Meson spectroscopy at non-zero temperature using lattice QCD

Sergio Chaves García-Mascaraque

Department of Physics,
Swansea University

September 2022

Submitted to Swansea University in fulfilment of the requirements for the Degree of Doctor of Philosophy

Swansea University
Prifysgol Abertawe

A Paula, a mi familia, y al futuro

Declaration

This work has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

- **Signed:** Sergio Chaves García-Mascaraque
- **Date:** 30th, September 2022

This thesis is the result of my own investigations, except where otherwise stated. Other sources are acknowledged by giving explicit references. A bibliography is appended.

- **Signed:** Sergio Chaves García-Mascaraque
- **Date:** 30th, September 2022

I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

- **Signed:** Sergio Chaves García-Mascaraque
- **Date:** 30th, September 2022

The University's ethical procedures have been followed and, where appropriate, that ethical approval has been granted.

- **Signed:** Sergio Chaves García-Mascaraque
- **Date:** 30th, September 2022

Abstract

This thesis explores two main topics: the effects of the temperature on several Quantum Chromodynamics mesonic observables, with a concrete focus on the temperature dependence of the mesonic mass spectrum, and numerical spectral reconstruction of lattice correlation functions employing deep neural networks. In the first two chapters, a brief introduction to standard lattice Quantum Chromodynamics and non-zero temperature field theory is provided. Using the tools presented in the introductory chapters, a complete spectroscopy analysis of the temperature dependence of several mesonic ground state masses is developed. From this study, novel results in the restoration of chiral symmetry as a function of the temperature are obtained by studying the degree of degeneracy between the $\rho(770)$ and $a_1(1260)$ states. Additionally, a complete study of the thermal effects affecting the mesonic $D_{(s)}$-sector below the pseudocritical temperature of the system is provided. A self-contained chapter discussing the pion velocity in the medium is also included in the document. The pion velocity is estimated as a function of the temperature using non-zero temperature lattice Quantum Chromodynamics. In addition, after providing a detailed introduction to the field of neural networks, their application to numerical spectral reconstruction is studied. A simple implementation in which deep neural networks are applied to numerical spectral reconstruction is tested in order to explore its limits and applicability.

Structure of the document

This document is structured as follows.

In Chapter (1), Quantum Chromodynamics and its lattice regularisation are briefly introduced, as all studies included in this thesis are performed on, or are related to, the framework of lattice Quantum Chromodynamics at non-zero temperature. In this first chapter, the effects of the temperature on the system are not taken into account — we review non-zero temperature quantum field theory in Chapter (2). After shortly introducing the basic concepts of quantum field theory and its lattice regularisation, we discuss how lattice quantum field theories can be simulated in a computer. From this discussion, we learn how one could estimate correlation functions on the lattice, one of the fundamental objects in the study of quantum field theories. We conclude this chapter analysing the so-called spectral decomposition of correlation functions.

In Chapter (2), thermal field theory is briefly reviewed; non-zero density thermal field theory is not discussed in this document. After introducing the basic concepts of thermal field theory, we define and analyse spectral functions, which are essential in the study of non-static quantities in thermal field theory, such as transport coefficients or viscosities. At the end of this chapter, we explore a connection between low temperature spectral functions and the spectral decomposition defined in Chapter (1).

Chapter (3) contains the bulk of studies performed of lattice correlation functions. In this chapter, we aim at exploring the temperature dependence of several meson masses from the analysis of different lattice correlation functions. Due to the fact that extracting meson masses from lattice correlation functions is inherently difficult, we develop an original methodology to analyse lattice correlation functions in order to extract their ground state masses. After the methodology is introduced in detail, we present some results obtained in our lattice setup, introduced in Appendix (A). In addition, a detailed discussion of the mesonic correlation functions analysed is provided in Appendix (B).

From the results obtained by applying our novel methodology, we are able to

analyse the temperature dependence of mesonic ground state masses before and after the pseudocritical temperature of the system. The results are divided into three main parts: in the first one, we discuss general trends encountered in the temperature dependence of ground state masses, which allows us to set some limits in the viability of the methodology as a function of the temperature; in the second part, we analyse the restoration of chiral symmetry as a function of the temperature from the degeneracy of the physical mesonic states $\rho(770)$ and $a_1(1260)$; to conclude, we perform a complete study over the mesonic $D_{(s)}$ sector across the hadronic phase of Quantum Chromodynamics. Our results in the restoration of chiral symmetry from a mesonic point of view, and the $D_{(s)}$ mesonic sector represent one of the few first-principles analysis on these two topics across multiple temperatures.

Chapter (4) presents a self-contained analysis of the pion velocity in the medium at different temperatures: the pion velocity is a chiral limit expression that describes the effects of the temperature on the pion dispersion relation. This chapter includes a complete derivation of the pion velocity expression, as well as a collection of results estimated in our lattice setup.

A brief introduction to the field of neural networks is provided in Chapter (5). In addition to presenting neural networks, this chapter also discusses how to train neural network models in order to apply them to real-world problems. A small introduction to convolutional neural networks is also provided in this chapter. This chapter serves as an introduction to Chapter (6).

To conclude, Chapter (6) presents a case-study in which neural networks are applied to the field of spectral reconstruction. After shortly introducing the inherent problems of spectral reconstruction, we construct a methodology that employs neural networks as the core objects to construct spectral functions. Once the methodology is presented, we report and discuss the results obtained from the application of the methodology to the problem in question in a controlled environment. Appendices (C) and (D) serve as complements to this chapter.

In this document, all quantities are expressed in natural units:

$$\hbar = c = k_B = 1.$$

Contents

Chapter 1

Quantum-chromodynamics and its lattice regularisation

The strong force between quarks and gluons is described by the theory of Quantum Chromodynamics (QCD) [1]. Since its advent in the 1970s, it has proven to provide an exceptional description of the vast majority of hadronic matter encountered in experiments [2]. Even though a perturbative analysis of QCD at relatively low temperature is not possible, due to its confining nature, some techniques, such as lattice QCD [3], allow for a reliable and systematic first-principles study of the theory at zero temperature.

However, no single theoretical method is known to reliably explore QCD at non-zero temperature and density. As a result, the complete phase-diagram of the system is unknown; see Ref. [4] for a sketch of the QCD phase-diagram. Fortunately, lattice QCD simulations can be employed to explore QCD at non-zero temperature and zero density. The temperature is strongly believed [5, 6] to make the QCD system transition from the so-called hadronic phase, where QCD is expected to be a confining theory, to the Quark-Gluon Plasma (QGP) phase, where thermal effects dominate and, therefore, quarks might not form bound states. In fact, some lattice studies [7–9] suggest that the transition from one phase to the other is indeed analytical: no order parameter is present in the transition.

In this thesis, we continue to explore QCD at non-zero temperature through the analysis of mesonic observables extracted from lattice simulations. We restrict ourselves to zero chemical potential. Before diving into the relevant studies performed, QCD and its lattice regularisation are briefly introduced. Once the lattice regularisation of QCD is presented, we discuss how to simulate it in a computer in order to extract meaningful information from the theory. In this chapter, we intro-

duce zero temperature quantum field theory; the addition of the temperature on the system is discussed in Chapter (2).

1.1 Introduction to QCD

In QCD, the strong force is modelled by an $SU(3)$ Yang-Mills interaction, in which fermionic quark fields interact with each other through a vector field carrying colour charge: the gluons. In QCD, the quark fields transform as the fundamental representation of the $SU(3)$ group. There exist several species of quark fields, called quark flavours. The non-abelian nature of $SU(3)$ permits bosonic self-interactions, which are not possible in abelian theories, such as quantum-electrodynamics. Throughout this document, the electroweak interaction between quark fields is assumed to be negligible when compared to the effects of the strong force and the temperature on the system. The local QCD lagrangian density defining the dynamics of QCD is:

$$\mathcal{L}_{\mathrm{QCD}}(x) = \sum_{f=1}^{N_f=6} \bar{\psi}_f(x)\left[\gamma^\mu D_\mu + m_q^f\right]\psi_f(x) - \frac{1}{4}\, F^i_{\mu\nu}(x) F^{\mu\nu,i}(x). \tag{1.1}$$

In the equation above $\psi_f(x)$ represents a fermionic quark field of flavour f; $\bar{\psi}_f(x) = \psi_f(x)^\dagger \gamma_0$ is an anti-quark field of flavour f; γ_μ represents one of the Dirac gamma-matrices; and $F^i_{\mu\nu}(x)$ is the gluonic field strength, defined for each of the 8 gluon species i, one per generator of the $SU(3)$ Lie algebra. The gluonic field strength governs the dynamics of the gluon fields and their self-interaction. The covariant derivative in eq. (1.1), D_μ, is defined as

$$D_\mu = \partial_\mu + i g_0\, T_i\, A^i_\mu(x), \tag{1.2}$$

where ∂_μ is the 4-dimensional partial derivative; g_0 is the coupling parameter; T_i represents one of the 8 generators of the $SU(3)$ Lie algebra; and $A^i_\mu(x)$ is a gluonic vector field with colour index i. The covariant derivative ensures that the lagrangian is gauge invariant. Note that the QCD lagrangian contains several free-parameters: the coupling constant g_0, and the quark masses m_q^f.

The complete QCD lagrangian in eq. (1.1) contains $N_f = 6$ different flavours, which are usually denoted by the names (labels): up (u), down (d), strange (s), charm (c), bottom (b) and top (t). The flavours are ordered by their masses.

At relatively low energies and temperatures, the strong interaction is known to be confining. As a result, a perturbative analysis of the theory is not possible; we are forced to employ non-perturbative tools. In standard conditions, quarks cannot

be detected in isolation; they always form bound states, called hadrons. There are two types of common hadrons: mesons, defined as the bound state of a quark and an antiquark; and baryons, which are generated through the cohesive interaction of three quarks. In this document, we are mainly interested in mesonic hadrons. More information about QCD can be found in Refs. [10–14].

In field theory, the main object defining the dynamics of the system is the action, defined as the integral of the lagrangian density over all Minkowskian space-time:

$$S = \int d^4x \, \mathcal{L}(x). \tag{1.3}$$

The lagrangian is related to the hamiltonian operator through a Legendre transform on the fields. Although the hamiltonian describes the dynamics of a physically measurable quantity, the energy of the system, it is barely employed in field theory as it does not explicitly display the symmetries of the theory.

1.1.1 Quantising a field theory

The action of a field theory defines its classical dynamics. In order to include quantum corrections to the quantities extracted from the theory, it must be quantised. There are several formalisms available to quantise a theory, being the so-called canonical formalism the most common one presented in introductory courses, as it can be interpreted as an *extension* of non-relativistic quantum mechanics.

Before briefly describing some aspects of quantum field theory, we start by reviewing non-relativistic quantum mechanics. In quantum mechanics, the set of states in which a particular system can be, correspond to states in a Hilbert space. By diagonalising the hamiltonian defining the dynamics of the system, we are able to generate a privileged complete set of states for which their energy is known; the measurable energy of the system always corresponds to one of the eigenvalues of the hamiltonian. The superposition principle states that the system can be in any possible linear combination of those states.

In quantum mechanics, for every experimentally measurable quantity, such as the momentum of a particle, its position or its angular momentum, there exists a hermitian operator acting on the Hilbert space of states; those hermitian operators are called observables. The link between experiment and theory is the expectation value of an observable, denoted as

$$\langle\psi|\hat{O}|\psi\rangle.$$

The expectation value corresponds to a scalar product between a *bra* state, $\langle\psi|$, and a *ket* state, produced by the action of the operator $\hat{O}$ over an initial state $|\psi\rangle$.

The expectation value defines the average outcome of experimentally measuring the physical quantity represented by $\hat{O}$.

To merge quantum mechanics with a special relativity covariant field theory using the canonical quantisation formalism, the starting point is to promote the fields defining the lagrangian as operators acting on some space of states. The operators are functionals of the space-time coordinates, which implies that space and time are placed on an equal footing. This differs from the quantum mechanical case, in which the spatial coordinates are treated as operators, while the time is assumed to be a free-parameter of the theory. In the case of non-interacting quantum field theory, the space of states corresponds to a Fock space: a direct sum of Hilbert spaces, each of them representing a fixed number of particles. It is worth stressing that one can try to construct a relativistic single-particle quantum mechanics similar to Schrödinger's equation, e.g. Dirac and Klein-Gordon equation. However, only by quantising a field theory, one is able to naturally deal with the fact that, due to the energy-mass equivalence of special relativity, particles can be created and destroyed.

In this new formalism, the field operators take the role of the quantum mechanical observables, therefore, we can compute their expectation values,

$$\langle \emptyset | \hat{O}(\vec{x}, t) | \emptyset \rangle,$$

where $|\emptyset\rangle$ represents the vacuum state, that is, the state in which no external excitations are present on the system. The action of $\hat{O}(\vec{x}, t)$ over the vacuum creates an excitation at a given space-time coordinate. In the non-interacting limit, the operator $\hat{O}(\vec{x}, t)$ can be decomposed into an infinite sum of creating and annihilating operators, similar to the ones emerging in the quantum mechanical harmonic oscillator.

We can extend the expectation value to several field operators. In quantum field theory, as the operators are functionals of the space and time, the expectation value of several operators is referred to by the name correlation function, formally written

$$\langle \emptyset | \hat{O}_n(\vec{x}_n, t_n) \cdots \hat{O}_1(\vec{x}_1, t_1) | \emptyset \rangle.$$

In the equation above, the operators defining the correlation function are assumed to be time-ordered: $t_1 \leq t_2 \leq \cdots \leq t_n$. In the particular case in which the equation above only contains two operators, then the expectation value is called a 2-point correlation function. Only 2-point correlation functions are discussed in this document.

Correlation functions are essential objects in the description of complex systems, such as quantum field theories and statistical mechanics. They are key to understanding phase transitions, spectroscopy, transport phenomena and scattering amp-

litudes. An example showing their importance is the so-called Lehmann-Symanzik-Zimmermann (LSZ) formula [14–18], which directly relates theoretically computable correlation functions to experimentally measurable scattering amplitudes.

Another formalism employed in quantum field theory is the so-called path integral formulation [19–21]. In this formulation, the field operators are treated as integration variables, and the correlation functions of the theory are defined as integrals over the complete space of configurations of the system. The path integral of a general action is defined as

$$Z = \int D\psi D\bar{\psi} DA_\mu \exp\left(i\, S[\psi, \bar{\psi}, A_\mu]\right). \tag{1.4}$$

We do not discuss the technical aspects of the path-integral formalism, such as the convergence of eq. (1.4), or the definition of the measures appearing in the integral.

The path integral can be viewed as a weighted (complex) average over all possible configuration paths in which the system can be; the weight of each path is determined through the complex phase in eq. (1.4). Provided that we analytically continue from real time t to imaginary-time, $\tau = -i\,t$, then, the argument in the exponential becomes real and negative. The imaginary-time transformation is usually called a Wick rotation, and it implies that the Minkowskian space-time of special-relativity is replaced with an Euclidean one.

In the path-integral formalism, Euclidean correlation functions of n operators are computed as follows:

$$\begin{aligned}&\langle\emptyset|\hat{O}_n(\vec{x}_n, t_n)\, \cdots\, \hat{O}_1(\vec{x}_1, t_1)|\emptyset\rangle = \\ &\frac{1}{Z}\int D\psi D\bar{\psi} DA_\mu O_n(\vec{x}_n, t_n)\, \cdots\, O_1(\vec{x}_1, t_1) \exp\left(-S[\psi, \bar{\psi}, A_\mu]\right).\end{aligned} \tag{1.5}$$

1.2 Short introduction to lattice field theory

As stated before, non-perturbative tools are required to extract meaningful information about QCD. At the time in which this document is being written, regularising the 4-dimensional Minkowskian space-time on a Euclidean lattice is the only known method that allows a systematic and reliable first-principles analysis of QCD at low energies. Replacing the continuous space-time of a field theory with a discrete mesh imposes a natural cut-off in the energy through the inverse of the lattice spacing, a; the lattice spacing corresponds to the minimum distance between two neighbouring points in the lattice. The regularisation of the space-time on a lattice combined with the path-integral formalism of quantum field theory allows the estimation of QCD observables at a given lattice spacing.

The first step towards extracting quantum observables on the lattice is the regularisation of the Euclidean space-time. This involves replacing the continuous space-time with a 4-dimensional finite lattice. The lattice contains a volume of $N_s^3\, N_\tau$ points, where N_s represents the size (number of points) of the lattice in each of the three spatial directions, and N_τ represents the size of the lattice in the temporal direction. In principle, N_s could be different for each spatial direction. The lattice, labelled Λ, is defined as a set of 4-dimensional sites n,

$$\Lambda = \{n = (n_x, n_y, n_z, n_\tau) \,|\, 0 < n_x, n_y, n_z < N_s \,;\, 0 < n_\tau < N_\tau\}. \tag{1.6}$$

All lattice coordinates are expressed in terms of the lattice spacing, for example: $\tau = n_\tau\, a$ and $x = n_x\, a$. The length of the lattice in each direction is: $L_s = N_s\, a$, $L_\tau = N_\tau\, a$.

As in any finite system, boundary conditions need to be defined. Fermionic fields respect anti-periodic boundary conditions, while bosonic field fulfil periodic boundary conditions.

As the lattice is discrete and finite, its Fourier representation is also discrete and finite. The set of all available frequencies is called the Brillouin zone, and they depend on the boundary conditions. In general, the allowed frequencies are $k_\mu = (2\pi/L_\mu)\, n$, with n an even (odd) natural number for periodic (anti-periodic) fields.

The lattice spacing is a parameter that needs to be fixed. Setting the scale can be done by fixing an observable measurable on the lattice to its corresponding experimental value. Once the lattice spacing is fixed, all other measurable quantities extracted from the lattice become predictions. For more information about how the scale is set in lattice QCD simulations, we refer to Ref. [22]. All quantities measured on the lattice are computed in terms of the lattice spacing, which implies that they are expressed in lattice units. Setting the scale requires the analysis of some lattice measurements, as a result, the lattice spacing is affected by statistical uncertainty.

Extracting observables that do not depend on the lattice spacing requires a continuum limit extrapolation of the lattice observables. Initially, one could think that any action defined on the lattice, and whose functional form recovers a particular desired continuum action in the limit $a \to 0$, is a sufficient condition to estimate quantities in the reference continuum action as $a \to 0$. However, a proper scaling law with the lattice spacing is also required. Consequently, for a lattice action to have the desired continuum limit, two conditions must be fulfilled: first, it has to recover the functional form of the reference continuous action in the limit of $a \to 0$; second, the observables measured on the lattice must scale with the lattice spacing accordingly, that is, they must approach a fixed point as the lattice spacing decreases.

There exist an infinite number of lattice actions whose limiting functional forms

are equivalent. To see this, imagine including second-order operators to a lattice lagrangian:

$$\mathcal{L}_{\text{lat}} = \mathcal{L} + a\,\mathcal{L}' + \dots \tag{1.7}$$

Under some general conditions, those higher-order artefacts vanish in the limit of $a \to 0$ faster than any lattice linear operator, thus recovering the same functional form. All actions whose continuum limit functional is the same to a target continuum action are said to naively recover a continuous action.

In order to ensure that a lattice action corresponds to a given continuum action in the limit of $a \to 0$, we use the fact that experimentally measurable quantities are finite. Now, imagine that there exist an observable θ measurable both experimentally and on the lattice; this observable has mass dimensions $d\theta$. We refer to the experimental value of θ as θ_{phys}, while its lattice estimate is labelled $\hat{\theta}$. The relationship between θ and $\hat{\theta}$ is

$$\theta = a^{-d\theta}\,\hat{\theta}, \tag{1.8}$$

where a is the lattice spacing. In principle, lattice actions contain free parameters, such as the bare quark masses m_q^f, or the coupling constant g_0. Consequently, the lattice measurement $\hat{\theta}$ can also depend on the values of the parameters:

$$\theta = a^{-d\theta}\,\hat{\theta}(g_0, m_q^f, \dots). \tag{1.9}$$

In the continuum limit, the left-hand side of eq. (1.9) remains finite, and its value is θ_{phys}. However, the right-hand side vanishes as $a \to 0$. As we do want the equality to hold in the continuum limit, there must be a set of values of the parameters $(g_0^\star, m_q^{f,\star})$ that keeps the lattice measurement finite as the lattice spacing reaches its continuum limit value. This corresponds to a phase transition in which the system loses its lattice nature to reach its continuum form. The values $g_0^\star, m_q^\star$ are known as the fixed-points of the renormalisation group equations.

In practice, in order to reach the continuum limit, one measures the same observable at different lattice spacings close to the critical point, and then performs a numerical extrapolation. As the lattice spacing shrinks, the volume spanned by the lattice also tends to zero. Consequently, in order to reach the continuum limit, the volume of the lattice must grow as the lattice spacing shrinks. This is called the thermodynamic limit of the theory, and reaching it is computationally expensive. More information about the lattice continuum limit can be found in Refs. [23–25].

All results presented in this document are measured at finite lattice spacing; no continuum limit is taken.

1.2.1 Building lattice actions

In order to demonstrate how to regularise the continuum QCD action on the lattice, the standard starting point is the limit of non-interacting fermions, which can be achieved by setting $g_0 = 0$ or $A_\mu = \mathbb{1}$ in the lagrangian density defined in eq. (1.1). In the non-interacting limit, the continuum action is

$$S_{\text{free}} = \int d^4x\, \bar{\psi}(x)\big[\gamma^\mu \partial_\mu + \mathbb{1} m_q\big]\psi(x). \tag{1.10}$$

As we work in discrete space-time, the integrals become sums over lattice sites. In addition, we need to replace the partial derivative operator with the corresponding finite difference operator, whose definition is not unique. We employ the following definition:

$$\partial_\mu f(x) \to \frac{f(n+\mu) - f(n-\mu)}{2a}. \tag{1.11}$$

The naive version of the free lattice fermionic action corresponds to

$$S^{\text{lat}}_{\text{free}} = a^4 \sum_n \bar{\psi}(n)\left[\sum_{\mu=0}^{4} \gamma_\mu \frac{\psi(n+\mu) - \psi(n-\mu)}{2a} + \mathbb{1} m_q\, \psi(n)\right]. \tag{1.12}$$

By analysing this lattice action, we can find a fundamental problem of lattice field theories containing fermionic matter. To encounter this problem, we start by defining the Dirac operator as the operator that acts on the quark bilinear in eq. (1.12):

$$\bar{\psi}\, D\psi.$$

The Dirac operator of eq. (1.12) is just

$$D(n,m) = \sum_{\mu=0}^{4} \gamma_\mu \frac{\delta_{n+\mu,m} - \delta_{n-\mu,m}}{2a} + m\, \delta_{n,m}. \tag{1.13}$$

This operator can be Fourier transformed on the lattice and then inverted, which generates the propagator of a non-interacting fermionic field:

$$S(k) = D^{-1}(k) = \frac{m_q \mathbb{1} - i a^{-1} \sum_\mu \gamma_\mu \sin(k_\mu a)}{m_q^2 + a^{-2} \sum_\mu \sin^2(k_\mu a)}. \tag{1.14}$$

The poles of the propagator, corresponding to the zeroes in the denominator, are interpreted as propagating particles. In general, we expect only one pole in the

quark propagator, corresponding to a quark of mass m_q. However, the propagator in eq. (1.14) describes 16 different particles, as there are 16 poles in the propagator: one located at $k = (0, 0, 0)$, and the other ones located at the corners of the lattice in Fourier space. These 15 clones are called doublers, and they share the same mass m_q.

Doublers are unwanted and should be removed from the propagator, as they lead to wrong physics [25].

A common way of removing the doublers in eq. (1.12) is through the inclusion of the so-called Wilson term [25] in the action. The Wilson term corresponds to a discretisation of the following operator

$$W_T = -\frac{r}{2}\partial_\mu\partial_\mu \tag{1.15}$$

where ∂_μ is the derivative operator and r is the Wilson r-parameter, usually set to $r = 1$. Adding the Wilson term to the action in eq. (1.12) implies that the Dirac operator in momentum space is

$$D(k) = m\mathbb{1} + \frac{i}{a}\sum_\mu \gamma_\mu \sin(k_\mu a) + \mathbb{1}\,\frac{r}{a}\sum_\mu [1 - \cos(k_\mu a)]. \tag{1.16}$$

The inverse of this operator is similar to eq. (1.14). However, it contains a dynamic mass contribution in the denominator. This dynamic mass term is zero at $k = (0, 0, 0)$ but non-zero in all other corners of the Brillouin zone. Therefore, the Wilson term eliminates all doublers at the cost of explicitly breaking chiral symmetry. There is an important theorem, called the Nielsen-Niyomiya no-go theorem [26, 27], which states that one cannot create a local lattice action that does not explicitly break chiral symmetry and, at the same time, does not include doublers.

Other fermionic lattice formulations allow the exclusion of doublers from the propagator: an example is the staggered fermion formulation [28]. As stated before, there are infinitely many lattice actions that share the same naive continuum action. Consequently, the lattice community has worked for years trying to find different lattice formulations with desirable properties. Some examples of these lattice formulations include the Symanzik's improvement scheme, in which higher order operators, such as the so-called clover term, are added to the action to cancel second order discretisation effects on the lattice observables [29–31]; the domain-wall fermionic formulation, in which a new space-time dimension is added to the fermions to avoid explicitly breaking chiral symmetry [32, 33]; or the twisted mass formulation, in which a complex term is added to the quark masses to remove second order discretisation effects on the masses of the states [34, 35].

In order to build lattice actions whose continuum limit naively corresponds to the interacting QCD action in eq. (1.1), we need to impose two postulates: first, lattice fermionic quark fields belong to the fundamental representation of the $SU(3)$ colour group; and second, the action must be invariant under local $SU(3)$ rotations on the fields, that is, it must be $SU(3)$ gauge invariant. To fulfil both postulates, the lattice quark fields must transform under local $SU(3)$ rotations in the following way:

$$\psi(n) \to \Omega(n)\psi(n); \quad \bar{\psi}(n) \to \bar{\psi}(n)\Omega^{\dagger}(n), \tag{1.17}$$

where $\Omega(n)$ is an element of $SU(3)$ and $\Omega^{\dagger}$ represents its hermitian conjugate. The mass term in eq. (1.12) is invariant under the transformations shown above. However, the kinetic term is not. To enforce complete gauge invariance, a new field $U_{\mu}(n)$ is introduced, that as

$$U_{\mu}(n) \to \Omega(n)U_{\mu}(n)\Omega^{\dagger}(n+\mu). \tag{1.18}$$

The gauge fields U_{μ} are called link variables, and are elements of $SU(3)$. They connect two lattice sites under a gauge transformation, as defined in eq. (1.18). The complete QCD lattice action is not derived in this document, but can be found in Refs. [23, 25]. The simplest lattice action that naively recovers the complete QCD lagrangian shown in eq. (1.1) is

$$S_{QCD}^{\text{lat}} = S_F[\psi, \bar{\psi}, U_{\mu}] + S_G[U_{\mu}], \tag{1.19}$$

where

$$S_F = a^4 \sum_{n\in\Lambda} \bar{\psi}(n) D[U_{\mu}](n)\psi(n), \tag{1.20}$$

being

$$D[U_{\mu}](n) = \sum_{\mu=0}^{4} \gamma_{\mu} \frac{U_{\mu}(n)\delta_{n,n+\mu} - U_{\mu}^{\dagger}(n-\mu)\delta_{n,n-\mu}}{2a} + m_q, \tag{1.21}$$

and

$$S_G = \frac{2}{g^2} \sum_{n\in\Lambda} \sum_{\mu<\nu} \operatorname{Re}\operatorname{Tr}\big[\mathbb{1} - U_{\mu\nu}(n)\big], \tag{1.22}$$

with $U_{\mu\nu}(n)$ being the plaquette operator, defined as

$$U_{\mu\nu}(n) = U_{\mu}(n)U_{\nu}(n+\mu)U_{\mu}^{\dagger}(n+\nu)U_{\nu}^{\dagger}(n). \tag{1.23}$$

Figure (1.1) shows how link variables and fermionic variables interact on the lattice.

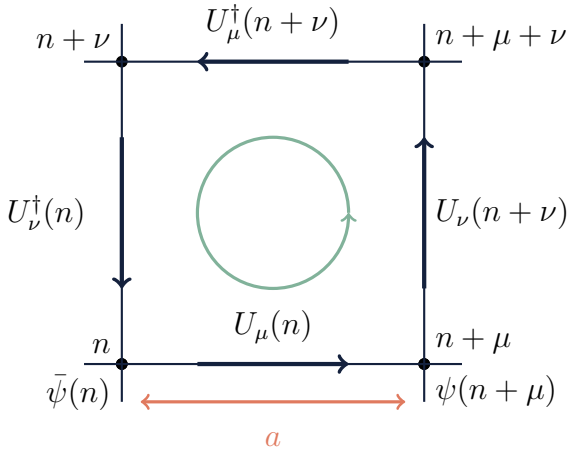


Figure 1.1: Diagram depicting a lattice cell and the relevant fermionic fields and gauge links. The fermionic fields, $\bar{\psi}(n)$ and $\psi(n+\mu)$, occupy the lattice sites n and $n+\mu$ respectively. These fields interact through the gauge link $U_\mu(n)$ pointing from n to $n+\mu$. The green arrows show the plaquette operator, defined in eq. (1.23).

1.2.2 Quantising a lattice field theory

Equation (1.19) corresponds to the standard implementation of the QCD action on the lattice. It contains three fundamental fields: the quark and antiquark fields, respectively represented by $\psi(n)$ and $\bar{\psi}(n)$, and the gluonic link variables, $U_\mu(n)$.

Regularising an action on the lattice is the first step towards its complete analysis; the next step is to quantise it. As in the continuous case, the Euclidean path integral on the lattice treats the lattice fields as integration variables. The importance of each field configuration is determined by its action. An example of a lattice Euclidean path-integral is

$$Z = \int D\psi(n) D\bar{\psi}(n) DU_\mu(n) \exp\big(-S[\psi(n), \bar{\psi}(n), U_\mu(n)]\big). \tag{1.24}$$

We can compute any correlation function of Euclidean lattice operators as

$$\begin{aligned}&\langle \hat{O}_m(n_m) \ldots \hat{O}_2(n_2)\, \hat{O}_1(n_1)\rangle = \\ &\frac{1}{Z}\int D\psi(n) D\bar{\psi}(n) DU_\mu(n) O_m(n_m) \ldots O_1(n_1) \exp\big(-S[\psi, \bar{\psi}, U_\mu]\big).\end{aligned} \tag{1.25}$$

Lattice correlation functions resemble continuous correlation functions, defined in eq. (1.5), after the substitution the space-time coordinate x with Euclidean lattice

site n, and the gluon field A_μ with the gauge links U_μ. Additionally, one must take care of the definition of the gauge link integration measure, DU_μ, as the integration must be performed over the $SU(3)$ group [25]. Furthermore, to ensure fermionic statistics, fermionic fields are treated as Grassmann variables [25].

An important result derived from the properties of Grassmann numbers is the atthews-Salam formula [25], which reads

$$Z = \int d\psi d\bar{\psi} \exp\big(\bar{\psi} D[U_\mu]\psi\big) = \det\big[D[U_\mu]\big]. \tag{1.26}$$

This relationship implies that the integral of any bilinear operator, such as the Dirac operator, of a combination of Grassmann variables is the determinant of the operator. Applying the Matthews-Salam relationship to eq. (1.24) allows us to obtain

$$Z = \int DU_\mu \det\big[D[U_\mu]\big] \exp\big(-S[U_\mu]\big). \tag{1.27}$$

In the case in which several flavours are included in the action, each flavour contributes with a factor of the determinant to the path integral.

To compute eq. (1.25), the following *probability distribution function* is defined:

$$\rho(U_\mu(n)) = \frac{1}{Z} \det\big[D[U_\mu(n)]\big] \exp\big(-S[U_\mu(n)]\big). \tag{1.28}$$

Employing the definition above, one computes eq. (1.25) as

$$\langle \hat{O}_m(n_m) \ldots \hat{O}_2(n_2)\, \hat{O}_1(n_1) \rangle = \int DU_\mu(n) O_m(n_m) \ldots O_1(n_1) \rho(U_\mu(n)). \tag{1.29}$$

1.2.3 Simulating lattice field theories

We know how to regularise an action on the lattice, and also the basic recipes to quantise it. As a result, we are in the position to start discussing how we could simulate a lattice quantum field theory on a computer. The main goal is to integrate eq. (1.25). A correlation function is the expectation value of a function depending on a set of random variables, in our case, the fields living in a 4-dimensional space-time.

We assume that all integrable fields in eq. (1.25) can be viewed as random variables living on a 4-dimensional lattice, whose probability distribution function is eq. (1.28). Provided that we sampled enough realisations of these random variables, then, we could use those artificially generated samples to compute the sample average of some observables. From the central limit theorem, we can be sure that the

sample average converges in probability to the expected value of a random variable as the number of samples increases. The question is: how do we sample the fields?

Naively, we could try generating random $SU(3)$ elements, one per each point in the lattice. Those randomly generated configurations can be employed to compute the fermion determinant at each configuration. After having collected numerous ensembles, we can estimate eq. (1.25) as long as we know how to evaluate correlation functions on a collection of ensembles. Once all estimates are generated, we can compute their sample average. This estimate will converge in probability to the population expected value as long the number of samples is large enough, and the samples are independent and identically distributed. The variance of this estimator decreases with the number of configurations available, as it corresponds to the variance of the sample mean.

There are two problems with this algorithm: it is completely inefficient, and it is not ensured to produce reliable estimates of the studied observables. Generally, in most probability distributions, there are some values of the random variables that will weight more than others. Consequently, we should try sampling those important configurations more frequently. To do so, we need to generate configurations of the fields that are correctly distributed. In statistics, this is called Monte Carlo sampling, a standard procedure used to generate samples with known distribution. This technique is widely used to simulate systems in material sciences, condensed matter or chemistry [36–38].

The basic goal behind Monte Carlo sampling in the context of lattice QCD is to generate a set of random variables Y distributed according to $P(Y)$ starting from a randomly selected initial configuration Y_0. In our case, the random variables are the gauge links, whose distribution is eq. (1.28). The ensemble generation is performed iteratively by updating the configurations treated as quasi-independent random variables. The dynamics of the evolution are modelled using a Markov chain. A Markov chain is a sequence of random variables $Y_0 \to Y_1 \to \cdots \to Y_n$ satisfying

$$P(Y_0, Y_1, \dots) = P(Y_0)P(Y_1|Y_0)P(Y_2|Y_1)\dots P(Y_n|Y_{n-1}), \tag{1.30}$$

where $P(Y)$ represents the probability distribution of the variable Y; $P(Y, X)$ represents the joint distribution; and $P(Y|X)$ is the conditional distribution of Y on X. The equation above is satisfied for all values of the random variables.

In our setup Y_t and Y_{t+1} represent two sequential configuration updates; an update is also called a Markovian step or Monte Carlo time step. Equation (1.30) simply implies that the chain has short-term memory: the value of a random variable Y_t only depends on the previous random variable in the chain Y_{t-1}; the conditional probability is sometimes called the transition probability. As we want all configurations to

be accessible at every Monte Carlo update, the transition probability must satisfy

$$\sum_{y\in\mathcal{Y}} P(Y_t = y|Y_{t-1} = x) = 1, \tag{1.31}$$

where $\mathcal{Y}$ is the set of all possible realisations of Y_t: in our particular case, the set of all possible configurations of the gauge links.

In addition, we would like our chain to eventually reach the equilibrium probability distribution: $P(Y = y)$, where Y is independent of all previous configurations. This can be ensured through the so-called detailed balance equation:

$$P(Y_t = y|Y_{t-1} = x)P(Y_{t-1} = x) = P(Y_{t-1} = x|Y_t = y)P(Y_t = y). \tag{1.32}$$

The detailed balance equation implies that jumping from configuration $Y_{t-1} = x$ to $Y_t = y$ is as likely as jumping from configuration $Y_t = y$ to $Y_{t-1} = x$. Summing over all possible values of Y_{t-1} in eq. (1.32) and making use of eq. (1.31) allows us to prove that we will eventually reach the equilibrium distribution $P(Y_t = y)$:

$$P(Y_t = y) = \sum_{x\in\mathcal{X}} P(Y_t = y|Y_{t-1} = x)P(Y_{t-1} = x). \tag{1.33}$$

Monte Carlo sampling suffers from two important limitations: thermalisation and correlation. Thermalisation means that we should allow the system to evolve towards its equilibrium distribution. As we start from a random initial configuration, the first sampled configurations are usually far from being distributed according to the desired equilibrium distribution. As a result, one should always let the Markov chain evolve for several iterations before saving the configurations. One can track the status of the Markov chain by measuring some quantities on the generated configurations. Once the chain is thermalised, we can start saving the configurations to compute any desired quantity with them.

However, note that, as the configurations are generated in a Markov chain, they are sequentially highly correlated: previous configurations are employed to generate new configurations — see eq. (1.30). The fact that the configurations are correlated has severe consequences, as correlated random variables are not independent and, therefore, they do not comply with the central limit theorem, thus generating biased expected value estimates. Correlation among Monte Carlo configurations can be reduced by either waiting some Markovian iterations between sequential measurements, or by binning contiguous configurations into a single estimate. In this context, binning is equivalent to computing the sample average. Information about the ensembles employed in the analysis presented in this document can be found in Ref. ([39]) and Appendix A.

An algorithm that employs uniformly distributed pseudo-random numbers in order to generate ensembles of an arbitrary random variable Y distributed according to $P(Y = y)$ is the Metropolis algorithm [40]. A simple Metropolis algorithm is sufficient to simulate a quenched approximation of QCD on the lattice. In the quenched approximation, the fermion determinants in eq. (1.28) are set to 1.

Including the fermion determinant in the simulations is expensive and difficult, as it involves the computation of the determinant of a large matrix that directly depends on the gauge fields. As a result, the determinant must be evaluated at each Monte Carlo step. Additionally, each distinguishable quark field contributes with a different determinant to eq. (1.28). In lattice simulations, it is common to include the determinant for some selected flavours, and leave some other flavours quenched, usually heavier quarks.

A standard way of including dynamical fermions in the simulations is through the concept of pseudo-fermions [41]. Including pseudo-fermions evades the computation of the fermion determinant but implies that the newly introduced pseudo-fermion action is non-local. As a result, the Monte Carlo transition probability depends globally on both the previous and the newly generated configurations, which makes its estimation computationally expensive. A standard algorithm applied to generate lattice QCD with dynamic quarks is the Hybrid Monte Carlo (HMC) algorithm [42, 43].

Nowadays, lattice QCD simulations treat the three lightest quark flavours (u, d, s) as dynamical quarks; this is sometimes called a 2+1 simulation, as the u and d quarks are assumed degenerate. Lattice simulations including dynamical charm quarks are becoming more and more common. The heaviest quarks, top and bottom, are normally inaccessible due to their large masses.

Computing correlation functions from lattice configurations

The result of our Monte Carlo sampling algorithm is a collection of N_b nearly independent configurations. Those configurations could have been generated in the quenched approximation, or taking into account several dynamic flavours. From the ensemble of configurations, any desired quantity can be computed on the lattice.

From now on, we assume that the N_b configurations are generated from a thermalised Markov chain. Additionally, the correlation between configurations is supposed to be minimal. Therefore, we can treat the configurations as independent and identically distributed random variables.

We would like to evaluate eq. (1.25) on the sample of configurations; this can be done independently for each configuration. Each measurement leads to an in-

dependent estimate of a particular correlation function, and the combination of all measured correlation functions into a single sample allows the computation of the ensemble average, which corresponds to an unbiased estimate of the population correlation function. Due to the fact that all analysis presented in this document deal with mesonic correlation functions, we restrict this discussion to correlation functions constructed from mesonic operators.

A mesonic operator is composed by a quark-antiquark pair located at a given space-time coordinate x. In principle, both quark fields could have different flavours. Additionally, in order to generate mesonic operators with different quantum numbers, we project the quark fields using different operators, for example, using the Dirac γ-matrices. In principle, other operators can be included depending on the implementation, such as derivatives or operators belonging to other spaces, e.g. flavour space. A general mesonic operator composed of a quark field ψ and an antiquark field $\bar{\psi}'$ is defined as

$$M(x) = \bar{\psi}'_{a,\alpha}(x)\ \Gamma_{ab,\alpha\beta}\ \psi_{b,\beta}(x), \tag{1.34}$$

where a and b are colour indices, and α and β are spinor indices. Note that ψ and $\bar{\psi}'$ might represent different quark flavours. The colour space part of Γ is δ_{ab} to ensure colourless bounded states.

The conjugate of $M(x)$ is denoted by $M^\dagger(x)$, and it is defined as

$$M^\dagger(x) = \psi^\dagger(x)\Gamma^\dagger\bar{\psi}'^\dagger(x) = \bar{\psi}(x)\gamma_0\Gamma^\dagger\gamma_0\psi'(x) = \bar{\psi}(x)\bar{\Gamma}\psi'(x), \tag{1.35}$$

where $\bar{\Gamma} = \gamma_0\Gamma\gamma_0$. In the derivation displayed above, we have employed

$$\bar{\psi} = \psi^\dagger\gamma_0, \tag{1.36}$$

The action of both mesonic operators on the system can be regarded as respectively creating an excitation of the same quantum numbers of M and $M^\dagger$. One can obtain the quantum numbers of the operator by studying how it transforms under different symmetries, such as charge conjugation or parity. Some information about different operators used in our simulations can be found in Appendix (B).

A mesonic 2-point correlation function is defined as the action of two mesonic operators at different points of the space-time. To simplify the derivation, we restrict ourselves to the continuous case. However, the final expression is always evaluated on the lattice by replacing all the integrals with sums, all the coordinates with lattice coordinates, and any operator with its lattice analogous. A mesonic 2-point correlation function is defined as

$$C(y,x) = \langle\emptyset|M(y)M^\dagger(x)|\emptyset\rangle. \tag{1.37}$$

Due to translation invariance of systems in equilibrium, 2-point correlation function depend on the distance: $C(y,x) = C(y-x)$. The first operator, $M^\dagger(x)$, is called the source operator, and it is generally placed at the origin of coordinates: $x = 0$. The second operator, $M(y)$, is called the sink operator. All physical states with the same quantum numbers contribute to the correlation function.

We can replace the mesonic operators with their definitions in terms of quark fields using both eq. (1.34) and eq. (1.35):

$$C(y-x) = \pm\langle\bar{\psi}(y)\Gamma_A\psi'(y)\,\bar{\psi}'(x)\bar{\Gamma}_B\psi(x)\rangle. \tag{1.38}$$

where Γ_A can be, in principle, different to Γ_B. We can use Wick's theorem [25] on the fermionic fields to contract them and obtain the following expression:

$$C(y-x) = -\operatorname{Tr}\left[\Gamma_A S_\psi(y-x)\bar{\Gamma}_B S_{\psi'}(x-y)\right] \tag{1.39}$$

where $S_\psi(y-x)$ is the quark propagator of flavour ψ, computed through the inverse of the Dirac operator. A derivation of eq. (1.39) can be found in Ref. [25]. This expression is only valid for non-singlet mesonic operators. In the case in which singlet operators are present, eq. (1.39) also contains disconnected contributions, which are terms dependent on only one propagator. Disconnected contributions are mandatory when simulating charge-neutral mesons, such as the neutral pion. They are difficult to compute as they are inherently noisy. No disconnected contributions are computed in our simulations.

Without loss of generality, we can place the source operator at the origin of coordinates, $x = 0$, leading to

$$C(y) = -\operatorname{Tr}\left[\Gamma_A\ S_\psi(y)\ \bar{\Gamma}_B\ S_{\psi'}(-y)\right]. \tag{1.40}$$

It is common to work in a time-momentum representation of $C(y)$, which implies that the correlation function depends on the Euclidean time $\tau_y = \tau$ and the external momenta of the system $\vec{k}$: $C(y) \to C(\tau,\vec{k})$. This representation can be achieved through:

$$S(\tau,\vec{k}) = \int d^3y\, e^{i\vec{k}\vec{y}} S(\tau,\vec{y}). \tag{1.41}$$

Using the Fourier transform of eq. (1.40), introducing the time-momentum representation of both propagators, and manipulating the integral, allows to obtain the time-momentum representation of the correlation function:

$$C(\tau,\vec{k}) = -\operatorname{Tr}\int d^3k\left[\Gamma_A S_\Phi(\tau,\vec{k})\Gamma_B S_\psi(-\tau,\vec{k})\right]. \tag{1.42}$$

In order to evaluate mesonic correlations on a given configuration U_μ, we compute two quark propagators, defined as the inverse of the Dirac operator in eq. (1.21). Computing the quark propagator involves the inversion of a huge and sparse matrix [25]. Consequently, one does not compute the whole quark propagator, as it is expensive both computationally and memory-wise. Instead, we can take advantage of the fact that the source operator is placed at a fixed space-time coordinate x in order to only compute the propagator from that source point to all possible sink locations on the lattice, represented by y; the sliced propagator represents a column of the Dirac operator. Note that, for each space-time coordinate x, there are 12 different values of the propagator, corresponding to the combinations of 3 colours and 4 spinor indices. One can select a column of the propagator employing a source vector G through

$$S_{i_0,j}^{\alpha_0,\beta}(y-x_0) = \sum_{x,\alpha,i} S_{i,j}^{\alpha,\beta}(y-x) G_{i_0}^{\alpha_0}(x-x_0). \tag{1.43}$$

In the case in which

$$G_{i_0}^{\alpha_0}(x-x) = \delta(x-x_0)\delta_{i,i_0}\delta_{\alpha,\alpha_0}, \tag{1.44}$$

then only one column of S is used in the computation of the propagator. These source vectors are called local sources. Non-local sources can be employed by defining different source vectors, for example Gaussian source vectors. Non-local sources can enhance the signal of the propagator as they include more information coming from the Dirac operator. Non-local sources are called smeared sources [25, 44, 45].

Local sources tend to uniform the influence of all states to the total correlation function, which sometimes hinders the analysis of the estimated correlation functions. Instead, smeared sources tend to improve the influence of asymptotic and stable states, which usually correspond to lower energy states. Including smeared sources is generally encouraged as it corresponds to adding more information to the approximated correlation functions. In our simulations, we generate correlation functions using both local and Wuppertal-smeared sources [46], combined with APE smearing of the link variables [47]. More information about the parameters defining our smeared sources can be found in Appendix (A).

The standard algorithm employed to compute propagators is the so-called conjugate gradient method [48].

Spectral decomposition of correlation functions at zero temperature

To conclude, we present an important result of quantum field theory, the so-called spectral decomposition of a correlation function. Through this decomposition, one

can study the spectroscopy of the theory from estimated correlation functions. The spectral decomposition of a correlation function reads

$$\langle \hat{O}_2(\tau, \vec{k} = \vec{0}) \hat{O}_1(0, \vec{0}) \rangle = \sum_{n=0}^{\infty} \langle \emptyset | \hat{O}_2 | n \rangle \langle n | \hat{O}_1 | \emptyset \rangle e^{-E_n \tau}. \tag{1.45}$$

The states contributing to the spectral decomposition are assumed to be hierarchically ordered in energy: $E_s > E_{s'}$, $\forall\, s > s'$.

Equation (1.45) can be derived by working in the Heisenberg picture of quantum mechanics and inserting a complete set of states that diagonalise each field operator. In Euclidean time, the spectral decomposition can be interpreted as a model for the contribution of each state to the total correlation function: the contribution of higher-order states will be exponentially suppressed as the temporal distance between both mesonic operators increases. This interpretation is only valid in the imaginary-time formalism of quantum mechanics; in the real-time formalism, the exponential weights in the spectral decomposition become complex. Figure (1.2) shows some real correlation functions extracted from lattice simulations at different temperatures. The complete derivation of the spectral decomposition can be found in Ref. [25].

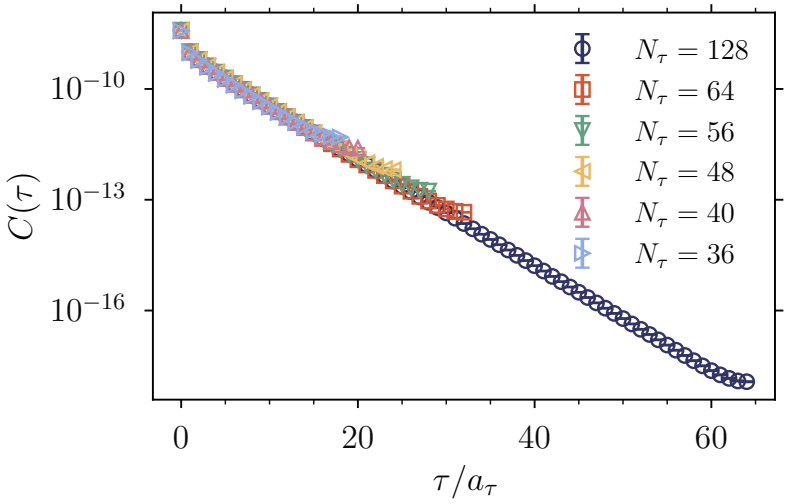


Figure 1.2: Example of simulated mesonic correlation functions in our lattice setup, described in detail in Appendix (A). The correlation functions correspond to the *sc* meson (strange-charm) in the pseudoscalar channel. Information about the physical states simulated can be found in Appendix (B).

From eq. (1.45), we learn that the correlation function should be dominated by

the lowest energy state at large temporal separations between the sink and the source operators, as long as $E_0 \gg E_{s>0}$. In contrast, at $\tau \to 0$, the contribution of all states is non-negligible. In this region, the correlation function is said to be contaminated by excited states.

Chapter 1. References

[1] H. Fritzsch, Murray Gell-Mann and H. Leutwyler. "Advantages of the Color Octet Gluon Picture". In: *Phys. Lett. B* 47 (1973), pp. 365–368. DOI: 10.1016/0370-2693(73)90625-4.

[2] P. A. Zyla et al. "Review of Particle Physics". In: *PTEP* 2020.8 (2020). DOI: 10.1093/ptep/ptaa104.

[3] Kenneth G. Wilson. "Confinement of Quarks". In: *Phys. Rev. D* 10 (1974). Ed. by J. C. Taylor, pp. 2445–2459. DOI: 10.1103/PhysRevD.10.2445.

[4] Tapan K. Nayak. "Probing the QCD phase structure using event-by-event fluctuations". In: *J. Phys. Conf. Ser.* 1602.1 (2020). Ed. by Rene Bellwied et al., p. 012003. DOI: 10.1088/1742-6596/1602/1/012003. arXiv: 2008.04643 [nucl-ex].

[5] Adam Bzdak et al. "Mapping the Phases of Quantum Chromodynamics with Beam Energy Scan". In: *Phys. Rept.* 853 (2020), pp. 1–87. DOI: 10.1016/j.physrep.2020.01.005. arXiv: 1906.00936 [nucl-th].

[6] M. M. Aggarwal et al. "An Experimental Exploration of the QCD Phase Diagram: The Search for the Critical Point and the Onset of De-confinement". In: (July 2010). arXiv: 1007.2613 [nucl-ex].

[7] Y. Aoki et al. "The Equation of state in lattice QCD: With physical quark masses towards the continuum limit". In: *JHEP* 01 (2006), p. 089. DOI: 10.1088/1126-6708/2006/01/089. arXiv: hep-lat/0510084.

[8] Y. Aoki et al. "The Order of the quantum chromodynamics transition predicted by the standard model of particle physics". In: *Nature* 443 (2006), pp. 675–678. DOI: 10.1038/nature05120. arXiv: hep-lat/0611014.

[9] Y. Aoki et al. "The QCD transition temperature: results with physical masses in the continuum limit II." In: *JHEP* 06 (2009), p. 088. DOI: 10.1088/1126-6708/2009/06/088. arXiv: 0903.4155 [hep-lat].

[10] William Marciano and Heinz Pagels. "Quantum chromodynamics". In: *Physics Reports* 36.3 (1978), pp. 137–276.

[11] Walter Greiner, Stefan Schramm and Eckart Stein. *Quantum chromodynamics*. Springer Science & Business Media, 2007.

[12] Frank Wilczek. "QCD made simple". In: *Phys. Today* 53N8 (2000), pp. 22–28. DOI: 10.1063/1.1310117.

[13] Matthew D Schwartz. *Quantum field theory and the standard model*. Cambridge University Press, 2014.

[14] Michael E Peskin. *An introduction to quantum field theory*. CRC press, 2018.

[15] H. Lehmann, K. Symanzik and W. Zimmermann. "On the formulation of quantized field theories". In: *Nuovo Cim.* 1 (1955), pp. 205–225. DOI: 10.1007/BF02731765.

[16] H. Lehmann, K. Symanzik and W. Zimmermann. "On the formulation of quantized field theories. II". In: *Nuovo Cim.* 6 (1957), pp. 319–333. DOI: 10.1007/BF02832508.

[17] M. Srednicki. *Quantum field theory*. Cambridge University Press, Jan. 2007. ISBN: 978-0-511-26720-8.

[18] Steven Weinberg. *The quantum theory of fields*. Vol. 2. Cambridge university press, 1995.

[19] Richard P Feynman, Albert R Hibbs and Daniel F Styer. *Quantum mechanics and path integrals*. Courier Corporation, 2010.

[20] R. J. Rivers. *Path integral methods in quantum field theory*. Cambridge Monographs on Mathematical Physics. Cambridge University Press, Oct. 1988. ISBN: 978-0-521-36870-4. DOI: 10.1017/CBO9780511564055.

[21] Ulrich Mosel. *Path integrals in field theory: An introduction*. Springer, 2004.

[22] Rainer Sommer. "Scale setting in lattice QCD". In: *PoS* LATTICE2013 (2014), p. 015. DOI: 10.22323/1.187.0015. arXiv: 1401.3270 [hep-lat].

[23] Heinz J. Rothe. *Lattice Gauge Theories : An Introduction (Fourth Edition)*. Vol. 43. World Scientific Publishing Company, 2012. ISBN: 978-981-4365-85-7. DOI: 10.1142/8229.

[24] M. Pilar Hernandez. "Lattice field theory fundamentals". In: *Les Houches Summer School: Session 93: Modern perspectives in lattice QCD: Quantum field theory and high performance computing*. Aug. 2009, pp. 1–91.

[25] Christof Gattringer and Christian B. Lang. *Quantum chromodynamics on the lattice*. Vol. 788. Berlin: Springer, 2010. ISBN: 978-3-642-01850-3. DOI: 10.1007/978-3-642-01850-3.

[26] Holger Bech Nielsen and Masao Ninomiya. *No-go theorum for regularizing chiral fermions*. Tech. rep. Science Research Council, 1981.

[27] Daniel Friedan. "A proof of the Nielsen-Ninomiya theorem". In: *Communications in Mathematical Physics* 85.4 (1982), pp. 481–490.

[28] John B. Kogut and Leonard Susskind. "Hamiltonian Formulation of Wilson's Lattice Gauge Theories". In: *Phys. Rev. D* 11 (1975), pp. 395–408. DOI: 10.1103/PhysRevD.11.395.

[29] K. Symanzik. "Continuum Limit and Improved Action in Lattice Theories. 1. Principles and φ^4 Theory". In: *Nucl. Phys. B* 226 (1983), pp. 187–204. DOI: 10.1016/0550-3213(83)90468-6.

[30] K. Symanzik. "Continuum Limit and Improved Action in Lattice Theories. 2. O(N) Nonlinear Sigma Model in Perturbation Theory". In: *Nucl. Phys. B* 226 (1983), pp. 205–227. DOI: 10.1016/0550-3213(83)90469-8.

[31] M. Luscher and P. Weisz. "On-Shell Improved Lattice Gauge Theories". In: *Commun. Math. Phys.* 97 (1985). [Erratum: Commun.Math.Phys. 98, 433 (1985)], p. 59. DOI: 10.1007/BF01206178.

[32] David B. Kaplan. "A Method for simulating chiral fermions on the lattice". In: *Phys. Lett. B* 288 (1992), pp. 342–347. DOI: 10.1016/0370-2693(92)91112-M. arXiv: hep-lat/9206013.

[33] Yigal Shamir. "Chiral fermions from lattice boundaries". In: *Nucl. Phys. B* 406 (1993), pp. 90–106. DOI: 10.1016/0550-3213(93)90162-I. arXiv: hep-lat/9303005.

[34] Yoshinobu Kuramashi. *Perspectives In Lattice Qcd-Proceedings Of The Workshop*. World Scientific, 2007.

[35] Andrea Shindler. "Twisted mass lattice QCD". In: *Phys. Rept.* 461 (2008), pp. 37–110. DOI: 10.1016/j.physrep.2008.03.001. arXiv: 0707.4093 [hep-lat].

[36] Brian L Hammond, William A Lester and Peter James Reynolds. *Monte Carlo methods in ab initio quantum chemistry*. Vol. 1. World Scientific, 1994.

[37] M Peter Nightingale and Cyrus J Umrigar. *Quantum Monte Carlo methods in physics and chemistry*. 525. Springer Science & Business Media, 1998.

[38] Raimundo R. dos Santos. "Introduction to Quantum Monte Carlo simulations for fermionic systems". In: (2003).

[39] G. Aarts et al. "Properties of the QCD thermal transition with Nf=2+1 flavors of Wilson quark". In: *Phys. Rev. D* 105.3 (2022), p. 034504. DOI: 10.1103/PhysRevD.105.034504. arXiv: 2007.04188 [hep-lat].

[40] Nicholas Metropolis et al. "Equation of state calculations by fast computing machines". In: *The journal of chemical physics* 21.6 (1953), pp. 1087–1092.

[41] DH Weingarten and DN Petcher. "Monte Carlo integration for lattice gauge theories with fermions". In: *Physics Letters B* 99.4 (1981), pp. 333–338.

[42] David JE Callaway and Aneesur Rahman. "Lattice gauge theory in the microcanonical ensemble". In: *Physical review d* 28.6 (1983), p. 1506.

[43] David JE Callaway and Aneesur Rahman. "Microcanonical ensemble formulation of lattice gauge theory". In: *Physical review letters* 49.9 (1982), p. 613.

[44] S. Gusken et al. "Nonsinglet Axial Vector Couplings of the Baryon Octet in Lattice QCD". In: *Phys. Lett. B* 227 (1989), pp. 266–269. DOI: 10.1016/S0370-2693(89)80034-6.

[45] C. Best et al. "Pion and rho structure functions from lattice QCD". In: *Phys. Rev. D* 56 (1997), pp. 2743–2754. DOI: 10.1103/PhysRevD.56.2743. arXiv: hep-lat/9703014.

[46] S. Gusken et al. "Nonsinglet Axial Vector Couplings of the Baryon Octet in Lattice QCD". In: *Phys. Lett. B* 227 (1989), pp. 266–269. DOI: 10.1016/S0370-2693(89)80034-6.

[47] M. Albanese et al. "Glueball Masses and String Tension in Lattice QCD". In: *Phys. Lett. B* 192 (1987), pp. 163–169. DOI: 10.1016/0370-2693(87)91160-9.

[48] Richard Barrett et al. *Templates for the solution of linear systems: building blocks for iterative methods*. SIAM, 1994.

Chapter 2

Introduction to thermal field theory

The system presented in the previous chapter does not fully correspond to the one we are interested in, as the effect of the temperature is not taken into account.

Understanding how the temperature affects the observables of a quantum field theory is a topic of long-standing interest. First, because we do live in a non-zero temperature universe: experimental quantities are never measured at zero temperature; the average temperature of the universe is non-zero, as indicated by cosmic microwave background radiation studies [1]. Second, as the temperature plays an important role in the description of most systems, then, so it does for QCD. For instance, in QCD, the temperature is directly responsible of the transition from a strongly interacting confining state, usually called the hadronic phase, to a weakly interacting system with the properties of a plasma, which is usually referred to as the Quark-Gluon Plasma (QGP) [2]. The reverse transition is believed to have taken place instants after the Big Bang [3], which implies that non-zero temperature QCD is crucial in our understanding of early-universe physics. Moreover, from an experimental point of view, the temperature is believed to be fundamental in the description of heavy-ion collisions, and the interior of dwarf stars [4–6].

2.1 The canonical ensemble of a quantum field theory

In order to immerse a quantum field theory in a thermal bath at temperature T, two formalisms are required: statistical mechanics and quantum field theory. The

combination of both formalisms is referred to as thermal field theory. The canonical ensemble of statistical mechanics [7], which describes the dynamics of a system placed in a thermal bath at fixed temperature T and volume V. The chemical potential is assumed to be zero for all species considered. Non-zero density thermal field theory can be formulated [8] using the grand-canonical ensemble [7], but it leads to severe complications when formulated as a lattice field theory. In this document, we restrict ourselves to zero density and non-zero temperature thermal field theory.

One of the fundamental objects in the canonical ensemble formulation of statistical mechanics is the density operator, defined as

$$\rho(\hat{H}, \beta) = \exp\big(-\beta\hat{H}\big), \tag{2.1}$$

where $\hat{H}$ represents the hamiltonian describing the dynamics of the underlying system, and β is the inverse of the temperature:

$$\beta = \frac{1}{k_B\, T}, \tag{2.2}$$

where k_B is the Boltzmann constant, which is equal to 1 in natural units. The density operator is sometimes called the Boltzmann weight factor, as it can be interpreted as weighting the likelihood of every state of the system. At low temperature, or large β, the states with high energy are suppressed as their probability is small due to the exponentially decaying nature of the density operator. In contrast, at large temperature, the contribution of all states is non-negligible as the exponential decay in eq. (2.1) is damped by the temperature.

From the density operator, we can compute the partition function, defined as

$$Z_\beta = \mathrm{Tr}\left[\rho(\hat{H}, \beta)\right] = \sum_n \langle n|e^{-\beta\hat{H}}|n\rangle, \tag{2.3}$$

where the states $|n\rangle$ form a complete set. The partition function can be interpreted as the sum of the probabilities of all states in the system. As a result, it can be viewed as a cumulative distribution function in probability theory, similar to the path integral in quantum field theory. The partition function of a discrete system is shown in eq. (2.3). In the case in which the hamiltonian allows continuous energy states, then, the sum in eq. (2.3) is replaced by an integral.

From the partition function, one can study any static thermodynamic quantity describing the bulk properties of the system [7], for example: its entropy, pressure, total internal energy, or enthalpy. Those quantities can be defined as the moments of the cumulative distribution function. In contrast, the study of non-static quantities,

that is, quantities that depend on the space-time coordinates, requires the analysis of statistical correlation functions. Correlation functions allow the study of important properties of the system, such as: production rates from a plasma, the spectroscopy of the theory, transport coefficients or shear and bulk viscosities [9–15].

In statistical mechanics, the expectation value of an operator at temperature T is defined as

$$\langle \hat{O} \rangle_\beta = \frac{1}{Z} \operatorname{Tr} \left[e^{-\beta \hat{H}} \hat{O} \right]. \tag{2.4}$$

We can extend this definition to 2 operators,

$$\langle \hat{A}\hat{B} \rangle_\beta = \frac{1}{Z} \operatorname{Tr} \left[e^{-\beta \hat{H}} \hat{A}\hat{B} \right]. \tag{2.5}$$

The operators are time-ordered: $t_A < t_B$. In thermal field theory, the operators correspond to field operators depending on the space-time. In this context, a 2-point correlation function is defined as

$$\langle \hat{O}(\vec{y}, t)\hat{O}^\dagger(\vec{x}, 0) \rangle_\beta = \frac{1}{Z} \operatorname{Tr} \left[e^{-\beta \hat{H}} \hat{O}(\vec{y}, t)\hat{O}^\dagger(\vec{x}, 0) \right]. \tag{2.6}$$

We can manipulate the 2-point correlation function definition in eq. (2.6) to investigate some of its properties:

$$\begin{aligned}
\langle \hat{O}(\vec{y}, t)\hat{O}^\dagger(\vec{x}, 0) \rangle_\beta &= \frac{1}{Z} \operatorname{Tr} \left[e^{-\beta \hat{H}} \hat{O}(\vec{y}, t)\hat{O}^\dagger(\vec{x}, 0) \right] \\
&= \frac{1}{Z} \operatorname{Tr} \left[e^{-\beta \hat{H}} \hat{O}(\vec{y}, t) e^{-\beta \hat{H}} e^{\beta \hat{H}} \hat{O}^\dagger(\vec{x}, 0) \right] \\
&= \frac{1}{Z} \operatorname{Tr} \left[\hat{O}(\vec{y}, t) e^{-\beta \hat{H}} e^{i(-i\beta)\hat{H}} \hat{O}^\dagger(\vec{x}, 0) e^{-i(-i\beta)\hat{H}} \right] \\
&= \frac{1}{Z} \operatorname{Tr} \left[\hat{O}(\vec{y}, t) e^{-\beta \hat{H}} \hat{O}^\dagger(\vec{x}, -i\beta) \right] \\
&= \pm \frac{1}{Z} \operatorname{Tr} \left[e^{-\beta \hat{H}} \hat{O}^\dagger(\vec{x}, -i\beta) \hat{O}(\vec{y}, t) \right] \\
&= \pm \langle \hat{O}^\dagger(\vec{x}, -i\beta) \hat{O}(\vec{y}, t) \rangle_\beta.
\end{aligned} \tag{2.7}$$

In the derivation above, we have made use of the invariance of the trace operator under cyclic permutations. The final sign of the correlation function depends on the spin statistics of the fields, as interchanging two fermionic fields leads to an overall change of sign due to the exclusion principle of fermions.

Equation (2.7) is fundamental in thermal field theory. To see this, we can work in Euclidean space-time by performing a Wick rotation: $t = -i\tau$. Substituting it with τ in eq. (2.7) leads to the following identity:

$$\langle \hat{O}(\vec{y}, \tau)\hat{O}^\dagger(\vec{x}, 0) \rangle_\beta = \pm \langle \hat{O}^\dagger(\vec{x}, \beta)\hat{O}(\vec{y}, \tau) \rangle_\beta. \tag{2.8}$$

Using the standard notation for correlation functions:

$$\langle \hat{O}(y,\tau)\hat{O}^{\dagger}(x,0)\rangle = C(\vec{y}-\vec{x},\tau). \tag{2.9}$$

We can write eq. (2.8) as

$$C(\vec{y}-\vec{x},\tau) = \pm C(\vec{y}-\vec{x},\tau-\beta). \tag{2.10}$$

The identity above states that Euclidean thermal field theory correlation functions are periodic or anti-periodic in time depending on the fields: bosonic correlation functions are periodic, while fermionic are anti-periodic. On the lattice, in order to ensure that this condition is fulfilled, it is common to impose periodic boundary conditions in the time direction for bosonic fields, and anti-periodic conditions for fermionic fields. Additionally, eq. (2.8) implies that the Euclidean time can only take values in the region $\tau \in [0,\beta]$, where β represents the inverse of the temperature.

Placing a quantum field theory in a thermal bath is equivalent to working in a finite Euclidean time, where the maximum time is equal to the inverse of the temperature. The temperature of the system increases as the time direction gets compressed. The limit $\beta \to \infty$ corresponds to zero-temperature field theory.

In order to quantise a thermal field theory, it is common to work in the path integral formalism of quantum mechanics, introduced in Chapter (1). The path integral computed from an initial to a final configuration of the fields is interpreted as the probability of the transition between both states. The transition probability from configuration Φ_a to configuration Φ_b of a system is defined as

$$\langle \Phi_a | e^{-it\hat{H}} | \Phi_b \rangle = \int_{\Phi_a}^{\Phi_b} D\Phi e^{iS[\Phi]}, \tag{2.11}$$

where $\hat{H}$ is the hamiltonian of the system, and $S[\Phi]$ its action. We can compare the path-integral with the continuous version of the partition function in eq. (2.3), which reads

$$Z_\beta = \mathrm{Tr}\left[\rho(\hat{H},\beta)\right] = \int D\Phi \, \langle \Phi | e^{-\beta\hat{H}} | \Phi \rangle. \tag{2.12}$$

Both identities are equivalent once we perform a Wick rotation over the time component in eq. (2.11):

$$\langle \Phi_a | e^{-\tau\hat{H}} | \Phi_b \rangle = \int_{\Phi_a}^{\Phi_b} D\Phi e^{-S[\Phi]} = \int_{\Phi_a}^{\Phi_b} D\Phi e^{-\int d\tau d^3x \mathcal{L}_E[\Phi]}, \tag{2.13}$$

where $\mathcal{L}_E$ represents the Euclidean version of the lagrangian density. Due to the fact that fields are periodic or anti-periodic in time, the time integral in the equation

above is only defined from $\tau = 0$ to $\tau = \beta$. Applying this condition leads to the definition of the thermal path integral,

$$Z_\beta = \int D\Phi \exp\big(- \int_0^\beta d\tau \int d^3x \mathcal{L}_E[\Phi]\big). \tag{2.14}$$

More information about thermal field theory can be found in Refs. [16–18].

The canonical partition function is almost identical to the Euclidean path integral in eq. (1.24) once periodic/anti-periodic boundary conditions on the fields are imposed. Including the effects of the temperature in lattice field theory is just a matter of imposing a set of boundary conditions on the fields, and interpreting the inverse of the temporal lattice length as the temperature. Therefore, the temperature of a lattice system can be computed using

$$T = \frac{1}{L_\tau} = \frac{1}{N_\tau\, a}. \tag{2.15}$$

As a result, we can modify the temperature of the system by either varying the number of points in the temporal direction while keeping the lattice spacing fixed, or by modifying the lattice spacing maintaining a constant temporal size.

The previously introduced thermal field theory formalism is expressed in terms of a complex time τ. However, we are not restricted to Euclidean thermal field theory. A standard introduction to real-time thermal field theory can be found in Ref. [16]. A recent review of the topic can be found in Ref. [19]. It is worth noting that the Euclidean time formalism is the most popular one in lattice field theory.

2.1.1 Thermal and screening correlation functions

In thermal field theory, for each particular pair of field operators defining a 2-point correlation function, there exist two different projections of the same correlation function. These projections contain different physics due to the fact that placing the system in a thermal bath breaks Lorentz invariance: the system treats differently temporal and spatial directions. The projections are called the thermal projection and the screening projection, which respectively lead to the so-called thermal correlation functions and screening correlation functions. To construct these projections, we use the fact that 2-point correlation functions are functions of the space-time: $C(\tau, \vec{y})$.

We can project the bare correlation function $C(\tau, \vec{y})$ into different representations by Fourier transforming the coordinates. One can Fourier transform both space and

time coordinates:

$$C(\omega, \vec{k}) = \int \frac{d\tau d^3x}{(2\pi)^4} \, C(\tau, \vec{x}) e^{i\omega\tau} e^{i\vec{k}\vec{x}}, \tag{2.16}$$

which produces the energy-momentum representation of the correlation function. As correlation functions are periodic in time, the frequency space forms a discrete set called the Matsubara frequencies [16–18]. For the shake of simplicity, in our current discussion, we treat the frequency space as continuous. However, the integral over ω should be replaced by a sum over the Matsubara frequencies, which correspond to $\omega_n = 2\pi nT;\ n \in \mathbb{N}$ for bosonic correlation functions.

In addition, a mixed representation of the correlation function can be obtained by integrating the temporal or spatial components of the correlation function. For example, the so-called thermal projection of the correlation function is obtained by Fourier transforming only the spatial coordinates of the original function,

$$C(\tau, \vec{k}) = \int \frac{d^3x}{(2\pi)^3} \, C(\tau, \vec{x}) e^{i\vec{k}\vec{x}}. \tag{2.17}$$

In contrast, the screening projection is generated by only Fourier transforming the temporal coordinates,

$$C(\omega, \vec{x}) = \int \frac{d\tau}{2\pi} \, C(\tau, \vec{x}) e^{i\omega\tau}. \tag{2.18}$$

An example stressing the difference between thermal and screening correlation functions is the so-called Debye mass, which is exclusively accessed from the analysis of screening correlations in thermal field theory. More information about the Debye mass and screening correlation functions can be found in Ref. [16].

2.2 Spectral functions

As stated before, Euclidean thermal field theory correlation functions can be used to investigate the non-static properties of the medium. However, in order to access the real-time dynamics of the system, we need to perform an analytic continuation from complex time τ to real time t. This can be done through the so-called spectral function. In addition to allowing the analytic continuation from Euclidean to Minkowskian space-time, spectral functions also grant access to the whole spectrum of states contributing to the correlation function, which is particularly useful to understand how the properties of the medium, such as the temperature, affect the physical states described by the theory.

The formal definition of a spectral function of a pair of operators $\hat{O}$ and $\hat{O}^\dagger$ is

$$\rho(t,\vec{x}) = \langle[\hat{O}(t,\vec{x}),\hat{O}^\dagger(0,\vec{0})]\rangle. \tag{2.19}$$

Note that the same label, ρ, is employed to refer to the density operator in eq. (2.1), but both quantities are completely unrelated. The equation above is specific for bosonic field operators. The fermionic version replaces the commutator, $[A,B] = AB - BA$, with the anti-commutator, $\{A,B\} = AB + BA$. As our interest lies in mesonic operators, which are indeed bosonic, we only discuss the former case. In the spectral function definition, we place the source operator $\hat{O}^\dagger$ at the origin of coordinates. This is possible due to translational invariance of systems in thermal equilibrium.

One can decompose eq. (2.19) into two independent real-time correlation functions,

$$\rho(t,\vec{x}) = \langle\hat{O}(t,\vec{x})\hat{O}^\dagger(0,\vec{0})\rangle - \langle\hat{O}^\dagger(0,\vec{0})\hat{O}(t,\vec{x})\rangle. \tag{2.20}$$

These two correlation functions are related to the so-called retarded and advanced correlation functions, defined as

$$C_R(t,\vec{x}) = i\theta(t)\langle[\hat{O}(t,\vec{x}),\hat{O}^\dagger(0,\vec{0})]\rangle = C_A(-t,-\vec{x}), \tag{2.21}$$

where $\theta(t)$ is the step function:

$$\theta(t) = \left\{ \begin{array}{ll} 1 & \text{if } t \geq 0 \\ 0 & \text{otherwise} \end{array} \right\}. \tag{2.22}$$

Substituting the definitions of C_R and C_A into eq. (2.19) allows us to write

$$\rho(t,\vec{x}) = -i[C_R(t,\vec{x}) - C_A(t,\vec{x})]. \tag{2.23}$$

From the equation above, we learn that all real-time correlation functions can be written in terms of the spectral function. More information about these identities can be found in Refs. [16, 18, 20].

Manipulating eq. (2.21) allows us to write a relationship between the spectral function and the retarded correlator,

$$C_R(t,\vec{x}) = i\theta(t)\rho(t,\vec{x}). \tag{2.24}$$

The energy-momentum representation of the equation above can be obtained by computing the following Fourier transform:

$$C_R(\omega,\vec{k}) = \int \frac{d\omega d^3k}{(2\pi)^4} i\theta(t)\rho(t,\vec{x})e^{i\omega t}e^{i\vec{k}\vec{x}}. \tag{2.25}$$

To compute the integral above, we employ the following identity

$$i\theta(t) = -\int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \frac{e^{-i\omega t}}{\omega + i\epsilon}, \tag{2.26}$$

where ϵ is a positive real number. The identity above allows us to perform the integrals in eq. (2.25), and arrive to

$$C_R(\omega, \vec{k}) = \int_{-\infty}^{\infty} \frac{d\omega'}{2\pi} \frac{\rho(\omega', \vec{k})}{\omega' - \omega - i\epsilon}. \tag{2.27}$$

The equation above is the dispersion relation, whose name comes from the fact that it looks like a continuous sum of propagating particles at different energies ω — recall the interpretation of poles in the quantum field theory propagators as particles. In principle, for the same set of field operators, there is a different spectral function at each temperature. This implies that having access to the spectral function of a particular 2-point correlation function at different temperatures grants us information about the possible effects of the temperature on the system.

There exists a similar integral relationship relating spectral functions and advanced correlation functions. More information about this can be found in Refs. [18, 20].

In addition, we can define the Euclidean version of a bosonic 2-point correlation function,

$$C_E(\tau, \vec{x}) = \langle \hat{O}(\tau, \vec{x}) \hat{O}^\dagger(0, \vec{0}) \rangle. \tag{2.28}$$

This definition is equivalent to the one written in eq. (2.6). Note that τ satisfies $0 \leq \tau \leq \beta$, being β the inverse of the temperature. Equation (2.28) can be Fourier transformed:

$$C_E(\omega_n, \vec{k}) = \int_0^{\beta} d\tau e^{i\omega_n \tau} C_E(\tau, \vec{k}). \tag{2.29}$$

The Fourier frequencies ω_n are the previously introduced Matsubara frequencies.

The time-momentum mixed representation of eq. (2.28) can be computed by summing over all the Matsubara frequencies in eq. (2.29),

$$C(\tau, \vec{k}) = T \sum_n e^{-i\omega_n \tau} C_E(\omega_n, \vec{k}), \tag{2.30}$$

which is equivalent to performing a discrete Fourier transform. In the equation above T represents the temperature.

As there must exist an analytic continuation from imaginary-time to real-time that links both real-time and imaginary-time correlation functions, then, there should

be a connection between C_E and ρ. This relationship is similar to the one written in eq. (2.27), and reads

$$C_E(\omega_n, \vec{k}) = \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \frac{\rho(\omega, \vec{k})}{\omega - i\omega_n}, \tag{2.31}$$

which implies that

$$C_R(\omega, \vec{k}) = C_E(i\omega_n \to \omega + i\epsilon, \vec{k}). \tag{2.32}$$

As a result, the spectral function can be used to access both real-time and imaginary-time correlation functions: it serves as a link between both formalisms.

However, obtaining the associated spectral function from a Euclidean correlation function is difficult, as Euclidean correlation functions are typically not accessible analytically. This complicates the computation of the integral in eq. (2.31) due to its complex and diverging nature. To make eq. (2.31) numerically tractable, we start by substituting it into eq. (2.30), leading to

$$C_E(\tau, \vec{k}) = \int_{-\infty}^{\infty} \frac{d\omega'}{2\pi} T \sum_n \frac{e^{-i\omega_n \tau}}{\omega' - i\omega_n} \rho(\omega', \vec{k}). \tag{2.33}$$

To proceed, we define the following quantity

$$\tilde{K}(\tau, \omega) = T \sum_n \frac{e^{-i\omega_n \tau}}{\omega' - i\omega_n}. \tag{2.34}$$

The right-hand side of $\tilde{K}$ is a well-known identity in thermal field theory [16, 17], which can be rewritten as

$$\tilde{K}(\tau, \omega) = e^{-\omega\tau}\Big[1 + \frac{1}{e^{\omega/T} - 1}\Big]. \tag{2.35}$$

In the case in which the field operators $\hat{O}$ and $\hat{O}^\dagger$ are hermitian, then the spectral function is odd,

$$\rho(-\omega, \vec{k}) = -\rho(\omega, \vec{k}), \tag{2.36}$$

which means that we can restrict the integral in eq. (2.33) to the positive real numbers. We can use this information to arrive at the integral relationship connecting spectral functions with their associated Euclidean correlation functions:

$$C_E(\tau, \vec{k}) = \int_0^{\infty} d\omega \; K(\tau, \omega) \rho(\omega, \vec{k}), \tag{2.37}$$

where $K(\tau, \omega)$ is known as the kernel of the transformation, whose definition is

$$K(\tau, \omega) = \tilde{K}(\tau, \omega) - \tilde{K}(\tau, -\omega) = \frac{\cosh(\omega\,(\tau - \frac{1}{2T}))}{\sinh(\omega/2T)}. \tag{2.38}$$

2.2.1 Low temperature spectral functions

There exists a connection between the spectral function relationship and the spectral decomposition shown in eq. (1.45). To simplify this discussion, the spectral decomposition is introduced again:

$$C_E(\tau, \vec{k}) = \sum_n \langle \emptyset | O_2 | n \rangle \langle n | O_1 | \emptyset \rangle e^{-E_n \tau}. \tag{2.39}$$

In the case in which hermitian sink and source operators are employed, we have

$$C_E(\tau, \vec{k}) = \sum_n |\langle \emptyset | \hat{O} | n \rangle|^2 e^{-E_n \tau}. \tag{2.40}$$

To find the connection between both eq. (2.40) and eq. (2.37), we can start by taking the low temperature limit of eq. (2.38),

$$\lim_{T \to 0^+} K(\tau, \omega) = \exp(-\omega\tau). \tag{2.41}$$

Inserting this limit into eq. (2.37) allows us to write

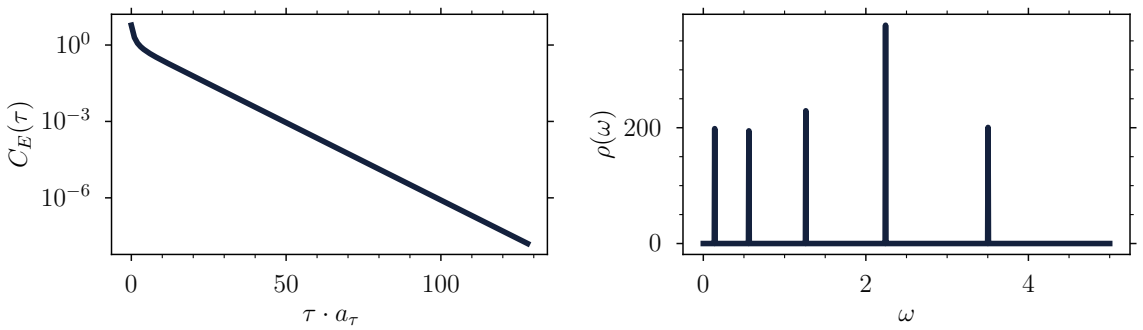
$$C_E(\tau, \vec{k}) = \int_0^\infty e^{-\omega\tau} \rho(\omega, \vec{k}). \tag{2.42}$$

This expression looks very similar to the spectral decomposition if we define the following low-temperature spectral function:

$$\rho(\omega, \vec{k}) = \sum_n |\langle \emptyset | O | n \rangle|^2 \, \delta(\omega - E_n) \tag{2.43}$$

Substituting this spectral function into eq. (2.42), and performing the Dirac delta integrals, we arrive at eq. (2.40).

The low-temperature relationship between eq. (2.40) and eq. (2.37) implies that the spectral function corresponds to a sum of independent and isolated peaks in this regime: each peak in the spectral function is located at a different energy. In the case in which the external momentum is zero, then the peaks are placed at the masses of the different states. Figure (2.1) contains a visual representation of this connection, where a randomly generated spectral function is employed to obtain its Euclidean correlation function through the application of eq. (2.40).



(a) Correlation function **(b)** Spectral function

Figure 2.1: Example of a low-temperature Euclidean correlation function and its corresponding spectral function. The spectral function is randomly generated, while the Euclidean correlation function is computed using eq. (2.42).

Chapter 2. References

[1] Yi-Kuan Chiang et al. "The Cosmic Thermal History Probed by Sunyaev–Zeldovich Effect Tomography". In: *Astrophys. J.* 902.1 (2020), p. 56. DOI: 10.3847/1538-4357/abb403. arXiv: 2006.14650 [astro-ph.CO].

[2] Joseph Kapusta, Berndt Müller and Johann Rafelski. *Quark-gluon plasma: theoretical foundations: an annotated reprint collection*. Gulf Professional Publishing, 2003.

[3] Kohsuke Yagi, Tetsuo Hatsuda and Yasuo Miake. *Quark-gluon plasma: From big bang to little bang*. Vol. 23. Cambridge University Press, 2005.

[4] T. Altherr. "Introduction to thermal field theory". In: *Int. J. Mod. Phys. A* 8 (1993), pp. 5605–5628. DOI: 10.1142/S0217751X93002216. arXiv: hep-ph/9307277.

[5] Wit Busza, Krishna Rajagopal and Wilke van der Schee. "Heavy Ion Collisions: The Big Picture, and the Big Questions". In: *Ann. Rev. Nucl. Part. Sci.* 68 (2018), pp. 339–376. DOI: 10.1146/annurev-nucl-101917-020852. arXiv: 1802.04801 [hep-ph].

[6] J. B. Kogut and M. A. Stephanov. *The phases of quantum chromodynamics: From confinement to extreme environments*. Vol. 21. Cambridge University Press, Dec. 2004. ISBN: 978-0-521-14338-7.

[7] Lev Davidovich Landau and Evgenii Mikhailovich Lifshitz. *Statistical Physics: Volume 5*. Vol. 5. Elsevier, 2013.

[8] Gert Aarts. "Introductory lectures on lattice QCD at nonzero baryon number". In: *J. Phys. Conf. Ser.* 706.2 (2016), p. 022004. DOI: 10.1088/1742-6596/706/2/022004. arXiv: 1512.05145 [hep-lat].

[9] Harvey B. Meyer. "The errant life of a heavy quark in the quark-gluon plasma". In: *New J. Phys.* 13 (2011), p. 035008. DOI: 10.1088/1367-2630/13/3/035008. arXiv: 1012.0234 [hep-lat].

[10] Debasish Banerjee et al. "Heavy Quark Momentum Diffusion Coefficient from Lattice QCD". In: *Phys. Rev. D* 85 (2012), p. 014510. DOI: 10.1103/PhysRevD.85.014510. arXiv: 1109.5738 [hep-lat].

[11] Harvey B. Meyer. "A Calculation of the bulk viscosity in SU(3) gluodynamics". In: *Phys. Rev. Lett.* 100 (2008), p. 162001. DOI: 10.1103/PhysRevLett.100.162001. arXiv: 0710.3717 [hep-lat].

[12] N. Astrakhantsev, Viktor Braguta and Andrey Kotov. "Temperature dependence of shear viscosity of $SU(3)$–gluodynamics within lattice simulation". In: *JHEP* 04 (2017), p. 101. DOI: 10.1007/JHEP04(2017)101. arXiv: 1701.02266 [hep-lat].

[13] Borsányi, Sz. and Fodor, Zoltan and Giordano, Matteo and Katz, Sandor D. and Pasztor, Attila and Ratti, Claudia and Schäfer, Andreas and Szabo, Kalman K. and Tóth, Balint C. "High statistics lattice study of stress tensor correlators in pure $SU(3)$ gauge theory". In: *Phys. Rev. D* 98.1 (2018), p. 014512. DOI: 10.1103/PhysRevD.98.014512. arXiv: 1802.07718 [hep-lat].

[14] Harvey B. Meyer. "A Calculation of the bulk viscosity in SU(3) gluodynamics". In: *Phys. Rev. Lett.* 100 (2008), p. 162001. DOI: 10.1103/PhysRevLett.100.162001. arXiv: 0710.3717 [hep-lat].

[15] N. Yu. Astrakhantsev, V. V. Braguta and A. Yu. Kotov. "Temperature dependence of the bulk viscosity within lattice simulation of $SU(3)$ gluodynamics". In: *Phys. Rev. D* 98.5 (2018), p. 054515. DOI: 10.1103/PhysRevD.98.054515. arXiv: 1804.02382 [hep-lat].

[16] Michel Le Bellac. *Thermal field theory*. Cambridge university press, 2000.

[17] Joseph I Kapusta and Charles Gale. *Finite-temperature field theory: Principles and applications*. Cambridge university press, 2006.

[18] Mikko Laine and Aleksi Vuorinen. *Basics of Thermal Field Theory*. Vol. 925. Springer, 2016. DOI: 10.1007/978-3-319-31933-9. arXiv: 1701.01554 [hep-ph].

[19] Torbjörn Lundberg and Roman Pasechnik. "Thermal Field Theory in real-time formalism: concepts and applications for particle decays". In: *Eur. Phys. J. A* 57.2 (2021), p. 71. DOI: 10.1140/epja/s10050-020-00288-5. arXiv: 2007.01224 [hep-th].

[20] Gert Aarts and Aleksandr Nikolaev. "Electrical conductivity of the quark-gluon plasma: perspective from lattice QCD". In: *Eur. Phys. J. A* 57.4 (2021), p. 118. DOI: 10.1140/epja/s10050-021-00436-5. arXiv: 2008.12326 [hep-lat].

Chapter 3

Meson thermal masses at non-zero temperature

The fate of hadrons under extreme conditions is one of the outstanding questions in the theory of strong interactions, QCD. As the temperature of the system increases, the low-temperature hadronic gas [1]— with confined quarks and broken chiral symmetry — evolves into a quark-gluon plasma (QGP), in which the lightest quarks are deconfined and chiral symmetry is restored. Evidence for this phenomenon comes from non-perturbative simulations of QCD on the lattice: from analysis of the pressure, entropy and quark number susceptibility across several temperatures, to studies on the chiral condensate and its susceptibility [2–7].

Due to the aforementioned evidence that such a transition exists in QCD, then, a first-principles study on the spectrum of mesonic states across several temperatures is expected to grant us meaningful information about the medium. Understanding how the temperature affects the spectrum of QCD is key to exploring deconfinement, as well as the restoration of $SU(2)_A$ chiral symmetry — the quark mass term in QCD explicitly breaks the invariance of the lagrangian under arbitrary $SU(2)$ rotations on the right-handed and left-handed quark spinors [8]. As a result, the goal of our studies is to investigate the temperature dependence of the ground state masses of several mesonic states. We do not focus on baryon matter, as some results in this sector can already be found in Refs. [9–11].

It is worth noting that we only deal with thermal correlation functions, that is, correlation functions expressed in their time-momentum representation: $C(\tau; \vec{k})$. Consequently, the targets of our analysis are the thermal masses of different mesonic correlation functions; see Chapter (2) for more information.

In order to investigate the temperature dependence of the masses of mesonic

QCD states, we analyse 2-point Euclidean thermal correlation functions extracted from lattice QCD simulations; information about the simulations can be found in Appendix (A) and references therein. In principle, the complete spectrum of states contained in a correlation function computed at an arbitrary temperature can be accessed through its associated spectral function; for a short introduction to spectral functions in thermal field theory, see Chapter (2). However, building reliable spectral functions at arbitrary temperatures is not possible at the date in which this document is being written. As a result, we restrict ourselves to a simpler and conservative spectroscopy analysis in which simulated correlation functions are fitted to functional models that try mimicking the contribution of each state to the correlation function. This regression-based analysis is common in lattice field theory simulations, and, although conceptually simple, it is difficult to apply in practice due to several factors, such as: the lack of prior information while building the models, the presence of multi-state contributions to the correlation functions, or the fact that Monte Carlo data is correlated.

The goal of the analysis presented in this chapter is to extract the ground state mass of a given mesonic 2-point thermal correlation function at a particular temperature T: $C(\tau;T)$. Although we allow multi-state contributions to our models, we do not aim to estimate any excited states. As a consequence, higher-order states can be regarded as control variables in our models; we do not report any excited states in our results. In order to estimate the ground state mass from a given correlation function, we apply a data analysis procedure derived from the one presented in Ref. [12]. In our methodology, no final result is manually selected, all estimates are systematically produced according to some predefined metrics.

This chapter is organised into two main sections: in the first one, we discuss in detail the methodology employed to analyse our mesonic correlation functions; once the methodology is presented, we present and discuss some results extracted employing the aforementioned methodology The results section is divided into three main subsections: the first one contains some general comments on the estimated temperature dependence of mesonic ground state masses; the second one discusses the restoration of the $SU(2)_A$ chiral symmetry through the degeneracy of the $\rho(770)$ and $a_1(1260)$ states; the last one contains a detailed analysis of the $D_{(s)}$ mesonic sector in the hadronic phase of QCD.

3.1 Methodology

The main goal of our analysis is to generate the best possible estimate of the population ground state mass M_0 contributing to a particular thermal correlation function

$C(\tau, T)$. The methodology derived in this section can also be applied to correlation functions with non-zero external momenta.

In order to estimate the ground state mass from a particular Euclidean correlation function, first, we need to specify how each state contributes to it. In our context, the contribution of each state is modelled employing a parametric function, referred to as a 1-state model. By combining several 1-state models, we generate a global model that hopefully captures the relevant information contained in the correlation function.

At zero temperature, the contribution of each state to the correlation function is well-described by the spectral decomposition of the correlation function — see eq. (1.45). At low temperature, correlation functions can be described in terms of independent asymptotic and isolated states. As a result, their zero-temperature spectral function can be decomposed into an infinite sum of separable δ-distributions of varying amplitudes, each of them located at a different energy:

$$\rho(\omega; T = 0) = \sum_{s=0}^{\infty} A_s \, \delta(\omega - M_s). \tag{3.1}$$

Through the following integral transform, sometimes called the spectral relationship,

$$C(\tau; T) = \int_0^{\infty} d\omega \, K(\tau, \omega; T) \rho(\tau, T), \tag{3.2}$$

we can associate spectral functions and Euclidean correlation functions. In the equation above, K represents the kernel of the transformation, defined as

$$K(\tau, \omega; T) = \frac{\cosh(\omega \, (\tau - \frac{1}{2T}))}{\sinh(\frac{\omega}{2T})}, \tag{3.3}$$

where T represents the temperature of the system. In Chapter (2), we learnt that the temperature of the system is related to the inverse of the temporal length. As a result, on the lattice,

$$T = \frac{1}{N_\tau a_\tau}, \tag{3.4}$$

where N_τ is the number of points in the temporal direction and a_τ is the lattice spacing in the temporal direction.

Integrating the right-hand side of eq. (3.2) with eq. (3.1) and eq. (3.3) leads to the following correlation function functional form:

$$C(\tau; T) = \sum_{s=0}^{\infty} A_s \, \cosh\big(M_s \, (\tau - \frac{1}{2T})\big). \tag{3.5}$$

The amplitudes in eq. (3.5) absorbed the τ-independent denominator of the kernel to simplify the notation.

From eq. (3.5), we can model the contribution of each independent state to the correlation function using the following 1-state parametric model:

$$f_s(\tau; M_s, A_s) = A_s \cosh\Big(M_s\,(\tau - \frac{1}{2T})\Big). \tag{3.6}$$

Sometimes, the parameters defining a 1-state model are encapsulated using a tuple: $\theta_s \equiv (M_s, A_s)$. The equation above depends on the temperature through N_τ — see Chapter (2).

Without loss of generality, the masses in eq. (3.5) can be hierarchically ordered: $M_s > M_{s'}$ if $s > s'$. In this ordered version of eq. (3.5), the lowest mode ($s = 0$) represents the ground state. All other states are referred to as excited states.

As stated before, the QCD medium is expected to be on two different phases: the hadronic phase, encountered at temperatures below the pseudocritical temperature of the system, T_c, and the QGP, found at $T > T_c$. The pseudocritical temperature is defined using the inflection point of the chiral condensate; see Appendix (A) for more information. The model displayed in eq. (3.5) is expected to be valid at low temperatures. However, its reliability is compromised as the temperature increases. Due to in-medium thermal effects, the independent states encountered at low temperature can broaden and mix, making the 1-state models in eq. (3.6) inadequate. This problem is even more severe at $T \gg T_c$, where non-hadronic collective excitations are expected to dominate the lightest degrees of freedom of the system.

Understanding the general functional form that models the contribution of each state to a QCD 2-point correlation function at an arbitrary temperature is still an open problem. Currently, not enough information is available to accurately model high temperatures. Consequently, we adopt a conservative approach in which the same functional model, corresponding to eq. (3.5), is assumed to be valid at all temperatures. We acknowledge that this simplistic model cannot capture all the information contained in a particular correlation functions once the temperature increases. However, we believe it is necessary to test the limits of the model with the temperature as a step towards further progress.

It is worth stressing that accepting the validity of eq. (3.5) at all temperatures is equivalent to assuming that the only way in which the states contributing to a particular correlation function can be affected by the temperature is through a possible shift in their masses and amplitudes: $\theta = \theta(T)$. As a result, under this assumption, the functional form of the spectral function does not vary with the temperature: all states remain independent and separable at all temperatures. From now on, we omit

all references to the temperature in our models, as the N_τ-dependence of eq. (3.5) implies that all models are temperature dependent.

The global model in eq. (3.5) contains an infinite number of states. Nevertheless, the information available in correlation functions simulated on the lattice is finite. As a result, the decomposition in eq. (3.5) must be truncated. A truncation of eq. (3.5) at order e models the contribution of the first e states to the correlation function; each state contributes according to eq. (3.6). A truncated version of eq. (3.5) at order e contains $N_\theta = 2\,e$ free parameters. An index s is used to label each independent state in an e-state model; the label runs from $s = 0$, representing the ground state, to $s = e - 1$, representing the highest-order state in the model.

A truncation of eq. (3.5) at order e is labelled

$$F_e(\tau;\theta) = \sum_{s=0}^{e-1} f_s(\tau;\theta_s) = \sum_{s=0}^{e-1} A_s \cosh\big(M_s\,(\tau - \frac{1}{2T})\big), \tag{3.7}$$

where $\theta = \{\theta_s \,|\, s \in [0, e-1]\}$ is the set of all free-parameters in the model. The order at which eq. (3.5) is truncated is a trade-off between bias and variance: including numerous states in the model makes the model more realistic, but, at the same time, it also increases the number of free-parameters to be determined, which complicates the regression.

In our analysis, we fix the maximum number of states in a given model to be $N_e = 4$, which implies that our largest model is

$$F_{e=4}(\tau;\theta) = \sum_{s=0}^{3} f_s(\tau;\theta_s). \tag{3.8}$$

Once the largest model is selected, we can construct several sub-models by truncating the largest model at all previous orders: $j < N_e$. The set composed by all sub-models of F_{N_e} is

$$\mathcal{M}[N_e] = \Big\{ F_e(\tau;\theta) \,|\, e \in [1, N_e] \Big\}. \tag{3.9}$$

In our particular case, $N_e = 4$, and our collection of sub-models is

$$\mathcal{M}[N_e = 4] = \Big\{ F_1(\tau;\theta),\, F_2(\tau;\theta),\, F_3(\tau;\theta),\, F_4(\tau;\theta) \Big\}. \tag{3.10}$$

The fact that the masses are distinguishable and ordered in eq. (3.5) allows the definition of a useful concept: the fit window. A fit window starting at time τ_0 and finishing at time τ_f is defined as the following discrete interval:

$$FW[\tau_0, \tau_f] = [\tau_0, \tau_f] = \{\tau_0, \tau_0 + a_\tau, \tau_0 + 2a_\tau, \ldots, \tau_f\}. \tag{3.11}$$

A particular fit window defines a set of correlation function values: different fit windows select different regions in the correlation function. The biggest fit window available in our analysis is $FW[0, N_\tau]$, which contains the complete correlation function. Figure (3.1) displays some fit windows over an artificial correlation function.

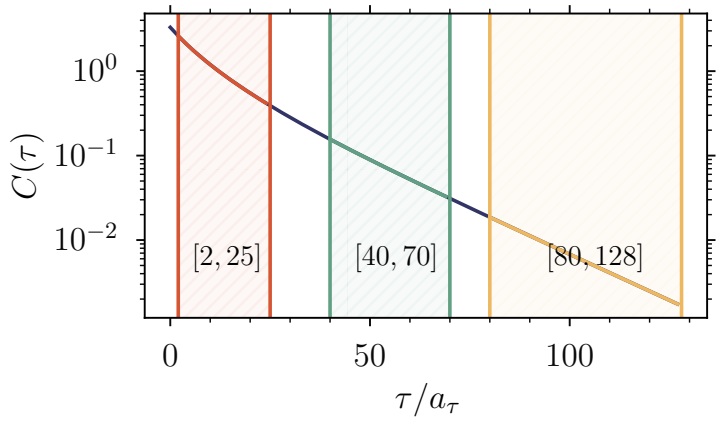


Figure 3.1: Visual examples of three fit window defined on an artificial correlation function of size $N_\tau = 256$. As bosonic correlation functions are periodic in time with period $N_\tau/2$, only half of the correlation function is displayed. The fit windows are defined according to eq. (3.11).

In the case in which the ground-state mass is significantly lighter than any other mass in eq. (3.7), $M_0 \ll M_{s>0}$, then $C(\tau)$ is dominated by its lowest energy mode at fit windows close to the middle of the lattice: $FW[\tau_0 \to N_\tau/2, \tau_f = N_\tau/2]$. In contrast, close to the source operator, $\tau_0 \to 0$, the contribution of all states is non-negligible. In this situation, the correlation function is said to be contaminated by excited states.

When the correlation function is dominated by its lowest energy state, it exhibits a plateau. In this plateau, the logarithm of the correlation function shows an approximate linear dependence:

$$\log[C(\tau \to N_\tau/2)] \simeq -M_0\,\tau. \tag{3.12}$$

To see this, take the logarithm of eq. (3.5) at $\tau \to N_\tau/2$, where all excited states are exponentially suppressed. Provided that a clear plateau is present, then, a simple 1-state model should be sufficient to completely model the large-τ region of the correlation function, which enables the extraction of a good estimate of the ground state mass. However, this situation is rarely encountered in most real correlation

functions. The absence of clear plateaus, combined with the fact that no prior information on the values of the parameters is available, implies that only employing 1-state models in the analysis is likely to lead to an incorrect estimate of the ground state mass, as unreasonable bias is included in the analysis. Hence, one should always take into account multi-state contributions to the correlation function, even when the target is only the ground-state mass.

From now on, in order to simplify our notation, we assume that τ_f is kept fixed to $\tau_f = N_\tau/2$ in eq. (3.11) for all fit windows considered. This implies that the notation $FW[\tau_0]$ is equivalent to $FW[\tau_0, \tau_f = N_\tau/2]$. We analyse the effects of varying τ_f in Subsection (3.1.5).

Our methodology can be classified as a non-linear parametric regression analysis in which all correlation functions considered are modelled using a truncation of eq. (3.5) at order N_e. However, due to the peculiarities of the data, a naive regression analysis is unlikely to produce reliable estimates of the parameters. As a result, it is important to understand the properties of lattice correlation function data in order to build a correct regression analysis.

3.1.1 Definitions and properties of lattice correlation function data

After computing the mesonic thermal correlation functions from the quark propagators evaluated via a sparse matrix inversion on some lattice configurations, we have access to N_b different estimates of a given population thermal correlation function $C(\tau; T)$. For each of these estimates, there are N_τ different Euclidean times at which the correlation function is measured. With the interest of simplifying the notation, we drop the temperature label T in the correlation function. The following discussion is T-independent.

We assume that our sample of estimates of $C(\tau)$ are generated in a thermalised Monte Carlo chain. Moreover, we also assume that the elapsed computer time between sequential configuration measurements is long enough so that the Markovian autocorrelation within ensembles is small. Both assumptions imply that, at fixed τ, all estimates of $C(\tau)$ can be treated as independent and identically distributed random variables. However, since the same ensemble is employed to measure all Euclidean times in a particular estimate of $C(\tau)$, the data is expected to be highly correlated in τ. Figure (3.2) shows an example of the correlation encountered in a real lattice correlation function. The correlation matrix Ξ is defined as

$$\Xi_{ij} = \frac{\Sigma_{ij}}{\sigma_i \sigma_j}, \tag{3.13}$$

where Σ represents the covariance matrix and σ_i the standard deviation of the i^{th} estimate in the signal. The correlation matrix has a value of 1.0 for perfectly linearly correlated variables, and -1.0 for linearly anti-correlated variables. In general, we expect Monte Carlo data to be positively correlated: $\Xi_{ij} > 0$.

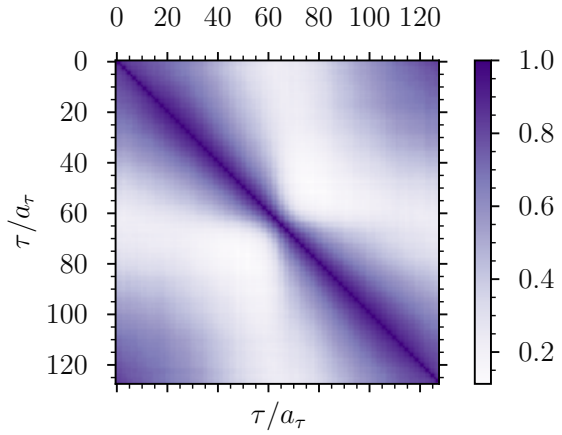


Figure 3.2: Correlation matrix of a real lattice QCD simulated correlation function. The correlation matrix is computed using eq. (3.13).

The sample composed by all estimates of the population correlation function can be viewed as a collection of N_b signals of size N_τ. An unbiased estimate of $C(\tau)$ can be computed through their sample average, $\hat{C}(\tau)$, defined as

$$\hat{C}(\tau) = \frac{1}{N_b} \sum_{b=1}^{N_b} C_b(\tau). \tag{3.14}$$

The variance of $\hat{C}(\tau)$ can be estimated using the standard error of the sample mean. The variance of the sample mean decreases proportionally with the number of samples available.

As stated before, we model $C(\tau)$ using a truncation of eq. (3.5) at order e, which implies that we assume

$$C(\tau) = F_{N_e}(\tau; \theta), \tag{3.15}$$

that is, F_{N_e} is assumed to be the correct model of the population correlation function. We also assume that all discrepancies between the computed data, $\hat{C}(\tau)$, and model, $C(\tau)$, are due to statistical errors, which can be modelled using an additive noise

random variable $u(\tau)$. In general,

$$\hat{C}(\tau) = C(\tau) + u(\tau) = F_{N_e}(\tau;\theta) + u(\tau) = \sum_{s=0}^{N_e-1} f_s(\tau;\theta_s) + u(\tau) \tag{3.16}$$

Due to the fact that $\hat{C}(\tau)$ is an unbiased estimator of $C(\tau)$, the expected value of the noise is zero:

$$\mathbb{E}[u(\tau)] = 0 \quad \forall\, \tau. \tag{3.17}$$

The standard deviation of $u(\tau)$ is equal to the standard error of $\hat{C}(\tau)$. As a result, the noise is not assumed homoskedastic, that is, it is not conditionally independent on τ: the standard errors of $\hat{C}(\tau)$ tend to depend on τ, as can be seen in Figure (3.3).

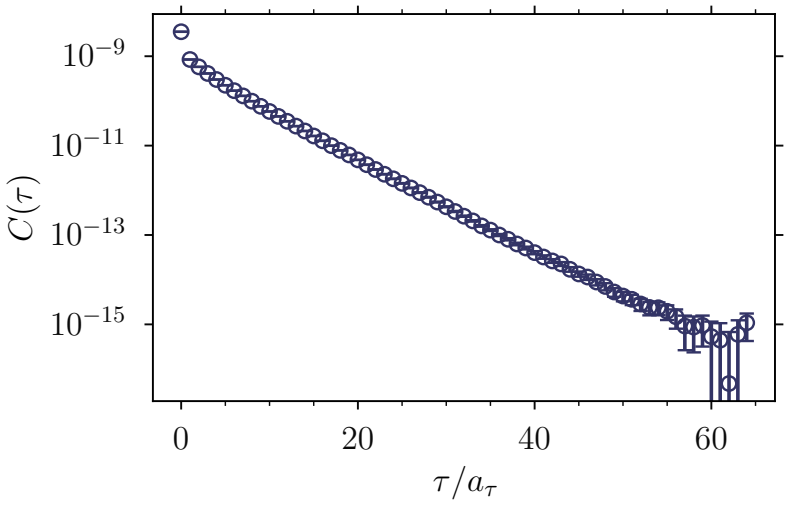


Figure 3.3: Lattice estimate of the axial vector (strange-strange) ss correlation function computed at the lowest temperature available in our simulations, $T = 47$ MeV. The correlation function is only plotted up to $\tau/a_\tau = N_\tau/2$. Detailed information about our simulations can be found in Appendix (A), while information about the physical states included in the simulations can be found in Appendix (B).

Having defined our model, we are in the position to discuss how we can estimate the values of the parameters θ that make eq. (3.16) as realistic as possible given the available data. To do so, we employ the framework of maximum likelihood. In this framework, we aim to find the values of the parameters that maximise the probability of having computed our particular estimate of $C(\tau)$ assuming that F_e is the true underlying population model with parameter values θ; the estimate $\hat{C}(\tau)$ is a

collection of N_τ random variables, one per Euclidean time. The likelihood of having obtained the data is represented by $P(\hat{C}(\tau)|\theta)$, which is equivalent to $P(u(\tau)|\theta)$.

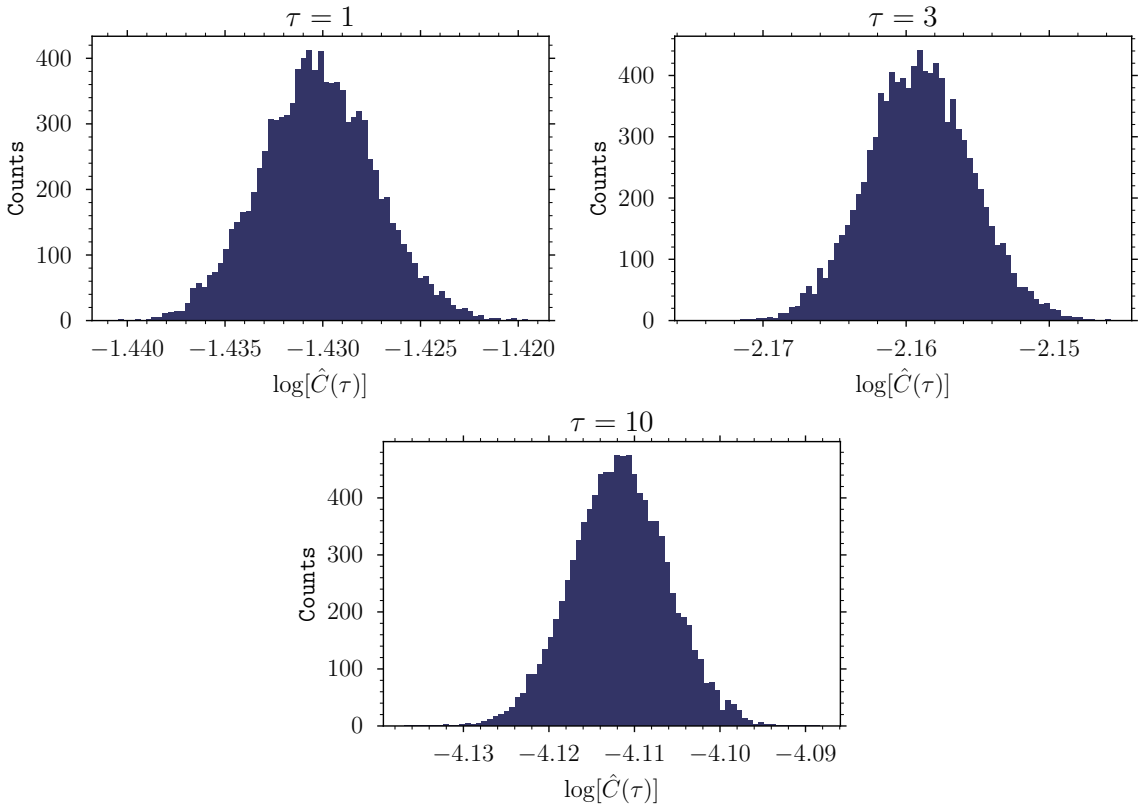


Figure 3.4: Empirical distribution of the correlation function central value, $\hat{C}(\tau)$, at three randomly selected Euclidean times. The data corresponds to the *ss* axial vector correlation function computed at $T = 47$ MeV. The empirical distribution is estimated using bootstrap [13]. All three Euclidean times displayed are randomly selected.

As the number of measured ensembles is large enough, and the samples are assumed independent and identically distributed at fixed τ, we can apply the central limit theorem to $\hat{C}(\tau)$, which implies that the sample average is expected to be normally distributed at each independent τ. Figure (3.4) shows the empirical distribution of $\hat{C}(\tau)$ at three randomly selected Euclidean times for the (strange-strange) *ss* axial vector mesonic correlation function computed at $T = 47$ MeV. As $\hat{C}(\tau)$ is normally distributed, then $u(\tau)$ is also normally distributed with mean $\vec{0}$ and covariance Σ, equal to the covariance $\hat{C}(\tau)$. The total likelihood of the data can be

modelled using a correlated multivariate normal distribution:

$$P(u|\theta) = \mathcal{N}(\mu = \vec{0}, \Sigma) = \mathcal{N}(\hat{C}(\tau) - F_e(\tau;\theta), \Sigma). \tag{3.18}$$

In the equation above, $\mu = \hat{C}(\tau) - F_e(\tau;\theta) = \vec{0}$ has dimension N_τ, and Σ has dimensions $N_\tau \times N_\tau$.

The multivariate normal likelihood distribution is well-known, and maximising it is equivalent to minimising the following target function:

$$\mathcal{L}(\theta) = \sum_{\tau,\tau'=0}^{N_\tau} \big[\hat{C}(\tau) - F_e(\tau,\theta)\big] \Sigma^{-1}_{\tau,\tau'} \big[\hat{C}(\tau') - F_e(\tau',\theta)\big]. \tag{3.19}$$

Equation (3.19) corresponds to minus the logarithm of the likelihood distribution function. In this context, the maximum likelihood estimate of the model parameters are the parameters that minimise eq. (3.19):

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}}\, \mathcal{L}(\theta). \tag{3.20}$$

Due to the inherent complexity of the parameter estimation, the minimisation of the target function in eq. (3.20) must be performed numerically.

Once the maximum likelihood parameters are estimated, their uncertainties can be approximated using the Fisher information [14]. In our particular case, we can approximate the covariance of the estimated parameters $\hat{\theta}$ using

$$\mathrm{Cov}(\theta_a, \theta_b) = \sum_{\tau,\tau'=0}^{N_\tau} J(\theta_a, \tau)\, \Sigma_{\tau,\tau'}\, J(\theta_b, \tau'), \tag{3.21}$$

where J represents the Jacobian of $\hat{C}(\tau)$ with respect to θ, which can be computed using the first derivatives of eq (3.7):

$$J(\theta_a, \tau) = \frac{\partial \hat{C}(\tau)}{\partial \theta_a} = \frac{\partial F_e(\tau,\theta)}{\partial \theta_a}. \tag{3.22}$$

In order to improve the stability of the regression, we fold the correlation function estimates around the midpoint in the temporal direction:

$$\hat{C}_F(\tau) = \frac{\hat{C}(\tau) + \hat{C}(N_\tau - \tau)}{2}. \tag{3.23}$$

This can be viewed as a data augmentation procedure, justified by the symmetry of the mesonic propagators under Euclidean time inversions.

In addition, all correlation functions samples $C_b(\tau)$ are divided by $G = \hat{C}(\tau = N_\tau/2)$, which ensures that $A_0 \simeq 1$ in eq. (3.7). The normalisation constant is treated as a real number, which avoids the computation of complicated ratios of random variables.

3.1.2 Initial parameters estimation

Correlated fits tend to be unstable due to the sometimes sparse nature of the covariance matrix in eq. (3.19). This implies that the estimated parameters can largely depend on the initial values provided to the minimisation routine. As a result, minimising the target function starting from initial parameters $\tilde{\theta}$ close to the true population parameters θ is key to obtaining reliable estimates. However, with almost no prior information about θ, the initial parameters must be estimated from scratch using the information at our disposal. To obtain a reliable initial estimate $\tilde{\theta}$, we propose an algorithm based on the one described in Ref. [12].

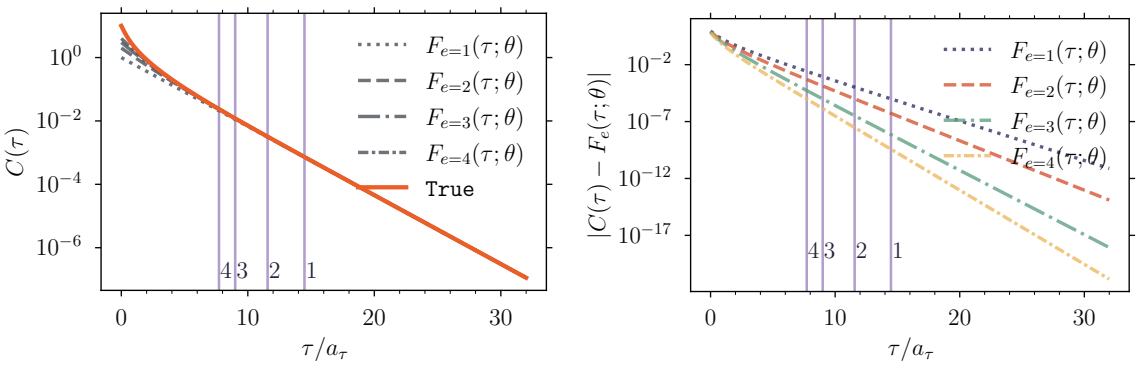
To avoid any confusion, we stress that $\tilde{\theta}$ represents the set of all parameters in a model composed by N_e states:

$$\tilde{\theta} = \{\theta_s = (A_s, M_s) \mid s \in [0, N_e - 1]\} \tag{3.24}$$

The algorithm employed to estimate the initial parameters of the regression needs two starting hyperparameters: the maximum number of states included in the model, N_e, and the initial Euclidean time at which the correlation function is expected to be well described by those N_e states; we refer to this initial time with the label $\tilde{\tau}_0(N_e)$. As stated before, we fix the maximum number of states in our model to be $N_e = 4$. In our analysis, we vary $\tilde{\tau}_0(N_e)$ from 2 to 5 depending on the temperature and the properties of the correlation function: for instance, high temperature correlation functions only use $\tilde{\tau}_0(N_e) = 2$, 3 as initial times. It is worth noting that $\tilde{\tau}_0(N_e)$ also defines the largest fit window used in the regression:

$$FW[\tilde{\tau}_0(N_e)] = [\tilde{\tau}_0(N_e), \tau_f]. \tag{3.25}$$

As explained before, the correlation function might show a plateau at $\tau \to N_\tau/2$. This is just a consequence of the exponential nature of the contribution of each state: lighter states survive at large τ, whereas heavier states get heavily suppressed as τ increases. As a consequence, there are regions in τ where different sub-models in $\mathcal{M}[N_e]$ can reliably describe the data. For example, close to $N_\tau/2$, one-state or two-state models are expected to model the data with high accuracy. A visual example of this behaviour is presented in Figure (3.5).



(a) Correlation function **(b)** Absolute difference

Figure 3.5: Example of an artificial correlation function and the ability of some models $F_e(\tau)$ to represent the correct data. The figure on the left shows the population correlation function, $C(\tau)$, and some approximate models $F_e(\tau)$. The figure on the right shows the absolute difference between $C(\tau)$ and the simplified models. The vertical lines are located at the first Euclidean time $\tau_0(e)$ holding $|C(\tau) - F_e(\tau)| \leq 1 \cdot 10^{-5}$.

Our goal is then to build an algorithm that automatically selects the approximate Euclidean time at which the correlation function can be properly described by a submodel of F_{N_e}. Using those initial times, we can iteratively estimate the parameters of each state in the largest model: $F_{N_e}(\tau;\theta)$. In order to select the approximate initial Euclidean times, which we label $\tilde{\tau}_0(e)$, we use the following recursive formula:

$$\tilde{\tau}_0(e) = \begin{cases} \tilde{\tau}_0(e = N_s) & \text{if } e = N_s \\ \frac{1}{\omega_1+\omega_2}\left[\omega_1\tau_0(e+1) + \omega_2 N_\tau/2\right] & \text{if } e < N_s. \end{cases} \tag{3.26}$$

For instance, we expect the fit window $FW[\tilde{\tau}_0(e=1)]$ to contain the Euclidean times at which the correlation function can be well described by only one state. The values of ω_1 and ω_2 in eq. (3.26) also depend on the number of states in the model. Their definitions, taken from Ref. [12], are shown in Table (3.1). Small variations of the values of ω_1 and ω_2 had no impact in the final results.

Table 3.1: Definitions of ω_1 and ω_2 as a function of the number of states e.

	$e = 1$	$e = 2$	$e > 2$
$\omega_1(e)$	2	4	6
$\omega_2(e)$	1	1	1

Once the initial times are generated, we proceed with the estimation of the initial parameters. To do so, we build an algorithm based on two key ingredients: the effective mass of a correlation function, and fits to models with different numbers of states at different fit windows.

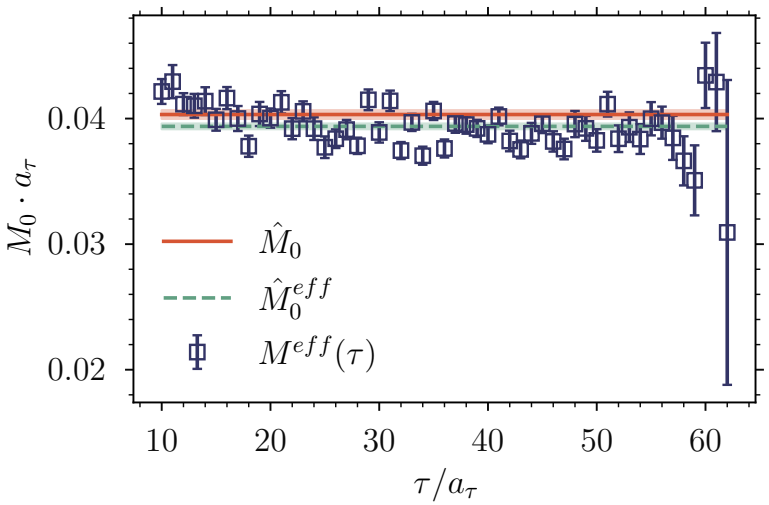


Figure 3.6: Effective mass of the lowest temperature lattice estimate of the lightest pseudoscalar correlation function. The results are expressed in lattice units. Two estimates of the ground state mass are present in the figure: $\hat{M}_0$, computed using our methodology; and $\hat{M}_0^{eff}$, corresponding to the median of $M^{eff}(\tau)$.

The effective mass M^{eff} of a correlation function at a fixed τ is defined as the mass of the ground state of a correlation function assuming that only one state contributes to the correlation function; the effective mass represents the combined contribution of all states in the correlation function. In the hadronic phase, and due to the fact that higher-order states are exponentially suppressed as τ increases, the effective mass should be similar to the actual ground state mass at Euclidean times close to the middle of the lattice: $\tau \to N_\tau/2$. Figure (3.6) contains an example of a real effective mass calculation. The effective mass can be extracted by solving the

following transcendental equation:

$$\frac{\hat{C}(\tau)}{\hat{C}(\tau + a_\tau)} = \frac{\cosh\left(M^{eff}\left(\tau - \frac{N_\tau}{2}\right)\right)}{\cosh\left(M^{eff}\left(\tau + a_\tau - \frac{N_\tau}{2}\right)\right)}. \tag{3.27}$$

Once the effective mass is defined, we can iteratively estimate the masses of all N_e states in ascending order: from the ground state, $s = 0$, to the heaviest state in the model, $s = N_e - 1$. To do so, we employ the effective mass as the starting guess for the mass of each state. In order to compute an initial estimate of a higher-order state mass using the effective mass, we subtract the contribution of the previously estimated lighter states to the correlation function.

$$\hat{X}_e(\tau) = \hat{C}(\tau) - F_{e-1}(\tau, \theta = \tilde{\theta}), \tag{3.28}$$

where F_e corresponds to a truncation of eq. (3.5) evaluated at the already estimated lower order parameters. In order to make eq. (3.28) valid for all states, the following convention must be adopted:

$$F_0(\tau; \theta) = 0, \tag{3.29}$$

which implies that $\hat{X}_1(\tau) = \hat{C}(\tau)$.

The newly generated subtracted correlation function, $\hat{X}_e(\tau)$, should contain the $s = e-1$ state in $\hat{C}(\tau)$ as its lowest energy state. As a result, eq (3.27) can be applied to $\hat{X}_e(\tau)$ at $\tau \to N_\tau/2$ in order to estimate the mass of this state. For example, in the case in which we already have access to an estimate of the ground-state mass and amplitude, then, the second-order subtracted correlation function, $\hat{X}_2(\tau)$, is defined as

$$\hat{X}_{e=2}(\tau) = \hat{C}(\tau) - F_{e=1}(\tau; \theta = \tilde{\theta}). \tag{3.30}$$

As long as the model accurately describes the data, and the previously estimated parameters are correctly estimated, the ground state of $\hat{X}_2(\tau)$ is expected to be the first excited state of $\hat{C}(\tau)$. As uncertainties propagate, extracting the effective mass of higher-order subtracted correlation functions $\hat{X}_{e\gg1}$ can be difficult and unreliable. Consequently, we use $1.5\,\tilde{M}_{s-1}$ as the initial guess for the mass for all states with $s \geq 2$.

After the initial masses are estimated using the effective mass, we employ a sequence of different fits to $\hat{X}_e$ and $\hat{C}(\tau)$ to improve our estimates of the initial parameters. In this process, the amplitudes of each state are also estimated, starting with a flat initial guess of $\tilde{A}_s = 1$ for all s. The minimisation routine used to estimate $\tilde{\theta}$ uses eq. (3.19) as the target function to minimise, i.e. it takes correlations in the data into account. The complete algorithm employed to estimate the initial values of the parameters is described in detail in Algorithm (1).

Once the initial parameters of the regression are estimated through the application of Algorithm (1), we need to generate our final estimate of the ground state mass. To do so, we proceed by extracting an estimate of the ground state mass for each available fit window smaller or equal to $FW[\tilde{\tau}_0(N_e)]$, that is, our largest fit window. The fit windows included in the analysis can be built by iteratively shrinking $\tilde{\tau}(N_e)$ by one lattice spacing.

Algorithm 1: Algorithm designed to estimate the initial parameters of all states in the largest model considered in the analysis: $F_{N_e}(\tau;\theta)$. All the operations can be performed in-place over a hash-map-like data structure: $\tilde{\theta}$. The notation $\tilde{\theta}[X]$ implies that we are accessing the element X from $\tilde{\theta}$.

Data: $\hat{C}(\tau)$, N_e, $\tilde{\tau}_0(N_e)$. `Optional:` $\tilde{M}_0^g$.
Result: Initial estimates of the parameters: $\tilde{\theta}$
Initialise: $\tilde{\theta} \leftarrow \{(\tilde{A}_s = 1, \tilde{M}_s = \tilde{M}_0^g \texttt{ or } 1)\,\forall\, s\}$
for $e \leftarrow 1; e \leq N_e; e++$ **do**
  `# Set variables for this sub-model:` $F_e(\tau;\theta)$
    $s \leftarrow e - 1$
    $\tau_0(e) \leftarrow$ Compute using eq. (3.26)
    $\hat{X}_e \leftarrow$ Compute using eq. (3.28)
  `# Update model's parameters iteratively`
    $\tilde{\theta}[M_s] \leftarrow$ if $e \leq 2$ then M_s^{eff} using $\hat{X}_e$ else $1.5\,\hat{M}_{s-1}$
    $\tilde{\theta}[M_s]$, $\tilde{\theta}[A_s] \leftarrow$ fit $F_1(\tau;\theta)$ on $\hat{X}_e(\tau)$ over $FW[\tilde{\tau}_0(e=1)]$
    if $e > 1$ **then**
      $\tilde{\theta}[M_s]$, $\tilde{\theta}[A_s] \leftarrow$ fit $F_e(\tau;\theta)$ on $\hat{C}(\tau)$ over $FW[\tilde{\tau}_0(e)]$ keeping fixed $\{\tilde{A}_{s'}\,\tilde{M}_{s'} \,|\, s' < s\}$
      $\tilde{\theta}[M_s]$, $\tilde{\theta}[A_s] \leftarrow$ fit $F_e(\tau;\theta)$ on $\hat{C}(\tau)$ over $FW[\tilde{\tau}_0(e)]$
  end
end
return $\tilde{\theta} = \{\tilde{\theta}_s \,|\, s \in [0, N_e - 1]\}$

In order to reduce any possible bias included by insisting that only a certain number of states e contribute in a particular fit window, we perform a multimodel analysis. The models employed correspond to a subset of all models in $\mathcal{M}[N_e]$. For each model included in a fit window, an estimate of the ground state mass is computed. We systematically compare all estimated ground state masses to generate our best final estimate within the current fit window in terms of some predefined metrics. Finally, once we have generated the best estimate of the ground state mass for all fit windows considered, we produce a final estimate of the ground state mass

independent of the fit window.

3.1.3 Regression at fixed fit window: $FW[\tau_0]$

In this subsection, we discuss how we compute our estimate of the ground state mass on a particular fit window $FW[\tau_0]$, where $\tilde{\tau}_0(N_e) \leq \tau_0 \leq N_\tau/2$. To produce the ground state mass at the current fit window, we propose fitting the data to a subset of all models contained in $\mathcal{M}[N_e]$. We do not include all models in all fit windows because simple models are not realistic at $\tau \to 0$, due to the fact that the contribution of excited states cannot be ignored in this region; and also due to the fact that complex models contain numerous free-parameters, which complicates their analysis as the number of points included in the fit window decreases, that is, for $FW[\tau_0 \to N_\tau/2]$. As a result, a model with e states is only included at a particular fit window if the following two conditions are met: first, the starting point of the fit window must satisfy

$$\tau_0 \geq \tilde{\tau}_0(e),$$

where $\tilde{\tau}_0(e)$ is computed using eq. (3.26); second, the number of free parameters in the model, N_θ, is smaller than the number of points included in the fit window:

$$N_\theta = 2\,e < |FW[\tau_0]| = |\tau_f - \tau|.$$

Having selected a collection of models that are included in a particular fit window, we proceed by fitting all those models to the same correlation function data. To increase the accuracy in the regression, and to avoid possible instabilities, we fit each model twice: first using the initial parameters extracted from the application of Algorithm (1), and then employing the final parameters obtained from the previous fit window as the initial parameter in this fit window; the second fit is only available from the second-largest window: $FW[\tilde{\tau}_0(N_e) + 1]$. As a result of our analysis, for each fit window, we produce $2\,N_m$ estimates of the ground state mass, where N_m is the number of models included in the regression; in principle, N_m can be different at each fit window. We label the resulting ground state masses using $\hat{M}_0^F(FW[\tau_0])$, where F denotes a particular model.

Although the states in the models are hierarchically ordered in mass, the resulting fitted states might not be. For instance, in a 3-state model, the third state might represent the ground state. As a consequence, we cannot blindly select the first state in the model as the ground state. In order to systematically order the states and correctly select the ground state, we apply the following sorting algorithm, similar to the one employed in Ref. [12]: we swap the order of two consecutive states in a

given model in the case in which the second state has a smaller amplitude than the first one; in the case in which the second state has a smaller mass than the first one; or if the first state mass is unreasonably small or large compared to the scale of the simulation, represented by a_τ^{-1}.

After producing the estimate of the ground state mass for all available models at the current fit window, we compare them to compute the best possible estimate of the ground state mass. The models are compared in terms of their predictability, that is, how well each model is able to explain the measured data; this is equivalent to measuring the likelihood of the model. The metric used to compare all models is the so-called corrected Akaike information criterion (AICc) [15–17].

The AICc has its roots in information theory, and measures the expected divergence between a model and an unknown ground truth model; the divergence is measured using the Kullback-Leibler (KL) divergence [18]. In our definition of the AICc, the model with the lowest AICc among all is the most likely to correctly describe the data:

$$\mathrm{AICc}(F_e) = N_\theta - \log(\hat{\mathcal{L}}) + \frac{N_\theta^2 + N_\theta}{|FW[\tau_0]| - N_\theta - 1}, \tag{3.31}$$

where N_θ corresponds to the number of free-parameters of the model; $\hat{\mathcal{L}}$ is the objective function defined in eq (3.19), and evaluated at the maximum likelihood parameter estimates: $\hat{\theta}$; and $|FW[\tau_0]|$ represents the number of points included in the current fit window. The AICc depends on the model through N_θ and $\hat{L}$.

From the AICc, we can compute the relative likelihood between two models F and F'

$$l(F', F) = \exp\Big(-\frac{1}{2}\big[\,\mathrm{AICc}(F') - \mathrm{AICc}(F)\big]\Big). \tag{3.32}$$

This quantity measures how likely F' is to correctly model the data when compared to F. Note that l does not include global information. As a result, there exists the possibility that models not included in the analysis might be better at describing the measured data at the current fit window.

The model with the lowest AICc among all available models represents the model with the highest likelihood, we refer to this model with the label F_b. From this model, we compute the relative model quality of all models in the collection through eq. (3.32), fixing F to be F_b. The relative model quality can be employed to obtain a final estimate of the ground state mass in the current fit window by taking into account all masses in a weighted average, where the weights correspond to $l(F_i, F_b)$, computed using eq. (3.32). This technique allows us to promote the influence of higher quality models in the final result while avoiding manually discarding any models; the contribution of low quality models are exponentially suppressed. We label

the estimate of the ground state mass at the current fit window using $\hat{M}_0(FW[\tau_0])$. The uncertainty of this estimate can be approximated using bootstrap [13].

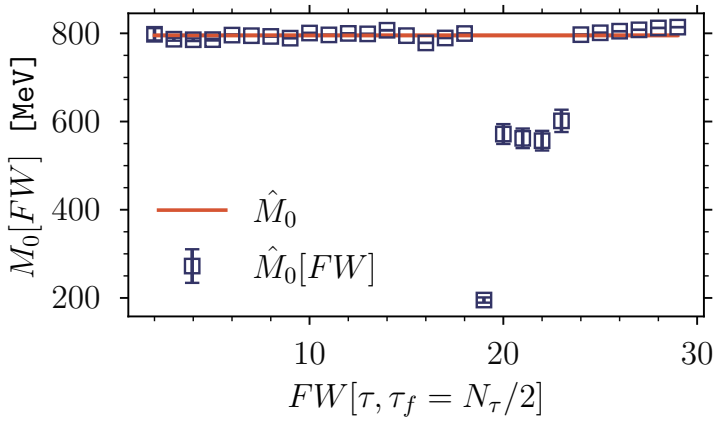


Figure 3.7: Estimated ground state mass as a function of the fit window $\hat{M}_0[FW]$ in the lightest vector channel meson, whose lowest energy mode corresponds to the $\rho(770)$ physical state. The data is computed at the second-lowest temperature available ($T = 97$ MeV), corresponding to a lattice of $N_\tau = 64$ points in the temporal direction. The horizontal line corresponds to the fit-window independent final estimate of the ground-state mass: $\hat{M}_0$.

3.1.4 Extracting the final estimate of the ground state mass $\hat{M}_0$

As a result of our analysis, for each available fit window, we are able to produce an estimate of the ground state mass, $\hat{M}_0(FW[\tau_0])$. The goal is then to compute a final estimate of M_0 independent of the fit window. This quantity, labelled $\hat{M}_0$ is computed using the median of all $\hat{M}_0(FW[\tau_0])$ available. Although the median is indeed a robust statistic, we decided to discard the outliers in the sample, as they can impact the final estimate of the mass; outliers can be identified using standard techniques, such as the Interquartile Range (IQR) interval. The uncertainty of the median statistic can be approximated using bootstrap [13]. A real example of the results obtained with our analysis can be found in Figure (3.7). The artefacts spotted in the figure at $19 \leq \tau \leq 23$ are sometimes encountered at fit windows where higher order models are difficult to regress (large number of degrees of freedom), and lower order models are not realistic enough (contribution of excited states). Their presence demonstrates why a variational analysis is required when performing a spectroscopy

analysis over lattice correlation functions.

3.1.5 The effect of varying τ_f on the ground state mass

As stated before, all fit windows included in the analysis are constructed from an initial time, τ_0, and a final time, τ_f. In the discussion of the methodology, we stressed that τ_f was always kept fixed to $N_\tau/2$ as a way of simplifying the notation. As a consequence, all fit windows employed in the analysis can be generated by just varying τ_0 while keeping τ_f constant to its maximum value: $\tau_f = N_\tau/2$. This is generally desired, as keeping τ_f to its largest value maximises the amount of information included in the regression. However, for some correlation functions, it may be desirable to avoid including the last few Euclidean times in the analysis as they can be highly noisy. For example, this is the case for some axial-vector correlation functions at low temperature; see Figure (3.8). Note that varying τ_f in the methodology presented before is trivial, as the only modification is the number of fit windows available in the fit, defined in eq. (3.11).

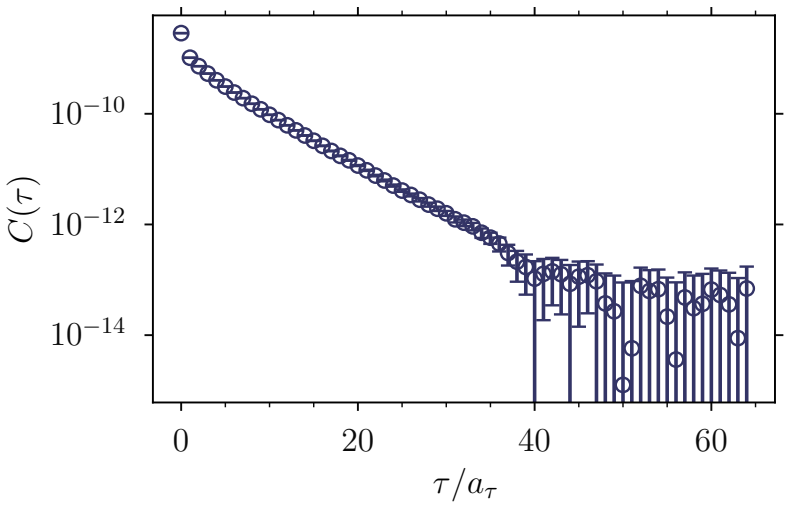


Figure 3.8: Lattice estimate of the lightest axial-vector mesonic correlation function. The correlation function represents the $a_1(1260)$ physical state.

In our particular case, we compute τ_f using a proportion of the maximum number of Euclidean times that can be included in the analysis, that is, $N_\tau/2$. We refer to the proportion with the label p, whose definition is

$$\tau_f = p\,\frac{N_\tau}{2}. \tag{3.33}$$

By selecting different τ_f, we can exclude noisy points in the analysis. However, one of our premises is to generate a methodology that avoids manually selecting any results. In order to remain honest to our promise, we use the following procedure to compute a systematic τ_f-independent ground state mass: first, we independently apply the previously discussed methodology using different fixed values of τ_f, which generates a set of ground state estimates depending on τ_f, $\hat{M}_0[FW;\tau_f]$; then, we combine all the estimates into a single sample; to conclude, we compute the median estimate of the combined sample after removing the outliers outside the IQR interval. No fit window is included more than once in the sample.

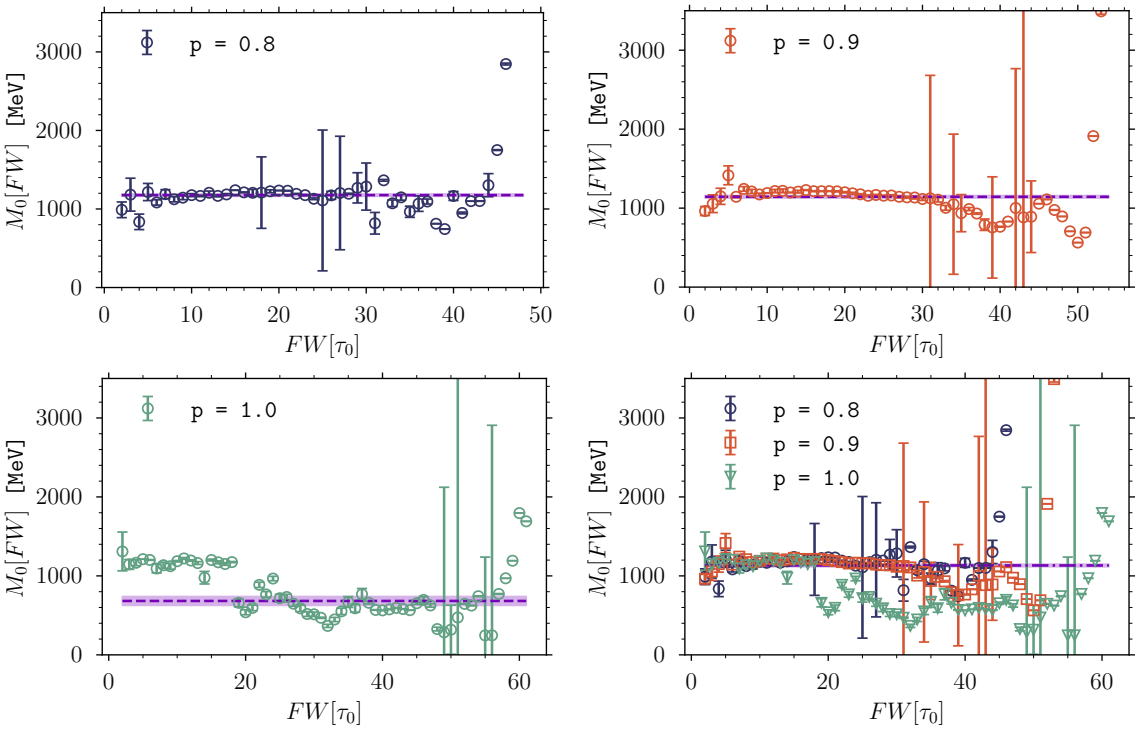


Figure 3.9: Analysis performed over the $a_1(1260)$ correlation function shown in Figure (3.8) using different values of τ_f. The different values of τ_f are constructed using $p = 0.8$, 0.9 and 1.0 in eq. (3.33).

The median of the combined sample represents the best estimate of the ground state mass that we can produce. This procedure can be applied as we are always extracting an estimate of the same underlying population ground state mass, M_0, independently of τ_f; the spectral function of C is independent of τ, and therefore, the same model is valid at all Euclidean times.

Figure (3.9) contains an experiment in which τ_f is varied to compute the mass of the $a_1(1260)$ physical state — the ground state of the lightest axial vector correlation function shown in Figure (3.8). The experiment demonstrates that varying τ_f leads to an improvement in the estimation of the ground state mass, as the extracted mass converges to the expected 1260 MeV of the $a_1(1260)$ state.

In our particular analysis, we vary τ_f using $p = 0.8$, 0.9 and 1.0 according to eq. (3.33). We do not shrink p more than 0.8 to enable a systematic analysis throughout all temperatures. As we employ a fixed-scale approach in our lattice simulations, the higher the temperature, the smaller the extent of the lattice in the temporal direction. As a result, shrinking p to values lower than 0.8 implies that we fit high temperature correlation function to less than 5 points.

3.1.6 Comments on uncertainties

The final outcome of our analysis on mesonic correlation functions is an estimate of the population ground state mass M_0, which we label $\hat{M}_0$. We can treat $\hat{M}_0$ as a random variable with expectation value $\mu_M = \text{median}\,\hat{M}_0[FW;\tau_f]$ and standard deviation σ_M equal to the standard error of the median, which can only be approximated using bootstrap [13]. $\hat{M}_0$ is expressed in lattice units.

We would like to express our ground state mass estimate in physical units, as it allows a direct comparison between theory, experiments and other theoretical predictions coming from non-lattice models. To do so, we require the inverse of the temporal lattice spacing a_τ^{-1}, which allows us to compute the following ground state mass:

$$\hat{M}_0^{\text{ph}} = \hat{M}_0\, a_\tau^{-1}. \tag{3.34}$$

This quantity is expressed in physical units, as the inverse of the lattice spacing has units of energy/space. As detailed in Appendix (A), our lattice spacing in the temporal direction has a value of $a_\tau^{-1} = 6079 \pm 13$ MeV. We treat a_τ^{-1} as a random variable of expected value $\mu_a = 6079$ MeV and standard deviation $\sigma_a = 13$ MeV.

Equation (3.34) implies that $\hat{M}_0^{\text{ph}}$ is a product of two random variables, $\hat{M}_0$ and a_τ^{-1}. We assume that both random variables are independent, which implies that the expectation value of $\hat{M}_0^{\text{ph}}$ is just

$$\mu_M^{\text{ph}} = \mu_M\, \mu_a. \tag{3.35}$$

In addition, the uncertainty of $\hat{M}_0^{\text{ph}}$, measured in terms of its standard deviation, can be calculated using the product of two independent random variables:

$$\sigma_M^{\text{ph}} = \sqrt{(\mu_M^2 + \sigma_M^2)\,(\mu_a^2 + \sigma_a^2) - (\mu_M\, \mu_a)^2}. \tag{3.36}$$

This formula can be approximated using

$$\sigma_M^{\text{ph}} \simeq \mu_M \, \mu_a \sqrt{\left(\frac{\sigma_M}{\mu_M}\right)^2 + \left(\frac{\sigma_a}{\mu_a}\right)^2}. \tag{3.37}$$

From now on, if a mass is expressed in physical units, then its central value is computed using eq. (3.35), and its uncertainty is approximated using eq. (3.36).

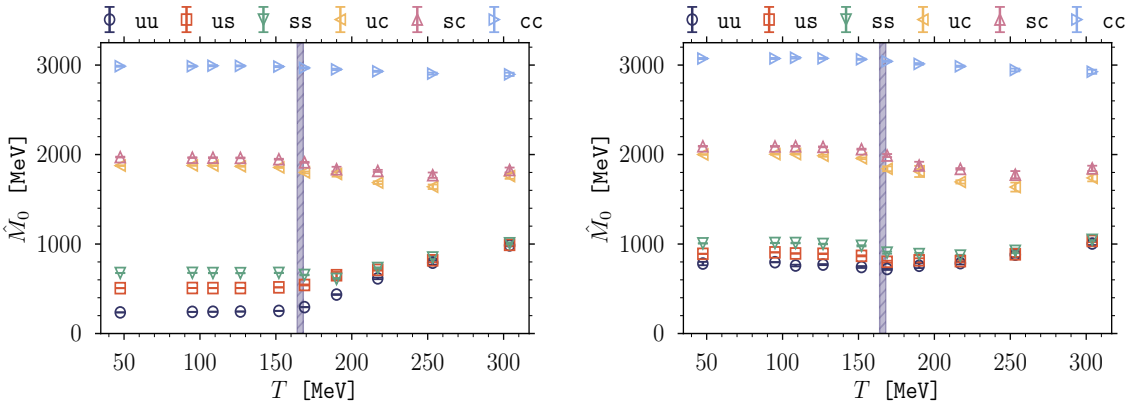
3.2 Results

In this last section, we present some results extracted from the application of the previously discussed methodology over mesonic correlation functions computed at different temperatures, flavour combinations and channels. Additionally, for each specific combination of quark content, channel and temperature, we have access to two correlation functions: one in which the sources employed in the computation of the quark propagator are local, which we denote with ll, and the other one using Gaussian smeared sources, which we denote with ss. Unless unclear, we use the same label to refer to smeared sources estimates, and the mesonic correlation function with flavour content strange-strange. More information about our lattice setup can be found in Appendix (A). In principle, the analysis on both estimates should yield similar masses in the low temperature regime, although smeared estimates are expected to be more accurate. As a result, we mainly report smeared estimates for the temperature dependence of the meson masses. However, we study the differences between both estimates in some particular cases. Information about the mesonic correlation functions available can be found in Appendix (B).

This section is divided into three parts. In the first one, we outline general trends encountered in the temperature dependence of mesonic ground state masses obtained through the application of the previously discussed methodology over lattice correlation functions. In this first part, we also discuss possible limits on the validity of the methodology as a function of the temperature. The second part mainly focuses on the restoration of the $SU(2)_A$ chiral symmetry through the degeneracy of the $\rho(770)$ and $a_1(1260)$ states as a function of the temperature. Additionally, we perform some fit-independent analysis over the $SU(2)_A$-related correlation functions to detect possible degeneracies. To conclude, the third and last part discusses in detail the $D_{(s)}$ mesonic sector throughout the hadronic phase, that is, states with uc and sc flavour content at $T < T_c$.

3.2.1 General trends in the ground state masses

The first set of results presented is the temperature dependence of the ground state mass of the pseudoscalar and vector channels; the pseudoscalar channel has quantum numbers $J^{PC} = 0^{-+}$, while the vector channel has quantum numbers $J^{PC} = 1^{--}$ — see Appendix (B). Due to the fact that in our simulations both light quarks are assumed degenerate, we only have access to 6 flavour combinations. As a result, for each channel, we plot the ground state masses for all 6 flavour combinations available, which are: uu, us, uc, ss, sc and cc[1]. In our analysis, we vary τ_f to diminish the possible impact of noise in the tail of the correlation function. The results of the analysis can be found in Figure (3.10).



(a) Pseudoscalar channel

(b) Vector channel

Figure 3.10: Temperature dependence of the ground state mass of all 6 flavour combinations available in the pseudoscalar and vector channels. The vertical line shows the pseudocritical temperature of the system: $T_c = 166 \pm 2$ MeV. The results are generated from correlation functions estimated from smeared sources.

The temperature dependence of the ground state masses shows two different trends: one at low temperature, $T < T_c$, and another present at high temperature, $T > T_c$. In the low temperature regime, where the QCD system is believed to be in its hadronic phase, minimal temperature dependence is encountered in the mass estimates: in the hadronic phase, eq. (3.5) is expected to accurately model the contribution of each state to the correlation function, as QCD is a confining theory. In this regime, the masses of the low-energy states might be affected by the

[1]The notation **sc** refers to a mesonic correlation function with quark content s and c.

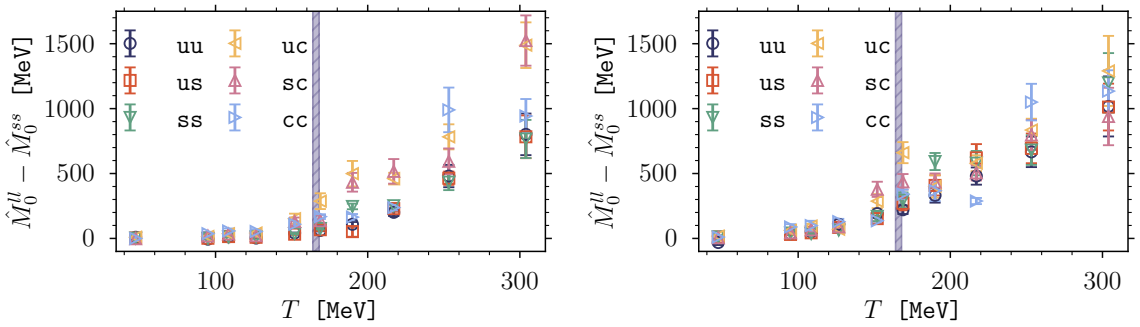
temperature, but the overall functional form of the spectral function should remain unaltered. The validity of eq. (3.5) as a model of the contribution of each state to the correlation function becomes questionable as the temperature of the system approaches the pseudocritical temperature. Once above T_c, the extracted quantities depend on the temperature. At these temperatures, the masses of the light mesons increase with T until becoming degenerate. Although the temperature dependence of charmed quarks above T_c is smaller, as the inherent energy scale of the charm quark is larger, the results show a slight temperature dependence on the D-meson states, uc and sc.

The abrupt increase in mass and the degeneracy of light mesons might be a consequence of the dominant collective excitations induced by the temperature: as the temperature increases, collective excitations and screening are expected to impact the mesonic spectrum. Due to the fact that the inherent light quark energy scales are smaller than the scale set by the temperature, the spectrum of those correlation functions is expected to be dominated by collective excitations, which lead to the degeneracy spotted at $T > T_c$ in the uu, us and ss mesons. It is important to stress that, due to the presence of thermal effects, one should interpret with caution the light mesons quantities extracted at $T > T_c$ as masses. If the system loses its confining properties, then light quarks might not create bound mesonic states, which implies that the corresponding spectral functions might not contain a distinguishable lowest energy mode. At high temperature, the concept of ground state is ill-defined. This behaviour questions the validity of eq. (3.5) at high T, at least in the light quark sectors. Although charmed states might survive in the QGP due to their inherent energy scale, one should also be cautious when interpreting results extracted using eq. (3.5) as masses at $T \gg T_c$.

Furthermore, as T increases, the uncertainty in our estimates also grows. The origin of this behaviour is two-fold. First, as we employ a fixed-scale approach in our ensemble generation, then the higher the temperature, the lower the number of Euclidean times at which the target correlation function is measured. Consequently, the number of degrees of freedom included in the regression increases with the temperature, thus making the estimation of the ground state mass at high temperature difficult. This problem is combined with the fact that the model used to fit the data at high temperature is not completely correct, which also impacts the uncertainty in the estimates; in principle, at high T, we are forcing the data to fit an invalid model.

Figure (3.11) contains the difference between the ground state mass extracted from local correlation functions and a smeared ones. It supports the idea that eq. (3.5) becomes invalid as the temperature of the system increases. In the low temperature regime, as expected, Figure (3.11) shows that both sources yield similar

ground state estimates. However, once the temperature of the system increases, local and smeared estimates diverge. This can be a consequence of the fact that smeared sources are specifically built to suppress the contributions from higher order states to the correlation function. Once the correlation function describes a plasma where collective excitations dominate, and no clear ground state is definable, the two types of sources produce completely different results. At high temperature, where the spectrum might be continuous due to thermal effects, smeared sources might impact the estimation of the correlation function unexpectedly. More research exploring the possible consequences of employing smeared sources in the computation of the quark propagator at high temperature is required.



(a) Pseudoscalar channel **(b)** Vector channel

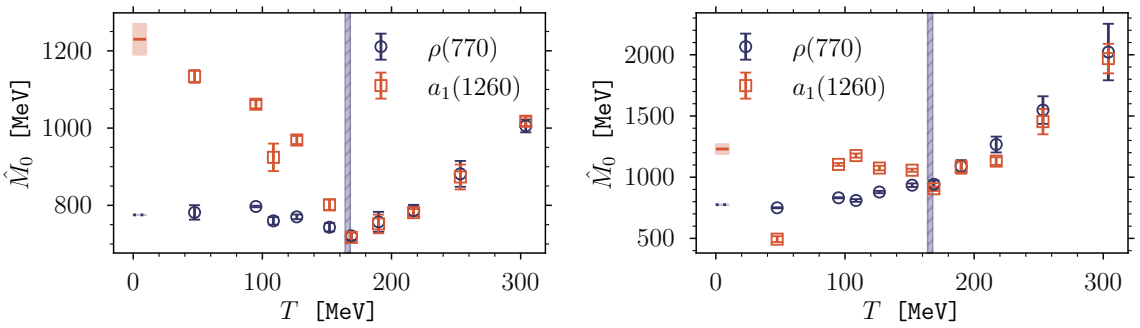
Figure 3.11: Difference between local, M_0^{ll}, and smeared, M_0^{ss}, estimates of the ground-state mass for the pseudoscalar and vector channels as a function of the temperature. The vertical line shows the pseudocritical temperature of the system: T_c.

The results displayed in Figure (3.11) demonstrate that local and smeared estimates start diverging around the pseudocritical temperature for both channels. A finer resolution at temperatures close to the critical temperature could allow us to discern the exact point at which both estimates diverge.

We would like to stress that, although the model in eq. (3.5) is not valid at high temperature, and, therefore, the quantities extracted at high T cannot be interpreted as masses, the results extracted still contain meaningful information about the system. For instance, the clear degeneracy in the light quark sector spotted in both plots contained in Figure (3.10) suggests that thermal effects dominate this sector.

3.2.2 Restoration of $SU(2)_A$ chiral symmetry

The restoration of chiral symmetry at finite temperature in QCD has been studied for decades, both in the mesonic sector [19], and the baryonic sector [9–11]. Mesonic studies are based on the analysis of the pion and scalar meson susceptibilities [20], while baryonic analysis focus on the temperature dependence of screening masses [21, 22], and the evolution of parity partners with the temperature [9–11]. In our particular case, we study chiral symmetry restoration by analysing the degeneracy of the lowest energy modes of the lightest vector and axial vector correlation functions, which respectively correspond to the $\rho(770)$ and the $a_1(1260)$ physical states. These two states are mixed under an $SU(2)_A$ chiral rotation, which implies that in the case in which the symmetry is restored, then both states should become degenerate; see Appendix (B) for more information. Equivalent analysis coming from chiral effective theories are also available in the literature; see Refs. [23–25] and references therein.



(a) Smeared sources **(b)** Local sources

Figure 3.12: Smeared and local ground state mass of the $SU(2)_A$-related states: $\rho(700)$, and $a_1(1260)$. The vertical line shows the pseudocritical temperature of the system: T_c. The zero-temperature estimates correspond to the latest results reported in Ref. [26].

Our results suggest that a degeneracy between both states is indeed present at $T > T_c$; the results are presented in Figure (3.12) for both local and smeared sources. The results display a clear degeneracy between both $SU(2)_A$-related states for both type of sources. With our limited temperature resolution, the temperature at which the symmetry is restored is close to T_c. Nevertheless, a higher temperature resolution around the pseudocritical temperature is required in order to find the exact temperature of the transition.

We would like to stress that our first-principle results on the degeneracy of the

$SU(2)_A$ chiral symmetry show a similar trend to several results extracted from effective theory calculations [23–25]. All effective theory calculations agree in a non-negligible temperature dependence in the $a_1(1260)$ at relatively low temperature. Additionally, all effective theories produce a downwards shift of the $a_1(1260)$ mass below T_c, and a small upwards shift of the $\rho(770)$ mass; it is at $T \simeq T_c$, where the states meet.

In addition, as expected from Figure (3.11), local and smeared estimates lead to completely different results above the pseudocritical temperature.

It is worth discussing the lowest temperature estimate of the $a_1(1260)$ mass obtained from a local correlation function. The data shows a clear problem in that particular estimate. This is possibly caused by the large level of noise encountered in the lowest temperature axial correlation function around the middle of the lattice. While testing the methodology, we found that excluding the tail of the correlation function from the fit led to a ground state mass closer to the expected 1260 MeV. However, if we want to keep our analysis independent of the temperature, we should also remove the same proportion of points at higher temperatures, which implies that we increase the degrees of freedom in the regression at high temperature. As a consequence, we decided to maintain relatively large τ_f values in the analysis: $p = 0.8, 0.9, 1.0$ in eq. (3.33). Although this choice makes the lowest temperature estimate of the local $a_1(1260)$ mass unreasonable, it allows a systematic exploration of the high temperature regime.

As stated before, the high temperature results should be interpreted as masses with caution. In an attempt to support the idea that estimated ground state masses contain information about the properties of the system even though they might not be regarded as isolated states, we study the degeneracy of the vector and axial channels at the level of the correlation function, that is, we perform a fit-independent analysis.

Degeneracy in the correlation functions

In order to look for degeneracies at the level of the correlation function, which indirectly test the results presented in Figure (3.12), we define a simple ratio between two correlation functions computed at the same temperature: $V(\tau;T)$ and $A(\tau;T)$, where V stands for vector and A stands for axial vector; both V and A correlation functions refer to the lightest flavour combination available, uu. The ratio, labelled $D(\tau;T)$, is a function of the Euclidean, defined as

$$D(\tau;T) = \frac{V(\tau;T)\,A(N_\tau/2;T)}{A(\tau;T)\,V(N_\tau/2;T)}. \tag{3.38}$$

This statistic can be estimated using our lattice vector and axial vector lattice correlation functions. The uncertainty in D can be estimated using bootstrap [13].

In eq. (3.38), the correlation functions evaluated at the middle point of the temporal direction avoid any possible required renormalisation. We treat $A(N_\tau/2;T)$ and $V(N_\tau/2;T)$ as numeric constants to avoid the computation of a ratio of 4 random variables.

We can analyse eq. (3.38) in order to understand its properties. To do so, we start by assuming that correlation functions involved in the computation of $D(\tau;T)$ can be modelled by the same functional form: eq. (3.5). As we are interested in the degeneracy of the ground state masses, we assume that close to the middle of the lattice, the correlation functions are dominated by their lowest energy mode. Consequently,

$$V(\tau;T) \simeq A_0^\rho \cosh(M_\rho\,(\tau - \frac{N_\tau}{2})), \tag{3.39}$$

and

$$A(\tau;T) \simeq A_0^{a_1} \cosh(M_{a_1}\,(\tau - \frac{N_\tau}{2})),. \tag{3.40}$$

Substituting eq (3.39) and (3.40) into eq. (3.38), and performing a Taylor expansion around $N_\tau/2$ allows us to write

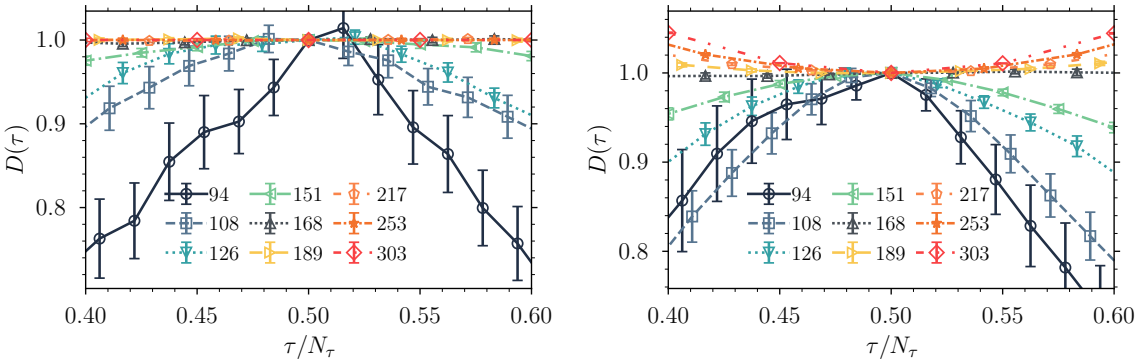
$$D(\tau;T) \simeq 1 + \frac{1}{2}\,(\tau - \frac{N_\tau}{2})\,(M_\rho^2 - M_{a_1}^2). \tag{3.41}$$

Around the middle of the lattice, the ratio is proportional to the difference of vector and axial vector ground state masses squared. Provided that the $SU(2)_A$ symmetry is restored, then both $\rho(770)$ and $a_1(1260)$ states are degenerate, which implies that $D(\tau;T)$ should be 1 around the middle of the lattice. This interpretation only applies as long as $A(\tau;T)$ and $V(\tau;T)$ are correctly modelled by eq. (3.5): $T < T_c$.

The estimated ratio computed from lattice correlation functions at different temperatures is shown in Figure (3.13) for both local and smeared sources. The lowest temperature available in our simulations, corresponding to a lattice of temporal size $N_\tau = 128$, is not included in the figures due to the fact that the lightest axial correlation function is extremely noisy around the middle of the lattice, as shown in Figure (3.8).

Figure (3.13) shows two different trends. First, below the pseudocritical temperature, located at $T = 166 \pm 2$ MeV, the ratio shows a concave shape ($\cap$), which is consistent with the experimental fact of the $a_1(1260)$ state being heavier than the $\rho(770)$ state. This behaviour is equivalent for both sources, and it is consistent with the temperature dependence of the ground state masses displayed in Figure (3.12).

Second, both sources lead to a ratio consistent with 1 close to the pseudocritical temperature — $T \simeq 166$ MeV, which suggests that a degeneracy is present in both channels at the level of the correlation function. Again, this behaviour is consistent with the high temperature estimates contained in Figure (3.12).



(a) Smeared sources **(b)** Local sources

Figure 3.13: $D(\tau;T)$ ratio computed using our lattice estimates of the $\rho(770)$ and the $a_1(1260)$ correlation functions at different temperatures. The labels represent the temperatures in MeV.

Above the pseudocritical temperature of the system, each source produces a different behaviour in the ratio: smeared sources produce a ratio consistent with 1 across all temperature above T_c, which implies that both channels are completely degenerate at the level of the correlation function — up to a normalisation constant; instead, local sources show a convex behaviour ($\cup$) above T_c. This difference supports the idea that the effects of smearing at high temperature should be further studied, as local and smeared estimates do not agree on the same behaviour.

3.2.3 $D_{(s)}$ mesons ground state masses in the hadronic phase

In this last collection of results, we focus on the temperature dependence of the charmed D and D_s mesonic ground state masses in the hadronic phase of QCD: $T < T_c$. The D mesons have quark content uc, while the D_s mesons replace a light quark with a strange quark, sc. Since the dawn of the field, charmed hadrons have been of interest to QGP phenomenology, with J/ψ suppression being one of the signatures of the formation of the plasma [27]. At high temperature, the charm quark is known to yield insight into the transport properties of the plasma [28, 29], while at low temperature the formation of open charmed states provides information

on charm-quark interaction in the medium [30]. As a result, D mesons have been investigated in Refs. [31–33].

As the $D_{(s)}$ mesons are formed by a combination of a charm quark and a light quark — u or s, their inherent energy scales allow them to show some relevant temperature dependence in the hadronic phase, while, at the same time, displaying a clear plateau in the effective mass. Light mesons, for example, can be affected by collective effects even at temperatures close to the transition temperature, while charmonium states (cc) tend to be too heavy to experience any thermal effects in the hadronic phase — see Figure (3.10). In addition, $D_{(s)}$ mesons are of interest in the hadronic effective theory community [34–40]. Consequently, the analysis of these states from a first-principles calculation is relevant both phenomenologically and due to the fact that they serve as a benchmark for effective theory calculations. Previous studies of open charm using lattice QCD include: an analysis of cumulants of net charm fluctuations [41], the extraction of screening masses in the D_s meson channel in the QGP [42], and the study of spectral functions obtained from D and D_s meson correlators on anisotropic lattices [43].

The results are presented in two parts: the first one focuses on the pseudoscalar and vector correlation function, while the second one is formed by the axial vector and scalar channels. In general, the pseudoscalar and vector correlation functions tend to yield better estimates than the scalar and axial vector correlation functions. This is a consequence of their inherently good signals, and the fact that their relatively low masses and quantum numbers do not allow them to decay into other states. We only report results extracted from smeared correlation functions in the $D_{(s)}$ mesonic sector.

Table (3.2) contains our lowest temperature results, extracted at $T = 47$ MeV, for both D and D_s mesons in all channels available. In addition, the most recent experimental masses extracted from Ref. [26] are also reported in the table. Our results are consistent with Ref. [44], which were measured at $T = 24$ MeV, generated by a lattice of size $N_\tau = 256$. It is worth mentioning that our uncertainties are generally smaller than the ones contained in Ref. [44], even when our lowest temperature lattice contains $N_\tau = 128$ different Euclidean times, half the value employed in the compared publication. Although their analysis uses a complex distillation algorithm where several correlation functions with the same quantum numbers are employed to isolate the states contributing to a particular correlation function, the analysis on each distilled correlation function is simpler than ours, which might explain the larger uncertainties in their estimates.

In order to assert that the results presented in Table (3.2) are plausible, we perform a simple analysis on our correlation functions. The analysis is based on

Table 3.2: D and D_s meson masses for all states available computed at the lowest temperature available: $T = 47$ MeV. The table shows the most recent PDG masses [26]. The results are expressed both in lattice units, as well as in physical units (MeV). The uncertainties of the results in MeV take into account the systematic uncertainty of the scale setting, as well as the statistical uncertainty generated by our methodology.

State	Channel	J^P	PDG MeV	$a_\tau m$	m MeV
D	Pseudoscalar	0^-	1869.65(5)	0.3086(1)	1876(4)
D^*	Vector	1^-	2010.26(5)	0.3291(1)	2001(4)
D_0^*	Scalar	0^+	2300(19)	0.3656(14)	2222(10)
D_1	Axial-vector	1^+	2420.8(5)	0.3823(70)	2325(43)
D_s	Pseudoscalar	0^-	1968.34(7)	0.3243(3)	1972(5)
D_s^*	Vector	1^-	2112.2(4)	0.3442(1)	2092(4)
D_{s0}^*	Scalar	0^+	2317.8(5)	0.3479(46)	2115(29)
D_{s1}	Axial-vector	1^+	2459.5(6)	0.3479(46)	2512(6)

the assumption that eq. (3.5) is a valid model for the contribution of each state to the correlation function. As a result, we model the behaviour of a particular low-temperature correlation function $C(\tau;T)$ close to the middle of lattice using the following:

$$\begin{aligned} C(\tau;T) &= A_0 \cosh(M_0\,(\tau - \frac{N_\tau}{2})) + A_e \cosh(M_e\,(\tau - \frac{N_\tau}{2})) \\ &= C_0(\tau;T) + C_e(\tau;T). \end{aligned} \tag{3.42}$$

In the equation above, C_0 models the contribution of the lowest energy mode, and C_e can be interpreted as encapsulating the effective contribution of all excited states to the correlation function. As $M_0 \ll M_e$, then C_e is a second-order effect at $\tau \to N_\tau/2$: $C(\tau;T) \simeq C_0(\tau;T)$.

From our simple model, we can construct the following ratio

$$R(\tau;T) = \frac{C(\tau;T)}{\mathcal{G}(\tau;T)}, \tag{3.43}$$

where $\mathcal{G}(\tau;T)$ is defined as

$$\mathcal{G}(\tau;T) = A_0 \cosh(M_R\,(\tau - \frac{N_\tau}{2})), \tag{3.44}$$

that is, a simple correlation function extracted from a 1-peak spectral function with zero width whose mass is equal to a given reference mass M_R — see eq. (3.2) with

$$\rho(\omega) = A\ 2\pi\ \delta(\omega - M_R), \tag{3.45}$$

where A is the amplitude of the spectral function. $\mathcal{G}(\tau;T)$ is a so-called reconstructed correlator [45–47].

Substituting eqs. (3.42) and (3.44) into eq. (3.43) leads to

$$R(\tau;T) = \frac{C_0(\tau;T) + C_e(\tau;T)}{\mathcal{G}(\tau;T)}. \tag{3.46}$$

In the case in which $M_R = M_0$, then, the ratio behaves as

$$R(\tau;T) = 1 + \frac{C_e(\tau;T)}{\mathcal{G}(\tau)}. \tag{3.47}$$

Due to the fact that higher-order states are exponentially suppressed at $0 \ll \tau < N_\tau/2$, we can expand the rightmost ratio using the Taylor expansion of a quotient of two hyperbolic cosines, leading to

$$\frac{C_e(\tau;T)}{\mathcal{G}(\tau;T)} \simeq \frac{A_e}{A_0}\,[1 + \frac{1}{2}\,(\tau - N_\tau)^2\,(M_e^2 - M_R^2)]. \tag{3.48}$$

A consequence of eq. (3.48) is that, if $M_R = M_0$ and $M_e \gg M_0$, then $R(\tau;T)$ evaluated at $\tau \to N_\tau/2$ is approximately

$$R(\tau;T) \simeq 1 + \frac{A_e}{A_0}, \text{ where } \quad 0 \ll \tau < N_\tau/2. \tag{3.49}$$

Provided that the ground state mass of a low temperature correlation function is close to a given reference mass, then the ratio $R(\tau;T)$ behaves like a constant.

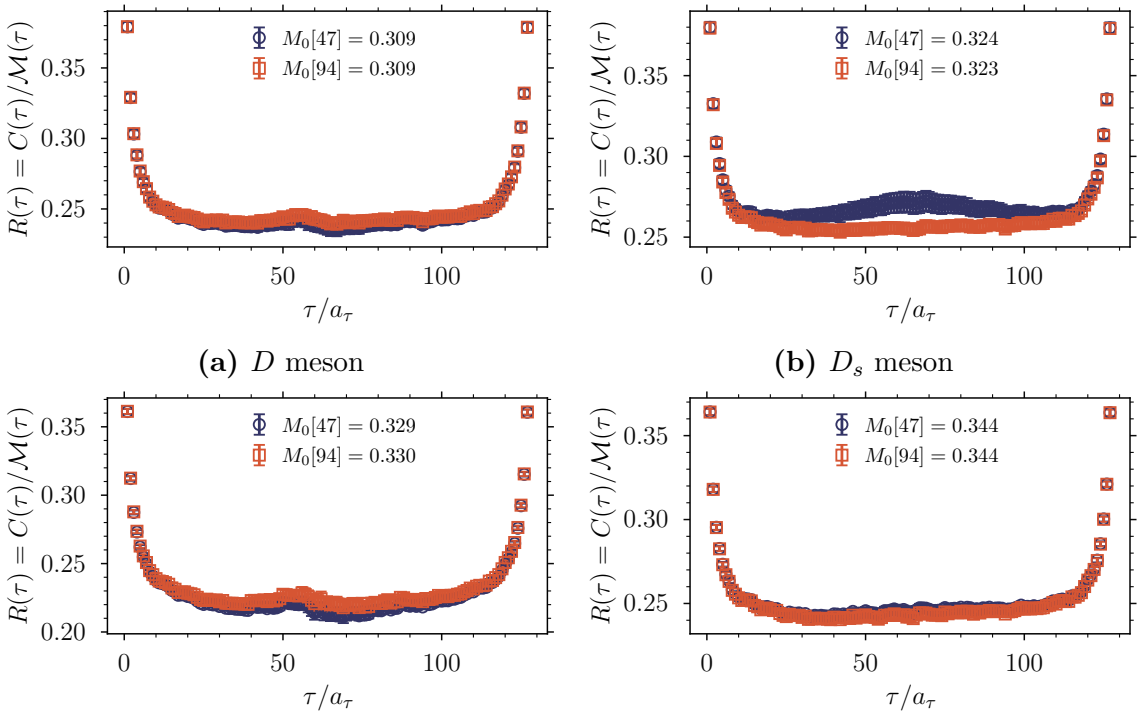
Through the computation of $R(\tau;T)$ in our correlation function data, we are able to visually validate our results, as $R(\tau;T)$ is sensitive to small changes in M_R:

$$\frac{C_0(\tau;T)}{\mathcal{M}(\tau;T)} \simeq 1 + \frac{1}{2}\,(\tau - N_\tau)^2\,(M_0^2 - M_R^2). \tag{3.50}$$

As in the previously introduced D-ratio — eq. (3.38) — $R(\tau;T)$ is a first-principles quantity that does not require any analysis on the correlation functions.

Figure (3.14) shows $R(\tau;T)$ as a function of τ/a_τ computed for both D and D_s mesons in the pseudoscalar and vector channels using the two lowest estimated

ground state masses as reference masses in $\mathcal{G}(\tau)$. The correlation function employed in the calculation, $C(\tau;T)$, corresponds to our lowest temperature estimate: $N_\tau = 128$. As expected, the results show that our estimated masses correctly model the large τ region of the correlation function.



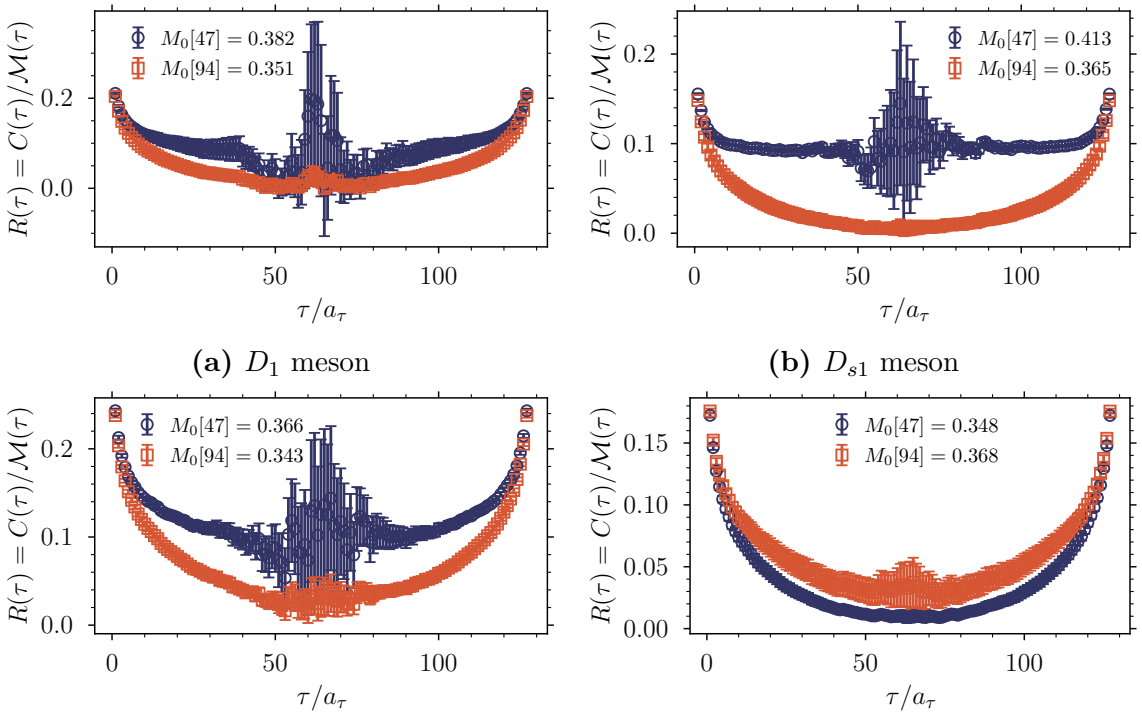
(a) D meson **(b)** D_s meson

(c) $D^\star$ meson **(d)** $D_s^\star$ meson

Figure 3.14: Pseudoscalar (top) and vector (bottom) ratios $R(\tau)$ using the fitted ground states masses at the two lowest temperatures: $N_\tau = 128, 64$. The correlation function employed in the computation of $R(\tau;T)$ corresponds to $N_\tau = 128$: $T = 47$ MeV.

Moreover, in Figure (3.15), we present $R(\tau;T)$ computed for the axial-vector and scalar channels. The data shows that extracting a reliable ground state mass for these two channels is difficult even at low temperature. The fact that the second-lowest temperature estimated ground state mass produces a ratio closer to a constant than the one extracted at lower temperature suggests that large temperature effects are affecting both channels. The presence of large temperature effects at low temperature was already encountered in the $a_1(1260)$ estimated masses, displayed in Figure (3.12).

The large difference in uncertainties encountered in the different subplots of Figure (3.15) is a consequence of the fact that the error in the estimated ground state masses employed as reference masses in $\mathcal{G}$ is taken into account when estimating $R(\tau;T)$. Due to the fact that our lowest temperature ground state estimate is less accurate than our second lowest temperature, $R(\tau;T)$ is noisier when computed when $M_0[T=47]$ than when computed with $M_0[T=97]$. We estimate the error in $R(\tau;T)$ using Monte Carlo error propagation.



(a) D_1 meson **(b)** D_{s1} meson

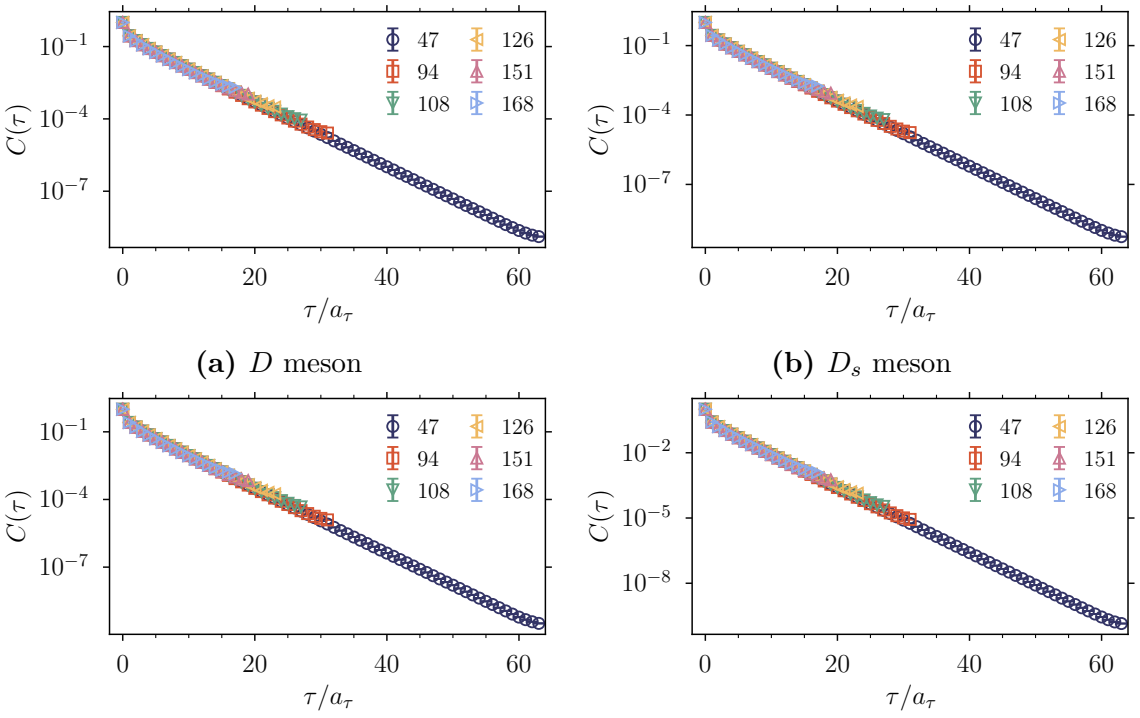
(c) $D_0^\star$ meson **(d)** $D_{s0}^\star$ meson

Figure 3.15: Axial-vector (top) and scalar (bottom) ratios $R(\tau)$ using the fitted ground states masses at the two lowest temperatures: $N_\tau = 128, 64$. The correlation function employed in the computation of $R(\tau;T)$ corresponds to $N_\tau = 128$: $T = 47$ MeV.

Pseudoscalar and vector channels

The lattice correlation functions employed in the extraction of the ground state masses for both the pseudoscalar and vector channels at different temperatures are presented in Figure (3.16). As expected, the correlation functions are periodic around

the middle point of the lattice in the temporal direction. Additionally, all correlation functions show a clear plateau at Euclidean times $0 \ll \tau < N_\tau/2$, which suggests that the spectral decomposition should validly model the correlation function data. As previously discussed, both channels yield good signals, which can be seen in the relatively low noise present in all signals.

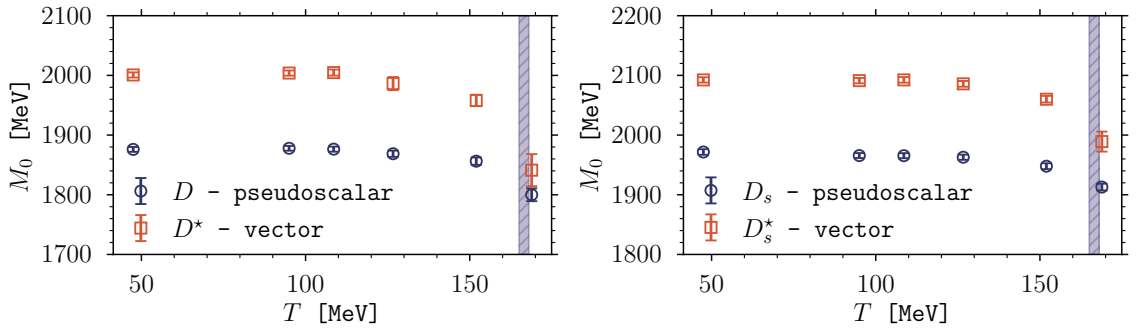


(a) D meson **(b)** D_s meson

(c) $D^\star$ meson **(d)** $D_s^\star$ meson

Figure 3.16: Pseudoscalar (top) and vector (bottom) correlation functions as a function of τ/a_τ. The labels represent the temperature in MeV.

Figure (3.17) contains the temperature dependence of all D and D_s mesons in both the pseudoscalar and vector channels. As expected, the data shows minimal temperature dependence in all ground state masses at temperatures below the critical temperature. Once the temperature of the system approaches the pseudocritical temperature, all states display a downwards shift in mass.



(a) D mesons **(b)** D_s mesons

Figure 3.17: Temperature dependence of the ground state mass of the D and D_s mesons in the pseudoscalar and vector channels. The vertical line denotes the pseudocritical temperature of the system.

Table (3.3) presents the values of the ground state masses at all temperatures considered. The results are in agreement with the ones reported in Ref. [34].

Table 3.3: D and D_s ground state masses (in MeV) as a function of the temperature in the hadronic phase for both pseudoscalar and vector channels. The uncertainty in the ground state masses combines the statistical uncertainty generated in our methodology and the systematic uncertainty from the scale setting.

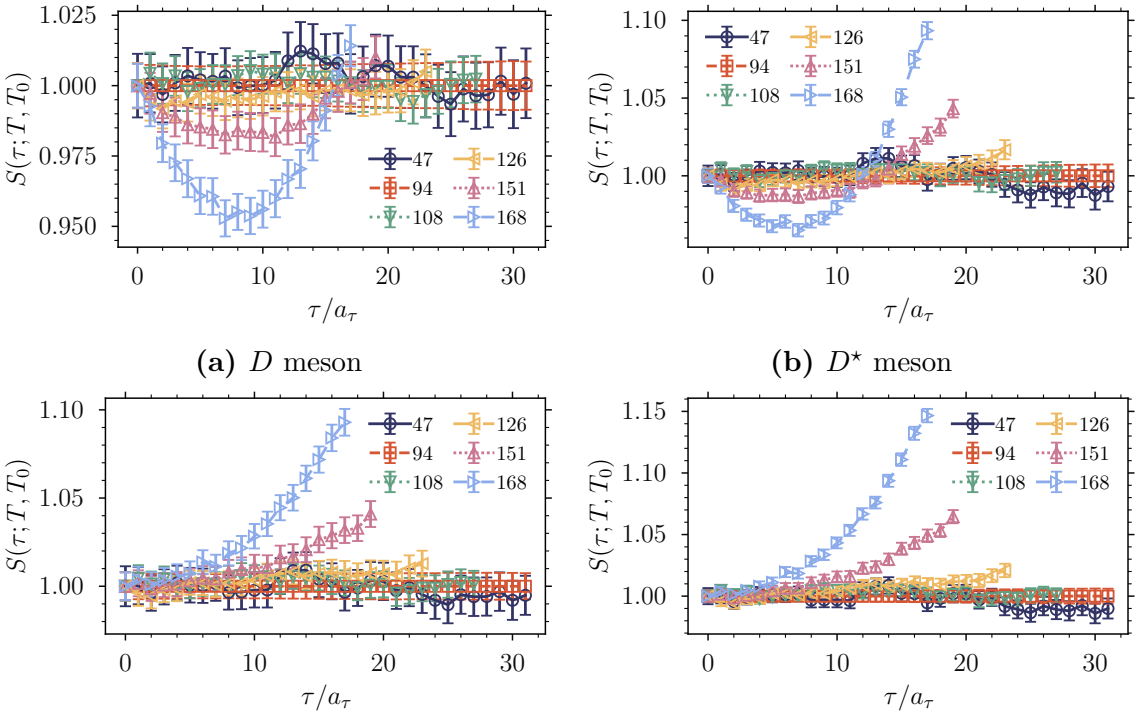
	J^P	PDG	T =47 MeV	95	109	127	152	169
D	0^-	1869.65(5)	1876(4)	1878(4)	1876(4)	1869(5)	1856(6)	1800(11)
D^*	1^-	2010.26(5)	2001(4)	2004(4)	2005(5)	1986(11)	1958(9)	1841(28)
D_s	0^-	1968.34(7)	1972(5)	1966(4)	1965(4)	1963(4)	1948(5)	1913(6)
D_s^*	1^-	2112.2(4)	2092(4)	2091(5)	2092(5)	2086(5)	2060(6)	1989(16)

In order to validate the results presented in Figure (3.17), we can compute a similar ratio to the one presented in eq. (3.43) across all temperatures available. This new ratio is defined as

$$S(\tau;T,T_0) = \frac{C(\tau;T)}{\mathcal{G}(\tau;T)} \Big/ \frac{C(\tau;T_0)}{\mathcal{G}(\tau;T_0)} = \frac{R(\tau;T)}{R(\tau;T_0)}. \tag{3.51}$$

Provided that the ground state mass of $C(\tau;T)$ is equal to the ground state mass of another reference correlation function $C(\tau;T_0)$, then $S(\tau;T,T_0)$ should be consistent with 1. Note that we divide each correlation function with a model in order to

suppress the bending produced by the hyperbolic cosines. The masses of $\mathcal{G}(\tau;T)$ and $\mathcal{G}(\tau;T_0)$ are equal, the only difference between both models is the temperature dependence induced by N_τ.



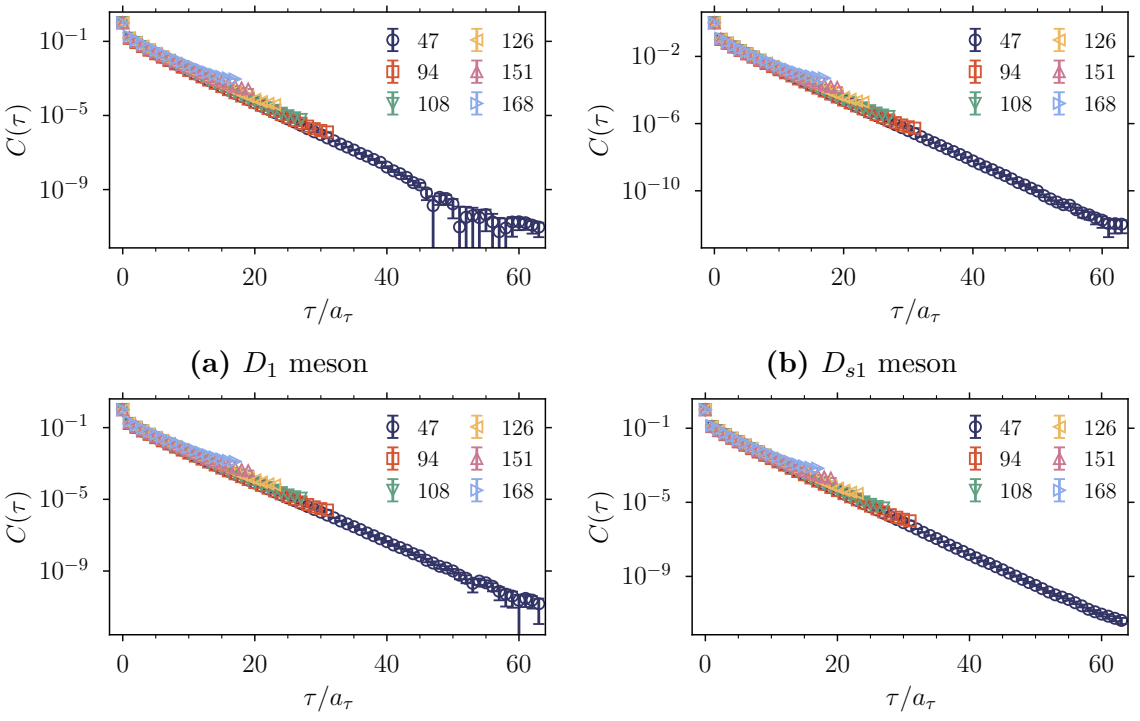
(a) D meson **(b)** $D^\star$ meson

(c) D_s meson **(d)** $D_s^\star$ meson

Figure 3.18: Pseudoscalar (top) and vector (bottom) ratio $S(\tau;T,T_0)$ as a function of τ/a_τ. The mass defining the models corresponds to our second-lowest temperature estimate, extracted on a lattice with $T = 97$ MeV. The labels in the figures represent the temperature in MeV.

A departure from 1 in $S(\tau;T,T_0)$ suggests that $M_0[T] \neq M_0[T_0]$. If this is indeed the case, then the temperature dependence of the correlation function is solely induced by N_τ, which implies that the spectral function of $C(\tau;T)$ is equivalent to the spectral function of $C(\tau;T_0)$. In this scenario, the correlation function is only modified by the temperature through N_τ — see eq. (3.2) and eq. (3.3). In contrast, if $S(\tau;T,T_0)$ is not consistent with 1, then we know that the temperature is affecting the correlation function non-trivially. However, from eq. (3.51), we cannot know if the thermal effects induce a mere shift in the ground state mass, or a complete change in the functional form of the spectral function. Nevertheless, as we work below the

pseudocritical temperature of the system, we expect small departures from $S(\tau) \simeq 1$ to be mainly caused by shifts in the mass of the lowest energy mode contributing to $C(\tau;T)$ in the pseudoscalar and vector channels.



(a) D_1 meson **(b)** D_{s1} meson

(c) $D_0^\star$ meson **(d)** $D_{s0}^\star$ meson

Figure 3.19: Axial vector (top) and scalar (bottom) correlation functions as a function of τ/a_τ. The labels in the figures represent the temperature in MeV.

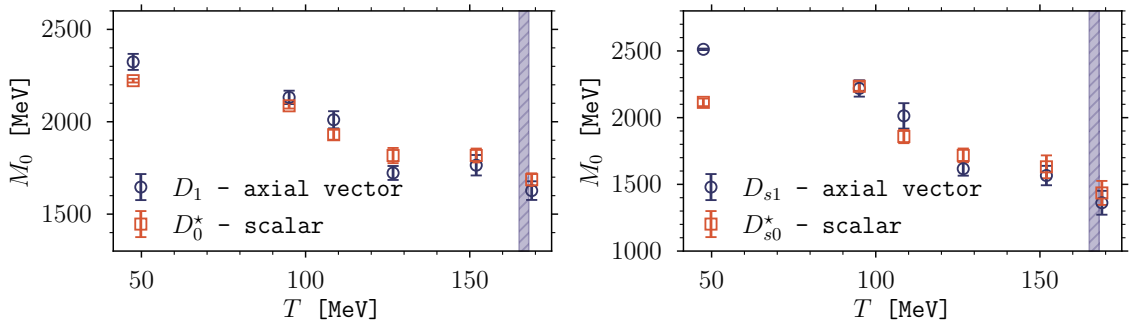
Figure (3.18) contains $S(\tau;T,T_0)$ computed on the pseudoscalar and vector channels in all relevant states. The data is consistent with the behaviour displayed by the ground state masses in Figure (3.17): at low temperature, the ground state masses are independent of the temperature, and, therefore, the spectral function is unaltered; however, once the temperature of the system approaches T_c, the masses of the states change with the temperature, which leads to a non-constant $S(\tau;T,T_0)$. The more S deviates from 1, the more we are certain that $M_0[T] \neq M_0[T_0]$. The transition from constant to non-constant ratio is spotted around $T = 151$ MeV. In our simulations, the pseudocritical temperature of the system, which is computed using the inflection point of the chiral condensate, has a value of $T_c = 166 \pm 2$ MeV; more information about the computation of T_c can be found in Appendix (A). As

in the previously defined ratios — eq. (3.38) and eq. (3.43), the computation of the ratio $S(\tau; T, T_0)$ does not involve any fitting.

Axial vector and scalar channels

Figure (3.19) displays the axial vector and scalar correlation functions employed in the extraction of the ground state masses. Although the scalar and axial vector correlation functions tend to suffer from an inherent signal loss as τ increases, which complicates the extraction of ground state masses, the $D_{(s)}$ meson correlation functions do not show this problem: a clear plateau is present at all temperatures. As a result, the difficult extraction of the ground state mass even at low temperature must be caused by another source — see Figure (3.15).

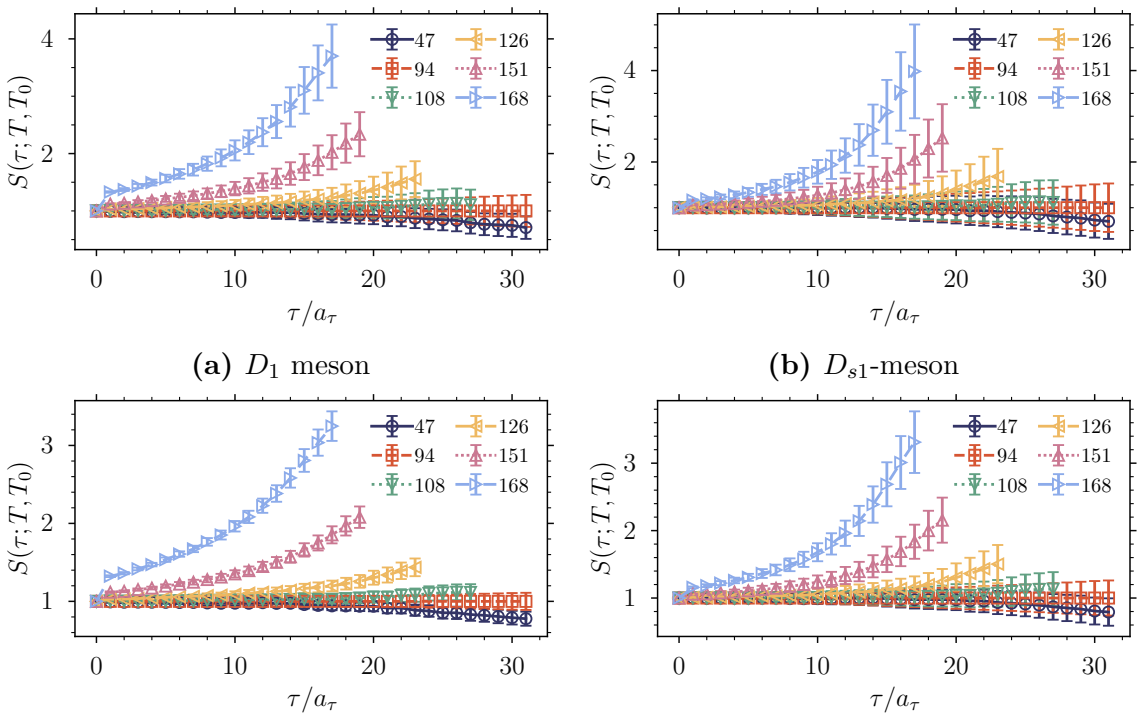
The temperature dependence of the estimated ground state masses in both relevant channels is shown in Figure (3.20). The results display large variations in the masses even at low temperatures: $T \to 0$. As stated before, this was already seen in the low-temperature estimates of the $a_1(1260)$ masses — see Figure (3.12). Although not entirely understood, we suspect that our scalar and axial vector operators might allow contributions from bound states, such as a pair ρ-π in the D_1 meson correlation function. The energy scales and quark numbers of the D mesons allow them to decay into other states, which might pollute the lowest energy mode of the correlation function. This phenomenon requires further investigation. The unstable nature of the estimated masses could also imply that the spectral decomposition is not a valid model of the correlation function even in the hadronic phase.



(a) D mesons **(b)** D_s mesons

Figure 3.20: Temperature dependence of the ground state mass of the D and D_s mesons in the axial vector and scalar channels. The vertical line denotes the pseudocritical temperature of the system.

As a first-principle test in Figure (3.20), we study the temperature dependence of the S-ratio defined in eq. (3.51) for both axial vector and scalar correlation functions. The results are presented in Figure (3.21), computed using the estimated ground state mass at $T = 97$ MeV as the reference mass for the models in $S(\tau; T, T_0)$. The results displayed in Figure (3.21) demonstrate that a clear temperature dependence is present at all temperatures: the lowest temperature correlation function does not produce a constant S-ratio; this was previously suggested by Figure (3.15). As the temperature increases, the deviation from a constant S is clearer.



(a) D_1 meson **(b)** D_{s1}-meson

(c) $D_0^\star$-meson **(d)** $D_{s0}^\star$-meson

Figure 3.21: Axial vector (top) and scalar (bottom) ratio $S(\tau; T, T_0)$ as a function of τ/a_τ. The mass defining the models corresponds to our second-lowest temperature estimate, extracted on a lattice with $T = 97$ MeV. The labels in the figures represent the temperature in MeV.

Whether the drastic τ-dependence encountered in the estimated S-ratio is due to a large shift in the ground state masses contributing to the correlation functions, as the results contained in Figure (3.20) suggest, or due to a change in the functional form of the spectral function in the hadronic phase, we cannot know with our current

tools and knowledge. However, from the computation of the S-ratio, we are sure that the temperature dependence of the correlation functions in these two channels cannot be merely induced by the mesonic kernel in eq. (3.2).

3.3 Conclusions

The methodology presented in this chapter allows a systematic exploration of the temperature dependence of ground state masses across a variety of channels, flavour combinations and temperatures. Our regression methodology does not rely on visual analysis of results, and reduces the bias introduced by choosing some hyperparameters of the analysis by varying those parameters as much as possible. By applying the methodology to our simulated thermal mesonic correlation functions, we are able to obtain results on different mesonic sectors. It is worth stressing that our results represent one of the few examples available in which mesonic states are systematically analysed at temperatures below and above the pseudocritical temperature of the system.

The temperature dependence of the pseudoscalar and vector channels ground state masses — see Figure (3.10) — are in agreement with the expected behaviour: at low temperature, eq. (3.5) validly models the contribution of each state to the correlation function; however, at $T > T_c$, the model becomes questionable, as the spectral function is no longer described by a sum of independent and isolated peaks. In the hadronic phase, minimal temperature dependence is present in the estimated ground state masses, which suggests that the temperature dependence of the correlation functions is mainly caused by *kinetical* effects, that is, it is only induced by the kernel in eq. (3.2). At high temperature, the light sector masses become degenerate, which implies that large collective effects are present in the system. As expected, these collective effects are suppressed in the heavy quark sector.

An important result of our analysis is the ubiquitous difference between local and smeared correlation functions at high temperature: both estimates lead to incompatible and diverging estimates at $T > T_c$, see Figure (3.11). This problem is spotted in the pseudoscalar and vector estimates, as well as in the axial vector channel. Further research in this direction is required to assert that smearing is acting as expected in high-temperature lattice simulations.

One of the main results of our analysis is the clear degeneration encountered in the $SU(2)_A$-related channels: $\rho(770)$ and $a_1(1260)$. This degeneration is spotted for both local and smeared sources, and in the masses as well as in the correlation functions: see Figure (3.12) and Figure (3.13) respectively. Our analysis suggest that the temperature at which the $SU(2)_A$ chiral symmetry is restored is close to the

pseudocritical temperature of the system.

In addition, due to the phenomenological interest surrounding these mesonic families, we studied the temperature dependence of the D and D_s mesons. Our analysis on these mesons represent the first systematic lattice QCD study performed throughout the whole hadronic phase of QCD. As Figure (3.17) suggests, the temperature dependence of the pseudoscalar and vector states is minimal at $T \ll T_c$; some temperature dependence is present at $T \simeq T_c$. These thermal effects are likely to be caused by a small shift in the ground state mass: Figure (3.16) and Figure (3.18) support this hypothesis. An analysis on the spectral functions of those states should allow a direct test on these results. In contrast, the axial vector and scalar states are heavily affected by the temperature even at $T \to 0$. This is suggested by both the masses, and the S-ratio: Figure (3.20) and Figure (3.21) respectively. We suspect that this large temperature dependence could be caused by the presence of bound states of lighter mesons in the correlation function, that is, the D-mesons decay into multiple states even at relatively low temperatures. Further research in this direction is required, for example by analysing equivalent correlation functions computed from different mesonic operators.

Chapter 3. References

[1] Y. Aoki et al. "The Order of the quantum chromodynamics transition predicted by the standard model of particle physics". In: *Nature* 443 (2006), pp. 675–678. DOI: 10.1038/nature05120. arXiv: hep-lat/0611014.

[2] Szabolcs Borsanyi et al. "Is there still any T_c mystery in lattice QCD? Results with physical masses in the continuum limit III". In: *JHEP* 09 (2010), p. 073. DOI: 10.1007/JHEP09(2010)073. arXiv: 1005.3508 [hep-lat].

[3] A. Bazavov et al. "The chiral and deconfinement aspects of the QCD transition". In: *Phys. Rev. D* 85 (2012), p. 054503. DOI: 10.1103/PhysRevD.85.054503. arXiv: 1111.1710 [hep-lat].

[4] Szabolcs Borsanyi et al. "The QCD equation of state with dynamical quarks". In: *JHEP* 11 (2010), p. 077. DOI: 10.1007/JHEP11(2010)077. arXiv: 1007.2580 [hep-lat].

[5] A. Bazavov et al. "Equation of state in (2+1)-flavor QCD". In: *Phys. Rev. D* 90 (2014), p. 094503. DOI: 10.1103/PhysRevD.90.094503. arXiv: 1407.6387 [hep-lat].

[6] Szabolcs Borsanyi et al. "Fluctuations of conserved charges at finite temperature from lattice QCD". In: *JHEP* 01 (2012), p. 138. DOI: 10.1007/JHEP01(2012)138. arXiv: 1112.4416 [hep-lat].

[7] A. Bazavov et al. "Fluctuations and Correlations of net baryon number, electric charge, and strangeness: A comparison of lattice QCD results with the hadron resonance gas model". In: *Phys. Rev. D* 86 (2012), p. 034509. DOI: 10.1103/PhysRevD.86.034509. arXiv: 1203.0784 [hep-lat].

[8] H. Sazdjian. "Introduction to chiral symmetry in QCD". In: *EPJ Web Conf.* 137 (2017). Ed. by Y. Foka, N. Brambilla and V. Kovalenko, p. 02001. DOI: 10.1051/epjconf/201713702001. arXiv: 1612.04078 [hep-ph].

[9] Gert Aarts et al. "Nucleons and parity doubling across the deconfinement transition". In: *Phys. Rev. D* 92.1 (2015), p. 014503. DOI: 10.1103/PhysRevD.92.014503. arXiv: 1502.03603 [hep-lat].

[10] Gert Aarts et al. "Light baryons below and above the deconfinement transition: medium effects and parity doubling". In: *JHEP* 06 (2017), p. 034. DOI: 10.1007/JHEP06(2017)034. arXiv: 1703.09246 [hep-lat].

[11] Gert Aarts et al. "Hyperons in thermal QCD: A lattice view". In: *Phys. Rev. D* 99.7 (2019), p. 074503. DOI: 10.1103/PhysRevD.99.074503. arXiv: 1812.07393 [hep-lat].

[12] Alexei Bazavov et al. "Meson screening masses in (2+1)-flavor QCD". In: *Phys. Rev. D* 100.9 (2019), p. 094510. DOI: 10.1103/PhysRevD.100.094510. arXiv: 1908.09552 [hep-lat].

[13] Bradley Efron and Robert J Tibshirani. *An introduction to the bootstrap*. CRC press, 1994.

[14] Erich L Lehmann and George Casella. *Theory of point estimation*. Springer Science & Business Media, 2006.

[15] Hirotugu Akaike. "A new look at the statistical model identification". In: *IEEE transactions on automatic control* 19.6 (1974), pp. 716–723.

[16] Hirotogu Akaike. "Information theory and an extension of the maximum likelihood principle". In: *Selected papers of hirotugu akaike*. Springer, 1998, pp. 199–213.

[17] D Anderson and K Burnham. "Model selection and multi-model inference". In: *Second. NY: Springer-Verlag* 63.2020 (2004), p. 10.

[18] Solomon Kullback and Richard A Leibler. "On information and sufficiency". In: *The annals of mathematical statistics* 22.1 (1951), pp. 79–86.

[19] R. Rapp and J. Wambach. "Chiral symmetry restoration and dileptons in relativistic heavy ion collisions". In: *Adv. Nucl. Phys.* 25 (2000), p. 1. DOI: 10.1007/0-306-47101-9_1. arXiv: hep-ph/9909229.

[20] Tanmoy Bhattacharya et al. "QCD Phase Transition with Chiral Quarks and Physical Quark Masses". In: *Phys. Rev. Lett.* 113.8 (2014), p. 082001. DOI: 10.1103/PhysRevLett.113.082001. arXiv: 1402.5175 [hep-lat].

[21] Carleton E. Detar and John B. Kogut. "The Hadronic Spectrum of the Quark Plasma". In: *Phys. Rev. Lett.* 59 (1987), p. 399. DOI: 10.1103/PhysRevLett.59.399.

[22] Irina Pushkina et al. "Properties of hadron screening masses at finite baryonic density". In: *Phys. Lett. B* 609 (2005), pp. 265–270. DOI: 10.1016/j.physletb.2005.01.006. arXiv: hep-lat/0410017.

[23] Stefan Struber and Dirk H. Rischke. "Vector and axialvector mesons at nonzero temperature within a gauged linear sigma model". In: *Phys. Rev. D* 77 (2008), p. 085004. DOI: 10.1103/PhysRevD.77.085004. arXiv: 0708.2389 [hep-th].

[24] A. Ohnishi, N. Kawamoto and K. Miura. "Brown-Rho Scaling in the Strong Coupling Lattice QCD". In: *Mod. Phys. Lett. A* 23 (2008). Ed. by Atsushi Hosaka et al., pp. 2459–2464. DOI: 10.1142/S0217732308029587. arXiv: 0803.0255 [nucl-th].

[25] Juan M. Torres-Rincon. "Degeneracy Patterns of Chiral Companions at Finite Temperature". In: *Symmetry* 13.8 (2021), p. 1400. DOI: 10.3390/sym13081400. arXiv: 2106.15700 [hep-ph].

[26] P. A. Zyla et al. "Review of Particle Physics". In: *PTEP* 2020.8 (2020). DOI: 10.1093/ptep/ptaa104.

[27] T. Matsui and H. Satz. "J/ψ Suppression by Quark-Gluon Plasma Formation". In: *Phys. Lett. B* 178 (1986), pp. 416–422. DOI: 10.1016/0370-2693(86)91404-8.

[28] Guy D. Moore and Derek Teaney. "How much do heavy quarks thermalize in a heavy ion collision?" In: *Phys. Rev. C* 71 (2005), p. 064904. DOI: 10.1103/PhysRevC.71.064904. arXiv: hep-ph/0412346.

[29] Hendrik van Hees and Ralf Rapp. "Thermalization of heavy quarks in the quark-gluon plasma". In: *Phys. Rev. C* 71 (2005), p. 034907. DOI: 10.1103/PhysRevC.71.034907. arXiv: nucl-th/0412015.

[30] V. Greco, C. M. Ko and R. Rapp. "Quark coalescence for charmed mesons in ultrarelativistic heavy ion collisions". In: *Phys. Lett. B* 595 (2004), pp. 202–208. DOI: 10.1016/j.physletb.2004.06.064. arXiv: nucl-th/0312100.

[31] Vitalii Ozvenchuk et al. "D-meson propagation in hadronic matter and consequences for heavy-flavor observables in ultrarelativistic heavy-ion collisions". In: *Phys. Rev. C* 90 (2014), p. 054909. DOI: 10.1103/PhysRevC.90.054909. arXiv: 1408.4938 [hep-ph].

[32] G. Aarts et al. "Heavy-flavor production and medium properties in high-energy nuclear collisions - What next?" In: *Eur. Phys. J. A* 53.5 (2017), p. 93. DOI: 10.1140/epja/i2017-12282-9. arXiv: 1612.08032 [nucl-th].

[33] Andrea Beraudo et al. "In-medium hadronization of heavy quarks and its effect on charmed meson and baryon distributions in heavy-ion collisions". In: *Eur. Phys. J. C* 82.7 (2022), p. 607. DOI: 10.1140/epjc/s10052-022-10482-y. arXiv: 2202.08732 [hep-ph].

[34] Glòria Montaña et al. "Impact of a thermal medium on D mesons and their chiral partners". In: *Phys. Lett. B* 806 (2020), p. 135464. DOI: 10.1016/j.physletb.2020.135464. arXiv: 2001.11877 [hep-ph].

[35] Glòria Montaña et al. "Pseudoscalar and vector open-charm mesons at finite temperature". In: *Phys. Rev. D* 102.9 (2020), p. 096020. DOI: 10.1103/PhysRevD.102.096020. arXiv: 2007.12601 [hep-ph].

[36] Glòria Montaña et al. "Open-charm Euclidean correlators within heavy-meson EFT interactions". In: *Eur. Phys. J. A* 56.11 (2020), p. 294. DOI: 10.1140/epja/s10050-020-00300-y. arXiv: 2007.15690 [hep-ph].

[37] Glòria Montaña Faiget. "Effective-theory description of heavy-flavored hadrons and their properties in a hot medium". PhD thesis. Barcelona U., 2022. arXiv: 2207.10752 [hep-ph].

[38] C. Fuchs et al. "D-mesons and charmonium states in hot pion matter". In: *Phys. Rev. C* 73 (2006), p. 035204. DOI: 10.1103/PhysRevC.73.035204. arXiv: nucl-th/0410065.

[39] Chihiro Sasaki. "Fate of charmed mesons near chiral symmetry restoration in hot matter". In: *Phys. Rev. D* 90.11 (2014), p. 114007. DOI: 10.1103/PhysRevD.90.114007. arXiv: 1409.3420 [hep-ph].

[40] T. Buchheim et al. "Chiral-partner D mesons in a heat bath within QCD sum rules". In: *J. Phys. G* 45.8 (2018), p. 085104. DOI: 10.1088/1361-6471/aab44e. arXiv: 1801.01472 [nucl-th].

[41] A. Bazavov et al. "The melting and abundance of open charm hadrons". In: *Phys. Lett. B* 737 (2014), pp. 210–215. DOI: 10.1016/j.physletb.2014.08.034. arXiv: 1404.4043 [hep-lat].

[42] Alexei Bazavov et al. "In-medium modifications of open and hidden strange-charm mesons from spatial correlation functions". In: *Phys. Rev. D* 91.5 (2015), p. 054503. DOI: 10.1103/PhysRevD.91.054503. arXiv: 1411.3018 [hep-lat].

[43] Aoife Kelly, Alexander Rothkopf and Jon-Ivar Skullerud. "Bayesian study of relativistic open and hidden charm in anisotropic lattice QCD". In: *Phys. Rev. D* 97.11 (2018), p. 114509. DOI: 10.1103/PhysRevD.97.114509. arXiv: 1802.00667 [hep-lat].

[44] Gavin K. C. Cheung et al. "Excited and exotic charmonium, D_s and D meson spectra for two light quark masses from lattice QCD". In: *JHEP* 12 (2016), p. 089. DOI: 10.1007/JHEP12(2016)089. arXiv: 1610.01073 [hep-lat].

[45] Saumen Datta et al. "Behavior of charmonium systems after deconfinement". In: *Phys. Rev. D* 69 (2004), p. 094507. DOI: 10.1103/PhysRevD.69.094507. arXiv: hep-lat/0312037.

[46] Gert Aarts et al. "Charmonium at high temperature in two-flavor QCD". In: *Phys. Rev. D* 76 (2007), p. 094513. DOI: 10.1103/PhysRevD.76.094513. arXiv: 0705.2198 [hep-lat].

[47] Gert Aarts and Jose M. Martinez Resco. "Continuum and lattice meson spectral functions at nonzero momentum and high temperature". In: *Nucl. Phys. B* 726 (2005), pp. 93–108. DOI: 10.1016/j.nuclphysb.2005.08.012. arXiv: hep-lat/0507004.

Chapter 4

The pion velocity in the QCD medium

In this chapter, we introduce and analyse the pion velocity in the QCD medium as a function of the temperature. The pion velocity, labelled u, is a dimensionless quantity that describes the temperature dependence of the pion dispersion relation in the chiral limit of light quarks — $m_q \to 0$ — and small external momenta — $\vec{k} \to \vec{0}$. The thermal dispersion relation measures the energy that a pion particle carries at a particular external momentum $\vec{k}$ and temperature T. We label this quantity using $\omega(M_\pi, \vec{k}; T)$, where M_π represents the rest mass of the pion. In the previously mentioned limits, the pion velocity and the thermal dispersion relation relate through

$$\omega(M_\pi, \vec{k};\ T) = u^2(T)\, \omega(M_\pi, \vec{k}; T = 0). \tag{4.1}$$

A chiral limit expression of the pion velocity expressed in terms of static quantities is

$$u^2(T) = -\frac{4\, m_q^2}{M_\pi^2} \left. \frac{C_{PP}(\tau, \vec{k} = \vec{0};\ T)}{C_{AA}(\tau, \vec{k} = \vec{0};\ T)} \right|_{\tau = N_\tau/2}. \tag{4.2}$$

This expression was first derived in Ref. [1], and its roots lie in the analysis of pion dynamics in the chiral limit of QCD presented in Refs. [2, 3].

In eq. (4.2), m_q represents the light quark mass, M_π is the pion mass, C_{PP} corresponds to the pseudoscalar (γ_5) thermal correlation function , and C_{AA} is the temporal axial vector ($\gamma_0\gamma_5$) thermal correlation function. The precise definitions of the pseudoscalar and axial vector correlation functions can be found in Appendix (B). Throughout this chapter, we make use of the Euclidean Dirac γ-matrices: γ_0, γ_1, γ_2 and γ_3, where γ_0 is the temporal γ-matrix.

This chapter is divided into two sections. In the first one, we explicitly derive the pion velocity expression following Ref. [1]. To do so, we first introduce the so-called Ward-Takahashi identities. Once the pion velocity expression is constructed, we estimate its temperature dependence in our lattice setup, described in detail in Appendix (A).

4.1 Derivation of the pion velocity

In order to construct eq. (4.2), the so-called Ward-Takahashi identities are required. These identities relate different correlation functions through symmetries of their action. As a result, they play a similar role to Noether's conserved currents for correlation functions. In addition to being essential in the derivation of the pion velocity expression, the Ward-Takahashi identities also allow the estimation of light quark masses on the lattice.

4.1.1 Ward-Takahashi identities

The Euclidean Ward-Takahashi identities emerge from the analysis of the variation of expectation values of field operators $\hat{O}$ under infinitesimal field transformations. In our particular case, we assume that $\hat{O}$ is a composite operator constructed from two fermionic fields: $\hat{O} = \hat{O}[\psi, \bar{\psi}]$. The Euclidean expectation value of $\hat{O}$ is defined as

$$\langle \hat{O}[\psi, \bar{\psi}] \rangle = \frac{1}{Z} \int D\psi D\bar{\psi}\, O[\psi, \bar{\psi}] \, \exp\big(- S_E[\psi, \bar{\psi}] \big), \tag{4.3}$$

where S_E represents the Euclidean action defining the dynamics of the system, and Z is the normalising partition function defined in eq. (1.4). As we are interested in the chiral limit of QCD, the fermionic fields, ψ and $\bar{\psi}$, represent the light quarks u and d, which in our particular case are assumed to be degenerate: $m_u = m_d = m_q$.

The fermionic quark fields transform under an infinitesimal transformation as

$$\psi' = \psi + \delta\psi \quad \text{and} \quad \bar{\psi}' = \bar{\psi} + \delta\bar{\psi}. \tag{4.4}$$

We can insert the transformed fermionic fields into all field-dependent objects in eq. (4.3) in order to construct the variation of $\langle \hat{O} \rangle$ with respect to both fermionic fields. The result is:

$$\langle \hat{O} \rangle' = \langle \hat{O} \rangle + \langle \delta\hat{O} \rangle + \langle \hat{O}\, \delta S_E \rangle + \langle \hat{O}\, J \rangle + \frac{\delta Z}{Z} \langle \hat{O} \rangle. \tag{4.5}$$

In the expansion above, the first term represents the variation of the field operator $\hat{O}$; the second term corresponds to the variation of the Euclidean action; the third term represents the variation of the integration measures, $D\psi$ and $D\bar{\psi}$, described by the determinant of the Jacobian matrix of the transformation, $J \simeq \partial\psi/\partial\psi'$; lastly, the last term corresponds to the variation of the partition function Z in eq. (4.3). Due to the infinitesimal nature of eq. (4.4), higher-order terms in $\delta\psi$ and $\delta\bar{\psi}$ vanish.

As we are mainly interested in transformations that leave the action invariant, we can safely discard the last term in eq. (4.5): $\delta Z \simeq \langle \delta S_E \rangle = 0$. Moreover, we only focus on non-anomalous transformations, that is, transformations that leave the measure invariant. This implies that the Jacobian term in eq. (4.5) is also discarded.

Assuming that the symmetry transformations leave the expectation value invariant, then

$$\langle \hat{O}' \rangle = \langle \hat{O} \rangle, \tag{4.6}$$

which leads to:

$$\langle \hat{O}\, \delta S_E \rangle = -\langle \delta \hat{O} \rangle. \tag{4.7}$$

We continue our analysis by characterising the transformations defined in eq. (4.4) as infinitesimal local variations depending on a real function $\omega(x)$, and belonging to the algebra of a Lie symmetry group. This class of transformations can be expressed as

$$\psi' = \psi + i\omega(x)\hat{T}\,\psi(x) \quad \text{and} \quad \bar{\psi}' = \bar{\psi} - i\omega(x)\bar{\psi}(x)\hat{T}, \tag{4.8}$$

where $\hat{T}$ corresponds to a generator of the group transformation. Additionally, we assume that $\omega(x)$ is infinitesimally small, $|\omega(x)| \ll 1$ for all x, and only non-zero in a small neighbourhood of the Euclidean space-time coordinate x. This locality condition ensures that no boundary terms emerge in our derivation.

To proceed, we apply the transformations defined in eq. (4.8) to the standard Euclidean fermionic action, defined as

$$S_E = \int d^4x\, \bar{\psi}(x)\big[\gamma_\mu \partial_\mu + m_q\big]\psi(x). \tag{4.9}$$

The variation of the Euclidean action at first order in $\omega(x)$ is just

$$\delta S_E = i \int d^4x\, \bar{\psi}\Big[\partial_\mu \omega \gamma_\mu \hat{T} + \omega\big[\gamma_\mu \hat{T} + \hat{T}\gamma_\mu\big]\partial_\mu + \omega\big[m_q \hat{T} + \hat{T} m_q\big]\Big]\psi, \tag{4.10}$$

where all space-time dependences have been omitted to simplify the notation. In principle, the quark mass m_q and the group generators $\hat{T}$ might not commute: an example of this can be found when the fermionic action is formulated in terms of a

non-degenerate $SU(2)_I$ isospin doublet, $\psi = (u, d)$. However, in our particular case, we treat m_q as a scalar parameter, which leads to

$$\delta S_E = i \int d^4x\, \bar{\psi} \Big[\partial_\mu \omega \gamma_\mu \hat{T} + \omega \big[\gamma_\mu \hat{T} + \hat{T}\gamma_\mu\big] \partial_\mu + 2 m_q \hat{T} \omega \Big] \psi. \tag{4.11}$$

The right-hand side of the equation above can be integrated by parts in order to eliminate the first term, which depends on the derivative of $\omega(x)$. To do so, we employ the following identity

$$\partial_\mu [\omega(x)\Phi(x)] = \partial_\mu \omega(x) \Phi(x) + \omega(x) \partial_\mu \Phi(x), \tag{4.12}$$

where $\Phi(x) = \gamma_\mu \hat{T} \psi(x)$. Integrating the equation above over all the Euclidean space-time, labelled $\mathcal{R}$, and applying Gauss's theorem, allows us to arrive at the following expression:

$$\int_{\mathcal{R}} d^4x\, \partial_\mu [\omega(x)\Phi(x)] = \Big[\omega(x)\Phi(x)\Big]_{\partial\mathcal{R}} = 0. \tag{4.13}$$

The last step is a consequence of the locality of $\omega(x)$: in the boundary of the region of integration ($\partial\mathcal{R}$), $\omega(x)$ vanishes.

The non-zero term in eq. (4.12) can be plugged into eq. (4.11) to produce

$$\delta S_E = i \int d^4x\, \omega \Big[- \partial_\mu [\bar{\psi}\gamma_\mu \hat{T} \psi] + \bar{\psi}(\gamma_\mu \hat{T} + \hat{T}\gamma_\mu)\partial_\mu \psi + 2 m_q \bar{\psi} \hat{T} \psi \Big]. \tag{4.14}$$

By inserting eq. (4.14) into eq. (4.7), we are able to build the following Ward-Takahashi identity:

$$\langle i \int d^4x\, \omega(x) \Big[- \partial_\mu [\bar{\psi}\gamma_\mu \hat{T} \psi] + \bar{\psi}(\gamma_\mu \hat{T} + \hat{T}\gamma_\mu)\partial_\mu \psi + 2 m_q \bar{\psi} \hat{T} \psi \Big] \hat{O} \rangle = 0. \tag{4.15}$$

To obtain the identity above, we have assumed that $\langle \delta \hat{O} \rangle = 0$ in eq. (4.7). The equation above must be true independently of $\omega(x)$.

In the particular case in which the group generators satisfy

$$\hat{T}\gamma_\mu + \gamma_\mu \hat{T} = 0, \tag{4.16}$$

then, we obtain one of the so-called Ward-Takahashi identities:

$$\langle \partial_\mu [\bar{\psi}(x)\gamma_\mu \hat{T} \psi(x)] \hat{O}(y) \rangle = 2 m_q \langle [\bar{\psi}(x) \hat{T} \psi(x)] \hat{O}(y) \rangle. \tag{4.17}$$

Note that the source operator in eq. (4.17), $\hat{O}(y)$, must strictly be located outside the neighbourhood of the sink operator: $y \neq x$. This condition ensures that no

boundary terms are present in the previously derived identities. A violation of this restriction implies that $\hat{O}(y)$ emerges in all integrals, making eq. (4.13) non-valid.

A particularly useful Ward-Takahashi identity is obtained by supposing that the action is invariant under an $SU(2)_A$ infinitesimal chiral transformation, defined as

$$\psi' = \psi + i\omega(x)\gamma_5\frac{\tau^a}{2}\psi(x) \quad \text{and} \quad \bar{\psi}' = \bar{\psi} + i\omega(x)\bar{\psi}(x)\gamma_5\frac{\tau^a}{2}, \tag{4.18}$$

where τ^a represents one of the Pauli matrices. For this particular symmetry, $\hat{T}$ is just $\gamma_5\tau^a/2$ in eq. (4.17). We can substitute the group generator $\hat{T}$ in eq. (4.17) with the appropriate version, and make use of the pseudoscalar and axial vector densities, defined as

$$P^a = \bar{\psi}\gamma_5\frac{\tau^a}{2}\psi \quad \text{and} \quad A^a_\mu = \bar{\psi}\gamma_\mu\gamma_5\frac{\tau^a}{2}\psi, \tag{4.19}$$

to construct the following identity:

$$\langle\partial_\mu A^a_\mu(x)\hat{O}'(y)\rangle = 2m_q\,\langle P^a(x)\hat{O}'(y)\rangle. \tag{4.20}$$

The identity above is sometimes referred to with the name *partially conserved axial current* (PCAC) identity.

Two important things can be learned from eq. (4.20). The first one is that, in the chiral limit, all correlation functions containing the axial divergence are zero. In this limit, it is said that the axial current is conserved. This is similar to stating that the following current equation is satisfied:

$$\partial_\mu A^a_\mu(x) = 0. \tag{4.21}$$

The second consequence is that, close to the chiral limit, eq. (4.20) allows a direct estimation of the light quark masses as a function of two measurable correlation functions.

On the lattice, in order to obtain a precise measure of m_q, the pseudoscalar density is usually placed in the source operator of eq. (4.20) as it produces accurate correlation function estimates. Although eq. (4.20) might seem to be valid for all quark flavours, it is not. To derive it, we have assumed that the action is invariant under $SU(2)_A$ transformations. This is only possible in the limit of small quark masses. As a result, only light quark masses can be estimated through the application of eq. (4.20). It is common to refer to the estimation of a quark mass computed using the PCAC relationship as its PCAC mass: m_q^{PCAC}.

In the case in which $\langle\delta\hat{O}\rangle \neq 0$, then eq. (4.7) transforms into:

$$\langle i\int d^4x\,\omega\Big[-\partial_\mu[\bar{\psi}\gamma_\mu\hat{T}\psi] + \bar{\psi}(\gamma_\mu\hat{T} + \hat{T}\gamma_\mu)\partial_\mu\psi + 2m_q\bar{\psi}\hat{T}\psi\Big]\hat{O}\rangle = \langle\delta\hat{O}\rangle, \tag{4.22}$$

where $\delta\hat{O}$ is computed using the transformation defined in eq. (4.8). We can select $\omega(x) = e^{ikx}$ and $\hat{O}(y) = P^b(x)$ to obtain

$$\frac{\delta^{ab}}{2}\langle\bar{\psi}\psi\rangle = ik_\mu \int d^4x e^{-ikx}\langle P^b(y)A^a_\mu(x)\rangle + 2m_q \int d^4x e^{-ikx}\langle P^b(y)P^a(x)\rangle. \quad (4.23)$$

The integrals in the equation above can be eliminated in order to obtain the following relationship:

$$\frac{1}{2}\delta^{ab}\delta^4(x-y)\langle\bar{\psi}\psi\rangle = \langle P^b(y)\partial_\mu A^a_\mu(x)\rangle + 2m_q\,\langle P^b(y)P^a(x)\rangle. \quad (4.24)$$

More details on the derivation of the Ward-Takahashi identity defined above can be found in Ref. [1].

4.1.2 The pion velocity expression

Having derived the Ward-Takahashi identities, we are in the position of building the pion velocity expression shown in eq. (4.2). We follow the derivation present in Ref. [1]. To derive eq. (4.2), we start from the following *ansatz*,

$$\int d\tau\langle P^a(0)\vec{A}^b(\tau,\vec{x})\rangle = \delta^{ab}f(r)\hat{u}_r, \quad (4.25)$$

where $\hat{u}_r$ represents the radial unit vector: $\hat{u}_r = \vec{r}/|\vec{r}|$.

To continue with the derivation, we take the chiral limit of eq. (4.24),

$$\langle P^b(0)\partial_\mu A^a_\mu(x)\rangle = \frac{1}{2}\langle\bar{\psi}\psi\rangle\delta^{ab}\delta^4(x). \quad (4.26)$$

The equation above can be integrated over the Euclidean space-time

$$\mathcal{R} = \lim_{R\to\infty}\,\{0 \le r \le R;\ 0 \le \tau \le \beta\} \quad (4.27)$$

in order to obtain

$$\int_0^\beta \int_{r<R} d\tau d^3x\,\langle P^b(0)\partial_\mu A^a_\mu(x)\rangle = \int_0^\beta \int_{r<R} d\tau d^3x\,\frac{1}{2}\langle\bar{\psi}\psi\rangle\delta^{ab}\delta^4(x). \quad (4.28)$$

Note that the left-hand side in the equation above can be transformed using

$$\partial_\mu[\langle P^b(0)A^a_\mu(x)\rangle] = \langle\partial_\mu P^b(0)A^a_\mu(x)\rangle + \langle P^b(0)\partial_\mu A^a_\mu(x)\rangle, \quad (4.29)$$

where the first term in the right-hand side is zero as it is independent of x. Plugging the above equation into the left-hand side of eq. (4.28) produces the following relationship:

$$\partial_\mu \langle P^b(0) A^a_\mu(x)\rangle = \partial_\tau \langle P^a(0) A^b_0(x)\rangle + \vec{\nabla}\,\langle P^a(0)\vec{A}(x)\rangle, \tag{4.30}$$

where the index 0 represents the time component τ of the vector field: $A_0 = A_\tau$. Integrating this new relationship over the same Euclidean space-time in eq. (4.28) leads to

$$\begin{aligned}&\int d^4x\, \partial_\mu \langle P^b(0) A^a_\mu(x)\rangle = \\ &\int d^3x\, \langle P^a(0) A^b_0(x)\rangle\Big|_0^\beta + \int d\tau d^3x\, \vec{\nabla}\langle P^a(0)\vec{A}^b(x)\rangle.\end{aligned} \tag{4.31}$$

The first term in the right-hand side is zero as the correlation function is periodic in time for bosonic operators. Gauss's theorem can be applied to the equation above in order to perform the integral of the second term in the right-hand side. The result is

$$\begin{aligned}\int d\tau d^3x\, \vec{\nabla}\,\langle P^a(0)\vec{A}^b(x)\rangle &= \int r^2 d\tau d\theta d\phi\, \langle P^a(0)\vec{A}^b(x)\rangle \\ &= \int r^2 d\sigma d\phi\, f(r) = 4\pi r^2 f(r),\end{aligned} \tag{4.32}$$

which can then be inserted into eq. (4.25), to find

$$f(r) = -\frac{\langle \psi\bar{\psi}\rangle}{8\pi r^2}. \tag{4.33}$$

Furthermore, another useful expression required in the derivation of the pion velocity can be obtained. To do so, we start by integrating eq. (4.24) over the Euclidean space-time:

$$\int_{\mathcal{R}} d^3x\, \langle P^a(0)\partial_\mu A^b_\mu(x)\rangle. \tag{4.34}$$

Again, we integrate this equation by parts to obtain

$$\begin{aligned}&\int_{\mathcal{R}} d^3x\, \langle P^a(0)\partial_\mu A^b_\mu(x)\rangle = \\ &\partial_\tau \int_{\mathcal{R}} d^3x\, \langle P^a(0) A_0(x) + \int_{\partial\mathcal{R}} d\vec{\sigma}\, \langle P^a(0)\vec{A}^b(x)\rangle,\end{aligned} \tag{4.35}$$

which must be equal to eq. (4.26) in the chiral limit.

Integrating the right-hand side of eq. (4.26) allows us to obtain

$$\frac{1}{2}\langle\bar{\psi}\psi\rangle\delta^{ab}\delta^4(\tau-0) = \\ \partial_\tau\int_{\mathcal{R}} d^3x\,\langle P^a(0)A_0(x)\rangle + \int_{\partial\mathcal{R}} d\vec{\sigma}\,\langle P^a(0)\vec{A}^b(x)\rangle. \tag{4.36}$$

In the case in which $\tau \neq 0$, this expression reduces to

$$\partial_\tau\int_{\mathcal{R}} d^3x\,\langle P^a(0)A_0(x)\rangle = -\int_{\partial\mathcal{R}} d\vec{\sigma}\,\langle P^a(0)\vec{A}^b(x)\rangle. \tag{4.37}$$

In Ref. [1], a useful relationship for the pseudoscalar temporal axial vector correlation function in the chiral limit is introduced:

$$\int d^3x\,\langle P^a(0)A_0^b(x)\rangle = \delta^{ab}\frac{\langle\psi\bar{\psi}\rangle}{2\beta}(\tau-\beta/2). \tag{4.38}$$

To continue our journey towards the construction of eq. (4.2), we require some spectral function definitions. In particular, we need the following spectral functions:

$$\delta^{ab}C_{PP}(\tau,\vec{k}) = \int d^3x\,e^{-i\vec{k}\vec{x}}\langle P^a(0)P^b(\tau)\rangle = \\ \delta^{ab}\int_0^\infty d\omega\,\rho_{PP}(\omega,\vec{k})\,\frac{\cosh(\omega\,(\beta/2-\tau))}{\sinh(\omega\beta/2)}, \tag{4.39}$$

$$\delta^{ab}C_{AP}(\tau,\vec{k}) = \int d^3x\,e^{-i\vec{k}\vec{x}}\langle P^a(0)A_0^b(\tau)\rangle = \\ \delta^{ab}\int_0^\infty d\omega\,\rho_{AP}(\omega,\vec{k})\,\frac{\sinh(\omega\,(\beta/2-\tau))}{\sinh(\omega\beta/2)}, \tag{4.40}$$

$$\delta^{ab}C_{AA}(\tau,\vec{k}) = \int d^3x\,e^{-i\vec{k}\vec{x}}\langle A_0^a(0)A_0^b(\tau)\rangle = \\ \delta^{ab}\int_0^\infty d\omega\,\rho_{AA}(\omega,\vec{k})\,\frac{\sinh(\omega\,(\beta/2-\tau))}{\sinh(\omega\beta/2)}. \tag{4.41}$$

The pseudoscalar Ward-Takahashi identity in eq. (4.24) can be applied to the spectral function definitions above to construct the following relationships:

$$2m_q\,\rho_{PP}(\omega,\vec{0}) = -\omega\,\rho_{AP}(\omega,\vec{0}), \tag{4.42}$$

$$\omega\,\rho_{AA}(\omega,\vec{0}) = 2m_q\,\rho_{AP}(\omega,\vec{0}). \tag{4.43}$$

Through the combination of both identities, a third identity can be generated,

$$\omega^2 \, \rho_{AA}(\omega, \vec{k} = \vec{0}) = -4m_q^2 \, \rho_{PP}(\omega, \vec{k} = \vec{0}) \tag{4.44}$$

In Ref. [1], the following chiral limit ρ_{AP} spectral function model is presented,

$$\rho_{AP}(\omega, \vec{k} = \vec{0}) = -\frac{\langle \psi\bar{\psi} \rangle}{2} \delta(\omega). \tag{4.45}$$

We would like to show that the spectral function defined above is, in fact, the exact spectral function for C_{AP} in the chiral limit and for no external momenta. To do so, we plug it inside eq. (4.40), which leads to the following expression

$$\int d^3x \, \langle P^a(0) A_0^b(x) \rangle = -\delta^{ab} \frac{\langle \psi\bar{\psi} \rangle}{2} \int_0^\infty d\omega \, \delta(\omega) \frac{\sinh(\omega \, (\beta/2 - \tau))}{\sinh(\omega\beta/2)}. \tag{4.46}$$

The right-hand side of the equation above can be integrated using the Dirac delta distribution and the following limit

$$\lim_{\omega \to 0} \frac{\sinh(\omega \, (\beta/2 - \tau))}{\sinh(\omega\beta/2)} = 2 \frac{(\beta/2 - \tau)}{\beta}. \tag{4.47}$$

As desired, after integration, we arrive to eq. (4.38).

The previous derivation allows us to learn that ρ_{AP} — see eq. (4.45) — couples to a massless excitation in the chiral limit. As a result, the same coupling is expected to be present in the limit of small momenta and quark masses. Moreover, eq (4.42) dictates that this coupling must also be present in the pseudoscalar correlation function. Using this information, a model for $\rho_{PP}(\omega, \vec{k})$ is proposed in Ref. [1]. The model is derived from previous results in hydrodynamic and zero temperature chiral expansion calculations [2–4]. The suggested model for ρ_{PP} is:

$$\rho_{PP}(\omega, \vec{k}) = \text{sign}(\omega) \, \Pi(k^2) \, \delta(\omega^2 - \omega_{\vec{k}}^2) + \ldots \tag{4.48}$$

where $\Pi(k^2)$ is a normalisation function, possibly dependent on the norm of the external momenta $k^2 = \vec{k} \cdot \vec{k}$.

In Ref. [1], the authors claim that, close to the chiral limit, $\omega_{\vec{k}}^2$, can be approximately described by a damped dispersion relation:

$$\omega_{\vec{k}}^2 \simeq u^2 \, (M_\pi^2 + \vec{k}^2) \tag{4.49}$$

This chiral model was first presented in Refs. [2, 3]. In the identity shown above, u^2 represents the pion velocity squared. At zero temperature u^2 is expected to be 1,

which recovers the standard relativistic energy-mass equivalence in which massless particles move at the speed of light. However, when the temperature increases, the pion velocity might decrease due to thermal effects. u^2 can be viewed as a damping coefficient that the medium imposes on the propagating pions. In this picture, we would expect u^2 to decrease as the temperature of the system increases. For more information about eq. (4.49), we refer to Ref. [1].

In addition, the authors of Ref. [1] derive an expression for the normalising function $\Pi(k^2)$ in eq. (4.48) in the limit of small quark masses and zero momenta. It reads

$$\Pi(k^2) = u^2 \, \frac{M_\pi^2 \langle \psi\bar{\psi}\rangle}{4m_q}. \tag{4.50}$$

This expression can be plugged into the definition of ρ_{PP} to produce the chiral limit pseudoscalar spectral function:

$$\rho_{PP}(\omega, \vec{k} = \vec{0}) = \text{sign}(\omega) u^2 \, \frac{m_\pi^2 \langle \psi\bar{\psi}\rangle}{4m_q} \, \delta(\omega^2 - \omega_{\vec{0}}^2) + \dots \tag{4.51}$$

From the pseudoscalar spectral function, ρ_{AP} and ρ_{AA} can be derived in the limit of small quark mass and zero external momenta through the spectral Ward-Takahashi identities present in eq. (4.42) and eq. (4.44) respectively:

$$\rho_{AP}(\omega, \vec{k} = \vec{0}) = -\frac{\omega_{\vec{0}} \langle \pi\bar{\psi}\rangle}{2} \, \delta(\omega^2 - \omega_{\vec{0}}^2) + \dots \tag{4.52}$$

$$\rho_{AA}(\omega, \vec{k} = \vec{0}) = -\text{sign}(\omega) \, m_q \langle \psi\bar{\psi}\rangle \, \delta(\omega^2 - \omega_{\vec{0}}^2) + \dots \tag{4.53}$$

We are finally in the position of constructing the pion velocity expression. To proceed, we make use of the following correlation function identity:

$$\partial_0^2 C(\tau) = \int_0^\infty d\omega \, \omega^2 \rho(\omega) \frac{\cosh(\omega \, (\beta/2 - \tau))}{\sinh(\omega\beta/2)}, \tag{4.54}$$

which can be easily derived from eq. (4.39). In the case in which $C(\tau) = C_{AA}(\tau)$, then the identity defined in eq. (4.44) produces the following equality:

$$\partial_0^2 C_{AA}(\tau) = -4m_q^2 C_{PP}(\tau). \tag{4.55}$$

The integral in the right-hand side of eq. (4.54) can be performed analytically using eq. (4.53). The result is

$$\partial_0^2 C_{AA}(\tau) = -\frac{m_q \langle \psi\bar{\psi}\rangle}{2\omega_{\vec{0}}} \, \omega_{\vec{0}}^2 \, \frac{\cosh(\omega_{\vec{0}} \, (\beta/2 - \tau))}{\sinh(\omega_{\vec{0}}\beta/2)}. \tag{4.56}$$

To compute the integral above, the following Dirac delta distribution property is required,

$$\delta(f(x)) = \sum_{x_i} \frac{\delta(x - x_i)}{|f'(x = x_i)|}, \tag{4.57}$$

where x_i represents one of the roots of $f(x)$ and f' is the first derivative of $f(x)$.

Employing eq. (4.56), the following ratio can be constructed,

$$\frac{\partial_0^2 C_{AA}(\tau)}{C_{AA}(\tau)} = \omega_{\vec{0}}. \tag{4.58}$$

In the limit of zero external momenta, $\omega_{\vec{0}} = u^2\, M_\pi$ from eq. (4.49). As a consequence, we arrive at the pion velocity expression:

$$u^2 = \frac{1}{M_\pi^2} \frac{\partial_0^2 C_{AA}(\tau, \vec{k} = \vec{0})}{C_{AA}(\tau, \vec{k} = \vec{0})} = -\frac{4m_q^2}{M_\pi^2} \frac{C_{PP}(\tau, \vec{k} = \vec{0})}{C_{AA}(\tau, \vec{k} = \vec{0})}. \tag{4.59}$$

In the last step, eq. (4.55) was employed. In Ref. [1], the authors set $\tau = \beta/2 = N_\tau/2$ in the pion velocity expression, leading to

$$u^2 = -\frac{4m_q^2}{M_\pi^2} \left. \frac{C_{PP}(\tau, \vec{k} = \vec{0})}{C_{AA}(\tau, \vec{k} = \vec{0})} \right|_{\tau = N_\tau/2}. \tag{4.60}$$

In principle, we are not bounded to estimate u^2 using correlation function evaluated at $\tau = \beta/2 = N_\tau/2$. The only requirement is to position the sink operator defining correlation functions used in the estimation of u^2 far from the source operator, usually located at $x = (\tau, x, y, z) = (0, 0, 0, 0)$, to avoid the presence of contact terms. For thermal correlation functions, this implies that $\tau \gg 0$. However, in thermal field theory, bosonic correlation functions are periodic in time with period N_τ. As a consequence, another source operator is placed at $\tau = N_\tau$. This implies that the maximum distance between sink and source operator in a thermal correlation function is obtained at $\tau = N_\tau$, which might explain why the pion velocity is only evaluated at $\tau = N_\tau/2$ in Ref. [1].

4.2 Lattice measurement of the pion velocity

Having derived the pion velocity expression, we are now in the position of estimating its temperature dependence in our lattice setup. Note that u^2 is only defined in the limit of small quark mass, which constrains the possible correlation functions

that could be employed in the estimation of u^2. In practical terms, estimating u^2 implies computing the ratio of the pseudoscalar correlation function, C_{PP}, and the axial vector correlation function, C_{AA}. Additionally, the pion mass M_π and the light quark mass m_q are required in the computation of u^2.

In order to study the temperature dependence of u^2, we use correlation functions measured at different temperatures. Details about the simulated correlation functions can be found in Appendix (A). In the estimation of u^2, the same masses m_q and M_π are employed at all temperatures: these masses correspond to our estimate at the lowest temperature available, generated in a lattice of temporal extent $N_\tau = 128$. By doing this, we ensure that the temperature dependence of u^2 is completely contained in the correlation functions. Additionally, we avoid possible inconsistencies in the definition of both m_q and M_π at different temperatures. Due to thermal effects, once the system has undergone a transition from the low temperature hadronic phase to the high temperature quark gluon plasma state, the pion mass might not be well-defined as thermal effects might dominate, and the spectrum of the theory might become continuous.

In our simulations, we produce two equivalent estimates of each correlation function: one simulated using local sources in the inversion of the quark propagator, and the other one using Gaussian smeared sources. More information about smearing can be found in Appendix (A). As a result, we can produce an estimate of u^2 employing both estimates of the same correlation function. To allow a fair comparison between both sources, we estimate m_q and M_π independently in each case.

From our analysis of mesonic ground state masses — see Chapter (3) — we know that the lowest temperature estimate ($T = 47$ MeV) of the pion mass, M_π, extracted using correlation functions computed from local sources is $M_\pi^{ll} = 245 \pm 2$ MeV. Additionally, the pion mass estimated from smeared correlation functions is $M_\pi^{ss} = 236.0 \pm 0.5$ MeV.

4.2.1 Measuring the quark mass m_q on the lattice

In addition to the pion mass, we require an estimate of the light quark mass m_q to correct the dimensions of u^2. We can estimate this light quark mass using the PCAC mass, defined as a rearrangement of eq. (4.20):

$$m_q^{PCAC} = \frac{1}{2} \frac{\langle \partial_\mu A_\mu^a(x) \hat{O}'(y) \rangle}{\langle P^a(x) \hat{O}'(y) \rangle}. \tag{4.61}$$

Applying eq. (4.29) to the equation above, we can express the PCAC mass as a function of a global derivative in the numerator:

$$m_q^{PCAC} = \frac{1}{2} \frac{\partial_\mu \langle A_\mu^a(x) \hat{O}'(y) \rangle}{\langle P^a(x) \hat{O}'(y) \rangle}. \tag{4.62}$$

In order to avoid inherently noisy correlation functions, we restrict ourselves to pseudoscalar densities in the source operator: $\hat{O}(y) = P^b(y)$, where the pseudoscalar density $P(y)$ is defined in eq. (4.19). As a result,

$$m_q^{PCAC} = \frac{\delta^{ab}}{2} \frac{\partial_\mu \langle A_\mu^a(x) P^b(y) \rangle}{\langle P^a(x) P^b(y) \rangle}. \tag{4.63}$$

As we are interested in thermal correlation functions, we need to apply a Fourier transform over the spatial coordinates of each correlation function in eq. (4.19). A non-trivial transformation is encountered in the numerator, as it contains a derivative operator. Nevertheless, it can be easily manipulated to obtain:

$$\begin{aligned} &\lim_{\vec{k} \to \vec{0}} \int \frac{d^3x}{(2\pi)^3} e^{-i\vec{k}\vec{x}} \partial_\mu \langle A_\mu^a(x) P^b(y) \rangle = \\ &\int \frac{d^3x}{(2\pi)^3} \Big[\partial_0 \langle A_0^a(x) P^b(y) \rangle + \vec{\nabla} \langle \vec{A}^a(x) P^b(y) \rangle \Big]. \end{aligned} \tag{4.64}$$

The second term in the right-hand side vanishes due to Gauss's theorem: all correlation functions decay to zero in the boundary of the space-time. Note that we are projecting our correlation functions to zero external momenta: $\vec{k} = \vec{0}$. As a result, the space-integrated correlation functions will only be functions of the Euclidean time τ, which allows us to write

$$m_q^{PCAC} = \frac{\delta^{ab}}{2} \frac{\partial_0 \langle A_0^a(\tau, \vec{k} = \vec{0}) P^b(0, \vec{k} = \vec{0}) \rangle}{\langle P^a(\tau, \vec{k} = \vec{0}) P^b(0, \vec{k} = \vec{0}) \rangle} = \frac{\delta^{ab}}{2} \frac{\partial_0 C_{AP}(\tau)}{C_{PP}(\tau)}, \tag{4.65}$$

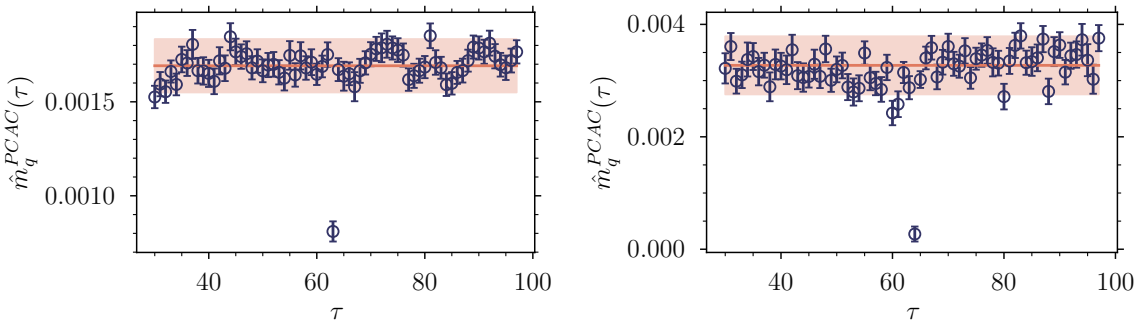
where the definitions of C_{AP} and C_{PP} in eq. (4.40) and (4.39) are used respectively.

In order to estimate m_q^{PCAC}, eq. (4.65) is evaluated on the lattice using correlation functions measured at the lowest temperature available, which in our case implies that we use a lattice of temporal extent $N_\tau = 128$: $T = 47$ MeV. In principle, the PCAC mass is defined at a variety of different Euclidean times τ. However, to avoid contact terms, we exclude all Euclidean times close to the source operator, located at $\tau = 0$ and $\tau = N_\tau$, due to the periodicity of bosonic correlation functions. As

a result, in our analysis, we extract the PCAC mass at all Euclidean times from $\tau = 30$ to $\tau = 98$. At all Euclidean times selected, we are able to produce an estimate of the PCAC mass, $\hat{m}_q^{PCAC}(\tau)$. Assuming that each $\hat{m}_q^{PCAC}(\tau)$ represents a random sample of the population PCAC mass, then the median of the sample corresponds to an estimate of m_q^{PCAC} independent of τ. The standard error of the median can be approximated using bootstrap. The median is used as a measure of central tendency to avoid the impact of outliers. This analysis is performed for both type of sources using local and smeared correlation functions. The results are presented in Figure (4.1), and they yield the following quark mass estimates in physical units:

$$\begin{aligned} &\text{Local sources:} && \hat{M}_q^{PCAC} = 10.28 \pm 0.02 \text{ MeV}, \\ &\text{Smeared sources:} && \hat{M}_q^{PCAC} = 19.89 \pm 0.04 \text{ MeV}. \end{aligned}$$

The time derivative present in eq. (4.65) is estimated numerically using the forward finite differences operator, which explains the artefact located at the middle point of the lattice, where a bend in the correlation function data is present due to the symmetry of bosonic correlation functions. In the computation of the PCAC mass in physical units, the systematic uncertainty generated in the scale setting of our simulations is taken into account.



(a) Local sources **(b)** Smeared sources

Figure 4.1: Estimates of the degenerate light quark masses $\hat{m}_q^{PCAC}$ for both type of sources available in the simulations. The orange line corresponds to the median of all samples, while the orange shaded band covers the standard error of the median, approximated using bootstrap. The results are expressed in lattice units.

4.2.2 Renormalisation of the pion velocity

Due to the fact that the pion velocity is constructed using a ratio of two distinct correlation functions, then, in principle, it should be renormalised. The pseudoscalar operator should be rescaled by its specific renormalisation constant Z_P, while the axial vector should use its appropriate constant, Z_A. As a result, the renormalised version of the pion velocity reads,

$$\bar{u}^2 = -\frac{Z_P^2}{Z_A^2}\frac{4m_q^2}{M_\pi^2}\left.\frac{C_{PP}(\tau,\vec{k}=\vec{0})}{C_{AA}(\tau,\vec{k}=\vec{0})}\right|_{\tau=N_\tau/2}. \tag{4.66}$$

The same principle can be applied to our estimate of the quark mass, the PCAC mass:

$$\bar{m}_q^{PCAC} = \frac{Z_A Z_P}{Z_P^2} m_q^{PCAC}. \tag{4.67}$$

Substituting m_q in $\bar{u}^2$ with $\bar{m}_q^{PCAC}$ leads to

$$\bar{u}^2 = \left(\frac{Z_P^2}{Z_A^2}\right)\left(\frac{Z_A^2 Z_P^2}{Z_P^2 Z_P^2}\right)u^2, \tag{4.68}$$

which is equal to

$$\bar{u}^2 = u^2, \tag{4.69}$$

as long as the same density operators are applied in the sink and source operators. For example, employing local operators in the sink and smeared operators in the source breaks this condition, as smeared and local operators could have different renormalisation constants for the same operator.

As a result, the pion velocity does not need to be renormalised. However, the PCAC masses presented in Figure (4.2) should be. It is worth stressing that no renormalisation constants are computed in our simulations. A correct renormalisation of the PCAC masses might explain the drastic differences between local and smeared estimates encountered in Figure (4.1).

4.2.3 The pion velocity on the lattice

Using the estimated PCAC quark mass and the pion mass for each source, we can finally compute the pion velocity at all temperatures available. The temperature dependence of the pion velocity is shown in Figure (4.2). For each temperature and source available, an estimate of the pion velocity is extracted. The lowest temperature estimate of the quark mass and pion mass are employed in all temperatures

analysed. The standard error of each measurement is estimated using Monte-Carlo error propagation.

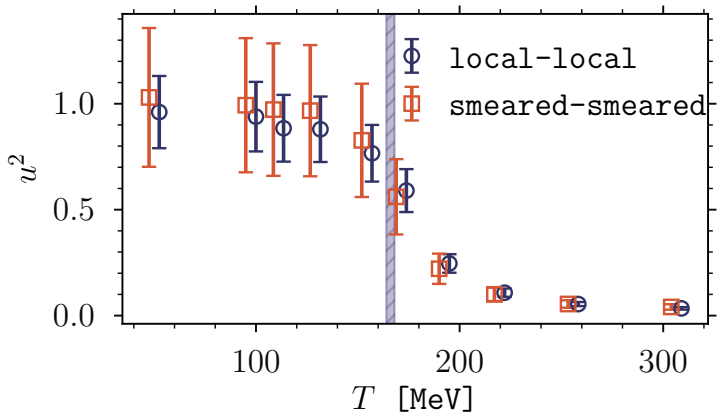


Figure 4.2: Estimate of the pion velocity squared u^2 as a function of the temperature for both sources. The green vertical band corresponds to the pseudocritical temperature T_c of the system. The presence of a right-shift in the temperature for local sources is merely visual.

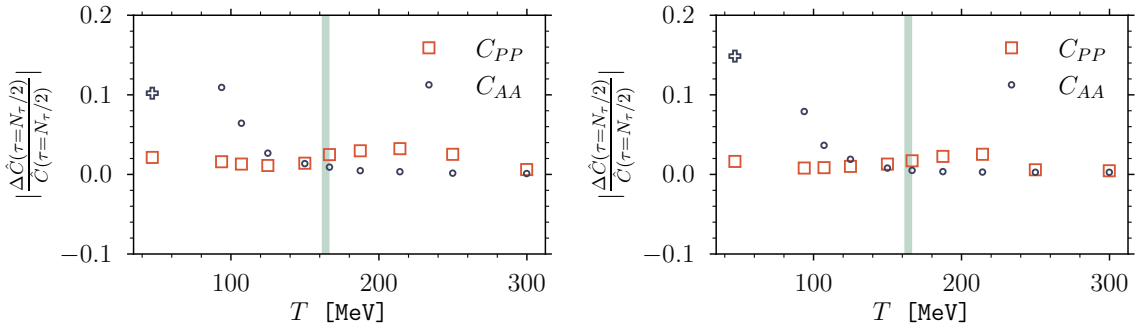
The results in Figure (4.2) shows equivalent trends for both sources at all temperatures. The pion velocity estimates can be divided into two clearly distinct categories: one composed by all estimates whose temperature is lower than the pseudocritical temperature of the system, located at $T_c = 166 \pm 2$ MeV; and another group generated by all pion velocities measured at temperatures above T_c. The first group roughly corresponds to the confining hadronic phase of QCD, while the second group represents the QCD quark-gluon plasma state. In the hadronic phase, the pion velocity barely depends on the temperature; this is a constant trend present in some physical quantities measured in the hadronic phase of QCD. Moreover, in this region, the pion velocity agrees with the expected zero temperature value: $u^2 = 1$. However, once the temperature increases, the pion velocity departs from its zero temperature value. It is close to the critical temperature of the system where the pion velocity drastically changes. As far as the data shows, u^2 seems to stabilise at high temperature:

$$\lim_{T \to \infty} u^2(T) \simeq 0.1. \tag{4.70}$$

An analysis of the pion velocity at higher temperatures would allow a direct test of this limit.

It is worth mentioning that our results agree with the ones present in Ref. [1], although we have access to higher temperatures. The uncertainties of their low temperature estimates are lower than ours, this is a consequence of our noisy low temperature axial vector correlation functions.

In Figure (4.2), low temperature estimates seem to have larger uncertainties than the high temperature measurements. However, it is generally expected a larger uncertainty in the high temperature estimates. As a consequence of this unexpected behaviour, an analysis of the source of this difference is performed. Note that the source of the difference must be in the lattice estimates of the thermal correlation functions used in the computation of u^2, that is, either in $\hat{C}_{AA}$ or in $\hat{C}_{PP}$. The effect cannot be produced by the estimates of M_π and m_q as they are independent of the temperature: the same values are employed at all temperatures.



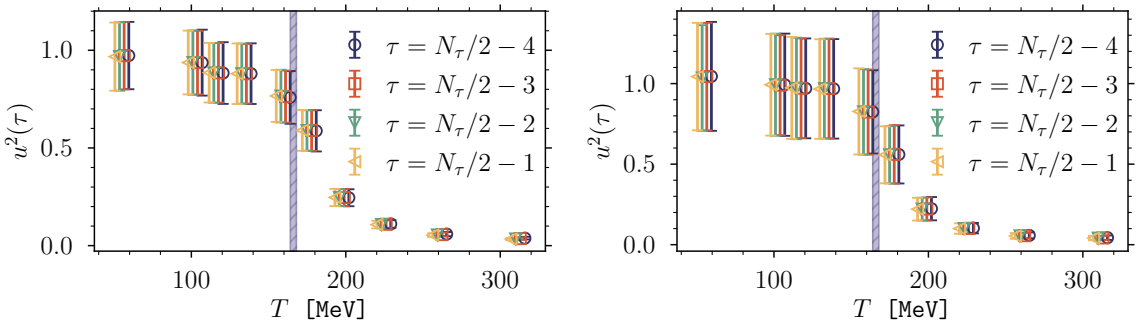
(a) Local sources **(b)** Smeared sources

Figure 4.3: Relative uncertainties in the lattice estimates of the C_{AA} and C_{PP} correlation functions measured at the middle point of the lattice in the temporal direction. Results are presented for all temperatures available. The lowest temperature estimate of the AA correlation function is divided by 10 as its relative uncertainty is considerably higher than the rest.

Figure (4.3) contains the relative uncertainty of the lattice estimates of both correlation functions, AA and PP, as a function of the temperature. The relative uncertainty is measured at the middle point of the lattice in the temporal direction and for both sources. The results show that the relative uncertainty of the AA correlation function decreases with the temperature, which directly impacts the estimation of the ratio between both correlation functions involved in the computation of the pion velocity. In our fixed-scale simulation, the relative uncertainty of axial vector correlation functions tends to decrease with the temperature. This suggests that the correlation function signal might be lost as the distance between sink and source

operators increases. This problem directly impacts the analysis of low temperature axial-vector properties, as $T^{-1} \sim N_\tau$.

Additionally, as stated in the derivation of the pion velocity, there is no constraint on the Euclidean time τ at which the pion velocity is measured as long as it is far from the source. Figure (4.4) tests this statement by extracting u^2 evaluating the correlation functions at different Euclidean times. Note that all used Euclidean times should be close to the middle point of the temporal direction of the lattice to avoid the presence of contact terms. The results are in agreement at all temperatures and for both types of sources.



(a) Local sources **(b)** Smeared sources

Figure 4.4: Temperature dependence of the estimation of the pion velocity at different Euclidean times τ for both type of sources used in the calculation: local and smeared. The shift in temperature for different τ is merely visual.

As we measure the same pion velocity at different Euclidean times, we could collect all the estimates into a single sample. From this combined sample, one can compute a final estimate of the target population parameter, u^2. This can be seen as data-augmentation technique in which the amount of information about the population parameter is increased. Such a procedure could lead to lower uncertainties in our estimates.

4.2.4 Conclusions

The pion velocity derived in eq. (4.2) corresponds to an effective theory calculation, which implies that it only works in the chiral limit of light quarks, and with small external momenta $\vec{k}$. However, the pion velocity can be explored in a complete thermal QCD lattice calculation through the study of the dispersion relation of the pseudoscalar correlation function at different temperatures. The dispersion relationship can

be analysed at a given temperature by studying the dependence of the ground state energy $E_0(M_\pi, \vec{k})$ with the external momenta $\vec{k}$ of the system. To do so, several correlation functions must be produce at the same temperature with different external momenta. For each correlation function, the ground state energy E_0 must be extracted, for example, through the analysis of the ground state contribution to a given pseudoscalar correlation function using the spectral decomposition defined in eq. (1.45). The inclusion of external momenta on the lattice can be done employing Fourier modes in the computation of thermal correlation functions or by applying the so-called partially-twisted boundary conditions [5, 6]; this second option allows the inclusion of arbitrary units of external momenta in the system.

In this first-principles analysis, a collection of different ground state energies would be produced at each temperature: $E_0(M_\pi, \vec{k}; T)$. In order to study the thermal dependence of the pion velocity, an independent estimate of the pion velocity at each temperature can be extracted by fitting the measured ground state energies to the pion dispersion relation model proposed in eq. (4.49):

$$E_0(M_\pi, \vec{k})^2 = u^2 \, (M_\pi^2 + \vec{k}). \tag{4.71}$$

For each temperature, we would estimate the pion velocity u^2. This analysis suffers from similar problems to ones encountered in the study of the spectrum of QCD from thermal correlation functions: at high temperature, thermal effects dominate, which obstructs the definition and computation of the ground state energy. As a consequence, the proposed analysis might be difficult to apply to really large temperatures: $T \gg T_c$. However, this method is not based on effective field theory assumptions. As a result, it can be used to estimate the thermal dependence of the velocity of different mesonic states, such as kaons (ss) or D-mesons (uc). Furthermore, this procedure could also be easily extended for baryonic states.

Chapter 4. References

[1] Bastian B. Brandt et al. "Chiral dynamics in the low-temperature phase of QCD". In: *Phys. Rev. D* 90.5 (2014), p. 054509. DOI: 10.1103/PhysRevD.90.054509. arXiv: 1406.5602 [hep-lat].

[2] D. T. Son and Misha A. Stephanov. "Real time pion propagation in finite temperature QCD". In: *Phys. Rev. D* 66 (2002), p. 076011. DOI: 10.1103/PhysRevD.66.076011. arXiv: hep-ph/0204226.

[3] D. T. Son and Misha A. Stephanov. "Pion propagation near the QCD chiral phase transition". In: *Phys. Rev. Lett.* 88 (2002), p. 202302. DOI: 10.1103/PhysRevLett.88.202302. arXiv: hep-ph/0111100.

[4] A. Schenk. "Pion propagation at finite temperature". In: *Phys. Rev. D* 47 (1993), pp. 5138–5155. DOI: 10.1103/PhysRevD.47.5138.

[5] Jonathan Flynn et al. "Partially twisted boundary conditions in lattice simulations". In: *PoS* LAT2005 (2006). Ed. by Christopher Michael, p. 352. DOI: 10.22323/1.020.0352. arXiv: hep-lat/0509093.

[6] J. M. Flynn, A. Juttner and C. T. Sachrajda. "A Numerical study of partially twisted boundary conditions". In: *Phys. Lett. B* 632 (2006), pp. 313–318. DOI: 10.1016/j.physletb.2005.10.042. arXiv: hep-lat/0506016.

Chapter 5

Introduction to neural networks

During the last decade, the theoretical developments and practical applications of neural networks to both academic and industry problems have witnessed a large increase. Their outstanding impact in most scientific fields can be understood in terms of their adaptability, flexibility, expressiveness, and computational efficiency. Additionally, the fact that collecting, manipulating, and storing large amounts of data has become easier and more affordable in the last few years generates an environment in which neural networks can thrive. As a result, nowadays, neural network models are applied to a variety of complex tasks, such as: standard classification and regression problems [1, 2], reinforcement learning [3, 4], graph theory [5], and natural language processing [6, 7]. The list of possible tasks to which neural networks can be applied is continuously growing as more research is performed to understand, and enhance, their inherent properties. In addition, the increase in computational power of graphical processing units (GPUs), and, more recently, the arrival of dedicated tensor processing units (TPUs), combined with the availability of highly optimised open-source software libraries [8–10] allows an almost straightforward application of most state-of-the-art neural network models to computationally demanding tasks.

This chapter is divided into two main sections. The first one contains a short introduction to the field of neural networks for which no previous knowledge about machine learning nor neural networks is required. In this section, we also discuss how to efficiently train neural network models. The second section contains a small introduction to convolutional neural networks, which are employed in Chapter (6).

5.1 Standard neural networks

The roots of neural network models can be traced to the decade of the 1950s, when researchers were trying to mathematically model decision-making. One of the simplest decision-making models is the perceptron [11], which is a simplified model of the behaviour of neurons in the brain. The perceptron unit proposes a deterministic output, also called the neuron's action, depending on some input variables.

Figure (5.1) contains a diagram showing a perceptron unit defined on an input of 4 dimensions, labelled $x = (x_1, x_2, x_3, x_4)$, and a 1-dimensional output, labelled y. Although the output in Figure (5.1) is 1-dimensional, perceptrons are also able to handle multidimensional outputs.

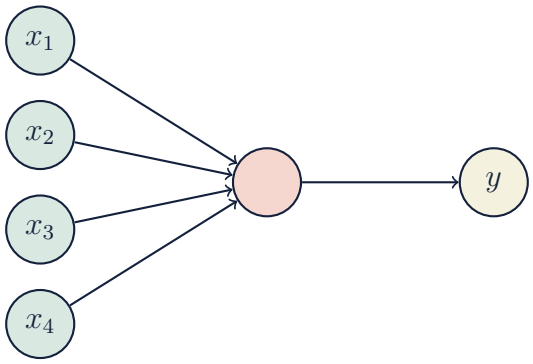


Figure 5.1: Diagram representing a perceptron unit. A vector of inputs $x = (x_1, x_2, x_3, x_4)$ is processed by the perceptron in order to produce an output signal y.

To see how the perceptron unit can be used to automate decision making, we focus on a particular task. Imagine that we wanted to decide whether we should go to the office or work from home on one particular morning. In this simple case, the decision space is binary: either we go, corresponding to an *activated* output, $y = 1$; or we stay at home, corresponding to a *turned off* outcome, $y = 0$. We assume that the process of making a decision is not irrational: we deliberately make a decision based on some reasonable conditions, and, moreover, the results are consistent for a fixed input. In addition, suppose we knew that only 4 independent variables affect our final decision. For instance, those 4 variables could represent: the distance between our home and the office; the weather that morning; the amount of work we need to do that precise day; and the current risk level of COVID-19 in our area. Note that all input variables should be mappable to a numeric value; this is called embedding a variable.

It is clear that each input variable should have a different impact in our final decision. For example, a high risk level of COVID-19 should encourage us to stay at home independently of all the other conditions. To model the importance of each input, we use some weights W. Each independent weight acts on a particular input variable, weighting its relevance in the final decision. The dimensions of the weights are not important, as we are not trying to measure a physically relevant quantity. However, we should normalise the inputs in order to avoid including artificial numerical bias towards any input variable. The scales of the inputs should be comparable so as not to artificially benefit any input variable. This is a constant requirement in machine learning: inputs should always be scaled appropriately.

Given some inputs $x = (x_1, \ldots, x_n)$ and some weights $W = (w_1, \ldots, w_n)$, the simplest non-trivial mathematical model that can be used to make decisions is

$$y = W^T \circ x, \tag{5.1}$$

where W^T denotes the transpose of W, and $\circ$ represents the standard matrix multiplication operator. The output of eq. (5.1) is a real number, but in our particular task, the decision space is binary. In order to transform the real output into a binary variable, we can introduce a threshold b, which implies that the mathematical model of the perceptron is transformed to

$$y = \begin{cases} 0 & \text{if } W^T \circ x < b \\ 1 & \text{if } W^T \circ x \geq b \end{cases} \tag{5.2}$$

The threshold is called the bias of the model, as it represents our own personal bias in the task in question: some people might enjoy going to the office more than others, so they will accept smaller values of y as an activated output. The bias defines the boundary that dictates whether $W^T \circ x$ is regarded as an activated or turned off outcome. The standard way of modelling a binary perceptron unit with a scalar bias b is:

$$y = \begin{cases} 0 & \text{if } (W^T \circ x - b) < 0 \\ 1 & \text{if } (W^T \circ x - b) \geq 0 \end{cases} \tag{5.3}$$

Equation (5.3) is the mathematical definition of the perceptron unit. In the case in which the output space is not binary, then the following mathematical model can be used:

$$y = W^T \circ x + b. \tag{5.4}$$

By itself, the perceptron unit is not too useful, as a random set of weights and bias values are unlikely to generate reasonable outputs for the particular task to which the model is applied. In order to make the perceptron model applicable to

real situations, its parameters need to be tuned accordingly, which implies that some kind of regression needs to be applied to our model. Before discussing how to tune the parameters of the model, we need to assume that our particular task, abstractly represented with the label T, contains a true underlying population mapping t that encodes the fundamental information needed to solve the problem: the mapping connects input variables x with their expected output variables y. In the simplest case, the output variables represent binary choices, but more complex output spaces can also be modelled by eq. (5.4). Mathematically, the mapping t is defined as the following morphism:

$$t : \mathcal{X} \to \mathcal{Y}, \tag{5.5}$$

where $\mathcal{X}$ is the space of all possible inputs x, and $\mathcal{Y}$ represents the space of target outputs. As a result, $y = t(x)$. In this context, the perceptron model defined in eq. (5.4) serves as a parametric model for the mapping t, which we denote with the label $\hat{t}_W$. It is common to encapsulate both weights and bias in a model under the same label, W. Additionally, the terms learnable parameters and weights are usually employed to refer to all the tunable parameters in a particular model. A particular choice of weights and bias in a model is usually called a configuration.

The goal is then to find the values of W and b that make the perceptron unit approximate the mapping t: $\hat{t}_W \simeq t$. This would enable us to process any input x belonging to $\mathcal{X}$ in order to produce its expected output: the model would be making predictions. A priori, we do not know the appropriate configuration. However, we can try finding it using real examples of the mapping. A collection of inputs x_T whose corresponding outputs $y_T = t(x_T)$ are known in advance is known as a training set. The training pairs can be collected from experiments, simulations or other sources. Note that the training set should contain a large enough collection of pairs (x_T, y_T) such that it represents a reliable approximation of the target mapping t. Once the training set is defined, we can try to find the model configuration that reproduces the training set with large accuracy, that is, the configuration of the model that holds

$$y_T \simeq \hat{t}_W(x_T), \quad \text{for all} \quad x_T. \tag{5.6}$$

However, we should be cautious, as the model might overfit the training dataset, that is, it can learn the values of W and b that mimic the training set heuristically without learning the fundamental features of the mapping. Overfitting is common when the model contains a huge number of tunable parameters. Nowadays, overfitting can be controlled by numerous techniques [12].

Using labelled training pairs (x_T, y_T) in order to find the model configuration that best approximates the mapping t is usually referred to as a supervised machine

learning problem. In contrast, learning the correct model configuration from just the input data x_T is called unsupervised learning. In this document, we only discuss the framework of supervised learning.

Although the perceptron model can be employed in some simple tasks, it has a fundamental problem: it can only be applied to problems whose output spaces $/mathcalY$ are linearly separable. Consequently, problems with non-linear target mappings cannot be solved with our current model. To circumvent this problem, we can add some artificial non-linearity to the model in eq. (5.4):

$$y = f(W^T \circ x + b). \tag{5.7}$$

The function f in eq. (5.7) is an element-wise non-linear function, called activation function. In this context, element-wise means that the function acts independently on each input variable:

$$f(x) = f(x_1, x_2) = [f(x_1), f(x_2)]. \tag{5.8}$$

Some commonly used activation functions are: sigmoid functions, hyperbolic tangents and rectified linear units (ReLU). A comprehensive list of common activation functions can be found in Ref. [8].

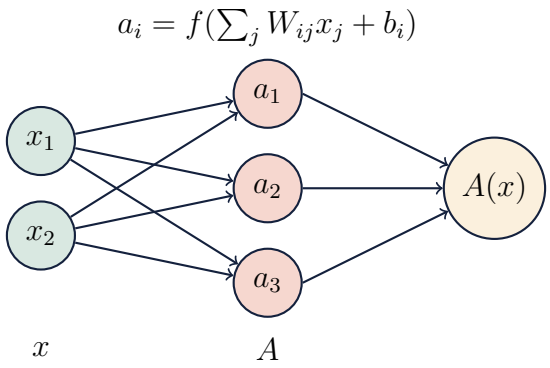


Figure 5.2: Diagram representing a layer composed by stacking 3 neurons. The layer takes 2-dimensional inputs, and produces a 1-dimensional output. The output of each neuron is fed to an activation function f, shared among all 3 neurons.

The non-linear perceptron model is still not flexible enough to learn any desired target mapping. In order to generate more flexible models, we can stack multiple perceptron units; a collection of non-linear perceptron units is usually called a neural network layer, and each perceptron model in a layer is usually called a neuron. In

general, all neurons in a layer share the same input, but each has its own independent set of weights and biases. Figure (5.2) contains a diagram representing a neural network layer composed by 3 neurons.

Neurons are non-linear models, which implies that their output is usually passed through an element-wise activation function, typically shared among all neurons forming the layer. The mathematical model of a simple linear neural network containing N_n neurons is similar to that of the non-linear perceptron unit defined in eq. (5.7), and reads

$$A = f(Z) = f(x \circ W + b). \tag{5.9}$$

In a neural network layer, W represents a matrix of dimensions $\dim[W] = (N_x, N_n)$, the input vector x has dimensions $\dim[x] = (1, N_x)$, and b holds $\dim[b] = (1, N_n)$. As a result, the output A, sometimes called the layer activation, has dimensions $(1, N_n)$.

The linear neural network layer defined in eq. (5.9) is called a feedforward layer. The adjective feedforward implies that the information only flows from input to output: the layer does not contain any loops nor backwards connections between neurons. In principle, non-feedforward layers can also be built and employed, however, training them tends to be difficult. This chapter only discusses feedforward neural networks.

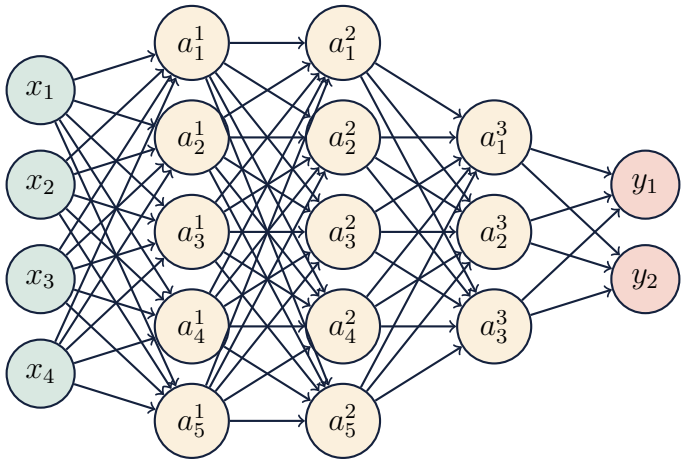


Figure 5.3: Diagram representing a feedforward neural network composed by 4 layers; three hidden layers and one output layer. Each of the 4 layers contains an independent set of weights, with dimensions (N_n^{l-1}, N_n^l), and an independent set of bias, with dimensions (N_n^l). Each layer processes its input according to eq. (5.9).

A complete feedforward linear neural network can be built by connecting N_L different layers. Each layer is labelled with an index l and contains N_n^l independent

neurons; the optimal number of neurons in each layer is not known in advance, therefore, it be tuned for the task in question. The output of each layer is passed to an activation function f^l. The activation function can be different for each layer in the network. In a feedforward network composed by N_L layers, there are $N_L - 1$ hidden layers whose output is not visible. The last layer produces the output of the neural network, $y = \hat{t}_W(x)$. A neural network can be viewed as a set of non-linear transformation applied sequentially over the input space. A diagram showing a complete feedforward neural network with 4 layers can be found in Figure (5.3).

Feedforward neural networks composed by several connected layers are known to be universal approximators, that is, they are able to approximate any arbitrary mapping with any desired precision; see Ref. [13] and references therein for a proof of this property. However, note that different architectures might be able to exploit the subtleties in the data better than others in order to speed up the convergence of the model to the correct solution. Furthermore, perfect convergence is usually hindered by the unavoidable uncertainties present in the data, the lack of adaptability of the model employed, or problems related to finding the correct solution numerically.

In a neural network composed by multiple layers, each sequential layer in the network helps the model extract the relevant features of the mapping, which implies that, in general, the deeper the network, the better. In the field of neural networks, the adjective deep implies that the model contains numerous connected layers. Deep neural networks tend to be computationally expensive to train and data intensive, as they contain huge number of tunable parameters. Additionally, training deep neural is inherently difficult as they suffer the so-called vanishing and exploding gradients problems [14]. Nevertheless, deep neural networks are nowadays fruitfully applied to a variety of complex tasks [15, 16].

The neural network mathematical model defined in eq. (5.9) allows processing several input examples simultaneously. To see this, imagine that we had a training dataset composed by N_b input pairs, x. We can stack all the inputs into a matrix of dimensions (N_b, N_x), and directly apply eq. (5.9) on the input matrix. In this case, the output of a neural network layer is another matrix of dimensions (N_b, N_n), where N_n is the number of neurons in the layer.

The topology of a neural network is called the network architecture. Different architectures can be generated by varying the number of hidden layers, the number of neurons in each layer, and the activation functions. Moreover, complex architectures can be built by modifying the mathematical model in eq. (5.9), or the connections between layers.

5.1.1 Training neural networks

We are now in the position of introducing the standard way of training neural networks to solve real-world problems. As stated before, training is just a synonym for learning the correct neural network configuration so that a target mapping t is correctly reproduced. Equivalent to the perceptron model, the neural network acts as a parametrisation of target mapping: $\hat{t}_W$.

Understanding the properties of the particular task to which a neural network is being applied allows us to decide the best architecture for the problem in question. However, it is important to note that, a priori, we cannot know whether a particular architecture will perform better or worse than other architectures on a given task. As a result, it is advisable to explore the available literature in order to find benchmarks on similar tasks before training a possibly non-optimal architecture.

In our supervised learning framework, we assume that the input space $\mathcal{X}$ and the output space $\mathcal{Y}$ can be treated as probability spaces. As a result, the training set corresponds to a sample of both input and output spaces. The network will learn from those samples to find a configuration that best approximates the target mapping t. In order to learn the correct mapping from the data, we use a penalty-based approach: the network will receive a penalty every time it generates a mapping $\hat{t}_W(x)$ that does not lead to the correct expected outcome, that is, y_T. This penalty is usually measured in terms of a loss/cost function:

$$\mathcal{L} = \mathcal{L}(y_T = t(x_T), \hat{y}_T = \hat{t}_W(x_T)). \tag{5.10}$$

Loss functions are scalar functions that measure some kind of distance between the target mapping, $t(x)$, and the current neural network approximation of that mapping, $\hat{t}_W(x)$. Several loss functions are available depending on the task in question. For example, in classification problems, the most common loss function is the so-called cross-entropy loss, which is derived from information theory, and measures the distance between two probability distribution functions. In contrast, in regression problems, the mean squared error (MSE) function is typically employed, although other options are also available, such as the $L1$-loss function or the Huber loss function. A list of common loss functions and their definitions can be found in Ref. [8].

Typically, training a neural network on a particular task using supervised learning requires several prior choices, such as: the network architecture, the loss function, and the initial values of the learnable parameters. Additionally, the values of some other hyperparameters might need to be chosen appropriately.

In our supervised framework, the optimal values of the learnable parameters,

labelled $\tilde{W}$, are those that fulfil the following condition,

$$\tilde{W} = \underset{W}{\operatorname{argmin}}\, \mathcal{L}(y_T = t(x_T), \hat{Y}_T = \hat{t}_W(x_T)). \tag{5.11}$$

In the equation above argmin means the set of weights at which the loss function has a minimum values. The optimal parameters depend on the loss function used, and they represent the configuration that makes the neural network mapping approximate the true target mapping according to $\mathcal{L}$. In this context, $\tilde{W}$ is found by minimising the loss function on the training set.

Finding the configuration that minimises eq. (5.11) is equivalent to finding the points at which the first derivative of $\mathcal{L}$ with respect to W is zero:

$$\left.\frac{\partial \mathcal{L}}{\partial w}\right|_{w=\tilde{w}} = 0, \quad \text{for all} \quad w \in W. \tag{5.12}$$

In most cases, eq. (5.12) represents an analytically intractable system of equations. As a result, numerical minimisation methods are required. One of the simplest methods available to find a solution to eq. (5.12) is the so-called gradient descent algorithm: a first-order numerical algorithm employed to find the minima of a scalar function $\mathcal{L}$. In gradient descent, the arguments of the function are updated iteratively until a minimum is found. The found solution might correspond to a local minimum; convergence to a global minimum is not ensured in gradient descent. As in any numerical minimisation algorithm, initial values for the function arguments are required; the initial values might affect the performance of the algorithm.

In gradient descent, at each iteration, we update each function argument depending on the variation of $\mathcal{L}$ when each individual learnable parameter w is infinitesimally modified. The variation of $\mathcal{L}$ is approximated using the Taylor expansion of $\mathcal{L}$ at first order, by defining

$$\Delta\mathcal{L} = \mathcal{L}(w + \delta w) - \mathcal{L}(w) \simeq \vec{\nabla}\mathcal{L}(w)\big|_{w=w+\delta x}\delta w, \tag{5.13}$$

where $\vec{\nabla}$ is the gradient operator, and δw is usually called the learning rate, typically labelled α.

For convex functions, the minimum of $\mathcal{L}$ is found when $\Delta\mathcal{L}$ is close to zero. As a consequence, if we update each weight iteratively using

$$w^{i+1} = w^i - \alpha\, \vec{\nabla}\mathcal{L}\big|_{w^i}, \tag{5.14}$$

eventually, we will arrive at a local minimum of $\mathcal{L}$, as gradient descent updates the function arguments in the opposite direction in which $\mathcal{L}$ grows.

Note that each weight in the neural network needs to be updated at every iteration. Consequently, the derivatives of the loss function with respect to all learnable parameters in the model are required. We can compute them using the chain rule. For instance, the derivative of $\mathcal{L}$ with respect to a given weight located in the layer l of a model with L total layers can be computed using

$$\frac{\partial \mathcal{L}}{\partial w} = \frac{\partial \mathcal{L}}{\partial \hat{t}_W(x_T)} \frac{\partial \hat{t}_W(x_T)}{\partial A^{L-1}} \frac{\partial A^{L-1}}{\partial Z^{L-1}} \frac{\partial Z^{L-1}}{\partial A^{L-2}} \cdots \frac{\partial Z^l}{\partial w}, \tag{5.15}$$

where w^l represents a particular parameter of the model, Z^l is the affine transformation at the layer l, defined as

$$Z^l = W^l \circ A^{l-1} + B^l, \tag{5.16}$$

and $A^l = f^l(Z^l)$ is its non-linear activation. The derivatives are easily computable at every stage as we know the functional forms of both Z and A at every layer. At each training iteration, each parameter in the model is updated according to eq. (5.14):

$$w_{i+1} = w_i - \alpha \left. \frac{\partial \mathcal{L}}{\partial w} \right|_{w_i}. \tag{5.17}$$

In practice, the derivatives are automatically computed using the framework of automatic differentiation (AD) [17, 18].

Nowadays, there exist a wide range of variations of the standard gradient descent algorithm with enhanced properties. Some of those variations are: the stochastic gradient descent algorithm (SGD), momentum-based gradient descent, or the ADAM algorithm. More information about gradient descent and some of its variations can be found in Ref. [19]. Additionally, note that eq. (5.13) is a first-order approximation of the total variation. Some algorithms use second order approximations of $\Delta\mathcal{L}$, which require the Hessian operator. The Hessian operator is the matrix containing all the second-order derivatives of $\mathcal{L}$ with respect to W. As a result, second-order optimisation algorithms tend to be computationally expensive.

The standard algorithm used to train neural networks is called forward/backward propagation. The algorithm is divided into two different stages: the forward propagation, and the backward propagation. In the first stage, the network produces its current estimate of the population mapping t for some training inputs x_T. After computing the model output, $\hat{y} = \hat{t}_W(x_T)$, we measure the distance between $\hat{y}$ and the expected true training outputs y_T according to a previously chosen loss function. Once the loss function is estimated on the training set, we proceed to the second stage: the backward propagation. In this part of the algorithm, the first derivatives

of the loss function with respect to all learnable parameters in the model are calculated. Once the gradients are computed, we perform an update of all parameters in the model according to eq. (5.17).

Due to the fact that in the first stage of the algorithm the information flows from input to output, and in the second stage the information flows backwards, training a neural network is usually represented as a continuous loop: the forward-backward training loop. Figure (5.4) contains a diagram representing the standard iterative algorithm used to train neural networks.

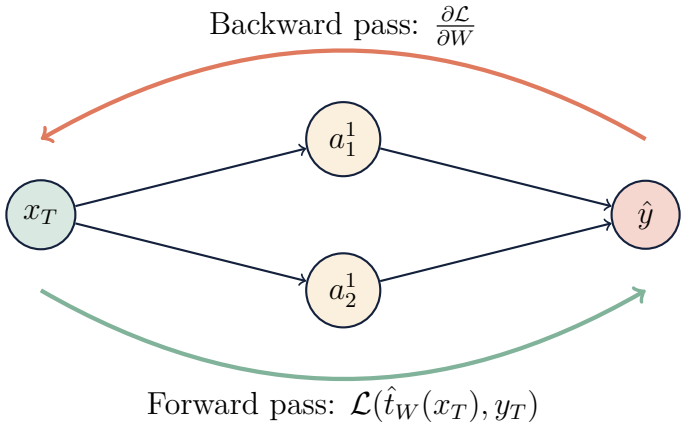


Figure 5.4: Diagram showing the standard forward-backward propagation algorithm used to train a neural network. In the forward propagation step, the network produces the current estimate of the population mapping $\hat{y} = \hat{t}_W(x_T)$. This estimate is employed to compute the empirical loss function at the current iteration using the expected training output, y_T. In the back-propagation step, the gradient of the empirical loss with respect to all parameters of the network is computed. Those gradients are used in eq. (5.17) to update the current network configuration. The loop is repeated until training is finished.

Training a neural network model using back-propagation is an iterative process. The network updates its parameters several times employing the same training set. Every time the network predicts the output of all inputs in the training datasets, we say that the network has performed a training epoch. In general, training is carried out for many epochs. Choosing the right number of epochs to train a network is not straightforward, as numerical optimisation methods are not ensured to reach a global minimum. As a consequence, networks tend to be trained for several epochs, until their performance plateaus. Additionally, while training, it is standard to test the

performance of the model in a test set, that is, a collection of inputs whose outputs are known, but not included at training. This is usually called validating the model.

Due to the fact that training sets tend to contain large number of examples, it is common to split the training set into mini-batches to reduce the computational cost. In addition, this can also help over-fitting, as the neural network updates its parameters at every iteration using only a proportion of the total dataset.

Some resources discussing neural networks, how to implement and train them, and their possible applications can be found in Refs. [20–22].

5.2 Convolutional neural networks

Feedforward linear networks perform well in a large number of tasks, but they assume that the input space is ultra-local, that is, each variable in the input space is completely independent. This assumption is reasonable for unstructured data belonging to a vector space, where no spatial connections between different variables exist. This condition is not always encountered in all types of data, for example in images, time series measurements or functional data structures. In all these examples, the information at a given location is affected by its neighbourhood, for example, the colour of a pixel in an image depends on its surroundings. As a result, a modified version of the standard linear neural network layer defined in eq. (5.9) that exploits this property could lead to improved performances for datasets with correlated input spaces.

We can exploit the locality in our data by making use of the concept of neighbourhood. For this to work, the input space should be measurable: there must be a notion of closeness between inputs. For instance: if the input space corresponds to images, the neighbourhood of a particular pixel is defined as all pixels in an area surrounding that pixel; in time-series data, the neighbourhood of a measurement is composed by all points in a region surrounding that particular measurement. Locality can be exploited when there is a notion of *volume* in the input data structures.

The goal is then to modify the model in eq. (5.9) so that it takes into account the local information in the input space in order to produce an output. A way of achieving this is by using convolutions. Fundamentally, a convolution is just an operator that applies a function, usually called kernel or filter, over an input function. We use the label K to refer to kernels, while the label F denotes input functions. In general, convolutions are defined as

$$F'(t) = (F \star K)(t) = \int_{-\infty}^{\infty} dsF(t)K(s-t), \tag{5.18}$$

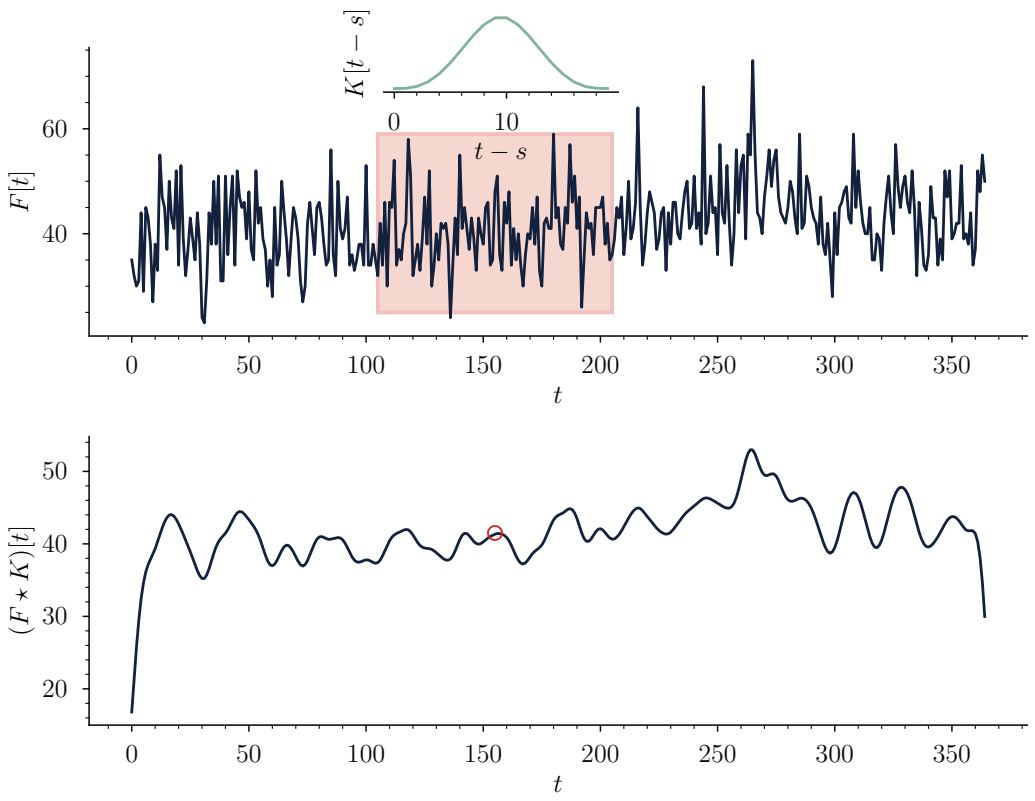


Figure 5.5: An example of a 1-dimensional convolution of a time series signal with a bell-shaped kernel K. In this case, the convolution takes place in a window of 20 measurements. The upper figure shows the original signal and the kernel acting on a particular region through eq. (5.19). The lower figure shows the result of the convolution.

where t represents the arguments of the function F. The operator $\star$ is usually employed to denote convolutions.

In our particular case, we are interested in discrete convolutions, that is, convolutions of discrete signals F with discrete kernels K:

$$F'[t] = (F \star K)[t] = \sum_s F[s]K[s-t]. \tag{5.19}$$

The notation $[t]$ implies that we are accessing the t^{th} element in the signal. The sum runs over all the possible values in which F is defined. The action of K over F at a given position t selects a neighbourhood of F centred at t, and produces a weighted

average of the input signal, where the weights are assigned by K. An example of a convolution over a 1-dimensional signal using a bell-shaped kernel can be found in Figure (5.5). The bell-shaped kernel used in Figure (5.5) belongs to the parametric family of Gaussian kernels, defined as

$$K(s-t;\sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\Big(- \frac{(s-t)^2}{\sigma^2}\Big), \tag{5.20}$$

where σ is a free-parameter completely specifying K. A value of $\sigma = 2$ was employed to produce Figure (5.5).

In order to generate a neural network layer that exploits the locality in the input signal, we promote the convolution operation in eq. (5.19) as the new *affine transformation* Z in eq. (5.9). This implies that a convolutional layer will process an input signal A^{l-1} using

$$Z^l[t] = (A^{l-1} \star K)[t] = \sum_s A^{l-1}[s] K^l[s-t] + b^l, \tag{5.21}$$

where b^l is an optional bias, and K^l is a learnable kernel defining the convolutional layer. To apply non-linearity to the layer, Z^l is usually passed through a non-linear activation function: $A^l = f^l(Z^l)$.

The question now is how to define the trainable kernel on each layer, which is equivalent to selecting the dependence of K with some learnable parameters. One possibility would be to use a parametric family of kernels, such as the Gaussian kernels defined in eq. (5.20). Through back-propagation on a training set, the parameters defining the kernel are tuned to approximate the target mapping. The main problem of this approach is its lack of flexibility: we cannot be sure that a given parametric family is optimal for the task in question. To avoid this problem, we can promote K to be a sequence of tunable real parameters that can be updated at every iteration. In this formulation, K can be viewed as a blank canvas that can be filled with different values to suit a particular task. This means that the space of available kernels is richer, at the cost of increased number of learnable parameters in the model. An example of a blank canvas kernel for 1-dimensional data with length 8 is

$$K = (k_1, k_2, k_3, k_4, k_5, k_6, k_7, k_8), \tag{5.22}$$

where each of the weights k_i is a tunable real number.

For finite signals, convolutions are not ensured to preserve lengths: the output signal might have a smaller length than the input signal. In general, the convolution of a signal of length L and a kernel of size K_x produces another signal with length

$$L' = L - K_x + 1. \tag{5.23}$$

In order to preserve lengths, it is common to pad the input signal with dummy values both at the beginning and the end of the signal. The convolution of a padded input signal F with p dummy values appended both at the beginning and the end of the signal, and a kernel of size K, produces another signal with length

$$L' = L - N_k + 1 + 2\,p. \tag{5.24}$$

In addition, it is sometimes common to skip sequential applications of the kernel over the input to compress the input signal; this is called striding the convolution. It is worth noting that convolutions allow the extraction of the fundamental features of the input signal at the cost of lowering its resolution. Figure (5.6) shows an example of a convolution between a 1-dimensional input signal of size $L = 5$ and a kernel of size $K_x = 3$.

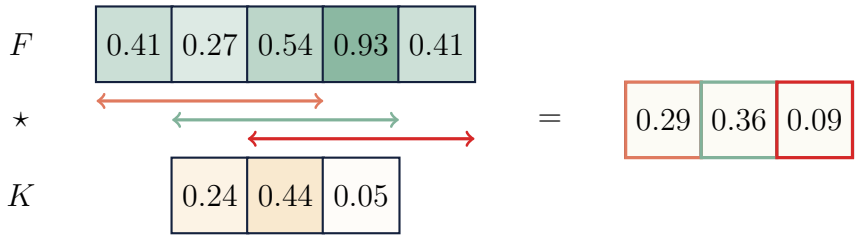


Figure 5.6: Example of a convolution between an input signal F of size $L = 5$, and a blank kernel of size $K_x = 3$. The output signal has size $L' = L - K_x + 1 = 3$. The convolution is applied by superposing the kernel over the signal starting from the beginning, applying the convolution operation defined in eq. (5.19), and then sliding the window to the next point in the signal. If stride was employed, then the window would be moved skipping s points between sequential applications.

Convolutions can also be applied to higher dimensional data, for example, images. In grey-scale image-processing, images are represented as matrices of dimensions (L_x, L_y) whose entries correspond to different pixels. Each pixel in the image represents a shade of grey. In this 2-dimensional context, kernels can be represented as 2-dimensional functions.

Applying a kernel K to an image I can be done through the 2-dimensional generalisation of eq. (5.19):

$$I'_{x,y} = (I \star K)_{x,y} = \sum_{i,j} I_{x-i,y-j} K_{i,j}, \tag{5.25}$$

where the entries of K dictate how I would transform under the convolution. Figure (5.7) contains a diagram where convolutions are employed to select the edges of

a grey-scale image. This is a common image processing technique included in most image manipulation programs.

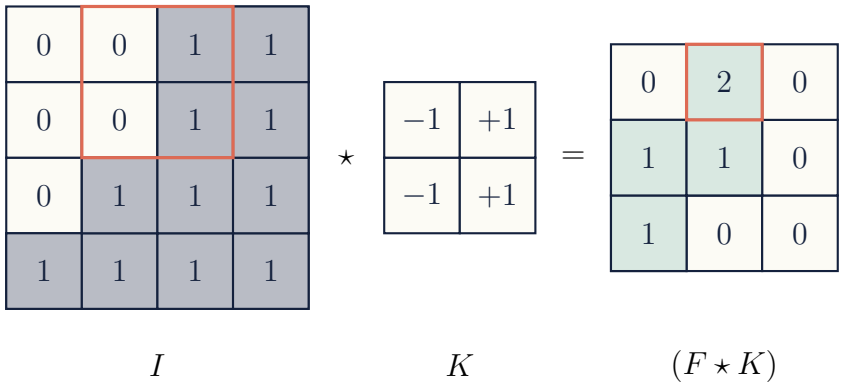


Figure 5.7: Diagram showing the convolution operation of a simplified binary 4×4 image with a 2×2 edge-selection filter. The resulting image shows the regions in which the original image contain edges.

When applied to neural networks, the grey-scale image-processing kernels are matrices of size (K_x, K_y), the entries of which correspond to learnable weights:

$$K = \begin{pmatrix} w_{1,1} & w_{1,2} & w_{1,3} \\ w_{2,1} & w_{2,2} & w_{2,3} \\ w_{3,1} & w_{3,2} & w_{3,3} \end{pmatrix} \tag{5.26}$$

Studying how convolutional layers could be applied to coloured images allows us to define an important concept: the channel or feature space. In coloured images, each pixel represents a 3-dimensional vector belonging to colour space. As a consequence, coloured images can be viewed as tensors of dimensions $(3, L_x, L_y)$, where the first dimension corresponds to the colour space of each pixel in the image. To process tensorial objects using convolutions, we would need to employ tensorial kernels. In the case of 2-dimensional images, the kernel dimensions would be: (C_o, C_i, K_x, K_y), where C_o is the number of output channels of the layer, and C_i is the number of channels of the input signal. The output of the layer would be a tensor of dimensions (C_o, L'_x, L'_y), where both L'_x and L'_y are computed following eq. (5.24). A possible implementation of a convolutional layer in this case is

$$Z^l_c = b^l_c + \sum_{k=0}^{C_i - 1} K^l_{c,k} \star I_k. \tag{5.27}$$

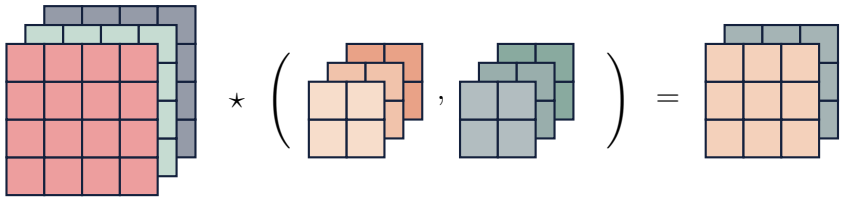


Figure 5.8: Diagram showing how convolutions could be applied to input data with multiple features, that is, multiple channels as encountered in coloured images. In this particular case, the input could be a coloured image of 4×4 dimensions, which can be represented by a tensor of dimensions $(3, 4, 4)$. As we would like to obtain an output containing 2 channels, we need to employ a tensorial kernel of dimensions $(2, 3, K_x, K_y)$, where $K_x = K_y = 2$. The convolved signal is computed using eq. (5.27), Consequently, its dimensions are $(2, 3, 3)$.

In eq. (5.27), c represents a particular output channel, k refers to one input channel, and $\star$ is the convolution operator defined in eq. (5.25). A given output channel is computed by adding several convolutions. The dimension of each output channel is $(1, L'_x, L'_y)$. This procedure allows us to compensate the loss of resolution in the input signal produced by the convolutions by increasing the feature/channel space. Typically, the output of the convolutional layer Z^l is fed to a non-linear activation function in order to add some non-linearity to the output space. Figure (5.8) contains a diagram showing how convolutions can be applied to coloured images.

To conclude our discussion about convolutional neural networks, we note that convolutional layers can be used in combination with regular linear layers. To do so, we can *flatten* the output tensors by stacking their outputs into a 1-dimensional array. Nowadays, it is frequent to include several convolutional layers in a neural network, mainly in the first stages of the architecture. After the input is processed by all convolutional layers, a set of linear layers is usually employed before generating the final output of the network.

Convolutional neural networks produce state-of-the-art results in a large variety of tasks. However, training them tends to be relatively expensive due to their large number of learnable parameters. Before applying convolutional layers to a given task, it is important to analyse the properties of the problem in question. Convolutions are mainly useful when dealing with data structures containing a notion of locality. For more information about convolutional neural networks, their applications, and recent developments, we refer to Refs. [23–26].

Chapter 5. References

[1] Eric A Wan. "Neural network classification: A Bayesian interpretation". In: *IEEE Transactions on Neural Networks* 1.4 (1990), pp. 303–305.

[2] Raphael Féraud and Fabrice Clérot. "A methodology to explain neural network classification". In: *Neural networks* 15.2 (2002), pp. 237–246.

[3] Yuxi Li. "Deep reinforcement learning: An overview". In: *arXiv:1701.07274* (2017).

[4] Leslie Pack Kaelbling, Michael L Littman and Andrew W Moore. "Reinforcement learning: A survey". In: *Journal of artificial intelligence research* 4 (1996), pp. 237–285.

[5] Zonghan Wu et al. "A comprehensive survey on graph neural networks". In: *IEEE transactions on neural networks and learning systems* 32.1 (2020), pp. 4–24.

[6] Prakash M Nadkarni, Lucila Ohno-Machado and Wendy W Chapman. "Natural language processing: an introduction". In: *Journal of the American Medical Informatics Association* 18.5 (2011), pp. 544–551.

[7] Yoav Goldberg. "A primer on neural network models for natural language processing". In: *Journal of Artificial Intelligence Research* 57 (2016), pp. 345–420.

[8] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach et al. Curran Associates, Inc., 2019, pp. 8024–8035. URL: `http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf`.

[9] Francois Chollet et al. *Keras*. 2015. URL: `https://github.com/fchollet/keras`.

[10] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: `https://www.tensorflow.org/`.

[11] Frank Rosenblatt. "The perceptron: a probabilistic model for information storage and organization in the brain." In: *Psychological review* 65.6 (1958), p. 386.

[12] Xue Ying. "An overview of overfitting and its solutions". In: *Journal of physics: Conference series*. Vol. 1168. 2. IOP Publishing. 2019, p. 022022.

[13] Kurt Hornik, Maxwell Stinchcombe and Halbert White. "Multilayer feedforward networks are universal approximators". In: *Neural networks* 2.5 (1989), pp. 359–366.

[14] Xavier Glorot and Yoshua Bengio. "Understanding the difficulty of training deep feedforward neural networks". In: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings. 2010, pp. 249–256.

[15] Wojciech Samek et al. "Explaining deep neural networks and beyond: A review of methods and applications". In: *Proceedings of the IEEE* 109.3 (2021), pp. 247–278.

[16] Alfredo Canziani, Adam Paszke and Eugenio Culurciello. "An analysis of deep neural network models for practical applications". In: *arXiv:1605.07678* (2016).

[17] Adam Paszke et al. "Automatic differentiation in pytorch". In: (2017).

[18] Atilim Gunes Baydin et al. "Automatic differentiation in machine learning: a survey". In: *Journal of Marchine Learning Research* 18 (2018), pp. 1–43.

[19] Sebastian Ruder. "An overview of gradient descent optimization algorithms". In: *arXiv:1609.04747* (2016).

[20] Ian Goodfellow, Yoshua Bengio and Aaron Courville. "Deep learning (adaptive computation and machine learning series)". In: *Cambridge Massachusetts* (2017), pp. 321–359.

[21] Nikhil Ketkar and Eder Santana. *Deep learning with Python*. Vol. 1. Springer, 2017.

[22] Michael A Nielsen. *Neural networks and deep learning*. Vol. 25. Determination press San Francisco, CA, USA, 2015.

[23] Rikiya Yamashita et al. "Convolutional neural networks: an overview and application in radiology". In: *Insights into imaging* 9.4 (2018), pp. 611–629.

[24] Jiuxiang Gu et al. "Recent advances in convolutional neural networks". In: *Pattern recognition* 77 (2018), pp. 354–377.

[25] Zewen Li et al. "A survey of convolutional neural networks: analysis, applications, and prospects". In: *IEEE transactions on neural networks and learning systems* (2021).

[26] Serkan Kiranyaz et al. "1D convolutional neural networks and applications: A survey". In: *Mechanical systems and signal processing* 151 (2021), p. 107398.

Chapter 6

Spectral reconstruction with neural networks

In the last section of Chapter (2), the spectral function of a Euclidean 2-point correlation function was introduced. From its definition, we learnt that it contained all the physical information encoded in the expectation value of a particular pair of field operators. This is a consequence of the fact that any combination of two field operators can be expressed in terms of a particular spectral function. As a result, extracting the associated spectral function of an imaginary-time correlation function allows the analytic continuation from Euclidean time τ to Minkowskian time t, therefore gaining access to the real-time dynamics of the system.

6.1 Introduction to spectral reconstruction

As previously discussed in Chapter (2), the relationship between a thermal Euclidean correlation function, $C_E(\tau, \vec{k})$, and its associated spectral function, $\rho(\omega, \vec{k})$, is an integral relationship, defined as

$$C_E(\tau, \vec{k}) = \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} K(\tau, \omega)\rho(\omega, \vec{k}). \tag{6.1}$$

We refer to eq. (6.1) with the name spectral relationship. K represents the kernel that dictates how to transform from frequency space to Euclidean time domain. In the interest of simplifying the notation, the term correlation function is employed to refer to *Euclidean* correlation functions. Consequently, we drop the label E in $C_E(\tau, \vec{k})$. Furthermore, the external momenta label is omitted: $C_E(\tau)$.

The kernel in eq. (6.1) follows from analyticity. For mesonic 2-point correlation functions, it is written as

$$K(\tau,\omega) = \frac{\cosh(\omega\,(\tau - \frac{1}{2T}))}{\sinh(\omega/2T)}, \tag{6.2}$$

where T represents the temperature of the system. Equation (6.2) is only defined for $0 < \tau < N_\tau = 1/T$.

Another way of expressing eq. (6.2) is

$$K(\tau,\omega) = \exp(-\omega\tau)[1 + n_B(\omega)] + \exp(\omega\tau) n_B(\omega), \tag{6.3}$$

where n_B is the Bose distribution,

$$n_B(\omega) = \frac{1}{\exp(\omega/T) - 1}. \tag{6.4}$$

In the low temperature limit, the mesonic kernel reduces to

$$\lim_{T\to 0^+} K(\tau,\omega) = \lim_{T\to 0^+} \frac{\cosh(\omega\,(\tau - \frac{1}{2T}))}{\sinh(\omega/2T)} = \exp(-\omega\,\tau). \tag{6.5}$$

All our experiments are performed employing the low temperature limit of the mesonic kernel, as it transforms a complex convolution integral into a simple Laplace transform.

In general, for complex systems such as QCD, only numerical estimates of Euclidean correlation functions can be computed, which implies that their corresponding spectral functions are only accessible through a numerical inversion of eq. (6.1). The task of inverting the spectral relation is referred to as the spectral reconstruction of the correlation function. Typically, high quality spectral functions are required in order to access the real-time dynamics of the system: low resolution spectral functions cannot be accurately integrated to generate a precise analytical continuation from Euclidean to Minkowskian space-time. Reconstructing high-quality spectral functions is an arduous task: first, because numerical estimates of correlation functions impede an analytical inversion of the spectral relation; and second, because the information encoded in a numerical correlation function is difficult to access. For instance, their high-energy information is heavily suppressed at $\tau > 0$ — see eq. (6.1) and (6.2). As a result, numerical spectral reconstruction is an ill-posed problem [1–4], where the desired information is easier to access in $\rho(\omega)$, than in the available $C(\tau)$.

In our studies, we assume that all correlation functions can be regarded as discrete signals of fixed length N_τ. Additionally, we also treat the spectral functions as discrete signals of length N_ω. In practice, $N_\tau \ll N_\omega$. Although the spectral functions are initially defined over the whole positive real line, numerical spectral reconstruction only aims to estimate their low-frequency region. As a result, in our experiments, the spectral functions are only defined over an equispaced region of frequencies:

$$\Omega = [\omega_0, \omega_f], \tag{6.6}$$

where ω_0 represents the initial frequency at which each spectral function is computed, and ω_f corresponds to the last frequency at which every ρ is defined. As the frequencies are equispaced, and the spectral functions are computed at N_ω different frequencies, then, the spectral function resolution is

$$\Delta\omega = \frac{|\omega_f - \omega_0|}{N_\omega}. \tag{6.7}$$

Other discretisations of Ω are also possible. From now on, the terms energy and frequency are employed indistinctly to refer to the spectral functions domain, Ω.

We can inspect the ill-posedness of spectral reconstruction by treating it as a regression task in which a spectral function model $\tilde{\rho}$ is varied in order to minimise the distance between the computed reference numerical correlation function C, and the correlation function generated by integrating $\tilde{\rho}$ in eq. (6.1) with the appropriate kernel. The resulting model correlation function is labelled $\tilde{C}$. In this context, ρ represents the unknown ground-truth target spectral function. To measure the distance between C and $\tilde{C}$, we define a norm $||\cdot||$:

$$||C(\tau) - \tilde{C}(\tau)|| = || \int \frac{d\omega}{2\pi} K(\tau, \omega) \big[\rho(\omega) - \tilde{\rho}(\omega)\big] ||. \tag{6.8}$$

The norm is defined for all Euclidean times at which C is defined; an example of a valid norm is the standard Euclidean norm. As we deal with numerical estimates of the correlation function, then, even in the case in which the left-hand side of eq. (6.8) fulfils

$$||C(\tau) - \tilde{C}(\tau)|| < \epsilon, \tag{6.9}$$

where ϵ is an arbitrary threshold, $\tilde{\rho}$ is not ensured to be equal to ρ. There are an infinite number of spectral functions different to ρ whose associated numerical correlation functions differ less than ϵ for all Euclidean times considered. This problem is even more severe once we take into account the fact that lattice correlation functions are statistical estimates of a true underlying population correlation function. As a

result, lattice estimates are affected by statistical noise, which aggravates the problem of choosing the appropriate model $\tilde{\rho}$ that represents the true underlying target spectral function $\rho(\omega)$.

It is important to note that the ill-posedness of spectral reconstruction is just a consequence of the fact that only numerical estimates of correlation functions can be computed. If we had access to the exact analytical correlation function, then, eq. (6.1) could be directly inverted as the Laplace transform is known to be an invertible transformation.

A strategy that could reduce the ill-posedness of spectral reconstruction consists on employing all available prior information on the properties of C and ρ, for example: positivity of the spectral functions, normalisation of the correlation functions, or asymptotic behaviour of ρ and C. However, using this information in our favour does not lead to a simple spectral reconstruction strategy, as the amount of prior information available is small compared to the complexity of the task.

Spectral reconstruction belongs to a set of problems known as inverse problems. Inverse problems are encountered in many fields, for instance, they appear in seismology [5, 6] and medical imaging [7]. In addition, spectral reconstruction is not specific to non-zero temperature QCD; for a recent review of spectral reconstruction in thermal QCD and other quantum systems, we refer to Ref. [8]. At the date in which this document is being written, spectral reconstruction is not completely under control. Nevertheless, there are different methods that try to approximately solve it. Some of these methods are: the Maximum Entropy Method (MEM) [9–16], Gaussian processes models [17], Kernel Ridge Regression [18], Backus-Gilbert method [19], or the Stochastic Average (SA) methodology [20–23].

This chapter presents and tests a simple methodology to perform numerical spectral reconstruction of numerical correlation functions. The methodology employs deep neural networks as the core tools to perform the reconstruction. The goal is to explore the limits of the methodology by testing it through different scenarios with variable complexity. Some basic knowledge about deep convolutional neural networks is assumed throughout this chapter. An introduction to the field to neural network and convolutional layers is provided in Chapter (5).

6.2 Deep neural networks in spectral reconstruction

This section contains some studies conducted with a simple methodology employing deep neural networks to spectral reconstruction. Deep neural networks are standard

feedforward neural networks containing numerous layers and, therefore, large number of learnable parameters. They are known to achieve better performances than shallower architectures in most tasks due to their large expressiveness. Although we restrict our analysis to deep convolutional neural networks, other architectures can also be employed.

Although the literature exploring the application of neural networks to the field of spectral reconstruction is limited, the few already available results are promising [24–26]. The experiments presented in this document are an extension of the ones presented in Ref. [26], where the authors introduce the simplest possible methodology in which neural networks can be employed to perform spectral reconstruction. In their work, the authors explore the performance of several architectures in different simple scenarios. However, the neural network architectures employed in Ref. [26] can be considered shallow by modern standards.

The methodology presented in Ref. [26] tries to exploit the previously mentioned expressiveness of deep neural networks in order to build a model in which correlation functions are mapped to their associated spectral functions without the need of explicitly inverting eq. (6.1). This mapping can be mathematically formulated as

$$f : \mathcal{C} \to \mathcal{R}, \tag{6.10}$$

where $\mathcal{C}$ represents the space of all possible Euclidean correlation functions involved in the reconstruction task, and $\mathcal{R}$ represents the space of all their associated spectral functions. Fundamentally, $\mathcal{C}$ represents the space containing all correlation functions appearing in the left-hand side of eq. (6.1), while $\mathcal{R}$ is the space composed by all spectral functions in the right-hand side of the spectral relation. In practice, the mapping processes correlation function objects as inputs, and produces their expected spectral functions as outputs:

$$f(C) = \rho. \tag{6.11}$$

Unlike other popular spectral reconstruction methodologies, such as MEM, our methodology assumes that the reconstruction mapping f is deterministic. The fact that the mapping is assumed deterministic does not mean that it cannot treat statistical estimates of the input correlation functions, as functions of random variables are also random variables. It only means that it does not assign probabilities to the reconstruction, that is, it does not try to model the posterior probability distribution function over the space of input correlation functions: $P(\rho|C)$.

In our experiments, all input correlation functions are assumed to be perfect estimates of a population correlation function. As a result, we do not analyse the uncertainty in the reconstruction. However, one should always perform an uncertainty analysis when dealing with real lattice correlation functions; remember that

a lattice estimate of a particular correlation functions contains N different (and independent) estimates of the same population correlation function. To measure the uncertainty in the reconstructed spectral function, one can process each of the N lattice estimates of the population correlation function through the reconstruction mapping defined in eq. (6.11). By doing so, we produce N estimates of the target spectral function. The sample average of those N spectral function estimates corresponds to an unbiased estimate of the true population spectral function as long as some standard conditions are fulfilled. The uncertainty in this spectral function can be estimated through the standard error of the sample mean.

Assuming that the mapping defined in eq. (6.10) exists, then we can parametrise it using a neural network model: $f = f_W$, where W represents the set of all learnable weights in the model. Provided that the fundamental internal features of the mapping are properly captured by the neural network, then the trained model could be used to process input correlation functions in order to obtain their respective spectral functions.

The task of learning the correct mapping can be formulated as a standard supervised regression task. In this framework, pairs composed of Euclidean correlation functions and their respective spectral functions are provided to the neural network model. A collection of pairs (C, ρ), each of them respectively sampled from $\mathcal{C}$ and $\mathcal{R}$, is called a training set. In principle, through back-propagation on a large and descriptive training dataset, the network model should be able to learn the internal representation of the mapping: $f \simeq f_W$. The learning procedure can be mathematically defined in terms of a loss function measuring the distance between the target training spectral function, $f(C) = \rho$, and the proposed spectral function by the network, $f_W(C) = \hat{\rho}$.

In order to teach the network the correct mapping, a large collection of good quality training pairs is required. However, we do not have access to real training pairs, as spectral reconstruction on real lattice data is not possible yet. To circumvent this difficulty, artificial datasets that try mimicking the real spaces are employed.

To generate our artificial training sets, we employ a series of assumptions about the properties of both $\mathcal{C}$ and $\mathcal{R}$. First, all correlation functions in our datasets are measured at $N_\tau = 64$ different Euclidean times; including correlation functions of variable lengths can be achieved by padding the signals with zeroes up to a maximum predefined length. Secondly, all spectral functions are assumed to be measured at $N_\omega = 1000$ different frequencies. In addition, the spectral functions are only defined over Ω, previously defined in eq. (6.6). In our experiments, we employ $\omega_0 = 0$ and $\omega_f = 8$, which results in a spectral functions' resolution of $\Delta\omega = 0.008$. The precise physical dimension of the energy space is not important in our experiments.

In order to construct the artificial training sets from which the mapping f is learned, we need to sample a collection of spectral functions from $\mathcal{R}$. To do so, we assume that all spectral functions in $\mathcal{R}$ can be decomposed employing a linear combination of N_p independent semi-positive definite parametric peaks $\Gamma_p(\omega;\theta_p)$, where θ_p is the set of real parameters completely specifying Γ_p. As a result, all spectral functions in $\mathcal{R}$ can be decomposed as

$$\rho(\omega) = \sum_{p=1}^{N_p} \Gamma_p(\omega;\theta_p). \tag{6.12}$$

Whether the decomposition above can be performed, and the type of peaks that should be used provided the decomposition is indeed valid, is still an open problem of spectral reconstruction in QCD.

In principle, different types of parametric peaks can be employed in eq. (6.12), we restrict ourselves to bell-shaped peaks:

$$\rho(\omega) = \sum_{p=1}^{N_p} A_p \, \exp\big(-\frac{1}{2}\frac{(\omega - M_p)^2}{W_p^2}\big). \tag{6.13}$$

The results of the experiments should be independent of the type of peaks chosen, as our methodology knows nothing about the functional shape of ρ. The parameters defining each peak are $\theta_p = (A_p, M_p, W_p)$, which correspond to each peak amplitude, center and width. The possible values that the parameters θ_p can take are

$$A_p \in [0.0, 1.0]; \quad M_p \in [0.1, 5.5]; \quad W_p \in [0.01, 0.20]. \tag{6.14}$$

In Ref. [26], the authors limit the minimum distance between two peak centres, M_p. Doing this can be helpful to restrict the possible structures appearing in $\mathcal{R}$, however, as it is not completely justified, we allow overlaps between peaks. A particular choice of the possible values that the peak parameters can take is referred to as a region of interest.

To make all examples in a particular dataset consistent, we normalise the spectral functions so that the following condition is fulfilled:

$$\int_0^\infty d\omega \rho(\omega) = 6.0. \tag{6.15}$$

Several spectral reconstruction training sets can be generated by sequentially applying eq. (6.13) with a particular fixed number of peaks. Not allowing variable

number of peaks in our datasets allows us to control the complexity of each training dataset: datasets composed by spectral functions with low number of peaks are expected to be simpler than datasets in which the spectral functions are generated through the combination of large number of peaks. In order to produce a particular training dataset, first, we sample N_b different spectral functions, each of them containing N_p peaks. The parameters defining each peak are uniformly sampled in the volume defined in eq. (6.14). Once a spectral function dataset is generated, we can produce its corresponding correlation function dataset by integrating eq. (6.1) over Ω using all previously generated spectral functions. As a result of the data-generation pipeline, we produce two linked datasets: $\mathbf{C}$, containing N_b input correlation functions, and $\mathbf{R}$, which is composed by their N_b associated spectral functions. Each dataset can be represented by a matrix of dimensions $N_b \times N_d$, where N_d is the length of each object: $N_d = N_\omega$ for spectral functions,

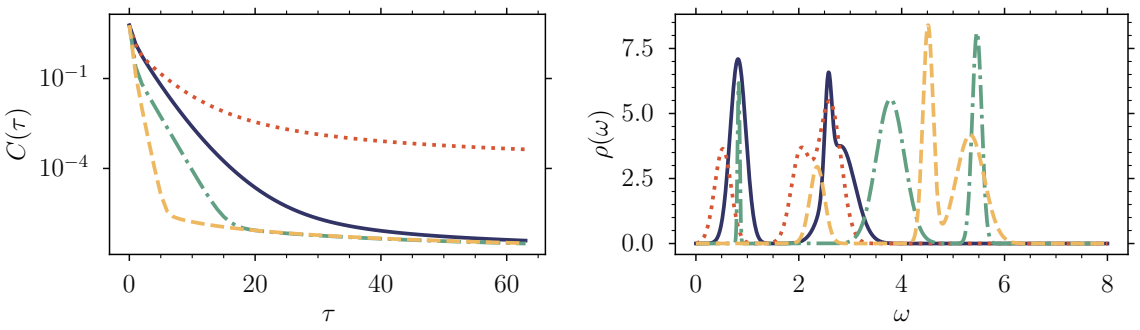
$$\mathbf{R}[N_b, N_p] = \begin{pmatrix} \rho_1(\omega_1) & \cdots & \rho_1(N_\omega) \\ \rho_2(\omega_1) & \cdots & \rho_2(N_\omega) \\ \vdots & \ddots & \vdots \\ \rho_{N_b}(\omega_1) & \cdots & \rho_{N_b}(N_\omega) \end{pmatrix}, \tag{6.16}$$

and $N_d = N_\tau$ for correlation functions,

$$\mathbf{C}[N_b, N_p] = \begin{pmatrix} C_1(\tau_1) & \cdots & C_1(N_\tau) \\ C_2(\tau_1) & \cdots & C_2(N_\tau) \\ \vdots & \ddots & \vdots \\ C_{N_b}(\tau_1) & \cdots & C_{N_b}(N_\tau) \end{pmatrix}. \tag{6.17}$$

Figure (6.1) contains some randomly selected training pairs extracted from a spectral function dataset with $N_p = 3$

As different training datasets can be generated by varying N_b and N_p, we require a notation to specify a particular dataset. In our case, $\mathbf{R}[N_b, N_p]$ represents a spectral function dataset containing N_b randomly sampled spectral functions, each of them generated through the combination of N_p peaks. Sometimes, we do not need to specify the number of examples in a particular dataset. In this situation, we use the notation $\mathbf{R}[N_p]$ to refer to an arbitrary long spectral function dataset whose spectral functions are generated through the combination of N_p peaks. The same notation can be applied to correlation function datasets. It is important to remember that the kernel, the energy region at which each spectral function is defined, the region of interest in which the parameters specifying each peak are sampled, the length of both C and ρ, and the functional form of the peaks employed to span each spectral function are the same in all training sets considered.



(a) Correlation functions **(b)** Spectral functions

Figure 6.1: Randomly selected training pairs (C, ρ) belonging to a training set in which the spectral functions are generated by combining $N_p = 3$ randomly sampled bell-shaped peaks. Training pairs share colour and line-style in both figures.

6.2.1 Basis expansion formulation of the mapping

The mapping f introduced in eq. (6.10) processes input correlation function signals of length 64, and produces output signals of length 1000. Due to the large difference between input and output sizes, it is advisable to reduce the degrees of freedom (d.o.f) involved in the reconstruction:

$$\text{d.o.f} = N_{out} - N_{in}. \tag{6.18}$$

This can be achieved by employing a different representation of the target spectral functions. Note that different representations of the input correlation functions can also be explored.

In Ref. [26], the authors explore two different representations of the spectral functions: the standard one, where the output of the mapping is just the complete spectral function; and a parametric formulation, where each spectral function in a given dataset $\mathbf{R}[N_p]$ is represented by a set of real parameters θ. Those parameters correspond to the ones defining each peak in a particular spectral function. As a result, in this representation, the target training set is not $\mathbf{R}[N_p]$, but a hypothetical $\mathbf{P}[N_p]$, composed by all parameters needed to completely specify each spectral function in $\mathbf{R}[N_p]$. In our particular case, the parametric formulation of the reconstruction mapping processes input correlation functions of length 64, and produces $3\,N_p - 1$ parameters representing their associated spectral functions; one parameter is completely determined by the normalisation condition in eq. (6.15). The reconstructed parameters can then be employed in eq. (6.13) to generate the desired target

spectral function ρ. Although this formulation greatly reduces the degrees of freedom involved in the reconstruction, it is not well-defined in the limit of large N_p: the size of the target space grows with the number of peaks. Additionally, this formulation only works for parametric peaks, which might not be desirable in future applications.

As a way of producing a generalisable formulation of the reconstruction mapping in which the number of degrees of freedom involved is reduced, we employ an orthogonal basis expansion to decompose each spectral function in $\mathbf{R}[N_p]$. In this formulation, every numerical spectral function is decomposed using a linear combination of orthogonal basis functions u_s, each of them accompanied by a real coefficient l_s:

$$\rho(\omega) \simeq \sum_{s=1}^{N_s} l_s \, u_s(\omega), \tag{6.19}$$

As the basis functions are shared among all spectral functions in $\mathbf{R}[N_p]$, we can represent each object in $\mathbf{R}[N_p]$ using a set of N_s real coefficients. Due to the assumed orthogonality of the basis functions,

$$\int_0^\infty d\omega \, u_s(\omega) \, u_{s'}(\omega) = \delta_{s,s'}, \tag{6.20}$$

each coefficient in the expansion can be computed applying the following relationship:

$$l_s = \langle \rho(\omega), u_s(\omega) \rangle, \tag{6.21}$$

where $\langle x, y \rangle$ denotes the usual Euclidean scalar product. We refer to the collection of all N_s basis functions with the label $\mathbf{U}$. As the number of basis functions employed in the right-hand side of eq. (6.19) is arbitrary, the expansion is not exact, hence the almost equal sign. The difference between the reference $\rho(\omega)$ and its expanded version, $\tilde{\rho}$, can be quantified using δ, which is defined as

$$\delta \leq ||\rho(\omega) - \tilde{\rho}(\omega)|| = ||\rho(\omega) - \sum_{s}^{N_s} l_s \, u_s(\omega)||. \tag{6.22}$$

In the equation above, $|| \cdot ||$ represents the standard Euclidean norm. For a fixed δ, different choices of basis function might lead to different number of basis functions N_s.

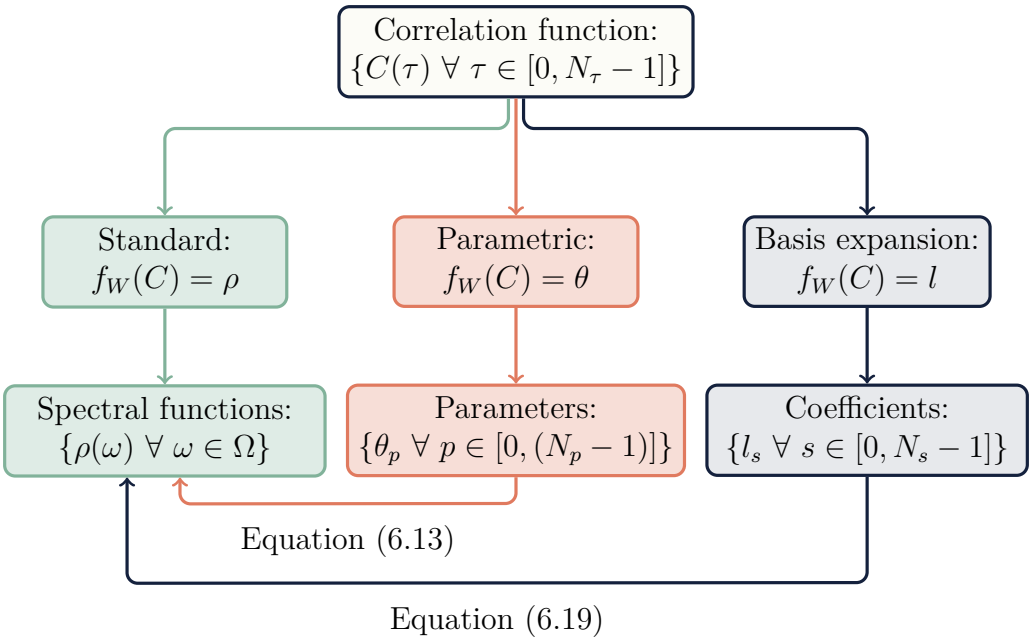


Figure 6.2: Diagram explaining the three formulations of the spectral reconstruction mapping. All three formulations process N_τ-dimensional input correlation functions. Although theoretically equivalent, each formulation produces a different representation of the spectral function as an output.

In the basis expansion formulation, the reconstruction mapping processes 64-dimensional input correlation functions, and produces a set of N_s coefficients, which depend on the particular choice of basis function employed in the decomposition. Through the application of eq. (6.19), the original spectral function can be reconstructed. If correctly implemented, this formulation has a well-defined large N_p limit, as the number of coefficients in the expansion should not depend on N_p. Additionally, this formulation is more flexible than the parametric one, as it easily allows the inclusion of variable number of peaks in the datasets. Figure (6.2) contains a diagram with all three mapping formulations discussed in the previous paragraphs, as well as the relationships between them.

The main drawback of the basis expansion formulation is that we need to decide which set of basis functions $\mathbf{U}$ is employed in eq. (6.19). We would like to use a set of basis functions that allows a high quality — low δ — decomposition of any spectral function in $\mathbf{R}[N_p]$ with as few coefficients as possible. In our experiments, we apply the singular value decomposition (SVD) over a spectral function dataset matrix — see eq. (6.16) — in order to generate $\mathbf{U}$. Figure (6.3) shows the first 4 basis

functions extracted by applying the SVD decomposition over a randomly generated spectral function dataset composed by spectral functions with 10 bell-shaped peaks. The SVD decomposition is a well-known data decomposition algorithm that allows a systematic dimensional reduction of complex datasets in terms of their directions of maximum variance; the SVD decomposition is directly linked to another well-known decomposition method, the principal component analysis (PCA). For more information about the SVD decomposition, see Refs. [27–29].

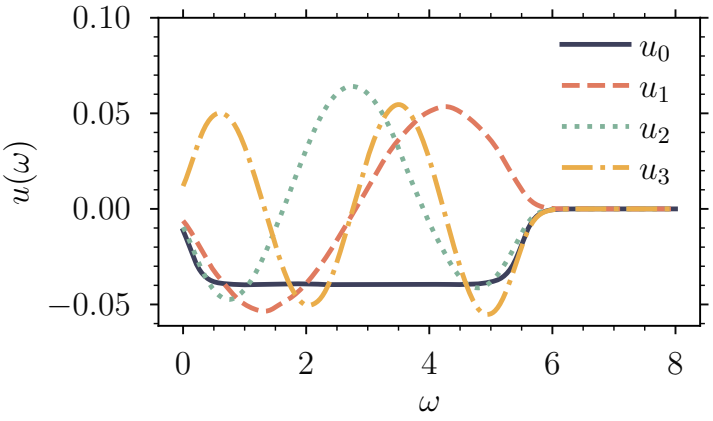


Figure 6.3: The 4 most relevant basis functions $u_s(\omega)$ extracted using the SVD decomposition of a dataset $\mathbf{R}[N_p = 10, N_b = 50000]$.

Applying the SVD decomposition over a spectral function dataset produces a sequence of N_ω orthogonal basis functions ordered by their relevance: the first basis functions contain the fundamental features of the dataset, while the last ones only represent the minor details. As a consequence, in order to reduce the degrees of freedom of the reconstruction, while at the same time maintaining a high-quality decomposition, we keep the first N_s most relevant basis functions, and discard the rest. The number of relevant basis functions used should be chosen so that the quality of the basis expansion, measured in terms of δ, is not compromised on any examples contained in the dataset. Note that for the SVD decomposition to work well, the number of examples in the dataset should be large enough so that it represents a descriptive sample of the properties of the underlying population space.

The fact that the first basis functions contain the fundamental features of the dataset is reflected on the size of the coefficients accompanying them: the first coefficients in the SVD decomposition tend to have a larger absolute value than the ones corresponding to less relevant basis functions. Therefore, it is sometimes useful to

quantify the information described by the first s coefficients in the expansion. To do so, we define the following quantity:

$$I_{\{l\}}(s) = \frac{\sum_i^s |l_s|}{\sum_i^{N_s} |l_s|}, \tag{6.23}$$

which is defined on a specific set of coefficients $\{l\}$. Note that $I_{\{l\}}(s)$ is a number between 0 and 1. From $I_{\{l\}}(s)$, one can compute the number of coefficients that contain less than a previously selected proportion of the total information, I_0. This number is represented by the label $B_{\{l\}}$, and its definition is

$$B_{\{l\}} = s_0 \quad \text{if} \quad I_{\{l\}}(s_0) \leq I_0. \tag{6.24}$$

$B_{\{l\}}$ is a positive integer less or equal to N_s. In our experiments, I_0 is set to $I_0 = 0.9$. Figure (6.4) shows some randomly selected coefficients extracted from datasets with variable number of peaks.

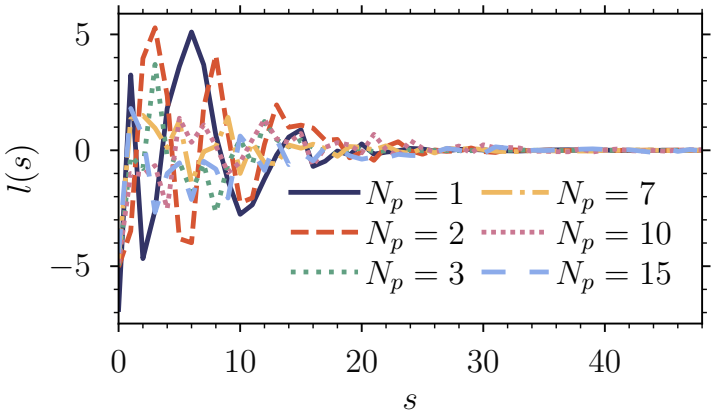


Figure 6.4: Randomly selected coefficients extracted from datasets with variable number of peaks. The coefficients are computed using the first $N_s = 128$ SVD basis functions.

6.2.2 Experiments and results

In order to explore the viability of the methodology presented before, we analyse the performance of the same deep neural network architecture on the task of learning the correct mapping f on different training datasets.

The neural network architecture employed in our experiments corresponds to a standard 34-layer convolutional residual network (ResNet) [30]. We decided to employ a convolutional architecture in our experiments in order to exploit the local information present in the input correlation function signals: the correlation function measured at a particular Euclidean time τ depends on the measured values of the same correlation function in a neighbourhood of τ, which is a consequence of the functional nature of the input correlation functions. At the time in which this document is being written, convolutional neural networks are the standard architectures employed when the input signals contain a notion of locality. For instance, they are ubiquitous in image processing, where the pixel values in a particular region tend to be largely related. A short introduction to convolutions in the context of neural networks can be found in Chapter (5). Despite convolutional neural networks being typically defined for 2-dimensional inputs, such as images, our implementation employs 1-dimensional convolutions.

The residual network architecture is known to perform exceptionally well in a large variety of complex tasks, such as image processing, protein classification or medical image. The ResNet model contains a combination of convolutional, batch-normalisation and pooling layers, as well as residual connections between those layers, which are known to suppress the problem of vanishing gradients in deep neural networks [31]. The 34-layer ResNet architecture contains more than 7 million learnable weights, and employs dropout [32, 33] to reduce possible over-fitting.

The goal of our experiments is to explore how the model behaves as the task of learning the correct reconstruction mapping increases its complexity. We vary the complexity of the task through the number of peaks in the dataset. Ideally, if the model is able to learn the fundamental features of the target mapping on each dataset, then we could conclude that the model is robust enough to compensate our lack of knowledge on the properties of spectral reconstruction. If this is in fact the case, then the large expressiveness of the model would suppress the ill-posedness of spectral reconstruction by learning the internal features of the reconstruction mapping. A similar strategy is applied in machine language translation. No one is able to model by hand the probability of finding the translated version of the word x after having measured an arbitrary long sequence of words $\{a, b, c, \dots\}$, as the search space is highly correlated, huge and sparse. However, one can use a sufficiently large and expressive model in order to learn the probability mapping by just providing the model with numerous translation examples.

As explained before, in our formulation of the problem, a basis expansion is employed to represent each spectral function in the dataset. As a result, the target of our regression is a set of coefficients generated through the application of eq. (6.21)

with **U** being the first N_s basis functions extracted from the SVD decomposition. The SVD decomposition is applied independently on each dataset. We do not allow shared basis functions among datasets to avoid employing non-optimal basis functions in the decomposition, which could externally affect the performance of the model. Nevertheless, Figure (6.5) demonstrates that the SVD basis functions extracted from large N_p datasets are able to completely reconstruct all spectral functions in lower N_p datasets with high-fidelity.

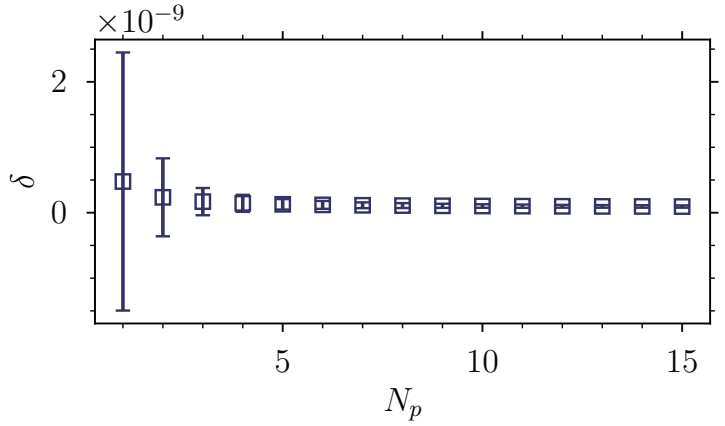


Figure 6.5: Figure showing the average and standard deviation of δ, defined in eq. (6.22), as a function of the number of peaks in a dataset. Note that δ is computed across each dataset using the first 128 basis functions extracted from the dataset with the largest number of peaks, $\mathbf{R}[N_p = 15]$. The number of examples in each dataset is $N_b = 50000$.

In our experiments, we keep the first $N_s = 128$ basis functions extracted from the SVD decomposition. This number seems to be sufficient to produce high-quality decompositions, as expressed by the small value of δ in all datasets shown in Figure (6.5). As a result, our ResNet model processes correlation functions of length $N_\tau = 64$ as inputs, and produces $N_s = 128$ coefficients as outputs.

In order to train the neural network over the different datasets, a loss function is required. In our case, we use a variation of the mean-squared error (MSE) loss on the basis expansion coefficients,

$$\mathcal{L}_L = \frac{1}{N_b} \sum_{b=1}^{N_b} \sum_{s=1}^{N_s} |l_s^b - \hat{l}_s^b|^2, \tag{6.25}$$

where $\hat{l}_s^b$ represents the set of predicted coefficients by the network, and l_s^b the true

expected coefficients. The loss function $\mathcal{L}_L$ makes the neural network output the correct coefficients for each input correlation function. The original spectral function can be reconstructed from those coefficients by applying eq. (6.19) with the previously computed 128 SVD basis functions.

In Ref. [26], the authors explore a combination of loss functions that aims to reduce the ill-posedness of the problem by forcing the model to reconstruct the exact expected spectral function, as well as its associated correlation function. These two losses are

$$\mathcal{L}_R = \frac{1}{N_b} \sum_{b=1}^{N_b} \sum_{\omega=1}^{N_\omega} |\rho^b(\omega) - \hat{\rho}^b(\omega)|^2, \tag{6.26}$$

and

$$\mathcal{L}_C = \frac{1}{N_b} \sum_{b=1}^{N_b} \sum_{\tau=1}^{N_\tau} |C^b(\tau) - \hat{C}^b(\tau)|^2. \tag{6.27}$$

The first loss function, $\mathcal{L}_R$, encourages the model to generate the coefficients that uniquely reconstruct the expected spectral function, while the second loss function, $\mathcal{L}_C$, forces the model to predict the sequence of coefficients from which the correct input correlation function C can be generated through the sequential application of eq. (6.19) and (6.1).

All three previously defined losses, $\mathcal{L}_L$, $\mathcal{L}_R$ and $\mathcal{L}_C$, can be included at training using the following linear combination:

$$\mathcal{L} = a\,\mathcal{L}_L + b\,\mathcal{L}_C + c\,\mathcal{L}_R, \tag{6.28}$$

where a, b and c are real numbers controlling the weight of each loss at training. Generally, a is set to 1, while b and c are always smaller than a. While preparing our experiments, different values of b and c were tested, but no significant increase in the performance was spotted. Although including both $\mathcal{L}_C$ and $\mathcal{L}_R$ seems to not cause a negative effect in the reconstruction, it considerably increases the training time, as both $\hat{C}$ and $\hat{\rho}$ need to be reconstructed from the coefficients predicted by the network at each training iteration. This procedure can be expensive depending on N_p and the implementation. As a result, we decided to ignore $\mathcal{L}_C$ and $\mathcal{L}_R$ in our final results, which implies that the loss function at training is just $\mathcal{L}_L$.

In order to reduce possible over-fitting at training, as well as constraining the values of the parameters, which could control the possible exploding gradients commonly spotted in deep neural networks, we add a regularisation term to our loss function. This term is just a weight-decay L_2 loss function, defined as

$$\mathcal{L} = \mathcal{L}_L + \gamma \sum_{w \in W} |w^2|. \tag{6.29}$$

In the equation above, W represents the set of all learnable parameters of the model. We set γ to $1 \cdot 10^{-4}$ in our experiments.

Additionally, before training the models, we pre-process the training datasets to help the neural network learn the correct mapping: pre-processing helps to avoid including numerical bias in the training examples, which could lead to poor improvements and even possible over-fitting. We prepare our training datasets by taking the logarithm of each independent input correlation function as a way of reducing their numerical exponential decay, which could cause the neural network to treat most part of the input signals as zero values. We do this by replacing each correlation function in $\mathbf{C}$ with their logarithm:

$$C(\tau) \to \log[C(\tau)], \quad \forall\, \tau.$$

It is worth noting that this pre-processing step does not seem to have a large impact on the final performance of the model, therefore, it may be safely skipped. However, it is advisable for really large input signals: $N_\tau \gg 1$.

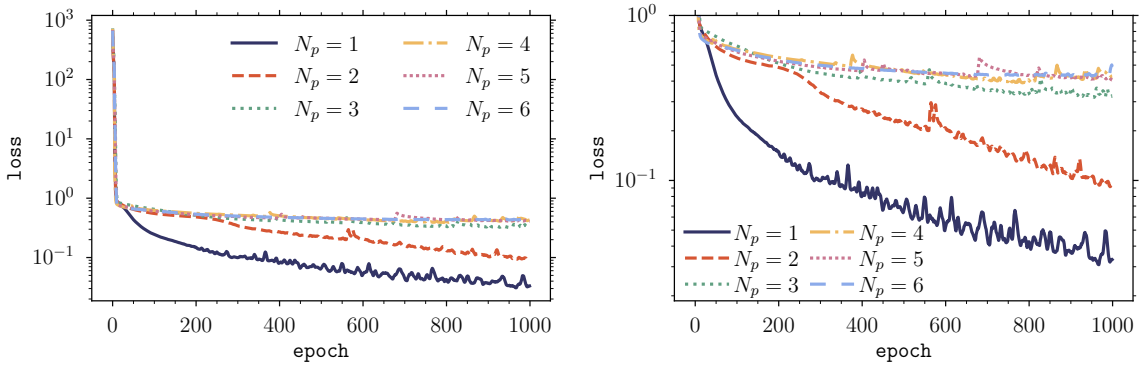
6.2.3 Performance of the model with N_p

Our first experiment tested the performance of the 34-layer ResNet model on 6 training datasets, where each dataset has a different fixed number of peaks N_p; the number of peaks is varied from $N_p = 1$ to $N_p = 6$. In this experiment, all datasets contain the same number of randomly generated examples, $N_b = 1.5 \cdot 10^5$.

We train the 34-layer ResNet architecture on each independent dataset for a total of 1000 epochs, which means that the network revisits all examples in the training datasets 1000 times in order to update its configuration. The number of epochs is chosen to allow the network to train for long enough computer time without having to wait long training times; training the model for 1000 epochs takes around 7 hours on a single Nvidia V100 GPU. In general, the total number of epochs should depend on the performance of the network in the training set, as well as in a test set. However, in order to allow a fair comparison between results extracted from different datasets, we fix the number of training epochs to 1000 for all datasets considered.

At each training epoch, the input and target training datasets are divided into paired stochastic mini-batches of 16448 examples each. For all training datasets, the algorithm employed to perform back-propagation is AdamW [34]; several other algorithms were tested, but no significant variation in performance was spotted. Before training, the learning rate is set of $\alpha = 0.01$, but its value is reduced at every epoch using the following recursive formula:

$$\alpha \leftarrow 0.997\, \alpha. \tag{6.30}$$



(a) Unrestricted y-scale **(b)** Restricted y-scale

Figure 6.6: Average empirical training loss as a function of the number of epochs. The training datasets contain the same number of examples, $N_b = 1.5 \cdot 10^5$, but different number of peaks N_p. The data does not include the weight-decay term in eq. (6.29).

The empirical losses at training time as a function of the number of epochs are presented in Figure (6.6). As the data shows, the neural network seems to struggle at learning the correct mapping as the number of peaks increases; there exists a major difference between the empirical losses at $N_p = 1, 2$, and greater number of peaks. In fact, the performance of the model saturates as N_p increases. Some visual reconstruction examples are provided in Appendix (C), which suggest that the network fails at learning to consistently resolve all peaks but the lowest energy ones.

Figure (6.6) suggests that training the neural network for longer could enhance the results over 1-peak and 2-peaks datasets, as the loss function does not seem to have plateaued in those particular cases. Varying the learning rate periodically [35] could also help to reduce the empirical loss function values in those particular datasets. In contrast, larger N_p datasets seem to plateau quite early, which supports the idea that the network suffers at learning the correct mapping as N_p increases.

The drastic variations spotted in the empirical training losses in Figure (6.6) might be caused by different problems. For instance, they can be a sign of exploding gradients, which are sometimes common in deep learning architectures. As a way of controlling this possible problem, we clip all network parameters' gradients using

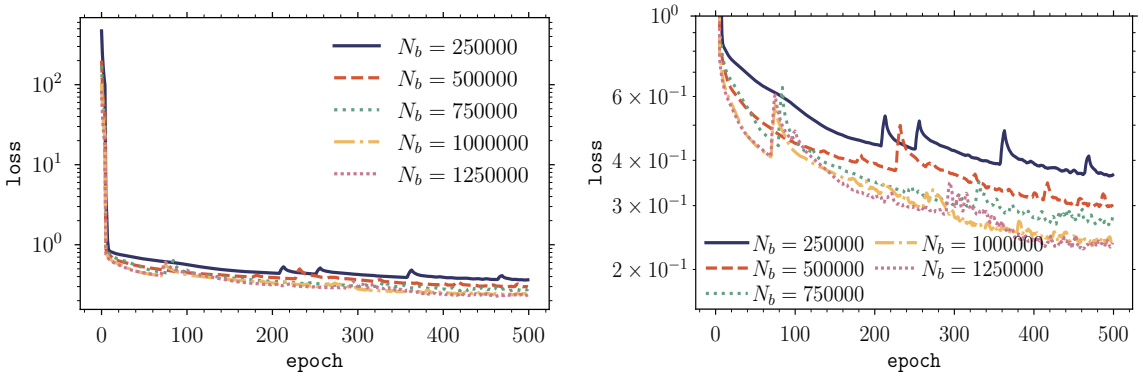
$$\nabla_\theta \mathcal{L} \leftarrow \min(\nabla_\theta \mathcal{L}, 10).$$

The gradients are computed with respect to the loss function in eq. (6.25). The clipping value is arbitrary, and should be tuned for each task. Another possible source

of the sudden variations found in the empirical losses is the stochastic nature of the mini-batches, which is without any doubt causing the small oscillations encountered in all loss function curves. Mini-batches can cause large oscillations in the training loss if outliers are present in the training datasets. If several outliers are randomly sampled into a single mini-batch, then they can lead to unexpected changes in the loss function gradients, thus incrementing the loss function abruptly. Furthermore, a non-optimal learning rate value can also cause sudden changes in the loss function.

6.2.4 Performance of the model with N_b

In order to demonstrate that the drop in performance with N_p is not caused by an insufficient number of training examples in the datasets, we perform another experiment. In this case, the same ResNet model is trained independently on several training datasets with different number of examples N_b. All datasets share the same number of peaks, $N_p = 3$. Due to the fact that training the 34-layer ResNet model on 1.25 million examples takes around 2 minutes per epoch on a single Nvidia V100 GPU, the model is only trained for 500 epochs in this experiment for all datasets considered.



(a) Unrestricted y-scale **(b)** Restricted y-scale

Figure 6.7: Average empirical training loss as a function of the number of epochs. All training datasets share the same number of peaks $N_p = 3$, but contain different number of examples N_b. The data does not include the weight-decay term in eq. (6.29).

The empirical losses at training for this experiment are presented in Figure (6.7). The results suggest that increasing the number of examples in the datasets is not a sufficient strategy to learn the correct reconstruction mapping: the empirical losses

are similar for all N_b, as the network only seems to be able to consistently resolve the lowest energy peak in each spectral functions. This is explicitly shown in the reconstruction examples provided in Appendix (C). The results in Figure (6.7) also suggest that increasing the training time could slightly improve the performance of the model in some datasets, as the loss function seems to not have reached a clear plateau for some particular datasets.

6.2.5 Analysis of the results

The empirical losses shown in Figure (6.6), combined with the reconstruction examples provided in Appendix (C), suggest that the network fails at learning the correct mapping as N_p increases. Therefore, it is important to understand the source of this drop in performance. Due to the extensive success of the ResNet architecture in a wide variety of complex tasks, we assume that it is expressive enough to not be the main source of the problem. In other words, an even larger and more complex model is also expected to struggle at learning the correct mapping as N_p increases. As a consequence, increasing the size of the model by making it deeper is not a viable strategy to improve the results in the proposed methodology. In addition, the results available in Figure (6.7) show that augmenting the number of examples in training datasets does not lead to a significant increase in the model performance. As a result, the inability of the neural network model to learn the correct large N_p mapping is most likely a consequence of the inherent properties of the methodology, that is, either the properties of the datasets or the basis expansion formulation.

It is worth mentioning that exploring the response of the model when trained over large datasets for a long time could grant us more information about the effects of both N_b and the number of epochs in the performance of the model. Additionally, testing the performance on the model over unseen examples is a required next step that would allow us to confirm that the model is indeed learning the internal features of the mapping and not just over-fitting the dataset. Some internal tests were performed in this direction while training the model, and the results showed that the network seemed to resolve the first low-energy peaks consistently on previously unseen examples.

From the analysis presented in Figure (6.5), we discard the possibility that the drop in performance is caused by a wrong choice of basis functions. The first 128 SVD basis functions are expressive enough to allow a high quality reconstruction of several independent spectral function datasets. This assumption is backed up by the data shown in Figure (6.8), which demonstrates that the average number of coefficients containing less than 90% of the total information in the dataset is independent of the

number of peaks in each spectral function: a similar number of relevant coefficients are required to describe the fundamental features of datasets with different number of peaks.

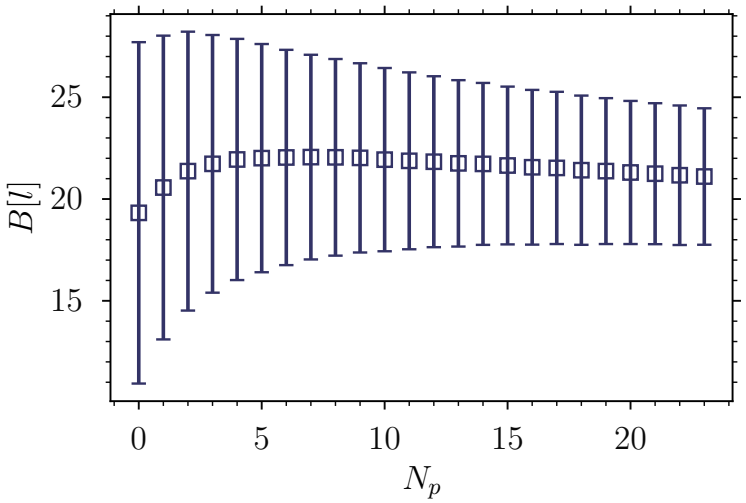


Figure 6.8: Average number of relevant coefficients and its standard deviation as a function of the number of peaks. In order to compute $B[l]$, we apply eq. (6.24) with $I_0 = 0.9$ on a coefficient dataset extracted from $\mathbf{R}[N_p, N_b = 50000]$. An independent set of basis functions is generated on each dataset.

Due to the fact that the saturation in the model performance spotted in Figure (6.6) is assumed to not be caused by the ResNet architecture, nor by the basis functions employed, and, additionally, increasing the number of examples in the training datasets does not seem to lead to a significant improvement in the reconstruction task, then the problem must be located in the properties of the training datasets, which ultimately means that the source of the problem is the inherent properties of the reconstruction task: the properties of the mapping.

In order to understand why the model struggles at learning the mapping as the number of peaks in each spectral function increases, as well as why the model is only able to consistently resolve the first peak in the spectral function, we analyse the properties of both input and target datasets. First, we start with the target space, represented by the SVD coefficients in our particular implementation: Figure (6.9) shows the variability of several target datasets as a function of the relevance of each coefficient, s; the variability is measured by the standard deviation of all coefficients in the dataset sharing the same basis function u_s. The data shows that the standard

deviation of the coefficients decays with s for a fixed number of peaks. This behaviour is expected from the results presented in Figure (6.4), and supported by Figure (6.8): less relevant coefficients tend to contain less information, and therefore, their absolute value is smaller. Additionally, Figure (6.9) demonstrates that the overall variability of the coefficients decays with the number of peaks. This implies that the spectral functions belonging to datasets with large number of peaks resemble each other more than the spectral functions belonging to low N_p datasets. As the landscape of possible spectral functions that can be generated through the combination of extremely large number of peaks in a fixed region of interest — see eq. (6.14) — decays with N_p, the standard deviation of the datasets saturates as N_p increases. This problem can be solved by increasing the region of interest in which the peaks are defined.

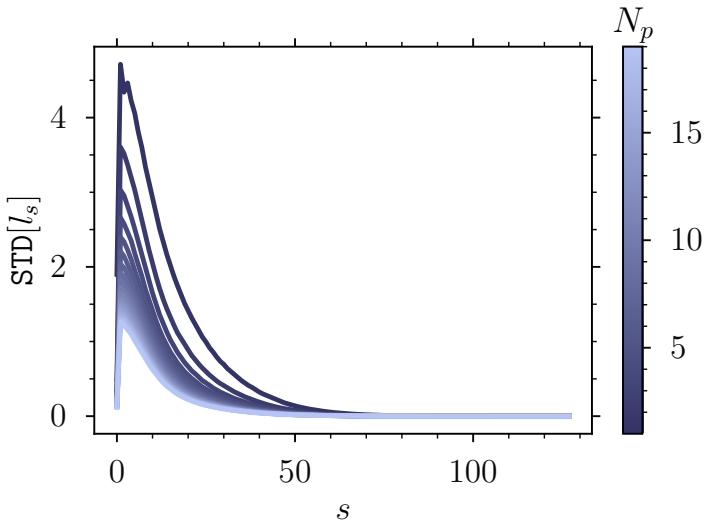


Figure 6.9: Estimated variability of different coefficient datasets computed through the standard deviation of all coefficients sharing the same basis function. All datasets employed contain $N_b = 50000$ examples. Only the first $N_s = 128$ coefficients are employed in the decompositions.

From the analysis performed on the coefficient datasets, we learn that the decrease in the performance with N_p cannot solely be caused by the properties of the target datasets. From a statistical point of view, the network should not find it more difficult to learn the fundamental features of low N_p datasets than large N_p datasets. In fact, large N_p datasets are statistically simpler than low N_p ones: the overall variability is lower for large N_p datasets than in low N_p ones, and, at the same time, the number of relevant coefficients is independent of N_p, as shown in Figure (6.8).

The same analysis can be applied to the input correlation function datasets; the results are presented in Figure (6.10). As in the previous analysis, the overall variability decays with the number of peaks. However, in this particular case, the overall standard deviation is around two orders of magnitude smaller than the ones found in the coefficient datasets with equivalent number of peaks. This implies that the correlation functions resemble each other more than the coefficients, which means that similar correlation functions might lead to completely different coefficients — ultimately, spectral functions. Moreover, the results demonstrate that the variability of the input datasets decays exponentially with τ, which is expected from eq. (6.1) and eq. (6.5).

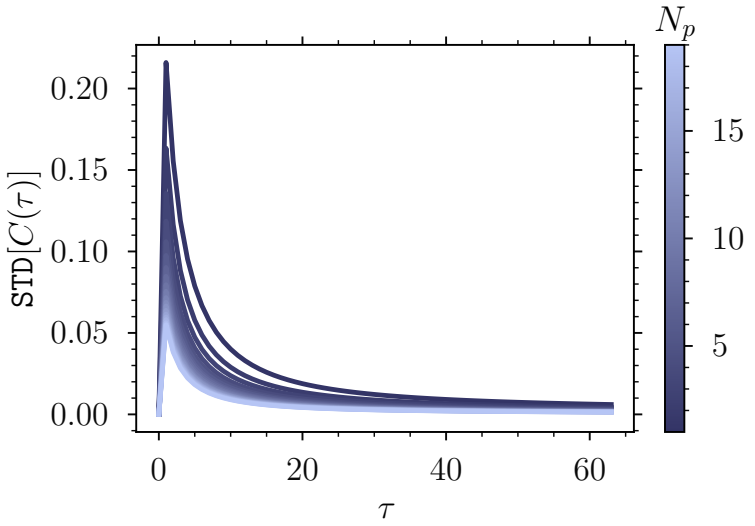
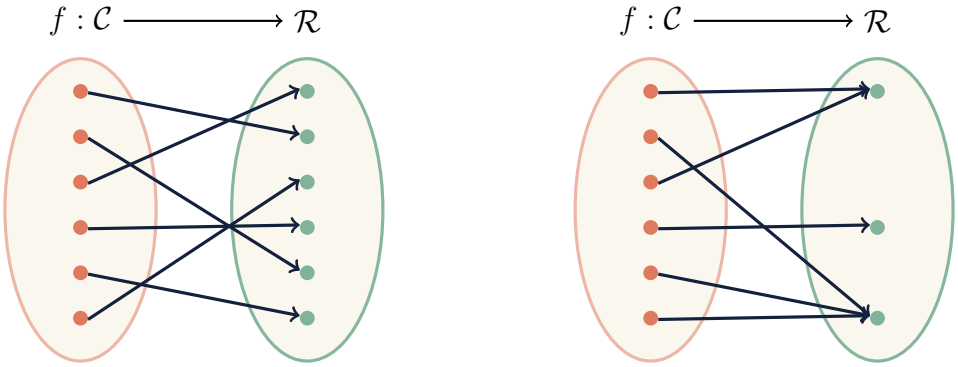


Figure 6.10: Estimated variability of different correlation function datasets computed through the standard deviation of all correlation functions at a given Euclidean time τ. All datasets employed contain $N_b = 50000$ examples.

The analysis performed on both datasets allow us to suggest some possible causes of the saturation in performance of the model with N_p. One of the causes is the loss of bijectivity of the numerical reconstruction mapping as the number of peaks in the datasets increases; a mapping is said to be bijective when there exists a one-to-one correspondence between input and output spaces; for each input, there is only one output; Figure (6.11) contains some diagrams depicting a bijective and a non-bijective mapping. We would like to stress that this loss of bijectivity is only present in the finite precision mapping learned by the neural network, the real underlying target mapping f is indeed injective, as eq. (6.1) is invertible.



(a) Bijective mapping **(b)** Non-bijective mapping

Figure 6.11: Diagram representing two abstract reconstruction mappings from the space of all correlation functions, $\mathcal{C}$, to the space of all spectral functions, $\mathcal{R}$. The figure on the left represents a bijective mapping, while the figure on the right represents a non-bijective mapping.

As N_p grows, there are multiple highly similar correlation functions whose spectral functions are completely different, and, at finite machine precision, the neural network is not able to differentiate them. This explains why the network is able to learn the correct mapping on 1-peak datasets, and even 2-peak datasets, where the probability of generating similar input correlation functions from completely different spectral functions is small. As the number of peaks increases, the chance of creating similar input correlation functions increases, which leads to a clear loss of bijectivity in the network mapping. This problem is aggravated by the exponential nature of the mesonic kernel. As peaks centred at $\omega \gg 0$ are exponentially suppressed, their contribution to the correlation function is negligible as τ increases. A visual analysis of this problem is provided in Appendix (D).

Although the neural network model is not able to resolve all peaks in the spectral function as N_p grows, it seems to consistently locate the correct lowest energy peak, as demonstrated in the results provided in Appendix (C). This is another consequence of the suppressed contribution of higher-order states to the correlation function. As the network is not able to discern between highly similar correlation functions, it treats all of them as equivalent samples. As a result, the network learns to produce the sample average of all coefficients sharing highly similar correlation function. Due to the fact that lower-energy states dominate, then their associated spectral functions are likely to share the same low-energy peaks. As a result, the sample average of those spectral functions is another spectral function with the correct communal low-energy dominant peaks; all non-shared high-energy peaks vanish in the sample

average. The same argument applies to the coefficients, as they are just another representation of the spectral functions. It is important to note that the network learns the average of all coefficients sharing similar correlation functions because it is being trained on the mean-squared error loss function defined in eq. (6.25), whose minimum corresponds to the sample average of each independent coefficient. Exploring different loss functions might help to alleviate this problem. Figure (6.12) demonstrates this idea.

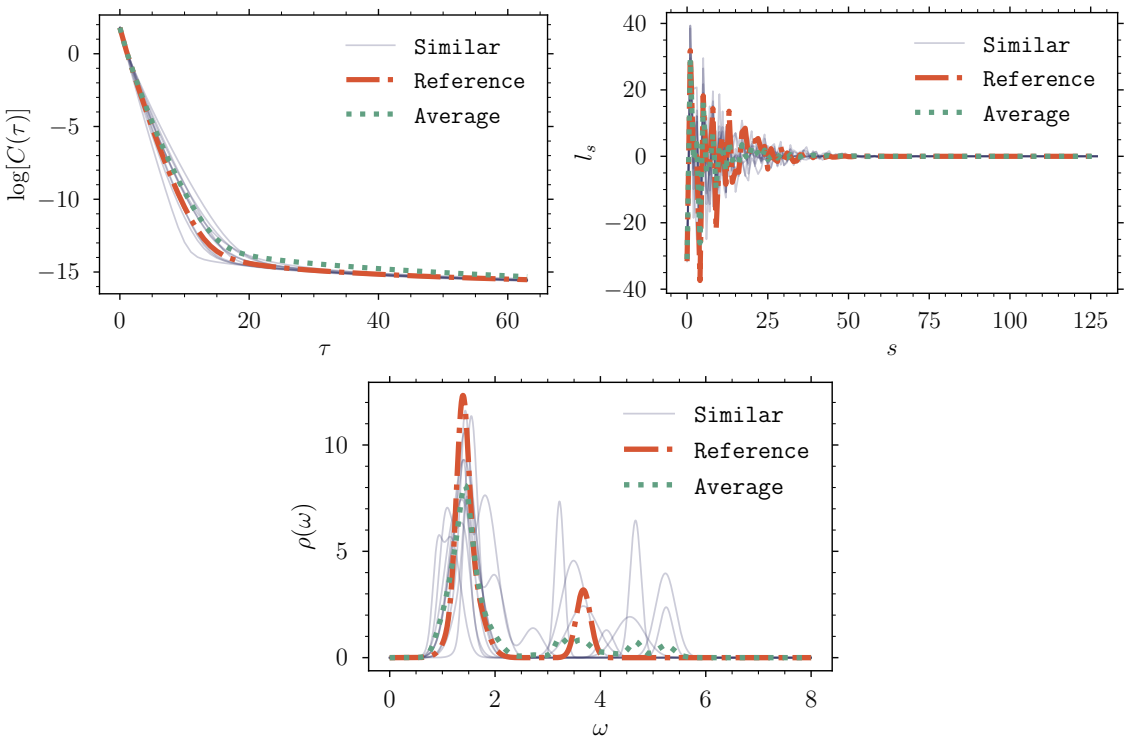


Figure 6.12: Examples of statistically equivalent correlation functions, their first 128 SVD coefficients, and their associated spectral functions. For more information about the definition of statistical equivalence, see Appendix (D).

In order to generate Figure (6.12), a reference correlation function was randomly chosen; the correlation function was sampled using a spectral function generated by randomly picking 4 bell-shaped peaks in the region of interest defined in eq. (6.14). Using this correlation function as a reference, we found 10 statistically equivalent correlation functions; for more information about our definition of statistical equivalency, we refer to Appendix (D). Once all 10 highly similar correlation functions

are found, we compute the sample average of their first 128 SVD coefficients using

$$l_s^{\text{avg}} = \frac{1}{10}\sum_{i=1}^{10} l_s^{\text{sim}}. \tag{6.31}$$

The spectral function of the sequence of average coefficients can be obtained by applying eq. (6.19) with the correct set of 128 SVD basis functions. Furthermore, the associated average correlation function can also be computed by integrating the spectral relation with the previously reconstructed spectral function.

6.2.6 Conclusions

From the results obtained, we can state that the proposed mapping methodology can only be truly efficient when applied to resolve the first low-energy peaks of the spectral functions. Higher order states are inaccessible due to the loss of bijectivity in the finite precision reconstruction mapping learned by the network, as well as the exponential decay of the input signal. Although this constraint hinders the generalisability of the methodology, it can also be used in our favour. Only low N_p datasets are required to extract the low-energy features of correlation functions, which implies that we only need to focus on generating datasets that mimic the low-energy region of the target spectral function space. A possible way of generating such complex datasets might involve varying the number of peaks in each spectral function, using different types of peaks, and employing non-parametric peaks. A neural network can be trained on these newly generated datasets in order to learn the correct low-energy reconstruction mapping. After training the model, it can be used to produce an approximate estimate of the spectral function low energy region from real input correlation functions. The outcome of the neural network can then be used as prior information for more involved methodologies. It is worth stressing that although neural networks are expensive to train, once trained, they can process large amounts of data in relatively short time.

In conclusion, spectral reconstruction is even more complex than it seems, and it cannot be completely controlled by brute force. This explains why some recent research is focusing on understanding the problem from the ground up [36].

There are different lines of research that could extend the studies presented in this document. For instance, neural networks models specifically tailored to deal with functional data structures might be of use in spectral reconstruction, as both C and ρ depend on a set of parameters: τ and ω respectively. Neural networks that deal with functional data structures are usually called functional neural networks; for more information about them, we refer to [37] and references therein. Related

to functional neural networks, one can explore the application of operator learning networks in spectral reconstruction [38–40]. Neural networks are known to be universal approximators of non-linear continuous functions. However, they also act as universal approximators of non-linear functionals, that is, mappings from a space of functions to the real line, and non-linear operators, which are mappings from a space of functions to another space of functions [41, 42]. This second case is relevant to spectral reconstruction, as eq. (6.1) connects a space of functions, $\mathcal{C}$, with another space of functions, $\mathcal{R}$. Recent research has produced impressive results applying deep neural networks to solve partial differential equations and inverse problems in fluid mechanics and other relevant fields [38, 39]. Another possible continuation of the presented results would imply the application of the proposed methodology to real lattice correlation functions. The results could be compared with other popular methodologies, such as MEM. For this specific task, the neural network could be trained on large and complex datasets composed by spectral functions with low number of peaks. The inaccessible high-frequency region can be safely avoided in the dataset generation.

A different possibility would be to study spectral reconstruction using a probabilistic approach in which the posterior probability of measuring ρ having already measured C is modelled. In this context, one could apply several neural network frameworks to model the posterior probability distribution function. For example, Bayesian neural networks [43–47] could be explored as a way of parametrising the target probability distribution function. A different approach would be to explore infinite-width neural networks [48–50], which are known to behave as Gaussian processes with learnable kernels.

As a way of reducing the degrees of freedom involved in the reconstruction and, at the same time, avoiding manually choosing a set of basis functions used to represent each spectral functions, one could try exploring the application of Variational Autoencoders (VAEs) [51, 52] to spectral reconstruction. VAEs are known to extract the fundamental features of complex datasets, which could allow the compression of both input and output signals with high efficiency.

To conclude, it is worth mentioning that attention-based architectures, such as the Transformer [53], have become very popular in the last few years. Due to their ability to take into account the relationships present in sequences, they are able to produce previously inaccessible results in multiple complex tasks [54, 55]. In spectral reconstruction, they could be applied to reduce the problems associated with the exponential decay present in the correlation functions: as an attention-based model is able to focus on different part of the input signal, it might be able to access the high-energy information contained in different regions of the input correlation

functions. Therefore, it might be able to extract different information from different parts of the input signal: for example, the $\tau \to 0$ region is known to be polluted by high-frequency states, while the $\tau \to \infty$ region is dominated by the lowest energy modes.

Chapter 6. References

[1] M. Asakawa, T. Hatsuda and Y. Nakahara. "Maximum entropy analysis of the spectral functions in lattice QCD". In: *Prog. Part. Nucl. Phys.* 46 (2001), pp. 459–508. DOI: 10.1016/S0146-6410(01)00150-8. arXiv: hep-lat/0011040.

[2] Gert Aarts and Jose Maria Martinez Resco. "Transport coefficients, spectral functions and the lattice". In: *JHEP* 04 (2002), p. 053. DOI: 10.1088/1126-6708/2002/04/053. arXiv: hep-ph/0203177.

[3] Harvey B. Meyer. "Transport Properties of the Quark-Gluon Plasma: A Lattice QCD Perspective". In: *Eur. Phys. J. A* 47 (2011), p. 86. DOI: 10.1140/epja/i2011-11086-3. arXiv: 1104.3708 [hep-lat].

[4] Alexander Rothkopf. "Heavy Quarkonium in Extreme Conditions". In: *Phys. Rept.* 858 (2020), pp. 1–117. DOI: 10.1016/j.physrep.2020.02.006. arXiv: 1912.02253 [hep-ph].

[5] Scott Elias and David Alderton. *Encyclopedia of Geology*. Academic Press, 2020.

[6] C Thurber and J Ritsema. "Theory and observations-seismic tomography and inverse methods". In: *Seismology and the Structure of the Earth* 1 (2007), pp. 323–360.

[7] M Zuhair Nashed, Jerry Zuhair Spinelli and Otmar Scherzer. *Inverse Problems, Image Analysis, and Medical Imaging: AMS Special Session on Interaction of Inverse Problems and Image Analysis, January 10-13, 2001, New Orleans, Louisiana*. Vol. 313. American Mathematical Soc., 2002.

[8] Hui Shao and Anders W. Sandvik. "Progress on stochastic analytic continuation of quantum Monte Carlo data". In: (Feb. 2022). arXiv: 2202.09870 [cond-mat.str-el].

[9] M. Asakawa, T. Hatsuda and Y. Nakahara. "Maximum entropy analysis of the spectral functions in lattice QCD". In: *Prog. Part. Nucl. Phys.* 46 (2001), pp. 459–508. DOI: 10.1016/S0146-6410(01)00150-8. arXiv: hep-lat/0011040.

[10] Thomas Spriggs et al. "A comparison of spectral reconstruction methods applied to non-zero temperature NRQCD meson correlation functions". In: *EPJ Web Conf.* 258 (2022), p. 05011. DOI: 10.1051/epjconf/202225805011. arXiv: 2112.04201 [hep-lat].

[11] Gert Aarts et al. "Spectral functions at small energies and the electrical conductivity in hot, quenched lattice QCD". In: *Phys. Rev. Lett.* 99 (2007), p. 022002. DOI: 10.1103/PhysRevLett.99.022002. arXiv: hep-lat/0703008.

[12] Alessandro Amato et al. "Electrical conductivity of the quark-gluon plasma across the deconfinement transition". In: *Phys. Rev. Lett.* 111.17 (2013). DOI: 10.1103/PhysRevLett.111.172001. arXiv: 1307.6763 [hep-lat].

[13] Gert Aarts et al. "Electrical conductivity and charge diffusion in thermal QCD from the lattice". In: *JHEP* 02 (2015), p. 186. DOI: 10.1007/JHEP02(2015)186. arXiv: 1412.6411 [hep-lat].

[14] G. Aarts et al. "What happens to the Υ and η_b in the quark-gluon plasma? Bottomonium spectral functions from lattice QCD". In: *JHEP* 11 (2011), p. 103. DOI: 10.1007/JHEP11(2011)103. arXiv: 1109.4496 [hep-lat].

[15] G. Aarts et al. "Melting of P wave bottomonium states in the quark-gluon plasma from lattice NRQCD". In: *JHEP* 12 (2013), p. 064. DOI: 10.1007/JHEP12(2013)064. arXiv: 1310.5467 [hep-lat].

[16] Gert Aarts et al. "The bottomonium spectrum at finite temperature from $N_f = 2 + 1$ lattice QCD". In: *JHEP* 07 (2014), p. 097. DOI: 10.1007/JHEP07(2014)097. arXiv: 1402.6210 [hep-lat].

[17] Jan Horak et al. "Reconstructing QCD spectral functions with Gaussian processes". In: *Phys. Rev. D* 105.3 (2022), p. 036014. DOI: 10.1103/PhysRevD.105.036014. arXiv: 2107.13464 [hep-ph].

[18] Sam Offler et al. "Reconstruction of bottomonium spectral functions in thermal QCD using Kernel Ridge Regression". In: *PoS* LATTICE2021 (2022), p. 509. DOI: 10.22323/1.396.0509. arXiv: 2112.02116 [hep-lat].

[19] Ben Page et al. "Spectral Reconstruction in NRQCD via the Backus-Gilbert Method". In: *PoS* LATTICE2021 (2022), p. 134. DOI: 10.22323/1.396.0134. arXiv: 2112.02075 [hep-lat].

[20] Olav F Syljuåsen. "Using the average spectrum method to extract dynamics from quantum Monte Carlo simulations". In: *Physical Review B* 78.17 (2008), p. 174429.

[21] Yan Qi Qin et al. "Amplitude Mode in Three-Dimensional Dimerized Antiferromagnets". In: *Phys. Rev. Lett.* 118.14 (2017), p. 147207. DOI: 10.1103/PhysRevLett.118.147207. arXiv: 1610.05164 [cond-mat.str-el].

[22] Hui Shao et al. "Nearly deconfined spinon excitations in the square-lattice spin-1/2 Heisenberg antiferromagnet". In: *Physical Review X* 7.4 (2017), p. 041072.

[23] Hélene Feldner et al. "Dynamical signatures of edge-state magnetism on graphene nano-ribbons". In: *Physical Review Letters* 106.22 (2011), p. 226401.

[24] S. -Y. Chen et al. "Machine learning spectral functions in lattice QCD". In: (Oct. 2021). arXiv: 2110.13521 [hep-lat].

[25] Lingxiao Wang, Shuzhe Shi and Kai Zhou. "Reconstructing spectral functions via automatic differentiation". In: (Nov. 2021). arXiv: 2111.14760 [hep-ph].

[26] Lukas Kades et al. "Spectral Reconstruction with Deep Neural Networks". In: *Phys. Rev. D* 102.9 (2020), p. 096001. DOI: 10.1103/PhysRevD.102.096001. arXiv: 1905.04305 [physics.comp-ph].

[27] Hervé Abdi. "Singular value decomposition (SVD) and generalized singular value decomposition". In: *Encyclopedia of measurement and statistics* (2007), pp. 907–912.

[28] H Andrews and CLIII Patterson. "Singular value decomposition (SVD) image coding". In: *IEEE transactions on Communications* 24.4 (1976), pp. 425–432.

[29] Michael E Wall, Andreas Rechtsteiner and Luis M Rocha. "Singular value decomposition and principal component analysis". In: *A practical approach to microarray data analysis*. Springer, 2003, pp. 91–109.

[30] Kaiming He et al. "Deep Residual Learning for Image Recognition". In: (Dec. 2015). DOI: 10.1109/CVPR.2016.90. arXiv: 1512.03385 [cs.CV].

[31] Xavier Glorot and Yoshua Bengio. "Understanding the difficulty of training deep feedforward neural networks". In: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings. 2010, pp. 249–256.

[32] Nitish Srivastava et al. "Dropout: a simple way to prevent neural networks from overfitting". In: *The journal of machine learning research* 15.1 (2014), pp. 1929–1958.

[33] Alex Labach, Hojjat Salehinejad and Shahrokh Valaee. "Survey of dropout methods for deep neural networks". In: *arXiv preprint arXiv:1904.13310* (2019).

[34] Ilya Loshchilov and Frank Hutter. *Decoupled Weight Decay Regularization*. 2017.

[35] Ilya Loshchilov and Frank Hutter. *SGDR: Stochastic Gradient Descent with Warm Restarts*. 2016. DOI: 10.48550/ARXIV.1608.03983.

[36] Shuzhe Shi, Lingxiao Wang and Kai Zhou. "Rethinking the ill-posedness of the spectral function reconstruction - why is it fundamentally hard and how Artificial Neural Networks can help". In: (Jan. 2022). arXiv: 2201.02564 [hep-ph].

[37] Barinder Thind, Kevin Multani and Jiguo Cao. "Deep learning with functional inputs". In: *Journal of Computational and Graphical Statistics* just-accepted (2022), pp. 1–27.

[38] Lu Lu, Pengzhan Jin and George Em Karniadakis. "Deeponet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators". In: *arXiv preprint arXiv:1910.03193* (2019).

[39] Zongyi Li et al. "Fourier neural operator for parametric partial differential equations". In: *arXiv preprint arXiv:2010.08895* (2020).

[40] Nicholas H Nelsen and Andrew M Stuart. "The random feature model for input-output maps between banach spaces". In: *SIAM Journal on Scientific Computing* 43.5 (2021), A3212–A3243.

[41] Tianping Chen and Hong Chen. "Approximation capability to functions of several variables, nonlinear functionals, and operators by radial basis function neural networks". In: *IEEE Transactions on Neural Networks* 6.4 (1995), pp. 904–910.

[42] Tianping Chen and Hong Chen. "Approximations of continuous functionals by neural networks with application to dynamic systems". In: *IEEE Transactions on Neural networks* 4.6 (1993), pp. 910–918.

[43] Laurent Valentin Jospin et al. "Hands-on Bayesian neural networks—A tutorial for deep learning users". In: *IEEE Computational Intelligence Magazine* 17.2 (2022), pp. 29–48.

[44] Charles Blundell et al. "Weight uncertainty in neural network". In: *International conference on machine learning*. PMLR. 2015, pp. 1613–1622.

[45] Anqi Wu et al. "Deterministic variational inference for robust bayesian neural networks". In: *arXiv preprint arXiv:1810.03958* (2018).

[46] Yeming Wen et al. "Flipout: Efficient pseudo-independent weight perturbations on mini-batches". In: *arXiv preprint arXiv:1803.04386* (2018).

[47] Ranganath Krishnan, Mahesh Subedar and Omesh Tickoo. "Specifying weight priors in bayesian deep neural networks with empirical bayes". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. 04. 2020, pp. 4477–4484.

[48] Matthias Seeger. "Gaussian processes for machine learning". In: *International journal of neural systems* 14.02 (2004), pp. 69–106.

[49] Roman Novak et al. "Neural tangents: Fast and easy infinite neural networks in python". In: *arXiv preprint arXiv:1912.02803* (2019).

[50] Jaehoon Lee et al. "Deep neural networks as gaussian processes". In: *arXiv preprint arXiv:1711.00165* (2017).

[51] Dor Bank, Noam Koenigstein and Raja Giryes. "Autoencoders". In: *arXiv preprint arXiv:2003.05991* (2020).

[52] Laurent Girin et al. "Dynamical variational autoencoders: A comprehensive review". In: *arXiv preprint arXiv:2008.12595* (2020).

[53] Ashish Vaswani et al. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017).

[54] Jacob Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *ArXiv* abs/1810.04805 (2019).

[55] Alec Radford et al. "Improving language understanding by generative pre-training". In: (2018).

Appendix A

Lattice setup

This appendix contains information about the lattice simulations employed in all lattice-related studies presented in this document. The configurations used to compute the correlation functions correspond to the anisotropic FASTSUM ensembles described in detail in Ref [1]. Those ensembles are based on the anisotropic lattice work of the HADSPEC collaboration at zero temperature [2, 3]. Our collection of ensembles is referred to by the name *Generation 2L* (Gen2L), where L stands for light. The definition of the anisotropic action governing the dynamics of the system is detailed in Ref [2]. Our ensembles contain $N_f = 2 + 1$ dynamic flavours, being the lightest quarks (u and d) degenerate and heavier than in nature; the strange quark mass is fixed to its physical value. Additionally, propagators for valence charm quarks are computed using the same relativistic action, as described in Ref [4, 5]. The heaviest quarks, bottom and top, are not accessible in our simulations. A collection of relevant parameters defining our ensembles can be found in Table A.1.

Table A.1: Relevant information about the Gen2L anisotropic FASTSUM ensembles: a_τ (a_s) is the temporal (spatial) lattice spacing; ξ the physical anisotropy; M_π the pion mass; and T_c is pseudocritical temperature, estimated via the inflection point of the renormalised chiral condensate [1]. The scale setting is discussed in detail in Ref [6].

$1/a_\tau$ MeV	a_s fm	$\xi = a_s/a_\tau$	N_s	M_π MeV	T_c MeV
6079(13)	0.1121(3)	3.453(6)	32	239(1)	166(2)

We adopt a fixed-scale approach in our simulations, which implies that the temporal lattice spacing is shared among all temperatures. As a result, the temperature

of the system can be modified through the variation of the size of the lattice temporal direction, labelled N_τ. The relationship between the temperature T, the lattice spacing a_τ, and the temporal extent of the lattice N_τ, is

$$T = \frac{1}{a_\tau \, N_\tau}. \tag{A.1}$$

More information about this relationship can be found in Chapter (2). In our simulations, a lattice of $N_\tau = 128$ sites leads to the lowest temperature available, $T = 47$ MeV; a lattice of size $N_\tau = 20$ generates our highest temperature, $T = 304$ MeV. Table (A.2) summarises all lattices simulated, their corresponding temperatures, and the number of configurations available at each temperature.

Table A.2: Table summarising all lattices available, as defined by the size of their temporal direction, N_τ. Each lattice represents a different temperature T, computed using eq. (A.1) with $a_\tau^{-1} = 6079$ MeV. Additionally, N_{conf} represents the number of configurations simulated at each temperature.

N_τ	128	64	56	48	40	36	32	28	24	20
T MeV	47	95	109	127	152	169	190	217	253	304
N_{conf}	1024	1041	1042	1123	1102	1119	1090	1031	1016	1030

All studies presented in this document focus on 2-point mesonic correlation function $C(y-x)$, constructed from two mesonic operators:

$$C(y-x) = \langle \hat{M}(y)\hat{M}^\dagger(x)\rangle, \tag{A.2}$$

where

$$\hat{M}(x) = \bar{\psi}_A(x)\Gamma(x)\psi_B(x) \tag{A.3}$$

and $\hat{M}^\dagger$ is the complex conjugate of $\hat{M}$. In the equation above, $\bar{\psi}_A$ represents an antiquark fermionic field of flavour A, and ψ_B a quark fermionic field of flavour B. Additionally, Γ is an operator acting on all relevant spaces, which dictates the quantum numbers of the mesonic operator $\hat{M}(x)$.

In a 2-point mesonic correlation function, the first operator, $\hat{M}^\dagger(x)$, is called the source operator, while the second one, $\hat{M}(y)$ is referred to as the sink operator. Due to translational invariance of systems in equilibrium, 2-point mesonic correlation functions are functions of the distance between sink and source operators: $C(y,x) = C(y-x)$. As a result, it is a standard practise to place the source operator at the origin of coordinates: $x = (\tau_x, \vec{x}) = (0, \vec{0})$.

All the analysis present in this document deal with thermal observables. All the correlation functions analysed are expressed in a time-momentum representation. We restrict ourselves to zero external momentum:

$$C(\tau, \vec{k} = \vec{0}) = \sum_{\vec{y}} \langle \hat{O}(\tau, \vec{y}) \hat{O}^{\dagger}(0, \vec{0}) \rangle. \tag{A.4}$$

This implies that our thermal correlation functions are only functions of the Euclidean time: $C(y - x) = C(\tau)$. Additionally, our correlation functions are $O(a^2)$-improved using the Symanzik improvement scheme for anisotropic lattices [2].

Due to the fact that we simulate three different flavours — recall $m_u = m_d$ — we are able to study six different mesonic flavour combinations: uu, us, uc, ss, sc and cc. Disconnected contributions to the mesonic correlation functions are not computed, which implies that the ground state contributing to the uu pseudoscalar correlation function is the charged pion.

In addition, for each available temperature, flavour combination and set of operators, we compute two different estimates of the same correlation function, which differ in the type of source used in the inversion of the Dirac operator. One estimate uses local sources — in effect delta functions — and the other one employs Gaussian smeared sources [7]. In order to apply smearing to a source vector G — see Chapter (1) — we apply the following functional transformation [8–12]:

$$G' = A\,(1 + \kappa H)^n G. \tag{A.5}$$

In the equation above: A is a normalisation constant, H is the spatial hopping part of the Dirac operator [2], and κ and n are two parameters controlling how smearing is applied: $n = 100$ and $\kappa = 5.5$. Additionally, the hopping term H contains APE-smeared links [13]. Smeared sources are specifically designed to decouple from the excited states contributing to the correlation function at small Euclidean time τ, thus enhancing the ground state signal. At low temperature, both types of source should yield similar estimates of the ground state mass.

Appendix A. References

[1] G. Aarts et al. "Properties of the QCD thermal transition with Nf=2+1 flavors of Wilson quark". In: *Phys. Rev. D* 105.3 (2022), p. 034504. DOI: 10.1103/PhysRevD.105.034504. arXiv: 2007.04188 [hep-lat].

[2] Robert G. Edwards, Balint Joo and Huey-Wen Lin. "Tuning for Three-flavors of Anisotropic Clover Fermions with Stout-link Smearing". In: *Phys. Rev. D* 78 (2008), p. 054501. DOI: 10.1103/PhysRevD.78.054501. arXiv: 0803.3960 [hep-lat].

[3] Huey-Wen Lin et al. "First results from 2+1 dynamical quark flavors on an anisotropic lattice: Light-hadron spectroscopy and setting the strange-quark mass". In: *Phys. Rev. D* 79 (2009), p. 034502. DOI: 10.1103/PhysRevD.79.034502. arXiv: 0810.3588 [hep-lat].

[4] Gavin K. C. Cheung et al. "Excited and exotic charmonium, D_s and D meson spectra for two light quark masses from lattice QCD". In: *JHEP* 12 (2016), p. 089. DOI: 10.1007/JHEP12(2016)089. arXiv: 1610.01073 [hep-lat].

[5] Liuming Liu et al. "Excited and exotic charmonium spectroscopy from lattice QCD". In: *JHEP* 07 (2012), p. 126. DOI: 10.1007/JHEP07(2012)126. arXiv: 1204.5425 [hep-ph].

[6] David J. Wilson et al. "The quark-mass dependence of elastic πK scattering from QCD". In: *Phys. Rev. Lett.* 123.4 (2019), p. 042002. DOI: 10.1103/PhysRevLett.123.042002. arXiv: 1904.03188 [hep-lat].

[7] S. Gusken et al. "Nonsinglet Axial Vector Couplings of the Baryon Octet in Lattice QCD". In: *Phys. Lett. B* 227 (1989), pp. 266–269. DOI: 10.1016/S0370-2693(89)80034-6.

[8] Gert Aarts et al. "Light baryons below and above the deconfinement transition: medium effects and parity doubling". In: *JHEP* 06 (2017), p. 034. DOI: 10.1007/JHEP06(2017)034. arXiv: 1703.09246 [hep-lat].

[9] Carleton E. Detar and John B. Kogut. "Measuring the Hadronic Spectrum of the Quark Plasma". In: *Phys. Rev. D* 36 (1987), p. 2828. DOI: 10.1103/PhysRevD.36.2828.

[10] Irina Pushkina et al. "Properties of hadron screening masses at finite baryonic density". In: *Phys. Lett. B* 609 (2005), pp. 265–270. DOI: 10.1016/j.physletb.2005.01.006. arXiv: hep-lat/0410017.

[11] Saumen Datta et al. "Nucleons near the QCD deconfinement transition". In: *JHEP* 02 (2013), p. 145. DOI: 10.1007/JHEP02(2013)145. arXiv: 1212.2927 [hep-lat].

[12] Gert Aarts et al. "Nucleons and parity doubling across the deconfinement transition". In: *Phys. Rev. D* 92.1 (2015), p. 014503. DOI: 10.1103/PhysRevD.92.014503. arXiv: 1502.03603 [hep-lat].

[13] M. Albanese et al. "Glueball Masses and String Tension in Lattice QCD". In: *Phys. Lett. B* 192 (1987), pp. 163–169. DOI: 10.1016/0370-2693(87)91160-9.

Appendix B

Studies on mesonic operators

This appendix contains some analysis performed on continuum mesonic operators. The contents are divided into two main sections: in the first one, we classify several operators by their quantum numbers, which allows us to specify the lowest energy modes contributing to 2-point correlation functions; in the second one, we include a brief study on the relationship between the vector and axial vector operators under $SU(A)_A$ chiral transformations. The first section simplifies the phenomenological interpretation of the correlation functions, while the second one opens a window to study chiral symmetry restoration in QCD as a function of the temperature.

We are mainly interested in mesonic 2-point correlation functions, which are composed by two mesonic operators. A mesonic operator is defined as

$$\hat{M}(x) = \bar{\psi}_A(x)\,\Gamma\,\psi_B(x), \tag{B.1}$$

where $\psi_A(x)$ represents a fermionic quark field of flavour A, $\bar{\psi}(x)$ a fermionic anti-quark field of flavour B, and Γ is an operator acting on all spaces to which the quark field belong; for example, spinor space, flavour space… The operator Γ defines the type of excitations generated by the pair of fermionic fields. It is worth mentioning that Γ could depend on the space-time, but we restrict ourselves to operators that are independent of the space-time coordinates. Mesonic operators are sometimes called mesonic densities or mesonic interpolators.

B.1 Classification of mesonic operators

To obtain the quantum number of a particular continuum operator under a given transformation, we study how the operator transforms under that transformation. For example, if a continuum operator $\hat{M}$ transforms under the operator $\hat{K}$ as $(-1)\,\hat{M}$,

then, we say that the quantum number of $\hat{M}$ under $\hat{K}$ is $K = (-1)$. It is then clear that the quantum numbers of a given operator grant us information about the object itself, thus allowing us to classify the operators in a clear and concise way. In addition, the quantum numbers indicate us which physical states contribute to a particular 2-point diagonal correlation function, as all states with the same quantum numbers to $\hat{M}$ will be excited.

In our particular case, the relevant quantum numbers used to classify the states are those associated with charge conjugation and parity transformations; we label the charge conjugation transformation with the label $\hat{C}$, while the parity one is labelled $\hat{P}$.

In order to study how eq. (B.1) transforms under both operators, first we need to know how fermionic fields transform under $\hat{C}$ and $\hat{P}$. These transformation rules can be derived by studying Lorentz invariance in field theory. Under charge conjugation, fermionic fields transform as

$$\hat{C}\psi(x) = \hat{C}\bar{\psi}^T(x), \quad \hat{C}\bar{\psi}(x) = -\psi^T(x)\hat{C}^{-1}. \tag{B.2}$$

While their transformation under parity is,

$$\hat{P}\psi(x) = \gamma_0\psi(x), \quad \hat{P}\bar{\psi}(x) = \bar{\psi}(x)\gamma_0. \tag{B.3}$$

Both transformations are not affected by the internal symmetries of the fields.

In order to extract the quantum numbers of different mesonic operators, it is useful to know some properties of the Dirac γ-matrices. First, through the combination of all γ-matrices, we can generate another matrix, labelled γ_5, and defined

$$\gamma_5 = \gamma_0\gamma_1\gamma_2\gamma_3. \tag{B.4}$$

The following properties of the Dirac γ-matrices are required in the derivation:

$$\gamma_\mu = \gamma_\mu^T = \gamma_\mu^{-1}; \quad \gamma_\mu^2 = \gamma_5^2 = \mathbb{1}. \tag{B.5}$$

Furthermore, these anti-commuting properties of the Dirac γ-matrices play an important role in the analysis:

$$\{\gamma_\mu, \gamma_\nu\} = 2\delta_{\mu\nu}; \quad \{\gamma_\mu, \gamma_5\} = 0. \tag{B.6}$$

To conclude, some useful identities relating γ-matrices and the charge conjugation operator $\hat{C}$ are:

$$\hat{C}^{-1}\gamma_\mu\hat{C} = -\gamma_\mu^T, \quad \hat{C}^{-1}\gamma_5\hat{C} = \gamma_5^T. \tag{B.7}$$

There exist a collection of important relationships between the charge conjugation and the parity quantum numbers, labelled C and P respectively, and the quantum numbers of angular momentum and spin, labelled L and S. These relationships emerge in the wave-mechanical quark model, and serve as a connection between the states contributing to quantum field theory 2-point correlation function, the quark model, and experimentally measurable states. The before-mentioned identities are:

$$C = (-1)^{L+S}; \quad P = (-1)^{L+1}. \tag{B.8}$$

In our case, the spin S is 0 or 1 due to the fact that mesons are bounded states of two fermionic fields, each with spin $\frac{1}{2}$, which implies $\frac{1}{2} \otimes \frac{1}{2} = 0, 1$. In principle, L can be any integer. Through the combination of both spin and angular momentum, we can generate the total angular momentum quantum number: $J = L + S$.

Studying how different mesonic operators transform under $\hat{C}$ and $\hat{P}$ is just a matter of applying the rules defined above to different mesonic operators. Note that different mesonic operators can be constructed by using different Γ operators. In this particular case, as $\hat{C}$ and $\hat{P}$ do not act on the flavour space, we can safely assume $\Gamma = \gamma$, being γ one of the Dirac γ-matrices. We are mainly interested in four different operators, which we call: the scalar operator, with $\Gamma = \mathbb{1}$; the pseudoscalar operator, $\Gamma = \gamma_5$; the vector operator, $\Gamma = \gamma_i$ with $i = 1, 2, 3$; and the axial vector operator, $\Gamma = \gamma_i\gamma_5$, with $i = 1, 2, 3$.

We do not explicitly derive the quantum numbers of all operators. Nevertheless, to show how one could extract the quantum numbers of the operators, we will analyse the vector operator in detail. To do so, we start by defining the vector mesonic operator:

$$\hat{V}_i(x) = \bar{\psi}_A(x)\gamma_i\psi_B(x). \tag{B.9}$$

Under parity, this operator transforms as

$$\hat{P}\left[\bar{\psi}_A(x)\gamma_i\psi_B(x)\right] = -\bar{\psi}_A(x)\gamma_i\psi_B(x), \tag{B.10}$$

which implies that the parity quantum number is $P = (-1)$. Under charge conjugation, $\hat{V}_i(x)$ transforms as

$$\hat{C}\left[\bar{\psi}_A(x)\gamma_i\psi_B(x)\right] = -\bar{\psi}_A(x)\gamma_i\psi_B(x), \tag{B.11}$$

which implies that $C = (-1)$ for the vector operator. For the charge conjugation operator to be defined, both fermionic fields must have the same flavour combination: $\bar{\psi}_A$, ψ_A. Typically, the operators are identified with the lowest energy modes they can excite. To obtain the lowest energy mode of the vector operator, we can use eq. (B.8)

with $L = 0$, which leads to $S = 1$. Therefore, our analysis has revealed that the lowest mode of the vector operator corresponds to a spin 1 meson; experimentally, the lowest energy mode is referred to as the $\rho(770)$ state. The complete set of quantum numbers for the relevant operators can be found in Table (B.1).

Table B.1: Main mesonic operators studied in our simulations, indicated by their names, defining operators Γ, and their J^{PC} quantum numbers.

Channel	Pseudoscalar	Vector	Axial-vector	Scalar
Operator	γ_5	γ_μ	$\gamma_\mu\gamma_5$	$\mathbb{1}$
J^{PC}	0^{-+}	1^{--}	1^{++}	0^{++}

Table (B.2) displays the lowest energy mode of each channel studied: scalar, pseudoscalar, vector and axial-vector. Each flavour combination possesses a different lowest energy mode. The names of the states are taken from the most recent Particle Data Group (PDG) database [1] using the information displayed in Table (B.1) as a reference.

Table B.2: Table containing the names of the lowest physical energy modes of each non-singlet mesonic operator produced in our simulations. As the u and d quarks are degenerate in our simulations, we employ the label uu to refer to the non-singlet ud flavour combination.

Name	uu	us	ss	uc	sc	cc
Scalar	$a_0(1450)$	$K_0^\star(1430)$	$f_0(1710)/f_0(1370)$	$D_0^*(2400)$	$D_{s0}^*(2317)^\pm$	$\chi_{c0}(1P)$
Pseudoscalar	$\pi^\pm$	K	$\eta/\eta'(958)$	D	$D_s^\pm$	$\eta_c(1S)$
Vector	$\rho(770)$	$K^\star(892)$	$\phi(1020)/\omega(782)$	D^*	$D_s^{*\pm}$	$J/\psi(1S)$
Axial vector	$a_1(1260)$	K_{1A}	$f_1(1420)/f_1(1285)$	$D_1(2430)$	$D_{s1}(2460)^\pm$	$\chi_{c1}(1P)$

B.2 $SU(2)_A$ related mesonic operators

In this section, we analyse the relationship between the vector and the axial vector mesonic operators under $SU(2)_A$ chiral transformations. The existence of such a relationship opens a window to study the restoration of chiral symmetry as a function of the temperature through the analysis of diagonal 2-point correlation functions composed by the vector and axial vector densities.

As we focus on $SU(2)_A$ chiral transformations, we restrict the discussion to the lightest flavour combination available: *ud*. In addition, we assume that $SU(2)_A$ chiral symmetry is restored and, as a consequence, the light sector quarks can be treated as an $SU(2)$ doublet:

$$\psi = \begin{pmatrix} u \\ d \end{pmatrix}; \quad \bar{\psi} = \begin{pmatrix} \bar{u} & \bar{d} \end{pmatrix}. \tag{B.12}$$

We are interested in the following vector and axial mesonic operators:

$$\hat{V}_i^a(x) = \bar{\psi}(x)\,\gamma_i\tau_a\,\psi(x); \quad \hat{A}_i^a(x) = \bar{\psi}(x)\,\gamma_i\gamma_5\tau_a\,\psi(x) \quad i = 1,2,3. \tag{B.13}$$

The τ matrices are the three generators of $SU(2)$, called Pauli matrices; they fulfil the following properties:

$$\{\tau_a, \tau_b\} = 2\,\delta_{ab}; \quad [\tau_a, \tau_b] = 2i\,\varepsilon_{abc}\,\tau_c. \tag{B.14}$$

Under $SU(2)_A$ transformations, the quark field doublet transforms as

$$\psi \to e^{+i\,\gamma_5\,\frac{\vec{\tau}\,\vec{\theta}}{2}}\psi \simeq (1 + i\,\gamma_5\,\frac{\vec{\tau}\,\vec{\theta}}{2})\,\psi, \tag{B.15}$$

$$\bar{\psi} \to \bar{\psi}e^{+i\,\gamma_5\,\frac{\vec{\tau}\,\vec{\theta}}{2}} \simeq \bar{\psi}\,(1 + i\,\gamma_5\,\frac{\vec{\tau}\,\vec{\theta}}{2}), \tag{B.16}$$

where $\vec{\theta}$ is a parameter of the transformation. The presence of $\vec{\tau}$ in the transformation implies that the axial transformation acts on the flavour space of the mesonic operator: see eq. (B.12).

We can apply an infinitesimal axial transformation to the vector operator defined in eq. (B.13) to obtain

$$\bar{\psi}\,\gamma_i\tau_a\,\psi \xrightarrow{SU(2)_A} \bar{\psi}\,\gamma_i\tau_a\,\psi + \varepsilon_{abc}\,\theta_a\,\bar{\psi}\,\gamma_i\gamma_5\tau_c\,\psi, \tag{B.17}$$

which can be expressed as

$$\hat{V}_i^a \xrightarrow{SU(2)_A} \hat{V}_i^a + \epsilon_{abc}\,\theta_a\hat{A}_i^c. \tag{B.18}$$

In the limit of restored chiral symmetry, the vector mesonic operator has contributions from the axial-vector density. Consequently, provided that chiral symmetry is restored, we expect to find degeneracies between diagonal correlation functions in the vector and the axial vector channels.

Appendix B. References

[1] P. A. Zyla et al. "Review of Particle Physics". In: *PTEP* 2020.8 (2020). DOI: 10.1093/ptep/ptaa104.

Appendix C

Spectral reconstruction examples

This appendix contains some randomly selected reconstruction examples extracted from different training datasets. The examples are drawn from the same datasets used to train the 34-layer ResNet in Chapter (6). The examples do not represent predictions of the network, as the model has already been exposed to the data while training.

The examples are meant to display the performance of the neural network model on the task of learning the correct reconstruction mapping in a particular dataset. The results are sampled using the network configuration at the last training epoch.

All figures presented in this appendix show three randomly selected examples for each dataset. For each example, the expected and predicted coefficients are shown:

$$l_s \quad \text{and} \quad \hat{l}_s.$$

In addition, their associated spectral functions are also contained in the examples:

$$\rho(\omega) \quad \text{and} \quad \hat{\rho}(\omega).$$

The spectral functions are generated using eq. (6.19), where the basis functions employed in the expansion correspond to the $N_s = 128$ most relevant basis function generated from the SVD decomposition of the spectral function training dataset: $\mathbf{R}[N_b, N_p]$.

In the first section, we present some reconstruction examples extracted from a collection of datasets containing the same number of examples, but variable number of peaks. The 34-layer ResNet model was trained over all those datasets for 1000 epochs. The second section contains some examples extracted from a collection of datasets sharing the same number of peaks, but variable number of examples. The model was trained for 500 epochs in this second case. More information about the model, and the training setup, can be found in Chapter (6).

C.1 Fixed N_b, variable N_p

The reconstruction examples presented in this section are extracted from the experiment shown in Figure (6.6). All training datasets share the same number of examples, $N_b = 1.5 \cdot 10^5$, but each of them contain spectral functions with different number of peaks N_p.

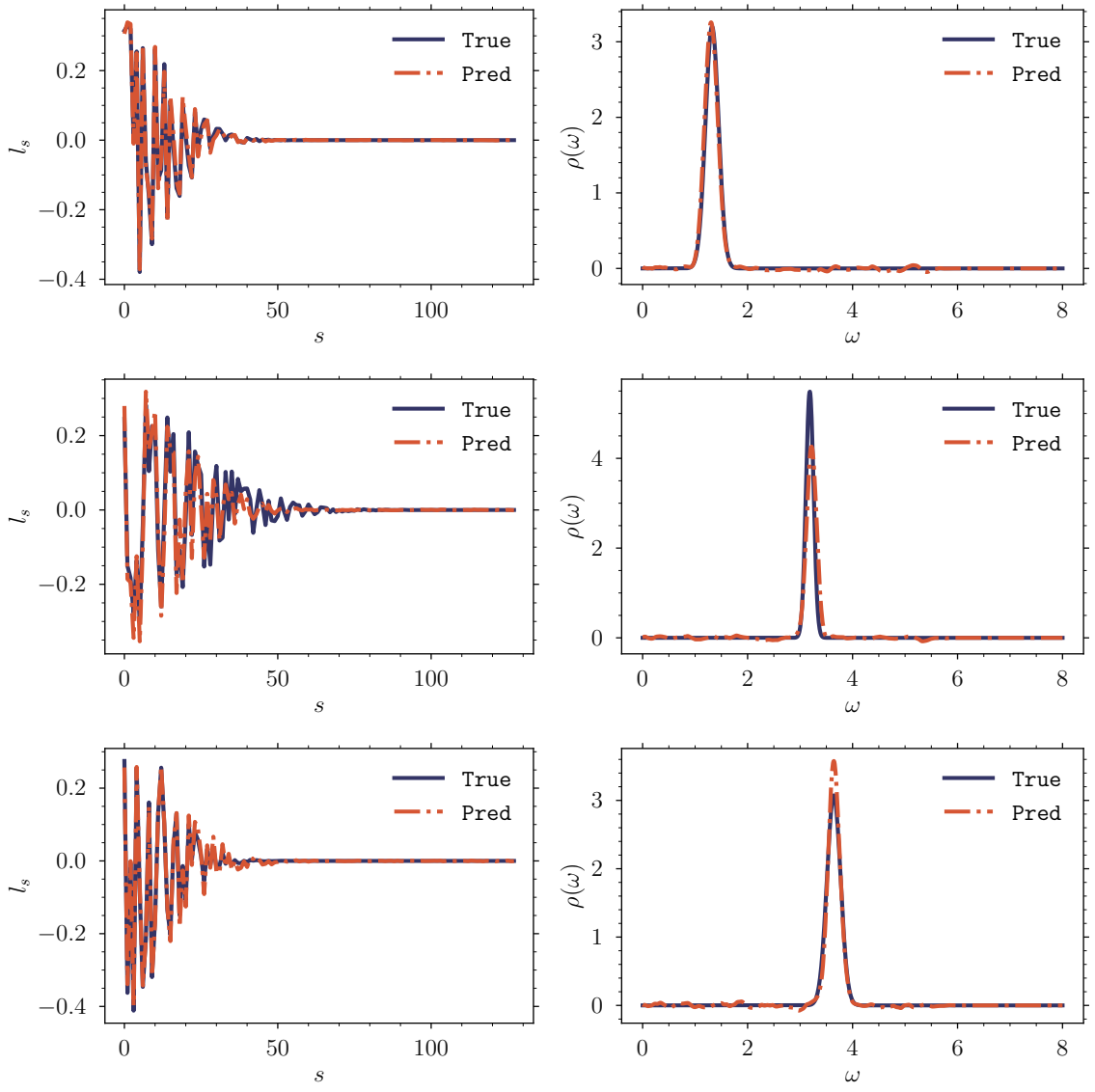


Figure C.1: Examples extracted from a dataset with $N_p = 1$

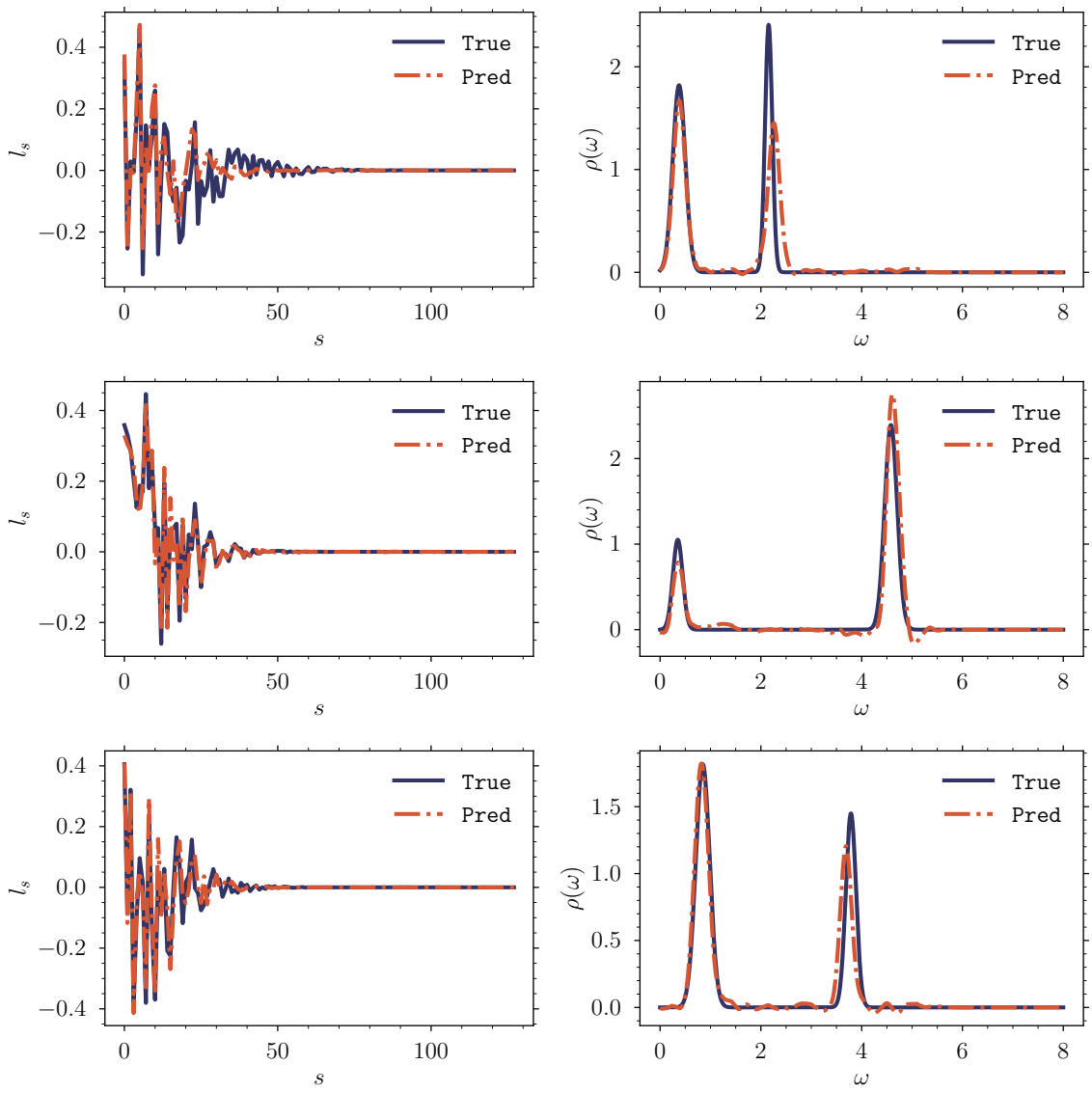


Figure C.2: Examples extracted from a dataset with $N_p = 2$

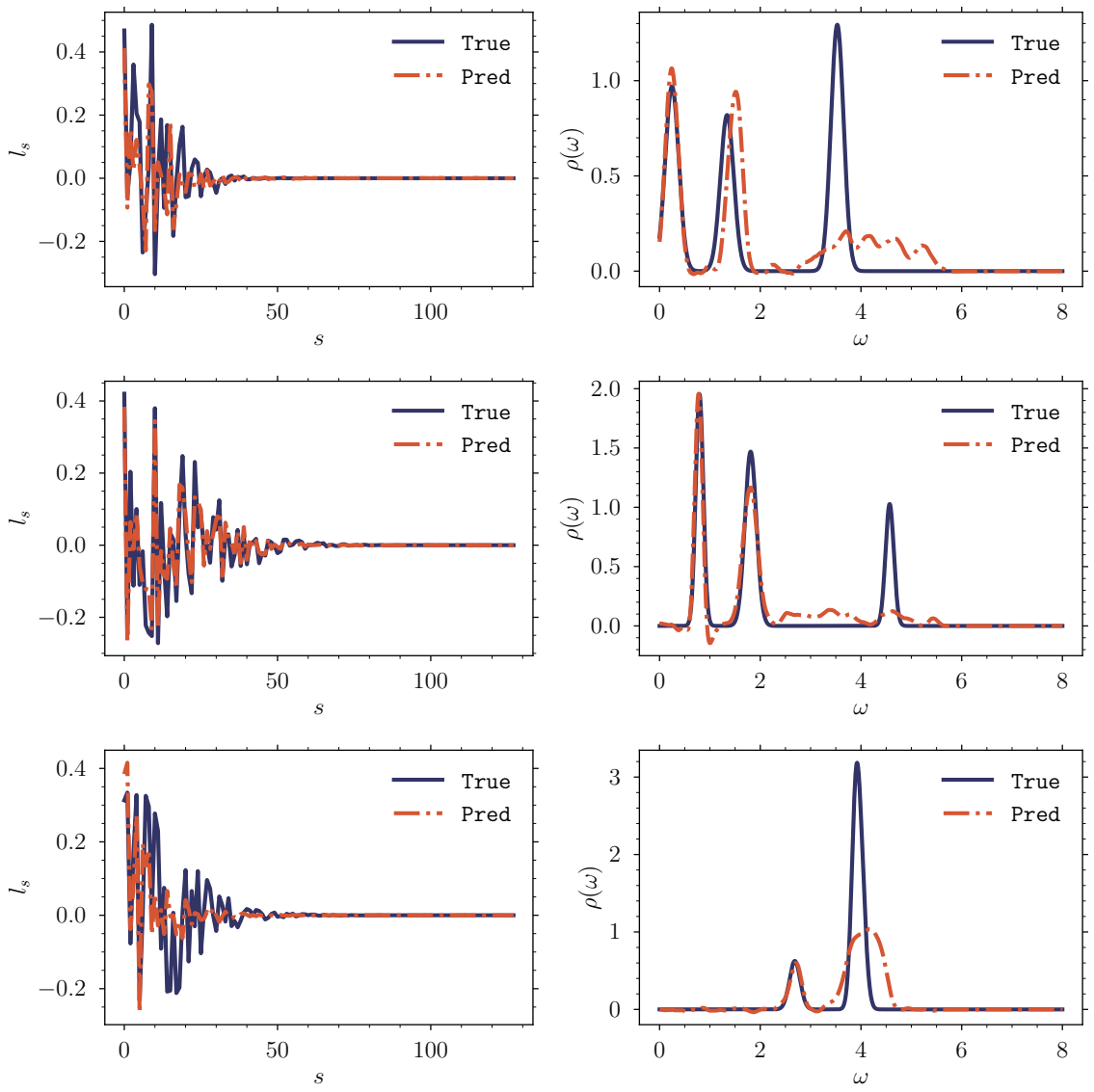


Figure C.3: Examples extracted from a dataset with $N_p = 3$

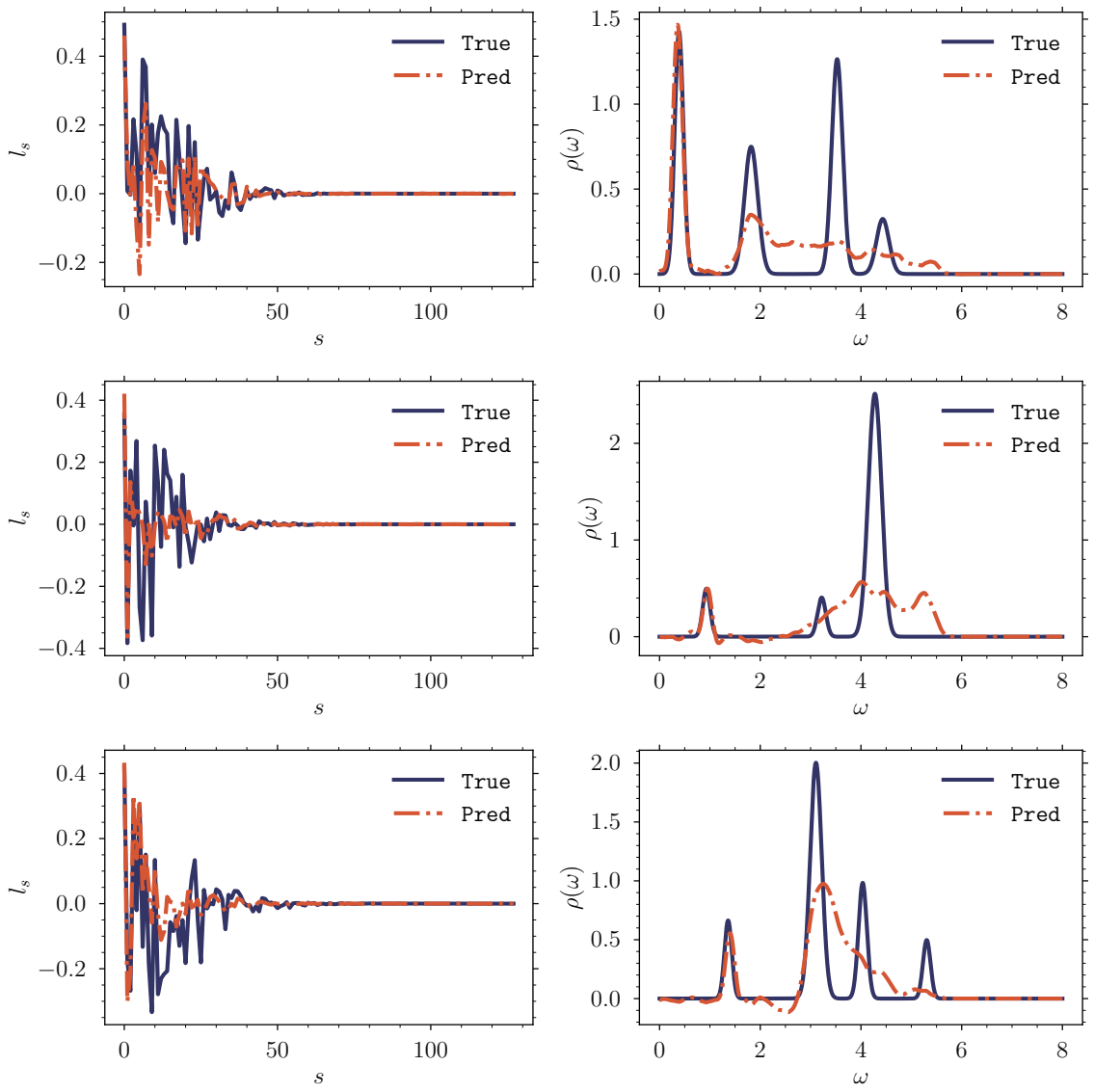


Figure C.4: Examples extracted from a dataset with $N_p = 4$

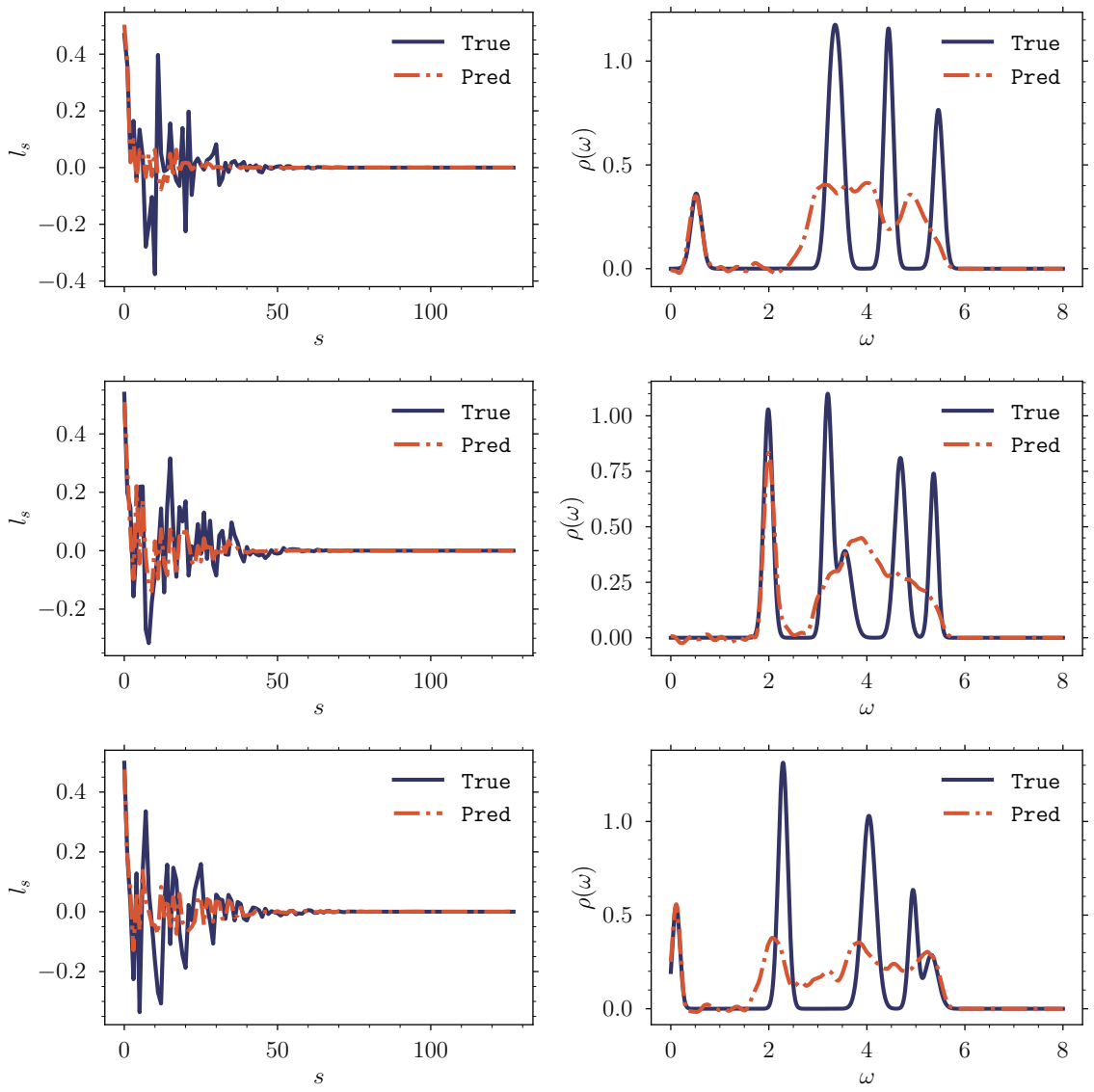


Figure C.5: Examples extracted from a dataset with $N_p = 5$

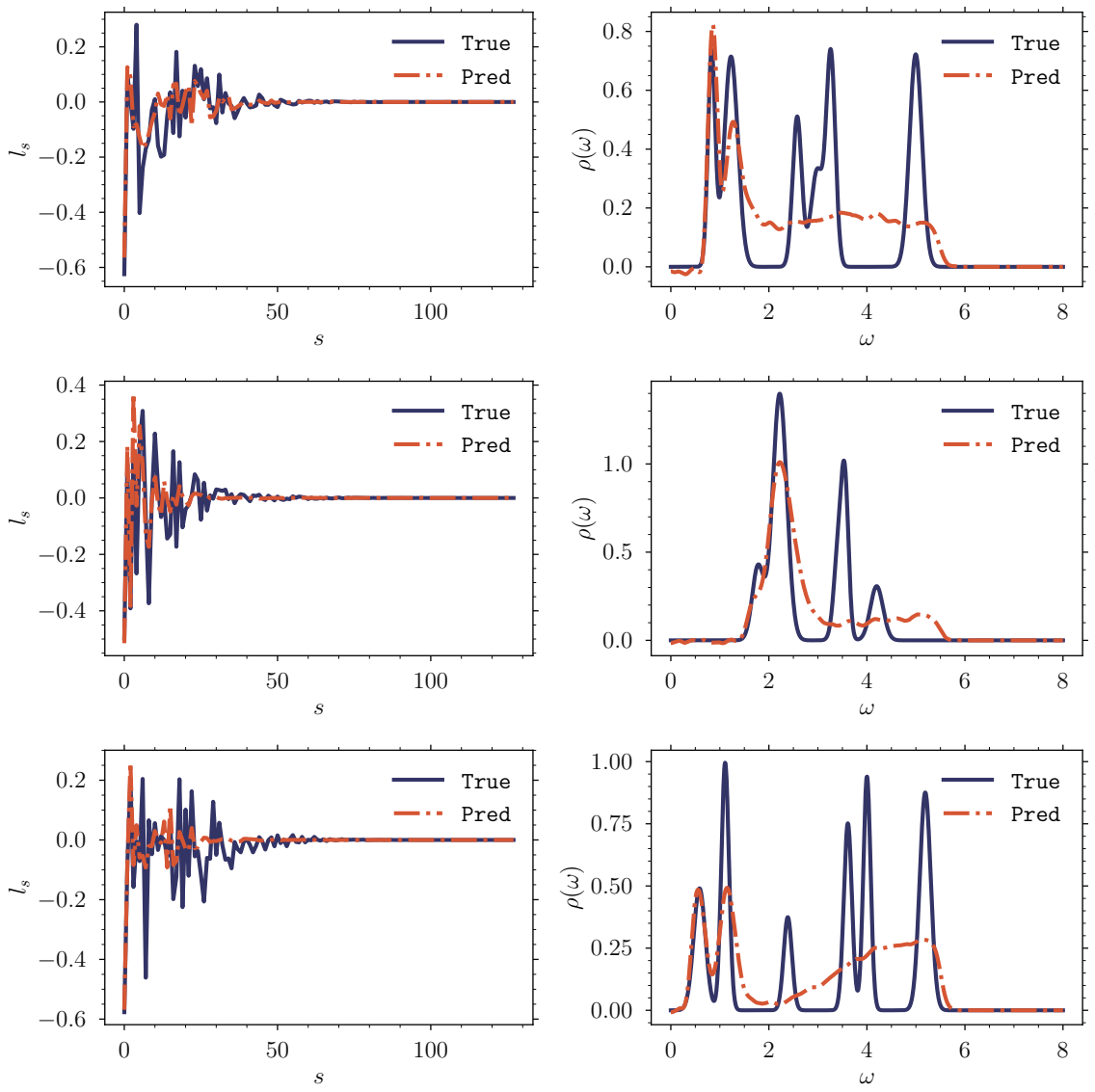


Figure C.6: Examples extracted from a dataset with $N_p = 6$

C.2 Fixed N_p, variable N_b

The reconstruction examples presented in this section are extracted from the experiment shown in Figure (6.7). All training datasets contain spectral functions with the same number of peaks, $N_p = 3$, but each of them has a different number of training examples N_b.

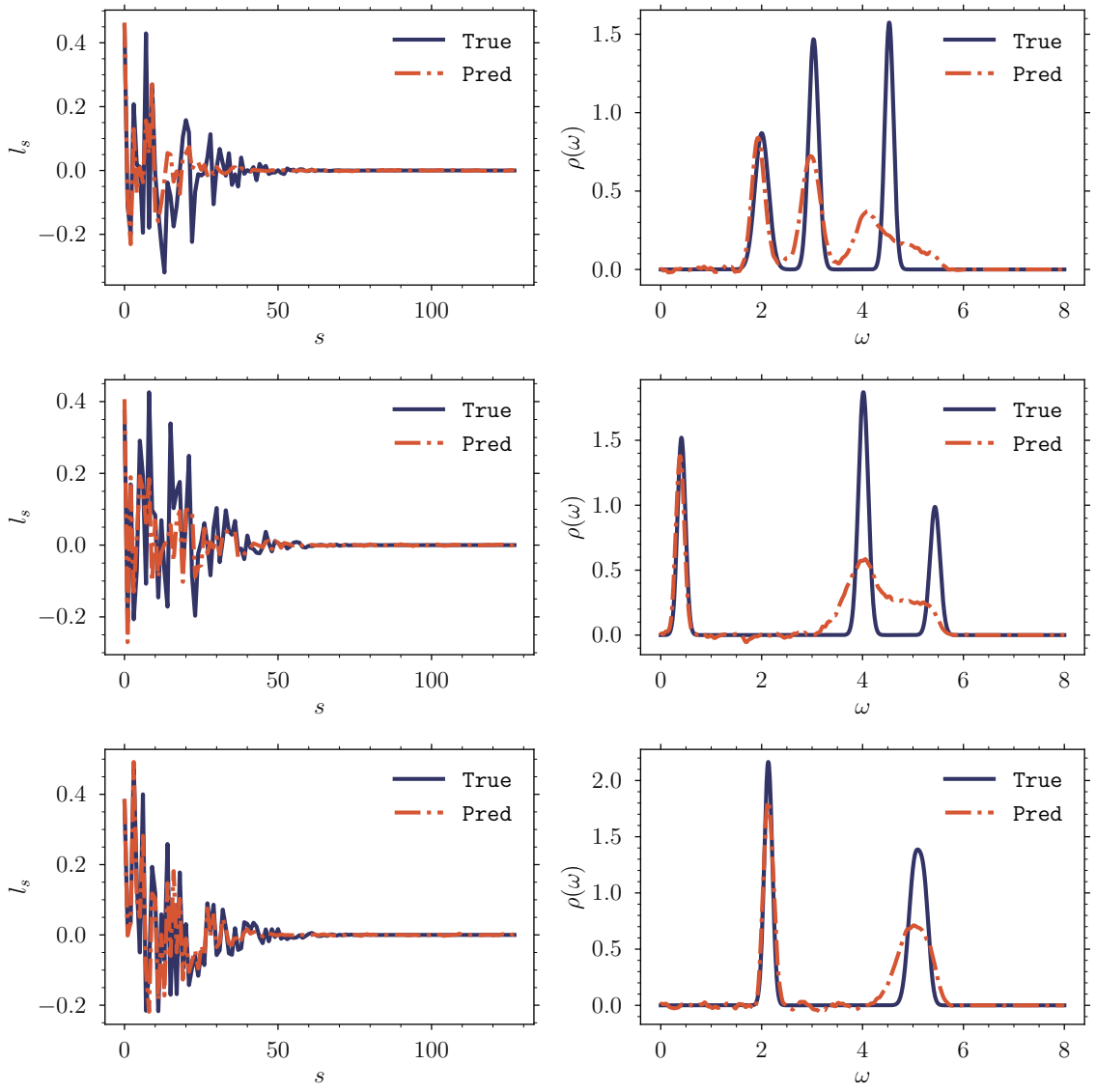


Figure C.7: Examples extracted from a dataset with $N_b = 250000$

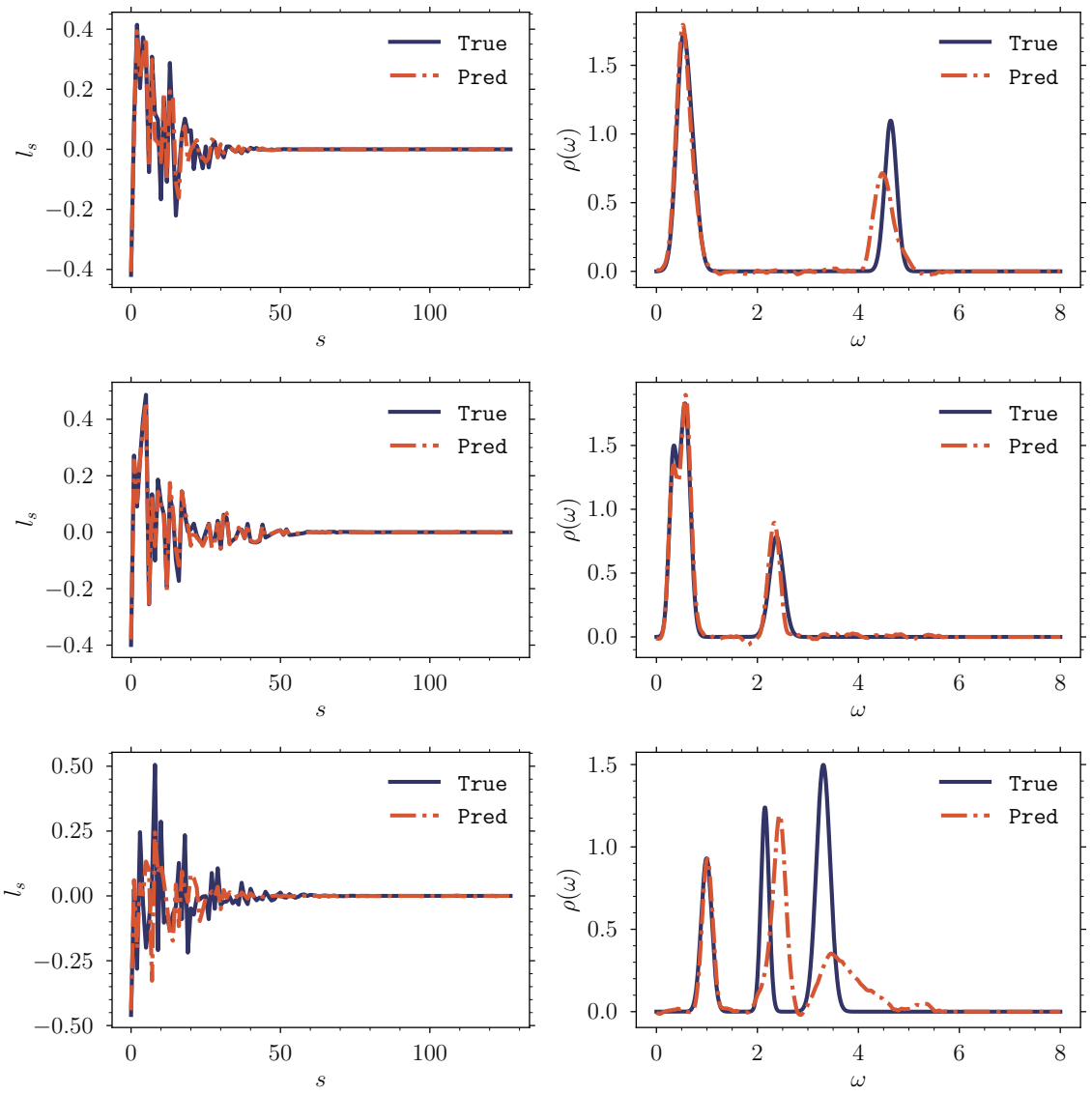


Figure C.8: Examples extracted from a dataset with $N_b = 500000$

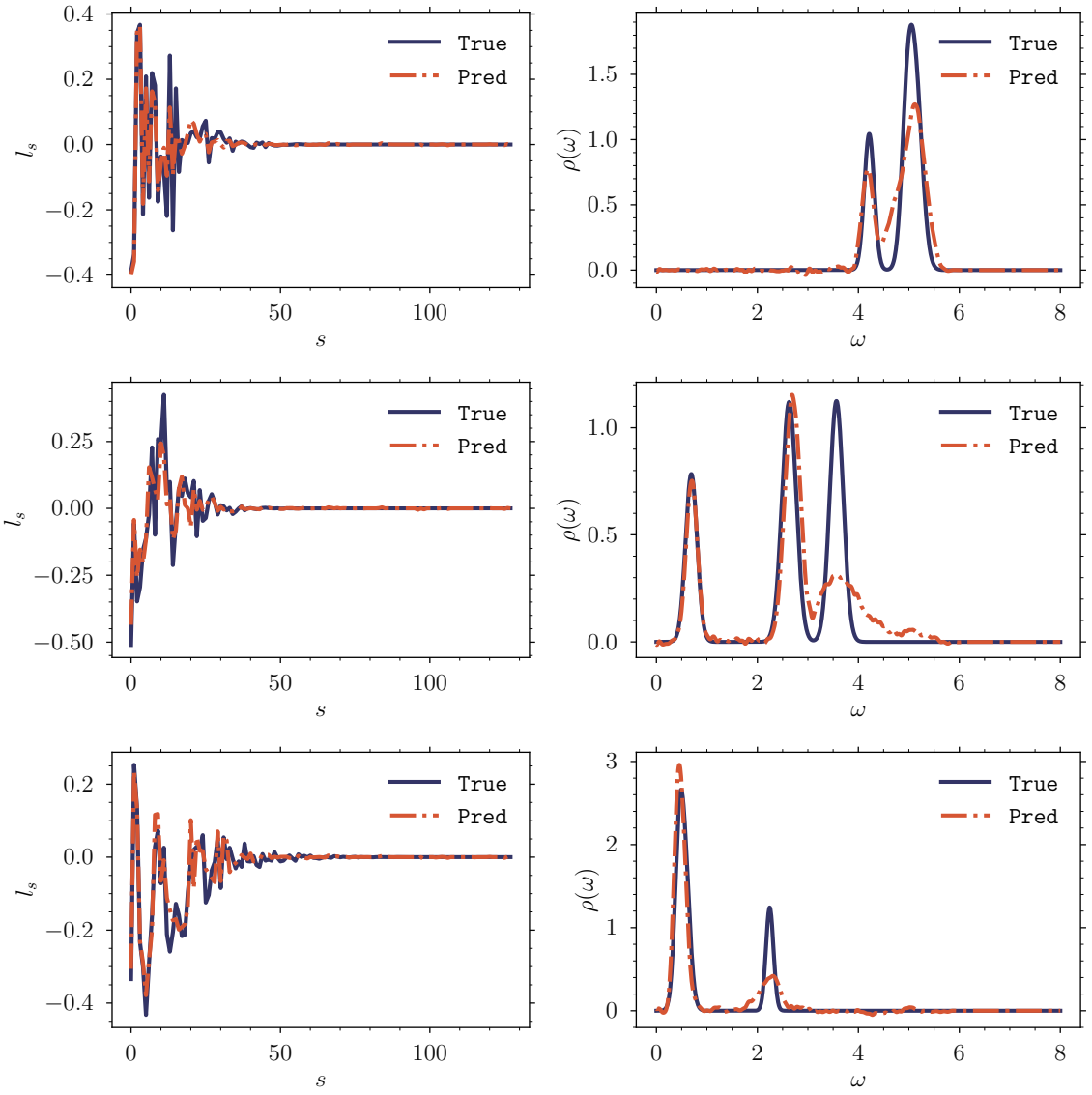


Figure C.9: Examples extracted from a dataset with $N_b = 750000$

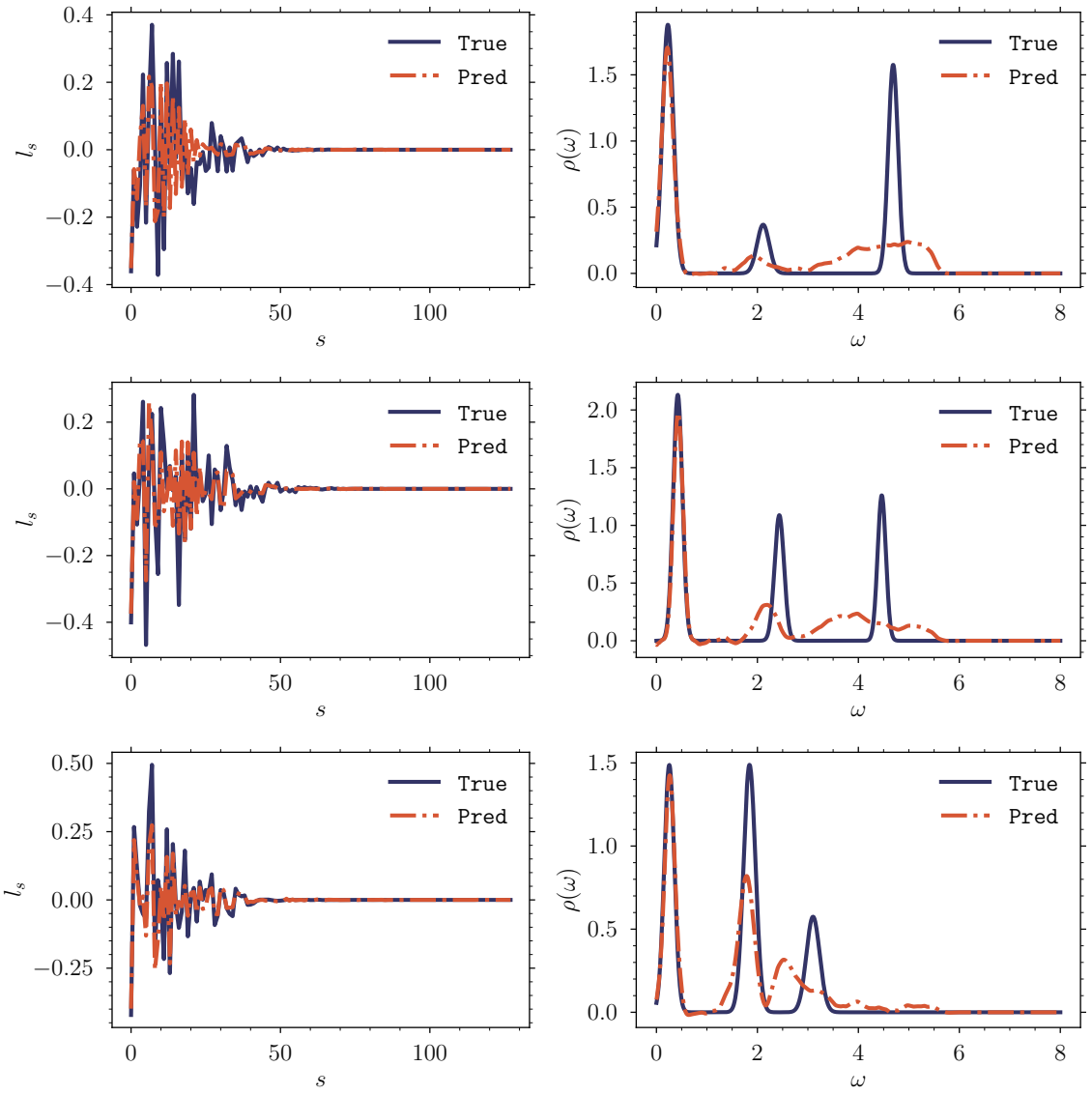


Figure C.10: Examples extracted from a dataset with $N_b = 1000000$

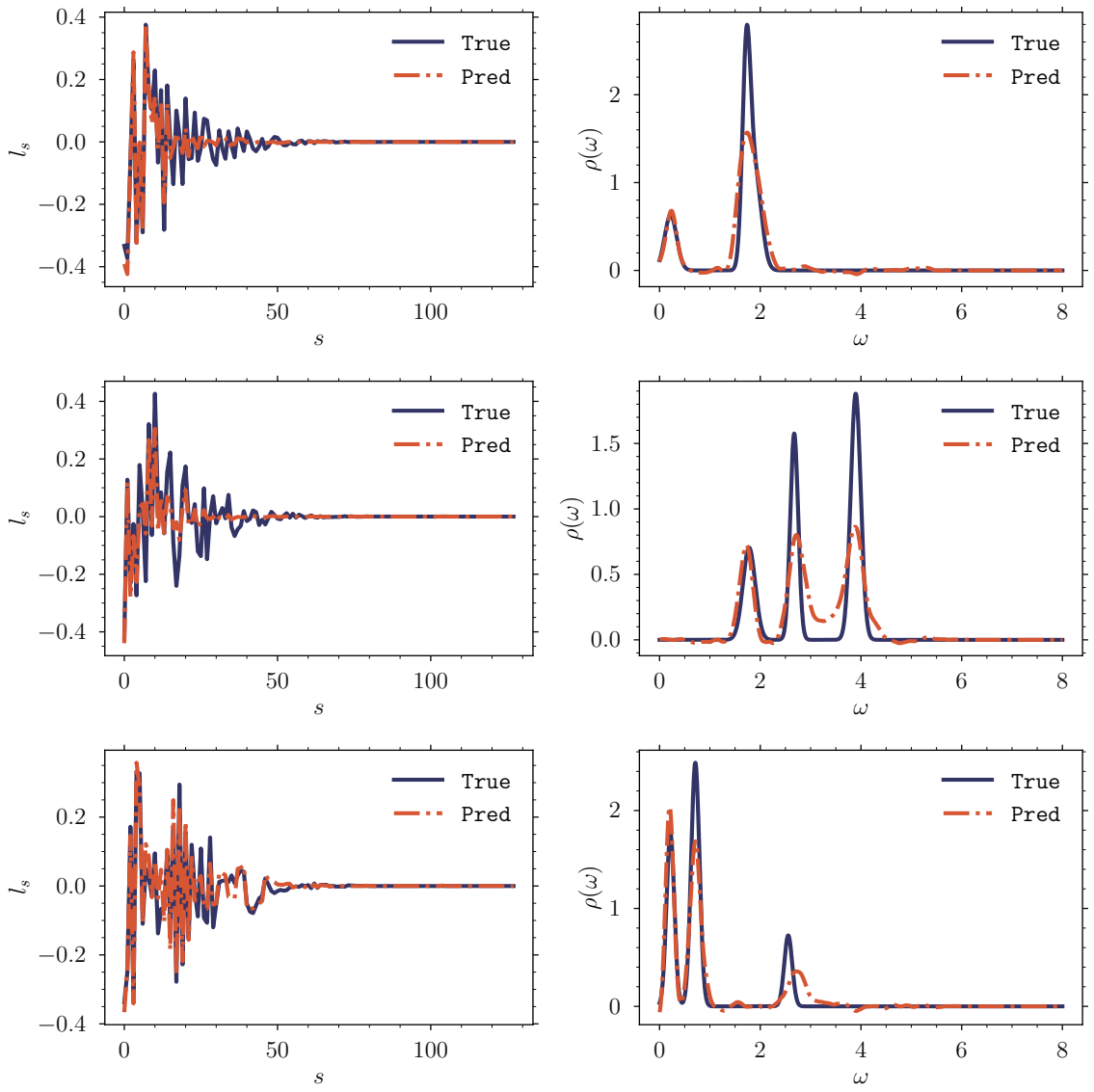


Figure C.11: Examples extracted from a dataset with $N_b = 1250000$

Appendix D

Study on statistically equivalent correlation functions

This appendix contains some analysis that support the idea that the required precision needed in the reconstruction mapping to resolve arbitrary high-energy peaks is extremely large, as the spectral reconstruction mapping becomes statistically non-injective as the number of peaks increases, that is, correlation functions that might be equivalent between statistical errors lead to completely different physics.

To perform the analysis, we first sample a reference spectral function generated through the combination of N_p bell-shaped peaks whose parameters are randomly sampled in the region of interest defined in eq. (6.14). After sampling a reference spectral function, we compute its associated correlation function by integrating eq. (6.1) with the low-temperature kernel defined in eq. (6.5); the resulting correlation function is labelled $C_r(\tau)$.

The second step of the analysis consists on looking for 25 statistically equivalent correlation functions. In this context, statistical equivalence means that a proposed correlation function $C(\tau)$ is contained inside the standard error band of $C_r(\tau)$:

$$\Big[C_r(\tau) - \mathrm{Err}[C_r(\tau)]\Big] \leq C_r(\tau) \leq \Big[C_r(\tau) + \mathrm{Err}[C_r(\tau)]\Big] \quad \forall\, \tau \tag{D.1}$$

The label $\mathrm{Err}[C_r(\tau)]$ denotes the standard error of the reference correlation function at a given Euclidean time. In principle, the errors are not heteroskedastic, that is, they depend on τ. For a given C to be statistically equivalent to C_r, the equation above must hold for all Euclidean times at which C and C_r are defined.

As our datasets are artificially generated, no sampling errors occur in the data-generation process, therefore, no notion of uncertainty is available in our sampled

correlation functions. As a result, we need to model the uncertainty in our estimates. To do so, we employ the following function to model the proportion of noise encountered in a given correlation function:

$$\Delta C(\tau) = \frac{N_0\, e^{N_1\, \tau}}{C(\tau)}, \tag{D.2}$$

where N_0 and N_1 are two constants defining the properties of the noise. Both constants can be fixed using the following equations:

$$N_0 = p_0\, C(\tau = 0); \quad N_f = \frac{1}{N_\tau} \log\big(\frac{p_{N_\tau}\, C(\tau = N_\tau)}{N_0}\big). \tag{D.3}$$

In the equation above p_0 denotes the proportion of noise in the correlation function at $\tau = 0$, and p_{N_τ} denotes the proportion of noise present in $C(\tau)$ at $\tau = N_\tau$.

From eq. (D.2), we can obtain the standard error of $C(\tau)$:

$$\mathrm{Err}[C(\tau)] = \Delta C(\tau)\, C(\tau). \tag{D.4}$$

Figure (D.1) shows two artificially generated correlation functions and their standard errors, computed using eq. (D.4) with $p_0 = 0.05$ and $p_{N_\tau} = 0.20$. In our experiments, we fix $p_0 = 0.05$ and $p_{N_\tau} = 0.10$.

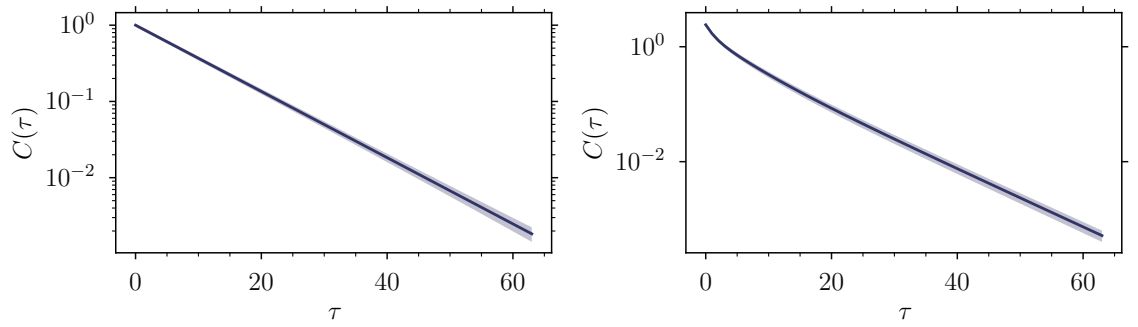


Figure D.1: Artificially generated correlation functions whose standard errors are computed using eq. (D.4). The noise model employed assumes a $p_0 = 0.05$ and a $p_{N_\tau} = 0.2$.

The results contained in the figures demonstrate that, as N_p increases, the landscape of possible spectral functions that lead to statistically similar correlation functions largely increases. As a result, the reconstruction mapping needs to be able

to resolve the minor details contained in the correlation function estimates to correctly determine the corresponding spectral function. It is worth mentioning that, in real state-of-the-art lattice QCD simulations, the proportion of noise encountered in correlation functions is around 10^{-4}, orders or magnitude below the ones shown in these experiments, which significantly reduces the chances of generating statistically equivalent correlation function from completely different spectral functions.

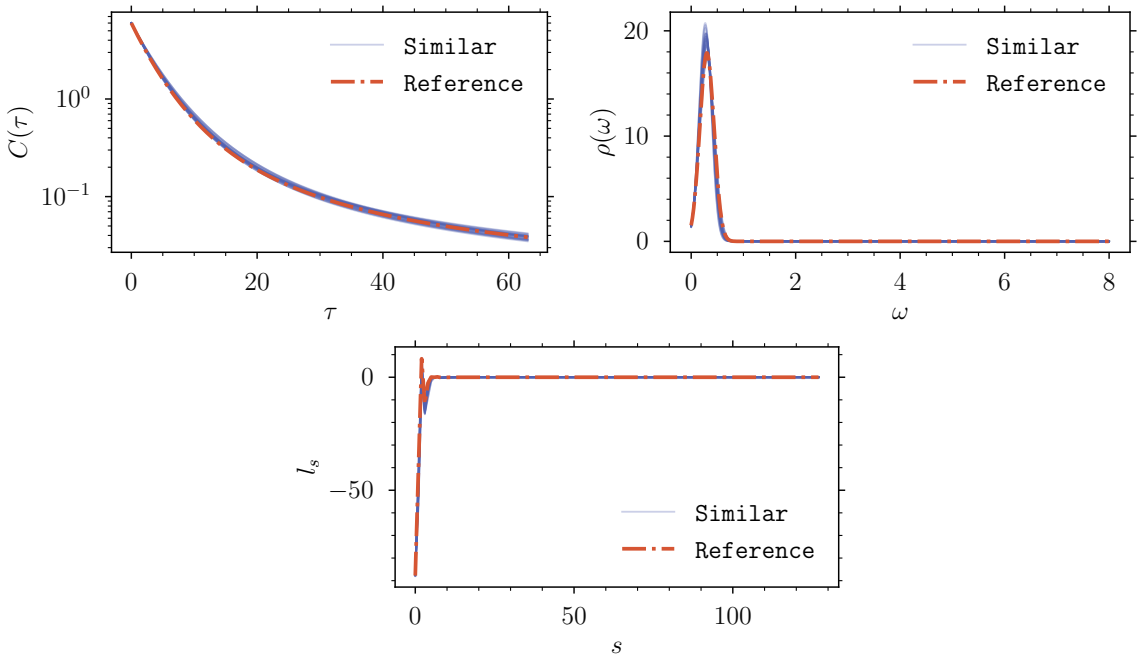


Figure D.2: Examples of statistically equivalent correlation functions, their associated spectral functions, and the respective first 128 SVD coefficients. All spectral functions are spanned by only one randomly sampled bell-shaped peak.

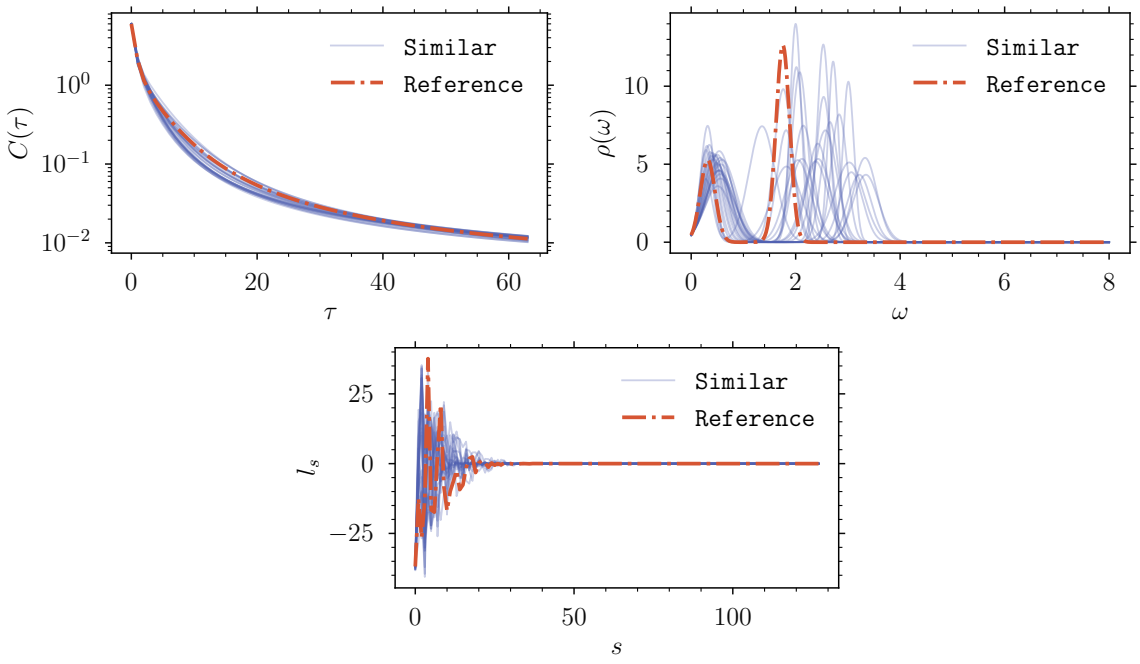


Figure D.3: Examples of statistically equivalent correlation functions, their associated spectral functions, and the respective first 128 SVD coefficients. All spectral functions are spanned by 2 randomly sampled bell-shaped peak.

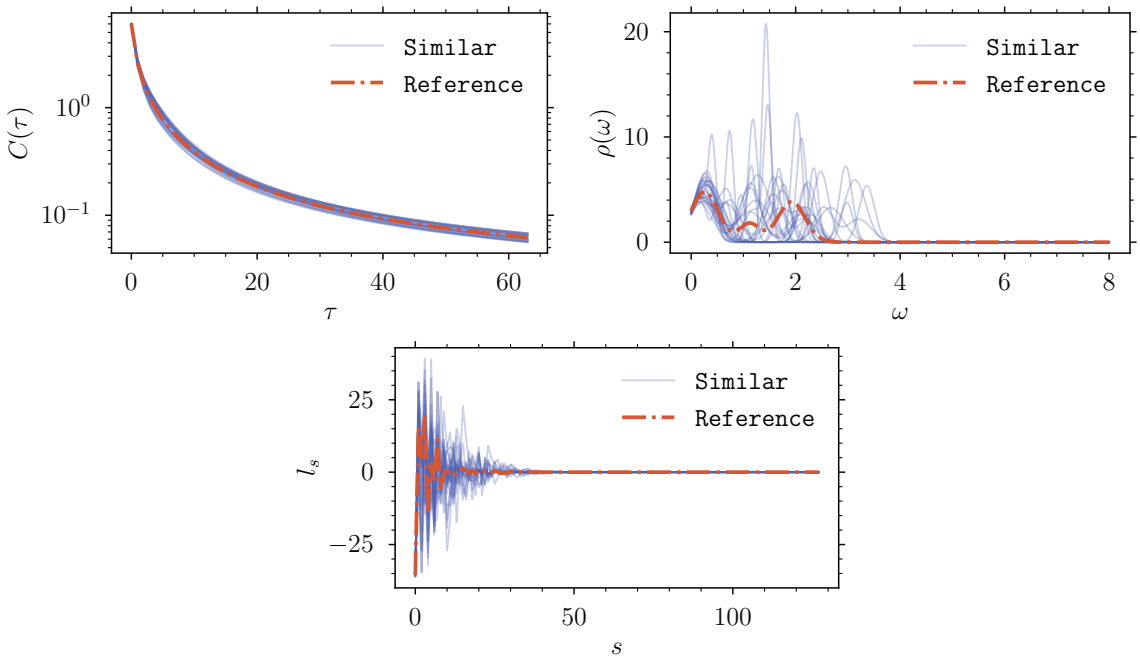


Figure D.4: Examples of statistically equivalent correlation functions, their associated spectral functions, and the respective first 128 SVD coefficients. All spectral functions are spanned by 3 randomly sampled bell-shaped peak.

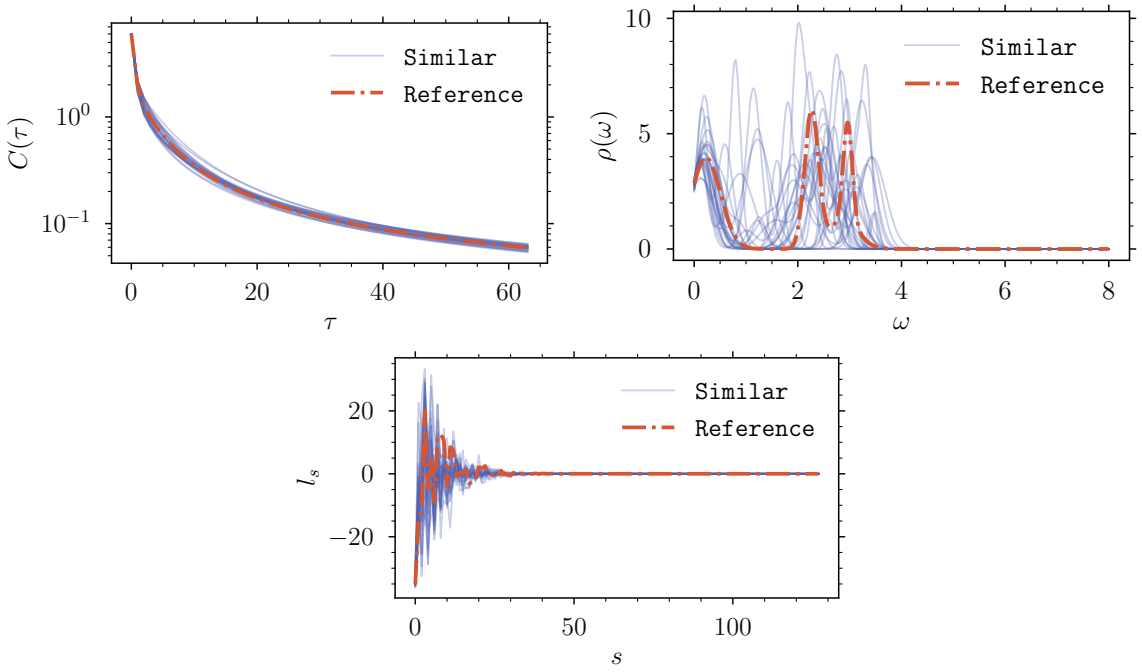


Figure D.5: Examples of statistically equivalent correlation functions, their associated spectral functions, and the respective first 128 SVD coefficients. All spectral functions are spanned by 4 randomly sampled bell-shaped peak.

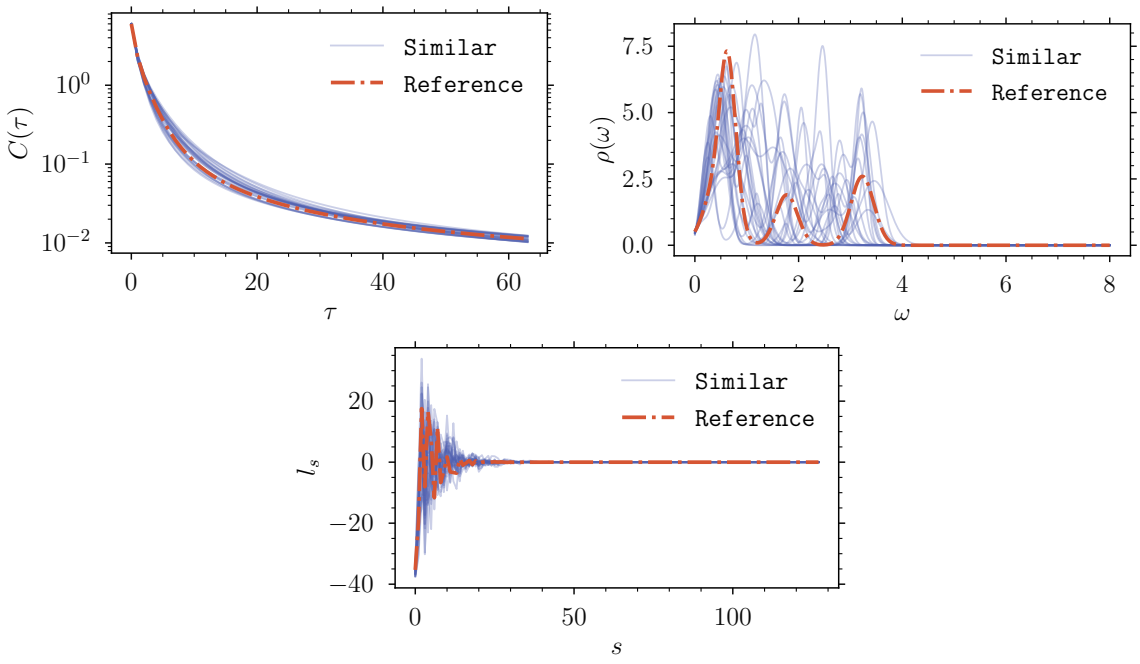


Figure D.6: Examples of statistically equivalent correlation functions, their associated spectral functions, and the respective first 128 SVD coefficients. All spectral functions are spanned by 5 randomly sampled bell-shaped peak.

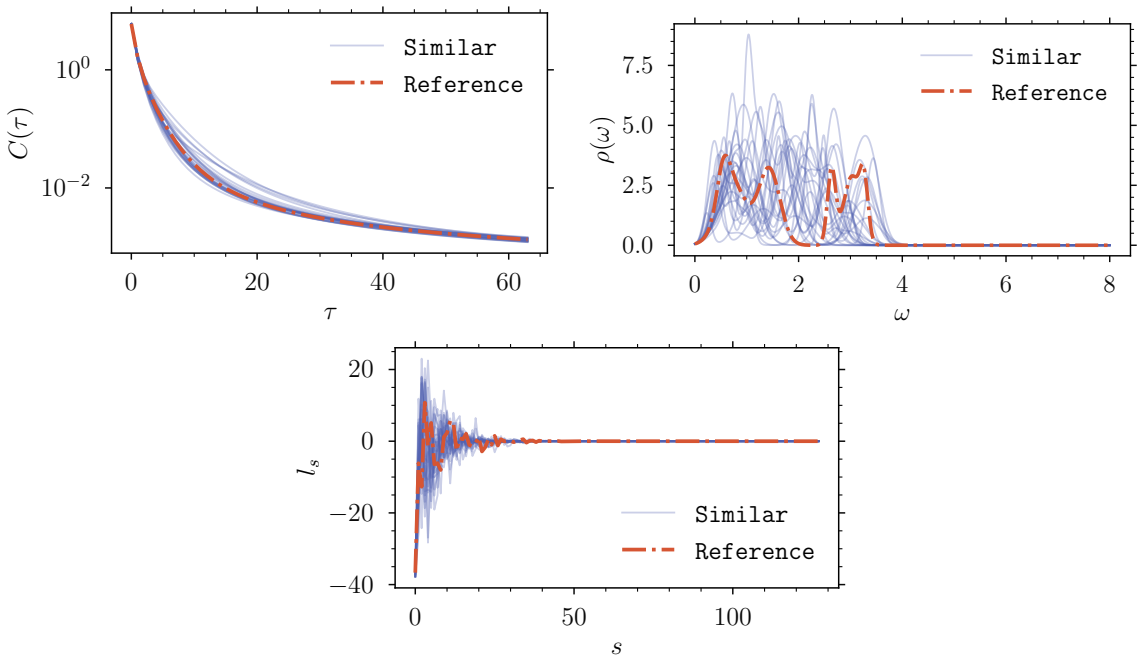


Figure D.7: Examples of statistically equivalent correlation functions, their associated spectral functions, and the respective first 128 SVD coefficients. All spectral functions are spanned by 6 randomly sampled bell-shaped peak.