Quantum Singular Value Transformation on NISQ Hardware

Yale Quantum Institute Visiting Student Researcher Program

Aryan Bawa

Advisors: Steven Girvin, Yongshan Ding

July 31, 2026

Outline

1. Introduction
   - Project Goals
   - What is the Quantum Singular Value Transformation?
   - What Are Noisy Intermediate Quantum (NISQ) Devices?
   - Quantum Circuit Basics
2. Constructing the QSVT
   - Quantum Signal Processing
   - Block Encoding and Linear Combination of Unitaries
   - Putting It All Together
3. QSVT on Hardware
   - Implementing the QSVT on NISQ Hardware
   - Further Considerations

Project Goals

Project Goals

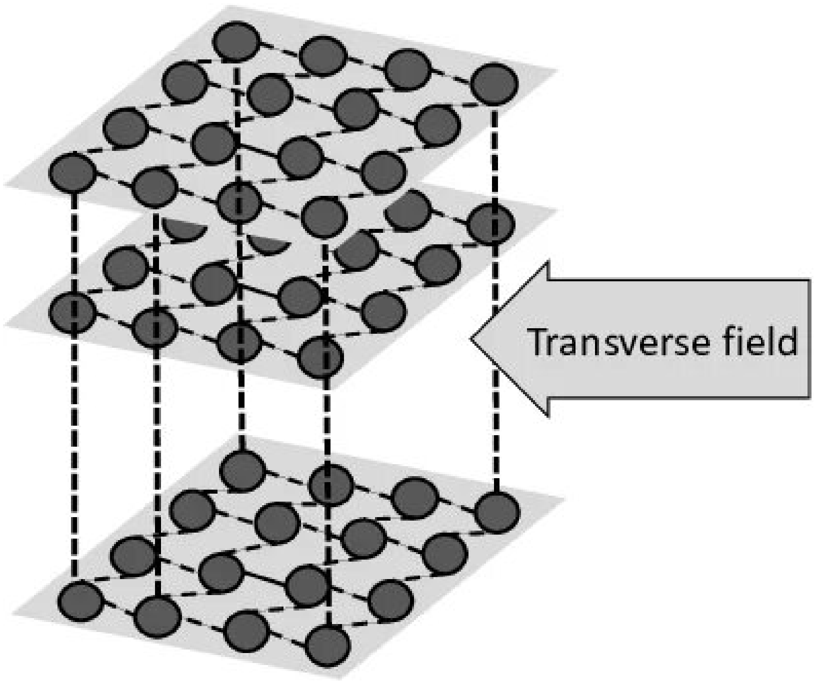
The aim of my project is to simulate quantum systems in time on current quantum hardware, using a technique known as the quantum singular value transformation.

- Much simplification is required to make quantum algorithms run on current hardware

Project Goals

The aim of my project is to simulate quantum systems in time on current quantum hardware, using a technique known as the quantum singular value transformation.

- Much simplification is required to make quantum algorithms run on current hardware



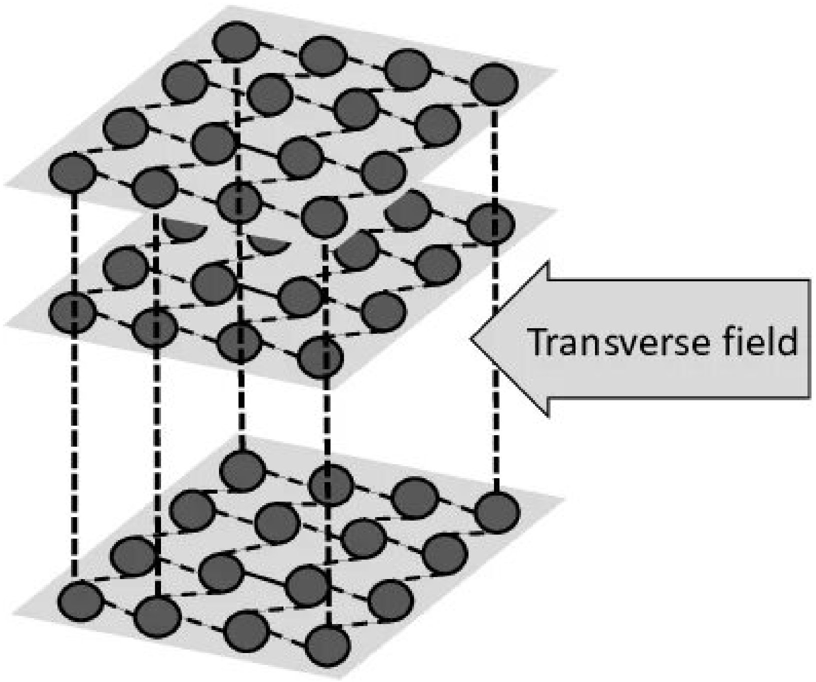
The Mixed Field Ising Hamiltonian:

$$H = -J\sum_{i=0}^{n-2} Z_i Z_{i+1} - \sum_{i=0}^{n-1} h_i X_i - m\sum_{i=0}^{n-1} Z_i$$

Project Goals

The aim of my project is to simulate quantum systems in time on current quantum hardware, using a technique known as the quantum singular value transformation.

- Much simplification is required to make quantum algorithms run on current hardware



The Mixed Field Ising Hamiltonian:

$$H = -J \sum_{i=0}^{n-2} Z_i Z_{i+1} - \sum_{i=0}^{n-1} h_i X_i - m \sum_{i=0}^{n-1} Z_i$$

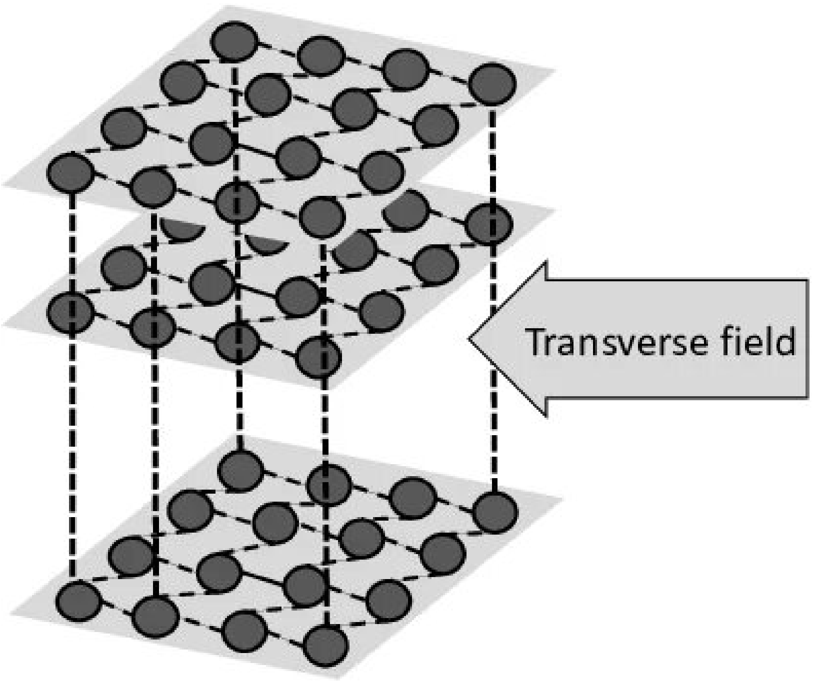
For $n = 3$:

$$H = -J Z_0 Z_1 - J Z_1 Z_2 - h_0 X_0 - h_1 X_1 - h_2 X_2$$
$$-m(Z_0 + Z_1 + Z_2)$$

Project Goals

The aim of my project is to simulate quantum systems in time on current quantum hardware, using a technique known as the quantum singular value transformation.

- Much simplification is required to make quantum algorithms run on current hardware



The Mixed Field Ising Hamiltonian:

$$H = -J \sum_{i=0}^{n-2} Z_i Z_{i+1} - \sum_{i=0}^{n-1} h_i X_i - m \sum_{i=0}^{n-1} Z_i$$

For $n = 3$:

$$H = -JZ_0Z_1 - JZ_1Z_2 - h_0X_0 - h_1X_1 - h_2X_2 \\ -m(Z_0 + Z_1 + Z_2)$$

We consider the $n = 3$ Ising spin chain with $h_i/J = -1.05$ for all i and $m/J = 0.5$ in our baseline Hamiltonian simulation.

What is the Quantum Singular Value Transformation (QSVT)?

What is the Quantum Singular Value Transformation (QSVT)?

Hamiltonian Simulation: A central problem of quantum computing is solving for the time dynamics of a quantum system. Given the Hamiltonian H of the system, this is equivalent to solving for the propagator e^{-iHt}.

What is the Quantum Singular Value Transformation (QSVT)?

Hamiltonian Simulation: A central problem of quantum computing is solving for the time dynamics of a quantum system. Given the Hamiltonian H of the system, this is equivalent to solving for the propagator e^{-iHt}.

To implement this propagator, we need a a transform of the operator H.

$$f(H) \text{ such that } f(x) = e^{-ixt}$$

By the spectral theorem, a function of an operator is a function of its eigenvalues.

What is the Quantum Singular Value Transformation (QSVT)?

Hamiltonian Simulation: A central problem of quantum computing is solving for the time dynamics of a quantum system. Given the Hamiltonian H of the system, this is equivalent to solving for the propagator e^{-iHt}.

To implement this propagator, we need a a transform of the operator H.

$$f(H) \text{ such that } f(x) = e^{-ixt}$$

By the spectral theorem, a function of an operator is a function of its eigenvalues.

Quantum Singular Value Transformation (In Words)

The **quantum singular value transformation (QSVT)** is a technique that allows one to polynomially transform the singular values of a given matrix.

- For square matrices, the singular values are equal to the eigenvalues.
- Importantly, the QSVT has been shown to be an important primitive for many other quantum algorithms.
- **Three components:** Quantum Signal Processing, Block Encoding, Upscaling (Qubitization)

What are Noisy Intermediate Quantum (NISQ) Devices?

What are Noisy Intermediate Quantum (NISQ) Devices?

Quantum computation has the potential to revolutionize many fields, from medicine to cryptography.

What are Noisy Intermediate Quantum (NISQ) Devices?

Quantum computation has the potential to revolutionize many fields, from medicine to cryptography.

However: The quantum states required to carry out computations are very sensitive, both to hardware defects and interactions from the environment. Small errors can build up during computations, which can cause programs to fail.

What are Noisy Intermediate Quantum (NISQ) Devices?

Quantum computation has the potential to revolutionize many fields, from medicine to cryptography.

However: The quantum states required to carry out computations are very sensitive, both to hardware defects and interactions from the environment. Small errors can build up during computations, which can cause programs to fail.

Two big limitations with current hardware:

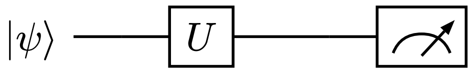
- Deeper circuits decohere states
- Cannot correct our errors at scale

Quantum Circuit Basics

Quantum Circuit Basics

Quantum gates are modeled by unitary operations. This is required by quantum mechanics.

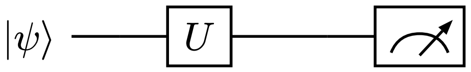
$$UU^\dagger = U^\dagger U = \mathbb{1} \text{ (identity)}$$



Quantum Circuit Basics

Quantum gates are modeled by unitary operations. This is required by quantum mechanics.

$$UU^\dagger = U^\dagger U = \mathbb{1} \text{ (identity)}$$



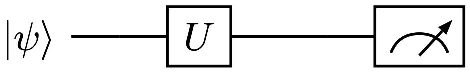
The above circuit is equal to a measurement of $U|\psi\rangle$. An arbitrary state is represented like so:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

Quantum Circuit Basics

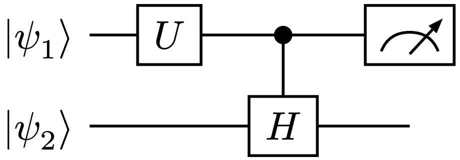
Quantum gates are modeled by unitary operations. This is required by quantum mechanics.

$$UU^\dagger = U^\dagger U = \mathbb{1} \text{ (identity)}$$



The above circuit is equal to a measurement of $U|\psi\rangle$. An arbitrary state is represented like so:

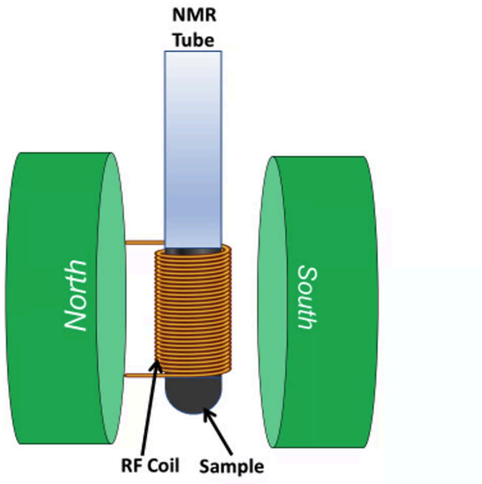
$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$



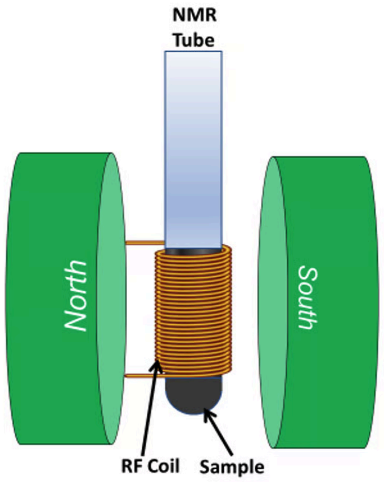
$$|\psi_1\rangle \otimes |\psi_2\rangle$$

Quantum Signal Processing

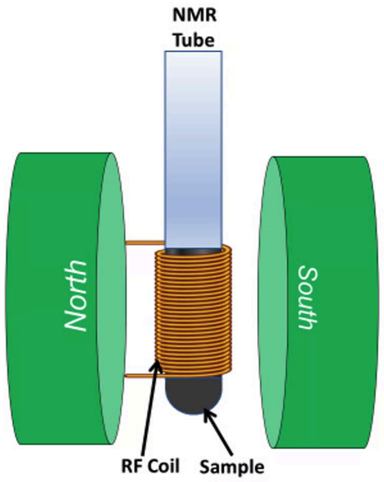
Quantum Signal Processing



Quantum Signal Processing

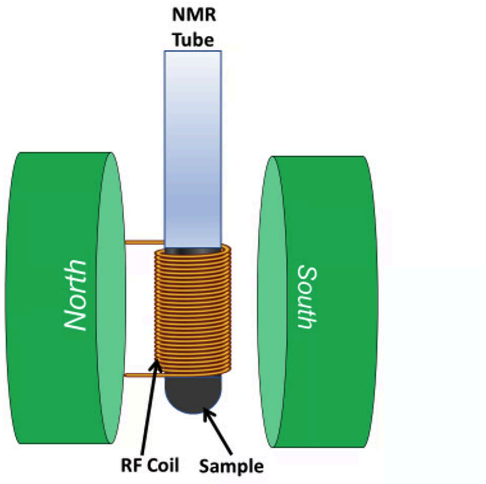


Quantum Signal Processing



$$|\psi\rangle \;—\; \boxed{e^{i\varphi_2 Z}} \;—\; \boxed{W(a)} \;—\; \boxed{e^{i\varphi_1 Z}} \;—\; \boxed{W(a)} \;—\; \boxed{e^{i\varphi_0 Z}}$$

Quantum Signal Processing



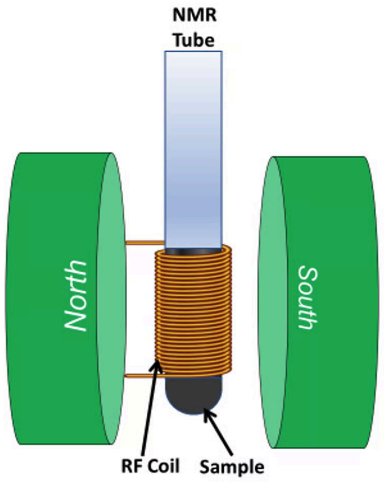
$$|\psi\rangle \; - \boxed{e^{i\varphi_2 Z}} - \boxed{W(a)} - \boxed{e^{i\varphi_1 Z}} - \boxed{W(a)} - \boxed{e^{i\varphi_0 Z}}$$

Quantum Signal Processing (QSP)

Given a rotation matrix parameterized by a and a rotation matrix parameterized by a set of angles $\{\varphi_k\}$, we can interleave these operators to create a matrix $U_{\vec{\varphi}}$ that is a polynomial of a in the top left corner:

$$e^{i\varphi_0 Z}W(a)e^{i\varphi_1 Z}W(a)e^{i\varphi_2 Z}...W(a)e^{i\varphi_k Z} = \begin{pmatrix} P(a) & \cdot \\ \cdot & \cdot \end{pmatrix}$$

Quantum Signal Processing



$$e^{-iHt} = \cos(Ht) - i\sin(Ht)$$

$$\cos(Ht) \approx \mathbb{1} - \frac{(Ht)^2}{2} + \frac{(Ht)^4}{24}$$

$$\sin(Ht) \approx Ht - \frac{(Ht)^3}{6} + \frac{(Ht)^5}{120}$$

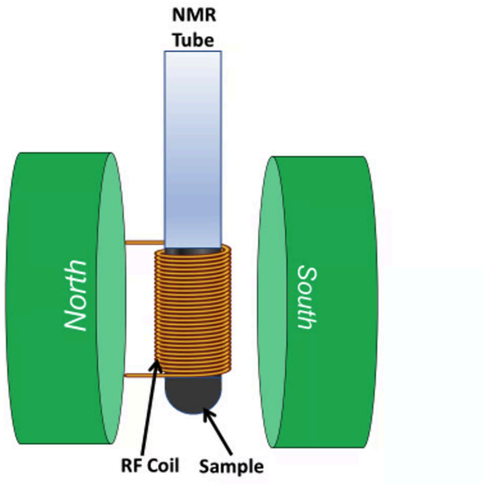
$$|\psi\rangle \;—\; \boxed{e^{i\varphi_2 Z}} \;—\; \boxed{W(a)} \;—\; \boxed{e^{i\varphi_1 Z}} \;—\; \boxed{W(a)} \;—\; \boxed{e^{i\varphi_0 Z}}$$

Quantum Signal Processing (QSP)

Given a rotation matrix parameterized by a and a rotation matrix parameterized by a set of angles $\{\varphi_k\}$, we can interleave these operators to create a matrix $U_{\vec{\varphi}}$ that is a polynomial of a in the top left corner:

$$e^{i\varphi_0 Z} W(a) e^{i\varphi_1 Z} W(a) e^{i\varphi_2 Z} \ldots W(a) e^{i\varphi_k Z} = \begin{pmatrix} P(a) & \cdot \\ \cdot & \cdot \end{pmatrix}$$

Quantum Signal Processing



$$e^{-iHt} = \cos(Ht) - i\sin(Ht)$$

$$\cos(Ht) \approx \mathbb{1} - \frac{(Ht)^2}{2} + \frac{(Ht)^4}{24}$$

$$\sin(Ht) \approx Ht - \frac{(Ht)^3}{6} + \frac{(Ht)^5}{120}$$

$$|\psi\rangle \;—\; \boxed{e^{i\varphi_2 Z}} \;—\; \boxed{W(a)} \;—\; \boxed{e^{i\varphi_1 Z}} \;—\; \boxed{W(a)} \;—\; \boxed{e^{i\varphi_0 Z}}$$

Quantum Signal Processing (QSP)

Given a signal rotation operator $W(a) = \begin{pmatrix} a & i\sqrt{1-a^2} \\ i\sqrt{1-a^2} & a \end{pmatrix}$ and a signal processing rotation operator $e^{i\varphi Z}$, we may interleave these operators to create a matrix $U_{\vec{\varphi}}$ that can be expressed as a polynomial function of a (given various restrictions on the polynomials):

$$e^{i\varphi_0 Z} \prod_{k=1}^{d} W(a) e^{i\varphi_k Z} = \begin{pmatrix} P(a) & iQ(a)\sqrt{1-a^2} \\ iQ^*(a)\sqrt{1-a^2} & P^*(a) \end{pmatrix}$$

Block Encoding and Linear Combination of Unitaries

Block Encoding and Linear Combination of Unitaries

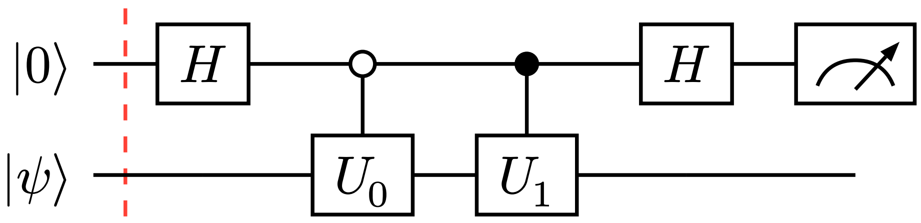
Block encoding is a method to encode any matrix A in the top left corner of a unitary. A standard construction uses a linear combination of unitaries (LCU):

$$\begin{pmatrix} A & \cdot \\ \cdot & \cdot \end{pmatrix}$$

Block Encoding and Linear Combination of Unitaries

Block encoding is a method to encode any matrix A in the top left corner of a unitary. A standard construction uses a linear combination of unitaries (LCU):

$$\begin{pmatrix} A & \cdot \\ \cdot & \cdot \end{pmatrix}$$

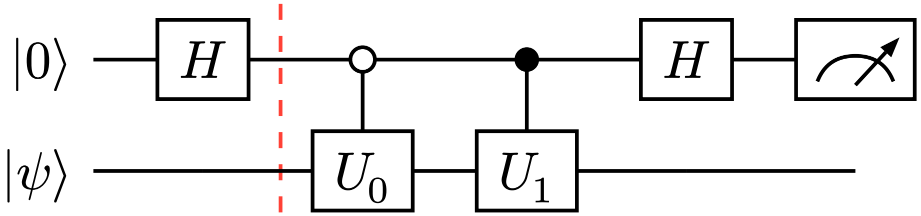


$$U = \begin{pmatrix} \mathbb{1} & 0 \\ 0 & \mathbb{1} \end{pmatrix} \qquad |0\rangle \otimes |\psi\rangle$$

Block Encoding and Linear Combination of Unitaries

Block encoding is a method to encode any matrix A in the top left corner of a unitary. A standard construction uses a linear combination of unitaries (LCU):

$$\begin{pmatrix} A & \cdot \\ \cdot & \cdot \end{pmatrix}$$



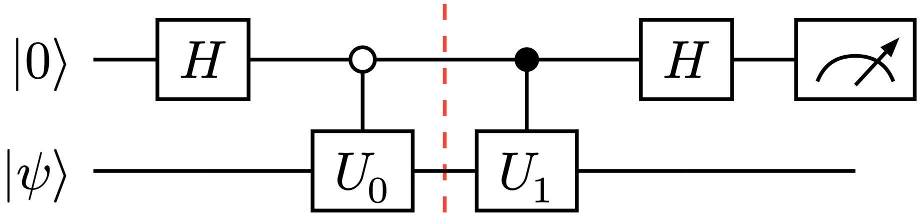
$$U = \frac{1}{\sqrt{2}} \begin{pmatrix} \mathbb{1} & \mathbb{1} \\ \mathbb{1} & -\mathbb{1} \end{pmatrix}$$

$$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |\psi\rangle$$

Block Encoding and Linear Combination of Unitaries

Block encoding is a method to encode any matrix A in the top left corner of a unitary. A standard construction uses a linear combination of unitaries (LCU):

$$\begin{pmatrix} A & \cdot \\ \cdot & \cdot \end{pmatrix}$$

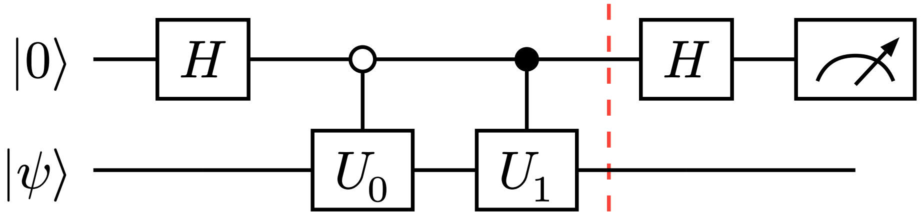


$$U = \frac{1}{\sqrt{2}}\begin{pmatrix} U_0 & U_0 \\ \mathbb{1} & -\mathbb{1} \end{pmatrix} \qquad \frac{1}{\sqrt{2}}(|0\rangle \otimes U_0|\psi\rangle + |1\rangle \otimes |\psi\rangle)$$

Block Encoding and Linear Combination of Unitaries

Block encoding is a method to encode any matrix A in the top left corner of a unitary. A standard construction uses a linear combination of unitaries (LCU):

$$\begin{pmatrix} A & \cdot \\ \cdot & \cdot \end{pmatrix}$$



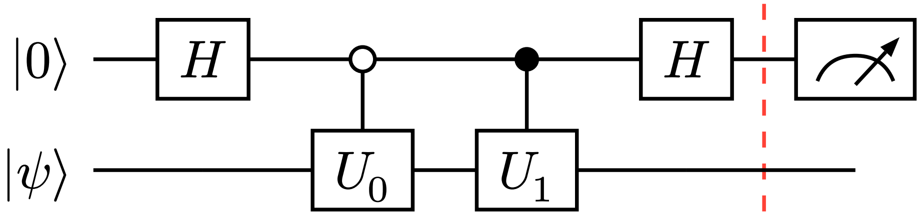
$$U = \frac{1}{\sqrt{2}} \begin{pmatrix} U_0 & U_0 \\ U_1 & -U_1 \end{pmatrix}$$

$$\frac{1}{\sqrt{2}} (|0\rangle \otimes U_0 |\psi\rangle + |1\rangle \otimes U_1 |\psi\rangle)$$

Block Encoding and Linear Combination of Unitaries

Block encoding is a method to encode any matrix A in the top left corner of a unitary. A standard construction uses a linear combination of unitaries (LCU):

$$\begin{pmatrix} A & \cdot \\ \cdot & \cdot \end{pmatrix}$$



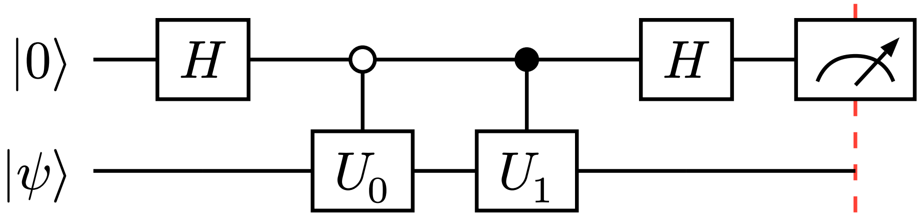
$$U = \frac{1}{2}\begin{pmatrix} U_0 + U_1 & U_0 - U_1 \\ U_0 - U_1 & U_0 + U_1 \end{pmatrix}$$

$$\frac{1}{2}(|0\rangle \otimes (U_0 + U_1)|\psi\rangle + |1\rangle \otimes (U_0 - U_1)|\psi\rangle)$$

Block Encoding and Linear Combination of Unitaries

Block encoding is a method to encode any matrix A in the top left corner of a unitary. A standard construction uses a linear combination of unitaries (LCU):

$$\begin{pmatrix} A & \cdot \\ \cdot & \cdot \end{pmatrix}$$



$$U = \begin{pmatrix} \frac{1}{2}(U_0 + U_1) & \cdot \\ \cdot & \cdot \end{pmatrix}$$

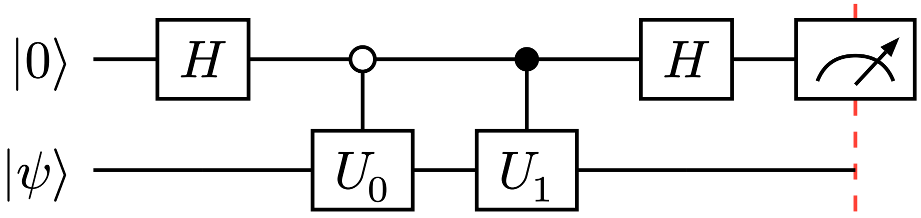
$$\frac{1}{2}(U_0 + U_1)|\psi\rangle$$

postselect $|0\rangle$

Block Encoding and Linear Combination of Unitaries

Block encoding is a method to encode any matrix A in the top left corner of a unitary. A standard construction uses a linear combination of unitaries (LCU):

$$\begin{pmatrix} A & \cdot \\ \cdot & \cdot \end{pmatrix}$$



$$U = \begin{pmatrix} \frac{1}{2}(U_0 + U_1) & \cdot \\ \cdot & \cdot \end{pmatrix} \qquad \frac{1}{2}(U_0 + U_1)|\psi\rangle$$

postselect $|0\rangle$

We can write any operator A as a sum of basis unitaries $\{P_i\}_{i=0,\dots,n}$ — $A = \sum_i^n \alpha_i P_i$. As such, **Any matrix can be block encoded like this.**

Block Encoding and Linear Combination of Unitaries

Block encoding is a method to encode any matrix A in the top left corner of a unitary. A standard construction uses a linear combination of unitaries (LCU):

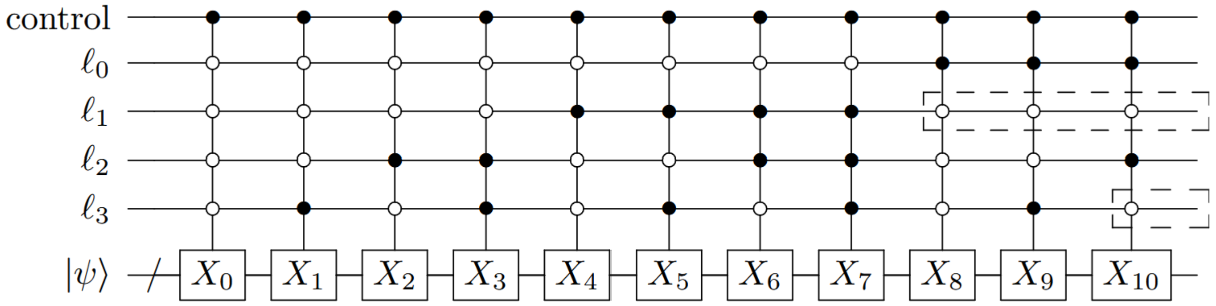
$$\begin{pmatrix} A & \cdot \\ \cdot & \cdot \end{pmatrix}$$

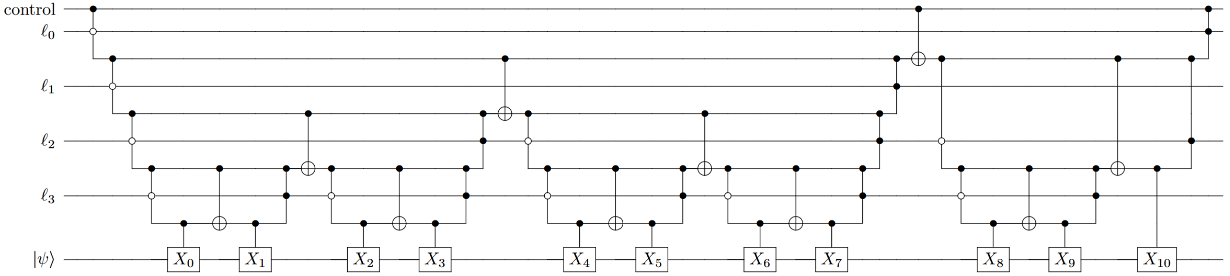
$$U = \begin{pmatrix} \frac{1}{2}(U_0 + U_1) & \cdot \\ \cdot & \cdot \end{pmatrix} \qquad \frac{1}{2}(U_0 + U_1)|\psi\rangle$$

postselect $|0\rangle$

We can write any operator A as a sum of basis unitaries $\{P_i\}_{i=0,\dots,n}$ — $A = \sum_i^n \alpha_i P_i$. As such, **Any matrix can be block encoded like this.**

Block Encoding and Linear Combination of Unitaries





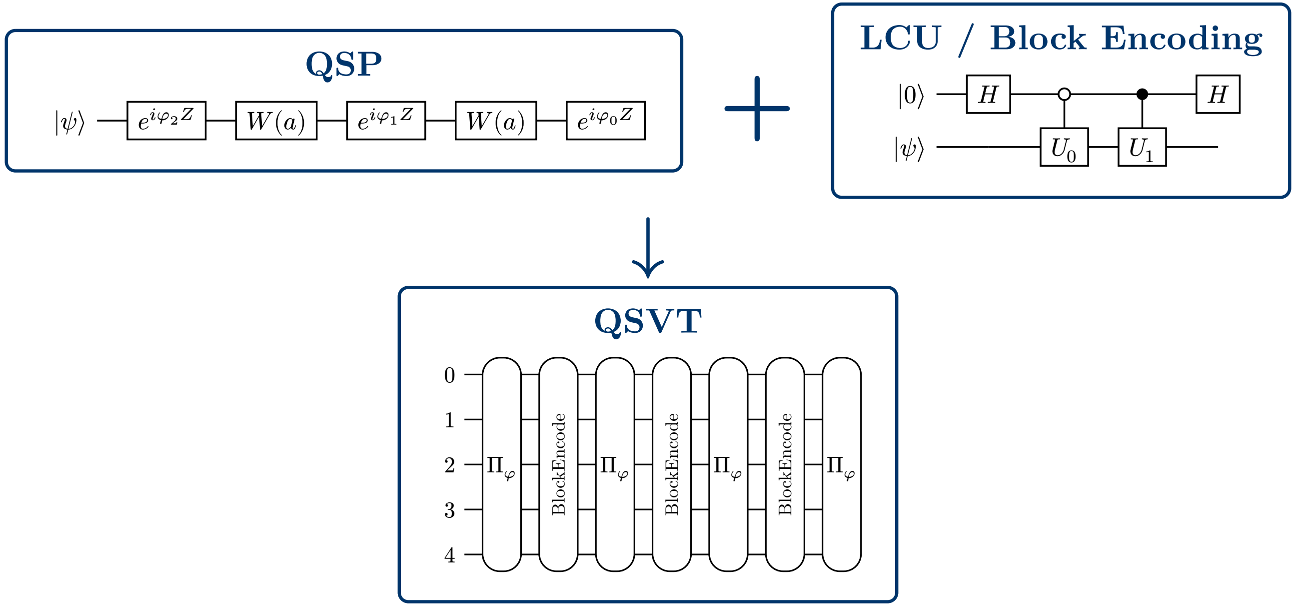
Putting it Together — The QSVT

Putting it Together — The QSVT

We can combine the earlier ideas by scaling QSP beyond single qubits using techniques known as qubitization or the cosine-sine decomposition, and considering a block encoded signal rotation operator.

Putting it Together — The QSVT

We can combine the earlier ideas by scaling QSP beyond single qubits using techniques known as qubitization or the cosine-sine decomposition, and considering a block encoded signal rotation operator.



Putting it Together — The QSVT

We can combine the earlier ideas by scaling QSP beyond single qubits using techniques known as qubitization or the cosine-sine decomposition, and considering a block encoded signal rotation operator.

Formal QSVT (Odd Parity)

Given a unitary U that block-encodes a matrix $A = \tilde{\Pi} U \Pi$ via projectors Π and $\tilde{\Pi}$, and projector-controlled phase shifts

$$\Pi_\varphi := e^{i\varphi(2\Pi - \mathbb{1})}, \quad \tilde{\Pi}_\varphi := e^{i\varphi(2\tilde{\Pi} - \mathbb{1})},$$

for odd degree d and phases $\vec{\varphi} = (\varphi_1, \ldots, \varphi_d)$ the interleaved sequence

$$U_{\vec{\varphi}} = \tilde{\Pi}_{\varphi_1} U \prod_{k=1}^{\frac{d-1}{2}} \left(\Pi_{\varphi_{2k}} U^\dagger \tilde{\Pi}_{\varphi_{2k+1}} U \right)$$

implements an odd polynomial transform of the singular values of A:

$$\tilde{\Pi} U_{\vec{\varphi}} \Pi = P^{(SV)}(A).$$

Implementing the QSVT on NISQ Hardware

Implementing the QSVT on NISQ Hardware

Unoptimized resource estimation:

```
gateset = {"X", "Z", "S", "T", "Hadamard", "CNOT"}
  --- Resources: ---
  Total wires: 14
    algorithmic wires: 8
    allocated wires: 6
      zero state: 6
      any state: 0
  Total gates : 6.562E+4
    'T': 3.665E+4,
    'CNOT': 1.289E+4,
    'X': 236,
    'Z': 3.800E+3,
    'S': 4.807E+3,
    'Hadamard': 7.244E+3
```

Implementing the QSVT on NISQ Hardware

Unoptimized resource estimation:

```
gateset = {"X", "Z", "S", "T", "Hadamard", "CNOT"}
  --- Resources: ---
  Total wires: 14
    algorithmic wires: 8
    allocated wires: 6
      zero state: 6
      any state: 0
  Total gates : 6.562E+4
    'T': 3.665E+4,
    'CNOT': 1.289E+4,
    'X': 236,
    'Z': 3.800E+3,
    'S': 4.807E+3,
    'Hadamard': 7.244E+3
```

Optimized Resource Estimation

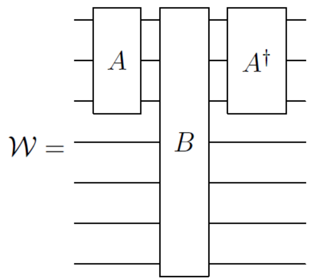
Not finished yet... Current progress — more efficient block encoding via multiplexor compilation of SELECT gates.

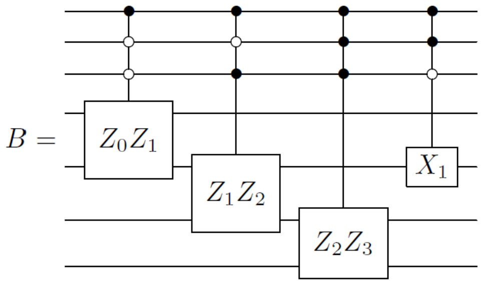
Unoptimized resource estimation:

```
gateset = {"X", "Z", "S", "T", "Hadamard", "CNOT"}
  --- Resources: ---
  Total wires: 14
    algorithmic wires: 8
    allocated wires: 6
      zero state: 6
      any state: 0
  Total gates : 6.562E+4
    'T': 3.665E+4,
    'CNOT': 1.289E+4,
    'X': 236,
    'Z': 3.800E+3,
    'S': 4.807E+3,
    'Hadamard': 7.244E+3
```

Optimized Resource Estimation

Not finished yet... Current progress — more efficient block encoding via multiplexor compilation of SELECT gates.



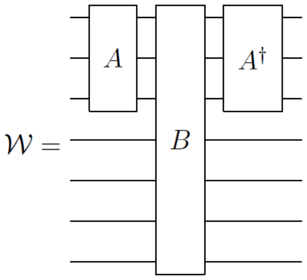


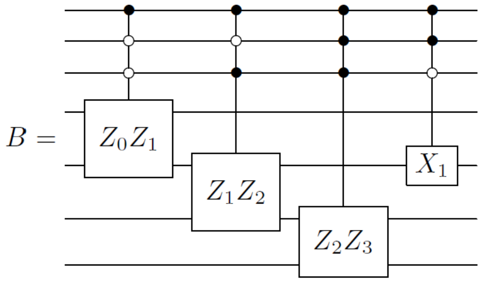
Unoptimized resource estimation:

```
gateset = {"X", "Z", "S", "T", "Hadamard", "CNOT"}
  --- Resources: ---
  Total wires: 14
    algorithmic wires: 8
    allocated wires: 6
      zero state: 6
      any state: 0
  Total gates : 6.562E+4
    'T': 3.665E+4,
    'CNOT': 1.289E+4,
    'X': 236,
    'Z': 3.800E+3,
    'S': 4.807E+3,
    'Hadamard': 7.244E+3
```

Optimized Resource Estimation

Not finished yet... Current progress — more efficient block encoding via multiplexor compilation of SELECT gates.





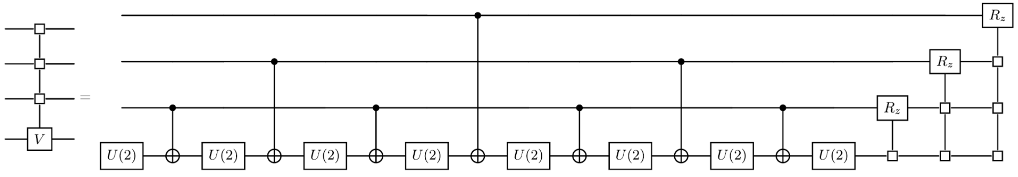


Figure 3. Decomposition for $M_n^{\mathcal{C}}(V)$ where V is a list of U(2) unitaries, $n = 1$, and $\mathcal{C} = 3$. The open squares indicate multiplexed controls. (See figures 2 and 9).

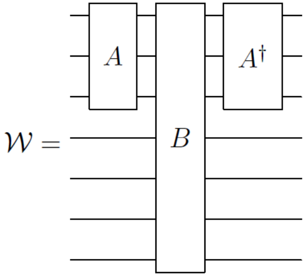
Implementing the QSVT on NISQ Hardware

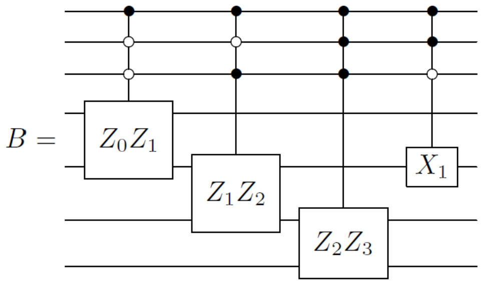
Unoptimized resource estimation:

```
gateset = {"X", "Z", "S", "T", "Hadamard", "CNOT"}
  --- Resources: ---
  Total wires: 14
    algorithmic wires: 8
    allocated wires: 6
      zero state: 6
      any state: 0
  Total gates : 6.562E+4
    'T': 3.665E+4,
    'CNOT': 1.289E+4,
    'X': 236,
    'Z': 3.800E+3,
    'S': 4.807E+3,
    'Hadamard': 7.244E+3
```

Optimized Resource Estimation

Not finished yet... Current progress — more efficient block encoding via multiplexor compilation of SELECT gates.





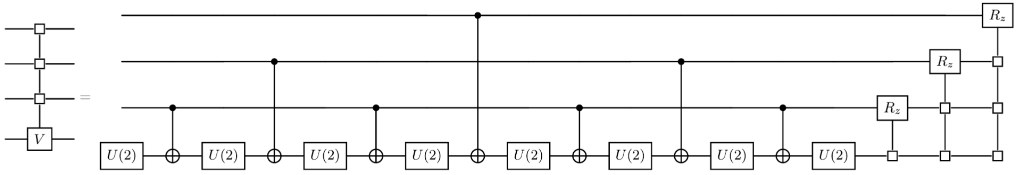


Figure 3. Decomposition for $M_n^{\mathcal{C}}(V)$ where V is a list of U(2) unitaries, $n = 1$, and $\mathcal{C} = 3$. The open squares indicate multiplexed controls. (See figures 2 and 9).

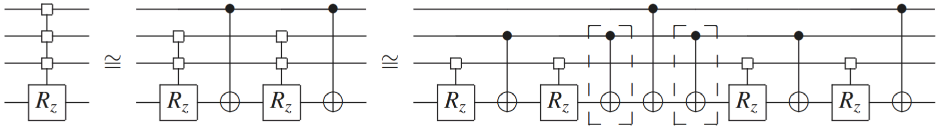


Figure 2: The recursive decomposition of a multiplexed R_z gate. The boxed CNOT gates may be canceled.

Future Considerations (Rest of the Summer and Beyond)

Future Considerations (Rest of the Summer and Beyond)

Only $\approx \frac{2}{3}$ of the way through the project term so a lot is still planned. Some of these ideas may extend further into the future as well.

Future Considerations (Rest of the Summer and Beyond)

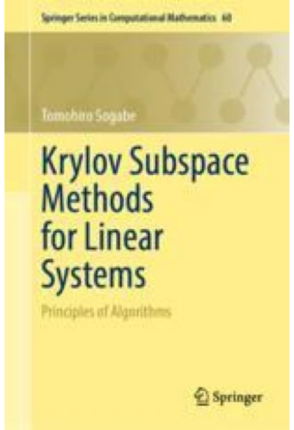
Only $\approx \frac{2}{3}$ of the way through the project term so a lot is still planned. Some of these ideas may extend further into the future as well.

- Krylov Subspace Methods — these allow for consideration of a low energy subspace, reducing H size
- Ultimately, execution on current NISQ Processors
- Incorporating error correction (building on previous work done with error correcting QSP at Yale)
- QSVT on NISQ devices for solving linear systems via matrix inversion (utilizes singular values)

Future Considerations (Rest of the Summer and Beyond)

Only $\approx \frac{2}{3}$ of the way through the project term so a lot is still planned. Some of these ideas may extend further into the future as well.

- Krylov Subspace Methods — these allow for consideration of a low energy subspace, reducing H size
- Ultimately, execution on current NISQ Processors
- Incorporating error correction (building on previous work done with error correcting QSP at Yale)
- QSVT on NISQ devices for solving linear systems via matrix inversion (utilizes singular values)



Thank you for your time!

Questions?

Thank you for your time!

Questions?

Summary:

- Quantum Singular Value Transformation is a technique to polynomially transform the singular values of an arbitrary matrix
- We can formulate many algorithms in terms of the QSVT, but it is computationally expensive.
- Block encoding is the most expensive step of QSVT, but it can be simplified in many ways by looking for structure.

Thank you for your time!

Questions?

Summary:

- Quantum Singular Value Transformation is a technique to polynomially transform the singular values of an arbitrary matrix
- We can formulate many algorithms in terms of the QSVT, but it is computationally expensive.
- Block encoding is the most expensive step of QSVT, but it can be simplified in many ways by looking for structure.

References:

[15] J. M. Martyn, Z. M. Rossi, A. K. Tan, and I. L. Chuang, A Grand Unification of Quantum Algorithms, PRX Quantum **2**, 40203 (2021).

[16] D. D. F. and R. J. W. Metromation, *Feature-Metromation*, https://preview.bnpmedia.com/pci-february-2026/feature-metromation/.

[17] J. M. Arrazola, Intro to QSVT, Pennylane Demos (2023).

[18] Y. Kikuchi, C. Mc Keever, L. Coopmans, M. Lubasch, and M. Benedetti, Realization of quantum signal processing on a noisy quantum computer, Npj Quantum Information **9**, 93 (2023).

[19] M. W. Sze, Y. Tang, S. Dilkes, D. M. Ramo, R. Duncan, and N. Fitzpatrick, Hamiltonian dynamics simulation using linear combination of unitaries on an ion trap quantum computer, Quantum Science and Technology **11**, 15023 (2026).

[20] V. V. Shende, S. S. Bullock, and I. L. Markov, Synthesis of Quantum Logic Circuits, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems **25**, 1000 (2006).

[21] H. M. Waidyasooriya and M. Hariyama, Highly-Parallel FPGA Accelerator for Simulated Quantum Annealing, IEEE Transactions on Emerging Topics in Computing **9**, 2019 (2021).

Thank you for your time!

Questions?

Summary:

- Quantum Singular Value Transformation is a technique to polynomially transform the singular values of an arbitrary matrix
- We can formulate many algorithms in terms of the QSVT, but it is computationally expensive.
- Block encoding is the most expensive step of QSVT, but it can be simplified in many ways by looking for structure.

Acknowledgements: I would like to thank my advisors, Steven Girvin and Yongshan Ding, for their support, and the Dartmouth Center for Career Design for funding my project.

email: aryan.bawa.27@dartmouth.edu

References:

[22] J. M. Martyn, Z. M. Rossi, A. K. Tan, and I. L. Chuang, A Grand Unification of Quantum Algorithms, PRX Quantum **2**, 40203 (2021).

[23] D. D. F. and R. J. W. Metromation, *Feature-Metromation*, https://preview.bnpmedia.com/pci-february-2026/feature-metromation/.

[24] J. M. Arrazola, Intro to QSVT, Pennylane Demos (2023).

[25] Y. Kikuchi, C. Mc Keever, L. Coopmans, M. Lubasch, and M. Benedetti, Realization of quantum signal processing on a noisy quantum computer, Npj Quantum Information **9**, 93 (2023).

[26] M. W. Sze, Y. Tang, S. Dilkes, D. M. Ramo, R. Duncan, and N. Fitzpatrick, Hamiltonian dynamics simulation using linear combination of unitaries on an ion trap quantum computer, Quantum Science and Technology **11**, 15023 (2026).

[27] V. V. Shende, S. S. Bullock, and I. L. Markov, Synthesis of Quantum Logic Circuits, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems **25**, 1000 (2006).

[28] H. M. Waidyasooriya and M. Hariyama, Highly-Parallel FPGA Accelerator for Simulated Quantum Annealing, IEEE Transactions on Emerging Topics in Computing **9**, 2019 (2021).