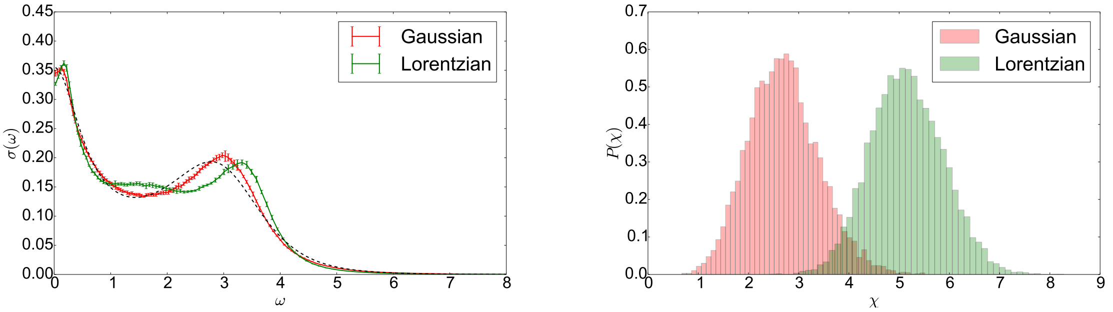
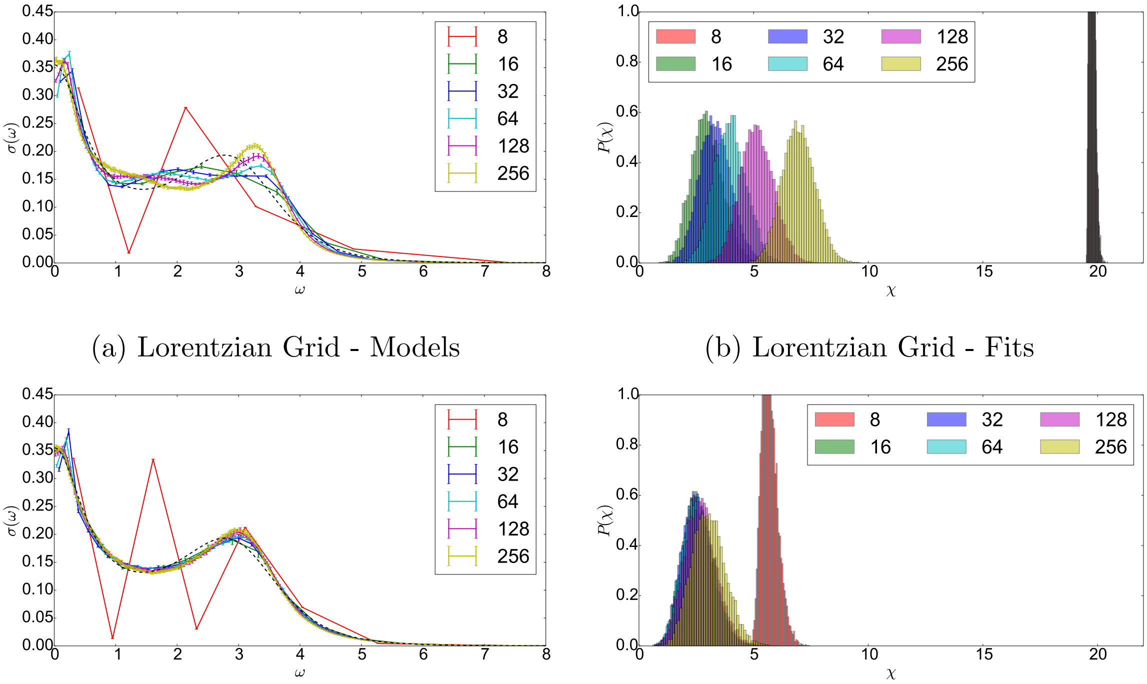


Figure 3.20.: StochS results for the test case 3 using Lorentzian and Gaussian grids, both of width parameter $\gamma = \sigma = 4$ and size $n = 128$. On the left, we show the original model (dashed black) vs. the averaged models (colored). On the right, we show the histograms of the fits of the sampled models. The samples of the Gaussian gird have substantially lower fits than the ones of the Lorentzian grid, thus the earlier is favorable by the data.



(a) Lorentzian Grid - Models

(b) Lorentzian Grid - Fits

(c) Gaussian Grid - Models

(d) Gaussian Grid - Fits

Figure 3.21.: StochS results for test case 3 using a Lorentzian grid (top) and a Gaussian grid (bottom) of width 4 and different sizes (labels). As the grid size increases, the histograms shift to the right and get broader. Therefore, lower grid sizes are favorable by the data. Notice also that the histograms of the Gaussian grid shift much slower than those of the Lorentzian grid. This weaker dependence on the grid size shows that the Gaussian grid is less biased than the Lorentzian grid and thus is a better choice.

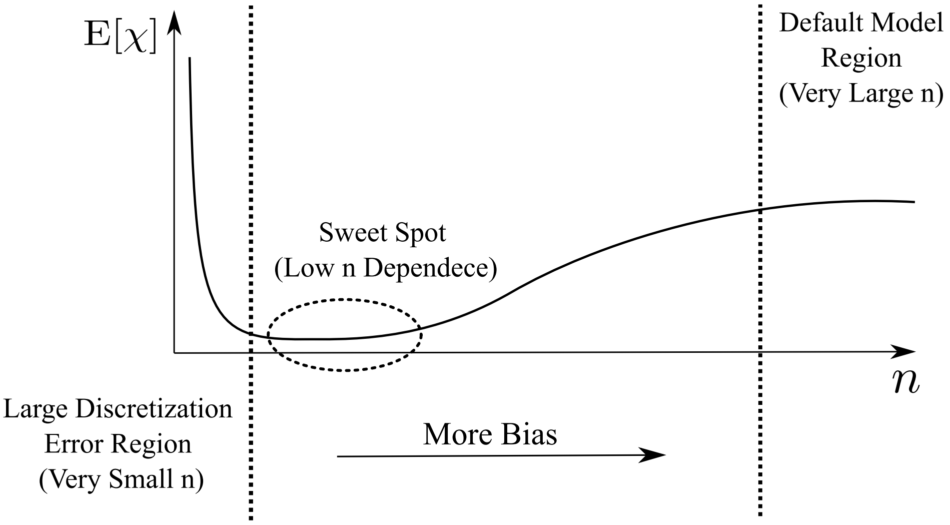

E[$\chi$]
Default Model Region (Very Large n)
Sweet Spot (Low n Dependece)
$n$
Large Discretization Error Region (Very Small n)
More Bias

Figure 3.22.: Schematic diagram of the mean fit behavior against the grid size. For very low sizes, the discretization error is dominating leading to a very bad fit. Once the grid size is large enough such that the discretization error becomes negligible relative to the noise on the data, increasing the grid size leads to more bias (worse fit and more smoothing). This dependence on the grid size starts out slowly and then accelerates till the average model approaches the default model (grid density) for very large sizes. Therefore, a lower dependence on the grid size indicates a better grid density, and when the density matches the exact model, the sweet spot extends to infinity.

cases shift to the right and get broader. According to the above criterion, this means that the data favors the lowest grid size (as long as the discretization error is negligible). This makes sense because the grid size acts as the strength of a default model (the grid density) and thus increasing the grid size biases the results towards the default model and away from the data (see Sec. 3.3.4). In the limit of $n \to \infty$, the result approaches the default model. Moreover, notice how fit histograms get worse faster for the Lorentzian grid than for the Gaussian one. Therefore, the dependence of the fit histogram on the grid size can also be used as another heuristic for selecting the grid density. Better densities have lower dependence on the grid size and vice versa. This behavior of the fit against the grid size is summarized in Fig. 3.22 and it is manifested even more pronouncedly in the following test case.

**Caution** It is good to remember that comparing the fit histograms of different grids is just a *heuristic*. Since it does not take into account prior probabilities of the grids, one should be careful not to use it blindly. For example, a grid whose points lie exactly at the peaks of the NNLS solution would be favorable by this heuristic over any other grid because it would overfit the data (see Fig. 3.23 and Fig. 3.24). Therefore, this criterion should only be used to compare ”reasonable” girds and guide us only in choosing between grids that are a priori equally probable.

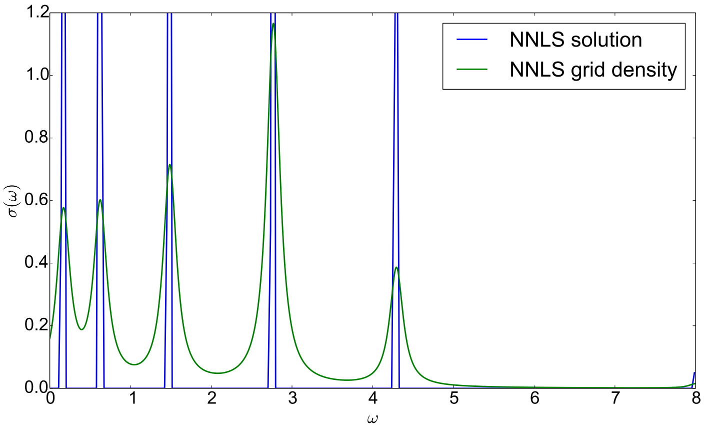


Figure 3.23.: NNLS solution of test case 3 obtained on a fine uniform grid from 0 to 8. We can use this solution to build a grid density for StochS by replacing its peaks with Lorentzians of fixed half-width (here 0.1), and whose positions and weights match the positions and weights of the peaks.

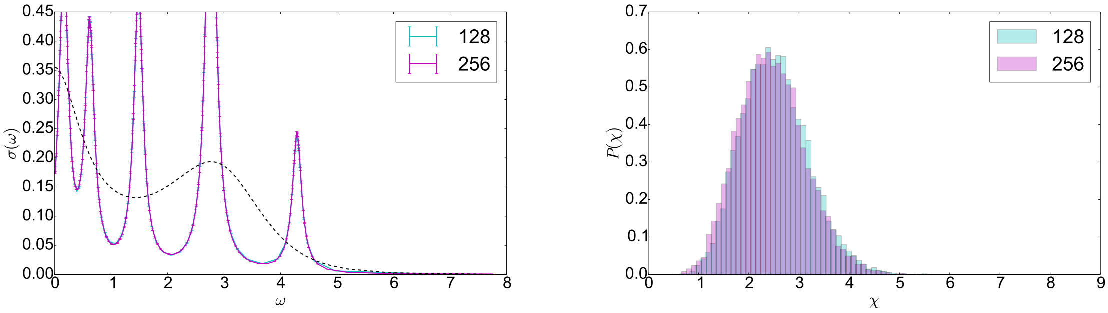


Figure 3.24.: StochS results for test case 3 using NNLS grid density (see Fig. 3.23) and different grid sizes (labels). Looking at the fit histograms only, this grid density appears superior over all other grids (compare with Fig. 3.20). However, comparing the results with the exact model, this grid is completely unacceptable. This is a reminder that the heuristic of fit histograms should be used only to compare a priori reasonable grids. Otherwise, it can lead to overfitting the data.

**Test case 4** In this test case, we consider the analytic continuation of a fermionic imaginary-time Green function. The model is simply a Gaussian of width 0.5 centered at zero. For computing the data, the integrals are discretized on a uniform grid extending from $-20$ to 20 with 8000 points. Green function values are generated at 60 equally-spaced $\tau$ points in the interval $[0, \beta]$ with $\beta = 50$. No noise is added to the data. In Fig. 3.25, we show StochS results using a Gaussian grid of width 2. These results have spurious features similar to the cutoff effect of test case 3, and the strong dependence of the fit histogram on the grid size indicates that something is wrong with the grid.

Reducing the width of the Gaussian grid density reduces the dependence on the grid size till the grid density matches that of the exact model. In this extreme case, the fit histograms shown in Fig. 3.25f are almost identical for different grid sizes.

**Grid recipe** Motivated by the previous discussion, we propose the following recipe for selecting the StochS grid. We use a grid whose density type is featureless e.g. a Gaussian, an exponential or a Lorentzian grid. These densities are characterized by their center and width parameters. The center is chosen at zero because it is where the data provides most of the information. To determine a reasonable starting value for the width, we utilize the non-negative least squares solution. This solution is composed of few sparse non-zero values (sharp peaks). Their positions can be considered as a sample drawn from our chosen grid density and used to estimate its width parameter. A widely used method for estimating the parameters of a density function is the maximum likelihood estimator (MLE). For example, the MLE of the width of a Gaussian density is the standard deviation of the sample

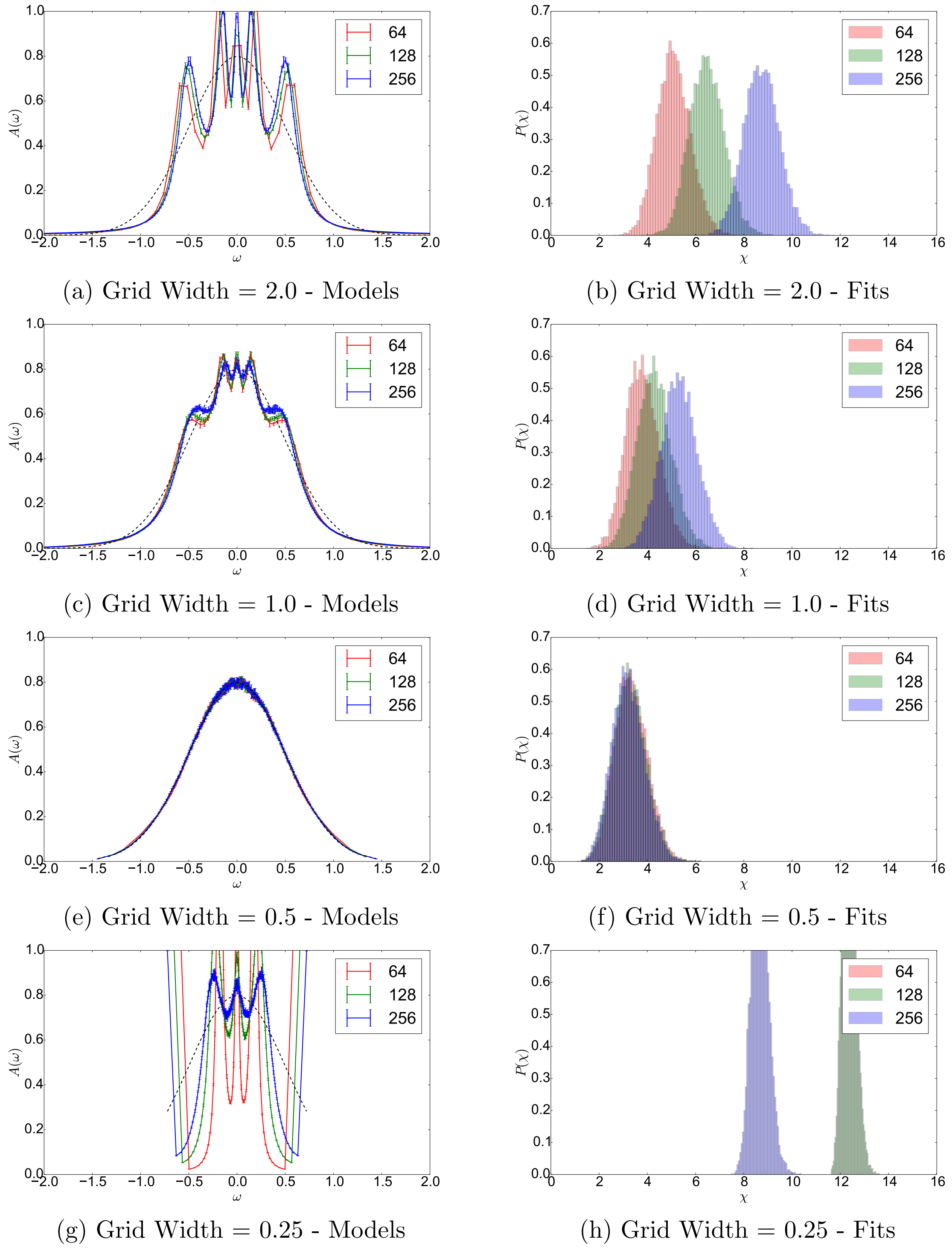
$$\hat{\sigma} = \sqrt{\frac{\sum_i F_i x_i^2}{\sum_i F_i}} \ , \tag{3.108}$$

and the MLE of the width of an exponential density is the mean absolute deviation

$$\hat{\lambda} = \frac{\sum_i F_i |x_i|}{\sum_i F_i} \ , \tag{3.109}$$

where $F_i, x_i$ are the weights and positions of the non-negative least squares peaks. Of course, obtaining the NNLS solution needs itself a grid. Fortunately, this solution is mostly grid independent. For large and fine enough grids, the positions of most peaks are stable. The only exception is the last peak (usually with very small weight) and also the first peak when the model extends to $-\infty$. These outliers represent the integral of the model's tail and are sensitive to the noise on the data. Moreover, they usually sit on the last and the first grid points respectively, and thus their locations depend on the cutoff of the NNLS grid. We exclude these peaks from estimating the width in Eqs. (3.108) and (3.109) to make it more reliable.

The above recipe provides us with a very good initial width estimate. For example, the width estimate for a Gaussian in test case 4 ranges between $0.35 - 0.52$ depending on the noise vector added to the data (Remember that the exact width is 0.5). To make sure that the estimated width is reliable, we apply StochS using not only one grid size $n$ but rather different grid sizes (usually powers of two). If the results and the fit histograms are relatively stable with increasing grid size, we can trust the results. Otherwise, we vary the width and choose the one for which the fit is the lowest and least dependent on the grid size. We could also try different types of grid densities and compare their dependence on the grid size. Usually this is not necessary, however, since the grid width is the most important factor affecting the results.



(a) Grid Width = 2.0 - Models

(b) Grid Width = 2.0 - Fits

(c) Grid Width = 1.0 - Models

(d) Grid Width = 1.0 - Fits

(e) Grid Width = 0.5 - Models

(f) Grid Width = 0.5 - Fits

(g) Grid Width = 0.25 - Models

(h) Grid Width = 0.25 - Fits

Figure 3.25.: StochS results for test case 4 using Gaussian grids of different widths (captions) and different sizes (labels). For the largest width, the results have spurious features and their fit histograms have a strong dependence on the grid size. As the width decreases, the spurious features and the grid size dependence become weaker till they disappear when the grid width matches the exact one 0.5. When the width is lowered even further, the discretization error becomes extremely large and the fit is very bad.

## 3.4. Gridless stochastic sampling (gStochS)

Instead of trying to find the grid with the maximum posterior probability, we can average over the results of different grids weighted by their posterior probability

$$\int d\mathbf{x}\ P(\mathbf{x}|\mathbf{g})\ f_{\text{StochS}}(x;\mathbf{x})\ . \tag{3.110}$$

Using Bayes rule for the grid (Eq. 3.104), this can be computed by proposing grids from their prior distribution $P(\mathbf{x})$ and averaging StochS results weighted by the marginal likelihood $P(\mathbf{g}|\mathbf{x})$

$$\frac{1}{P(\mathbf{g})}\int d\mathbf{x}\ P(\mathbf{x})\ P(\mathbf{g}|\mathbf{x})\ f_{\text{StochS}}(x;\mathbf{x})\ . \tag{3.111}$$

As we saw earlier, computing the marginal likelihood is very hard in practice. Fortunately, we do not need it, because it can be canceled out with the normalization constant of StochS as following

$$\begin{aligned}
&\frac{1}{P(\mathbf{g})}\int d\mathbf{x}\ P(\mathbf{x})\ P(\mathbf{g}|\mathbf{x})\ f_{\text{StochS}}(x;\mathbf{x})\\
=&\frac{1}{P(\mathbf{g})}\int d\mathbf{x}\ P(\mathbf{x})\ P(\mathbf{g}|\mathbf{x})\ \int d\mathbf{F}\ P(\mathbf{F}|\mathbf{g},\mathbf{x})\ f(x;\mathbf{F},\mathbf{x}) && \text{(using Eq. 3.96)}\\
=&\frac{1}{P(\mathbf{g})}\int d\mathbf{x}\ P(\mathbf{x})\ \cancel{P(\mathbf{g}|\mathbf{x})}\ \int d\mathbf{F}\ \frac{P(\mathbf{g}|\mathbf{F},\mathbf{x})P(\mathbf{F}|\mathbf{x})}{\cancel{P(\mathbf{g}|\mathbf{x})}}\ f(x;\mathbf{F},\mathbf{x}) && \text{(using Eq. 3.97)}\\
\propto&\int d\mathbf{x}\ P(\mathbf{x})\int_{\mathbf{F}\geq 0} d\mathbf{F}\ e^{-\chi^2(\mathbf{F},\mathbf{x})/2}\ f(x;\mathbf{F},\mathbf{x})\ .
\end{aligned}$$

The price we have to pay, however, is developing a new sampling algorithm where each sample is composed of both the grid points and the model integrals. This is discussed in the next section.

A simple and straightforward prior for the grid points is $P(\mathbf{x}) = \prod_i p(x_i)$ where grid points are drawn identically and independently from some density function $p(x)$. Note that a valid gird requires an ordering of its points which is not guaranteed by this prior. Nevertheless, we can impose the ordering implicitly inside the mapping $f(x;\mathbf{F},\mathbf{x})$. Substituting back in the last integral, we get

$$f_{\text{gStochS}}(x) = \frac{1}{C}\int d\mathbf{x}\ \prod_i p(x_i)\int_{\mathbf{F}\geq 0} d\mathbf{F}\ e^{-\chi^2(\mathbf{F},\mathbf{x})/2}\ f(x;\mathbf{F},\mathbf{x})\ , \tag{3.112}$$

where $C$ normalizes the expression. We call this sampling over grids with such uncorrelated prior: *gridless stochastic sampling (gStochS)* .

**Comparison with StochS** It is important to emphasize the difference between the prior density function $p(x)$ in gStochS and the grid density function $\rho(x)$. The grid points in

StochS are fixed and distributed according to $\rho(x)$ with or without data. In contrast, the grid points in gStochS are allowed to move and they follow the prior density $p(x)$ only in the absence of data. As explained later, the data in gStochS can override this prior and average over appropriate grid densities. Interestingly, since each StochS grid corresponds to a gamma process, we can also see gStochS as an average over the default model of gamma process. In Appendix A, we derive the discrete exponential process as the stochastic process (integration measure) corresponding to gStochS.

**Comparison with Beach's delta sampling** By using delta functions in the mapping $f(x;\mathbf{F},\mathbf{x})$, with model integrals representing the weights and grid positions representing the shifts

$$f(x;\mathbf{F},\mathbf{x}) = \sum_i F_i \delta(x - x_i), \tag{3.113}$$

gStochS is technically equivalent to the delta function walker scheme suggested by Beach in [31]. Nevertheless, there are several conceptual differences. First, Beach presents his method as an extension of StochS with "an additional degree of freedom that turns out to be equivalent to default model." This fails to recognize that StochS already has a similar degree of freedom represented by the grid density, which even has a stronger effect on the results than the prior denisty of gStochS. Second, he derives the method as a discretization of some integration measure. As shown in the appendix, this integration measure corresponds to a multinomial process whose discretization leads to a set of delta functions with varying shifts and *fixed weights*. The actual integration measure corresponding to this method is derived in the appendix. Finally, Beach uses a fictitious temperature parameter and chooses the value corresponding to a jump in the specific heat. The use of such a parameter is not justified in the Bayesian framework. Given the correct covariance matrix of the data noise, this parameter should be set to one [3]. Otherwise, we are risking over- or under-fitting the data.

**Comparison with Mishchenko's stochastic optimization** Using constant functions in the mapping $f(x;\mathbf{F},\mathbf{x})$, the models sampled by gStochS can be seen as a set of contiguous rectangles with $\mathbf{F}$ representing weights and $\mathbf{x}$ representing the positions

$$f(x) = \sum_i \frac{F_i}{x_{i+1} - x_i} \text{ rect} \left( \frac{x - x_i}{x_{i+1} - x_i} \right) .$$

This is comparable to the stochastic optimization method by Mishchenko [34, 35], where the model is represented as a sum of independent rectangles

$$f(x) = \sum_i h_i \,\text{rect} \left( \frac{x - x_i}{w_i} \right) ,$$

with $x_i, h_i$ and $w_i$ representing the position, height and width of each rectangle, respectively. The basic idea of Mishchenko's method is to average several models that fit the data well enough, but do not overfit it. Finding those "good" models is done by minimizing a deviation measure which is taken as the $L_1$-norm of the difference between the actual data and the data produced by the model. This method can be recast in Bayesian, albeit strange, terms as following: The likelihood is set to a constant for models that have deviation measures below a certain threshold and zero otherwise. In contrast to the Gaussian likelihood, this likelihood is not justified by any knowledge about the noise on the data. The prior is hard to specify because of the technicalities applied during the updates. However, in its simplest form, the model parameters $x_i, h_i$ and $w_i$ are sampled from flat distributions with cutoffs that are hyper-parameters of the method. Moreover, the sampling in this method is quite inefficient leading to a huge computational cost.

### 3.4.1. Sampling algorithm

The multidimensional integral of gStochS (Eq. 3.112) is evaluated using a Monte Carlo sampling algorithm. We start from some initial model on an initial grid. We use a StochS grid with density $p(x)$ and $n$ points as the initial grid and the perturbed data sampling (PDS) solution as the initial model.

The model integrals are then updated on the current grid as done in StochS. Given the model integrals, the grid points are updated one at a time using a Metropolis-Hastings algorithm explained below. All samples of $\mathbf{F}$ and $\mathbf{x}$ are stored during the sampling, and the average model $f_{\text{gStochS}}(x)$ can be evaluated later at any point $x$ by evaluating each sampled model $f(x; \mathbf{F}, \mathbf{x})$ at this point and averaging the result.

The Metropolis-Hastings algorithm for sampling grid points has the following acceptance probability

$$r = \frac{e^{-\chi^2(x_i')/2}}{e^{-\chi^2(x_i)/2}} \frac{p(x_i')}{p(x_i)} \frac{q(x_i' \to x_i)}{q(x_i \to x_i')} \,, \tag{3.114}$$

where $q(x_i \to x_i')$ is the proposal distribution of moving grid point $i$ from an old position $x_i$ to a new position $x_i'$. In order to get a high acceptance rate, we need a proposal distribution that takes into account both the data factor $e^{-\chi^2/2}$ and the prior density function $p(x)$. Suppose that the data factor can be approximated by a Gaussian of mean $\mu_\chi$ and width $\sigma_\chi$, this Gaussian alone can be a good proposal probability for most cases. However, when a grid point is far way from zero or its weight is very small, the data provides very little information and the prior density $p(x)$ becomes more important. Therefore, we combine the data Gaussian with another Gaussian centered around the old position $x_i$, whose width equals the width $w$ of the density $p(x)$. The product of the two Gaussians is again a Gaussian with mean $\mu = \sigma(\mu_\chi/\sigma_\chi + x_i/w)$ and width $\sigma = 1/(\sigma_\chi^{-1} + w^{-1})$. We use this as our proposal distribution which leads to an efficient sampling of the grid points.

**Kernel evaluation points** Computing the data corresponding to each sample requires evaluating the kernel at some point in each grid interval. A simple choice is using the midpoint. However, this choice leads to some unnecessary computational overhead when the grid points are sampled. Updating one grid point would affect the midpoints of two neighboring intervals and the kernel needs to be reevaluated twice. A convenient solution is to evaluate the kernel at either end point of the interval. We choose the right end point for positive intervals and the left end point for negative ones. This allows us to get a symmetric result around zero when it exists. To avoid ambiguity for an interval extending from negative to positive $x$, we fix an extra gird point at $x = 0$. Note that with the introduction of this extra fixed point, the number of sampled grid points becomes the same as the number of intervals and thus the sizes of the position vector $\mathbf{x}$ and the integrals vector $\mathbf{F}$ are now equal. Moreover, the kernel needs to be evaluated only at the sampled grid points. We can now write the data corresponding to the sample $[\mathbf{F}, \mathbf{x}]$ in vector form as

$$\mathbf{g}_{[\mathbf{F},\mathbf{x}]} = \mathbf{K}(\mathbf{x})\ \mathbf{F}\ . \tag{3.115}$$

Notice how the data vector is a linear function of the model integrals but a non-linear function of the grid points.

**Gaussian approximation of $e^{-\chi^2(\mathbf{x})/2}$:** For efficient sampling, we need a quadratic approximation of the data fit as a function of the grid points. Let us write the data fit as a function of the new grid position $x'_i$

$$\chi^2(x'_i) = \left\| \mathbf{g} - \sum_{j\neq i} \mathbf{K}(x_j)F_j - \mathbf{K}(x'_i)F_i \right\|^2\ . \tag{3.116}$$

Now we expand the kernel vector $\mathbf{K}(x'_i)$ to second-order around the old position $x_i$

$$\mathbf{K}(x'_i) = \mathbf{K}(x_i) + \partial\mathbf{K}(x_i)\ [x'_i - x_i] + \frac{1}{2}\ \partial^2\mathbf{K}(x_i)\ [x'_i - x_i]^2 + \dots \tag{3.117}$$

Substituting back in the fit expression and keeping only terms to the second-oder in $x'_i$, we find

$$\chi^2(x'_i) \approx \mathbf{r}^{\mathrm{T}}\mathbf{r} - 2\left[\mathbf{r}^{\mathrm{T}}\partial\mathbf{K}_i F_i\right][x'_i - x_i] + \left[F_i^2 \partial\mathbf{K}_i^{\mathrm{T}}\partial\mathbf{K}_i - F_i\mathbf{r}^{\mathrm{T}}\partial^2\mathbf{K}_i\right][x'_i - x_i]^2\ , \tag{3.118}$$

where $\mathbf{r} := \mathbf{g} - \sum_j \mathbf{K}(x_j)F_j$ is the old residual vector and $\partial\mathbf{K}_i := \partial\mathbf{K}(x_i)$, $\partial^2\mathbf{K}_i := \partial^2\mathbf{K}(x_i)$. By completing the squares, it can be written in the following suggestive form

$$\chi^2(x'_i) \approx (x'_i - \mu_\chi)^2/\sigma_\chi^2 + \text{const.} \tag{3.119}$$

where $\mu_\chi = x_i + F_i\mathbf{r}^{\mathrm{T}}d\mathbf{K}/\sigma_\chi^2$ and $\sigma_\chi^{-2} = F_i^2 d\mathbf{K}^{\mathrm{T}}d\mathbf{K} - F_i\mathbf{r}^{\mathrm{T}}d^2\mathbf{K}$. As a result, the data fit factor $e^{-\chi^2/2}$ can be approximated at the old position by a Gaussian of mean $\mu_\chi$ and width $\sigma_\chi$. Note that when the fit has negative curvature at the old position $x_i$, the width will be negative and this Gaussian approximation breaks down. In this case, we use as a proposal distribution only the Gaussian centered around the old position $x_i$, whose width equals the width $w$ of the prior density $p(x)$.

**Symmetric cases** For cases where the model extends only from 0 to $\infty$, the Gaussian proposal may give inadmissible negative values. We take the absolute value and make sure that the acceptance ratio contains the correct proposal probability i.e. the probability of proposing both the negative and positive values.

**Moving several points** In hope of having larger sampling steps, we derived the quadratic approximation of the fit as a function of more than one grid point and used it to form a multidimensional Gaussian proposal distribution. However, we found this to give less efficient sampling than single updates. The problem is two-fold. First, updating several points together means that it is more likely to include a point where the quadratic approximation fails. Second, points that are near to each other, give rise to a singular or near-singular covariance matrix of the proposal probability. This leads to very large steps, for which the approximation is not valid anymore.

**Sampling model integrals** The model integrals $\mathbf{F}$ are sampled using blocked modes sampling with a random power-of-two block size. Interlacing of blocks is not necessary here as in StochS because the grid points are moving and there is no danger of effects at the block boundaries. However, the movment of grid points implies that the kernel matrix is changing, so the SVD of its blocks should be recalculated after each update of the grid points. Since SVD is computationally heavy, we try to utilize its result more, by performing several consecutive updates of the model integrals for each sampled grid.

**Averaging and binning** Averaging gStochS samples requires evaluating the mapping $f(x;\mathbf{F},\mathbf{x})$ on some fixed grid. We call it the binning grid to distinguish it from the grids sampled by gStochS. Let us denote its intervals as $\mathcal{B}_i$ and call them bins. The binning would be different depending on the mapping $f(x;\mathbf{F},\mathbf{x})$. Assuming that each gStochS sample represents a set of delta functions with weights $F_i$ and positions $x_i$, the $i$th bin average is computed as

$$f_i \approx \frac{1}{N}\sum_{k=1}^{N}\frac{1}{\mathrm{len}(\mathcal{B}_i)}\sum_{x_j^k\in\mathcal{B}_i} F_j^k\ , \tag{3.120}$$

where $k$ is indexing gStochS samples and $N$ is the total number of samples. Alternatively, we can assume a constant value inside each interval $\mathcal{I}_j^k$ of the $k$th grid sample. This implies that the corresponding model integral $F_j^K$ should be split proportionally among the bins that intersects this interval

$$f_i \approx \frac{1}{N}\sum_{k=1}^{N}\frac{1}{\mathrm{len}(\mathcal{B}_i)}\sum_{j=1}^{n}\frac{\mathrm{len}(\mathcal{B}_i\cap\mathcal{I}_j^k)}{\mathrm{len}(\mathcal{I}_j^k)}\ F_j^k\ . \tag{3.121}$$

The latter type of binning can be thought of as a linear interpolation of the earlier one and thus it leads to a smoother average. Nevertheless, whatever binning we use, the averages are similar when the sampled grids are fine enough (i.e. the grid size $n$ is large enough), and the difference is only visible when they are very coarse. For simplicity,

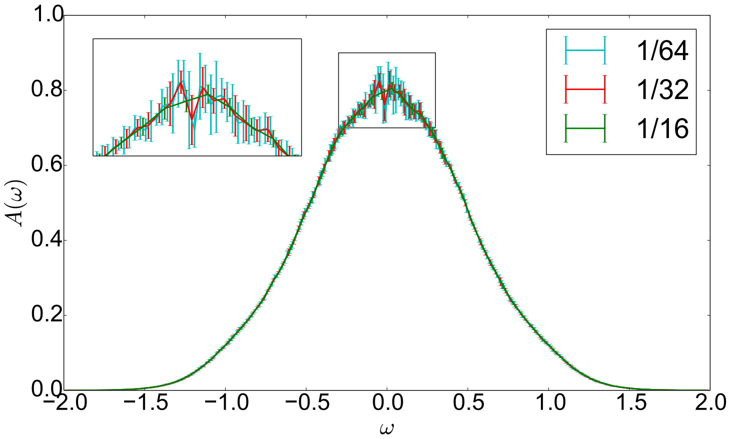


Figure 3.26.: gStochS average results for test case 4 using uniform binning grids of different bin sizes (labels). Increasing the bin size reduces the statistical error in the average.

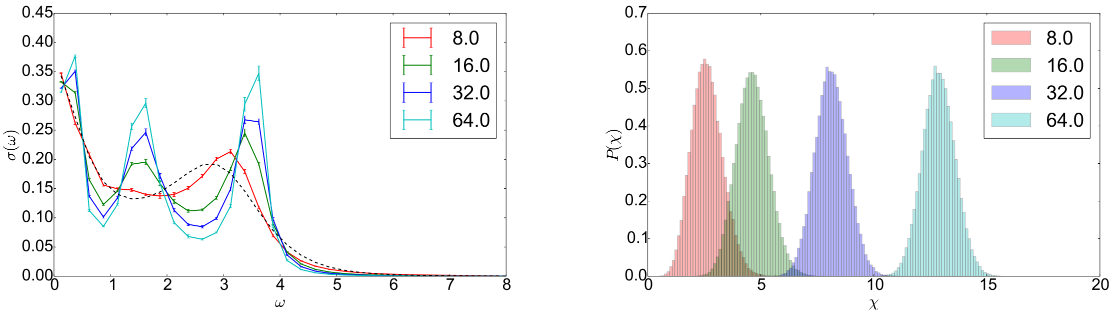
we use the delta binning on a uniform binning grid. Pay attention that the bin size affects the statistical error of its average. Larger bins have lower fluctuations and better averages. Roughly speaking, the error goes down with the square root of the bin size, and sometimes even faster due to negative correlation between neighboring bins. In Fig. 3.26, we show how increasing the bin size reduces the fluctuations in a gStochS average.

### 3.4.2. Density dependence

The cutoff dependence of StochS (see Fig. 3.27) was our primary motivation for studying the grid dependence and developing gStochS, so we would like to check first this effect in gStochS. We use a uniform prior density with cutoffs: 8, 16, 32 and 64. To make the results comparable with those of StochS, we use the same grid sizes as in StochS. The results are shown in Fig. 3.28 and there is almost no effect of the cutoff on the result. In Fig. 3.29, we also show gStochS using Gaussian and Lorentzian prior densities both with width parameter 4 and grid size 128. Unlike StochS, the two densities give almost identical results (compare with Fig. 3.21) .

Do these results show that gStochS is independent of the prior density function $p(x)$? No! The density $p(x)$ still acts as a default model and the grid size still acts as its strength. For example, Fig. 3.30 shows that gStochS gives back the density function in the absence of any data except normalization.

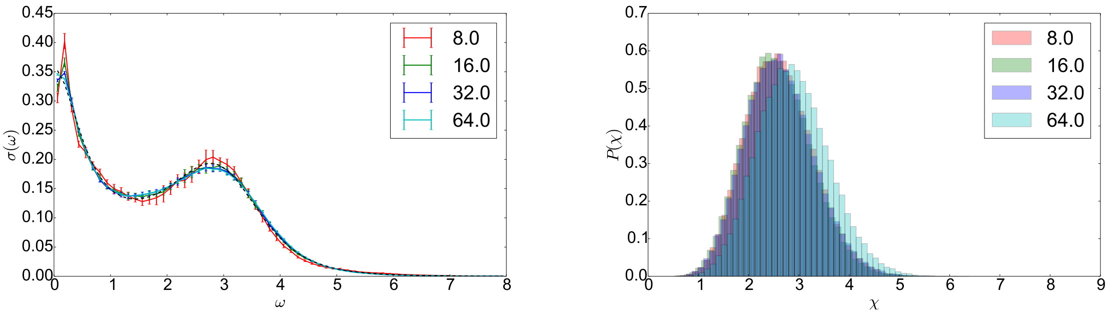
Moreover, there are still test cases where the density function affects the results of gStochS considerably. For example Fig. 3.31 shows gStochS results for test case 4 using a Gaussian prior density of different widths. Like in StochS, using a density with large width leads to spurious features whose strength increases with that width. The effect is, however, weaker in gStochS than in StochS (compare with Fig. 3.25). Notice how the



(a) Models (Fig. 3.11 repeated)

(b) Fits

Figure 3.27.: StochS results of test case 3 using uniforms grid of spacing 0.25 and increasing cutoff (label). As the cutoff increases, spurious features develops and the result gets worse.



(a) Models

(b) Fits

Figure 3.28.: gStochS results of test case 3 using uniform densities of increasing cutoff (label). The grid sizes are respectively: 32, 64, 128 and 256. The results are almost independent of the cutoff (compare to StochS results in Fig. 3.27).

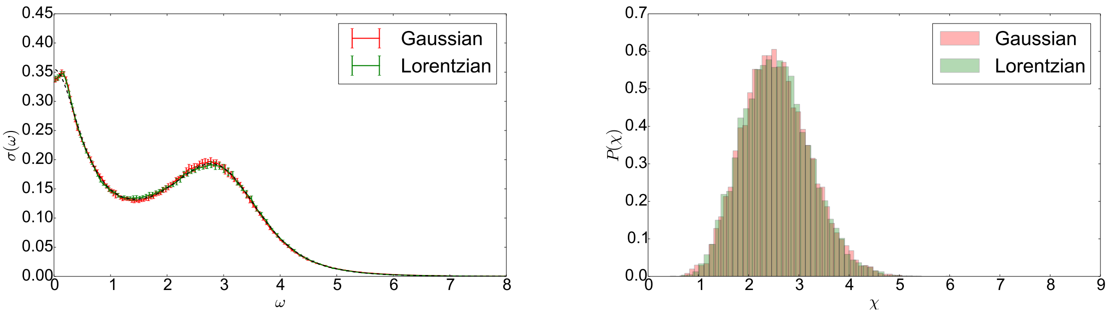


Figure 3.29.: gStochS results of test case 3 using a Gaussian density and a Lorentzian density both of width 4 and size 128. The results are almost identical (compare to StochS results in Fig. 3.21).

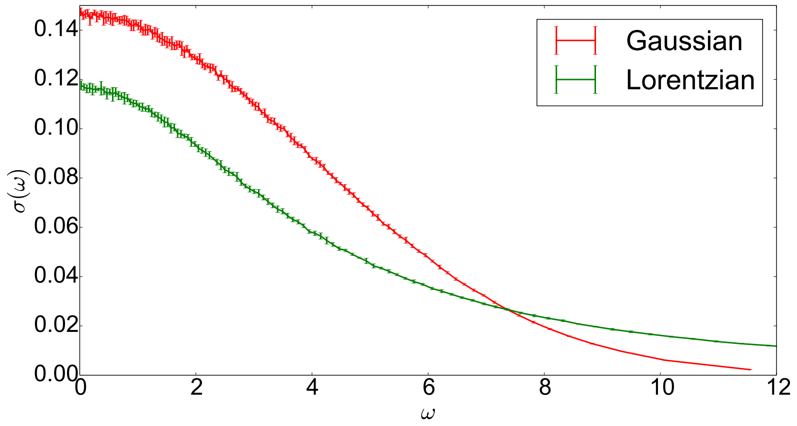
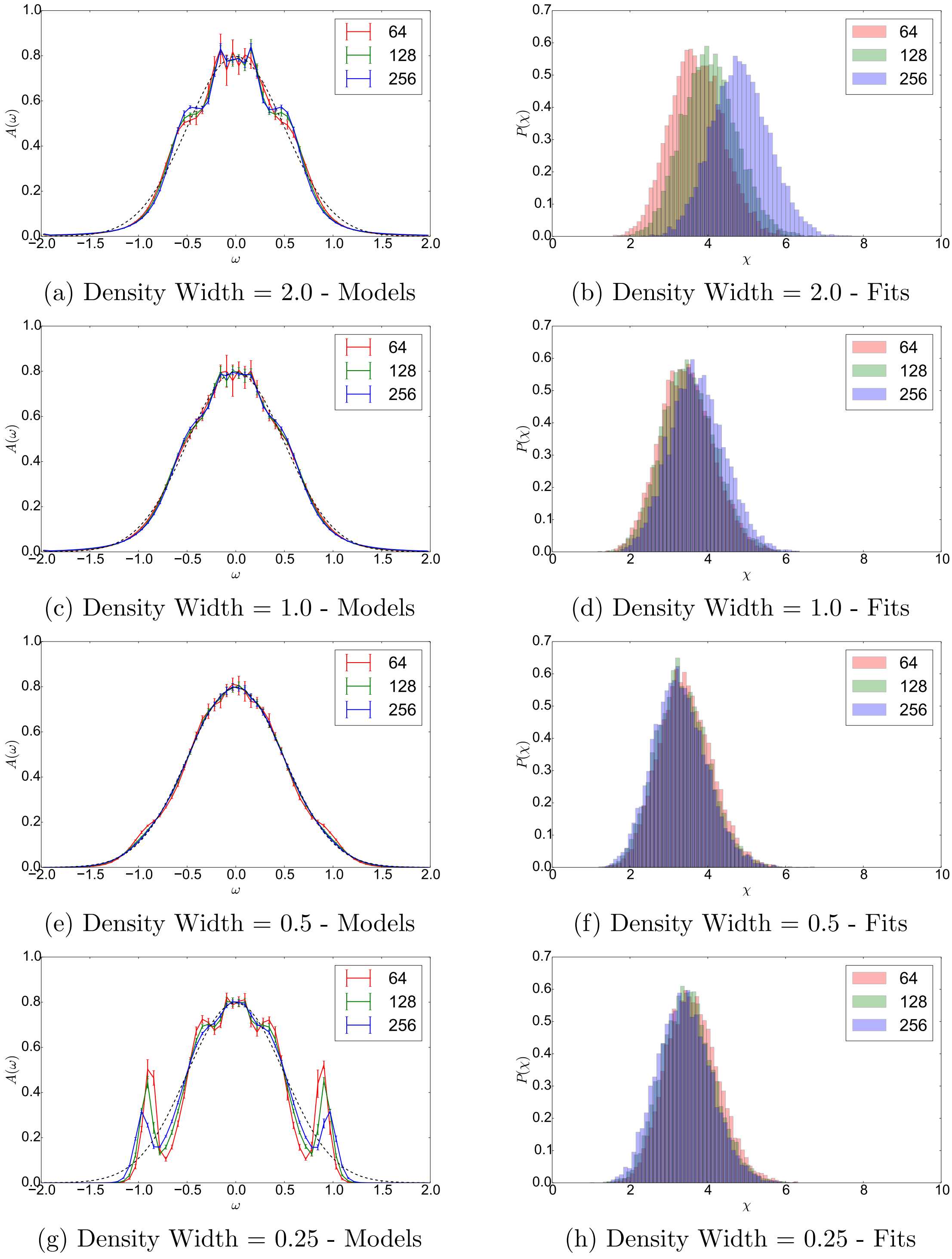


Figure 3.30.: gStochS results of test case 3 using a Gaussian density and a Lorentzian density and only the model normalization as data. In both cases, gStochS gives the prior density function itself confirming that this density acts as a default model.

fit gets better, and the dependence of the histograms on the grid size becomes weaker as the width becomes smaller. The fit histograms for widths 0.25 and 0.5 are almost the same (strictly speaking those of 0.25 are scarcely worse than those of 0.5), so one cannot judge which width is better from the fit alone. However, comparing the results, 0.5 has clearly the lowest dependence on the grid size and thus it is preferable. This is a reminder that the fit histograms are only heuristics for determining which density is better. We should also always check the dependence of the resulting models on the grid size.

In general, we can select the prior density of gStochS as we selected the grid density in StochS. Use some featureless density function centered at zero and estimate its initial width from the NNLS solution. Then vary the width and choose the value for which the dependence of both the results and fit histograms on the grid size is the weakest. This can be repeated for different types of density functions and choosing again the type with the best fit histograms and the least dependence on the grid size.

**Effective grid density** In Fig. 3.32, we show the histogram of the grid positions for test case 3 using a uniform prior density and cutoff 16. Despite the flat prior that extends up to 16, the data guides the grid points to reorganize themselves and move to the important region near zero. Consequently, we see that although both StochS and gStochS have a default model, the latter allows the data to override this prior information leading to less biased results. We can use this histogram as an *effective grid density* in StochS to approximate gStochS as shown in Fig. 3.33. Of course, this is not a practical procedure because we already need the gStochS result to find this effective grid. Nevertheless, the matching results show that this effective grid is the most favorable gird sought by StochS grid recipe.



(a) Density Width = 2.0 - Models

(b) Density Width = 2.0 - Fits

(c) Density Width = 1.0 - Models

(d) Density Width = 1.0 - Fits

(e) Density Width = 0.5 - Models

(f) Density Width = 0.5 - Fits

(g) Density Width = 0.25 - Models

(h) Density Width = 0.25 - Fits

Figure 3.31.: gStochS results for test case 4 using Gaussian prior densities of different widths (captions) and different sizes (labels). Compared to StochS results (Fig. 3.25), the spurious features and the dependence on the grid size is much weaker but still exits.

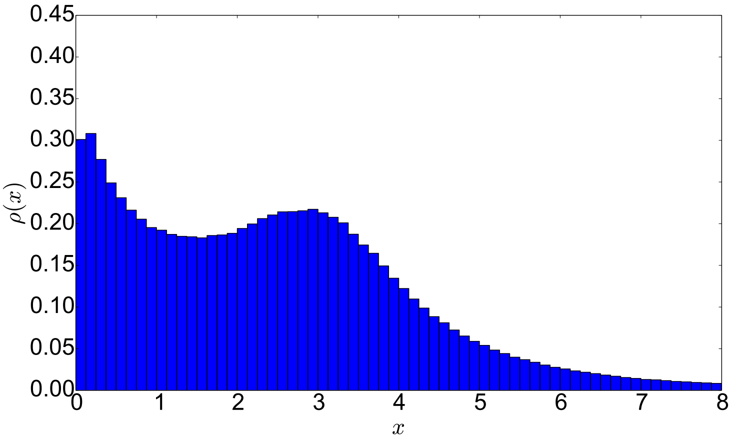


Figure 3.32.: Histogram of the grid positions sampled by gStochS for test case 3 using a uniform density of cutoff 16. This histogram can be thought as an effective grid density to be used in StochS (see Fig. 3.33).

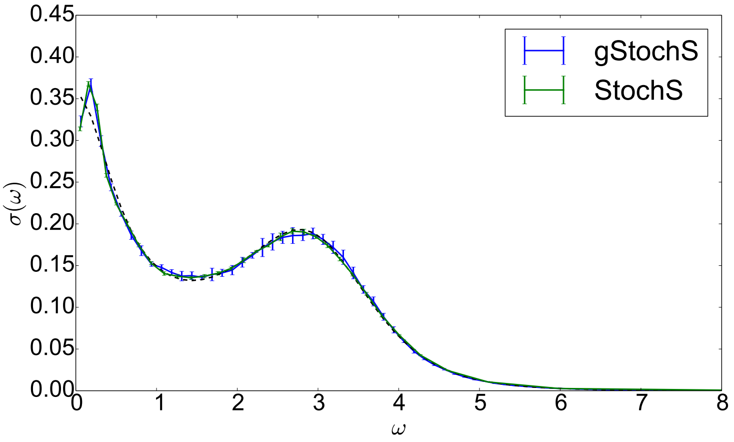


Figure 3.33.: Comparison of gStochS using a uniform prior density against StochS using the corresponding effective grid density (see Fig. 3.32).

## 3.5. Extended Stochastic Sampling (eStochS)

As explained in the last section, gStochS still has a dependence on the prior density $p(x)$. In most cases, this dependence is very weak and any reasonable choice would give equally good results, but sometimes the results still depend noticeably on the chosen density, especially its width $w$. The first option to deal with this dependence is to reuse our StochS grid recipe and choose the width with the best fit histograms and the least dependence on the grid size.

Another option is to average over the width $w$ weighted by its posterior probability

$$\int dw \; P(w|\mathbf{g}) \; f_{\mathrm{gStochS}}(x;w) \; . \tag{3.122}$$

This requires specifying a prior for the width, the simplest of which is a flat distribution. Following steps similar to those between Eq.3.110 and Eq. 3.112, we get

$$\frac{1}{C}\int dw \int d\mathbf{x} \prod_i p(x_i; w) \int_{\mathbf{F}\geq 0} d\mathbf{F}\ e^{-\chi^2(\mathbf{F},\mathbf{x})/2}\ f(x; \mathbf{F}, \mathbf{x})\ . \tag{3.123}$$

One could go about sampling the width directly but it would be inefficient. The reason is that updating the width would change the prior probabilities of all grid points. Therefore, for a large number of grid points, one would be forced to take very small updates to achieve a reasonable acceptance rate. The more grid points, the less efficient the sampling is. There is, however, a much better way.

We notice that unlike the grid points $\mathbf{x}$ and model integrals $\mathbf{F}$, this width parameter $w$ is not directly related to the data so the above expression can be rearranged such that the integral over the width is a function of the grid points only

$$\frac{1}{C}\int d\mathbf{x} \int_{\mathbf{F}\geq 0} d\mathbf{F}\ e^{-\chi^2(\mathbf{F},\mathbf{x})/2}\ f(x; \mathbf{F}, \mathbf{x}) \underbrace{\int dw \prod_i p(x_i; w)}_{:=P(\mathbf{x})}\ . \tag{3.124}$$

We can perform the width integral $P(\mathbf{x})$ analytically for the following family of density functions

$$p(x; w) \propto \frac{1}{w}\exp\left[-\frac{1}{q}\left(\frac{|x|}{w}\right)^q\right] \qquad \text{where} \ \ q > 0\ . \tag{3.125}$$

This is known as the *exponential power distribution* and it includes the Gaussian distribution ($q = 2$), the Laplace (aka double-exponential) distribution ($q = 1$) and the uniform distribution ($q \to \infty$). Using this density, the integral over the width reads

$$P(\mathbf{x}) \propto \int dw\ \frac{1}{w^n}\exp\left[-\sum_i \frac{1}{q}\left(\frac{|x_i|}{w}\right)^q\right] = \int dw\ \frac{1}{w^n}\exp\left[-\frac{1}{q}\frac{\|\mathbf{x}\|_q^q}{w^q}\right]\ , \tag{3.126}$$

where the $L_q$-norm[2] is defined by

$$\|\mathbf{x}\|_q := \left(\sum_i |x_i|^q\right)^{1/q}\ . \tag{3.127}$$

We use this norm to make the following change of variable

$$z := \frac{\|\mathbf{x}\|_q}{w} \Rightarrow \frac{dw}{dz} = -\frac{\|\mathbf{x}\|_q}{z^2}\ , \tag{3.128}$$

and the integral over the new variable is independent of the grid points

$$P(\mathbf{x}) \propto \frac{1}{\|\mathbf{x}\|_q^{n-1}} \int dz\ z^{n-2}\exp\left[-\frac{z^q}{q}\right]\ . \tag{3.129}$$

[2] For $q < 1$, this expression does not define a norm because it violates the triangle inequality. Nevertheless, our results still hold also in that case.

Substituting back in the original expression Eq. (3.124) and absorbing the integral over $z$ in the overall normalization constant we get

$$f_{\text{eStochS}}(x) = \frac{1}{C} \int d\mathbf{x} \; \frac{1}{\|\mathbf{x}\|_q^{n-1}} \int\limits_{\mathbf{F} \geq 0} d\mathbf{F} \; e^{-\chi^2(\mathbf{F},\mathbf{x})/2} \; f(x;\mathbf{F},\mathbf{x}) \; . \tag{3.130}$$

We call this sampling method: *extended stochastic sampling (eStochS)*.

To conclude, using the $L_q$-norm in eStochS is equivalent to using a $q$-th power exponential density function in gStochS and integrating flat over its width parameter. More specifically, using the $L_2$-norm, eStochS is equivalent to using a Gaussian prior density in gStochS and integrating over its standard deviation. Similarly, using the $L_1$-norm is equivalent to using an exponential (or Laplacian) density and integrating over the scale parameter while using the $L_\infty$-norm corresponds to using a uniform density and integrating over the cutoff.

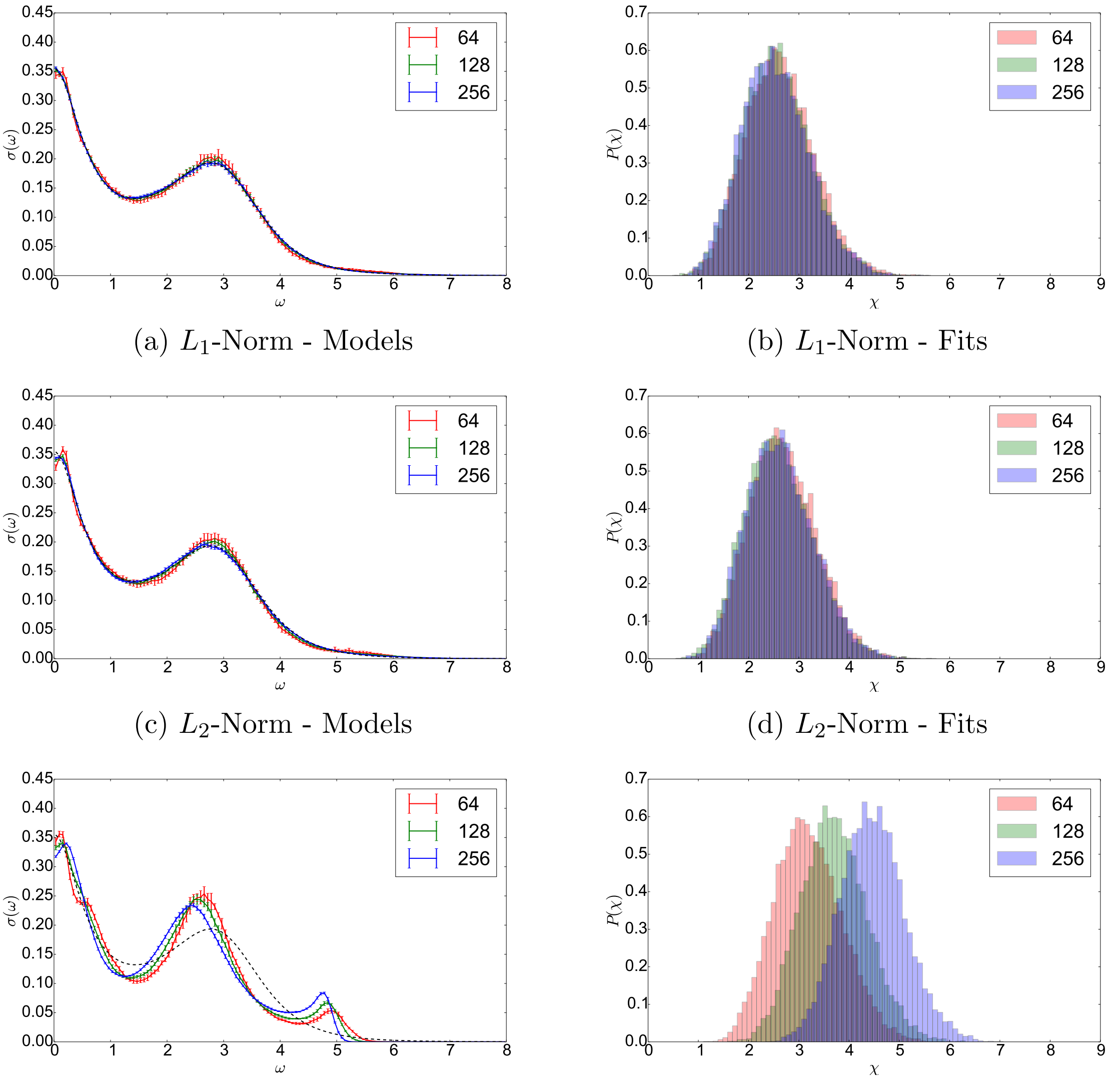
**Sampling algorithm** The sampling algorithm of gStochS can be easily adapted to eStochS. We simply replace the prior density function in the acceptance ratio by the power ratio of the norms of grid samples. We also choose a reasonable value for the width parameter $w$ of the proposal distribution e.g. estimated from the NNLS solution.

**Alternative derivation** The previous discussion derives eStochS by integrating over the width parameter of gStochS. Here we present a different way of arriving to eStochS, which is how we actually found it. Since grid points in gStochS are already free to relocate themselves, one may wonder: wouldn't it be sufficient to take the limit of the width of the prior density to infinity and let the data determine the width of each grid sample (measured by the norm of its vector)? The answer is negative as evident from the bad results of test case 4 using large widths (see Fig. 3.31). The problem lies in the prior distribution of the grid points. Taking the width to infinity, all grid vectors become equally probable. However, there are more vectors with large norm than with small one (this idea was also used in Sec. 2.9). Therefore, the grid samples would almost certainly have a large norm even if the data says otherwise. The solution is to find the density[3] of vectors with a specific norm $r$ and divide by it. For the $L_2$-norm, this density is nothing but the volume of an $n-1$ dimensional hypersphere with radius $r$ and it is proportional to $r^{n-1}$. This gives eStochS, which can now be seen as gStochS with a flat prior over the $L_2$-norm. This new prior is uninformative about the width of a grid sample and thus allows the data to choose it correctly.

## 3.5.1. Results

In Fig. 3.34 and 3.35, we show eStochS results for test cases 3 and 4, respectively. The results for the $L_1$- and $L_2$- norms are quite similar to each other in both cases, while the

[3] Here we are talking about the density of points in an $n$-dimensional space. This should not be confused with the prior density function $p(x)$.



(a) $L_1$-Norm - Models

(b) $L_1$-Norm - Fits

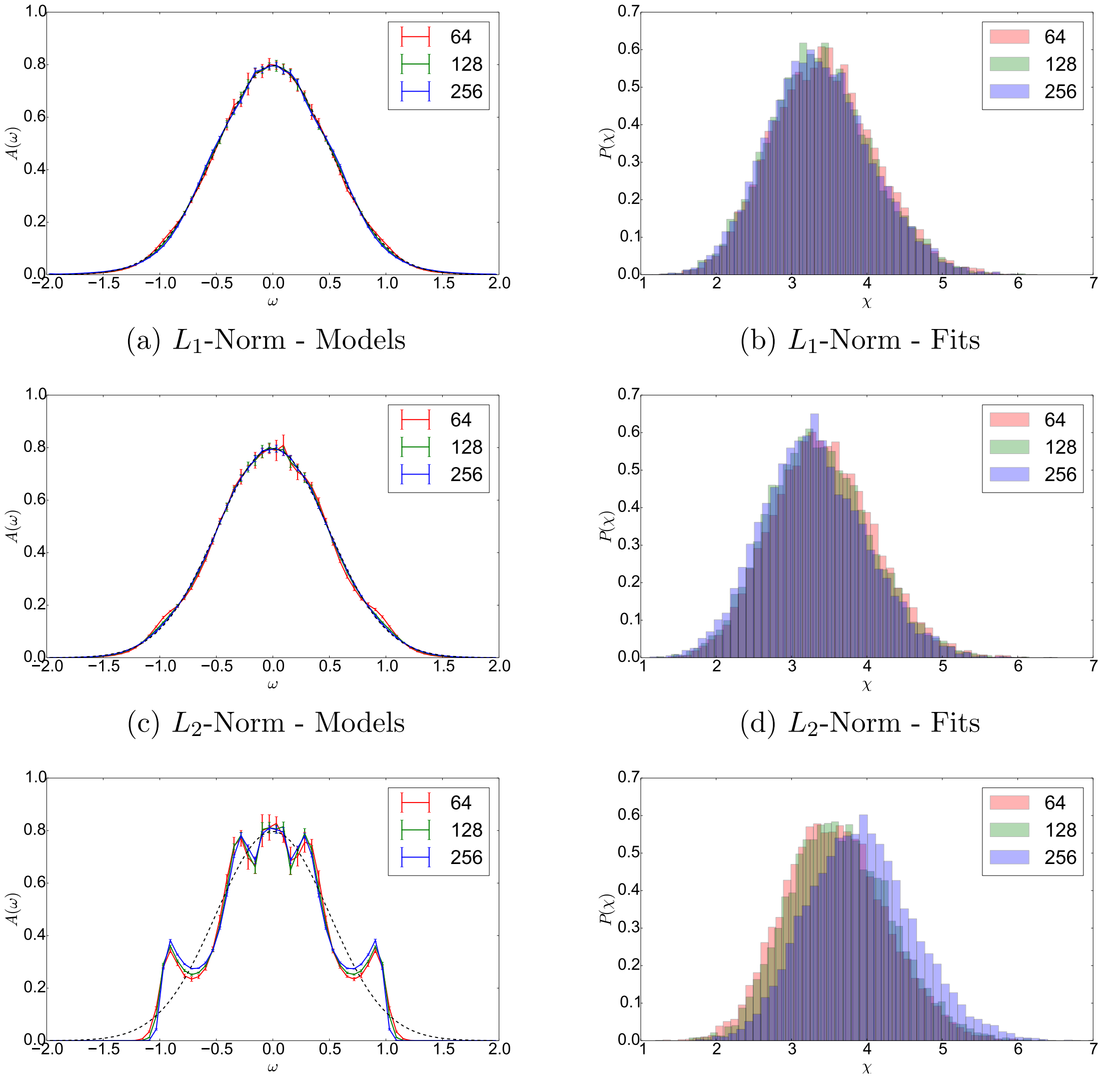
(c) $L_2$-Norm - Models

(d) $L_2$-Norm - Fits

(e) $L_\infty$-Norm - Models

(f) $L_\infty$-Norm - Fits

Figure 3.34.: eStochS results for test case 3 using different norms (captions) and different sizes (labels). We see from the fit histograms that the results using the $L_1$- and $L_2$- norms are reliable, while that of the $L_\infty$-norm is not.



(a) $L_1$-Norm - Models

(b) $L_1$-Norm - Fits

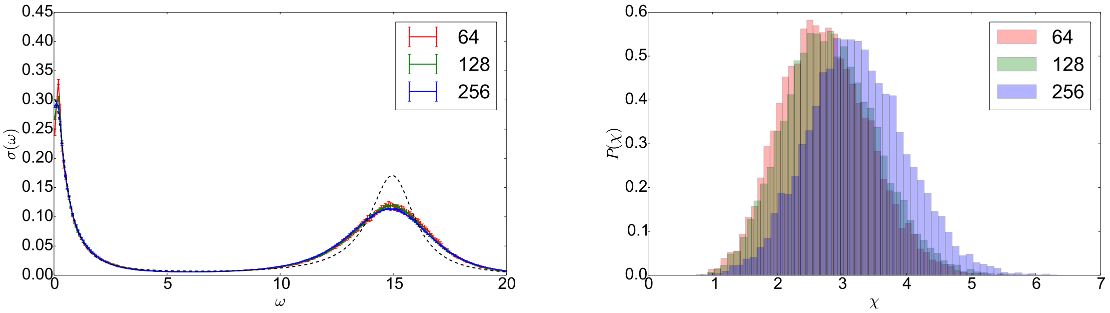
(c) $L_2$-Norm - Models

(d) $L_2$-Norm - Fits

(e) $L_\infty$-Norm - Models

(f) $L_\infty$-Norm - Fits

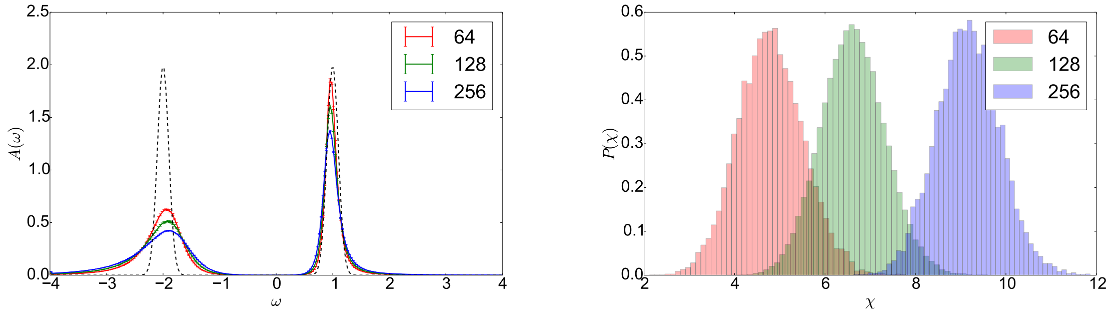
Figure 3.35.: eStochS results for test case 4 using different norms (captions) and different sizes (labels). Here again, the $L_\infty$-norm gives bad results.



(a) Models

(b) Fits

Figure 3.36.: eStochS results for test case 3 using $L_2$-norm and different sizes (labels).



(a) Models

(b) Fits

Figure 3.37.: eStochS results for test case 2 (with Gaussian peaks) using $L_2$-norm and different grid sizes (labels).

$L_\infty$-norm gives bad results. The dependence of the fit histograms for the $L_1$- and $L_2$-norms on the grid size is very weak which indicates a good agreement with the data. On the other hand, the fits for the $L_\infty$-norm get worse faster as the grid size increases indicating a bad choice. We have noticed this trend in all the test cases we performed: $L_1$- and $L_2$- norms give similar results while $L_\infty$-norm gives a much worse result. The bad performance of eStochS using the $L_\infty$-norm is surprising in view of the good results of gStochS using a uniform density (see Fig. 3.28). This is discussed further in Sec. 3.5.2.

The grid size independence of the fit histograms for the $L_1$- and $L_2$-norms is quite impressive. This can be explained by the fact that the exact models are smooth enough such that the exponential or Gaussian default model implied by eStochS agrees very well with the data. On the other hand, when the exact model is less smooth, we would expect a larger dependence on the grid size.

To test this, we shift the second peak of test case 3 to the far right and show the result in Fig. 3.36. Now both the models and the fit histograms show a slight dependence on the grid size as expected. We observe this even more clearly in Fig. 3.37 showing eStochS results of test case 2 (with Gaussian peaks). Due to the large gap in this case, a Gaussian default model does not agree as well with the data as in other test cases. Therefore,

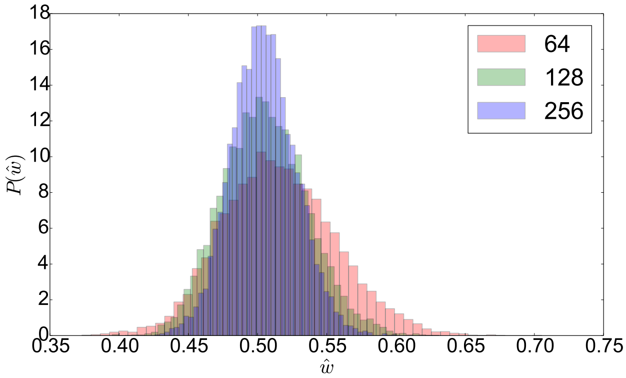


Figure 3.38.: Histograms of the scaled $L_2$-norm of grid samples for test case 4 using eStochS with different grid sizes (labels). The scaled $L_2$-norm of a grid sample $\mathbf{x}$ is calculated as the standard deviation of its points $\hat{w} \coloneqq \sqrt[2]{\|\mathbf{x}\|_2^2/n} = \sqrt{\sum_i x_i^2/n}$. The histograms are centered around the width of the exact model 0.5.

there is a clear dependence on the grid size. As the grid size increases, the two peaks become smoother and the result becomes more biased towards the default model which is a Gaussian covering the gap.

**Effective width** In Fig. 3.38, we show the histogram of the scaled $L_2$-norm of grid samples for test case 4. We calculated the scaled norm of a grid sample as $\hat{w} \coloneqq \sqrt{\sum_i x_i^2/n}$ which approximates the width of the effective grid density. As the grid size $n$ increases, the histogram becomes narrower, because this quantity converges to the effective density's width. The mean of this histogram can be thought of as an *effective width* to be used in gStochS for approximating eStochS. This effective width is what our gStochS recipe would try to find. Notice how eStochS found that the optimal width is 0.5 without the need of applying the recipe. This justifies, after the fact, our earlier reasoning that gStochS results of width 0.5 are more supported by the data than the results of width 0.25 despite both of them having similar fit histograms (Refer to Fig. 3.31 for gStochS results).

### 3.5.2. eStochS vs. fit histograms

Let us reexamine eStochS results for test case 3 using the $L_\infty$-norm shown in Fig. 3.34e. These results are equivalent to averaging over the results of gStochS using uniform prior densities with different cutoffs. According to Eq. 3.122, each result would be weighted by the posterior probability of its cutoff $P(w|\mathbf{g})$. Using the grids sampled by eStochS, we can approximate this distribution by the histogram of the maximum grid point shown in Fig. 3.39. The histogram shows that the sampled cutoff varies roughly around 5.

Remembering that gStochS results of this test cases were good for large cutoffs up to 64 (see Fig. 3.28a), it may then be surprising that eStochS averages only over such

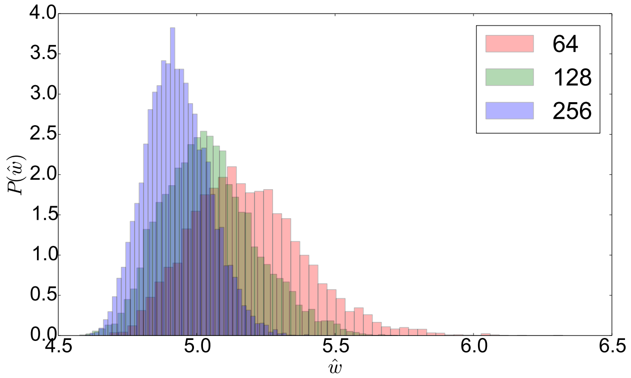


Figure 3.39.: Histograms of the scaled $L_\infty$-norm norm of grid samples for test case 3 using eStochS with different grid sizes (labels). The scaled $L_\infty$-norm of a grid sample $\mathbf{x}$ is calculated as the maximum grid point $\hat{w} := \lim_{q\to\infty} \sqrt[q]{\|\mathbf{x}\|_q^q/n} = \max|x_i|$.

small cutoffs. Since the fit histograms of gStochS for different cutoffs are equally good (Fig. 3.28b), one would expect that eStochS should average over a larger range of cutoffs, but it does not!

To understand the reason, note that the posterior probability used in eStochS is inversely proportional to $\mathrm{E}[e^{\chi^2/2}]$ and cannot be predicted from the fit histograms (see Eq. 3.107). Having a good fit histogram, i.e. a low mean fit $\mathrm{E}[\chi]$, does not guarantee a low expectation value $\mathrm{E}[e^{\chi^2/2}]$ because this value is very sensitive to the tail of the histogram. Therefore, although the fit histograms are quite reliable in choosing good values of the width, they cannot be used to predict the values averaged by eStcohS.

Since we can get an eStochS result at roughly the same cost of a gStochS result for a single width value, it is convenient to apply eStochS first. Its fit histograms can tell us when something goes wrong. Only then, we have to apply gStochS for different width values and judge the best one using the fit histograms.

## 3.6. Conclusion

In the original stochastic sampling method (StochS), we used a flat prior over the non-negative models and averaged them to smooth out the details that are not supported by the data. One would expect that such an ”uninformative” prior would provide unbiased results. Using our efficient sampling algorithm, BMS, we were able to perform calculations on larger and larger grids. This revealed that StochS results depend implicitly on the grid where the grid density acts as a default model and the grid size as its strength. This important grid dependence of StochS was beforehand unknown in the literature. We gave an explanation for the effect.

We provided a recipe for selecting a suitable grid and showed how to use the fit histograms to judge the quality of different grids. We also used the dependence of

$$\underbrace{\int dw \quad \underbrace{\int d\mathbf{x}P(\mathbf{x};w) \underbrace{\int_{\mathbf{F}\geq 0} d\mathbf{F}\ e^{-\chi^2(\mathbf{F},\mathbf{x})/2}\ f(x;\mathbf{F},\mathbf{x})}_{\text{StochS}}}_{\text{gStochS}}}_{\text{eStochS}}$$

Figure 3.40.: The hierarchal structure of stochastic sampling methods.

the results and histograms on the grid size to check how much bias the grid density introduces into the results. With the proper choice of grid, StochS gives very good and robust results and is able to resolve both sharp and broad features in the model.

To reduce the grid dependence, we extended StochS into a gridless method (gStochS) by sampling a fixed number of grid points from a reasonable density function centered at zero. In gStochS, this function also acts as a default model but its effect on the result is much reduced in comparison to the effect of the grid density on StochS results. In some test cases, the results were, to a large extent, even independent of the chosen density function. In other cases, they still depended critically on its width. We used a recipe to choose the proper width that is similar to the StochS grid recipe.

To avoid fixing the width, we went one step further and extended gStochS to sample flat over the width parameter of the default model. The new method (eStochS) is then able to automatically find the width values compatible with the data and average over them. The prior used by eStochS assumes that the model is concentrated around zero with unknown width and structure. For many cases, this is our best prior guess and it gives us good and robust results.

Our approach to grid dependence led us to the hierarchy of stochastic sampling methods shown in Fig. 3.40. However, it is by no means the final answer to the choice of a grid prior. We can still extend the method by varying over other parameters of the default model, and each new parameter requires specifying a prior whose choice is not unique. For example, if we have sufficient reason to suspect the existence of a gap at zero, then we could extend the default model to parametrize the width of the gap. We may start with a simple recipe to determine the width and consult the fit histograms to the determine the values supported by the data. If we then detect a strong dependence on the gap width that requires trying many different values, it may worth developing an algorithm to sample this parameter.

We could even average over the grid size. We actually tried this for eStochS using a float prior and found, as expected, that the method chooses a high number of points when the default model is highly compatible with the data as in test case 4. But when the prior assumption of being concentrated around zero is not satisfactory enough, like in test case 2 with two distinct peaks, the method chooses a very low number of points. In the end, we decided to keep the grid size as an independent control parameter of all stochastic sampling methods. A strong dependence on the grid size indicates a strong bias towards the prior and way from data, while a weaker dependence indicates more robust results.

# 4. Realistic Case Studies of Stochastic Analytic Continuation

In this chapter, we apply stochastic sampling methods to two realistic test cases taken from the recent literature where the exact result is unknown. The first case is obtaining the spectral function from DMFT calculations, while the second is obtaining the spin susceptibility from lattice QMC calculations. The results show that these methods do not only perform well but also indicates when the data does not agree with the default model. We also report the timings of the different methods and compare their scaling with the theoretical estimate.

## 4.1. Spectral function from DMFT

We received from D. Bergeron and A. M. Tremblay the data of DMFT calculations for the Hubbard model with U = 6 [19]. The relation between the data and the model is as following

$$\mathcal{G}(i\omega_n) = \int \frac{d\omega}{2\pi} \frac{1}{i\omega_n - \omega} \quad A(\omega) \; . \tag{4.1}$$

The data is the Fermionic Green function $\mathcal{G}(i\omega_n)$ and was given at the first 400 Matsubara frequencies $\omega_n = (2n+1)\pi/\beta$ with inverse temperature $\beta = 100$. An estimation of the covariance matrix of the data noise was given. Tremblay et al. also provided us with their analytic continuation using MaxEnt. They used a Gaussian default model with width 3.6 and regularization parameter $\alpha = 500$. Their result is shown in Fig. 4.1.

We start by applying StochS to this test case. Since the model extends over the whole real axis, we use a Gaussian grid centered at zero. Using the non-negative least squares solution, we estimate the second moment of the spectral function to be around 3.6 and use it as an initial width of the Gaussian. We perform StochS using different grid sizes $n$ and show the results in Fig. 4.2e. In Fig. 4.2f, we also show their fit histograms. The very low dependence of the models and the fit histograms on the grid size indicates that a Gaussian grid of width 3.6 is indeed a good choice and agrees well with the data. To check the stability of the results as a function of the grid width, we vary it on a logarithmic scale around 3.6. In the other plots of Fig. 4.2, we show StochS results using lower and higher values of the width. The grid of width 1.8 is too narrow to resolve the tail of the model leading to a high discretization error. The grid of width 2.5 gives very good results that are similar to width 3.6. Looking at the fit histograms, we see that width 2.5 is slightly preferable to 3.6. However, looking at the models and their grid size dependence, we conclude that the latter is more reliable. As the width increases

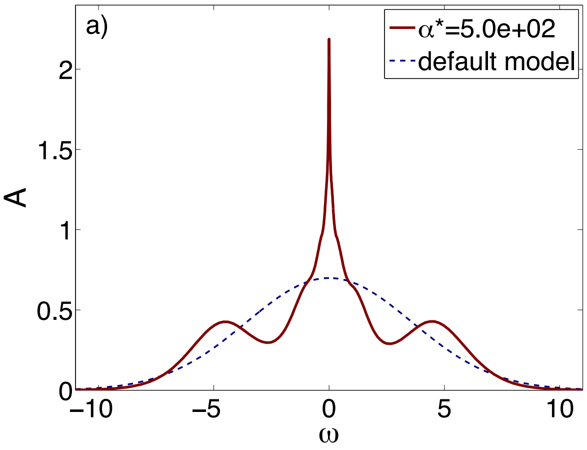


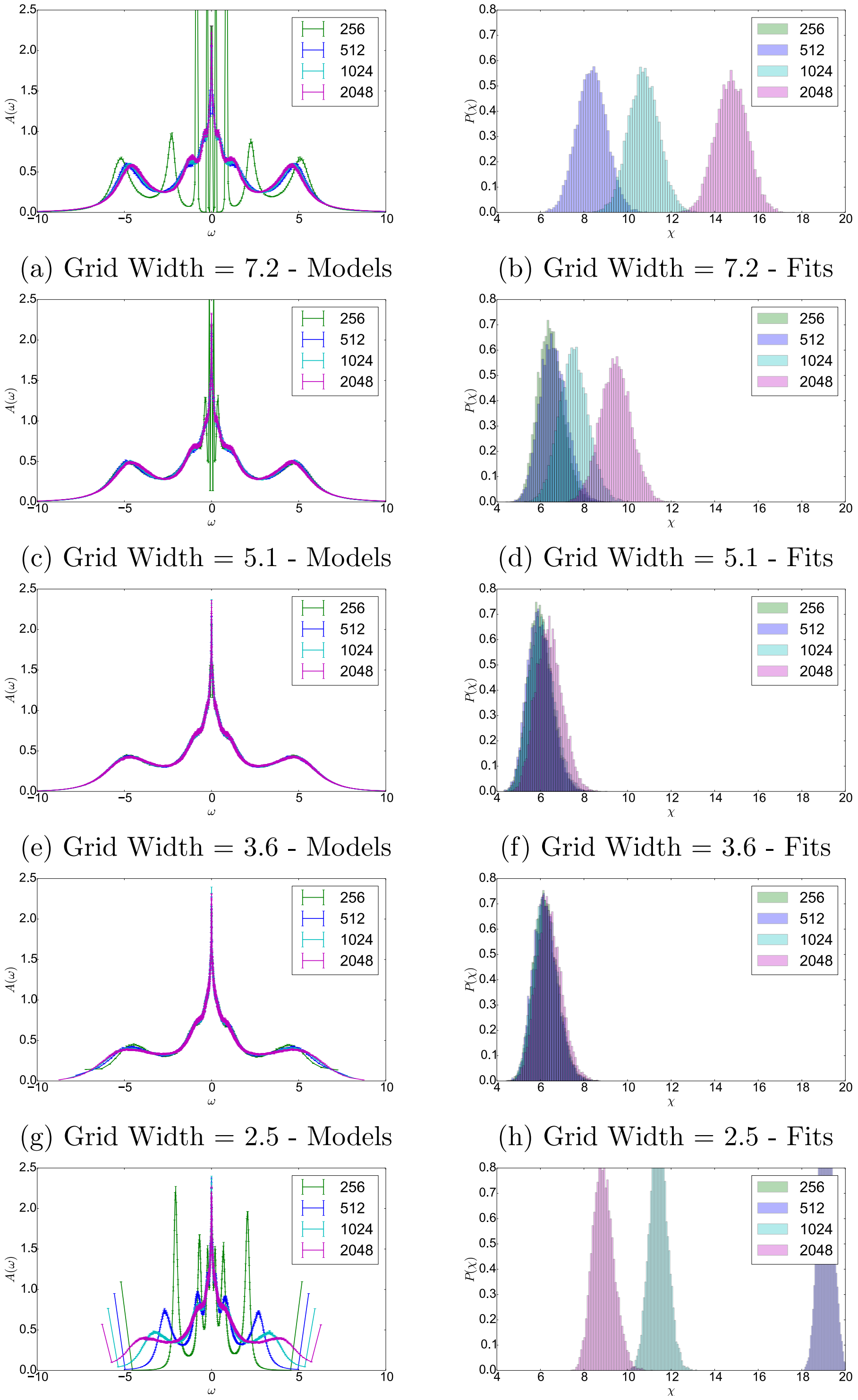
Figure 4.1.: MaxEnt result (solid red) for the DMFT test case (provided by D. Bergeron et al. [19]). The default model (dashed blue) is a Gaussian with width 3.6.

above 3.6, the fit histograms show a stronger dependence on the grid size, indicating a disagreement with the data.

In practice, we would have been satisfied with the StochS results for width 2.5 or 3.6. However, we also apply gStochS and eStochS here for completeness. In Fig 4.3, we show the results of gStochS using a Gaussian prior density function for the same widths used in StochS. Here again, we get the best results using a prior density with width 2.5 or 3.6. The results of width 2.5 are now more similar to those of width 3.6 and both are similar to StochS results using width 3.6. Note also that the results of other widths get better in comparison to StochS, confirming that sampling over the grid points in gStochS makes the results less biased.

In Fig. 4.4, we show the results of eStochS, which are similar to the best ones obtained by gStochS. Moreover, we show in Fig. 4.5 the histograms of the scaled norm of the sampled grids. This quantity approximates the width of the effective grid density, so these histograms show us that the sampled values of the width are centered around the best value 3.6. They also show that the posterior probability $P(w|\mathbf{g})$ of width 2.5 is very low, despite being supported by the fit histograms of gStochS. This is a reminder that the fit histograms and eStochS do not always agree when judging the quality of a width value. In this test case, the disagreement is minor and eStochS gives equally good results. Yet, eStochS is faster and easier because it needs only runs for different grid sizes, while in gStochS, we need such runs for each width separately.

Comparing stochastic sampling results with MaxEnt, we see that they are almost identical to MaxEnt! This should not be surprising because the data is of high-quality (its noise standard deviation is of the order $10^{-6}$), so this test case is quite easy. This is confirmed by the fact that even simpler and cheaper methods like non-negative Tikhonov provide results of comparable quality (see Fig. 4.6).



(a) Grid Width = 7.2 - Models

(b) Grid Width = 7.2 - Fits

(c) Grid Width = 5.1 - Models

(d) Grid Width = 5.1 - Fits

(e) Grid Width = 3.6 - Models

(f) Grid Width = 3.6 - Fits

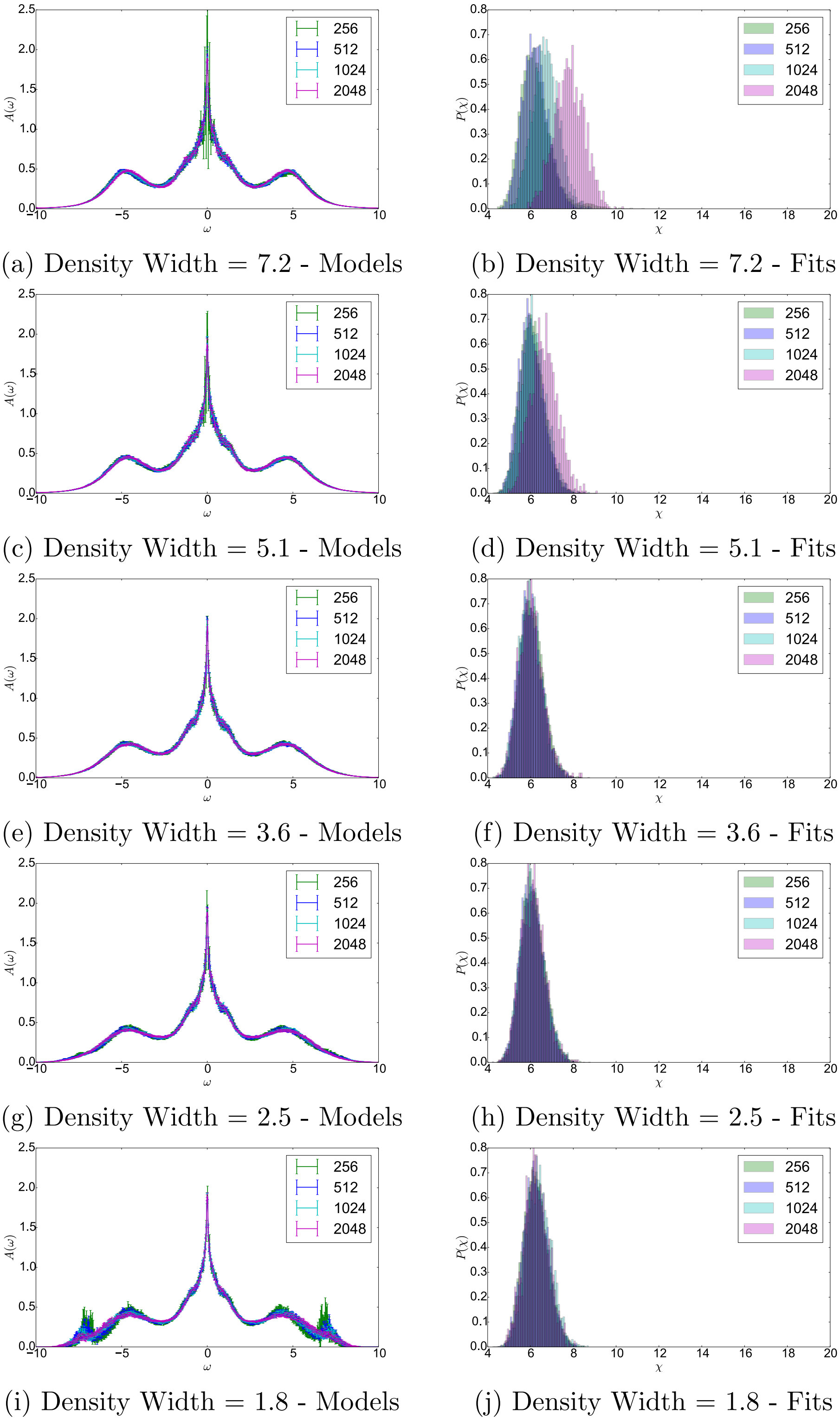
(g) Grid Width = 2.5 - Models

(h) Grid Width = 2.5 - Fits

(i) Grid Width = 1.8 - Models

(j) Grid Width = 1.8 - Fits

Figure 4.2.: StochS results for the DMFT test case using Gaussian grids of different widths (captions) and different sizes (labels).



(a) Density Width = 7.2 - Models

(b) Density Width = 7.2 - Fits

(c) Density Width = 5.1 - Models

(d) Density Width = 5.1 - Fits

(e) Density Width = 3.6 - Models

(f) Density Width = 3.6 - Fits

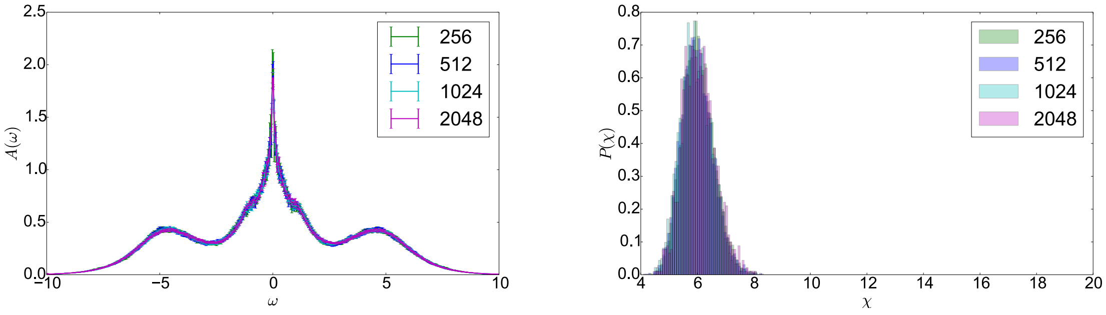
(g) Density Width = 2.5 - Models

(h) Density Width = 2.5 - Fits

(i) Density Width = 1.8 - Models

(j) Density Width = 1.8 - Fits

Figure 4.3.: gStochS results for the DMFT test case using Gaussian prior densities of different widths (captions) and different sizes (labels).



(a) Models

(b) Fits

Figure 4.4.: eStochS results for the DMFT test case using $L_2$-norm and different grid sizes (labels). The tiny fluctuations are sampling noise.

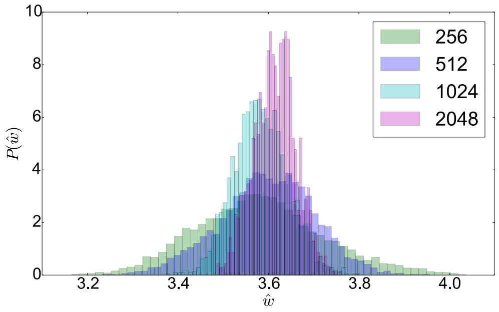


Figure 4.5.: Histograms of the scaled $L_2$-norm of grid samples for the DMFT test case using eStochS with different grid sizes (labels). The scaled $L_2$-norm of a grid sample $\mathbf{x}$ is calculated as the standard deviation of its points $\hat{w} := \sqrt[2]{\|\mathbf{x}\|_2^2/n} = \sqrt{\sum_i x_i^2/n}$.

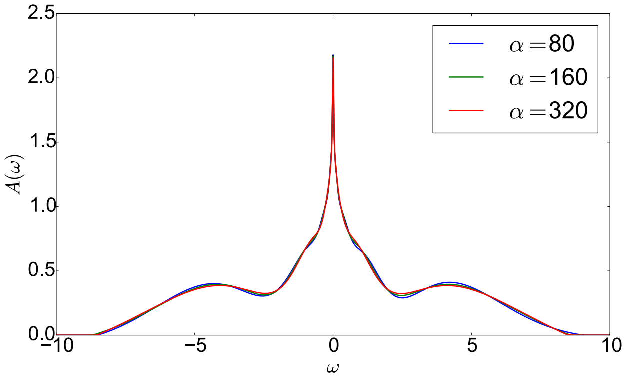


Figure 4.6.: Non-negative Tikhonov results for the DMFT test case using different values of the regularization parameter $\alpha$ (labels). This range of values was chosen according to the discrepancy principle.

## 4.2. Dynamical spin structure factor from lattice QMC

We received from J. Becker et al. from S. Wessel group at RWTH Aachen the data of lattice QMC calculations for the Spin-1 Heisenberg chain [36]. The goal is estimating the the dynamical spin structure factor $S^{zz}(\omega)$. The relation between the data $\chi^{zz}(i\omega_n)$ and the model $S^{zz}(\omega)$ has the following form

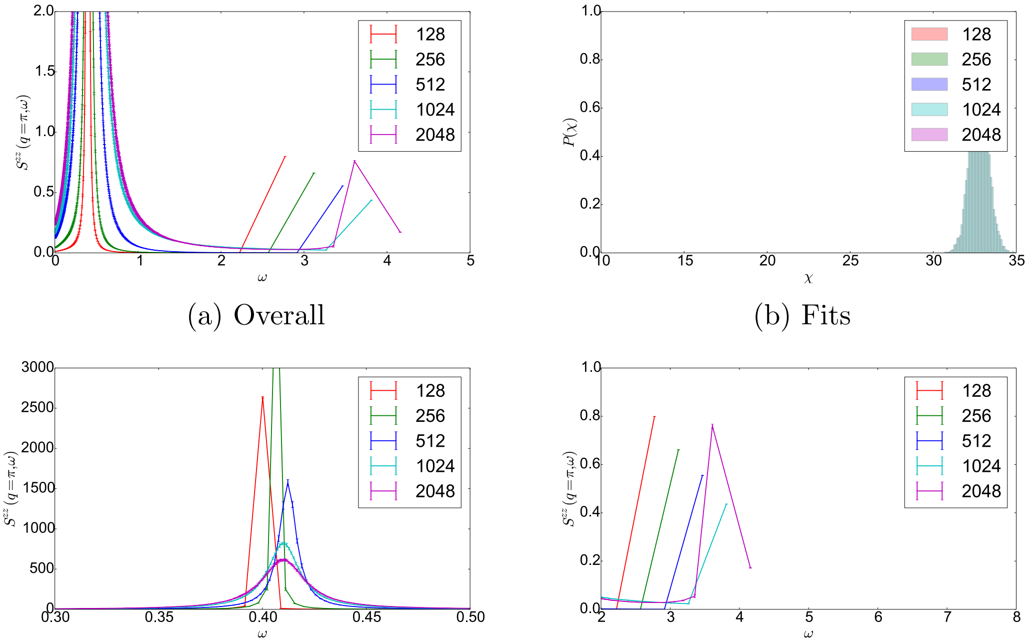
$$\chi^{zz}(i\omega_n) = \int_0^\infty \frac{d\omega}{\pi} \, \frac{\omega(1 - e^{-\beta\omega})}{\omega_n^2 + \omega^2} \quad S^{zz}(\omega) \; . \tag{4.2}$$

The data was given at the first 201 bosonic Matsubara frequencies $\omega_n = 2n\pi/\beta$ with inverse temperature $\beta = 24$. The covaraince matrix of the data noise is assumed to be diagonal and estimations of its diagonal elements were given.

First, we apply StochS. Since the model in this test case extends over the positive half-line only, we use an exponential grid. Using non-negative least squares, we estimate the first moment of the model to be around 0.5 and use it as an initial width of the grid. In Fig. 4.7, we show the results using different grid sizes. Clearly, this grid is too narrow to resolve the tail of the model leading to a high discretization error. Therefore, we increase the width of the grid logarithmically up to 16.0 and show the results using different widths in Fig. 4.8. As the width increases to 4.0, the fit histograms shift to the left. Increasing the width further shifts the histograms back to the right, so we conclude that width 4.0 is the best width. In Fig. 4.9, we show close-ups of the peak and the tail for the different widths. Notice that as the width increases, the number of grid points near zero decreases leading to a lower resolution of the peak. Also the hump in the tail is shifted to the right and more pronounced for large widths.

The strong dependence of the StochS results on the grid size indicates that the grid does not agree well with the data. This is understandable because the default model, which is the exponential grid density, is very different from the exact model, which evidently has a gap at zero, a very sharp peak and a long tail. We repeat the calculations using gStochS and expect an improvement. Fig. 4.10 shows that gStochS results indeed have a weaker dependence on the grid size. Checking the fit histograms, we find that the large width 16.0 is the best and has even better fits than those of StochS with width 4.0. In Fig. 4.10, we show close-ups of the peak and the tail. The main difference between StochS and gStochS results is the height of the sharp peak. StochS gives a peak that is about 3 times higher than that of gStochS. However, we know from experience that when the grid density disagrees with the data, StochS tends to make features sharper than they really are. Therefore, gStochS results should be more reliable. This is also indicated by the fit histograms. The grid size dependence of the peak in gStochS is expected since increasing the grid size biases the results towards the exponential default model. Another difference between the two methods is the hump in the tail, which is broader in gStochS and less dependent on the grid size.

In Fig. 4.12, we show eStochS results using the $L_1$-norm. The hump in the tail looks different from gStochS results. Checking the histograms of the scaled norm, shown in Fig. 4.13, we see that the sampled values of the width are actually less than 1.0. This is may seem surprising since the fit histograms of gStochS indicate that the values of the



(a) Overall

(b) Fits

(c) Peak

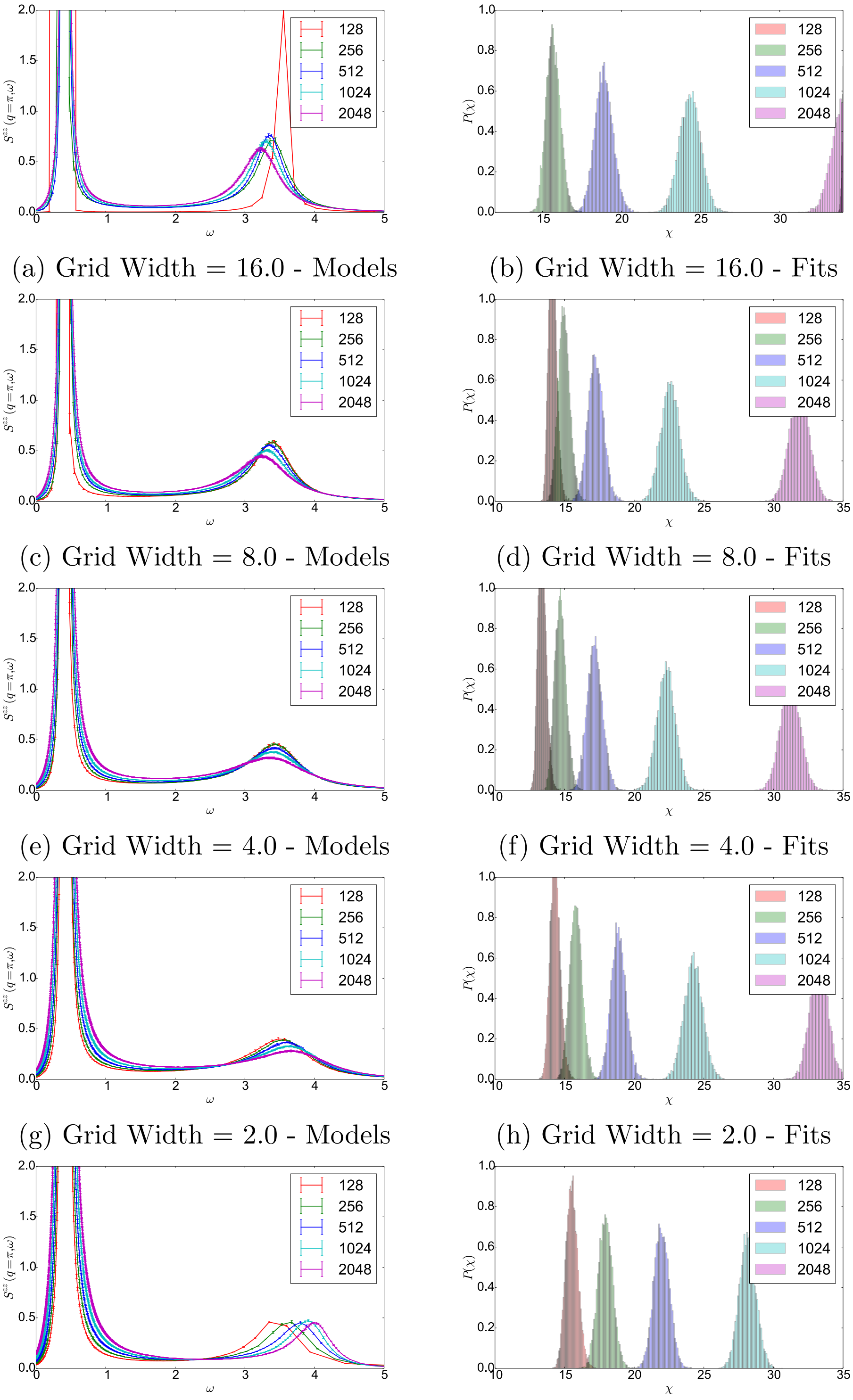
(d) Tail

Figure 4.7.: StochS results for the lattice QMC test case using an exponential grid of width 0.5 and different grid sizes (labels).

width up to 16.0 are equally good and that high values are even slightly better than small ones. This apparent contradiction between the fit histograms and eStochS comes from the fact that they use different criteria to judge the quality of a width value. The center of a fit histogram represents the expectation value E$[\chi]$, while the widths in eStochS are weighted by their posterior probability $P(w|\mathbf{g})$ which is proportional to E$[e^{\chi^2/2}]$ (see Sec. 3.5.2). So far we have found that the fit histograms provide a better criterion for judging the quality of stochastic sampling results. Moreover, the scaled norm histograms (see Fig. 4.13) shift to lower values as the grid size increases, indicating a disagreement between the data and the prior of the width in eStochS. Therefore, we conclude that gStochS with width 16.0 and grid size 128 is our most reliable result.

Becker et al. also provided us with their stochastic analytic continuation result using Beach's method [31]. They used a completely flat default model with 1500 delta functions. This method is closely related to gStochS and gives similar results as shown in Fig. 4.14.

It is important to note that the peak in all stochastic sampling results is about 1000 times larger than the hump in the tail. We expect that for such difficult test cases, that stochastic sampling methods to be the choice due to their ability of resolving both sharp and broad features. For comparison, we show in Fig. 4.15 the results of non-negative Tikhonov for a wide range of the regularization parameter $\alpha$. None provides a result of a quality that comes close to that of stochastic sampling!



(a) Grid Width = 16.0 - Models

(b) Grid Width = 16.0 - Fits

(c) Grid Width = 8.0 - Models

(d) Grid Width = 8.0 - Fits

(e) Grid Width = 4.0 - Models

(f) Grid Width = 4.0 - Fits

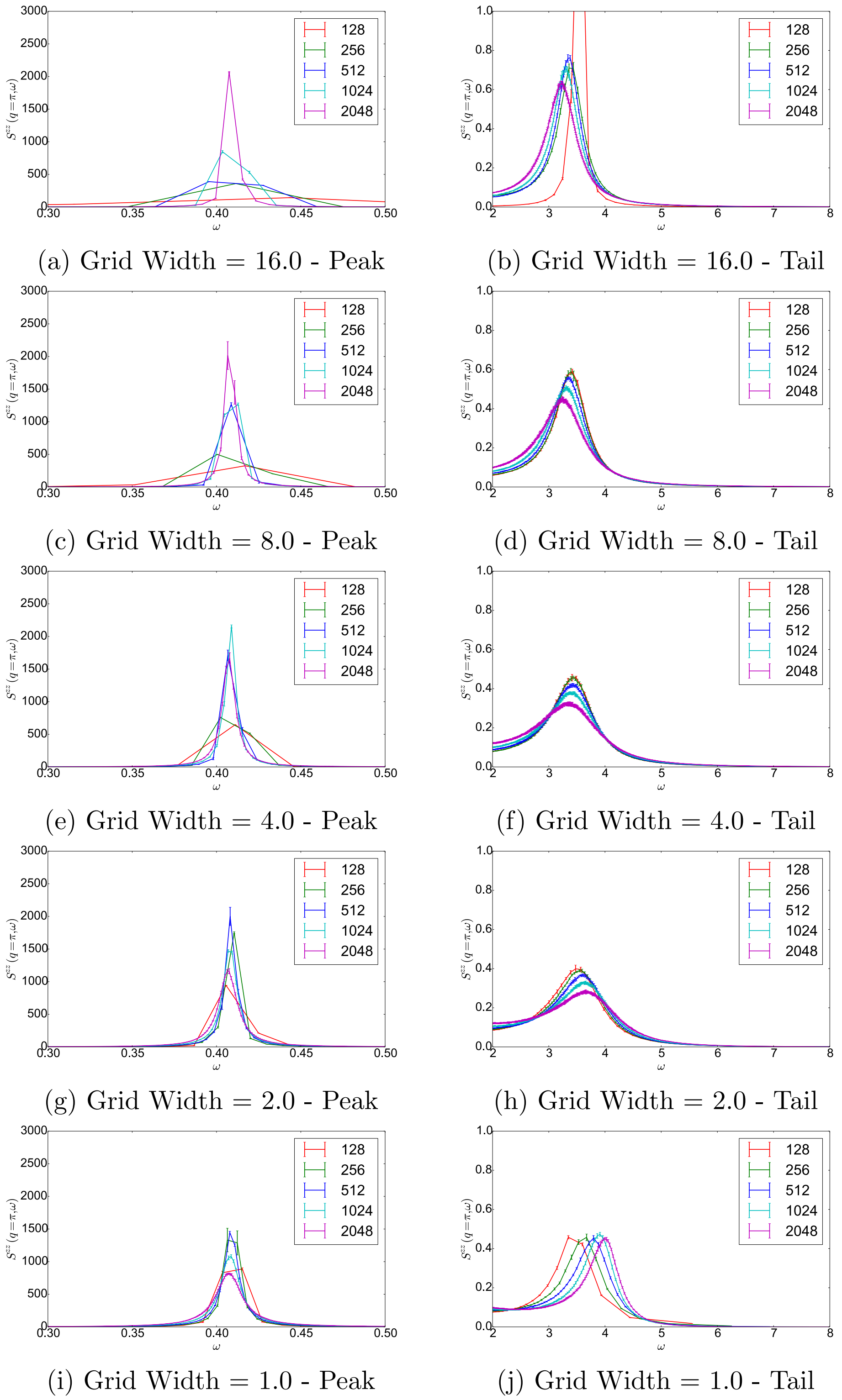
(g) Grid Width = 2.0 - Models

(h) Grid Width = 2.0 - Fits

(i) Grid Width = 1.0 - Models

(j) Grid Width = 1.0 - Fits

Figure 4.8.: StochS results for QMC lattice test case using exponential grids of different widths (captions) and different sizes (labels). Checking the fit histograms, the best result is using width 4.0.



(a) Grid Width = 16.0 - Peak

(b) Grid Width = 16.0 - Tail

(c) Grid Width = 8.0 - Peak

(d) Grid Width = 8.0 - Tail

(e) Grid Width = 4.0 - Peak

(f) Grid Width = 4.0 - Tail

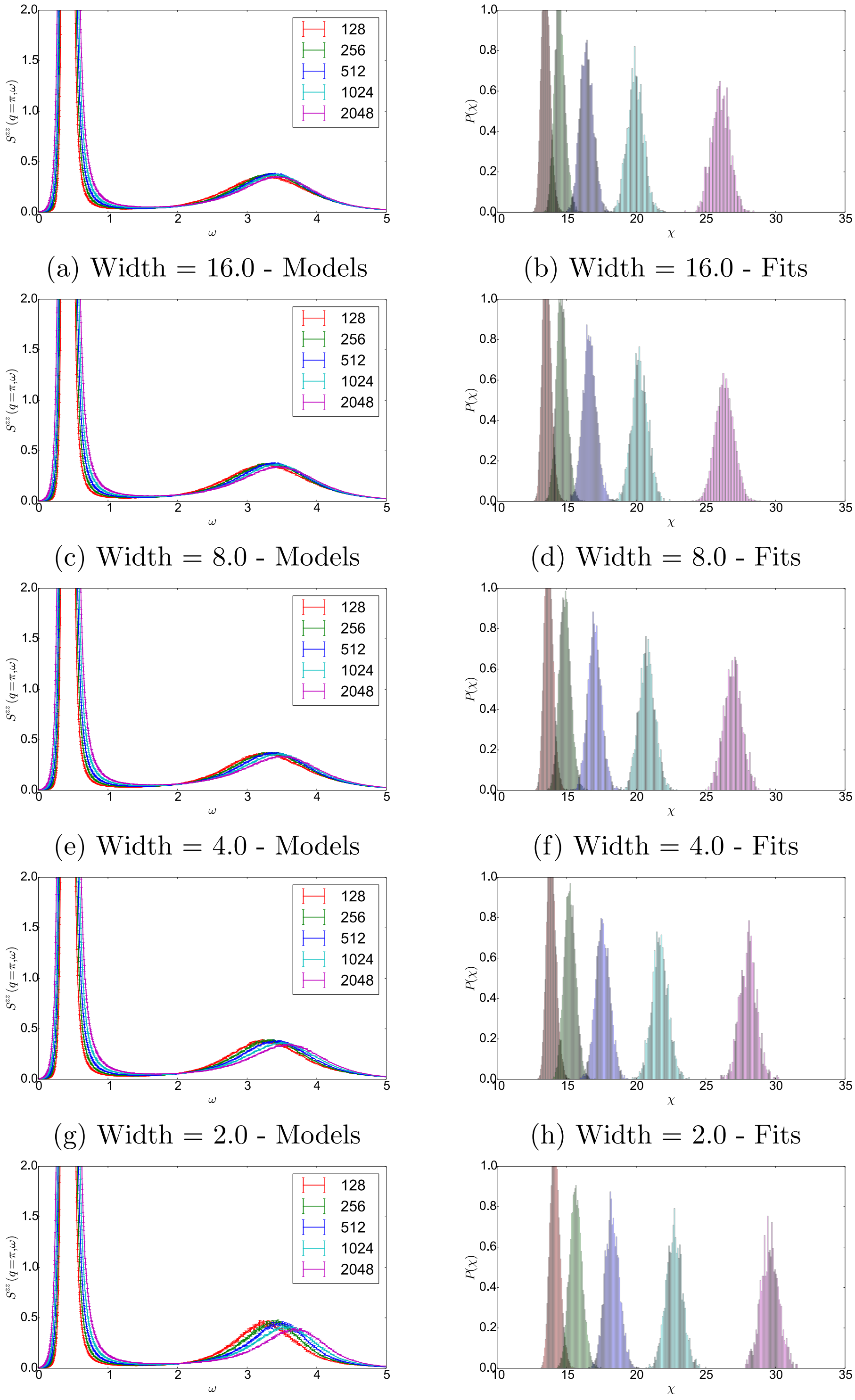
(g) Grid Width = 2.0 - Peak

(h) Grid Width = 2.0 - Tail

(i) Grid Width = 1.0 - Peak

(j) Grid Width = 1.0 - Tail

Figure 4.9.: Zooms on the peaks and tails of StochS results for QMC lattice test case using exponential grids of different widths (captions) and different sizes (labels). Note that the peak is about 1000 times larger than the tail.



(a) Width = 16.0 - Models

(b) Width = 16.0 - Fits

(c) Width = 8.0 - Models

(d) Width = 8.0 - Fits

(e) Width = 4.0 - Models

(f) Width = 4.0 - Fits

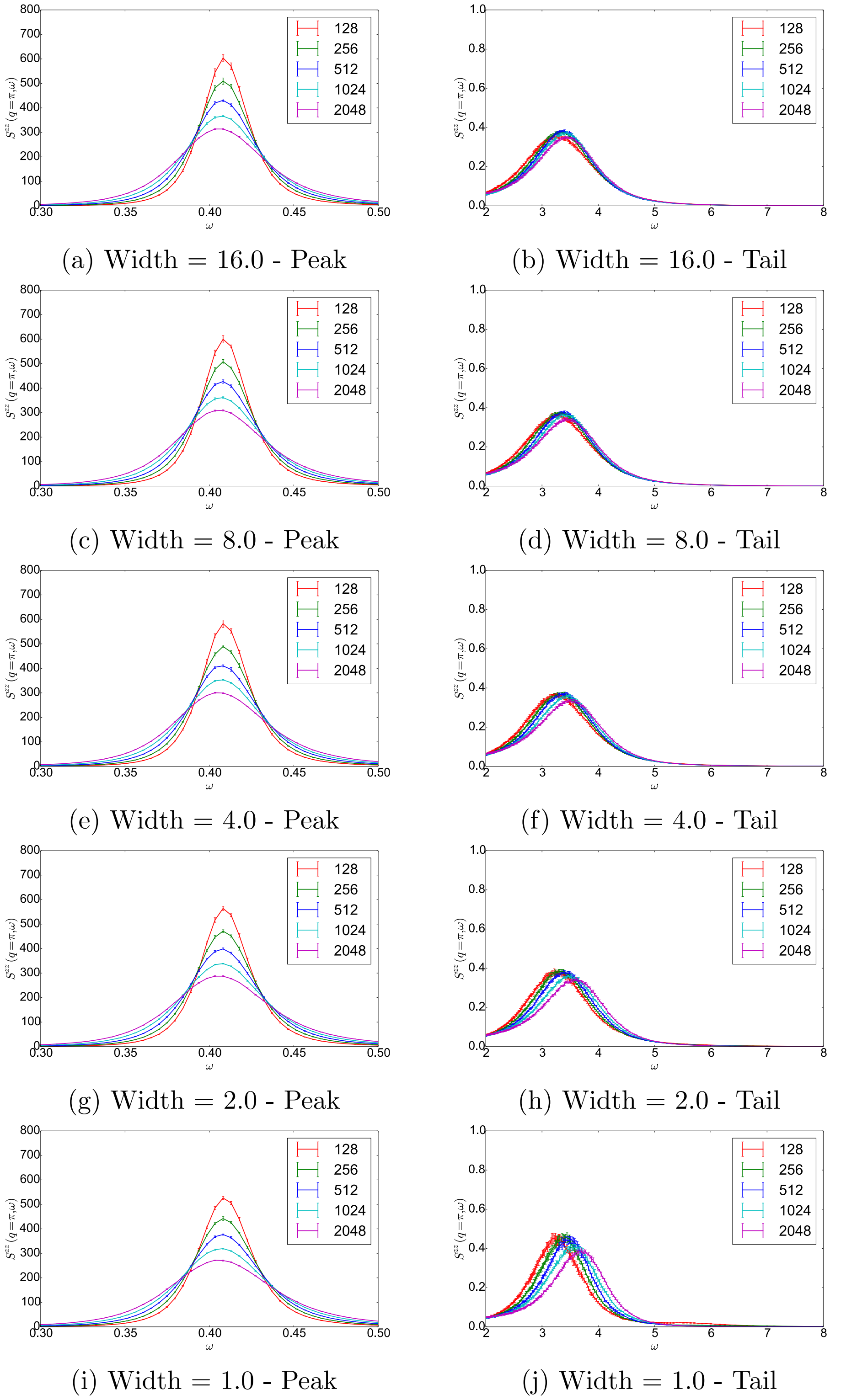
(g) Width = 2.0 - Models

(h) Width = 2.0 - Fits

(i) Width = 1.0 - Models

(j) Width = 1.0 - Fits

Figure 4.10.: gStochS results for QMC lattice test case using exponential prior densities of different widths (captions) and different sizes (labels). Checking fit histograms, the best result is using width 16.0.



(a) Width = 16.0 - Peak

(b) Width = 16.0 - Tail

(c) Width = 8.0 - Peak

(d) Width = 8.0 - Tail

(e) Width = 4.0 - Peak

(f) Width = 4.0 - Tail

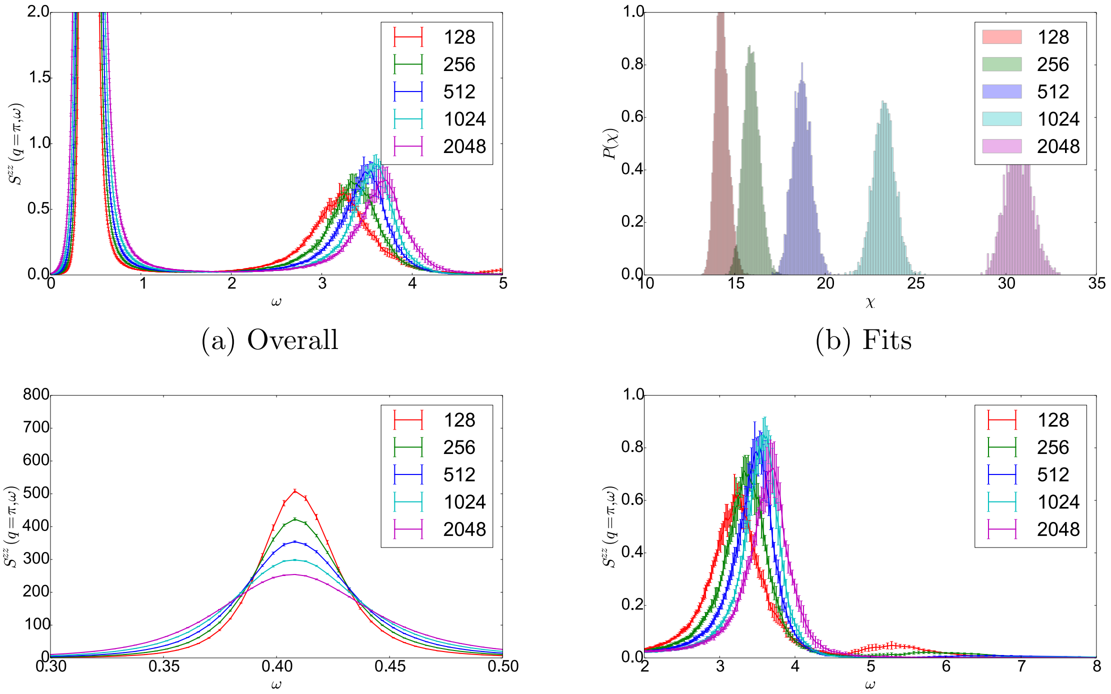
(g) Width = 2.0 - Peak

(h) Width = 2.0 - Tail

(i) Width = 1.0 - Peak

(j) Width = 1.0 - Tail

Figure 4.11.: Zooms on the peaks and tails of gStochS results for QMC lattice test case using exponential prior densities of different widths (captions) and different sizes (labels). Note that the peak is about 1000 times larger than the tail.



(a) Overall

(b) Fits

(c) Peak

(d) Tail

Figure 4.12.: eStochS results for the lattice QMC test case using $L_1$-norm and different grid sizes (labels).

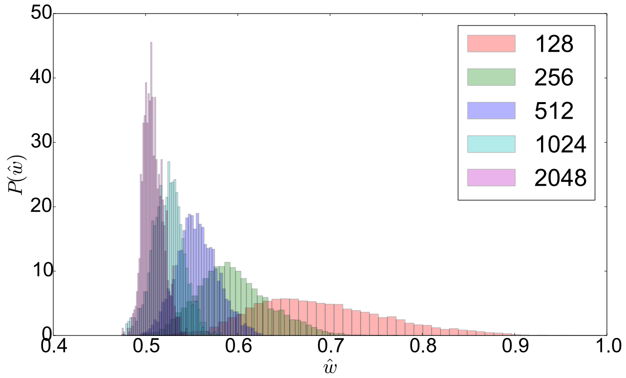
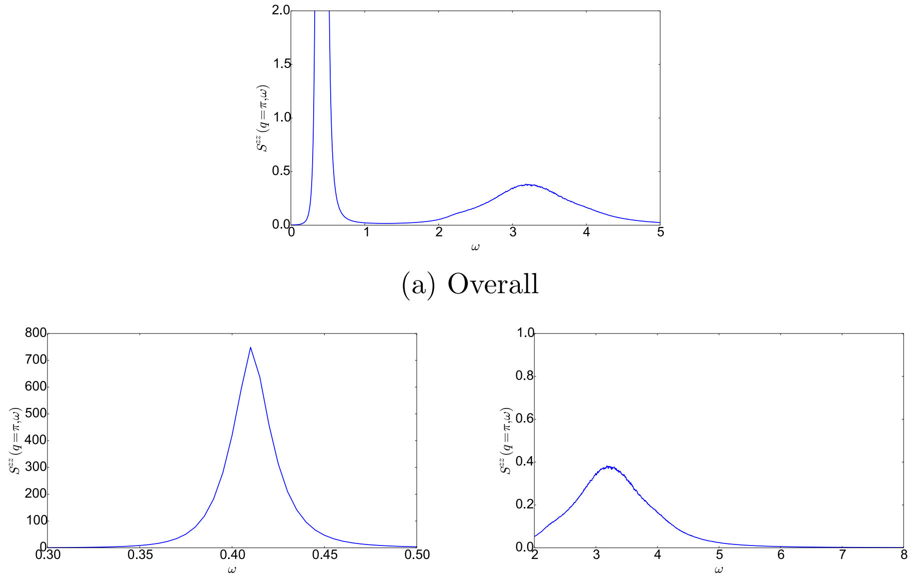


Figure 4.13.: Histograms of the scaled $L_1$-norm of grid samples for the lattice QMC test case using eStochS with different grid sizes (labels). The scaled $L_1$-norm of a grid sample $\mathbf{x}$ is calculated as the mean of its points $\hat{w} \coloneqq \|\mathbf{x}\|_1/n = \sum_i |x_i|/n$.

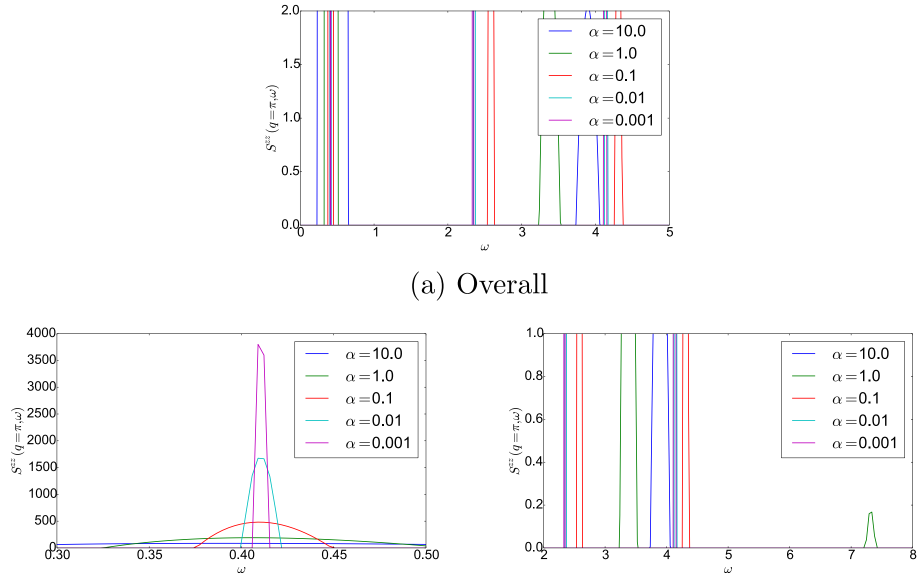


(a) Overall

(b) Peak

(c) Tail

Figure 4.14.: Stochastic analytic continuation result using Beach's method [31] for the lattice QMC test case (provided by Becker et al. [36]). They used a completely flat default model with 1500 delta functions.



(a) Overall

(b) Peak

(c) Tail

Figure 4.15.: Non-negative Tikhonov (NNT) results for the lattice QMC test case using a wide range of values for the regularization parameter $\alpha$ (labels). Clearly, NNT does not work at all for this test case.

## 4.3. Setup and running times

Each stochastic sampling result was obtained by averaging the samples of 8 independent runs. Each run generated $8N$ samples where only every 8th sample is kept, while the rest are discarded. This saved storage space and removed some of the correlation between consecutive samples. The error bars of a result are computed as the standard deviation of the averages of the independent runs, so they are not affected by the correlations in the samples of the individual runs. Each run was executed on a single core Intel(R) Xeon(R) CPU X5570 @ 2.93GHz.

In StochS, we used $N = 10000$. In gStochS/eStochS, the used $N$ depended on the grid size $n$. We used a lower number of samples $N$ for larger grid sizes because the error bars of these methods using a fixed binning grid scales roughly with $\sqrt{Nn}$. Therefore, keeping the product $Nn$ fixed gives roughly the same error bars. For $n = 128$, we used $N = 20000$ and then linearly decreased $N$ as $n$ increases.

The timings of a single run of StochS for different grid sizes $n$ are presented in Table 4.1. Although the theoretical scaling of the StochS algorithm is quadratic $O(n^2)$, the table shows that our Python implementation scales almost linearly! This can be explained by the fact that we have an $O(n)$ Python loop, inside which we call several $O(n)$ functions from the Numpy library, which are implemented natively. Apparently, the Python overhead is so large that the cost of Numpy functions is negligible for the used values of $n$.

To test this interpretation, we wrote a simple Python script shown in Listing 4.1. This script has a similar structure to StochS code. It has an outer Python loop of size $n$, inside which we find the minimum of an $n$-sized array using a Numpy function *amin*. In Fig. 4.16, we plot the timings of this Python code for different values of $n$. As expected for small values of $n$, the computational time is linear. It starts to deviate from this around $n = 2084$ till it becomes quadratic after $n = 16385$.

To check the Python overhead, we also wrote a similar C code, shown in Listing 4.2, and plotted its timing for comparison. This code has the expected quadratic scaling even for small $n$ where it outperforms the Python code significantly. For $n = 1024$, the

| $n$ | DMFT | lattice QMC |
|---|---|---|
| 128 | 18 min | 17 min |
| 256 | 40 min | 35 min |
| 512 | 80 min | 70 min |
| 1024 | 150 min | 125 min |
| 2048 | 300 min | 265 min |

Table 4.1.: Total running times of StochS for the DMFT and lattice QMC test cases for different grid sizes $n$.

C code is about 10 times faster than the Python code. We expect a similar speed-up when reimplementing StochS in C.

For similar reasons, the running times of gStochS/eStochs are also linear with the grid size $n$ (for a fixed number of samples $N$). However, since the number of samples was taken to be inversely proportional to the grid size, we got roughly constant running times. For the DMFT test case, the running time was about 18 hours, while for the lattice QMC test case, it was about 3 hours. Again, with a C implementation, we expect these running times to be reduced by an order of magnitude for typical values of $n$. The difference in running times between the two test cases is mainly due to the difference in the size of the data vector $m$. For the DMFT test case, we have 400 complex values i.e. 800 real values. On the other hand, for the lattice QMC, we have 201 real data values. Theoretically, gStochS/eStochS should scale as $O(m^2)$ because moving one grid point requires reevaluating a kernel vector of size $m$ and transforming it into a basis where the data is uncorrelated i.e. a multiplication by a matrix of size $m \times m$.

```python
import numpy as np
import time

def f(x):
    return np.amin(x)


sizes = [16, 32, 64, 128, 256, 512, 1024, 2048, 4096, 8192, 16384,
    32768];
print "n\tTime"
for n in sizes:
    x = np.random.rand(n)
    begin = time.clock()
    for iters in range(1000):
        for i in range(n):
            b = f(x)
    end = time.clock()
    time_spent = end-begin
    print("%d\t%f"%(n, time_spent))
```

Listing 4.1: Python Code

```c
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

double f(int n, double x[])
{
  double min_x = x[0];
  for(int i=1;i<n;i++)
  {
    if(x[i]<min_x)
      min_x = x[i];
  }
  return min_x;
}
```

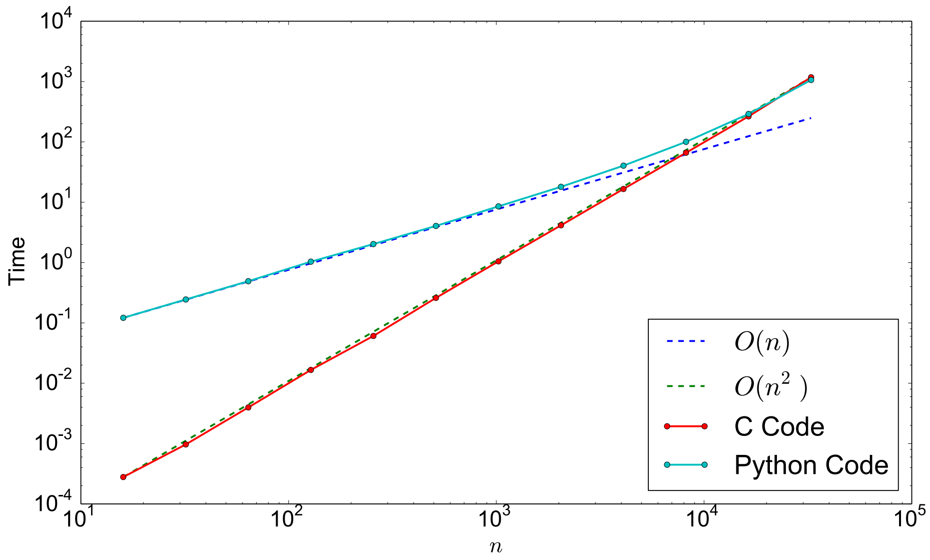


Figure 4.16.: A comparison of the running times of a simple script written in Python (Listing 4.1) and C languages (Listing 4.2). Although the algorithm scales quadratically, the Python script shows a linear behavior for small input size $n$. The reason is that Numpy function *amin*, which is implemented natively, is much faster than the rest of the Python code.

```c

int main(int argc, char *argv[] )
{
  int sizes[12] = { 16, 32, 64, 128, 256, 512, 1024, 2048, 4096, 8192,
   16384, 32768 };
  printf("n\tTime\n");
  for(int k=0;k<12;k++)
  {
    int n = sizes[k];
    double x[n];
    for (int i = 0; i < n; i++)
        x[i] = rand()/ (RAND_MAX + 1.0);
    clock_t begin = clock();
    for(int iters = 0;iters<1000;iters++)
      for(int i = 0; i<n; i++)
        volatile double b = f(n,x);
    clock_t end = clock();
    double time_spent = (double)(end - begin) / CLOCKS_PER_SEC;
    printf("%d\t%f\n", n, time_spent);
  }
   return 0;
}
```

Listing 4.2: C Code

# Summary and Outlook

This thesis deals with the analytic continuation problem for the fermionic and bosonic spectral densities. In chapter 1, we start from the origin: the analyticity of Green and correlation functions. We use the greater and lesser Green functions as building blocks to define a unified Green function on the whole complex time plane. By taking different contours, we are able to express all Green functions of time as different faces of the same entity. Similarly, we use the retarded and advanced Green functions to define the Green function in the complex frequency plane, and relate all Green functions of frequency to it. We systemically show how to move from the time domain to the frequency domain and back. Our presentation shows a nice duality between Green functions in the two domains. We similarly discuss the analytic structure of correlation functions and their relations. We then derive the spectral densities of Green and correlation functions and show that they form positive-definite matrices. This implies that their diagonal elements are non-negative functions in all different bases. Obtaining these diagonal spectral functions from QMC data is the analytic continuation problem we set out to solve.

In chapter 2, we study this problem as a Fredholm integral equation of the first kind. Discretizing the integral on a grid gives us a finite-dimensional linear system. We show that the ill-posedness comes from the exponentially decaying singular values of the resulting kernel matrix. We use the ill-posedness to our advantage and provide a simple formula for estimating the noise level on the data using its SVD expansion coefficients. We then discuss a few regularization methods like truncated SVD and Tikhonov regularization. We explain the non-negative least squares (NNLS) method and suggest a modification that improves its convergence rate. We demonstrate how non-negativity constraints alone act as an important regularization. We also develop a new regularization method, the perturbed data sampling (PDS), that averages over different NNLS solutions using data with artificially added noise. We use PDS later as a good initialization for the stochastic sampling methods.

In chapter 3, we introduce Bayesian inference and show how different methods like MaxEnt, Tikhonov, and NNLS can be expressed within this framework. We then define StochS as a Bayesian method with a flat prior and a mean estimator. We develop blocked modes sampling (BMS), an efficient stochastic sampling algorithm that utilizes the SVD of blocks of the kernel matrix. Having a flat prior, one would assume the StochS results to be unbiased. Surprisingly, we find that the discretization grid acts as an implicit default model affecting the results. We explain this effect by showing that a flat prior on the grid implies a gamma prior on other grids. By including this prior explicitly, we simulate the results of one grid using others. We also relate StochS to gamma stochastic processes where the flat prior arises naturally from projecting such a process on a specific grid. Faced with the grid dependence, we provide two heuristics for determining when a grid is reliable: fit histograms and grid size dependence.

We then develop gridless stochastic sampling (gStochS) by drawing the grid points independently from a prior density function which acts as an explicit default model. Sampling over grid points incurs an additional computational cost, but reduces the effect of the default model dramatically. For some test cases, however, the results are still sensitive to the width of the default model. So we extend the method further into eStochS, by averaging over different widths. For typical densities, we can perform the width average analytically and get eStochS at basically no extra cost in comparison to gStochS. In most cases, eStochS is able to find the proper width without the need for a recipe. In cases where it fails, we can identify this failure from the fit histograms and the grid size dependence, and find a better estimate by varying the width by hand. An example is given in chapter 4, where we test our approach using real data from DMFT and lattice QMC calculations.

In eStochS, we use a default model that is concentrated around zero with varying width. We could try to extend the method one more time by using a multi-peak structure with varying centers and widths. But even then, one can construct test cases where further extensions are needed. In principle, this is a never-ending process because there is an infinite number of possible parameters. The more flexible the default model becomes, the less biased the results are, but the harder the sampling gets. In practice, we have to identify the most relevant parameters and stop at a point that achieves a balance between performance and efficiency. Moreover, the choice of the prior over these parameters is not unique, and we should always consult our heuristics to check whether our assumptions are consistent with the data.

Besides the possibility of extending the prior, there are still two open questions in stochastic sampling. The first is extending the approach to the analytic continuation of non-diagonal elements. As discussed earlier, the spectral functions form a positive-semidefinite matrix, i.e., for each frequency $x$, the set of functions $f_{i,j}(x)$ form a hermitian matrix with non-negative eigenvalues. One approach to sample these functions is updating individual elements $f_{i,j}(x)$ of the positive-semidefinite matrix directly (alongside the transpose $f^*_{j,i}(x) = f_{i,j(x)}$ to preserve hermiticity). We can show that positive-definiteness can be preserved by restricting the updated values to bounded intervals. Computing the bounds efficiently is, however, not trivial. Also it is not clear whether the updates allowed by these bounds would be large enough for efficient sampling. Another approach is writing the sampled functions in terms of frequency-dependent eigenvalues $\lambda(x)$ and frequency-dependent eigenvectors $\mathbf{v}(x)$

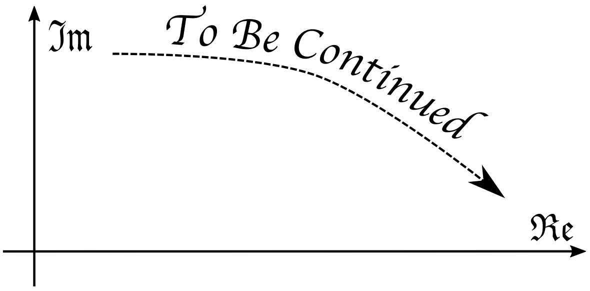
$$f_{i,j}(x) = \sum_k v_{i,k}(x)\lambda(x)v_{k,j}(x) \ .$$

The eigenvalues should be non-negative and the eigenvectors should form a unitary transformation. We already know how to sample non-negative functions and thus we only need to extend the algorithm to sampling unitary transformation matrices.

The second open problem is quantifying the uncertainty in the stochastic sampling result. So far we used only the mean of the posterior as the final estimate, implicitly assuming that uncertainty in the positions of the features should be reflected by smoothening of these features to the extent of the uncertainty. Nevertheless, we should remember that the mean is just one model while the Bayesian framework provides us with the probability of every possible model and thus the uncertainty in our estimate. Since the sampled models are evaluated on a binning grid and averaged, this uncertainty can be summarized in the covariance matrix of the bin values (these *intrinsic* variances should not be confused with the statistical errors in computing the average, although the two are proportional). The problem is that the variances of typical bins are extremely large to the extent of being useless. This should not come as a surprise because the bins are usually narrow, and neither the data nor the prior provide information about such fine details of the model. In order to get useful information, we need to reduce the variances of the bins by increasing their sizes. Think of it as a trade-off between vertical uncertainty (variance) vs. horizontal uncertainty (bin size). We developed an algorithm to find the optimal trade-off and the preliminary results are promising. It starts with one large bin and recursively splits the bins such that only statistically significant features (i.e. features with high posterior probability) are resolved. The downside is that the resulting binning is often too coarse to be visually appealing. More work in this direction needs to be done.

To conclude, we have developed a hierarchy of stochastic sampling methods. They share their flat prior over non-negative model integrals **F** and differ by their prior over grid points **x**. Each new method is extending the preceding one by averaging over the most relevant parameters of its grid prior. Our hierarchal approach to priors alongside with the diagnostic tools of fit histograms and grid size dependence, provides not only a set of promising analytic continuation methods, but more generally, a framework for the systematic development of even more complex and reliable stochastic sampling methods that can be tailored to the different challenges in analytic continuation.

$$\underbrace{\ldots\ldots \underbrace{\int dw \; \underbrace{\int d\mathbf{x} P(\mathbf{x};w) \underbrace{\int_{\mathbf{F}\geq 0} d\mathbf{F}\; e^{-\chi^2(\mathbf{F},\mathbf{x})/2}\; f(x;\mathbf{F},\mathbf{x})}_{\text{StochS}}}_{\text{gStochS}}}_{\text{eStochS}}}_{\vdots}$$



# A. Prior Stochastic Processes

Analytic continuation boils down mathematically to estimating a density-like function $f(x)$. In order to apply Bayesian inference to the analytic continuation problem, we need to assign prior probabilities to the space of these density-like functions. Defining a probability measure, known as a **stochastic process**, on such infinite-dimensional spaces is a complicated subject that requires a measure-theoretic treatment of probability. We will discuss the admissible priors in familiar terms sacrificing mathematical rigorousness for readability. Our approach follows closely [37]. We will also relate the different methods to their stochastic processes giving a framework to understand StochS, MaxEnt and Beach [31] methods on the same footing.

## A.1. General construction of admissible processes

The basic idea is that one can define a stochastic process on an infinite-dimensional space by specifying all finite-dimensional probability distributions in a "consistent" way. We will construct a family of finite-dimensional representations of $f(x)$ and specify the consistency condition that should by satisfied by the corresponding distributions. Let $\Omega$ be the domain over which the function $f(x)$ is defined with end points $a$ and $b$. Using a grid of points $\pi = \{x_0, .., x_n\}$ such that

$$a = x_0 < x_1 < x_2 < ... < x_n = b \ ,$$

we partition $\Omega$ into $n$ subintervals

$$\mathcal{I}_i \coloneqq \{x \in \Omega : x_{i-1} < x \leq x_i\} \ ,$$

and define the integrals of $f(x)$ over these subintervals as

$$F_i \coloneqq \int_{x_{i-1}}^{x_i} dx \ f(x) \ .$$

The $n$-dimensional vector $\mathbf{F}_\pi = [F_1, ..., F_n]$ then provides a finite-dimensional representation[1] of the function $f(x)$ on the specified grid $\pi$. If $\Pi$ is the set of all possible girds, then we get $\mathcal{F} = \{\mathbf{F}_\pi : \pi \in \Pi\}$ a family of all finite-dimensional representations of $f(x)$.

[1] We could have used instead the average values $f_i = F_i/(x_{i+1} - x_i)$. However, the function $f(x)$ has a finite integral, so its average value over an infinite interval (like the last interval for $\Omega = \mathbf{R}$) is zero, which leads to a loss of information on that interval when using the average. Moreover, specifying the consistency condition is much easier using the integrals than averages.

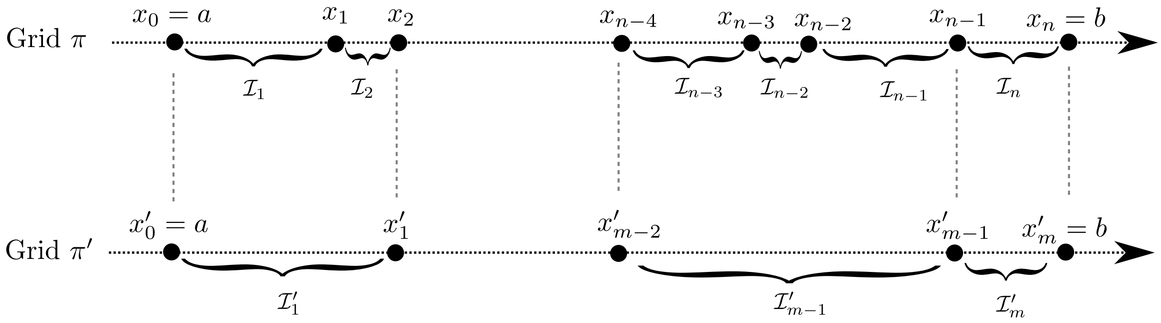


Figure A.1.: Schematic drawing of some grid $\pi$ and some coarsening of it $\pi'$.

We can now specify a probability distribution $P_\pi$ for each $\mathbf{F}_\pi$ and form a family of all finite-dimensional distributions $\mathcal{P} = \{P_\pi : \pi \in \Pi\}$. This family of finite dimensional distributions determines uniquely a stochastic process on the infinite-dimensional space of $f(x)$. Knowing that $f(x)$ is non-negative with a finite integral implies that $\mathbf{F}_\pi$ should also be non-negative and finite. Therefore, any probability distribution on the space of finite non-negative vectors is admissible. However, once this distribution is specified on a certain grid, the admissible probability distributions on other grids are restricted by the following consistency condition.

Let $\pi = \{x_0, ..., x_n\}$ be some grid with probability distribution $P_\pi$ and let the grid $\pi' = \{x'_0, ..., x'_m\}$ be some coarsening of this grid (see Fig. A.1). The intervals of $\pi'$ can be seen as merging of the intervals of $\pi$

$$\mathcal{I}'_1 = \bigcup_{i \in s_1} \mathcal{I}_i \quad , \quad ...... \quad , \quad \mathcal{I}'_{n'} = \bigcup_{i \in s_m} \mathcal{I}_i \ ,$$

where $s_1, ..., s_m$ are index sets of the merged intervals. Moreover, the integral of $f(x)$ over the coarse intervals is the sum of the the integrals over the corresponding fine intervals

$$F'_1 = \sum_{i \in s_1} F_i \quad , \quad ...... \quad , \quad F'_m = \sum_{i \in s_m} F_i \ .$$

We know that when $p(x, y)$ is a joint probability distribution of two random variables $x$ and $y$, then their sum $z := x + y$ has the probability distribution $q(z) = \int dxdy \ p(x, y)\delta(z - x - y)$. We can generalize this to obtain the probability distribution on the coarse grid $P_{\pi'}$ from the probability distribution on the fine one $P_\pi$ as

$$\begin{aligned} &P_{\pi'}(F'_1, ..., F'_m) \\ &\quad = \int dF_1 \ ... \ dF_n \quad P_\pi(F_1, ..., F_n) \quad \delta(F'_1 - \sum_{i \in s_1} F_i) \ ... \ \delta(F'_m - \sum_{i \in s_m} F_i) \ . \end{aligned} \tag{A.1}$$

Therefore, the probability distribution on some grid completely determines the probability distribution on any coarser grid by the above relation and the probability distribution on any finer grid is restricted such that it satisfies the above relation. This same idea was used to derive the priors implicitly assumed by StochS on different grids (see Sec. 3.3.3).

To sum up, a family $\mathcal{P}$ of probability distribution on all possible grids defines an admissible prior process for density-like functions $f(x)$ if:

1. Each probability distribution $P_\pi \in \mathcal{P}$ has support only on finite non-negative vectors.

2. The consistency condition (A.1) is satisfied for any two distributions $P_\pi, P'_\pi \in \mathcal{P}$, whenever grid $\pi'$ is a coarsening of grid $\pi$.

**Remark.** *The family of probability distributions need not be defined on all possible grids from the outset. It is sufficient to be defined on a rich enough set of grids that can be used to generate all possible grids. For example, suppose $\Omega = [0, 1]$, then the set of dyadic grids $\{0, \frac{1}{2}, 1\}, \{0, \frac{1}{4}, \frac{2}{4}, \frac{3}{4}, 1\}, ....,$ etc. can used to generate all possible grids because any real number can be arbitrarily closely approximated by dyadic rationals of the form $i/2^n$, where $i$ is an integer and $n$ is a natural number. Indeed, this is how the so-called* ***dyadic tailfree process*** *is defined [38, Sec. 1.10] and there are many other processes which are initially defined on such a set of grids and then extended naturally to all possible grids.*

## A.2. Examples of prior processes

### A.2.1. Gamma process

The gamma process is one of the simplest admissible processes. As the name suggests, it uses the gamma distribution, so we discuss it briefly first. The gamma distribution is a univariate distribution with two parameters, a shape parameter $\alpha > 0$ and a rate parameter $\beta > 0$, and it is supported on the positive real axis. It has the following probability density function

$$p(x;\ \alpha, \beta) = \frac{\beta^\alpha\ x^{\alpha-1}\ e^{-x\beta}}{\Gamma(\alpha)} \quad \text{for} \quad x \geq 0\ , \tag{A.2}$$

where $\Gamma(t)$ is the gamma function, and the following mean and variance

$$\mu = \frac{\alpha}{\beta}, \quad \sigma^2 = \frac{\alpha}{\beta^2}\ . \tag{A.3}$$

Note that when $\alpha = 1$, we get back the familiar exponential distribution. Fig. A.2 shows the gamma distribution for different shape and rate values.

The gamma distribution has an important summation property that the sum of independent gamma random variables with the same rate parameter is also a gamma random variable

$$X_i \sim \text{Gamma}(\alpha_i, \beta) \Rightarrow \sum_i X_i \sim \text{Gamma}\left(\sum_i \alpha_i, \beta\right)\ . \tag{A.4}$$

This property was proven for a simple case in Sec. 3.3.3 and used to simulate one StochS grid on another.

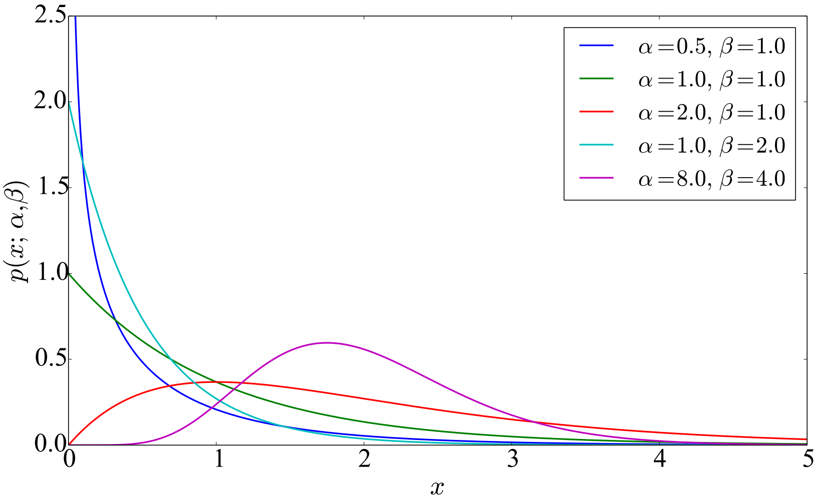


Figure A.2.: The probability density function of a gamma random variable for different values of shape parameter $\alpha$ and rate parameter $\beta$.

The **gamma process** has three parameters: (1) a positive real number $\lambda$, called the *rate parameter*, (2) a positive real number $\alpha$, called the *concentration parameter*, and (3) a normalized positive function $D$ defined on the domain $\Omega$, called the *default model.* This process is defined such that $F_i$, the integral of $f(x)$ over any interval $\mathcal{I}_i$ is a gamma random variable

$$F_i \sim \text{Gamma}\left(\alpha \int_{\mathcal{I}_i} dx\ D(x),\ \lambda\right)\ . \tag{A.5}$$

Moreover, whenever the intervals $\mathcal{I}_i$ and $\mathcal{I}_j$ are disjoint, the random variables $F_i$ and $F_j$ are independent. This definition satisfies the conditions for an admissible process. First, the samples are finite and non-negative. Second, due to the summation property of gamma variables, the distribution over fine intervals is consistent with the distribution over coarser ones. For example, let the integral over an interval $\mathcal{I}$ be $F$. Its distribution assigned from the outset is consistent with the distribution resulting from summing the integrals $F_i$ and $F_j$ over any two subintervals $\mathcal{I}_i$ and $\mathcal{I}_j$ forming a partitioning of $\mathcal{I}$

$$\begin{aligned} F = F_i + F_j \sim \text{Gamma}\left(\alpha \int_{\mathcal{I}_i} dx\ D(x) + \alpha \int_{\mathcal{I}_j} dx\ D(x),\ \lambda\right) = \\ \text{Gamma}\left(\alpha \int_{\mathcal{I}_i \cup \mathcal{I}_j} dx\ D(x),\ \lambda\right) = \\ \text{Gamma}\left(\alpha \int_{\mathcal{I}} dx\ D(x),\ \lambda\right)\ . \end{aligned}$$

**Relation to StochS** Interestingly, each gamma process has a special grid where the distributions of $F_i$ simplify to exponential distributions. We get this grid by dividing the domain $\Omega$ into $n$ intervals such that the integrals of the default model over these intervals are equal. Since the default model has unit integral, we have $\int_{\mathcal{I}_i} dx\ D(x) = 1/n$, and choosing $n = \alpha$ gives us the following distribution

$$F_i \sim \text{Gamma}(n\ \frac{1}{n},\ \lambda)\ \Rightarrow\ F_i \sim \text{Exp}(\lambda)\ .$$

Taking the zero limit of the rate parameter $\lambda \to 0$, the exponential distribution approaches a uniform distribution over all non-negative values which is nothing but the prior used by StochS! Therefore, applying StochS on a grid with $n$ points and whose density is $\rho(x)$ corresponds to using a gamma prior process with a default model $D(x) = \rho(x)$, a concentration parameter $\alpha = n$ and zero rate parameter $\lambda = 0$.

### A.2.2. Dirichlet process

The Dirichlet process is directly related to the gamma process and it is just a normalization of it. Let $f(x)$ be random functions drawn using the gamma process, then the normalized functions $f(x)/\int_\Omega dx f(x)$ follow the Dirichlet process. Therefore, the Dirichlet process can be used as a prior when we know that the function should be normalized like in the analytic continuation of Green functions.

As we will show later, the Dirichlet process can also be defined directly using the Dirichlet distribution. This a multivariate distribution defined on the unit simplex[2] and parametrized by a vector of positive real numbers $\boldsymbol{\alpha}$. It has the following probability density function

$$p(x_1, ..., x_n;\ \alpha_1, ..., \alpha_n) = \frac{\Gamma(\sum_i \alpha_i)}{\prod_i \Gamma(\alpha_i)}\ x_1^{\alpha_1 - 1} ...\ x_n^{\alpha_n - 1}\ , \tag{A.6}$$

and the following mean and variance

$$\mu_i = \frac{\alpha_i}{\alpha_0},\quad \sigma_i^2 = \frac{\alpha_i(\alpha_0 - \alpha_i)}{\alpha_0^2(\alpha_0 + 1)}\ , \tag{A.7}$$

where $\alpha_0 = \sum_i \alpha_i$. Note that when $\alpha_1 = ... = \alpha_n = 1$, we get a uniform distribution over the simplex. Fig. A.3 shows several Dirichlet distributions for $n = 3$.

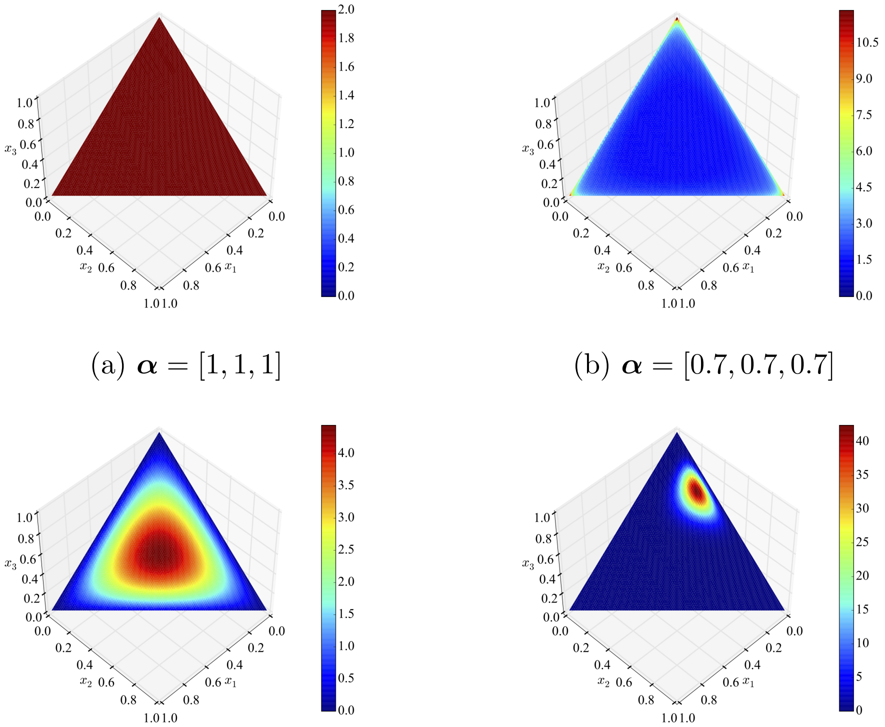
Similar to the relation between the processes, the Dirichlet distribution can be obtained by normalizing $n$ independent gamma distributions as following:

$$y_1 \sim \text{Gamma}(\alpha_1, \beta),\ ...,\ y_n \sim \text{Gamma}(\alpha_n, \beta) \tag{A.8}$$

$$\Rightarrow\quad \frac{y_1}{\sum_i y_i}, ..., \frac{y_n}{\sum_i y_i} \sim \text{Dir}(\alpha_1, ..., \alpha_n)\ , \tag{A.9}$$

Note that the parameters of the resulting Dirichlet distribution are independent of the rate parameter $\beta$. The summation property (A.4) of the gamma distribution translates

[2] It the set of vectors whose components are non-negative and sum up to one



(a) $\boldsymbol{\alpha} = [1, 1, 1]$

(b) $\boldsymbol{\alpha} = [0.7, 0.7, 0.7]$

(c) $\boldsymbol{\alpha} = [2, 2, 2]$

(d) $\boldsymbol{\alpha} = [3, 8, 20]$

Figure A.3.: The probability density function of the Dirichlet distribution on the unit simplex of $\mathbf{R}^3$. Note that when $\boldsymbol{\alpha} = [\alpha, \alpha, \alpha]$, the distribution is symmetric around the middle of the simplex. For $\alpha = 1$, we get the uniform distribution (case a). For $\alpha < 1$, the density is concentrated at the vertices (case b), while for $\alpha > 1$, it is concentrated in the middle.

into the following aggregation property of the Dirichlet distribution:

$$x_1, ..., x_n \sim \mathrm{Dir}(\alpha_1, ..., \alpha_n) \tag{A.10}$$

$$\Rightarrow x_1, ..., x_i + x_j, ..., x_n \sim \mathrm{Dir}(\alpha_1, ..., \alpha_i + \alpha_j, ..., \alpha_n) \ , \tag{A.11}$$

where the new distribution has $n - 1$ parameters.

The **Dirichlet process** has two parameters: (1) a positive real number $\alpha$, called the *concentration parameter*, and (2) a normalized positive function $D$ defined on the domain $\Omega$, called the *default model*. This process is defined such that $\mathbf{F}_\pi$, the integrals of $f(x)$ over grid $\pi$, follow the Dirichlet distribution

$$\mathbf{F}_\pi = [F_1, ..., F_n] \sim \mathrm{Dir}\left(\alpha \int_{\mathcal{I}_1} dx\ D(x),\ ...,\ \alpha \int_{\mathcal{I}_n} dx\ D(x)\right) \ . \tag{A.12}$$

This process is admissible because the samples are non-negative and finite and the consistency condition is satisfied due to the aggregation property of the Dirichlet distribution.

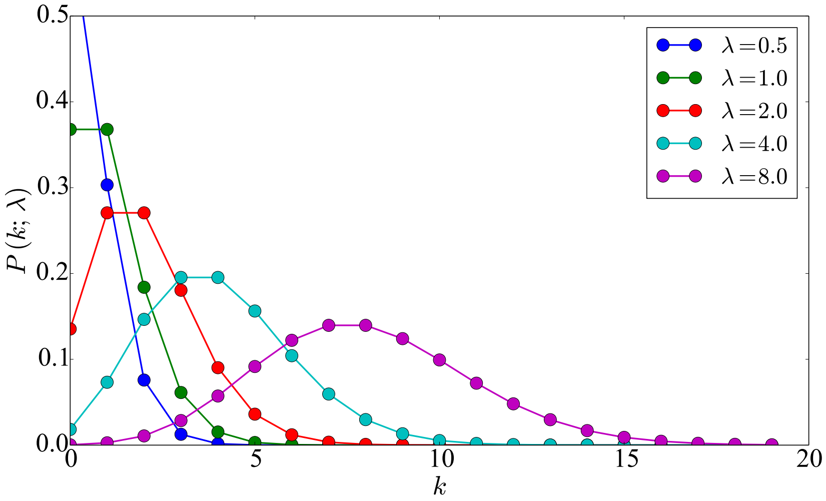


Figure A.4.: The probability mass function of a Poisson random variable for different rate parameters $\lambda$. Note that this function is defined only over non-negative integers.

Note that the samples of the Dirichlet process are always normalized. This is why it does not have a rate parameter like the gamma process. This parameter is canceled when the gamma process is normalized into a Dirichlet process.

**Relation to StochS** Similar to the gamma process, the Dirichlet process has a special grid where the finite-dimensional distribution becomes uniform. This grid is obtained by dividing the domain into $\alpha$ intervals where the integrals of the default model $D$ over the intervals are equal. This uniform prior over normalized vectors is the prior used by StochS when a sum rule is imposed.

### A.2.3. Poisson process

Another interesting process is the Poisson process. This process gives quantized functions whose integrals are multiples of a specific quantum $q$. It uses the Poisson distribution to describe the number of quanta in each interval. The Poisson distribution is a discrete distribution with the following probability mass function

$$P(k;\lambda) = \frac{\lambda^k e^{-\lambda}}{k!}\,, \tag{A.13}$$

where $\lambda$ is called the rate parameter and controls both the mean and variance of this distribution $\mu = \sigma = \lambda$. Similar to the gamma distribution, the Poisson distribution has the following summation property

$$k_i \sim \text{Poisson}(\lambda_i) \Rightarrow \sum_i k_i \sim \text{Poisson}\left(\sum_i \lambda_i\right)\,.$$

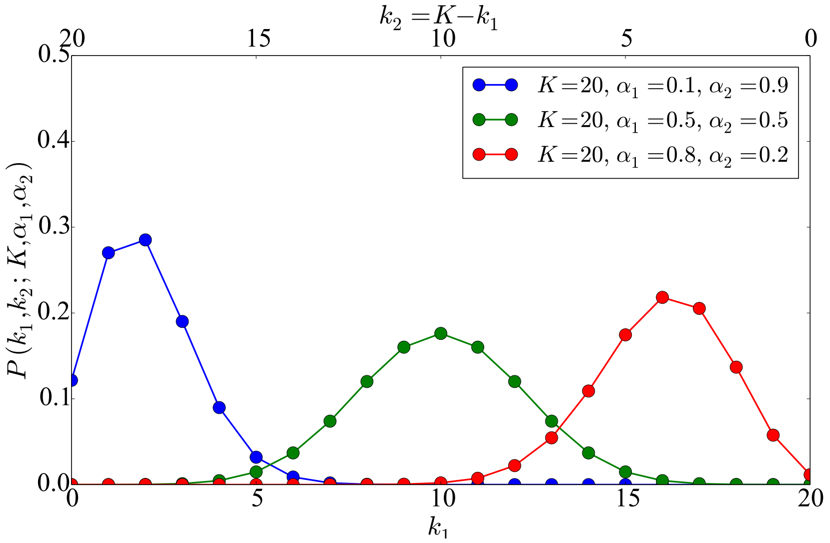


Figure A.5.: The probability mass function of a binomial random variable for different parameters. Note that parameters $\lambda_1, \lambda_2$ sum up to one and variables $k_1, k_2$ sum up to $K$.

In Fig. A.4, we show the Poisson distribution for different rate parameters.

The **Poisson process** has three parameters: (1) a positive real number $q$ called the *quantum*, (2) a positive real number $\alpha$ called the *concentration parameter*, and (3) a normalized positive function $D$ defined over the domain $\Omega$, called the *default model*. This process is defined such that the number of quanta $k_i$ in any interval $\mathcal{I}_i$ is following a Poisson distribution

$$F_i = k_i q \quad \text{where} \quad k_i \sim \text{Poisson}\left(\alpha \int_{\mathcal{I}_i} dx\ D(x)\right) . \tag{A.14}$$

Besides, $k_i$ is independent of $k_j$, the number of quanta in any other non-overlapping interval $\mathcal{I}_j$. As for the gamma process, the consistency of this definition follows from the summation property of Poisson random variables.

## A.2.4. Multinomial process

The multinomial process is defined in terms of the multinomial distribution. It is a multivariate discrete distribution parameterized by a positive integer $K$ and a vector of positive real numbers $\boldsymbol{\lambda}$ that sum up to one. It is defined on the set of non-negative integers that sum up to $K$ and has the following probability mass function

$$P(k_1, ..., k_n; K, \lambda_1, ..., \lambda_n) = \frac{K!}{k_1!...k_n!} \lambda_1^{k_1} ... \lambda_n^{k_n} . \tag{A.15}$$

Its mean and variance read

$$\mu_i = K\lambda_i \quad \sigma_i^2 = K\lambda_i(1 - \lambda_i) . \tag{A.16}$$

In Fig. A.5, we show the multinomial distribution for $n = 2$ (also known as the binomial distribution) and different parameters.

The multinomial distribution can be obtained from $n$ independent Poisson random variables $k_i$ by imposing a restriction on their sum $\sum_i k_i = K$ i.e. conditioning independent Poisson random variables on their sum gives a multinomial distribution as following

$$\begin{aligned} & k_1 \sim \text{Poisson}(\lambda_1), ..., k_n \sim \text{Poisson}(\lambda_n) \\ \Rightarrow \quad & k_1, ..., k_n | K \sim \text{Mult}(K, \lambda'_1, ..., \lambda'_n) \ , \end{aligned}$$

where $\lambda'_i = \lambda_i / \sum_j \lambda_j$. As a result, imposing a normalization constraint on the Poisson process translates into a restriction on the total number of quanta to $1/q$ and leads to the Multinomial process defined below.

The **multinomial process** has two parameters: (1) a positive integer $K$ called the number of quanta, and (2) a normalized positive function $D$ defined on the domain $\Omega$, called the *default model.* This process is defined such that the integrals of $f(x)$ over grid $\pi$ takes the form $\mathbf{F}_\pi = q[k_1, ..., k_n]$, with $q = 1/K$ and $k_1, ..., k_n$ following the multinomial distribution

$$k_1, ..., k_n \sim \text{Multi}\left(K, \int_{\mathcal{I}_1} dx\ D(x),\ ...,\ \int_{\mathcal{I}_n} dx\ D(x)\right) \ . \tag{A.17}$$

**Relation to MaxEnt** The multinomial process is important due to its relation to the MaxEnt method. Let us look at the logarithm of a multinomial probability

$$\ln P(k_1, ..., k_n) = \ln K! - \sum_i \ln k_i! + \sum_i k_i \ln \lambda_i$$

In the limit of very large $K$, we can use the Sirling's formula $\ln x! \approx x \ln x - x$ to get the following approximation

$$\begin{aligned} \ln P(k_1, ..., k_n) &\approx K \ln K - K - \sum_i (k_i \ln k_i - k_i) + \sum_i k_i \ln \lambda_i \\ &= \sum_i k_i \ln K - \sum_i k_i \ln k_i + \sum_i k_i \ln \lambda_i \\ &= -\sum_i k_i \ln \frac{k_i}{K\lambda_i} \ . \end{aligned}$$

For a multinomial process on a grid $\pi$, we have $\lambda_i = \int_{\mathcal{I}_i} dx\ D(x)$ and $k_i = F_i/q = KF_i$ so the probability distribution of the integrals $F_i = \int_{\mathcal{I}_i} dx\ f(x)$ reads

$$P(F_1, ..., F_n) \approx e^{KS(F_1,...,F_n)} \qquad \text{(limit of } K \to \infty) \tag{A.18}$$

where

$$S(F_1, ..., F_n) := -\sum_i F_i \ln \frac{F_i}{\int_{\mathcal{I}_i} dx\ D(x)} = -\sum_i \int_{\mathcal{I}_i} dx\ f(x) \ln \frac{\int_{\mathcal{I}_i} dx\ f(x)}{\int_{\mathcal{I}_i} dx\ D(x)} \ . \tag{A.19}$$

For a fine enough grid, a function can be approximated by its average values on the grid intervals, so the exponent can be written as

$$S \approx -\int_\Omega dx\ f(x) \ln \frac{f(x)}{D(x)}\ , \qquad \text{(limit of } n \to \infty) \tag{A.20}$$

which is easily recognizable as the entropy of the function $f(x)$ relative to the default model $D(x)$. To summarize, we can say that the finite-dimensional distribution on a fine grid of a multinomial process with a large number of quanta can be approximated with the prior

$$P(f) = e^{KS(f)}\ . \tag{A.21}$$

A clear problem with this approximation is that it is only valid in the limit $K \to \infty$, where the prior becomes extremely sharp around the default model suppressing any knowledge from the data [39]. To be more explicit, assuming Gaussian-distributed data with $\chi(f)$ representing the fit to the data $g$, the posterior distribution has the following form

$$P(f|g) \propto e^{KS(f)-\chi^2(f)/2}\ . \tag{A.22}$$

As $K$ increases to infinity, the entropy term becomes infinitely large in comparison to the fit term. A workaround this infinitely sharp prior is to make the data also "artifcially" sharp by introducing a parameter $\beta$ into the fit term and taking the limit $\beta \to \infty$. The trick to get a meaningful result is keeping the ratio $\alpha := K/\beta$ fixed

$$P(f|g) \propto e^{KS(f)-\beta\chi^2(f)/2} = e^{\beta\left[\alpha S(f)-\chi^2(f)/2)\right]}\ . \tag{A.23}$$

Since the exponent goes to infinity, the mean of this distribution is the same as its maximum which is nothing but the MaxEnt solution. To conclude, we identify the MaxEnt solution as a solution using a multinomial prior process in the limit of infinite number of quanta. In this limit, the prior is infinitely sharp, so the data is also made infinitely sharp keeping the ratio of the two fixed. When the functions are not normalized, the above argument can be repeated to show that MaxEnt with the generalized entropy

$$S = \int_\Omega dx\ f(x) - D(x) - f(x) \ln \frac{f(x)}{D(x)} \tag{A.24}$$

is similarly related to the Poisson process [39].

**Is MaxEnt Bayesian?** Using the entropy prior $e^{\alpha S(f)}$ on any specific grid is clearly valid. The problem is that moving to a coarser grid gives us generally a distribution that cannot be expressed as an entropy prior [39]. Therefore, the consistency condition (A.1) cannot be satisfied and the entropy prior is not admissible. Even in the previous paragraph where the entropy prior arises as the limit of the admissible multinomial or Poisson processes, it is valid only on very fine grids. More importantly, the data (i.e. the likelihood) has to be made infinitely sharp to get meaningful results, which is not justified in the Bayesian framework. This inconsistency of MaxEnt was missed earlier due to the side effects of a Gaussain approximation used in the algorithm. Nevertheless, it can be still seen and used as a powerful regularization method [26].

**Comments on Beach's identification of MaxEnt** In his paper [31], Beach identifies MaxEnt as a limit of stochastic analytic continuation. Taking Beach's identification at face value, it looks different from our identification, but they are actually the same. Beach argues in physical terms, while we argue in mathematical ones. He talks about a system of interacting classical fields at a fictitious temperature and treats the fit to that data as the energy. To make the connection to multinomial processes, think of these classical fields as the distribution of $K$ classical particles over $n$ energy levels where both $K$ and $n$ go to infinity. Then the multinomial distribution arises naturally as the number of ways of distributing $K$ classical particles into $n$ energy levels. The parallel of the "artificially" sharp data assumption is the average energy constraint implied by the fixed temperature. Finally, using the mean field solution corresponds to using the saddle point solution. This is similar to what we did by taking the maximum as a representative of the mean. It is worth pointing that Beach presents MaxEnt as a limit of "the" stochastic analytic continuation, failing to recognize that there are different stochastic methods corresponding to different priors and that his identification works only for the multinomial prior.[3] Moreover, he discusses two stochastic methods, none of which has the multinomial prior! The first one is Sandvik's original method [4] which is equivalent to StochS and has the gamma prior (or the Dirichlet prior for normalized models). The second one is a method suggested by Beach himself. We will show later in Sec. A.3.1 that Beach's method (which is equivalent to gStochS) corresponds to an extended form of the multinomial process.

## A.3. Discrete representation

An important subclass of processes[4] is discrete processes where the sampled function $f(x)$ can be represented as a countable sum of delta functions

$$f(x) = \sum_{i=1}^{K} w_i \; \delta(x - x_i) \; , \tag{A.25}$$

where $K, w_i$ and $x_i$ are random or deterministic variables. As surprising as it may seem, all the four prior processes discussed earlier are actually discrete ones.

For example, fixing the number of deltas to $K$ and setting $w_i = q = 1/K$, while drawing the positions $x_i$ identically and independently from a probability distribution $D(x)$ gives us the multinomial process. This is easy to understand by realizing that for any finite grid, the number of delta functions falling in the grid intervals must follow the multinomial distribution of Eq. A.17. Since the weights of the delta functions are fixed, the family $\mathbf{F}_\pi$, the integrals of $f(x)$ on finite girds $\pi$, forms a multinomial process.

For the Dirichlet process, the positions $x_i$ are also drawn identically and independently from the default model $D(x)$. However, the number of delta functions is infinite $K = \infty$

[3] He assumes an integration measure over normalized positive functions without realizing that such a measure is not unique.

[4] Not all admissible processes are discrete. For example, Polya tree process generates continuous samples. However, all the processes we are interested in are discrete.

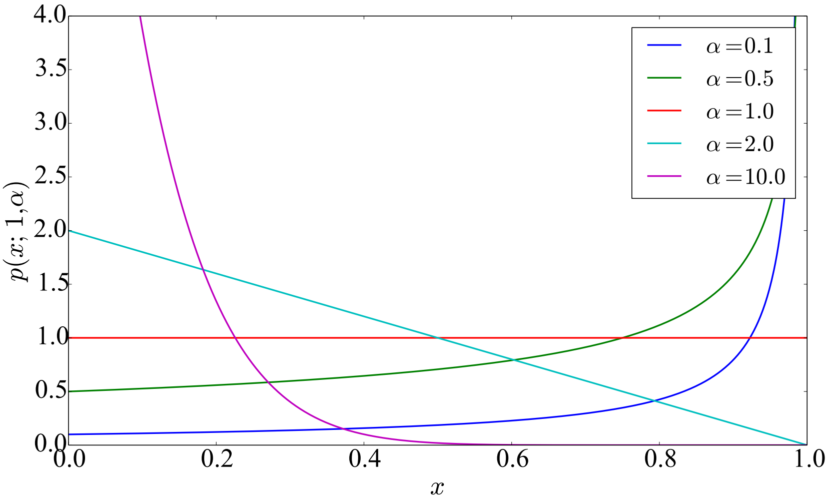


Figure A.6.: The probability density function of beta distribution $\mathrm{Beta}(1, \alpha)$ for different values of $\alpha$.

and the weights are defined as $w_i = \beta_i \prod_{j=1}^{i-1}(1-\beta_j)$ where random variables $\beta_j$ are drawn interdependently and identically from the beta distribution $\mathrm{Beta}(1, \alpha)$ (see Fig. A.6). You can imagine generating the weights as taking a stick of length one and breaking it at $\beta_1$. The length of the first piece $\beta_1$ is taken to be the first weight $w_1$. Then we take the other piece (whose length is $1 - \beta_1$) and break it at $\beta_2$ of its length to obtain two pieces. The length of the first $(1-\beta_1)\beta_2$ is used as the second weight $w_2$ while the other piece is broken recursively to obtain the other weights $w_3, w_4, ...$, etc. This is called the stick-breaking construction of a Dirichlet process and it can be shown to be equivalent to the earlier definition [40]. Using this construction in practice, requires approximating the Dirichlet samples using a finite number of delta functions $K$. Fig. A.7 shows the concentration parameter $\alpha$ versus the average number of delta functions required to cover 99% of the total weight of a Dirichlet process sample. Notice that the relation is linear: $K \approx 4\ \alpha$. A discrete representation of the gamma process is more complicated and can be found in [41].

## A.3.1. Discrete exponential process

As we have seen, the multinomial process has a fixed number $K$ of delta functions whose weights are fixed to $w_i = 1/K$ and whose positions are drawn identically and independently from a default model $D(x)$

$$\begin{aligned} f(x) &= \sum_{i=1}^{K} w_i\ \delta(x - x_i)\ , \\ \text{where:} \quad w_i &= 1/K \\ x_i &\overset{iid}{\sim} D(x)\ . \end{aligned}$$

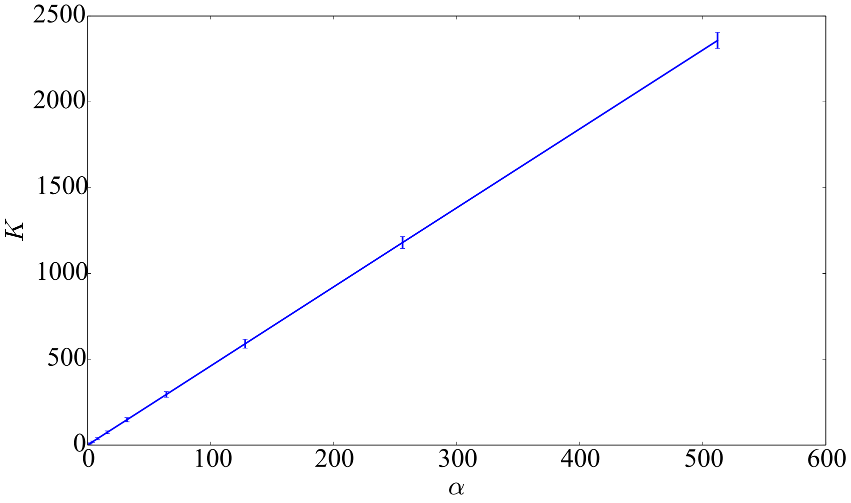


Figure A.7.: The number of delta functions required using stick-breaking construction to get 99% of the total weight of a Dirichlet process with concentration parameter $\alpha$. Since this number is not deterministic, we plotted the mean with its standard deviation as an error bar.

We can extend this process by varying not only the positions but also the weights. The most straightforward way to vary the weights is to draw them identically and independently from an exponential distribution $\text{Exp}(\lambda)$. This choice is one of the simplest distributions over non-negative values and it allows us to get a flat distribution over non-negative values by taking the limit $\lambda \to 0$ later.[5] The positions $x_i$ are drawn as always identically and independently from a default model $D(x)$.

To put this process on equal footing with earlier processes and to illustrate how one can move from the discrete representation to the grid representation, we will derive the finite-dimensional distribution of the integrals $\mathbf{F}_\pi$ on any grid $\pi$. For simplicity, we focus first on a single grid interval $\mathcal{I}$. Assuming that $F$ is the integral of $f(x)$ over $\mathcal{I}$, we want the probability distribution of $F$ denoted as $P_\mathcal{I}(F)$. Since $F$ equals the sum of the weights of delta functions falling in $\mathcal{I}$

$$F = \sum_{x_i \in \mathcal{I}} w_i \ ,$$

the desired probability equals the sum over all ways that delta functions can fall in $\mathcal{I}$ such that the sum of their weights equal $F$

$$P_\mathcal{I}(F) = \sum_{l=0}^{K} C(l) * p_1(l) * p_2(l) \ ,$$

[5] Taking this limit is only sensible after forming the posterior where the data would determine the total weight.

where $C$ is the number of ways of choosing $l$ delta functions out of $K$ ones, $p_1$ is the probability that $l$ delta functions are in $\mathcal{I}$ and $p_2$ is the probability that the sum of their weights equals $F$.

The number $C$ is simply the binomial coefficient

$$C(l) = \binom{K}{l} \; .$$

The probability that a delta function falls in $\mathcal{I}$ is

$$q := \int_{\mathcal{I}} dx \; D(x) \; .$$

Therefore, the probability of exactly $l$ delta functions falling in $\mathcal{I}$ reads

$$p_1(l) = q^l (1-q)^{K-l} \; .$$

Finally, the probability that the sum of $l$ weights equals $F$ reads

$$\frac{1}{\lambda^l} \int dw_1 ... dw_l \; e^{-\frac{\sum_i w_i}{\lambda}} \delta \left( \sum_i w_i - F \right) = \frac{1}{\lambda^l} e^{-\frac{F}{\lambda}} \frac{F^{l-1}}{\Gamma(l)} \; .$$

This formula works only for $l > 0$, because when no delta function falls inside the interval $\mathcal{I}$, the integral is zero with probability one. So we have

$$p_2(l) = \begin{cases} \delta(F) & l = 0 \\ \lambda^l e^{-\lambda F} \frac{F^{l-1}}{\Gamma(l)} & 0 < l \leq K \end{cases} .$$

Putting things together, we get

$$P_{\mathcal{I}}(F) = (1-q)^K \delta(F) \; + \sum_{l=1}^{K} \frac{K!}{l!(K-l)!} \, q^l (1-q)^{K-l} \, \lambda^l e^{-\lambda F} \frac{F^{l-1}}{\Gamma(l)} \; . \qquad \text{(A.26)}$$

Extending this result to a grid of two intervals is straightforward. Let us re-denote $l, q$ and $F$ from above as $k_1, D_1$ and $F_1$, respectively. Besides, denote the number of delta functions in the complement interval of $\mathcal{I}$ as $k_2$, the integral of $D(x)$ over it as $D_2$ and the integral of $f(x)$ as $F_2$. The joint probability density of $F_1$ and $F_2$ can then be written as

$$\begin{aligned} P_{\mathcal{I}}(F_1, F_2) = \sum_{k_1, k_2 = 0}^{K} & \delta_{k_1 + k_2, K} \frac{D_1^{k_1} {D_2}^{k_2}}{k_1! k_2!} \\ & \left[ \delta_{k_1,0} \; \delta(F_1) + (1 - \delta_{k_1,0}) \; \lambda^{k_1} e^{-\lambda F_1} \frac{F_1^{k_1 - 1}}{\Gamma(k_1)} \; \right] \qquad \text{(A.27)} \\ & \left[ \delta_{k_2,0} \; \delta(F_2) + (1 - \delta_{k_2,0}) \; \lambda^{k_2} e^{-\lambda F_2} \frac{F_2^{k_2 - 1}}{\Gamma(k_2)} \; \right] \; . \end{aligned}$$

This can be generalized easily to the n-dimensional case

$$
\begin{aligned}
P_{\mathcal{I}}(F_1, ..., F_n) = \sum_{k_1,...,k_n=0}^{K} \delta_{\sum_i k_i, K} \frac{D_1^{k_1}}{k_1!} \; ... \; \frac{D_n^{k_n}}{k_n!} & \\
\left[\delta_{k_1,0}\ \delta(F_1) + (1-\delta_{k_1,0})\ \lambda^{k_1} e^{-\lambda F_1} \frac{F_1^{k_1-1}}{\Gamma(k_1)}\right] & \cdot \\
..... \left[\delta_{k_n,0}\ \delta(F_n) + (1-\delta_{k_n,0})\ \lambda^{k_n} e^{-\lambda F_n} \frac{F_n^{k_n-1}}{\Gamma(k_n)}\right] &
\end{aligned}
\tag{A.28}
$$

It is important to note, that although we are able to derive the distribution on a grid, it is still far more efficient to draw samples and perform averaging using the discrete representation, as done in the algorithm of gStochS.

To get normalized functions from the previous process, we can impose normalization on the total weight and take the limit $\lambda \to 0$. This allows the weights to vary uniformly while summing up to one, which is equivalent to drawing them from the flat Dirichlet distribution $\text{Dir}(1, 1, ..., 1)$[6]. The finite-dimensional distribution of integrals on a grid can be derived similar to the previous process. We only need to impose normalization and recalculate the probability that the sum of $l$ weights equals $F$. The probability distribution of the weights is $\text{Dir}(1, 1, ..., 1)$, so the probability distribution of the sum of $l$ weights, using the aggregation property of Dirichlet distribution, is $\text{Dir}(l, K-l)$ which reads

$$\frac{\Gamma(K)}{\Gamma(l)\ \Gamma(K-l)}\ F^{l-1}\ (1-F)^{K-l-1}\ ,$$

and we get the following distribution of integrals on a grid with $n$ intervals

$$
\begin{aligned}
P_{\mathcal{I}}(F_1, ..., F_n) = \delta\left(\sum_i F_i - 1\right) \Gamma(K) K! \sum_{k_1,...,k_n=0}^{K} \delta_{\sum_i k_i, K} \frac{D_1^{k_1}}{k_1!} \; ... \; \frac{D_n^{k_n}}{k_n!} & \\
\left[\delta_{k_1,0}\ \delta(F_1) + (1-\delta_{k_1,0})\ \frac{F_1^{k_1-1}}{\Gamma(k_1)}\right] \; ... \; \left[\delta_{k_n,0}\ \delta(F_n) + (1-\delta_{k_n,0})\ \frac{F_n^{k_n-1}}{\Gamma(k_n)}\right] & 
\end{aligned}
\ . \tag{A.29}
$$

**Relation to Beach's method** From the discrete representation, it is easy to see that this process corresponds to the method suggested by Beach (without parallel tempering) [31]. We should point out that Beach does not consider the number of delta functions as a parameter. Instead, he assumes that he is approximating an integration measure and that the more delta functions, the better the approximation. However, this assumption is not valid, because it can be shown [43] that in the limit of $K \to \infty$, this process (or integration measure) becomes concentrated at the default model $D(x)$, suppressing any knowledge from the data.

[6] It is worth noting that using the distribution $\text{Dir}(\alpha/K, \alpha/K, ..., \alpha/K)$ with $K \to \infty$ instead of the flat one gives the Dirichlet process [42].

**Relation to gStochS and eStochS** Since gStochS with delta functions mapping is technically equivalent to Beach's method, the discrete exponential process is also the stochastic process of gStochS where the grid size equals the number of delta functions $n$ and the default model $D(x)$ equals the prior density $p(x)$. It is also the stochastic process of eStochS if we redefine the probability of a delta function falling in the interval $\mathcal{I}_i$ as $D_i := \int dw \int dx D(x; w)$; this accounts for the averaging over the width of the default model. Using mappings other than delta functions, would lead to different but similar stochastic processes. The principle for deriving these process is, however, the same.

# B. Analytic Continuation of Non-Diagonal Spectral Functions

Assuming we have an algorithm for the analytic continuation of diagonal elements only. Could we use it to reconstruct non-diagonal elements? To answer this question, let us take a $2 \times 2$ spectral matrix of fermionic Green functions

$$\mathbf{A}(\omega) = \begin{bmatrix} A_{11}(\omega) & A_{12}(\omega) \\ A^*_{12}(\omega) & A_{22}(\omega) \end{bmatrix}$$

We can transform this matrix using a unitary transformation $\mathbf{U}$ to get another matrix

$$\tilde{\mathbf{A}}(\omega) = \mathbf{U}^\dagger \mathbf{A}(\omega) \mathbf{U} \,.$$

Since the matrices $\mathbf{A}, \tilde{\mathbf{A}}$ are Hermitian, the only non-diagonal element we need to calculate is the upper triangular one $A_{1,2}$, which is completely determined by the diagonal elements $A_{11}, A_{22}, \tilde{A}_{11}, \tilde{A}_{22}$. So, the non-diagonal elements of a $2 \times 2$ matrix can be calculated using its diagonal elements in two different bases.

Now Suppose we have a $3 \times 3$ matrix

$$\mathbf{A}(\omega) = \begin{bmatrix} A_{11}(\omega) & A_{12}(\omega) & A_{13}(\omega) \\ A^*_{12}(\omega) & A_{22}(\omega) & A_{23}(\omega) \\ A^*_{13}(\omega) & A^*_{23}(\omega) & A_{33}(\omega) \end{bmatrix}$$

To calculate $A_{12}$, $A_{13}$ and $A_{23}$, we apply the above idea to each of the following sub-matrices, respectively

$$\begin{bmatrix} A_{11}(\omega) & A_{12}(\omega) \\ A^*_{12}(\omega) & A_{22}(\omega) \end{bmatrix}, \begin{bmatrix} A_{11}(\omega) & A_{13}(\omega) \\ A^*_{13}(\omega) & A_{33}(\omega) \end{bmatrix}, \begin{bmatrix} A_{22}(\omega) & A_{23}(\omega) \\ A^*_{23}(\omega) & A_{33}(\omega) \end{bmatrix} \,.$$

In general, the non-diagonal elements of higher dimensional matrices are determined by the diagonal elements of the $2 \times 2$ sub-matrices containing those elements in two different bases.

So far we have demonstrated that it is possible in principle to reconstruct the whole spectral matrix using only diagonal elements in different bases. If those diagonal elements were constructed out of exact Green functions without noise or numerical errors, the reconstructed matrix will be positive semidefinite since all spectral matrices are positive semidefinite. However, the Green function data is incomplete and has noise, so we may ask the following question: is it enough to impose the non-negativity of the analytically continued diagonal elements to ensure the positive semidefinite nature of the reconstructed matrix? The answer is NO! We will demonstrate this by showing that a $2 \times 2$ matrix could have positive diagonal elements in two different bases (even in infinitely many bases) and still not be positive semidefinite. As a result, imposing the non-negativity of diagonal elements in a finite number of bases cannot guarantee the positive semi-definiteness of the spectral matrix.

Let us consider the following matrix

$$\mathbf{M} = \begin{bmatrix} 1 & 0 \\ 0 & -\epsilon \end{bmatrix}$$

where $\epsilon$ is a very small positive number. This matrix is clearly not positive semidefinite because it has a negative eigenvalue $-\epsilon$. Now let us rotate this matrix by an angle $\theta$ using the transformation

$$\mathbf{U}(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$

The transformed matrix reads

$$\mathbf{M}(\theta) = \mathbf{U}^{\dagger}(\theta)\mathbf{A}\mathbf{U}(\theta) = \begin{bmatrix} \cos^2(\theta) - \epsilon\sin^2(\theta) & -\cos(\theta)\sin(\theta)(1+\epsilon) \\ -\cos(\theta)\sin(\theta)(1+\epsilon) & \sin^2(\theta) - \epsilon\cos^2(\theta) \end{bmatrix}$$

Let us take two transformations for different angles $\theta_1 = 45°$ and $\theta_2 = 30°$

$$\mathbf{M}(\theta_1) = \frac{1}{2}\begin{bmatrix} 1-\epsilon & -(1+\epsilon) \\ -(1+\epsilon) & 1-\epsilon \end{bmatrix}$$
$$\mathbf{M}(\theta_2) = \frac{1}{4}\begin{bmatrix} 3-\epsilon & -\sqrt{3}(1+\epsilon) \\ -\sqrt{3}(1+\epsilon) & 3-\epsilon \end{bmatrix}$$

Both of these matrices have positive diagonal elements and they are related by a rotation of $\theta_1 - \theta_2 = 15°$. Nevertheless, they are not positive semidefinite. Therefore, imposing non-negativity of diagonal elements in two different bases is not sufficient to guarantee the positive semi-definiteness of the matrix.

There are actually an infinite number of bases where the above matrix can have non-negative diagonal elements. These bases are rotated by any angle that satisfy the following conditions

$$\cos^2(\theta) - \epsilon\sin^2(\theta) \geq 0 \Rightarrow \cos^2(\theta) \geq \frac{\epsilon}{2}$$
$$\sin^2(\theta) - \epsilon\cos^2(\theta) \geq 0 \Rightarrow \sin^2(\theta) \geq \frac{\epsilon}{2}$$

In Fig. B.1 we show on the unit circle, the angles $\theta$ for which the matrix $\mathbf{M}(\theta)$ has non-negative diagonal elements. Therefore, a matrix could have non-negative diagonal elements in an infinite number of basis but still not be positive semidefinite.

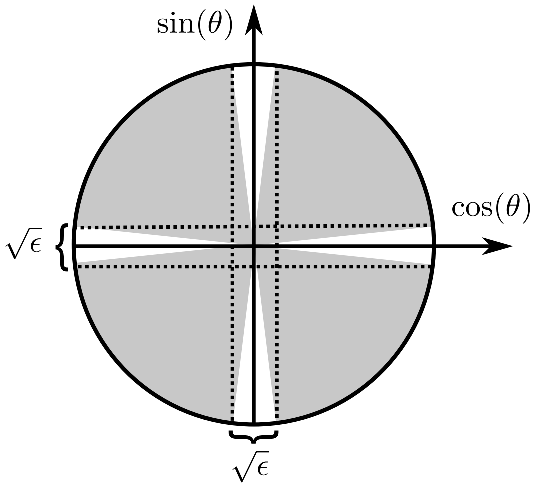


Figure B.1.: The shaded sectors of unite circle correspond to angles $\theta$ for which the matrix $\mathbf{M}(\theta)$ has non-negative diagonal elements.

## Topology of positive semi-definite matrices

It is easy to show that the sum of positive semi-definite matrices is also a positive semi-definite matrix. A Hermitian matrix $\mathbf{M}$ is positive semi-definite if and only if $z^T\mathbf{M}z \geq 0$ for any vector $z$. Let $\mathbf{A}$ and $\mathbf{B}$ be two positive semi-definite matrices, then they are Hermitian and satisfy $z^T\mathbf{A}z \geq 0$ and $z^T\mathbf{B}z \geq 0$, respectively. Their sum $\mathbf{A}+\mathbf{B}$ is then also Hermitian and satisfies $z^T(\mathbf{A}+\mathbf{B})z \geq 0$. Therefore, the sum is positive semi-definite.

Moreover, let $\lambda \in [0,1]$, then

$$z^T\left[\lambda\mathbf{A} + (1-\lambda)\mathbf{B}\right]z = \lambda z^T\mathbf{A}z + (1-\lambda)z^T\mathbf{B}z \geq 0$$

Therefore, $\lambda\mathbf{A} + (1-\lambda)\mathbf{B}$ is positive semi-definite, and positive semi-definite matrices form a convex set.

These two results are important for extending StochS to spectral matrices. The first one ensures that the average result is a positive definite matrix. The second one allows us to sample the space of positive definite matrices by varying one matrix element at a time and ensures that the allowed values for this element is a connected interval.

**Note** The Green function has diagonal structure for all values iff the spectral function has diagonal structure for all values. The idea of simultaneously diagonalizing spectral matrices for all $\omega$ does not work in general because in general we cannot find a single transformation that simultaneously diagonalize the spectral matrices of all $\omega$.

# Bibliography

[1] M. Jarrell and J. E. Gubernatis, Phys. Rep. **269**, 133 (1996).

[2] O. Gunnarsson, M. Haverkort, and G. Sangiovanni, Phys. Rev. B **82**, 165125 (2010).

[3] O. F. Syljuåsen, Phys. Rev. B **78**, 174429 (2008).

[4] A. W. Sandvik, Phys. Rev. B **57**, 10287 (1998).

[5] J. J. Kelly, *Graduate Mathematical Physics, With MATHEMATICA Supplements*, Wiley, 2006.

[6] M. Sing, Introduction to photoemission spectroscopy, in *DMFT at 25: Infinite Dimensions: Lecture Notes of the Autumn School on Correlated Electrons 2014*, edited by E. Pavarini, E. Koch, D. Vollhardt, and A. Lichtenstein, Schriften des Forschungszentrums Jülich Reihe Modeling and Simulation, Forschungszentrum Jülich, 2014.

[7] D. A. B. Miller, *Quantum Mechanics for Scientists and Engineers*, Cambridge University Press, 2008.

[8] A. L. Fetter and J. D. Walecka, *Quantum theory of many-particle systems*, International series in pure and applied physics, McGraw-Hill, 1971.

[9] G. Baym and N. D. Mermin, J. Math. Phys. **2**, 232 (1961).

[10] W. Rudin, *Real and complex analysis*, Mathematics series, McGraw-Hill, 1987.

[11] H. Bruus and K. Flensberg, *Many-Body Quantum Theory in Condensed Matter Physics: An Introduction*, Oxford Graduate Texts, OUP Oxford, 2004.

[12] C. P. Simon and L. Blume, *Mathematics for Economists*, Norton, 1994.

[13] P. C. Hansen, Inverse Prob. **8**, 849 (1992).

[14] J. Waldvogel, Towards a general error theory of the trapezoidal rule, in *Approximation and Computation*, edited by W. Gautschi, G. Mastroianni, and T. M. Rassias, volume 42 of *Springer Optimization and Its Applications*, pages 267–282, Springer New York, 2011.

[15] C. Schwartz, J. Comput. Phys. **4**, 19 (1969).

[16] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Johns Hopkins Studies in the Mathematical Sciences, Johns Hopkins University Press, 1996.

[17] P. C. Hansen, *Discrete inverse problems: insight and algorithms*, volume 7, SIAM, 2010.

[18] W. H. Press, *Numerical Recipes: The Art of Scientific Computing*, Cambridge University Press, 2007.

[19] D. Bergeron and A.-M. S. Tremblay, Phys. Rev. E **94**, 023303 (2016).

[20] P. C. Hansen, SIAM Rev. **34**, 561 (1992).

[21] L. Florack, R. Duits, and J. Bierkens, Tikhonov regularization versus scale space: a new result [image processing applications], in *Image Processing, 2004. ICIP '04. 2004 International Conference on*, volume 1, pages 271–274 Vol. 1, 2004.

[22] C. L. Lawson and R. J. Hanson, *Solving least squares problems*, volume 161, SIAM, 1974.

[23] D. Chen and R. J. Plemmons, Nonnegativity constraints in numerical analysis, in *Symposium on the Birth of Numerical Analysis*, pages 109–140, 2009.

[24] M. A. Hanson, J. Math. Anal. Appl. **80**, 545 (1981).

[25] H. W. Kuhn, Nonlinear programming: a historical view, in *Traces and Emergence of Nonlinear Programming*, pages 393–414, Springer, 2014.

[26] D. Sivia and J. Skilling, *Data Analysis: A Bayesian Tutorial*, Oxford science publications, OUP Oxford, 2006.

[27] K. Vafayi and O. Gunnarsson, Phys. Rev. B **76**, 035115 (2007).

[28] C. Robert, Stat. Comput. **5**, 121 (1995).

[29] G. Casella and E. I. George, Am. Stat. **46**, 167 (1992).

[30] G. O. Roberts and A. F. M. Smith, Stoch. Proc. Appl. **49**, 207 (1994).

[31] K. S. D. Beach, eprint arXiv:cond-mat/0403055 (2004).

[32] R. E. Kass and A. E. Raftery, J. Am. Stat. Assoc. **90**, 773 (1995).

[33] A. E. Raftery, M. A. Newton, J. M. Satagopan, and P. N. Krivitsky, Estimating the integrated likelihood via posterior simulation using the harmonic mean identity, in *Bayesian Statistics 8*, pages 1–45, 2006.

[34] A. S. Mishchenko, Stochastic optimization method for analytic continuation, in *Correlated Electrons: From Models to Materials: Lecture Notes of the Autumn School on Correlated Electrons 2012*, edited by E. Pavarini, E. Koch, F. Anders, and M. Jarrell, Schriften des Forschungszentrums Jülich Reihe Modeling and Simulation, Forschungszentrum Jülich, 2012.

[35] A. S. Mishchenko, N. V. Prokof'ev, A. Sakamoto, and B. V. Svistunov, Phys. Rev. B **62**, 6317 (2000).

[36] J. Becker et al., eprint ArXiv:1703.04652 (2017).

[37] S. Sibisi and J. Skilling, J. R. Stat. Soc. Series B Stat. Methodol. **59**, 217 (1997).

[38] E. G. Phadia, *Prior Processes and Their Applications: Nonparametric Bayesian Estimation*, SpringerLink : Bücher, Springer Berlin Heidelberg, 2013.

[39] J. Skilling and S. Sibisi, Priors on measures, in *Maximum Entropy and Bayesian Methods*, pages 261–270, Springer, 1996.

[40] J. Sethuraman, Stat. Sin. **4**, 639 (1994).

[41] A. Roychowdhury and B. Kulis, Gamma processes, stick-breaking, and variational inference, in *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, pages 800–808, PMLR, 2015.

[42] H. Ishwaran and M. Zarepour, Can. J. Stat. **30**, 269 (2002).

[43] H. Ishwaran and M. Zarepour, Stat. Sin. **12**, 941 (2002).