A Simple Quantum Linear System Solver and (Tunable) VTAA

Gengzhi Yang

November 6, 2024

Contents

Recently, the paper [LS24] introduces a new quantum linear system solver algorithm that achieves the optimal query complexity on the state preparation oracles while keep a near-optimal scling in terms of the block-encoding of the input matrix. A main technique used there is the (Tunable) Variable Time Amplitude Amplification, which was originally introduced in [Amb12]. Here in this note, I try to reproduce part of their results.

1 Quantum Linear System Solver Based on QSVT

1.1 Previous Results

First of all, let us recall the quantum linear system solver introduced by QSVT [Gil+19].

Theorem 1 ([Gil+19], Corollary 69)**.** *Let $\epsilon, \delta \in (0, \frac{1}{2}]$, then there is an odd polynomial $P \in \mathbb{R}[x]$ of degree $\mathcal{O}(\frac{1}{\delta}\log(1/\epsilon))$ that is ϵ-approximating $f(x) = \frac{3}{4}\frac{\delta}{x}$ on the domain $I = [-1,1]\backslash[-\delta,\delta]$, moreover it is bounded 1 in absolute value.*

So consider a linear system problem as follows:

$$Ax = b, \tag{1}$$

if we can access A by accessing O_A and $O_A^\dagger$, where they are $(\alpha_A, a, 0)$-block-encodings that block-encode A and $A^\dagger$, consider the polynomial $p(x)$ defined in Theorem 1, we can implement the $x \to \frac{3}{4}\frac{1/\kappa}{x}$transformation of the singular values of $A^\dagger/\alpha_A$ with an ϵ accuracy, using $\mathcal{O}(\kappa\log(1/\epsilon))$ queries to O_A and $O_A^\dagger$. Note that after the transformation what we have is a $(\frac{4}{3}\|A^{-1}\|, a+1, \frac{4}{3}\|A^{-1}\|\epsilon)$-block-encoding of A^{-1}. Denote this block-encoding as $O_{A^{-1}}$, where

$$(\langle 0^{a+1}| \otimes I)O_{A^{-1}}(|0^{a+1}\rangle \otimes I) = \frac{1}{\frac{4}{3}\|A^{-1}\|}A^{-1} + \Lambda, \tag{2}$$

and $\|\Lambda\| \leq \epsilon$. Suppose the oracle O_b prepares $|b\rangle$,

$$O_{A^{-1}}(|0^{a+1}\rangle \otimes O_b|0\rangle) = |0^{a+1}\rangle\,(\frac{1}{\frac{4}{3}\|A^{-1}\|}A^{-1} + \Lambda)\,|b\rangle + |\perp\rangle\,. \tag{3}$$

So in order to achieve an error of η, namely

$$\left\| \frac{(\frac{1}{\frac{4}{3}\|A^{-1}\|}A^{-1} + \Lambda)\left|b\right\rangle}{\|(\frac{1}{\frac{4}{3}\|A^{-1}\|}A^{-1} + \Lambda)\left|b\right\rangle\|} - \frac{A^{-1}\left|b\right\rangle}{\|A^{-1}\left|b\right\rangle\|} \right\| \leq \eta, \tag{4}$$

i.e.

$$\left\| \frac{(A^{-1} + \frac{4}{3}\|A^{-1}\|\Lambda)\left|b\right\rangle}{\|(A^{-1} + \frac{4}{3}\|A^{-1}\|\Lambda)\left|b\right\rangle\|} - \frac{A^{-1}\left|b\right\rangle}{\|A^{-1}\left|b\right\rangle\|} \right\| \leq \eta, \tag{5}$$

$$\left\| \frac{(A^{-1} + \frac{4}{3}\|A^{-1}\|\Lambda)\left|b\right\rangle}{\|(A^{-1} + \frac{4}{3}\|A^{-1}\|\Lambda)\left|b\right\rangle\|} - \frac{(A^{-1} + \frac{4}{3}\|A^{-1}\|\Lambda)\left|b\right\rangle}{\|A^{-1}\left|b\right\rangle\|} + \frac{(A^{-1} + \frac{4}{3}\|A^{-1}\|\Lambda)\left|b\right\rangle}{\|A^{-1}\left|b\right\rangle\|} - \frac{A^{-1}\left|b\right\rangle}{\|A^{-1}\left|b\right\rangle\|} \right\| \leq \eta. \tag{6}$$

Consider

$$\begin{aligned} &\left\| \frac{(A^{-1} + \frac{4}{3}\|A^{-1}\|\Lambda)\left|b\right\rangle}{\|(A^{-1} + \frac{4}{3}\|A^{-1}\|\Lambda)\left|b\right\rangle\|} - \frac{(A^{-1} + \frac{4}{3}\|A^{-1}\|\Lambda)\left|b\right\rangle}{\|A^{-1}\left|b\right\rangle\|} \right\| \\ &\leq \|(A^{-1} + \frac{4}{3}\|A^{-1}\|\Lambda)\left|b\right\rangle\| \frac{\frac{4}{3}\|A^{-1}\|\epsilon}{\|A^{-1}\left|b\right\rangle\|^2} \leq \frac{8\|A^{-1}\|\epsilon}{3\|A^{-1}\left|b\right\rangle\|}, \end{aligned} \tag{7}$$

the last inequality holds given $\frac{4}{3}\|A^{-1}\|\epsilon \leq \|A^{-1}\left|b\right\rangle\|$.

For the remaining part,

$$\left\| \frac{(A^{-1} + \frac{4}{3}\|A^{-1}\|\Lambda)\left|b\right\rangle}{\|A^{-1}\left|b\right\rangle\|} - \frac{A^{-1}\left|b\right\rangle}{\|A^{-1}\left|b\right\rangle\|} \right\| \leq \frac{4\|A^{-1}\|\epsilon}{3\|A^{-1}\left|b\right\rangle\|}. \tag{8}$$

So taking $\epsilon = \mathcal{O}(\frac{\|A^{-1}|b\rangle\|}{\|A^{-1}\|}\eta)$ suffices to make Equation 4 hold.

If we directly use the amplitude amplification to boost the success probability, we need

$$\mathcal{O}\left(\frac{1}{\|(\frac{1}{\frac{4}{3}\|A^{-1}\|}A^{-1} + \Lambda)\left|b\right\rangle\|}\right) = \mathcal{O}(\frac{\|A^{-1}\|}{\|A^{-1}\left|b\right\rangle\|}) \tag{9}$$

queries to $O_{A^{-1}}$ and O_b, notice that we then need $\mathcal{O}(\frac{\kappa\|A^{-1}\|}{\|A^{-1}|b\rangle\|}\log(\frac{\|A^{-1}\|}{\|A^{-1}|b\rangle\|\eta}))$ queries to O_A and $O_A^\dagger$. In summary, we know that, in order to achieve an accuracy ϵ in $\frac{A^{-1}|b\rangle}{\|A^{-1}|b\rangle\|}$, we need

$$\mathcal{O}\left(\frac{\|A^{-1}\|}{\|A^{-1}\left|b\right\rangle\|}\mathrm{Cost}(O_b) + \frac{\kappa\|A^{-1}\|}{\|A^{-1}\left|b\right\rangle\|}\log\left(\frac{\|A^{-1}\|}{\|A^{-1}\left|b\right\rangle\|\eta}\right)\mathrm{Cost}(O_A)\right) \tag{10}$$

in total.

Remark 2. *Notice that*

$$\frac{\|A^{-1}\|}{\|A^{-1}\left|b\right\rangle\|} \leq \|A\|\|A^{-1}\| = \kappa, \tag{11}$$

meaning in the worst case senario, the linear system solver based on QSVT scales quadratically on the condition number. While for the queries to the state preparation oracle, it achieves the optimal scaling.

1.2 Optimal Scaling with Block Preconditioning

Much surprisingly, compared with [Cos+22; Dal24], the optimal scaling ($\mathcal{O}(\kappa\log(1/\epsilon))$) about O_A can be easily achieved with the technique called Block Preconditioning.

For the QLSP problem

$$A\left|x\right\rangle = \left|b\right\rangle, \tag{12}$$

consider $S = s\ket{b}\bra{b} + (I - \ket{b}\bra{b}) = O_b(\frac{1-s}{2}(I - 2\ket{0}\bra{0}) + \frac{1+s}{2}I)O_b^\dagger$, for some $s \in (0,1)$. Then it is easy to verify that

$$S^{-1} = \frac{1}{s}\ket{b}\bra{b} + (I - \ket{b}\bra{b}), \tag{13}$$

since

$$(s\ket{b}\bra{b} + (I - \ket{b}\bra{b}))\left(\frac{1}{s}\ket{b}\bra{b} + (I - \ket{b}\bra{b})\right) = I. \tag{14}$$

The important observation is that

$$(SA)^{-1}\ket{b} = A^{-1}S^{-1}\ket{b} = \frac{1}{s}A^{-1}\ket{b}, \tag{15}$$

meaning S can be used as a *preconditioner*. The above equation indicates $\|(SA)^{-1}\ket{b}\| = \frac{1}{s}\|A^{-1}\ket{b}\|$. The rest is to bound $\frac{\|(SA)^{-1}\|}{\|(SA)^{-1}\ket{b}\|}$.

$$\begin{aligned}\|(SA)^{-1}\| = \|A^{-1}S^{-1}\| = \|A^{-1}(\frac{1}{s}\ket{b}\bra{b} + (I - \ket{b}\bra{b}))\| = \|\frac{1}{s}A^{-1}\ket{b}\bra{b} + A^{-1}(I - \ket{b}\bra{b})\| \\ \leq \left(\frac{1}{s^2}\|A^{-1}\ket{b}\bra{b}\|^2 + \|A^{-1}(I - \ket{b}\bra{b})\|^2\right)^{1/2} \leq \left(\frac{1}{s^2}\|A^{-1}\ket{b}\|^2 + \|A^{-1}\|^2\right)^{1/2}.\end{aligned} \tag{16}$$

Suppose we have an estimation to $t = \mathcal{O}(\|A^{-1}\ket{b}\|)$, take $s = \frac{t}{\alpha_{A^{-1}}}$, we have

$$\|(SA)^{-1}\| = \mathcal{O}(\alpha_{A^{-1}}), \quad \|(SA)^{-1}\ket{b}\| = \mathcal{O}(\alpha_{A^{-1}}) \tag{17}$$

and

$$\frac{\|(SA)^{-1}\|}{\|(SA)^{-1}\ket{b}\|} = \mathcal{O}(1). \tag{18}$$

Notice that in the current scenario, we still have $\kappa_{SA} \approx \kappa_A$, so the final cost should be

$$\mathcal{O}\left(\kappa\log(1/\epsilon)\mathrm{Cost}(O_b) + \kappa\log\left(1/\epsilon\right)\mathrm{Cost}(O_A)\right). \tag{19}$$

Remark 3. *We should notice that the preconditioner S actually does not improve the condition number of A, but it boost the parameter $\kappa \leq \frac{\|A^{-1}\|}{|A^{-1}\ket{b}\rangle}$ into $\frac{(SA)^{-1}}{(SA)^{-1}\ket{b}} = \mathcal{O}(1)$. The reason we cannot achieve the optimal scaling on the query complexity about O_b is we need O_b to construct S.*

2 (Tunable) Variable Time Amplitude Amplification

If we have a sequence of quantum algorithms $A_1, A_2, \cdots, A_m$, a starting state $\ket{\psi_0}$, and the success is flagged by a projection $\overline{\Pi_b} = I - \Pi_b$, naively apply the algorithms leads to a success probability of

$$p_{\mathrm{succ}} = \|\overline{\Pi_b}A_m\cdots A_1\ket{\psi_0}\|^2. \tag{20}$$

Then it is quite easy to see that we may apply $\mathcal{O}(1/\sqrt{p_{\mathrm{succ}}})$ rounds of vanalla amplitude amplification, which leads to a total cost of

$$\mathcal{O}\left(\frac{1}{\sqrt{p_{\mathrm{succ}}}}\mathrm{Cost}(\ket{\psi_0}) + \frac{1}{\sqrt{p_{\mathrm{succ}}}}\sum_{j=1}^{m}\mathrm{Cost}(A_j)\right). \tag{21}$$

2.1 VTAA

In [LS24], they give a formal definition of Variable time algorithm amplification. Here I present it without any modification:

Definition 4 (Variable Time Algorithm and Amplification)**.** *A variable time quantum algorithm is a 3-tuple* $\left(\{\Pi_j\}_{j=0}^m, \Pi_b, \{A_j\}_{j=0}^m\right)$ *satisfying the following axioms.*

1. Π_j *are orthogonal projectoins partially ordered as* $0 = \Pi_0 \le \Pi_1 \le \cdots \le \Pi_m = I$*.*
2. Π_b *is an orthogonal projection commuting with all* Π_j*:* $\Pi_b\Pi_j = \Pi_j\Pi_b$*.*
3. A_j *are unitaries such that* $A_j\Pi_{j-1} = \Pi_{j-1}$*,* $A_0 = I$*.*

   A variable time amplification algorithm is a 5-tuple $\left(\{\Pi_j\}_{j=0}^m, \Pi_b, \{A_j\}_{j=0}^m, \{\widetilde{A_j}\}_{j=0}^m, |\psi_0\rangle\right)$ *that additionally satisfies*

4. $\widetilde{A_j}$ *are unitaries such that* $\frac{\overline{\Pi_j\Pi_b}\widetilde{A_j}|\psi_0\rangle}{\|\overline{\Pi_j\Pi_b}\widetilde{A_j}|\psi_0\rangle\|} = \frac{\overline{\Pi_j\Pi_b}A_j\widetilde{A_{j-1}}|\psi_0\rangle}{\|\overline{\Pi_j\Pi_b}A_j\widetilde{A_{j-1}}|\psi_0\rangle\|}$*.*

It is worth noting here that by letting $A_j = \Pi_{j-1} + \overline{\Pi_{j-1}}A_j\overline{\Pi_{j-1}}$ suffices to keep $A_j\Pi_{j-1} = \Pi_{j-1}$. One should understand this as A_j is a unitary controlled on $\overline{\Pi_{j-1}}$. So the the registers that algorithms $\{A_j\}_{j=1}^m$ are acting on gradually shrinks. VTAA is an algorithm that captures this property. And the requirement 4 guarantees that finally we can get the state we want.

Another important property given by the axioms is

$$\frac{\left\|\overline{\Pi_h\Pi_b}A_h\cdots A_{j+1}\widetilde{A_j}\,|\psi_0\rangle\right\|}{\left\|\overline{\Pi_k\Pi_b}A_k\cdots A_{j+1}\widetilde{A_j}\,|\psi_0\rangle\right\|} = \frac{\left\|\overline{\Pi_h\Pi_b}A_h\cdots A_{l+1}\widetilde{A_l}\,|\psi_0\rangle\right\|}{\left\|\overline{\Pi_k\Pi_b}A_k\cdots A_{l+1}\widetilde{A_l}\,|\psi_0\rangle\right\|}, \quad \text{for } 0 \le l, j \le k, h \le m. \tag{22}$$

And a specital form of it is

$$\frac{\left\|\overline{\Pi_{l+1}\Pi_b}A_{l+1}\widetilde{A_l}\,|\psi_0\rangle\right\|}{\left\|\overline{\Pi_l\Pi_b}\widetilde{A_l}\,|\psi_0\rangle\right\|} = \frac{\left\|\overline{\Pi_{l+1}\Pi_b}A_{l+1}\cdots A_1\,|\psi_0\rangle\right\|}{\left\|\overline{\Pi_l\Pi_b}A_l\cdots A_1\,|\psi_0\rangle\right\|}, \tag{23}$$

meaning the potentailly good amplitudes remains the same, regardless of whether we consider the pre- or poset-amplified algorithms.

A natural choice of $\widetilde{A_j}$ is

$$\widetilde{A_j} = \left(-\left(I - 2A_j\widetilde{A_{j-1}}\,|\psi_0\rangle\langle\psi_0|\,\widetilde{A_{j-1}}^\dagger A_j^\dagger\right)\left(I - 2\overline{\Pi_j\Pi_b}\right)\right)^{r_j} A_j\widetilde{A_{j-1}}, \tag{24}$$

which can be understood as performing the amplitude amplification at the intermediate state j.

Definition 5 (Variable Time Nested Amplitude Amplification)**.** *A variable time nested amplitude amplification is a* 5*-tuple* $\left(\{\Pi_j\}_{j=0}^m, \Pi_b, \{A_j\}_{j=0}^m, \{r_j\}_{j=0}^m, |\psi_0\rangle\right)$ *that additionally satisfies*

4 r_j *are nonnegative integers for* $j = 1, \cdots, m$*, which define*

$$\begin{aligned} \widetilde{A_0} &= I, \\ \widetilde{A_j} &= \left(-\left(I - 2A_j\widetilde{A_{j-1}}\,|\psi_0\rangle\langle\psi_0|\,\widetilde{A_{j-1}}^\dagger A_j^\dagger\right)\left(I - 2\overline{\Pi_j\Pi_b}\right)\right)^{r_j} A_j\widetilde{A_{j-1}}. \end{aligned} \tag{25}$$

The cost of the algorithm thus becomes

$$\mathrm{Cost}(\widetilde{A_j}\,|\psi_0\rangle) = (2r_j+1)\left(\mathrm{Cost}(A_j) + \mathrm{Cost}\left(\widetilde{A_{j-1}}\,|\psi_0\rangle\right)\right). \tag{26}$$

Unwrap the recursion, we obtain a variable time nested amplitude amplification $\widetilde{A_m}$, which has a query cost of

$$\begin{aligned}
&\mathrm{Cost}(\widetilde{A_m}\,|\psi_0\rangle) = (2r_m+1)\left(\mathrm{Cost}(A_m) + \mathrm{Cost}\left(\widetilde{A_{m-1}}\,|\psi_0\rangle\right)\right)\\
&= (2r_m+1)\left(\mathrm{Cost}(A_m) + (2r_{m-1}+1)\left(\mathrm{Cost}(A_{m-1}) + \mathrm{Cost}\left(\widetilde{A_{m-2}}\,|\psi_0\rangle\right)\right)\right)\\
&= \sum_{j=1}^{m}\prod_{k=j}^{m}(2r_k+1)\mathrm{Cost}(A_j) + \prod_{k=1}^{m}(2r_k+1)\mathrm{Cost}(|\psi_0\rangle).
\end{aligned} \tag{27}$$

Now it comes to the time to compare Equation 21 and Equation 27. To analyze Equation 27, the first thing we want to assure is that we did not overshoot. Because

$$\left\|\overline{\Pi_j\Pi_b}\widetilde{A_j}\,|\psi_0\rangle\right\| = \sin\left((2r_j+1)\arcsin\left(\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A_{j-1}}\,|\psi_0\rangle\right\|\right)\right), \tag{28}$$

let us try to restrict

$$(2r_j+1)\arcsin\left(\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A_{j-1}}\,|\psi_0\rangle\right\|\right) \le \frac{\pi}{2}. \tag{29}$$

Because $x \le \sin(\frac{\pi}{2}x)$ holds for $x \in [-1,1]$, the inequality above holds if

$$(2r_j+1)\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A_{j-1}}\,|\psi_0\rangle\right\| \le 1. \tag{30}$$

Define the loss factor as

$$\mathrm{loss}_j = \frac{\sin\left((2r_j+1)\arcsin\left(\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A_{j-1}}\,|\psi_0\rangle\right\|\right)\right)}{(2r_j+1)\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A_{j-1}}\,|\psi_0\rangle\right\|} = \frac{\left\|\overline{\Pi_j\Pi_b}\widetilde{A_j}\,|\psi_0\rangle\right\|}{(2r_j+1)\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A_{j-1}}\,|\psi_0\rangle\right\|}, \tag{31}$$

use the estimation of Equation 23, we have

$$\begin{aligned}
&\prod_{j=1}^{m}\mathrm{loss}_j\\
&= \frac{\left\|\overline{\Pi_m\Pi_b}\widetilde{A_m}\,|\psi_0\rangle\right\|}{(2r_m+1)\left\|\overline{\Pi_m\Pi_b}A_m\widetilde{A_{m-1}}\,|\psi_0\rangle\right\|}\frac{\left\|\overline{\Pi_{m-1}\Pi_b}\widetilde{A_{m-1}}\,|\psi_0\rangle\right\|}{(2r_{m-1}+1)\left\|\overline{\Pi_{m-1}\Pi_b}A_{m-1}\widetilde{A_{m-2}}\,|\psi_0\rangle\right\|}\cdots\frac{\left\|\overline{\Pi_1\Pi_b}\widetilde{A_1}\,|\psi_0\rangle\right\|}{(2r_1+1)\left\|\overline{\Pi_1\Pi_b}A_1\widetilde{A_0}\,|\psi_0\rangle\right\|}\\
&= \left(\prod_{j=1}^{m}\left(\frac{1}{2r_j+1}\right)\right)\left(\prod_{l=1}^{m-1}\frac{\|\overline{\Pi_l\Pi_b}\|A_l\cdots A_1\,|\psi_0\rangle}{\|\overline{\Pi_{l+1}\Pi_b}\|A_{l+1}\cdots A_1\,|\psi_0\rangle}\right)\left(\frac{\left\|\overline{\Pi_m\Pi_b}\widetilde{A_m}\,|\psi_0\rangle\right\|}{\|\overline{\Pi_1\Pi_b}A_1\,|\psi_0\rangle\,\|}\right) = \frac{\left\|\overline{\Pi_m\Pi_b}\widetilde{A_m}\,|\psi_0\rangle\right\|}{\|\overline{\Pi_m\Pi_b}A_m\cdots A_1\,|\psi_0\rangle\|}\prod_{j=1}^{m}\left(\frac{1}{2r_j+1}\right).
\end{aligned} \tag{32}$$

If the above equation has an estimation of $\Omega(1)$, we can estimate Equation 27. Note that similarly we have

$$\prod_{j=1}^{k}\mathrm{loss}_j = \frac{\left\|\overline{\Pi_k\Pi_b}\widetilde{A_k}\,|\psi_0\rangle\right\|}{\|\overline{\Pi_k\Pi_b}A_k\cdots A_1\,|\psi_0\rangle\|}\prod_{j=1}^{k}\left(\frac{1}{(2r_j+1)}\right). \tag{33}$$

Thus we can estimate

$$\prod_{j=1}^{m}\mathrm{loss}_j = \frac{\left\|\overline{\Pi_m\Pi_b}\widetilde{A_m}\,|\psi_0\rangle\right\|}{\|\overline{\Pi_m\Pi_b}A_m\cdots A_1\,|\psi_0\rangle\|}\prod_{j=1}^{m}\left(\frac{1}{2r_j+1}\right) = \frac{\left\|\overline{\Pi_m\Pi_b}\widetilde{A_m}\,|\psi_0\rangle\right\|}{(p_{\mathrm{succ}})^{1/2}}\prod_{j=1}^{m}\left(\frac{1}{2r_j+1}\right), \tag{34}$$

$$\prod_{j=k}^{m}\text{loss}_j = \frac{1}{(p_{\text{succ}})^{1/2}}\left\|\overline{\Pi_{k-1}\Pi_b}A_{k-1}\cdots A_1\left|\psi_0\right\rangle\right\| \frac{\left\|\overline{\Pi_m\Pi_b}\widetilde{A_m}\left|\psi_0\right\rangle\right\|}{\left\|\overline{\Pi_{k-1}\Pi_b}\widetilde{A_{k-1}}\left|\psi_0\right\rangle\right\|}\prod_{j=k}^{m}\left(\frac{1}{2r_j+1}\right). \tag{35}$$

In the original VTAA [Amb12], they choose r_j to be

$$\frac{1}{3\sqrt{m}} \le (2r_j+1)\|\overline{\Pi_j\Pi_b}A_j\widetilde{A_{j-1}}\left|\psi_0\right\rangle\| \le \frac{1}{\sqrt{m}}. \tag{36}$$

Now let us analyze the lower bound of $\prod_{j=1}^{m}\text{loss}_j$. To do so, we need a proposition of the Dirichlet kernel.

Lemma 6 (Tight bounds on the Dirichlet kernel)**.** *For any $\rho \ge 3$ and real number $\theta \ge 0$ such that $0 \le \rho\theta \le \frac{\pi}{2}$, we have*

$$1-\frac{1}{6}\rho^2\sin^2(\theta) \le \frac{\sin(\rho\theta)}{\rho\sin(\theta)} \le 1-\frac{4\pi-8}{\pi^3}\rho^2\sin^2(\theta). \tag{37}$$

Then we know

$$1-\frac{1}{6}(2r_j+1)^2\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A_{j-1}}\left|\psi_0\right\rangle\right\|^2 \le \text{loss}_j \le 1-\frac{4\pi-8}{\pi^3}(2r_j+1)^2\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A_{j-1}}\left|\psi_0\right\rangle\right\|^2. \tag{38}$$

Thus

$$1 \ge \prod_{j=1}^{m}\text{loss}_j \ge \prod_{j=1}^{m}\left(1-\frac{1}{6}(2r_j+1)^2\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A_{j-1}}\left|\psi_0\right\rangle\right\|^2\right) = \Omega(1), \tag{39}$$

since $\sum_{j=1}^{m}(2r_j+1)^2\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A_{j-1}}\left|\psi_0\right\rangle\right\|^2 \le 1$.

Now we are ready to bound Equation 27.

$$\Omega(1) = \prod_{j=1}^{m}\text{loss}_j = \frac{\left\|\overline{\Pi_m\Pi_b}\widetilde{A_m}\left|\psi_0\right\rangle\right\|}{(p_{\text{succ}})^{1/2}}\prod_{j=1}^{m}\left(\frac{1}{2r_j+1}\right) \le \frac{1}{(p_{\text{succ}})^{1/2}}\prod_{j=1}^{m}\left(\frac{1}{2r_j+1}\right), \tag{40}$$

leading to the estimation of

$$\prod_{j=1}^{m}(2r_j+1) \le \frac{1}{(p_{\text{succ}})^{1/2}}. \tag{41}$$

Furthermore,

$$\begin{aligned}\Omega(1) = \prod_{j=k}^{m}\text{loss}_j &= \frac{1}{(p_{\text{succ}})^{1/2}}\left\|\overline{\Pi_{k-1}\Pi_b}A_{k-1}\cdots A_1\left|\psi_0\right\rangle\right\| \frac{\left\|\overline{\Pi_m\Pi_b}\widetilde{A_m}\left|\psi_0\right\rangle\right\|}{\left\|\overline{\Pi_{k-1}\Pi_b}\widetilde{A_{k-1}}\left|\psi_0\right\rangle\right\|}\prod_{j=k}^{m}\left(\frac{1}{2r_j+1}\right)\\ &\le \frac{\sqrt{m}}{(p_{\text{succ}})^{1/2}}\left\|\overline{\Pi_{k-1}\Pi_b}A_{k-1}\cdots A_1\left|\psi_0\right\rangle\right\|\prod_{j=k}^{m}\left(\frac{1}{2r_j+1}\right).\end{aligned} \tag{42}$$

This gives us

$$\prod_{j=k}^{m}(2r_j+1) \le \frac{\sqrt{m}}{(p_{\text{succ}})^{1/2}}\left\|\overline{\Pi_{k-1}\Pi_b}A_{k-1}\cdots A_1\left|\psi_0\right\rangle\right\|. \tag{43}$$

So the final query complexity should be

$$\mathcal{O}\left(\frac{1}{(p_{\text{succ}})^{1/2}}\text{Cost}(A_1\left|\psi_0\right\rangle) + \left(\frac{m}{p_{\text{succ}}}\right)^{1/2}\sum_{j=2}^{m}\|\overline{\Pi_{j-1}\Pi_b}A_{j-1}\cdots A_1\left|\psi_0\right\rangle\|\text{Cost}(A_j)\right). \tag{44}$$

Gengzhi: One must be aware that in the query complexity I computed above, it is slightly different to that in [LS24]: I did not have a $\sqrt{m}$ term in front of the state preparation oracle and A_1.

At a first glance, compared with Equation 21, one would thought the complexity of VTAA is higher. So another property comes into play:

$$1 = \|\overline{\Pi_0 \Pi_b} |\psi_0\rangle \| \geq \|\overline{\Pi_1 \Pi_b} A_1 |\psi_0\rangle \| \geq \cdots \geq \|\overline{\Pi_m \Pi_b} A_m \cdots A_1 |\psi_0\rangle \| = \|\overline{\Pi_b} A_m \cdots A_1 |\psi_0\rangle \| = (p_{\text{succ}})^{1/2}. \tag{45}$$

Thus it requires a more detailed analysis to compare those two algorithms.

Remark 7. *Here in the description of the algorithm (VTAA), the norm of the final state* $\left\|\widetilde{A_m} |\psi_0\rangle\right\|$ *scales about* $\mathcal{O}(\frac{1}{\sqrt{m}})$. *In practice, one may use a different amplitude amplification schedule at the final step thus the norm is close to unit.*

2.2 Tunable VTAA

The difference that the Tunable VTAA made is setting a tighter threshold that determines the amplitude amplification schedule for each step.

Definition 8 (Tunable Variable Time Amplitude Amplification)**.** *A tunable variable time amplitude amplification is a 5-tuple* $\left(\{\Pi_j\}_{j=0}^m, \Pi_b, \{A_j\}_{j=0}^m, \{\alpha_j\}_{j=0}^m, |\psi_0\rangle\right)$ *that additionally satisfies*

4 α_j *are nonnegative real numbers for* $j = 1, \cdots, m$ *and* $\alpha_0 = 1$, *which define*

$$\begin{aligned} r_j &= \min\left(r \in \mathbb{Z}_{\geq 0} \middle| (2r+1) \left\|\overline{\Pi_j \Pi_b} A_j \widetilde{A_{j-1}} |\psi_0\rangle\right\| \geq \frac{\sqrt{\alpha_j}}{3}\right), \\ \widetilde{A_0} &= I, \\ \widetilde{A_j} &= \left(-\left(I - 2A_j \widetilde{A_{j-1}} |\psi_0\rangle \langle\psi_0| \widetilde{A_{j-1}}^\dagger A_j^\dagger\right)\left(I - 2\overline{\Pi_j \Pi_b}\right)\right)^{r_j} A_j \widetilde{A_{j-1}}. \end{aligned} \tag{46}$$

Clearly, instead of directly choose every $\alpha_j = \frac{1}{\sqrt{m}}$, it is more flexible to introduce some freedom here by allowing different threshold for each step.

To show there is no overshoot, one must admit $\sum_{j=1}^m \alpha_j = \mathcal{O}(1)$ and $\alpha_j \leq 1$. Then, by definition, it is easy to show that

$$(2r_j + 1) \left\|\overline{\Pi_j \Pi_b} A_j \widetilde{A_{j-1}} |\psi_0\rangle\right\| \leq \sqrt{\alpha_j} \leq 1. \tag{47}$$

Moreover, one can show the equivalence between VTAA and Tunable VTAA, which is kind of obvious by just looking at the definition, since it switches from choosing $\{r_j\}_{j=1}^m$ to choosing $\{\alpha_j\}_{j=1}^m$.

Since nontrivial amplitude amplification is not performed on every step, let say they only happen at $s_1, \cdots, s_l$, where $s_0 = 0 < 1 \leq s_1 \leq \cdots \leq s_l = m$. Note that I assume that at the final state, we will always do the amplitude amplification(Even though it might be a trivial step). This makes the following complexity slightly different to what they have in [LS24]. Pre-merge all other trilvial steps, the cost becomes

$$\mathrm{Cost}(\widetilde{A_m} |\psi_0\rangle) = \sum_{v=1}^{l} \prod_{u=v}^{l} (2r_{s_u} + 1)\mathrm{Cost}(A_{s_v} \cdots A_{s_{v-1}+1}) + \prod_{u=1}^{l} (2r_{s_u} + 1)\mathrm{Cost}(|\psi_0\rangle). \tag{48}$$

To see what exactly the cost is, we, again, need to dive into the analysis of the loss factor. Similarly, with $\sum_u \alpha_{s_u} = \mathcal{O}(1)$, we know the product of the loss factors is $\Omega(1)$.

The computation is basically the same, except we need to re-estimate

$$\|\overline{\Pi_k \Pi_b} \widetilde{A_k} |\psi_0\rangle \| = \Omega(\sqrt{\alpha_k}). \tag{49}$$

So the final complexity is

$$\mathrm{Cost}(\widetilde{A_m}\,|\psi_0\rangle) = \sum_{v=2}^{l} \frac{1}{(\alpha_{s_v} p_{\mathrm{succ}})^{1/2}} \mathrm{Cost}(A_{s_v} \cdots A_{s_{v-1}+1}) + \frac{1}{(p_{\mathrm{succ}})^{1/2}} \left(\mathrm{Cost}(|\psi_0\rangle) + \mathrm{Cost}(A_{s_1} \cdots A_1)\right). \tag{50}$$

Another important observation is about how many stages we are performing these non-trivial amplitude amplifications. Leverage Equation 34 again,

$$3^l \leq \prod_{u=1}^{l} (2r_{s_u} + 1) = \mathcal{O}\left(\frac{1}{(p_{\mathrm{succ}})^{1/2}}\right) \tag{51}$$

gives us the estimation

$$l = \mathcal{O}\left(\log_3\left(\frac{1}{(p_{\mathrm{succ}})^{1/2}}\right)\right). \tag{52}$$

It is interesting to lower bound the complexity of Tunable VTAA 50. Use the so-called weighted mean inequality, one may show the cost can be optimized to

$$\begin{aligned}\mathrm{Cost}(\widetilde{A_m}\,|\psi_0\rangle) = \mathcal{O}\Bigg(& \frac{1}{(p_{\mathrm{succ}})^{1/2}} \left(\sum_{v=2}^{l} \left(\|\overline{\Pi_{s_{v-1}}\Pi_b} A_{s_{v-1}} \cdots A_1 \,|\psi_0\rangle\, \|\mathrm{Cost}(A_{s_v} \cdots A_{s_{v-1}+1})\right)^{2/3}\right)^{3/2} \\ &+ \frac{1}{(p_{\mathrm{succ}})^{1/2}} \left(\mathrm{Cost}(|\psi_0\rangle) + \mathrm{Cost}(A_{s_1} \cdots A_1)\right)\Bigg).\end{aligned} \tag{53}$$

I omitted the exact computation here since it can be easily found in the paper [LS24] and the proof itself is to directly apply the inequality.

References

[Amb12] Andris Ambainis. "Variable time amplitude amplification and quantum algorithms for linear algebra problems". In: *STACS'12 (29th Symposium on Theoretical Aspects of Computer Science)*. Vol. 14. LIPIcs. 2012, pp. 636–647.

[Cos+22] Pedro CS Costa et al. "Optimal scaling quantum linear-systems solver via discrete adiabatic theorem". In: *PRX quantum* 3.4 (2022), p. 040303.

[Dal24] Alexander M Dalzell. "A shortcut to an optimal quantum linear system solver". In: *arXiv preprint arXiv:2406.12086* (2024).

[Gil+19] András Gilyén et al. "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*. 2019, pp. 193–204.

[LS24] Guang Hao Low and Yuan Su. "Quantum linear system algorithm with optimal queries to initial state preparation". In: *arXiv preprint arXiv:2410.18178* (2024).