Quantum Memory and Quantum Storage

Jamal Slim

DESY - FTX

jamal.slim@desy.de

Software weekly meetings
February 1, 2024

Quantum storage

- Let $b^m = (b_1^m, \cdots b_n^m)$ be the n-bit nonary representation where each feature x_i^m id represented by a single bit, i.e., $x_i^m = b_i^m$
- M is the number of non-zero amplitude
- n is the number of bits
- Goal: $M << n$
- The following quantum system is required

$$|\ell_1, \cdots, \ell_n; a_1 a_2; s_1, \cdots, s_n\rangle$$

  - $\{\ell_i\}$: loading register
  - $\{a_i\}$: ancilla register
  - $\{s_i\}$: storage register

Quantum storage

- The system is initialized to the ground state as

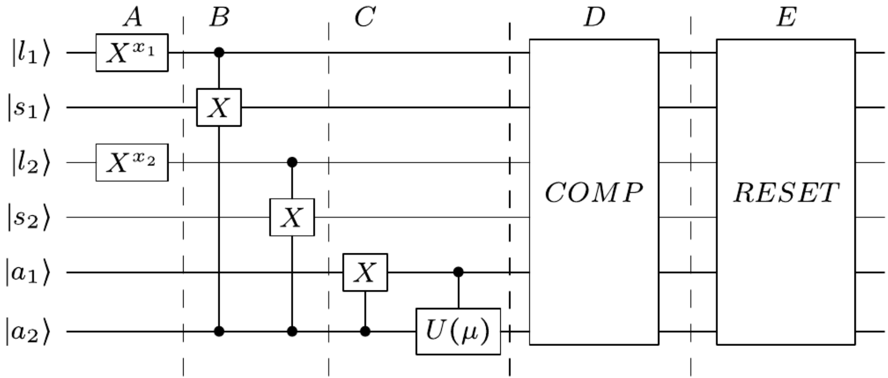
$$|0, \cdots, 0; 00; 0, \cdots, 0\rangle$$

- Apply the Hadamard gate to a_2 leads to

$$\frac{1}{\sqrt{2}}|0, \cdots, 0; 00; 0, \cdots, 0\rangle + \frac{1}{\sqrt{2}}|0, \cdots, 0; 01; 0, \cdots, 0\rangle$$

- The algorithm loads iteratively patterns into the loading register and breaks up the correct length of terms from the processing branch to add it to the memory branch
- The superposition of patterns is built gradually

Quantum storage



Quantum storage

- Assume after the first *m* training vectors have already been encoded after *m* iterations

$$\frac{1}{\sqrt{M}}\sum_{k=1}|0,\cdots,0;00;x_1^k,\cdots,x_n^k\rangle+\sqrt{\frac{M-m}{M}}|0,\cdots,0;01;0,\cdots,0\rangle$$

- The memory branch stores the first *m* input in its storage register while the storage register of the processing branch is in the ground state
- In both branches, the loading register is also in the ground state

Quantum storage

- In the $m+1$ step
  - write the $(m+1)^{\text{th}}$ pattern into the qubits if the load register by applying ab $X-$gate to all of the qubits that correspond to non-zero bits in the input pattern
  - In the processing branch, the pattern gets copied into the storage register using a CNOT-gate on each of the n-qubit
  - Execution only on the processing branch ($A_2 = 1$)

$$\frac{1}{\sqrt{M}} \sum_{k=1} |x_1^{m+1}, \cdots, x_n^{m+1}; 00; x_1^k, \cdots, x_n^k\rangle + \sqrt{\frac{M-m}{M}} |x_1^{m+1}, \cdots, x_n^{m+1}; 01; x_1^{m+1}, \cdots, x_n^{m+1}\rangle$$

Quantum storage

- Using a CNOT-gate
  - flip $a_1 = 1$ if $a_2 = 1$ (processing branch)
- Applying single qubit unitary $U(\tau)$

$$U_{a_2}(\tau) = \begin{pmatrix} \sqrt{\frac{\tau-1}{\tau}} & \frac{1}{\sqrt{\tau}} \\ \frac{1}{\sqrt{\tau}} & \sqrt{\frac{\tau-1}{\tau}} \end{pmatrix}$$

- $\tau = M + 1 - (m + 1)$
- Applying $U_{a_2}(\tau)$ to a_2 controlled by a_1

$$\mathbb{1}_\ell \otimes c_{a_2} U(\tau) \otimes \mathbb{1}_s$$

Quantum storage

- This splits the processing branch into 2 sub-branches
  - one can be added to the memory branch
  - one remains in the processing branch for the next step
- Applying single qubit unitary $U(\tau)$

$$\frac{1}{\sqrt{M}}\sum_{k=1}|x_1^{m+1},\cdots,x_n^{m+1};00;x_1^k,\cdots,x_n^k\rangle +$$
$$\frac{1}{\sqrt{M}}\sum_{k=1}|x_1^{m+1},\cdots,x_n^{m+1};10;x_1^{m+1},\cdots,x_n^{m+1}\rangle +$$
$$\sqrt{\frac{M-(m+1)}{M}}|x_1^{m+1},\cdots,x_n^{m+1};11;x_1^{m+1},\cdots,x_n^{m+1}\rangle$$

Quantum storage

- Now, to add the sub-branch marked by $|a_1 a_2\rangle = |10\rangle$ to the memory branch
- Flip a_1 back to 1
- Comparison block
  - to confine this operation to the desired branch, compare if the loading and storage registers are in the same state which is only true for the processing branch $a_2 = 1$ for the desired subbranch.
- Reset to the ground state: reversing the previous operations