INTERNATIONAL CENTRE for THEORETICAL SCIENCES
TATA INSTITUTE OF FUNDAMENTAL RESEARCH

Computational Methods for Quantum Many-Body Physics

Sthitadhi Roy[1] and Arnab Sen[2]

June 3, 2023

[1]ICTS-TIFR, Bengaluru
[2]IACS, Kolkata

Lecture 3

Recap of Lecture 1-2

- **Exact diagonalisation** for spin systems
  - Constructing the basis states
  - Encoding them as binary strings
  - Efficient ways of tabulating them so that they are easy to look up [Lin Tables]
  - Implementing $U(1)$ symmetry
  - constructing the Hamiltonian

Recap of Lecture 1-2

- **Exact diagonalisation** for spin systems
  - Constructing the basis states
  - Encoding them as binary strings
  - Efficient ways of tabulating them so that they are easy to look up [Lin Tables]
  - Implementing $U(1)$ symmetry
  - constructing the Hamiltonian
- **Sparsity of Hamiltonian matrix:** representing the matrices as sparse matrices; CSR, COO formats
- **Efficient sparse matrix-vector multiplication**

Recap of Lecture 1-2

- **Exact diagonalisation** for spin systems
  - Constructing the basis states
  - Encoding them as binary strings
  - Efficient ways of tabulating them so that they are easy to look up [Lin Tables]
  - Implementing $U(1)$ symmetry
  - constructing the Hamiltonian
- **Sparsity of Hamiltonian matrix:** representing the matrices as sparse matrices; CSR, COO formats
- **Efficient sparse matrix-vector multiplication**
- **Lanczos algorithm for exact diagonalisation**
  - eigenvalues/eigenvectors near the extremities of the spectrum
  - diagonalise within a truncated Krylov subspace
  - useful for ground states and low excited states

This Lecture

Target eigenvalues/eigenvectors at arbitrary energy densities in the spectrum

- Shift-invert ED
- Polynomially filtered ED

References

- Kernel Polynomial Methods [arXiv:cond-mat/0504627v2]
- Shift-invert [arXiv:1803.05395]
- POLFED [arXiv:2005.09534]

Lanczos algorithm

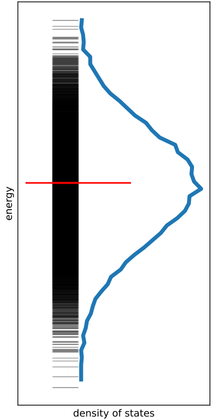
When does Lanczos work best?

- target eigenvalues are near the extremities of the spectrum
- eigenvalues near the target are well separated from each other
- density of states is very low near the target

How to target eigenvalues near the middle of the spectrum?

- not the extremities of the spectrum
- density of states very high

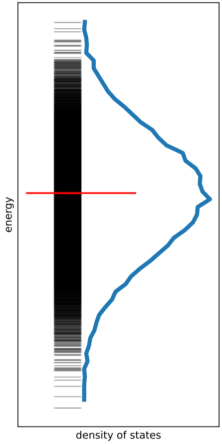
XXZ chain with $L = 14$ in $S^z = 0$ sector



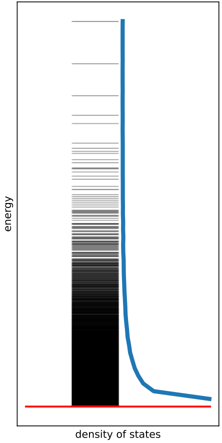
Key idea:

- transform the Hamiltonian to move the target to the extremities
- transformation keeps the eigenvectors invariant

Bare spectrum



Spectrum of $(H - \sigma \mathbb{I})^2$



$$H \to (H - \sigma \mathbb{I})^2$$

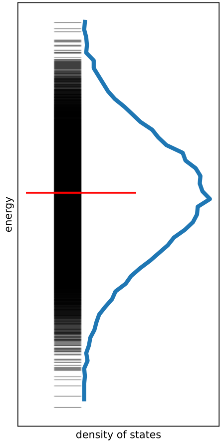
- target moved to extremities of spectrum
- but density of states very high; Lanczos will take very long to converge

Shift-Invert Exact Diagonalisation

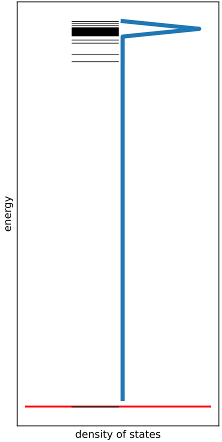
Key idea:

- transform the Hamiltonian to move the target to the extremities
- transformation keeps the eigenvectors invariant

Bare spectrum



Spectrum of $(H - \sigma\mathbb{I})^{-1}$



$$H \rightarrow (H - \sigma\mathbb{I})^{-1}$$

- target moved to extremities of spectrum
- density of states also low
- inverting a large matrix is computationally expensive
- Need to efficiently multiply

$$(H - \sigma\mathbb{I})^{-1} |\psi\rangle = |\phi\rangle$$

Shift-Invert Exact Diagonalisation

- How to efficiently multiply

$$(H - \sigma\mathbb{I})^{-1} |\psi\rangle = |\phi\rangle$$

- Given a $|\psi\rangle$ find the solution $|\phi\rangle$ to the system of equations

$$(H - \sigma\mathbb{I}) |\phi\rangle = |\psi\rangle$$

- Want to avoid inverting the matrix explicitly
- **Key step: LU decomposition of $(H - \sigma\mathbb{I})$**

$$(H - \sigma\mathbb{I}) = \mathrm{P} \cdot \mathrm{L} \cdot \mathrm{U}$$

  - P : permutation matrix
  - L : lower triangular matrix
  - U : upper triangular matrix

Two steps:

- perform the LU decomposition
- solve the system of equations of the form $A\mathbf{x} = \mathbf{y}$ using the LU decomposition

Lower triangular matrix

$$\mathrm{L} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ l_{21} & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \cdots & 1 \end{bmatrix}$$

Upper triangular matrix

$$\mathrm{U} = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_{nn} \end{bmatrix}$$

Gaussian elimination with partial pivoting

- Gaussian elimination: using row operations to eliminate the lower triangular part
- swapping rows of the matrix ⇒ necessitates the permutation matrix

$$A^{(0)} = \begin{bmatrix} 2 & 1 & 3 \\ 4 & 2 & 7 \\ -2 & 2 & -1 \end{bmatrix}$$

Next, we perform the row operations to eliminate the coefficients below the first entry in the first column:

$$\text{Step 1:} \quad R_2 \leftarrow R_2 - 2R_1$$
$$\text{Step 2:} \quad R_3 \leftarrow R_3 + R_1$$

This yields:

$$A^{(1)} = \begin{bmatrix} 2 & 1 & 3 \\ 0 & 0 & 1 \\ 0 & 3 & 2 \end{bmatrix} \quad \text{and} \quad L^{(1)} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}$$

We repeat the process to eliminate the coefficients below the second entry in the second column:

LU decomposition

$$\text{Step 3:} \quad R_3 \leftarrow R_3 - 3R_2$$

This yields:

$$A^{(2)} = \begin{bmatrix} 2 & 1 & 3 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \end{bmatrix} \quad \text{and} \quad L^{(2)} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & -3 & 1 \end{bmatrix}$$

Finally, we have the upper triangular matrix U:

$$U = A^{(2)} = \begin{bmatrix} 2 & 1 & 3 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \end{bmatrix}$$

And the lower triangular matrix L:

$$L = L^{(2)} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & -3 & 1 \end{bmatrix}$$

Solving a system of equations using LU decomposition

- We need to solve

$$A\mathbf{x} = \mathbf{y} \Rightarrow L\ U\ \mathbf{x} = \mathbf{y}$$

- Do it in two steps
  - define $U\mathbf{x} = \mathbf{z}$ and solve $L\mathbf{z} = \mathbf{y}$
  - solve $U\mathbf{x} = \mathbf{z}$
- Since L and U are lower and upper triangular matrices respectively, use forward and backward substitution to solve for $\mathbf{x}$
- once we solve for $\mathbf{x}$ we have effectively implemented

$$\mathbf{x} = A^{-1}\mathbf{y}$$

Solving a system of equations using LU decomposition

- We need to solve

$$A\mathbf{x} = \mathbf{y} \Rightarrow L\ U\ \mathbf{x} = \mathbf{y}$$

- Do it in two steps
  - define $U\mathbf{x} = \mathbf{z}$ and solve $L\mathbf{z} = \mathbf{y}$
  - solve $U\mathbf{x} = \mathbf{z}$
- Since L and U are lower and upper triangular matrices respectively, use forward and backward substitution to solve for $\mathbf{x}$
- once we solve for $\mathbf{x}$ we have effectively implemented

$$\mathbf{x} = A^{-1}\mathbf{y}$$

Solution for z using forward substitution

$$z_1 = y_1$$
$$z_2 = y_2 - L_{21} z_1$$
$$z_3 = y_3 - L_{31} z_1 - L_{32} z_2$$
$$\ddots \ddots$$
$$z_n = y_n - \sum_{j=1} n - 1 L_{nj} z_j$$

Solution for x using backward substitution

$$x_n = \frac{z_n}{U_{nn}}$$
$$x_{n-1} = \frac{z_{n-1} - U_{n-1,n} x_n}{U_{n-1,n-1}}$$
$$\ddots \ddots$$
$$x_1 = \frac{z_1 - \sum_{j=2}^{n} U_{1j} x_j}{U_{11}}$$

Summary of Shift-Invert ED

- Given a Hamiltonian H and target eigenvalue σ
- Effectively do Lanczos ED on a transformed Hamiltonian

$$H \to (H - \sigma \mathbb{I})^{-1}$$

- need to efficiently multiply $(H - \sigma \mathbb{I})^{-1}$ to vectors without losing sparsity or computing the inverse explicitly
  - LU decomposition of $(H - \sigma \mathbb{I})^{-1}$
  - Use the LU to solve for $(H - \sigma \mathbb{I})^{-1} |\psi\rangle = |\phi\rangle$ and implement the inverse

Polynomially filtered ED

Key idea: Transform the Hamiltonian using kernel polynomials which have a recursive structure

- Transformation

$$H \to P_\sigma^K(H) = \frac{1}{D}\sum_{n=0}^{K} c_n^\sigma T_n(H)$$

where $T_n(x)$ is the n^{th} Chebyshev polynomial and

- the coefficients

$$c_n^\sigma = \sqrt{4-3\delta_{n0}}\cos(n\cos^{-1}\sigma)$$

- Coefficients obtained from expanding a Dirac-delta around centred at σ in Chebyshev polynomials
- Normalisation D ensures $P_\sigma(\sigma) = 1$
- the summation above can be computed efficiently using known recursion relations for Chebyshev polynomials