Variational Quantum Linear Solvers

Joakim Colpier Anna Ingmer Boye

Delft University of Technology, The Netherlands

Contents

1. Previous Research

VQLS algorithm

Goal

Use Dynamic ansatz in Variational Quantum Linear solver to solve $\mathbf{A}\bar{x} = \bar{b}$, and test performance.

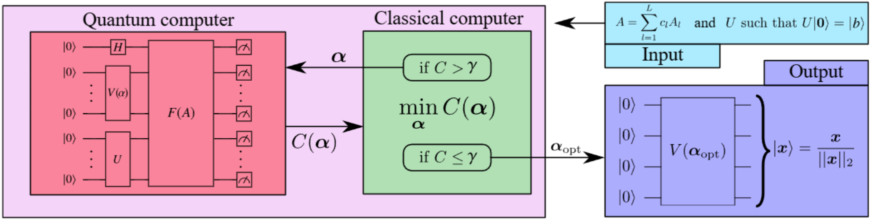


Figure: Schematic diagram for the VQLS algorithm[1]

[1] C. Bravo-Prieto, R. LaRose, M. Cerezo, Y. Subasi, L. Cincio, and P. J. Coles (Nov. 2023). "Variational Quantum Linear Solver". In: vol. 7. Verein zur Förderung des Open Access Publizierens in den Quantenwissenschaften, p. 1188. DOI: 10.22331/q-2023-11-22-1188

Comprehensive Library of Variational LSE Solvers

Goal

Use Dynamic ansatz in Variational Quantum Linear solver to solve $\mathbf{A}\bar{x} = \bar{b}$, and test performance.

We use the results from the paper Meyer et al.[2] It implements a Variational Quantum Linear equation solver with two different cost functions (global & local) and a 'dynamic Ansatz':

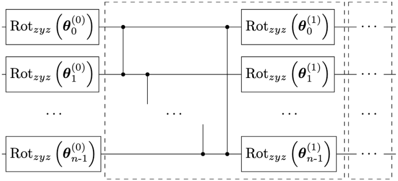


Figure: The dynamic circuit with dynamic depth d

[2] N. Meyer, M. Röhn, J. Murauer, A. Plinge, C. Mutschler, and D. D. Scherer (Sept. 2024). "Comprehensive Library of Variational LSE Solvers". In: *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)*. IEEE, 1–4. DOI: 10.1109/qce60285.2024.10242

Dynamic Ansatz

Idea:

- Start with only one layer, solve for the coarser problem
- Add a layer and use the solution to the coarser problem as an Ansatz for the finer problem
- Reiterate until you have reached an answer which is close enough (or no improvements have been made in a while)

Hypothesis

Potential benefits of dynamic Ansatz:

1. faster
2. not as prone to barren plateaus
3. better at handling noise

Loss functions

Goal

Minimize loss function to find $|x\rangle = |V(\alpha)\rangle$ such that $A|x\rangle = |b\rangle$

Global Loss Function

$$C_G = 1 - |\langle b|\Psi\rangle|^2 \qquad \Psi = \frac{A|V(\alpha)\rangle}{\sqrt{\langle V(\alpha)|A^\dagger A|V(\alpha)\rangle}} \tag{1}$$

- Evaluates the overlap with the full state → requires full state measurement.
- Defines the **global** minimum, but has more barren plateaus.

Loss functions

Local Loss Function

$$C_L = \langle V(\boldsymbol{\alpha})| \, H_L \, |V(\boldsymbol{\alpha})\rangle \qquad H_L = A^\dagger U \left(\mathbb{I} - \frac{1}{n} \sum_{j=1}^{n} |0_j\rangle\langle 0_j| \otimes \mathbb{I}_{\bar{j}} \right) U^\dagger A \quad (2)$$

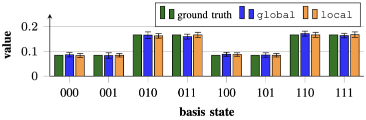
$|b\rangle = U\,|0\rangle$
$|0_j\rangle$ is the zero state on qubit j
$\mathbb{I}_{\bar{j}}$ is the identity on all qubits except j

- Requires subsystem measurements, as defined by the local Hamiltonian → no need for full state measurement.
- Can end in **local** minima, but has less barren plateaus as it is operating on subspaces.

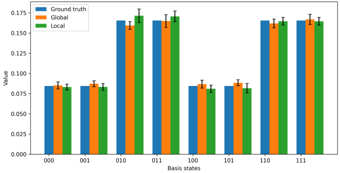
2. Methods

Recreating results from paper

The code from the paper we used could be found on GitHub
We performed the same example problem as described in the text, and got similar results



(a) Results from paper



(b) Recreated results

Figure: Results from paper and our execution of code.

Dynamic vs static ansatz

We implemented the static from Bravo-Prieto et al. 2023 to test againt the dynamic ansatz:

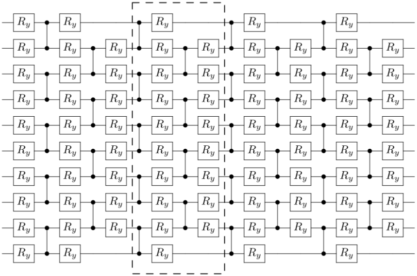


Figure: Fixed layer ansatz from [3]

[3] C. Bravo-Prieto, R. LaRose, M. Cerezo, Y. Subasi, L. Cincio, and P. J. Coles (Nov. 2023). "Variational Quantum Linear Solver". In: vol. 7. Verein zur Förderung des Open Access Publizierens in den Quantenwissenschaften, p. 1188. DOI: 10.22331/q-2023-11-22-1188

The problems/matrices

$$\mathbf{A}\bar{x} = \bar{b}$$

The linear equation solvers will be tested on the following matrices **A**:

- 2D Poisson matrix
- Sparse matrices with different condition numbers
- Dense matrices with different condition numbers

We study condition number following the test of Patil et al.[4].
We will be assessing performance under:

- Static vs dynamic Ansatz
- Noiseless vs noisy environment
- Global vs local loss function

For simplicity, we will let $\bar{b} = \frac{1}{\sqrt{2^{qubits}}}\begin{pmatrix} 1 & 1 & \dots & 1 \end{pmatrix}$ throughout the tests.

[4] H. Patil, Y. Wang, and P. S. Krstić (2022). "Variational quantum linear solver with a dynamic ansatz". In: *Phys. Rev. A* 105 (1), p. 012423. DOI: 10.1103/PhysRevA.105.012423

Sparse matrices – condition number tests

To test the VQLS on matrices with various conditions numbers, we implemented the "matrix generator" from Bravo-Prieto et al. 2023 [5]:

$$A = \frac{1}{\xi}\left(\sum_{j=1}^{n} X_j + J\sum_{j=1}^{n-1} Z_j Z_{j+1} + \eta\mathbb{I}\right) \tag{3}$$

$$b = H^{\otimes n}\left|0\right\rangle \qquad \lambda_{min} = \frac{1}{\kappa} \qquad \lambda_{max} = 1 \tag{4}$$

here κ is the condition number.

[5] C. Bravo-Prieto, R. LaRose, M. Cerezo, Y. Subasi, L. Cincio, and P. J. Coles (Nov. 2023). "Variational Quantum Linear Solver". In: vol. 7. Verein zur Förderung des Open Access Publizierens in den Quantenwissenschaften, p. 1188. DOI: 10.22331/q-2023-11-22-1188

Dense matrices – condition number test

Generally, matrices $\mathbf{A}$ to solve $\mathbf{A}\bar{x} = \bar{b}$ are sparse.
However, we want to stress test the system and will thus use complex dense matrices.They are generated as follows.

Generating dense matrices with condition number c

1. Generate complex random matrix $\mathbf{A}$
2. Decompose $\mathbf{A} = U\Sigma V^T$ and create $\tilde{\Sigma}$ diagonal matrix with ordered random values from σ_1/c to σ_1 (where $\sigma_1 = \Sigma_{11}$ largest singular value)
3. Recompose $\tilde{\mathbf{A}} = U\tilde{\Sigma}V^T$

$\Rightarrow \kappa(\mathbf{A}) = \frac{\max\sigma}{\min\sigma} = \frac{\sigma_1}{\sigma_1/c} = c$

Noise simulation

We use Qiskit AER as a quantum simulator [6].
To simulate the noise, we use qiskit's GenericBackendV2 [7].
We were not able to test the code on IBM's real quantum hardware, as it now requires credit card information.

[6] Qiskit Development Team (2025). *Qiskit Aer documentation*. Url: https://qiskit.github.io/qiskit-aer/

[7] *GenericBackendV2* (n.d.). Url: https://docs.quantum.ibm.com/api/qiskit/qiskit.providers.fake_provider.GenericBackendV2

TRC

We cannot use time as a metric, since we are performing the tests on a quantum simulator.
Following Patil et al.[8], we use the *Total Resource Cost* (TRC).

TRC for Dynamic and Static Ansätze

Dynamic Ansatz:

$$TRC_{ADA} := \sum_{i=0}^{z_{ADA}} d_{ADA}(i)$$

Static Ansatz:

$$TRC_{ASA} := z_{ASA} d_{ASA}$$

z_{ADA}: Total amount of iterations (with dynamic Ansatz)
z_{ASA}: Total amount of iterations (with static Ansatz)
$d_{ADA}(i)$: Number of layers in iteration i
d_{ASA}: Predefined number of layers in static Ansatz

[8] H. Patil, Y. Wang, and P. S. Krstić (2022). "Variational quantum linear solver with a dynamic ansatz". In: *Phys. Rev. A* 105 (1), p. 012423. DOI: 10.1103/PhysRevA.105.012423

3. Results

Results - Sparse - Condition number test

Sparse matrix simulations:

- 3 qubits
- 10 runs per c
- Test different c between $[1.04, 158]$
- Local loss only

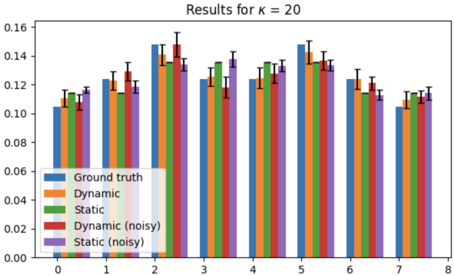
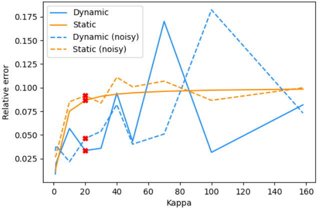
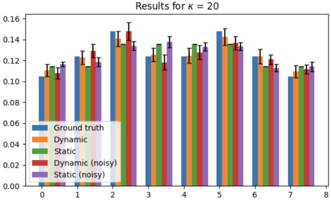


Figure: Components of $|x\rangle$ averaged over 10 runs for $c = 20$ with local loss.

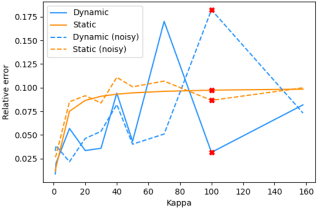
Results - Sparse - Condition number test



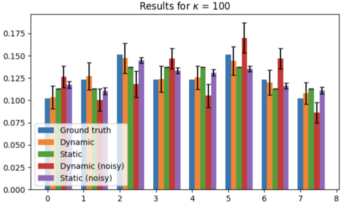
(a) Relative error



(b) Components



(c) Relative error

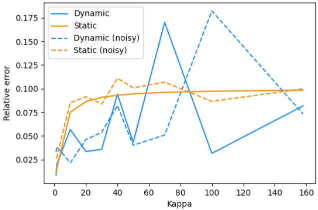


(d) Components

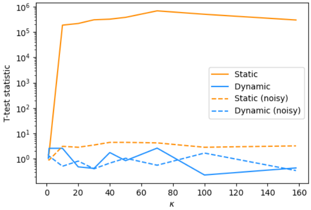
Results - Sparse - Condition number test

T-test statistic - test whether sample could have come from a population with μ_0

$$t = \frac{\bar{x} - \mu_0}{\sigma_{std}/\sqrt{N}} \qquad \begin{aligned} \bar{x} &: \text{Sample mean} \\ \mu_0 &: \text{True mean} \end{aligned} \tag{5}$$



(a) Relative error



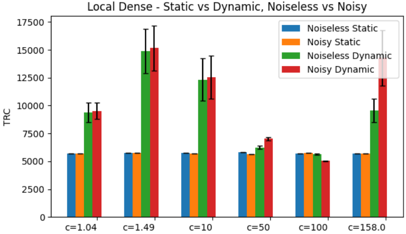
(b) T-test

Results - Sparse - Condition number test

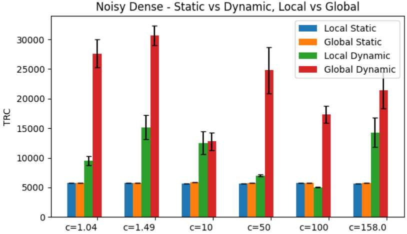
Table: P-values - the probability of observing results as extreme as, or more extreme, assuming $\mu_0 = \mu_{samp}$.

κ	Static	Static (noisy)	Dynamic	Dynamic (noisy)
1.04	0.04	0.4	0.2	0.2
1.49	0.009	0.4	10^{-6}	0.06
10	10^{-44}	10^{-8}	10^{-6}	0.94
20	10^{-44}	10^{-7}	0.96	0.59
30	10^{-46}	10^{-10}	0.98	0.99
40	10^{-46}	10^{-14}	0.003	0.81
50	10^{-46}	10^{-14}	0.6	0.24
70	10^{-50}	10^{-13}	10^{-6}	0.89
100	10^{-48}	10^{-07}	0.99	0.007
158	10^{-47}	10^{-09}	0.98	0.99

Results – TRC for Dense matrix



(c) Noisy vs. Noiseless



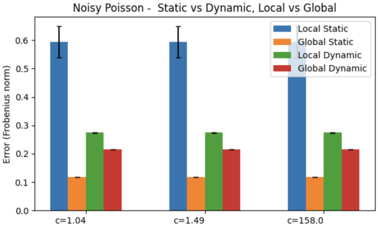
(d) Local vs. Global

Figure: Dense matrix TRC

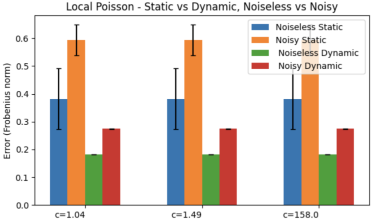
- As expected, static performs the same almost always
- Static lower – only one epoch is ran, same stopping criterion
- Noisy and global take longer
- For $c \in [50, 100]$, local dynamic is ran as many times as static

TRC for Poisson follows the same pattern.

Results – 2D Poisson matrix errors



(a) Local vs. Global



(b) Noisy vs. Noiseless

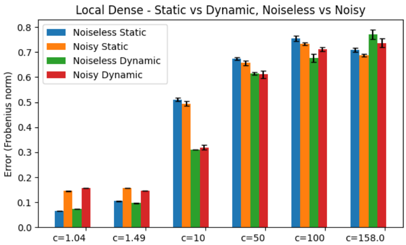
Figure: Poisson matrix errors

(a) Local performs much worse than global – especially static
- Prone to getting stuck at local minima, does not happen as fast for dynamic due to coarseness (explaining variance)
- Global static performs best

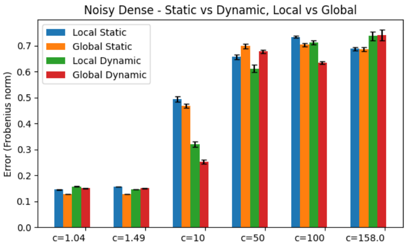
(b) Adding noise increases error

(b) Dynamic performs better than static – again, dynamic seems to helps coarsen out local minima (explaining variance)

Results – dense matrix errors



(a) Noisy vs. Noiseless



(b) Local vs. Global

Figure: Dense matrix errors

- Error gets larger for larger c

(a) Dynamic seems to be performing better for $c \in [10, 100]$

(a) Noise does not seem to play a big role for $c > 10$

(b) Dynamic circuits seem to avoid local minima, but only for $c = 10$ (perhaps the sweet spot between a rugged landscape and barren plateaus)

(b) Dynamic circuits seem to have higher variance on average (61%)

4. Conclusion

Conclusion

- In general we are able to estimate $|V(\alpha))\rangle$.
- In the case of sparse matrices, the static ansatz produces very misleading results for increasing condition numbers. Here, the dynamic ansatz outperforms the static when taking the uncertainty into account.
- The dynamic ansatz seems to help ignoring shallow minima and noise when using the local loss function. Overall, we see global loss outperforming local loss, which might be due to the small number of qubits.
- There does not seem to be a linear trend with the condition number and the error/TRC using a dynamic ansatz.

5. Further research

Further research

Several points could be expanded upon.

1. Due to time constraints, we only used 3-4 qubits
   - In the future, could see the effects of qubits on our results
   - This is particularly relevant as usually the matrices are solved for more than 3-4 qubits
2. We made relatively few iterations
   - This can lead to skewed results
   - Especially problematic since we seem to have high variance
3. Could have used more types of matrices
4. Test problem specific local loss functions to match Pauli string decomposition of A.
5. Test global vs local for increasing qubits (barren plateaus for large dimensions)
6. Use more condition numbers c to get a better picture
7. Better parameters for static and dynamic Ansätze

References I

Bravo-Prieto, C., R. LaRose, M. Cerezo, Y. Subasi, L. Cincio, and P. J. Coles (Nov. 2023). "Variational Quantum Linear Solver". In: vol. 7. Verein zur Förderung des Open Access Publizierens in den Quantenwissenschaften, p. 1188. DOI: 10.22331/q-2023-11-22-1188.

GenericBackendV2 (n.d.). Url: https://docs.quantum.ibm.com/api/qiskit/qiskit.providers.fake_provider.GenericBackendV2.

Meyer, N., M. Röhn, J. Murauer, A. Plinge, C. Mutschler, and D. D. Scherer (Sept. 2024). "Comprehensive Library of Variational LSE Solvers". In: *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)*. IEEE, 1–4. DOI: 10.1109/qce60285.2024.10242.

Patil, H., Y. Wang, and P. S. Krstić (2022). "Variational quantum linear solver with a dynamic ansatz". In: *Phys. Rev. A* 105 (1), p. 012423. DOI: 10.1103/PhysRevA.105.012423.

Qiskit Development Team (2025). *Qiskit Aer documentation*. Url: https://qiskit.github.io/qiskit-aer/.