

$\log n$ - Space, $n^{3/2}$ Time Quantum Sort.

David Ponarovsky

January 29, 2024

It has been proven that any quantum algorithm in the quantum circuits which sorts at time T and storage space S has to satisfy the restriction $TS = \Omega(n^{3/2})$ [Kla03]. In the regime of $S \geq \log^3(n)$, it has been shown that the bound is tight up to logarithmic factors. However, in the regime where S is strictly $\Theta(\log(n))$, not much advancement has been reached beyond $T = \Theta(n^{1\frac{1}{2}} \log n)$. Here, we present a quantum algorithm that sorts with $\log(n)$ storage memory and $\Theta(n^{3/2})$ time. We achieved this by quantifying the sorting algorithm invented by Stanley P. Y. Fung [Fun21], who coined its name - "ICan'tBelieveItCanSort" - due to the surprise of having such a simple sorting algorithm.

Result: Sorting A_1, A_2, \dots, A_n

```
1 for  $i \in [n]$  do
2   for  $j \in [n]$  do
3     if  $A_i < A_j$  then
4       swap  $A_i \leftrightarrow A_j$ 
5     end
6   end
7 end
```

Algorithm 1: "ICan'tBelieveItCanSort" alg.

Result: Sorting A_1, A_2, \dots, A_n

```
1 swap  $A_1 \leftrightarrow \max A$ 
2 for  $i \in [n - 1]$  do
3   Find the first  $k$  such  $A_k > A_i$ 
4   Set  $A \leftarrow A_1, A_2 \dots A_{k-1}, A_i, A_k, A_{k+1}, \dots, A_{i-1}, A_{i+1} \dots, A_n$ 
5 end
```

Algorithm 2: "ICan'tBelieveItCanSort" alg.

References

- [Kla03] Hartmut Klauck. *Quantum Time-Space Tradeoffs for Sorting*. 2003. arXiv: [quant-ph/0211174](#) [quant-ph].
- [Fun21] Stanley P. Y. Fung. "Is this the simplest (and most surprising) sorting algorithm ever?" In: *CoRR* abs/2110.01111 (2021). arXiv: [2110.01111](#). URL: <https://arxiv.org/abs/2110.01111>.

Result: Sorting A_1, A_2, \dots, A_n

```

1 swap  $A_1 \leftrightarrow \max A$ 
2 for  $i \in [n - 1]$  do
3   Set  $\text{current} \leftarrow \text{head.next}$ 
4    $k\text{-pointer} \leftarrow$  Find the first ' $k < i$ ' node such ' $A_k > A_i$ ' using Grover querying the follow
5   Ask if (  $\text{node.color} = \text{red}$  and  $\text{node.value} > \text{current.value}$ 
6   and  $\text{node.back.value} \leq \text{current.value}$  )
7   Set  $\text{head.next} \leftarrow \text{head.next.next}$ 
8   Set  $\text{head.next.back} \leftarrow \text{head}$ 
9   Set  $\text{current.next} \leftarrow k\text{-pointer}$ 
10  Set  $\text{current.back} \leftarrow k\text{-pointer.back}$ 
11  Set  $\text{current.back.next} \leftarrow \text{current}$ 
12  Set  $\text{current.color} \leftarrow \text{red}$ 
13 end
```

Algorithm 3: "Quantum ICan'tBelieveItCanSort" alg.