Lattice field theories
Exercise sheet 4 – Fermion Correlators and Mass Extraction

Lectures: Jan Pawlowski j.pawlowski@thphys.uni-heidelberg.de
Felipe Attanasio pyfelipe@thphys.uni-heidelberg.de

Tutorials: Felipe Attanasio pyfelipe@thphys.uni-heidelberg.de

Institut für Theoretische Physik, Uni Heidelberg

Please see the supplemental material at the end of this exercise sheet for explanations and hints.

Exercise 8: Simplifying the propagator

Consider the Euclidean correlator of a *free* fermion created at time t' and annihilated at time $t' + \delta$:

$$\langle \psi_{t'+\delta}\, \overline{\psi}_{t'} \rangle = \sum_n \langle 0|\psi|n\rangle\langle n|\overline{\psi}|0\rangle \left[e^{-E_n\delta} - e^{-E_n(N_t-\delta)}\right] , \tag{1}$$

Use the fact that we are working in the free case to show that only the $n = 1$ element of eq. (1) survives, and that the result is proportional to $\sinh(E_n(N_t/2 - \delta))$.

Exercise 9: Inverting matrices and extracting masses

The correlator of eq. (1) can also be obtained in the path-integral formulation via

$$\langle \psi_y \overline{\psi}_x \rangle = \frac{\delta}{\delta \overline{J}_y} \frac{\delta}{\delta J_x} \ln Z \bigg|_{J=\overline{J}=0} = (D^{-1})_{y,x} , \tag{2}$$

where D represents the Dirac operator. In d spacetime dimensions and in the Wilson formulation, it is given by

$$D_{x,y} = (d\,r + m)\mathbb{1}\delta_{x,y} - \frac{1}{2}\sum_{\mu=0}^{d-1} \left[(r - \gamma_\mu)\delta_{x,y+\hat{\mu}} + (r + \gamma_\mu)\delta_{x,y-\hat{\mu}}\right] , \tag{3}$$

where $x = (x_0, \vec{x})$ and $y = (y_0, \vec{y})$, $\hat{\mu}$ represents a unit vector in the μ direction, r is the Wilson parameter, and $\mathbb{1}$, γ_μ act in Dirac space.

Consider now the correlator of a zero momentum fermion created at the origin and annihilated at time $t = \delta$:

$$C(\delta) = \frac{1}{V}\sum_{\vec{y}} \langle \psi_{(\delta,\vec{y})}\, \overline{\psi}_0 \rangle = \frac{1}{V}\sum_{\vec{y}} (D^{-1})_{(\delta,\vec{y}),0} = \frac{1}{V}\sum_{\vec{y}}\sum_{x} (D^{-1})_{(\delta,\vec{y}),x} v_x , \tag{4}$$

where $v_x \equiv v_{x_0,\vec{x}} = s\,\delta_{x_0,0}\,\delta_{\vec{x},\vec{0}}$ and s is a spinor.

Implement the Conjugate Gradient (CG) algorithm and use it to compute eq. (4) using the Dirac operator in $1+1$ dimensions. Use $N_t = 64$ and $V = N_x = 32$, $r = 1$, and $0.01 \leq m \leq 1.0$. In this exercise we do not care about a particular spin orientation, so just like in QFT we compute the correlator for both spin up, $s = [1 \quad 0]^\intercal$, and spin down, $s = [0 \quad 1]^\intercal$, and sum them.

1. Measure the number of iterations used by CG as a function of the bare fermion mass. You should see that it increases for lower masses. Can you think of an explanation?

2. You should observe that $C(\delta)$, for $\delta > 0$, is proportional to $\sinh(am_R(N_t/2 - \delta))$, where m_R is the renormalised fermion mass. It is related to the bare mass via

$$am_R = \ln\left[1 + Z\,(am - am_c)\right]\,, \tag{5}$$

   where Z is a renormalisation constant and m_c is a shift to the mass caused by the Wilson term. In the free case, $Z = 1$ and $m_c = 0$. This provides a non-trivial check of your implementation: you should be able to extract the bare mass by fitting $C(\delta)$ with the expected sinh behaviour and verify if it agrees with the input value.

Supplemental material

For this exercise sheet we work with *free* Wilson fermions. Consider the creation and annihilation operators for fermions at *zero spatial momentum* at time t

$$\psi_t \equiv \frac{1}{V}\sum_{\vec{x}} \psi_{t,\vec{x}}\,, \tag{6}$$

$$\overline{\psi}_t \equiv \frac{1}{V}\sum_{\vec{x}} \overline{\psi}_{t,\vec{x}}\,, \tag{7}$$

where V stands for the spatial volume. With those operators we can look at the (Euclidean) correlator of a zero momentum fermion created at time t and annihilated at time t', $\langle\psi_{t'}\overline{\psi}_t\rangle$.

Note that fermionic fields are anti-periodic in the temporal direction, i.e., $\psi_{t+N_t,\vec{x}} = -\psi_{t,\vec{x}}$, where N_t is the temporal extent of the lattice. Consider, then, a particle source at $t = t'$ and a sink at $t = t' + \delta$. The sink, because of the anti-periodic boundary conditions, "sees" another source[1] at $t = t' + N_t$. Alternatively, the source at $t = t'$ "sees" two sinks: one at $t = t' + \delta$ and another at $t = N_t + t' - \delta$. Of course, the minus sign from the *anti*-periodic BC have to be taken into account.

This correlator contains two terms: one involving the particle "propagating" forwards from t' to $t' + \delta$, and another with the particle "propagating" backwards from t' to 0, which is identified with N_t, thus picking up a minus sign, and then to $t' + \delta$. It can be written, using energy eigenstates $|n\rangle$, as

$$\begin{aligned}
\langle\psi_{t'+\delta}\,\overline{\psi}_{t'}\rangle &= \langle 0|\psi_{t'+\delta}\,\overline{\psi}_{t'}|0\rangle - \langle 0|\psi_{N_t+t'-\delta}\,\overline{\psi}_{t'}|0\rangle && (8)\\
&= \sum_n \left[\langle 0|\psi_{t'+\delta}|n\rangle\langle n|\overline{\psi}_{t'}|0\rangle - \langle 0|\psi_{N_t+t'-\delta}|n\rangle\langle n|\overline{\psi}_{t'}|0\rangle\right] && (9)\\
&= \sum_n \Big[\langle 0|\psi\, e^{-\hat{H}(t'+\delta)}|n\rangle\langle n|e^{\hat{H}t'}\,\overline{\psi}|0\rangle - \langle 0|\psi\, e^{-\hat{H}(N_t+t'-\delta)}|n\rangle\langle n|e^{\hat{H}t'}\,\overline{\psi}|0\rangle\Big] && (10)\\
&= \sum_n \langle 0|\psi|n\rangle\langle n|\overline{\psi}|0\rangle\left[e^{-E_n\delta} - e^{-E_n(N_t-\delta)}\right]\,, && (11)
\end{aligned}$$

where we have used the facts that $\hat{H}|n\rangle = E_n|n\rangle$ and that the vacuum is invariant under time evolution. Since the propagator only depends on the time difference δ we shall, for simplicity, set $t' = 0$.

The propagator $\langle\psi_y\overline{\psi}_x\rangle$, where $x = (x_0, \vec{x})$ and $y = (y_0, \vec{y})$, can also be computed via the path-integral formulation by taking appropriate derivatives of the partition function,

$$\langle\psi_y\overline{\psi}_x\rangle = \frac{\delta}{\delta\overline{J}_y}\frac{\delta}{\delta J_x}\ln Z\bigg|_{J=\overline{J}=0} = (D^{-1})_{y,x}\,, \tag{12}$$

[1] Note that this is unrelated to the fermion doubling problem, as this is only due to boundary conditions.

where D represents the Dirac operator.

The Dirac operator is a sparse matrix[2]. In d spacetime dimensions and in the Wilson formulation, it is given by

$$D_{x,y} = (d\,r + m)\mathbb{1}\delta_{x,y} - \frac{1}{2}\sum_{\mu=0}^{d-1}\left[(r-\gamma_\mu)\delta_{x,y+\hat{\mu}} + (r+\gamma_\mu)\delta_{x,y-\hat{\mu}}\right] , \tag{13}$$

where $x = (x_0, \vec{x})$ and $y = (y_0, \vec{y})$, $\hat{\mu}$ represents a unit vector in the μ direction, r is the Wilson parameter, and $\mathbb{1}$, γ_μ act in Dirac space.

The Dirac matrix is of size $N_t V \times N_t V$, where N_t is the temporal extent of the lattice and V the spatial volume, thus making a direct computation of its inverse infeasible except for very small lattices in low dimension. We can reduce the problem by considering the propagator of a point source at $x = (0, \vec{0})$ to another point y. This way, we can compute the correlator at zero momentum as

$$C(\delta) = \frac{1}{V}\sum_{\vec{y}}\langle\psi_{(\delta,\vec{y})}\,\overline{\psi}_0\rangle = \frac{1}{V}\sum_{\vec{y}}(D^{-1})_{(\delta,\vec{y}),0} = \frac{1}{V}\sum_{\vec{y}}\sum_{x}(D^{-1})_{(\delta,\vec{y}),x}v_x\,, \tag{14}$$

where $v_x \equiv v_{x_0,\vec{x}} = s\,\delta_{x_0,0}\,\delta_{\vec{x},\vec{0}}$ and s is a spinor. In other words, we do not need the full inverse of the Dirac operator, but rather "just" the first column.

Owing to its sparsity, it is common to employ iterative methods to compute the solution to the linear system $A\mathbf{v}_{\text{out}} = \mathbf{v}_{\text{in}}$, where $\mathbf{v}_{\text{in}}$ and $\mathbf{v}_{\text{out}}$ are vectors and A is a matrix. One popular method is known as the *Conjugate Gradient* (CG) method. A pseudo-code follows:

function name: ConjugateGradient
input : A vector $\mathbf{v}_{\text{in}}$, a Hermitian matrix A, iteration limit $N_{\max}$ and target inversion precision ϵ
output : A vector $\mathbf{v}_{\text{out}}$ and the number of iterations needed for inversion k

```
r_0 ← v_in − A v_out ;                          // initial residue
p_0 ← r_0 ;                                     // initial search direction
for 0 ≤ k < N_max do                            // iterative procedure
    α ← (r_k^T r_k) / (p_k^T A p_k);
    v_out ← v_out + α p_k ;                     // update solution to linear system
    r_{k+1} ← r_k − α A p_k ;                   // update residue
    if |r_{k+1}| < ε then                       // if desired precision has been reached
        break ;                                 // leave algorithm
    end
    β ← (r_{k+1}^T r_{k+1}) / (r_k^T r_k);
    p_{k+1} ← r_{k+1} + β p_k ;                 // update search direction
    k ← k + 1;
end
```

[2]Roughly speaking, a sparse matrix is a matrix where most entries are zero.

Note that CG requires the matrix A to be Hermitian. One common trick is then to write $A = D\,D^\dagger$ and then obtain our solution as $\mathbf{v}_{\text{result}} = D^\dagger A^{-1}\mathbf{v}_{\text{in}} = D^{-1}\mathbf{v}_{\text{in}}$.

Hint: Euclidean γ-matrices in the chiral basis

$$\gamma_0 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad \gamma_1 = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}. \tag{15}$$

Big hint: since we know how to construct each element of D and it is a sparse matrix, it is *very* convenient to write it as a function along the lines of

function name: applyDirac
input : An array `in[t][x]` of dimensions $N_t \times N_x$
output : An array `out[t][x]` of dimensions $N_t \times N_x$
for $0 \le t < N_t$ and $0 \le x < N_x$ **do** `// traverse the entire lattice`
  `out[t][x]` $\leftarrow (2r+m)$ `*in[t][x];`
  `out[t][x]` $\leftarrow$ `out[t][x]-` $(r-\gamma_0)/2$ `*in[t+1][x];`
  `out[t][x]` $\leftarrow$ `out[t][x]-` $(r+\gamma_0)/2$ `*in[t-1][x];`
  `out[t][x]` $\leftarrow$ `out[t][x]-` $(r-\gamma_1)/2$ `*in[t][x+1];`
  `out[t][x]` $\leftarrow$ `out[t][x]-` $(r+\gamma_1)/2$ `*in[t][x-1];`
end

This way we can write the vector $\mathbf{v}_{\text{out}} = D\mathbf{v}_{\text{in}}$ as `applyDirac(v_in, v_out)` or `v_out = applyDirac(v_in)`.

Note: The above algorithm does **not** include the minus signs from anti-periodic boundary conditions. You should include that in your code! A similar algorithm can be written for $D^\dagger$. Note that in our formulation the Dirac operator is real.