

# Mass Spectroscopy

Patrick Oare

## 1 Overview

The idea of mass spectrometry is to use the lattice to calculate two point functions, and then to take these two point functions and extract physics. The central equation to this is the following:

$$C(n_t) := \langle \mathcal{O}(n_t) \bar{\mathcal{O}}(0) \rangle = \sum_k \langle 0 | \hat{\mathcal{O}} | k \rangle \langle k | \hat{\mathcal{O}}^\dagger | 0 \rangle \exp(-n_t E_k) \quad (1)$$

Here  $|k\rangle$  represents any intermediate state which can be connected to the operator  $\mathcal{O}$  with energy  $E_k$ , and  $n_t$  is the time slice we are calculating the correlation function on. The operators  $\hat{\mathcal{O}}$  and  $\hat{\mathcal{O}}^\dagger$  are related to the objects  $\mathcal{O}$  and  $\bar{\mathcal{O}}$  by the following identity:

$$\mathcal{O}(\alpha) \delta(\alpha - \alpha') := \langle \alpha' | \hat{\mathcal{O}} | \alpha \rangle \quad (2)$$

where  $|\alpha\rangle$  is a continuous eigenbasis of the space you are considering.

Using the relation between two point functions and matrix elements, we may extract physics about the energies of nucleon states. To do this, we consider evaluating two point functions when  $\mathcal{O}$  is an **interpolator** for the state we want to analyze. By an interpolator, we mean that  $\mathcal{O}$  creates or destroys the state of interest. For example, a proton interpolator  $\chi(n)$  will destroy a proton at site  $n$ , and its adjoint  $\bar{\chi}(n)$  will create a proton at site  $n$ .

So, suppose that we want to calculate the mass of a hadron  $h$  with corresponding interpolator  $\chi$ . Then using Equation 1, we have:

$$\langle \chi(n_t) \bar{\chi}(0) \rangle = \sum_k \langle 0 | \hat{\chi} | k \rangle \langle k | \hat{\chi}^\dagger | 0 \rangle \exp(-n_t E_k) \quad (3)$$

Because the operator  $\hat{\chi}^\dagger$  creates a hadron state  $\hat{\chi}^\dagger | 0 \rangle$ , the sum over intermediate states  $|k\rangle$  will only connect states with the quantum numbers of the hadron of interest to the ket  $\hat{\chi}^\dagger | 0 \rangle$ . So, the sum  $\sum_k$  will only index over intermediate states which contain at least one hadron of species  $h$ .

Consider taking the large  $n_t$  limit of this equation. When  $n_t$  gets large, intermediate states with a larger energy  $E_k$  become exponentially suppressed, and the dominant term in this series is the state  $|k\rangle$  with the lowest energy. Here is the key point: *the lowest energy state is exactly the ground state, and the energy will be given (at zero momentum) by  $E_0 = m_h$* . Thus, as we take  $n_t$  to be larger and larger, we can extract the hadron mass  $m_h$  by calculating the two point function of the interpolating operator:

$$\langle \chi(n_t) \chi^\dagger(0) \rangle = A \exp(-n_t E_0) (1 + O(e^{-n_t \Delta E})) \quad (4)$$

where now  $A$  is a constant we can determine with a data fit.

## 1.1 Momentum Projection

The state of the hadron we have created with the interpolator has not been completely specified; although we know it lies on a fixed timeslice, we do not know the momentum of the hadron. If we compute the correlation function  $\langle \chi(\mathbf{n}, n_t) \chi^\dagger(0) \rangle$ , we will know the location of the particle we have destroyed, but we will not know its momentum. We instead want to make sure that we project onto a state of definite momentum, which we will do with a Fourier transform:

$$\langle \chi(\mathbf{p}, n_t) \chi^\dagger(0) \rangle = \frac{1}{\sqrt{|\Lambda_3|}} \sum_{\mathbf{n} \in \Lambda_3} e^{-i\mathbf{a}\mathbf{p} \cdot \mathbf{n}} \langle \chi(\mathbf{n}, n_t) \chi^\dagger(0) \rangle \xrightarrow{n_t \rightarrow \infty} A \exp(-n_t E_0(\mathbf{p})) + \dots \quad (5)$$

where  $\Lambda_3 = \{(\mathbf{n}, n_t) \in \Lambda : \mathbf{n} \in \mathbb{Z}^3\}$  is the set of lattice points on our fixed time slice ( $\Lambda$  is the full lattice). The important point here is that we know the dispersion  $E(\mathbf{p})$ , which is simply:

$$E(\mathbf{p}) = \sqrt{\mathbf{p}^2 + m_h^2} \quad (6)$$

In practice, we will often project onto  $\mathbf{0}$  momentum, which is easily achievable through summing on spatial points in the lattice:

$$\langle \chi(\mathbf{p} = \mathbf{0}, n_t) \chi^\dagger(0) \rangle = \frac{1}{\sqrt{|\Lambda_3|}} \sum_{\mathbf{n} \in \Lambda_3} \langle \chi(\mathbf{n}, n_t) \chi^\dagger(0) \rangle \xrightarrow{n_t \rightarrow \infty} A \exp(-n_t m_h) + \dots \quad (7)$$

## 1.2 Effective Mass

In theory we would like to take the limit as  $n_t \rightarrow \infty$  to remove contamination from the higher energy states, but on a finite lattice this is not possible. So, we need to define a way to know when  $n_t$  is large enough that the energy we extract from this method is sufficiently close to the mass of the hadron. A simple way to do this is to define a mass scale by noticing that as  $n_t \rightarrow \infty$ ,  $C(n_t) \rightarrow A e^{-n_t m_h}$ . In this limit, we can extract the mass by taking a ratio of correlation functions:

$$\frac{C(n_t)}{C(n_t + 1)} \xrightarrow{n_t \rightarrow \infty} e^{m_h} \quad (8)$$

This motivates the definition of the **effective mass**:

$$m_{eff}(n_t) := \log \left( \frac{C(n_t)}{C(n_t + 1)} \right) \quad (9)$$

As  $n_t$  becomes sufficiently large,  $m_{eff}(n_t)$  approaches a constant and plateaus at the value  $m_{eff} = m_h$ . At smaller time slices,  $m_{eff}(n_t)$  will not appear to be constant but instead will also contain contamination from higher energy modes. Thus in essence, it suffices to compute the effective mass and to fit a constant line to the region where it plateaus to determine the actual mass of the hadron.

Because we often use periodic boundary conditions on the lattice, it is often useful to take this into account and define a variation of the effective mass, which is the **cosh-corrected effective mass**. Let  $N_T$  be the temporal extent of the lattice. Then:

$$\frac{C(n_t)}{C(n_t + 1)} = \frac{\cosh(m_{eff}(n_t - N_T/2))}{\cosh(m_{eff}(n_t - N_T/2 + 1))} \quad (10)$$

### 1.3 Bootstrapping

Assume now that we have calculated values of the interpolation function on  $n$  gauge field configurations  $\{U_i\}_{i=1}^n$ , and that our lattice has size  $L^3 \times T$ . Let the values of the interpolators be contained in the set:

$$\{C(i, t)\}_{i=1, t=1}^{n, T} \quad (11)$$

where  $C(i, t)$  is the value of the interpolator on the  $i$ th gauge field configuration, momentum projected to 0 on the  $t$ th time slice. Generally these values  $C(i, t)$  will be stored in a matrix  $C_{it}$  to make them easier to work with. If we simply want to ensemble average of these interpolators, at this point we could average them over the configuration index to find:

$$\langle C(t) \rangle = \frac{1}{n} \sum_{i=1}^n C(i, t) \quad (12)$$

This would give us an average for our correlation function computed at each time slice, which is what we originally sought after.

However, this is not the best way to establish statistics on this problem. Instead, we consider a method called **bootstrapping**, which will allow us to produce a larger amount of ensembles from our initial data  $C(i, t)$ . With this method, we consider generating  $N_{boot}$  new ensembles from our original one. To do this, fix  $t$  and pick  $n$  entries from  $C(i, t)$  randomly *with replacement*. Denote these  $n$  samples by  $A_1(i, t)$ . Repeat this sampling procedure  $N_{boot}$  times until you have a collection  $A_1(i, t), \dots, A_{N_{boot}}(i, t)$  of ensembles sampled from your original data. We may use the notation  $A(b, i, t)$  for  $A_b(i, t)$ , where  $b = 1, \dots, N_{boot}$ ,  $j = 1, \dots, n$ , and  $t = 1, \dots, T$  to refer to these ensembles and to emphasize its structure as a 3-dimensional matrix.

The ensemble  $\mathbb{A} := \{A_1, \dots, A_{N_{boot}}\}$  will be used for our data analysis. Suppose that we are trying to compute the effective mass  $m_{eff}$  from this set of data. For each dataset  $A \in \mathbb{A}$  (recall this is a matrix of size  $n \times T$ ), we can compute the values of the effective mass, denoted  $m_A$ , by the following:

$$m_A(i, t) = \log \left( \frac{A(i, t)}{A(i, t+1)} \right) \quad (13)$$

At this point, we may average  $m_A(i, t)$  over each gauge field configuration in  $A$  to get an estimate on its mean:

$$\hat{m}_A(t) = \frac{1}{n} \sum_{i=1}^n m_A(i, t) \quad (14)$$

For each time slice  $t$ , we use the set  $\{\hat{m}_A(t) : A \in \mathbb{A}\}$  to perform an estimate of the effective mass and its error (note  $N_{boot} = |\mathbb{A}|$ ):

$$\mu_m(t) = \frac{1}{|\mathbb{A}|} \sum_{A \in \mathbb{A}} \hat{m}_A(t) \quad (15)$$

$$\sigma_m(t)^2 = \frac{1}{|\mathbb{A}| - 1} \sum (\hat{m}_A(t) - \mu_m(t))^2 \quad (16)$$

Using the bootstrap method, we may calculate statistics on functions other than just the effective mass. Particular quantities of importance to calculate other than the effective mass are the value of the correlation function and the cosh-corrected effective mass. To calculate other quantities (say a quantity  $X$  which depends on  $C(i, t)$ ), simply compute this quantity  $X_A(i, t)$  for each ensemble  $A \in \mathbb{A}$  as we did for  $m_A(i, t)$  and follow the rest of the calculation precisely.

## 2 Example: Pion Mass

We will be computing the mass of the pion  $\pi^+$  in this example.

### 2.1 Pion interpolator

The  $\pi^+$  interpolator (which can be checked by observing its transformation properties under symmetries like parity and charge conjugation) is:

$$\chi_\pi(n) = \bar{d}(n)\gamma^5 u(n) \quad (17)$$

To perform computations with this on the lattice, we can work out an expression for the fermionic part of the two-point function for this interpolator:

$$\begin{aligned} \langle \chi_\pi(n) \bar{\chi}_\pi(m) \rangle_F &= \langle (\bar{d}(n)_\alpha \gamma_{\alpha\beta}^5 u(n)_\beta) (\bar{u}(m)_\rho \gamma_{\rho\sigma}^5 d(m)_\sigma) \rangle_F \\ &= -\gamma_{\alpha\beta}^5 \gamma_{\rho\sigma}^5 \langle u(n)_\beta \bar{u}(m)_\rho \rangle_F \langle d(m)_\sigma \bar{d}(n)_\alpha \rangle_F \\ &= -\gamma_{\alpha\beta}^5 \gamma_{\rho\sigma}^5 D_u^{-1}(n|m)_{\beta\rho} D_d^{-1}(m|n)_{\sigma\alpha} \\ &= -tr \{ \gamma^5 D_u^{-1}(n|m) \gamma^5 D_d^{-1}(m|n) \} \end{aligned}$$

where we picked up a  $-$  sign because the fields are Grassmannian. Note we use Wick's theorem to turn the products  $\langle u\bar{u} \rangle$  and  $\langle d\bar{d} \rangle$  of fermion fields into propagators  $D_u^{-1}$  and  $D_d^{-1}$ . However, this current expression requires computing a fermion propagator at the source  $m$  and tying it up at  $n$ , and doing the same with creating a source at  $n$  and tying it up at  $n$ . To simplify this further, we exploit  $\gamma^5$ -hermicity of the propagator:

$$D^{-1}(m|n) = \gamma^5 D^{-1}(n|m)^* \gamma^5 \quad (18)$$

Hence we have the final form for this equation which we will use:

$$\langle \chi_\pi(n) \chi_\pi(m) \rangle_F = -tr \{ D_u^{-1}(n|m) D_d^{-1}(n|m)^* \} \quad (19)$$

In particular, to compute the  $\pi^+$  mass we will put a source at the origin and project it onto  $\mathbf{0}$  momentum on time slice  $n_t$ . This is (where  $|\Lambda_3|$  is the size of the spatial lattice):

$$C(n_t) := \langle \chi_\pi(\mathbf{p} = \mathbf{0}, n_t) \bar{\chi}_\pi(0) \rangle_F = \frac{1}{\sqrt{|\Lambda_3|}} \sum_{\mathbf{n} \in \Lambda_3} \langle \chi_\pi(\mathbf{n}, n_t) \bar{\chi}_\pi(0) \rangle_F \quad (20)$$

or explicitly:

$$C(n_t) = -\frac{1}{\sqrt{|\Lambda_3|}} \sum_{\mathbf{n} \in \Lambda_3} tr \{ D_u^{-1}(\mathbf{n}, n_t | 0) D_d^{-1}(\mathbf{n}, n_t | 0)^* \} \quad (21)$$

## 2.2 Code

The code I wrote to compute the pion mass does the following steps. Suppose that a set of gauge field configurations  $\mathcal{U} = \{U_i\}_{i=1}^N$  is given. In my case, I used the configurations in `/data/d10b/ensembles/isoClover/cl3_16_48_b6p1_m0p2450`, where there were 254 configurations (although one configuration was not loading in, so I used  $N = 253$  configurations). I begin by initializing the following parameters:

- A lattice  $L$  of size  $16^3 \times 48$ .
- The parameter  $\beta = 6.1$ .
- Quark masses  $m = m_u = m_d = .2450$ .
- The hopping parameter  $\kappa = \frac{1}{2m+8}$ . I have checked and am not positive how to specify the mass parameter in the clover inverter instead of the  $\kappa$  value, so I decided to just compute this from the masses and put it directly into the inverter.
- The volume of the lattice.

Once these parameters are specified, I loop over each gauge field configuration  $U \in \mathcal{U}$  and do the following:

1. Create clover objects  $CL_u$  and  $CL_d$  for the up and down quarks. I initialize these objects with:
  - Periodic boundary conditions in space and antiperiodic boundary conditions in time
  - Hopping parameters  $\kappa_u = \kappa_d = \kappa$  which were specified above.
  - Sheikholeslami-Wohlert coefficients of  $c_{sw} = 1$ .