A Linear Time Quantum Algorithm for Pairwise Sequence Alignment

Md. Rabiul Islam Khan, Shadman Shahriar, and Shaikh Farhan Rafid

Department of Computer Science and Engineering, Brac University, Dhaka, Bangladesh

Sequence Alignment is the process of aligning biological sequences in order to identify similarities between multiple sequences. In this paper, a Quantum Algorithm for finding the optimal alignment between DNA sequences has been demonstrated which works by mapping the sequence alignment problem into a path searching problem through a 2D graph. The transition which converges to a fixed path on the graph is based on a proposed oracle for profit calculation. By implementing Grover's search algorithm, our proposed approach is able to align a pair of sequences and figure out the optimal alignment within linear time [1], which hasn't been attained by any classical deterministic algorithm. In addition to that, the proposed algorithm is capable of quadratic speeding up to any unstructured search problem by finding out the optimal paths accurately in a deterministic manner, in contrast to existing randomized algorithms that frequently sort out the sub-optimal alignments, therefore, don't always guarantee of finding out the optimal solutions.

1 Introduction

Since the midst of the last century, there has been an issue to be addressed regarding classical computers, that these are not sustainable to keep up the pace with emerging necessities of speeding up the processing of information. Due to these limitations, quantum mechanics is being considered as a powerful counterpart in the race of designing and introducing a future automaton to cope up with the challenges [2].

The alignment of sequences; arranging sequences of DNA, RNA or Proteins to demonstrate regions of similarity in the biological, structural and identical connections requires adequate amount of information and time [3]. With the progress of quantum computation in this decade, the sequence alignment problem can be solved with nearly full precision in contrast to previous probabilistic approaches.

Quantum mechanics harnesses the phenomenon of superposition, entanglement, tunneling and annealing to solve problems that take a tremendous amount of time [4]. Superposition allows quantum bits to be represented with 0, 1 or both at the same time [5]. If two systems are strongly co-related to each other then gaining the information of one system will immediately provide the information for the other system; this effect is quantum entanglement [6].

In this work, we look forward to harness the computational capabilities of quantum computers to solve the problem of aligning sequences. Rather than taking a probabilistic approach that is quite practical for classical computers, a deterministic approach has been proposed along with the proper guarantee to figure out the optimal path has been provided. We can also be hopeful to speed up the time required for figuring out a desired solution.

1.1 Sequence Alignment

Sequence alignment is the technique of analyzing and uncovering similarities between biological sequences; in a variety of bio-informatics applications, sequence alignment technique is used to align sequences of DNA, RNA, proteins and non-biological sequences as well [7].

Finding the optimal alignments of DNA sequences has been one of the most challenging aspects of Bio informatics. To find the optimal alignment, there have been several computational approaches for both pairwise and multiple/global sequences. These approaches include the naive approaches, slow yet working dynamic programming approach and more efficient approaches for large databases such as heuristic and probabilistic approaches.

With the progression of Quantum computers within this decade, the applications of these computing machines are being tested theoretically. Particularly

Md. Rabiul Islam Khan: rabiul.islam.khan@g.bracu.ac.bd

in bio informatics, quantum algorithms are believed to solve some of the most complex computational problems. Although DNA sequence alignment is a decade long computational challenge and has been proven to be useful in many applications of bio informatics, there has not been any significant approaches to develop a suitable Quantum algorithm in order to find the optimal alignment.
The most naive approach is to search the similarities between two sequences i.e. pairwise alignment. Needleman and Wunsch (1970) presented the inaugural approach with dynamic programming which was meant for protein sequences [8]. However, because of the running time and memory requirements, the dynamic approaches are proven to be quite impractical.
Moreover, although heuristic and stochastic methods are more efficient, none of these algorithms properly guarantee to sort out the optimal alignment. Also, running time and memory requirements also have been an issue in case of the classical algorithmic approaches.
DNA sequence alignment algorithms have been built and implemented to both pairwise and global alignments and has been a topic of extensive studies. The algorithms can be classified as deterministic, stochastic and heuristic. Needleman and Wunsch (1970) presented a dynamic approach for global alignments to sort out protein sequences which was the first approach to find sequence alignments [8]. Similar approach for local alignments have been initiated by Smith and Waterman (1981) [9]. The dynamic algorithms can guarantee to find optimal paths on the condition of defining a good scoring function which is quite unsuitable for larger sequences. These algorithms create a matrix, where each cell represents the similarity score of the sub-string of the first sequence ending at that row and the sub-string of the second sequence ending at that column. The algorithm then fills in the matrix by comparing each residue of the two sequences and scoring their similarity.
The Gibbs sampler approach proposed by Lawrence, Altschul, Boguski, Liu, Neuwald, and Wootton (1993) presented a stochastic approach [10]. Stochastic approach works better than dynamic approach for larger datasets but there remains the probability of returning an optimal or suboptimal path. Although, we are primarily focusing on deterministic approaches to address our problem statement. Stochastic approaches are quite similar to heuristic approaches.
Heuristic approach may also return a suboptimal path. These algorithms define problem specific search techniques in contrast to the stochastic approaches. Pevzner (1992) presented some examples and spotted the core differences between the dynamic and heuristic approaches [11].
More recent works include, Sanchez, Salami, Ramirez and Valero (2006) who presented a micro-architecture performance analysis of recognized biological applications in order to compare and align sequences [12]. They adopted a methodology based on simulation and performed detailed workload characterization of the applications regarding sequence comparison as well as the alignment task.

1.2 Quantum Algorithms for Aligning Sequences

The sequence alignment problem is being addressed to be solved and performed under a potential quantum computer for the last couple of years and several algorithms have been proposed. A pattern matching algorithm based on the hamming distance named *QiBAM* has been proposed by Sarkar et al.(2019) which can provide quadratic speedup and can be implemented using Grover's algorithm [13].
Quite different from the previous approaches, an updated approach of connecting dot-matrix plotting and quantum pattern recognition has been initiated by Prousalis (2019) in order to improve the process of aligning sequences [14].

2 Framework

In order to address the sequence alignment problem, this work is focusing on finding out the optimal sequence of any pairwise alignment; dealing with a pair of sequences in contrast to global alignment which deals with a large number of sequences in a database.
The proposed method starts with building a proper graph. For a pair of sequences, a 2-Dimensional graph is required to be implemented. The method may refer to path searching algorithms with the basis of calculated path cost/profit.

2.1 2D Graph

The method to build a proper graph has been initiated from Needleman and Wunsch (2002) [8]. A

generated edit graph for a pairwise sequence alignment is demonstrated in figure 1. If we have two sequences of DNA,

A T G G T C A G C
A C G G T C

Here the lengths of the sequences are 9 and 6 respectively. Therefore, the generated 2D array will have total of (10×7) = 70 nodes.

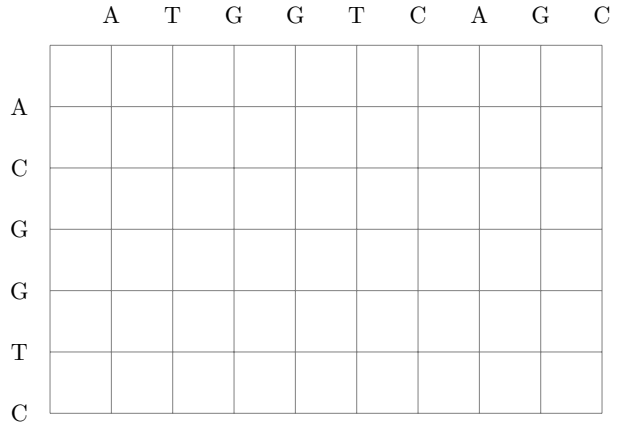

Figure 1: Edit graph

2.2 Generate Paths through the Edit Graph

After building a graph for the problem statement, next step is to generate paths through the graph. Here, every path corresponds to an alignment. But not all paths may correspond to the optimal alignment.
If the optimal path is figured out through 'Edit Distance' (Minimum number of operations required to change one sequence to another so that the sequences completely match each other), the paths with the lowest 'Edit Distance' (lowest number of mismatched characters) will correspond to the optimal path generated through the graph.
The transition (path) from one node to another should be either horizontal (right), vertical (below) or diagonal (right-below).

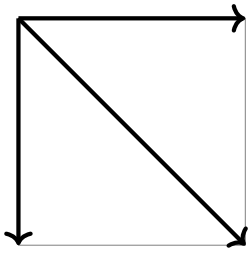

Figure 2: Possible transitions from a node

For the sequences in section 2.1, following figure demonstrates a generated optimal path though the graph which corresponds to the optimal alignment.

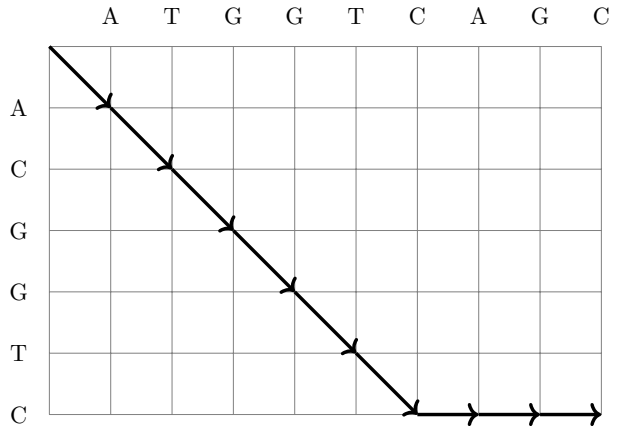

Figure 3: Optimal path generated through the edit graph

The marked path of the above graph corresponds to the following alignment,

A T G G T C A G C
A C G G T C _ _ _

The gaps correspond to "indel". Mismatch happens when characters of the same index do not match. The "edit operation" indicates three operations in case of mismatches or indels.

- Substitution is required for mismatches,
- Insertion is required in case of indels.
- Deletion is the process of removing a character.

The goal here is to minimize the differences between two sequences or in other words, generate a path through the graph with the minimum number of mismatches possible, considering all possible paths/combinations possible.

3 Organization

3.1 Path Cost/Profit

In order to find out the optimal path, a proposition has been made on how the path cost/profit should be calculated. The particular method works by assigning cost/profit to each of the edges. In this paper, instead of cost, profit shall be calculated based on the proposed oracle. The path with the highest profit shall correspond to the optimal path.
A single block of the graph/grid shall act as a unit for profit calculation. For the following figure,

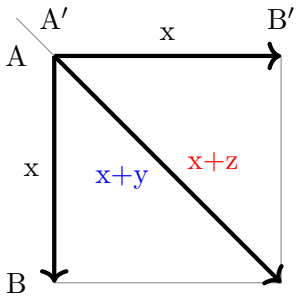

Figure 4: Possible path profits for single transition

- x, y, z > 0
- z $>$ y
- {A, A′, B, B′} $\in$ {A, G, C, T}

Here,

1. If, Tr(A, A′) = (B, B′) and B = B′, then profit should be x+z.
2. If, Tr(A, A′) = (B, B′) and B $\neq$ B′, then profit should be x+y.
3. If, Tr(A, A′) $\neq$ (B, B′), then profit should be x.

Although, x, y and z can be any natural number, x, y, z $\in \mathbb{N}$ with respect to the above conditions, throughout this work, the value of x, y and z shall be 1, 1, 2 respectively to insure simplicity in circuit building.

3.2 Quantum Implementation

To demonstrate the transition in a single block, a composite state of two qubits shall be used; the first qubit represents a horizontal transition and the second qubit represents a vertical transition. Here, the orientation can be vice versa as well.

Composite State	Transition	Profit	Direction
$\|00\rangle$	No Transition	0	•
$\|01\rangle$	Vertical (Lower)	x	↓
$\|10\rangle$	Horizontal (Right)	x	→
$\|11\rangle$	Diagonal (Lower Right)	(x+y)/(x+z)	↘

Table 1: Quantum representation of Transitions

3.3 Number of Transitions

- If, there are a pair of sequences, whose length are m and n respectively then at most (m + n) transitions are required.
- If, m = n, then at least {(m + n) / 2} transitions are required.
- If, m $\neq$ n and $m > n$, then at least {n + (m - n)} transitions are required.
- If, m $\neq$ n and $m < n$, then at least {m + (n - m)} transitions are required.

If, t transitions are required to sort out the optimal path, t pairs of qubits or, simply $2t$ qubits are required to demonstrate every transitions.

3.4 Profit, Horizontal and Vertical Nodes

If there are a pair of sequences, whose length are m and n respectively then maximum profit should be,

- $3n$, if m = n.
- $3n + (m - n)$, if, $m \neq n$ and $m > n$.
- $3m + (n-m)$ otherwise.

The number of qubits required for profit calculation should be the minimum number of bits required for the binary representation of the maximum profit. For the demonstration of horizontal and vertical shifts/transitions, number of qubits required shall be the minimum number of bits required to represent m and n.

3.5 The Letters in the Sequence

As, there can be four separate letters, A, T, G and C in the sequences, two qubits may suffice for the quantum representation. Although, there are no formal conventions, table 2 can be followed to signify biological properties of DNA.

Composite State	Letter
$\|00\rangle$	A
$\|01\rangle$	C
$\|10\rangle$	G
$\|11\rangle$	T

Table 2: Quantum representation of Nucleotide Bases

3.6 Quantum Random AccessMemory

In case of a classical computer, a RAM or Random Access Memory randomly addresses 2^n memory cells with n bits. A qRAM or quantum Random Access Memory can address the quantum superposition of 2^n memory cells with n qubits. The concept of qRAM has been proposed by Giovannetti, Lloyd and Maccone (2008) [15].
The proposed qRAM accesses memory addresses coherently using quantum superposition. To access a superposition of memory cells, the superposition of addresses must be inherited by an address register 'a' and through a superposition of data inherited by a data register 'd', qRAM passes the superposition of data to the quantum computer which needed to access the superposition of memory cells.
If we have two registers; an address register $|j\rangle$ and a data register $|D_j\rangle$,

$$\sum_j \psi\,|j\rangle_a \quad \xrightarrow{qRAM} \quad \sum_j \psi\,|j\rangle_a\,|D_j\rangle_d \qquad (1)$$

Here, $\sum_j \psi\,|j\rangle_a$ corresponds to the superposition of addresses. The data D_j is stored in the *j*th address/location of the memory cell.

$$|j\rangle \rightarrow |D_j\rangle \qquad (2)$$

To match sequences, two separate qRAM's are needed in order to generate specific characters/letters from the pair of DNA sequences and should correspond to a specific address/index of the address register.

4 Circuit and Algorithm

4.1 Path Generation and Index Calculation

By applying Hadamard gate to the pairs of transition qubits, we can generate all possible paths through the property of quantum superposition. The path profits are calculated simultaneously along the process. Total number of steps required are equal to the number of total pair of transition qubits (One qubit corresponds to the horizontal shift and another corresponds to the vertical shift). Path generation for a single step is shown in figure 5.
Here, to match two characters (horizontal and vertical) in case of any lower diagonal transition, it is required to keep track of their particular indexes. We are using two counters which increments along with every corresponding transition.

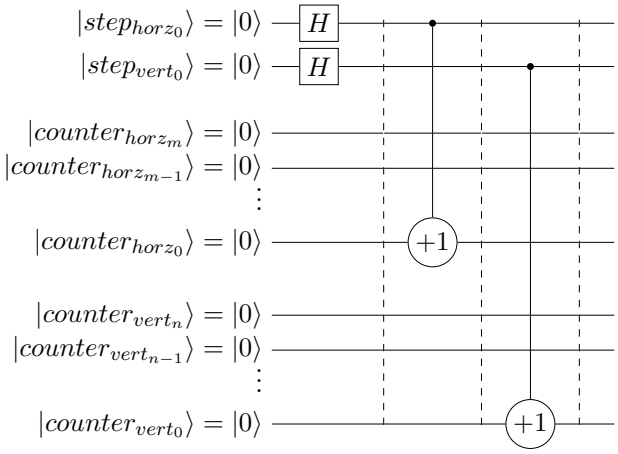

Figure 5: Path generation and counter incrementation for a single step

4.2 Quantum Adder Circuit

Although there are several methods and circuits for addition operations in quantum computers, we are referring to Draper (2000) [16].
Primary reason behind choosing this approach is for being convenient in terms of qubit size reduction and deduction of the necessity of carry bits. This method can be implemented using the concept of Quantum Fourier Transform (QFT).

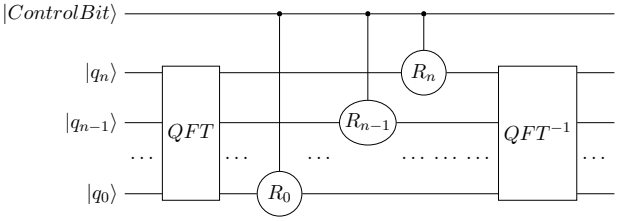

Figure 6: Quantum adder circuit

To add two numbers, the algorithm works as follows:

1. Apply QFT on any of the numbers.

2. Apply controlled phase gates (Transform Addition) on the transformed qubits. The rest of the qubits should act as control bits. Here, phase gates should operate as conditional rotation matrices,

$$R_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{\frac{i\pi}{2^k}} \end{bmatrix} \quad (3)$$

3. Perform IQFT (Inverse Quantum Fourier Transform) to acquire the result of addition.

In this work, the maximum profit from a single step is equal to 1.

4.3 Circuit for matching characters

To match two separate characters, the proposed design requires four qubits as each character require two qubits. To get any character corresponding to a specific index, two qRAM's are applied; for horizontal and vertical sequences accordingly.

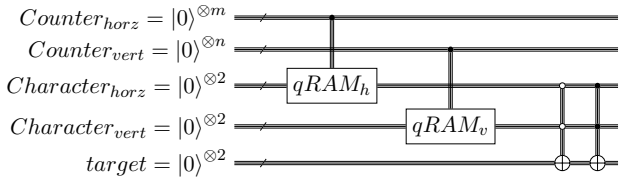

Figure 7: Circuit for matching two characters using qRAM

A sample demonstration is provided in figure 7. Both of the target bits would be $|1\rangle$ if the characters matches each other.

4.4 Profit Calculation

Completed circuit for calculating path along with the generated path is given below. This approach ensures the traversal/visit to every possible nodes in contrast to probabilistic/heuristic approaches. Also, total number of steps are also not increased and therefore, can figure out the optimal alignment using deterministic approach with nearly full precision. As mentioned previously, if there are *t* number of transitions/steps, total *2t* registers are required for those steps. For the counters, number of qubits will be equal to the number of bits required for the binary representation of the length of the horizontal and vertical sequence.

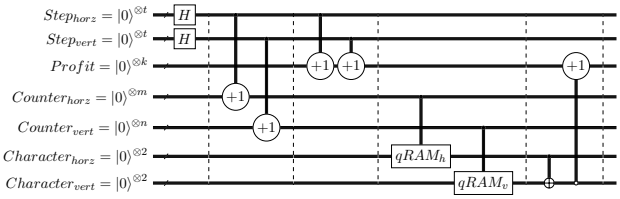

Figure 8: Quantum circuit to generate path and calculate profit

4.5 Finding the Path with Maximum Profit

To find out the optimal path, this work is focusing on the approach taken by Ahuja and Kapoor (1999) [17]. The proposed method uses Grover's search algorithm to find out the maximum element from an unsorted array.
The algorithm for finding out the maximum profit:

1. Start with any initial guess of an index *a* from an array *D* of length N, such that a $\in$ (0, ..., N - 1).

2. Repeat a loop for O($\sqrt{N}$) times:

   - Take a initialized state using n-bit Hadamard transfomation;
     $|\psi\rangle = \sum_i \frac{1}{\sqrt{N}} |i\rangle |a\rangle$. *i* is the index of the maximum element.
   - Find marked states using Grover's algorithm such that the following oracle's are satisfied.

     $$f_i(j) = 1, if\, D[j] > D[i]\, and\, f_a(x) = 1 \quad (4)$$

   - Make measurements. Replace *a* with the result of the measurements to make a new guess.

3. Return the index of the maximum element.

4.6 Complete Circuit

The complete circuit would be the merge between modified circuit of Grover's algorithm for finding out the maximum element from an unsorted array and the circuit for calculating profit by generating paths through the edit graph. Full circuit demonstration is provided in figure 9.

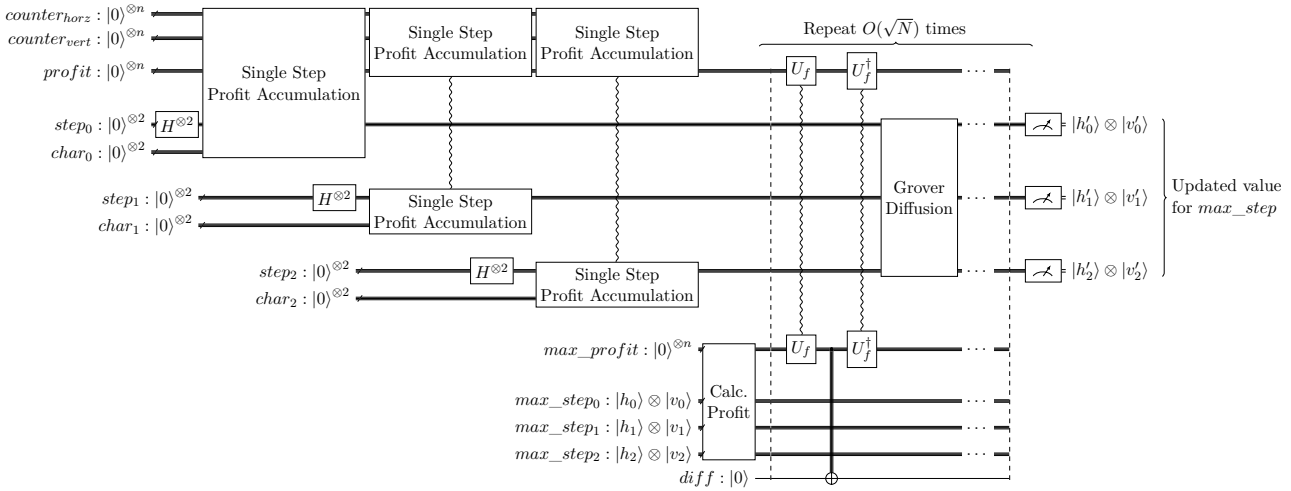

Figure 9: Quantum Circuit for Pairwise Sequence Alignment

4.7 Complexity and Analysis

In the first section of generating the paths and calculating profit, N (or, M, let N is the length of the first sequence and M is the length of the second sequence.) steps would be required on average. To find out the optimal path, the searching process based on Grover's algorithm requires $\sqrt{N}$ steps [1]. Let, N be the average length of the sequences.

Total number of steps will be the total time required by both the processes. Therefore, time complexity of the algorithm is $O(N) + O(\sqrt{N})$. After eliminating the non-dominant term it turns to be $O(N)$.

It should be noted that, the algorithm is able to find out the accurate solution in linear time. Therefore, it can surpass the existing randomized approaches as these algorithm don't find out the optimal alignments in every iteration although the runtime is nearly the same. Again, the algorithm provides quadratic speeding up to the deterministic algorithms.

5 Conclusion

DNA Sequence alignment is one of the topics of extensive research in computational biology and bioinformatics. In this work, we proposed a quantum algorithm to find out the optimal alignment by implementing a deterministic method based on a graph traversing problem and Grover's search algorithm. The algorithm guarantees of finding out the optimal alignment in linear time which is not achievable by existing classical algorithms. We hope to further extend our work to solve path searching and graph traversal problems in near future.

References

[1] Grover, L. A Fast Quantum Mechanical Algorithm for Database Search. *Proceedings Of The 28th Annual ACM Symposium On Theory Of Computing*. pp. 212-219 (1996)

[2] Aaronson, S. Quantum Computing Since Democritus. (Cambridge University Press,2013)

[3] Gollery, M. Bioinformatics: Sequence and Genome Analysis. *Clinical Chemistry*. **51**, 2219-2220 (2005)

[4] Hidary, J. Quantum Computing: An Applied Approach. (Springer,2021)

[5] Yanofsky, N. & Mannucci, M. Quantum Computing for Computer Scientists. (Cambridge University Press,2008)

[6] Nielsen, M. & Chuang, I. Quantum Computation and Quantum Information. (American Association of Physics Teachers,2002)

[7] Mount, D. Bioinformatics: Sequence and Genome Analysis. (Cold Spring Harbor Laboratory Press, New York,2004)

[8] Needleman, S. & Wunsch, C. A General Method Applicable to the Search for Similarities in the Amino Acid Sequence of Two Proteins. *Journal Of Molecular Biology*. **48**, 443-453 (1970)

[9] Smith, T. & Waterman, M. Identification of Common Molecular Subsequences. *Journal Of Molecular Biology*. **147**, 195-197 (1981)

[10] Lawrence, C., Altschul, S., Boguski, M., Liu, J., Neuwald, A. & Wootton, J. Detecting Subtle Sequence Signals: A Gibbs Sampling Strategy for Multiple Alignment. *Science*. **262**, 208-214 (1993)

[11] Pevzner, P. Multiple Alignment, Communication Cost, and Graph Matching. *SIAM Journal On Applied Mathematics*. **52**, 1763-1779 (1992)

[12] Sánchez, F., Salami, E., Ramirez, A. & Valero, M. Performance Analysis of Sequence Alignment Applications. *2006 IEEE International Symposium On Workload Characterization*. pp. 51-60 (2006)

[13] Sarkar, A., Al-Ars, Z., Almudever, C. & Bertels, K. An Algorithm for DNA Read Alignment on Quantum Accelerators. *ArXiv Preprint arXiv:1909.05563*. (2019)

[14] Prousalis, K. & Konofaos, N. A Quantum Pattern Recognition Method for Improving Pairwise Sequence Alignment. *Scientific Reports*. **9**, 1-11 (2019)

[15] Giovannetti, V., Lloyd, S. & Maccone, L. Quantum Random Access Memory. *Phys. Rev. Lett.*. **100**, 160501 (2008,4), https://link.aps.org/doi/10.1103/PhysRevLett.100.160501

[16] Draper, T. Addition on A Quantum Computer. *ArXiv Preprint quant-ph/0008033*. (2000)

[17] Ahuja, A. & Kapoor, S. A Quantum Algorithm for Finding the Maximum. *ArXiv Preprint quant-ph/9911082*. (1999)