# The HHL algorithm (Quantum Algorithms for linear systems of equations)

↓

- Harrow, Hassidim and Lloyd algorithm. 2019

- Solve $A\vec{x}=b$ using a quantum computer.

  $A$: known, $b$: known, with $N$ variables; $\vec{x}$: unknown

  Problem: $N$ can be very large.

  "Gaussian" elimination (高斯消元法, 求解多元一次方程组)

  ↑

  best classical algorithm: $O(N^3)$

  If $A$ is sparse (No more than $S<<N$ entries per row),

  then it takes time $O(N\log N) \Rightarrow O(\frac{ks}{(\log\epsilon)}\cdot N\log N)$, where $k$ is the condition number, and $s$ is the sparsity of $A$

  minimize $|A\vec{x}-\vec{b}|$ ⇒ cojugate gradient descent

  - keep track with where you've been, and don't go back.
  - take a $x$, to see if its surrondings has a better result. do this for many times, move to the direction which looks better.

Classical algorithm

Quantum mechanical operator

↑

$A|x\rangle=|b\rangle$, $|x\rangle, |b\rangle$ are over $\log_2 N$ qubits.

($n=\log_2 N$ qubits)

$A$ is the Hermitian matrix, fulfils that $A=A^\dagger$,

where $A^\dagger$ is the conjugate transpose of $A$.

① encode

Constrnct $|x\rangle=A^{-1}|b\rangle$ ② goal

if $A=A^H$, can implement $e^{-iAt}|b\rangle$ ③ processing

↳ if $A$ is sparse, takes $O(\log_2 N)$

‖Note: to read out all components of $\vec{x}$, one needs to peaforme the procedure for at least $N$ times.

However, one is often not interested in $\vec{x}$, but in some expectation value $\vec{x}^\dagger M\vec{x}$. ↳ a quantum mechanical operator ‖

evaluate $\langle x|M|x\rangle$ ④ evaluation

Quantum algorithm

Step 1. Assume $A=A^\dagger$ ($A$ is Hermitian),

if not, solve $\begin{pmatrix}0 & A\\ A^\dagger & 0\end{pmatrix}\begin{pmatrix}0\\ \vec{x}\end{pmatrix}=\begin{pmatrix}\vec{b}\\ 0\end{pmatrix}$ to obtain $\vec{y}=\begin{pmatrix}0\\ \vec{x}\end{pmatrix}$

⇒ Can solve $A\vec{x}=\vec{b}$ for any $A$.

Step 2. $A$ is sparse ($S<<N$ entries/row)

⇒ $e^{-iAt}|b\rangle$

Step 3. If we can diagonalize $A$:

$UAU^\dagger=\begin{pmatrix}\lambda_1 & & 0\\ & \lambda_2 & \\ 0 & & \lambda_n\end{pmatrix}$ ⇒ Eigen composition 特征分解, a special case of Sigular value decomposition (SVD, 奇异值分解)

$\Rightarrow A^{-1}=U^\dagger\begin{pmatrix}\lambda_1^{-1} & & 0\\ & \lambda_2^{-1} & \\ 0 & & \lambda_n^{-1}\end{pmatrix}U$

Step4. Use Quantum phase estimation algorithm

— find eigenvalues & eigen vectors of A

$e^{-iA\otimes p}|b\rangle|0\rangle \overset{U_g}{=} \sum_j \beta_j |\psi_j\rangle|\lambda_j t\rangle \xrightarrow{\text{turn it into phase}} \sum_j \beta_j e^{i\Delta\lambda_j^{-1}}|\psi_j\rangle|\lambda_j t\rangle$

$|b\rangle = \sum_j \beta_j |\psi_j\rangle$ → eigen vectors of A

$\lambda_j$ → eigen values of A

$\Downarrow U_g^{-1}$

$\sum_j \beta_j e^{i\Delta\lambda_j^{-1}}|\psi_j\rangle|0\rangle = e^{i\Delta A^{-1}}|b\rangle|0\rangle$

$\xrightarrow[\text{enough}]{\text{if } \Delta \text{ small}} (\mathbb{I} + i\Delta A^{-1})|b\rangle|0\rangle$ (extract)

$\Downarrow$

get $A^{-1}|b\rangle$, in # of steps to accuracy $\epsilon \sim O\left(\frac{k^2 s^2}{\epsilon}\log_2 N\right)$

$k = \frac{\lambda_{max}}{\lambda_{min}}$ (condition number)

This means that if we have smaller eigen values, it takes longer time to find them.