Hybrid quantum-classical algorithms for solving eigenvalue problems

Nihar Shah

@niharmayur@iisc.ac.in
Department of Computational and Data Sciences
Indian Institute of Science

December 26, 2024

To Find n Smallest Eigenvalues

$$\left(-\frac{1}{2}\nabla^2 + V(x)\right)\psi_I(x) = \lambda_I \psi_I(x)$$

Assumptions

- Finite Difference Method (FDM) using a 2^{nd} order accurate Central Difference Scheme for Laplacian.
- Uniform grid: $\Delta x = \Delta y = \Delta z$.
- One-Dimensional problem.

Discretized Equation

$$\begin{bmatrix} \ddots & \ddots & 0 & \dots & 0 \\ \ddots & \ddots & \ddots & \dots & \vdots \\ 0 & -\frac{1}{2} & 1+V(x) & -\frac{1}{2} & 0 \\ \vdots & \dots & \ddots & \ddots & \ddots \\ 0 & \dots & 0 & \ddots & \ddots \end{bmatrix} \begin{bmatrix} \vdots \\ \psi(x_i - \Delta x) \\ \psi(x_i) \\ \psi(x_i + \Delta x) \\ \vdots \end{bmatrix} = \lambda \begin{bmatrix} \vdots \\ \psi(x_i - \Delta x) \\ \psi(x_i) \\ \psi(x_i + \Delta x) \\ \vdots \end{bmatrix}$$

$$\mathbf{H}\psi = \lambda\psi$$

where **H** is a Real Symmetric (Tridiagonal) Matrix.

Popular Subspace Iteration (SI) Methods

Krylov Subspace Iteration Method	Chebyshev Subspace Iteration Method
Fast	Slower
Optimal in a certain sense	Geared towards subspaces (vs individual eigenvalues)
Requires one starting vector	
Not easy to update	Updates are Easy
Changes in **H** not allowed	Tolerates changes in **H**

Table 1: Comparison between Krylov and Chebyshev subspace iteration method

Quantum Chebyshev Filtered Subspace Iteration (ChFSI) Algorithm (Part 1)

Algorithm 1 Quantum Chebyshev Filtered Subspace Iteration (ChFSI) Algorithm

1: **Input:** Matrix $\mathbf{H}$, initial guess $\mathbf{\Psi_0}^{(0)} \in \mathbb{C}^{M\times N}$, Chebyshev polynomial degree k, subspace size $N(N > \tilde{N})$.

2: **Output:** n smallest eigenvalues and corresponding eigenvectors.

3: **For iterations** $t = 0, 1, 2, \ldots, \max_{1\le i\le n} |\tilde{\lambda}_i{}^{(t)} - \tilde{\lambda}_i{}^{(t)}| \le \epsilon$ (**Convergence Criteria**)

4: **Chebyshev Filtering:**

$$\mathbf{\Psi_{\mathit{F}}}^{(t)} = T_k(\mathbf{H})\mathbf{\Psi_0}^{(t)}$$

where T_k is the Chebyshev polynomial of degree k of the first kind.

5: **Rayleigh-Ritz Projection:**

Compute Projected entries:

$$\tilde{\mathbf{H}} = \mathbf{\Psi}_{\boldsymbol{F}}^{(\boldsymbol{t})\dagger}\mathbf{H}\mathbf{\Psi}_{\boldsymbol{F}}^{(\boldsymbol{t})}$$

Quantum Chebyshev Filtered Subspace Iteration (ChFSI) Algorithm (Part 2)

1: Compute Overlap Matrix:

$$\tilde{\mathbf{S}} = \boldsymbol{\Psi}_{\boldsymbol{F}}{}^{(t)\dagger} \boldsymbol{\Psi}_{\boldsymbol{F}}{}^{(t)}$$

2: **Eigen-decomposition:** (on a classical computer)

$$\tilde{\mathbf{H}} \mathbf{Q}^{(t)} = \tilde{\mathbf{S}} \mathbf{Q}^{(t)} \tilde{\boldsymbol{\Lambda}}^{(t)}$$

3: **Quantum Subspace rotation:**

$$\boldsymbol{\Psi}_{\mathbf{0}}{}^{(t+1)} = \boldsymbol{\Psi}_{\boldsymbol{F}}{}^{(t)} \mathbf{Q}^{(t)}$$

Details

Given Eigenvalue Problem (n smallest eigenvalues):

$$\mathbf{HX} = \mathbf{\Lambda X}$$

where $\mathbf{X} \in \mathbb{C}^{m \times m}, \mathbf{H} \in \mathbb{C}^{m \times m}, \mathbf{\Lambda} \in \mathbb{C}^{m \times m}$

- Initial guess with 1 random vector $|\psi_0\rangle$ (instead of n random vectors):

$$\mathbf{\Psi_0}^{(0)} = \begin{bmatrix} \vdots & \vdots & \cdots & \vdots \\ |\psi_0\rangle & T_1(\mathbf{H})|\psi_0\rangle & \cdots & T_{n-1}(\mathbf{H})|\psi_0\rangle \\ \vdots & \vdots & \cdots & \vdots \end{bmatrix}$$

We define i^{th} column of $\mathbf{\Psi_0}^{(t)}$ as $\left|\psi_i^{(t)}\right\rangle$. Here, $\left|\psi_i^{(0)}\right\rangle = T_i(\mathbf{H})|\psi_0\rangle; \forall i = 0, \ldots, n-1.$

Continued

For iterations $t = 01, 2, \ldots,$ till convergence.

- **Chebyshev Filtering:**

$$\boldsymbol{\Psi_F}^{(t)} = T_k(\mathbf{H})\boldsymbol{\Psi_0}^{(t)} = \begin{bmatrix} \vdots & \vdots & \cdots & \vdots \\ T_k(\mathbf{H})\left|\psi_0\right\rangle & T_k(\mathbf{H})\,T_1(\mathbf{H})\left|\psi_0\right\rangle & \cdots & T_k(\mathbf{H})\,T_{n-1}(\mathbf{H})\left|\psi_0\right\rangle \\ \vdots & \vdots & \cdots & \vdots \end{bmatrix}$$

$$= \begin{bmatrix} \vdots & \vdots & \cdots & \vdots \\ T_k(\mathbf{H})\left|\psi_0^{(t)}\right\rangle & T_k(\mathbf{H})\left|\psi_1^{(t)}\right\rangle & \cdots & T_k(\mathbf{H})\left|\psi_{n-1}^{(t)}\right\rangle \\ \vdots & \vdots & \cdots & \vdots \end{bmatrix}$$

Rayleigh-Ritz Projection

- To find entries of the projected entries $\tilde{\mathbf{H}} = \boldsymbol{\Psi}_{\boldsymbol{F}}{}^{(t)\dagger}\mathbf{H}\boldsymbol{\Psi}_{\boldsymbol{F}}{}^{(t)}$, which is
  - $\tilde{H}_{ij} = \left\langle \psi_i^{(t)} \middle| T_k(\mathbf{H})\mathbf{H}T_k(\mathbf{H}) \middle| \psi_j^{(t)} \right\rangle$
  - $\tilde{S}_{ij} = \left\langle \psi_i^{(t)} \middle| T_k(\mathbf{H})\mathbf{H}T_K(\mathbf{H}) \middle| \psi_j^{(t)} \right\rangle$

  $\forall\, i, j = 0, \ldots, n-1$
- Simplify using Property of Chebyshev Polynomials:

$$T_i(x)\,T_j(x) = \frac{1}{2}\left(T_{i+j}(x) + T_{|i-j|}(x)\right)$$

- Matrix elements of $\tilde{\mathbf{H}}$ and $\tilde{\mathbf{S}}$ are linear combinations of the inner products.

$$\tilde{H}_{ij} = \frac{1}{4}\left(\left\langle \psi_i^{(t)} \middle| T_{2k+1}(\mathbf{H}) + 2T_1(\mathbf{H}) + T_{2k-1}(\mathbf{H}) \middle| \psi_j^{(t)} \right\rangle\right)$$

$$\tilde{S}_{ij} = \frac{1}{2}\left(\left\langle \psi_i^{(t)} \middle| T_{2k+1}(\mathbf{H}) + T_0(\mathbf{H}) \middle| \psi_j^{(t)} \right\rangle\right)$$

$$\tilde{H}_{ij} = \frac{1}{4}\left(\left\langle \psi_i^{(t)} \middle| T_{2k+1}(\mathbf{H}) \middle| \psi_j^{(t)} \right\rangle + 2\left\langle \psi_i^{(t)} \middle| \mathbf{H} \middle| \psi_j^{(t)} \right\rangle + \left\langle \psi_i^{(t)} \middle| T_{2k-1}(\mathbf{H}) \middle| \psi_j^{(t)} \right\rangle\right)$$

$$\tilde{S}_{ij} = \frac{1}{2}\left(\left\langle \psi_i^{(t)} \middle| T_{2k+1}(\mathbf{H}) \middle| \psi_j^{(t)} \right\rangle + \left\langle \psi_i^{(t)} \middle| \psi_j^{(t)} \right\rangle\right)$$

$\forall\, i, j = 0, \ldots, n-1$. Thus, we need to evaluate just the four inner products (can be done entirely parallel on a quantum computer) using **Amplitude Estimation Algorithm** as subroutine.

Theorem: (Block-encoding of sparse-access matrices) Gilyén et al. 2019

Let $\mathbf{H} \in \mathbb{C}^{M\times M}; M = 2^m$ be a matrix that is s_r-row-sparse and s_c-column sparse, and each value of $\mathbf{H}$ has absolute value at most 1. Suppose we have access to the following sparse access oracles acting on two $(m+1)$ qubit registers

$$O_r : |i\rangle\,|k\rangle \to |i\rangle\,|r_{ik}\rangle \quad \forall i \in [M]-1, k \in [s_r]$$

$$O_c : |l\rangle\,\left|j\right\rangle \to \left|c_{lj}\left|j\right\rangle\right\rangle \quad \forall l \in [s_c], j \in [M]-1,$$

where r_{ij} is the index for the j-th non-zero entry of the i-th row of $\mathbf{H}$, or if there are less than i nonzero entries than it is $j+2^m$, and similarly c_{ij} is the index for the i-th non-zero entry of the j-th column of $\mathbf{H}$, or if there are less than j non zero entries, then it is $i+2^m$. Additionally assume that we have access to an oracle O_A that returns the entries of A in a binary description

$$O_A : |i\rangle\,\left|j\right\rangle |0\rangle^{\otimes b} \to |i\rangle\,\left|j\right\rangle \left|a_{ij}\right\rangle \quad \forall i, j \in [M]-1$$

where a_{ij} is a b-bit binary description (for simplicity we assume here that the binary representation is exact) of the ij matrix element of A. Then we can implement a $(\sqrt{s_r s_c}, m+3, \epsilon)$ block-encoding of A with a single use of O_r, O_c, two uses of O_A and additionally using $O(m + \log^{2.5}\left(\frac{s_r s_c}{\epsilon}\right))$ one and two qubit gates while using $O(b, \log^{2.5}\left(\frac{s_r s_c}{\epsilon}\right))$ ancilla qubits.

- Recall that $\mathbf{H}$ is a banded-s-sparse matrix. This can be encoded as Unitary $\mathbf{U_H} \in HBE(s, \lceil \log s \rceil, 0)$

References

Gilyén, András et al. (June 2019). "Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics". In: *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*. STOC '19. ACM. DOI: 10.1145/3313276.3316366. URL: http://dx.doi.org/10.1145/3313276.3316366.

Thank you!

Nihar Shah

‘niharmayur@iisc.ac.in’