Internship Report:

Quantum protocol for arbitrary phase transformation

Ashish Nandan

Institute of Applied Physics, Friedrich Schiller University Jena, 07745 Jena, Germany
date

Supervisors:

Siavash Davani
Institute of Applied Physics, Friedrich Schiller University Jena, 07745 Jena, Germany
Max Planck School of Photonics, 07745 Jena, Germany

Prof. Martin Garttner
Abbe Center of Photonics, Friedrich Schiller University Jena, 07745 Jena, Germany

Dr. Falk Eilenberger
Institute of Applied Physics, Friedrich Schiller University Jena, 07745 Jena, Germany
Max Planck School of Photonics, 07745 Jena, Germany
Fraunhofer Institute for Applied Optics and Precision Engineering IOF, 07745 Jena, Germany

Abstract

In this project we implement a programmable quantum protocol and simulate the application of arbitrary phase transformations on quantum states efficiently. We apply a phase of our choice on a statevector using an ancilla register as a controller and analyze the resulting fidelity and error.

$$|\psi'\rangle = \psi(x)\, e^{i\alpha|\phi(x)|^2}\, |x\rangle$$

Here $e^{i\alpha|\phi(x)|^2}$ is the phase factor that we want to apply and can be manipulated for different phases. The primary and ancilla registers are initialized using Matrix Product States (MPS), and after that we use controlled-phase operations over multiple iterations.

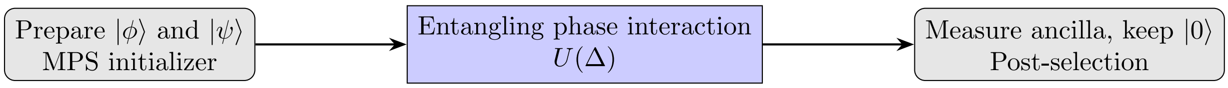
Contents

1 Introduction

In quantum mechanics, phase plays a crucial role in determining interference patterns and quantum state evolution.

2 Theory

We will look into the theory into three simples steps as below:



2.1 Initialization

Initilization is one of te major challanges in preparation of quantum circuit. A single qubit only has two amplitudes to set, but for n qubits, a general state is a superposition over 2^n basis states, each with its own complex amplitude. To prepare a state $|\psi\rangle$ starting from $|0\cdots0\rangle$, one would need a unitary U such that

$$U|0\cdots0\rangle = |\psi\rangle .$$

For highly entangled or complex states, designing such a unitary directly would require an exponentially large number of gates, which is infeasible, especially on real hardware, where noise and decoherence further complicate the process. This problem is solved by mps_initilizer here.The MPS representation captures the essential structure of such states as a sequence of smaller matrices rather than storing all 2^n amplitudes explicitly. By using the MPS initializer, the quantum state can be constructed efficiently, piece by piece, bypassing the exponential complexity problem for states of limited entanglement. This allows one to prepare a meaningful initial state in a circuit or simulator without having to implement a massive and impractical unitary.

In our context the ancilla register $|\phi\rangle$ is supposed to encode the phase profile $\phi(x)$. Without a proper initialization, the ancilla would remain in $|0\rangle$, carrying no information about the phase to be applied. By initializing $|\phi\rangle$ using a Matrix Product State (MPS), we efficiently encode the amplitude information of $\phi(x)$ into the ancilla register. Similarly, the primary register $|\psi\rangle$ can be initialized via MPS if it is in a nontrivial state, such as a structured wavefunction, to ensure the simulation accurately represents the starting condition.

2.2 Oracle $U(\Delta)$

After initialization, comes the heart of the whole protocal, $U(\Delta)$ i.e oracle. The oracle is designed in such a way that is applies phase on the primary register $|\psi\rangle$ based on the ancilla register $|\phi\rangle$ which acts as a controller. We can contral the applied phase because the phase applied depends upon the sttae of $|\phi\rangle$. As changing the state of $|\phi\rangle$ can change $e^{i\alpha|\phi(x)|^2}$ which changes the output. Mathematically oracle can be defned as

$$U(\Delta) \;=\; I \otimes I \;+\; (e^{i\Delta} - 1) \sum_x P_x \otimes P_x,$$

where

$$P_x = |x\rangle \langle x| \,, \qquad e^{i\Delta} - 1 = 2ie^{i\Delta/2} \sin\left(\tfrac{\Delta}{2}\right).$$

So the operator appears as the identity plus a projector onto the diagonal subspace $\sum_x P_x \otimes P_x$ weighted by that phase factor. We can see that for samll Δ we can expand

$$U(\Delta) = I \otimes I + i\Delta \sum_x P_x \otimes P_x + \mathcal{O}(\Delta^2).$$

This shows that $U(\Delta)$ is a weak, conditional phase kick on the subspace and only applicable when the two registers hold the same index.

In my implementation, I follow the scalable construction described in the paper: for each bit position $j = 0, \ldots, n-1$, I compare the corresponding qubits of the two registers and store an "equal/not-equal" flag in ancilla qubits (or compute the flags in another equivalent way). Once these flags are in place, I apply a single multi-controlled phase gate that triggers only when all flags indicate equality, and then I uncompute the flags to reset the ancillas.

Concretely, I first compute per-qubit flags that tell me whether each bit-pair matches. The appendix of the paper gives one method using CNOT-like gates and their variants; depending on the comparator I use, the flags can indicate equality or inequality. Next, I apply an n-controlled phase gate $P(\Delta)$, which applies the global phase $e^{i\Delta}$ only when the entire string matches. Finally, I uncompute the flags so that the ancillas return to their clean state.

Because the multi-controlled phase can be decomposed into $\mathcal{O}(n)$ elementary gates, and the flag computation/uncomputation uses $2n$ CNOT-style operations, the total gate count for $U(\Delta)$ scales as $\mathcal{O}(n)$. This makes $U(\Delta)$ efficient to implement as an oracle in terms of both gate depth and register size.

2.3 Measurement and Post-selection

The third and most crucial part is measurement and readout part.When we prepare the ancilla register in initial step, we used mps_initializer to entangle the ancilla register internally. We did it by applying by initializing in a rotated basis state $|\phi_0\rangle = U_\phi |0\rangle$. After the oracle acts, we will have the outcome the involves the superposition of both primary and ancila register. But we do not need ancilla in our measurnment, so we apply post selection measurnment and trace out ancilla register. However, the measurement must be done in the computational basis $\{|\mu\rangle\}$, not in the rotated basis $\{|\phi_\mu\rangle\}$.

To overcome this, one applies the adjoint rotation $U_\phi^\dagger$ before measurement. The key observation is that

$$U_\phi^\dagger |\phi_\mu\rangle = U_\phi^\dagger\big(U_\phi |\mu\rangle\big) = |\mu\rangle .$$

Thus, by applying $U_\phi^\dagger$ followed by a standard measurement in the computational basis, the outcome $|\mu\rangle$ is effectively equivalent to having projected onto $|\phi_\mu\rangle$ in the rotated basis. The post-selection step involves keeping only the measurement outcome $\mu = 0$, which corresponds to the ancilla being projected back onto its initial state $|\phi_0\rangle$. This ensures that the desired phase transformation is applied to the primary register $|\psi\rangle$, while all other outcomes (which would lead to incorrect transformations) are discarded. The probability of successfully obtaining the $\mu = 0$ outcome can be made high by choosing a small enough Δ and repeating the process multiple times if necessary.

After post-selection, the state of the primary register takes the form

$$\psi_0'(x) \propto \psi(x) \left(1 + 2i\, e^{i\Delta/2} \sin\frac{\Delta}{2}\, |\phi(x)|^2\right). \tag{1}$$

For the successful outcome $\mu = 0$, the ancilla is projected onto its prepared state

$$|\phi_0\rangle = |\phi\rangle. \tag{2}$$

In this case, the data amplitudes evolve as

$$\psi_0'(x) \propto \psi(x) \left(1 + 2i\, e^{i\Delta/2} \sin\frac{\Delta}{2}\, |\phi(x)|^2\right). \tag{3}$$

Using the small-Δ expansion

$$2i\, e^{i\Delta/2} \sin\frac{\Delta}{2} = i\Delta + \mathcal{O}(\Delta^2), \tag{4}$$

we obtain

$$\psi_0'(x) \propto \psi(x) \left(1 + i\Delta|\phi(x)|^2 + \mathcal{O}(\Delta^2)\right). \tag{5}$$

The probability of this successful outcome is

$$P(0) = \sum_x |\psi(x)|^2 \left(1 - 4\sin^2\frac{\Delta}{2}|\phi(x)|^2\big[1 - |\phi(x)|^2\big]\right) \geq 1 - \sin^2\frac{\Delta}{2}. \tag{6}$$

For all other outcomes $\mu \neq 0$, the post-measurement state of the primary register is

$$\psi'_{\mu\neq 0}(x) \propto \psi(x)\, \phi^*_\mu(x)\, \phi(x), \tag{7}$$

which occurs with small probability and is discarded in the post-selection step.

Thus, by retaining only the cases with $\mu = 0$, we ensure that the protocol succeeds and that the ancilla has returned to its original rotated state.

2.4 Iterations

After a single cycle, the primary register state transforms as

$$\psi(x) \mapsto \psi(x)\, e^{i\Delta|\phi(x)|^2} + O(\Delta^2), \tag{8}$$

so one cycle imprints only a small phase proportional to Δ. To apply the full target phase

$$\psi(x) \mapsto \psi(x)\, e^{i\alpha|\phi(x)|^2}, \tag{9}$$

we can repeat the cycle m times. Each successful cycle contributes another factor of $e^{i\Delta|\phi(x)|^2}$, so after m cycles, the accumulated transformation is

$$\psi(x) \mapsto \psi(x) \left(e^{i\Delta|\phi(x)|^2}\right)^m = \psi(x)\, e^{im\Delta|\phi(x)|^2}. \tag{10}$$

To match the target phase, we choose Δ and m such that

$$m\Delta = \alpha, \quad \text{or equivalently,} \quad \Delta = \frac{\alpha}{m}. \tag{11}$$

Keeping Δ small is important because the success probability of one cycle is

$$P(0) \geq 1 - O(\Delta^2), \tag{12}$$

which approaches 1 for small Δ. If Δ is too large, the ancilla may collapse into $\mu \neq 0$ and corrupt the data, which is why the algorithm uses many small steps rather than one large step.

The overall success probability after m cycles is approximately

$$P_{\text{success}} \approx (1 - O(\Delta^2))^m \approx \exp(-O(m\Delta^2)). \tag{13}$$

Substituting $m = \alpha/\Delta$ gives

$$P_{\text{success}} \approx \exp(-O(\alpha\Delta)). \tag{14}$$

Therefore, if we desire an overall failure probability ϵ, we should choose

$$\Delta = O\left(\frac{\epsilon}{\alpha}\right), \tag{15}$$

which implies the number of iterations required is

$$m = \frac{\alpha}{\Delta} = O\left(\frac{\alpha^2}{\epsilon}\right). \tag{16}$$

3 Implementation and Results

Appendix: Derivation of Eqs.

By definition, the oracle acts as

$$U(\Delta)\,|x\rangle|y\rangle = \begin{cases} e^{i\Delta}|x\rangle|y\rangle & x = y, \\ |x\rangle|y\rangle & x \neq y. \end{cases} \tag{17}$$

This can be written compactly as

$$U(\Delta)\,|x\rangle|y\rangle = \Big[1 + (e^{i\Delta} - 1)\delta_{x,y}\Big]\;|x\rangle|y\rangle. \tag{18}$$

Promoting $\delta_{x,y}$ to operator form gives

$$\delta_{x,y} = \sum_z \langle x|z\rangle\langle y|z\rangle = \sum_z \langle x| \otimes \langle y|\,\big(|z\rangle \otimes |z\rangle\big), \tag{19}$$

so the operator is

$$U(\Delta) = I \otimes I \;+\; (e^{i\Delta} - 1)\sum_z |z\rangle\langle z| \otimes |z\rangle\langle z|. \tag{20}$$

We begin with the oracle definition:

$$U(\Delta) \;=\; I \otimes I \;+\; (e^{i\Delta} - 1)\sum_x \;|x\rangle\langle x| \;\otimes\; |x\rangle\langle x|. \tag{21}$$

- The first $|x\rangle\,\langle x|$ acts on the data register (basis index x).
- The second $|x\rangle\,\langle x|$ acts on the software register (basis index y).

The input state of data ($|\psi\rangle$ register) and software ($|\phi\rangle$ register) is

$$|\Psi_{\text{in}}\rangle = \sum_x \psi(x)\,|x\rangle \otimes |\varphi_0\rangle. \tag{22}$$

Now consider the action of $U(\Delta)$ on a fixed data component $|x\rangle$:

$$U(\Delta)\,\big(|x\rangle \otimes |\varphi_0\rangle\big) = \Big(I \otimes I + (e^{i\Delta} - 1)\sum_{x'} |x'\rangle\langle x'| \otimes |x'\rangle\langle x'|\Big)\,\big(|x\rangle \otimes |\varphi_0\rangle\big) \tag{23}$$

$$= |x\rangle \otimes |\varphi_0\rangle \;+\; (e^{i\Delta} - 1)\,|x\rangle \otimes \big(|x\rangle\langle x|\,|\varphi_0\rangle\big). \tag{24}$$

We can now factor out $|x\rangle$:

$$= |x\rangle \otimes \Big(\big[I + (e^{i\Delta} - 1)\,|x\rangle\langle x|\big]\,|\varphi_0\rangle\Big). \tag{25}$$

Thus, defining

$$U_x \ = \ I + (e^{i\Delta} - 1)P_x, \qquad P_x = |x\rangle\langle x|, \tag{26}$$

we obtain

$$U(\Delta)\left(|x\rangle \otimes |\varphi_0\rangle\right) = |x\rangle \otimes \left(U_x|\varphi_0\rangle\right). \tag{27}$$

Finally, by expanding the full input state we find

$$|\Psi_{\text{out}}\rangle = \sum_x \psi(x)\,|x\rangle \otimes \left(U_x|\varphi_0\rangle\right). \tag{28}$$

Appendix: b

Let U_φ be an ancilla initializer such that $U_\varphi|\mu\rangle = |\varphi_\mu\rangle$; in particular $|\varphi_0\rangle \equiv |\varphi\rangle$. The joint state after applying the circuit $I \otimes U_\varphi^\dagger\, U(\Delta)\, I \otimes U_\varphi$ to $|\psi\rangle \otimes |0\rangle$ is

$$\begin{aligned}|\Gamma\rangle &= \big(I \otimes U_\varphi^\dagger\big) U(\Delta) \big(I \otimes U_\varphi\big)\, |\psi\rangle|0\rangle \\ &= \sum_x P_x \otimes \big(U_\varphi^\dagger U_x U_\varphi\big)\, |\psi\rangle|0\rangle. \end{aligned} \tag{B1}$$

Measuring the ancilla in the computational basis with outcome μ yields (unnormalized)

$$|\psi'_\mu\rangle \propto \big(I \otimes \langle\mu|\big)|\Gamma\rangle = \sum_x \psi(x)\, \langle\mu|U_\varphi^\dagger U_x U_\varphi|0\rangle\, |x\rangle, \tag{B2}$$

where $\psi(x) = \langle x|\psi\rangle$. Using $|\varphi_\mu\rangle = U_\varphi|\mu\rangle$ and $|\varphi_0\rangle = |\varphi\rangle$,

$$\langle\mu|U_\varphi^\dagger U_x U_\varphi|0\rangle = \langle\varphi_\mu|U_x|\varphi_0\rangle.$$

With $U_x = I + 2ie^{i\Delta/2}\sin(\frac{\Delta}{2})P_x$ one obtains

$$\langle\varphi_\mu|U_x|\varphi_0\rangle = \langle\varphi_\mu|\varphi_0\rangle + 2i\, e^{i\Delta/2} \sin\Big(\tfrac{\Delta}{2}\Big)\langle\varphi_\mu|P_x|\varphi_0\rangle. \tag{B3}$$

Writing $\varphi_\mu(x) = \langle x|\varphi_\mu\rangle$ gives $\langle\varphi_\mu|P_x|\varphi_0\rangle = \varphi_\mu^*(x)\varphi_0(x)$, so

$$\langle\varphi_\mu|U_x|\varphi_0\rangle = \langle\varphi_\mu|\varphi_0\rangle + 2i\, e^{i\Delta/2} \sin\Big(\tfrac{\Delta}{2}\Big)\varphi_\mu^*(x)\varphi_0(x).$$

For the "good" measurement outcome $\mu = 0$ (so $|\varphi_0\rangle = |\varphi\rangle$), $\langle\varphi_0|\varphi_0\rangle = 1$ and

$$\psi'_0(x) \propto \psi(x)\Big[1 + 2i\, e^{i\Delta/2} \sin\Big(\tfrac{\Delta}{2}\Big)\, |\varphi(x)|^2\Big]. \tag{B4}$$

The probability of obtaining $\mu = 0$ is

$$P(0) = \sum_x |\psi(x)|^2 \Big|1 + 2i\, e^{i\Delta/2} \sin\Big(\tfrac{\Delta}{2}\Big)\, |\varphi(x)|^2\Big|^2,$$

which simplifies to

$$P(0) = \sum_x |\psi(x)|^2 \Big(1 - 4\sin^2\tfrac{\Delta}{2}\, |\varphi(x)|^2 \big(1 - |\varphi(x)|^2\big)\Big). \tag{B5}$$

Using $|\varphi(x)|^2(1 - |\varphi(x)|^2) \leq \frac{1}{4}$ we obtain the lower bound

$$\boxed{P(0) \geq 1 - \sin^2\Big(\tfrac{\Delta}{2}\Big)} \tag{B6}$$

so for small Δ the success probability is close to unity.

Remarks.

- For $\Delta \ll 1$, $2ie^{i\Delta/2}\sin(\Delta/2) \approx i\Delta$, so the per-cycle factor in (B4) approximates $e^{i\Delta|\varphi(x)|^2}$. Repeating m cycles with $\Delta = \alpha/m$ approximates $e^{i\alpha|\varphi(x)|^2}$.

- If Δ is not small the $\sin(\Delta/2)$ prefactor introduces nonlinear dependence on Δ (oscillations), explaining non-monotonic fidelity vs. m when α is fixed and $\Delta = \alpha/m$ exits the small-Δ regime.

- Measurement outcomes $\mu \neq 0$ are low-probability branches that lead to corrupted transformations proportional to $\varphi_\mu^*(x)\varphi(x)$ (see B3); they are the origin of rare but damaging error events.

- In your circuit: U_φ is the MPS initializer on the ancilla, the controlled-phase/multi-controlled-phase implements the $U(\Delta)$-style coupling, then $U_\varphi^\dagger$ and ancilla measurement implement the projection onto $|\mu\rangle$.