Quantum Pattern Recognition for Local Sequence Alignment

Konstantinos Prousalis
Department of Informatics
Aristotle University of Thessaloniki
Thessaloniki, Greece
kprousalis@csd.auth.gr

Nikos Konofaos
Department of Informatics
Aristotle University of Thessaloniki
Thessaloniki, Greece
nkonofao@csd.auth.gr

***Abstract*—Over the last two decades, there have been some challenging proposals on the field of pattern recognition by means of quantum technology. The application of them is considered for a popular branch of bioinformatics which analyzes massive amounts of sequence data for genes and proteins. More specific, the Smith-Waterman algorithm is studied under the more general term of local sequence alignment. The steps of this algorithm are totally reformed and powered by the quantum mechanics computing theory. The proposed method is based on R. Schützhold's pattern recognition quantum algorithm. A binary and unstructured data set is formed after the comparison of the sequences under alignment which is used by R. Schützhold's algorithm to identify and locate potential patterns. It is achieved with the aid of a spatial light modulator. The adopted quantum algorithm exhibits an exponential speed-up in comparison with its classical counterparts.**

Keywords—Smith-Waterman algorithm; pattern recognition; quantum Fourier transform; quantum parallelism

I. INTRODUCTION

In recent years, there has been a growing interest in quantum pattern recognition, since some methods promise to locate and identify certain patterns in unstructured data sets [1-4]. These achievements may meet the needs of several known problems, i.e. in biology, many international endeavors by various organisms implement genome projects generating extremely large databases of biological data that need analysis. Sequence alignment is a popular practice to shed lights on the relations between genes or proteins, leading to a better understanding of their homology and functionality.

The sequence alignment is a process of arranging the sequences of discrete objects to identify regions of similarity. This is an important method especially in bioinformatics for the arrangement of the sequences of protein, DNA, or RNA, that may be a consequence of functional, structural or evolutionary relationships between sequences. It may help to produce valuable information about biological procedures. Similar applications also exist for non-biological sequences, such as quantifying the edit distance cost between two dissimilar strings in a natural language or in financial data.

The common computational approaches are the global and the local alignments. Global alignment tries to span the entire length of all query sequences. All letters and nulls in each sequence must be aligned. In contrast, local alignment identifies regions of similarity within long sequences that are often widely variant. Various algorithms have been proposed to various versions of the sequence alignment problem. Dynamic programming offers slow, but correct methods. Some known examples of dynamic programming are the Needleman-Wunsch [5] and the Smith-Waterman [6] algorithms for global and local alignments, respectively. For large scale databases, heuristic algorithms or probabilistic methods have been fabricated, but they do not guarantee to find the best matches. This work focuses on local sequence alignment and especially on the Smith-Waterman algorithm.

The main problem in local sequence alignment with the Smith-Waterman algorithm is to maintain optimal local alignment for large-scale projects, so alternative means should be employed. Various proposals try to contribute to this problem with the most prevalent those in [1-3]. The main configuration is to adopt a spatial light modulator which maps a single incoming optical mode of one or more photons on thousands outgoing optical modes over the high number of pixels of a complex 2-dimentional image. It can happen in a programmable way and the inverse is also feasible. The unstructured data set is loaded in this image and then a quantum algorithm, like the one in [1], runs to inquire particular patterns. The powerful effects of quantum parallelism and quantum Fourier transform are employed to confront effectively the sequence alignment process [7-8].

The following section, Section II, gives a brief description of a useful quantum algorithm, known as Quantum Fourier Transform (QFT). It is the quantum analogue of the classical Discrete Fourier Transform and is used as a subroutine in some interesting quantum algorithms, such as the Shor algorithm, the quantum phase estimation algorithm, or the hidden subgroup problem. The QFT is necessary to be introduced for the understanding of the rest part. Section III gives a condensed description of the classical Smith-Waterman algorithm and Section IV describes how the solution of the local alignment problem of Smith-Waterman algorithm can be improved drastically with the aid of Schützhold's pattern recognition quantum algorithm. In the last section, Section V, some conclusions are discussed.

978-1-5386-3920-7/17/$31.00 ©2017 IEEE

II. THE QUANTUM FOURIER TRANSFORM

The quantum Fourier transform is a unitary linear operator of the Hilbert space that acts similar to the classical discrete Fourier transform. The conventional notation used is slightly different and the inputs are replaced by the quantum states of the quantum computing system.

A simple quantum register may have n qubits and its basic quantum states -known as the orthonormal basis- are usually represented in the decimal system $|0\rangle$, … a …, $|N\text{-}1\rangle$ where $N=2^n$ and ¨a¨ an intermediate random state. So, when the QFT operator acts on one of the basis's orthonormal states, see Eq. (1), the following transformation takes place:

$$|a\rangle \mapsto \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i jk/N} |k\rangle \qquad (1)$$

However, when the QFT operator acts on a superposition of the basis's states as in Eq. (2), the transformation is evolved as follows:

$$\sum_{a=0}^{N-1} x_a |a\rangle \mapsto \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \sum_{a=0}^{N-1} x_a e^{2\pi i \frac{ak}{N}} |k\rangle \qquad (2)$$

and if the internal sum Σ of the right part of Eq. (2) is substituted by Eq. (3)

$$y_c = \sum_{a=0}^{N-1} x_a e^{2\pi i \frac{ak}{N}} |k\rangle \qquad (3)$$

then the more compact form of Eq. (4) is obtained:

$$\sum_{a=0}^{N-1} x_a |a\rangle \mapsto \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} y_c |k\rangle \qquad (4)$$

By way of black box computing, QFT takes as input a basic state of the orthonormal basis and transforms it into a superposition of all the basic states with each state having the same amplitude, but different phase.

The unitary property ensures that the reversible application of QFT is possible. If the QFT circuit is executed in reverse, then the inverse QFT can be performed on a quantum computer.

III. THE CLASSICAL SMITH-WATERMAN ALGORITHM

Temple F. Smith and Michael S. Waterman are the inventors of this valuable algorithm. Though the Smith-Waterman algorithm was initially proposed in 1981 [6], it still continues to be a significant utility for today's local sequence alignment software tools, since it is incorporated in several known software packages such as FASTA [9], SWIPE [10], or BLAST[11].

This algorithm compares segments of all possible lengths between two strings of nucleic acid sequences, or protein sequences, to determine similar regions. The similarity measure can also be optimized. One of its great properties is that guarantees finding of the optimal local alignment with respect to the scoring system being used (including the substitution matrix and the gap-scoring scheme). In fact this algorithm aligns two sequences by matches/mismatches, insertions, and deletions. Both insertions and deletions are the operations that introduce gaps and try to shape matches which are represented by dashes.

The Smith-Waterman algorithm is broken into three compact and discrete computing steps in order to be described. Initially, some considerations take place before introducing the algorithm. The two sequences to be aligned are symbolized by $A=a_1a_2...a_n$ and $B=b_1b_2...b_m$ and the substitution matrix and the gap penalty scheme are determined as follows:

a. $s(a,b)$ is the function that gives the similarity score of the elements that constitute the two sequences and

b. $W_k = kW_1$ is the function that gives the penalty of a gap that has a length of k. The gap penalty designates scores for insertion or deletion.

Since the substitution matrix and the gap penalty schemes are determined, the three steps are defined:

1) Create a scoring matrix H of size $(n+1)\times(m+1)$ and initialize its first row and first column by setting 0.

2) Fill in the scoring matrix using the Eq. (5):

$$H_{ij} = \max \begin{cases} H_{i-1,j-1} + s(a_i,b_j), \\ \max_{k\geq 1}\{H_{i-k,j} - W_k\}, \\ \max_{\ell\geq 1}\{H_{i,j-\ell} - W_l\}, \\ 0 \end{cases} \qquad (5)$$

where $1 \leq i \leq n$ and $1 \leq j \leq m$.

3) The best local alignment is located by starting at the highest score in H and ending at a matrix cell that has a score of 0 in a recursive or traceback way.

An illustrative example for the alignment of DNA sequences A=CACCGTAA and B=AACCAGTCG is demonstrated. The substitution function gives $s(a_i,b_j)$=+3 when $a_i=b_j$ and -3 when $a_i\neq b_j$. The gap penalty is $W_k=kW_1$ with W_1=2. The matrix in Fig. 1 shows the finished scoring process. The best local alignment is generated in the reverse direction of the highlighted numbers.

		C	A	C	C	G	T	A	A
	0	0	0	0	0	0	0	0	0
A	0	0	3	1	0	0	0	3	3
A	0	0	3	1	0	0	0	3	6
C	0	3	1	6	4	2	0	1	4
C	0	3	1	4	9	7	5	3	2
A	0	1	6	4	7	6	4	8	6
G	0	0	4	3	5	10	8	6	5
T	0	0	2	1	3	8	13	11	9
C	0	3	1	5	4	6	11	10	8
G	0	1	0	3	2	7	9	8	7

3	6	9	7	10	13	
A	C	C	-	G	T	
A	C	C	A	G	T	

Fig. 1. The scoring matrix with a marked traceback result

The Smith-Waterman algorithm has a cubic computational complexity in time and a quadratic complexity in space. Large-scale problems often cannot be practically confronted and are replaced by computationally more efficient alternatives as Gotoh proposed in 1982 [12], Altschul and Erickson proposed in 1986 [13], and Myers and Miller proposed in 1988 [14]. However, the generality of all these algorithms is reduced.

IV. Quantum Smith-Waterman Algorithm

The main concept is to create an unstructured picture of black and white cells with the aid of the substitution matrix mentioned in the Smith-Waterman algorithm and then apply the Schützhold's quantum algorithm parameterized in such a way in order to recognize and locate the desired regions.

A. Formulation of the problem

The first task is to set properly the rectangular $n \times m$ array, following the substitution matrix mentioned in step 1 of the Smith-Waterman algorithm. The rectangular array has unit cells which are either absorptive (black) or reflective/transparent (white), assuming that it adopts a perfect or tolerant behavior in absorption or reflection. The white cells will be named as *points*. If they have a homogeneous density ρ, then the array contains $\rho.(n.m)$ points.

The patterns are supposed to be approximately resilient under at least two symmetry transformations into different directions. The position of all points may be located by shining appropriately focused light beams on the array and measuring the reflection or transmission.

The construction of the array should have white cells only in the points where a match or crossover occurs between the two sequences under alignment. The same example with the previous alignment between *A* and *B* sequences is considered and the array should look like the matrix in Fig. 2.

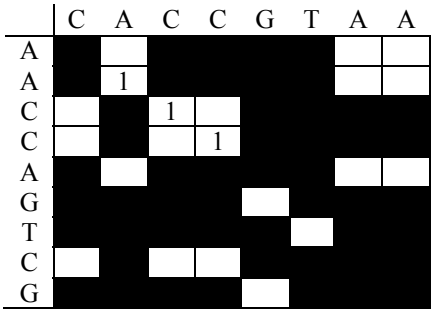

Fig. 2. The substitution matrix of the Smith-Waterman algorithm converted into an array with white and black cells.

The adopted trick is to recognize and locate the longest possible diagonal lines in the picture. These diagonal lines seem to securely indicate potential regions of similarities among sequences. Diagonal lines may not be perfectly formed in the picture, but can be good indicators for potential local alignments. The longest diagonal line in Figure 2 is marked with 1's. Further operations may detect neighboring diagonal lines. So, a small fraction x of these points forms a pattern in a connected region. The pattern is considered to be linear for simplicity and not necessarily rectangular, and may not be perfect. Average symmetries are sufficient.

However, one more step should be taken in order to make more effective the application of the quantum algorithm. Since the whole process is confronted as a linear array, it is better to transform the proposed matrix in a way where the diagonals will be horizontal or vertical lines maintaining the rectangular picture. Such a transformation would be time consuming. An alternative solution may be possible by configuring the angle of the spatial light modulator since the adopted technology allows it.

The array data will be viewed as a quantum black box (6). The input state encodes the coordinates x and y of a point in the array as n- and m- qubit strings, respectively. The third one qubit register $|0\rangle$ is used for the output function $f(x,y)$ which returns 1 if there is a point at these coordinates and 0 if not.

$$BB: \begin{bmatrix} |x\rangle \\ |y\rangle \\ |0\rangle \end{bmatrix} \rightarrow \begin{bmatrix} |x\rangle \\ |y\rangle \\ |f(x,y)\rangle \end{bmatrix} \qquad (6)$$

A possible physical realization of this configuration is given in [1] by using quantum optics tools.

B. Applying Schützhold's Quantum Algorithm

Since the array structure is ready, the algorithm is applied to recognize and locate potential diagonal lines in the unstructured data set. The algorithm is organized and presented in three steps:

1) Black-Box run: The Hadamard gate is applied to the input n-qubit and m-qubit $|0\rangle$ states, respectively, and in only one run of the black box all possible values of the coordinates (x,y) are inquired with the powerful quantum parallelism effect. The desired superposition is produced in Eq. (7).

$$BB\begin{bmatrix} H^{(n)}|0^{(n)}\rangle \\ H^{(m)}|0^{(m)}\rangle \\ |0\rangle \end{bmatrix} = \frac{1}{\sqrt{NM}} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \begin{bmatrix} |x\rangle \\ |y\rangle \\ |f(x,y)\rangle \end{bmatrix} \qquad (7)$$

2) Superposition by measurement: A measurement is applied on the third register. If the result is 1, the state $|\Psi\rangle$ is prepared as a superposition of the coordinates $|x\rangle$ and $|y\rangle$ of all the points (white cells), see Eq. (8). Otherwise, it is implied that the complementary set is prepared by assuming that an ideal black box was adopted. By measuring $f = 1$, then it is considered to reorganize the array by dividing it into M rows of length N and combining them all to one string of length $S=NM$. The coordinate of a given point can be expressed as a $n + m = s$-digit binary number $z = x + Ny$. The corresponding quantum state is given by $|z\rangle = |x\rangle \otimes |y\rangle$. The superposition state $|\Psi\rangle$ is prepared as:

$$|\Psi\rangle = \frac{1}{\sqrt{\rho S}} \sum_{\ell=1}^{\rho S} |z_\ell\rangle \qquad (8)$$

where $0 \le z_\ell \le S\text{-}1$ denotes the position of ℓ-th point.

3) QFT application: The QFT is applied to the basis element $|z\rangle$, see Eq. 9, and the superposition state $|\Psi\rangle$ will be transformed in the following form:

$$QFT|\Psi\rangle = \sum_{k=0}^{S-1}\sum_{\ell=1}^{\rho S}\frac{1}{S\sqrt{\rho}}\exp(2\pi i\frac{z_\ell k}{S})|k\rangle \qquad (9)$$

Measuring then $|k\rangle$ will obtain useful information about the pattern. This step may lead to peaks of the factor of $|z\rangle$ at certain values of k under a typical length scale of the pattern and reveal the suspected regions. If there is no pattern within the data, the measurement of k yield just noise, except $k = 0$.

So far, the problem of feature selection (detection of a pattern) is accomplished, but for the localization and classification of the pattern more information should be extracted from the peaks in the measurements of k. Once a small amount of measured wave-numbers is obtained, the rest analysis can be accomplished by classical algorithms.

C. Analysis of pattern localisation

Schützhold's localization method in [1] is described for linear and complicated patterns, but in this study we focus on patterns of single lines. A line may not be perfectly shaped to the degree that the density of points within a line-width of, e.g. $D/2$, deviates by an acceptable finite amount $\Delta\rho$ from the mean ρ, in average. The key parameters are the length L of the line and its deviation angle $-\pi/2 \le \vartheta \le \pi/2$ from a vertical one. The values of L and ϑ can be inferred from the location of the peaks. So, the points z marking the center of a particular line are given by

$$z = z_0 + [\mathbf{N}\,(N + \tan\vartheta)]_{\text{integer}} \qquad (10)$$

Every row of the pattern generates peaks at

$$k = [\,\mathbf{N}\cos\vartheta\frac{S}{D} \pm O(\frac{M}{D\sqrt{\chi}})]_{\text{integer}} \qquad (11)$$

and the sum of all rows interferes constructively only if k is fine-tuned according to

$$k = [\,\mathbf{N}\cos\vartheta\frac{N - \tan\vartheta}{N} \pm O(\frac{1}{\sqrt{\chi}})]_{\text{integer}} \qquad (12)$$

In both Eq. (11) and Eq. (12), the second term denotes the width of the peak, while the position and the width of the peaks can be obtained from the associated Laue function $f_{Laue}(\xi,\kappa) = \sin^2(\pi\xi\kappa)/\sin^2(\pi\kappa)$, with $k = \kappa\cos\vartheta\, S/D$ and $\xi = O(N\sqrt{\chi})$ for Eq. (10) and $\kappa = k(N+\tan\vartheta)/S$ and $\xi = O(M\sqrt{\chi})$ for Eq. (11).

The most predominant peaks in the measurements of k occur for values which satisfy both conditions Eq. (11) and Eq. (12), concurrently. Thus, the wave-numbers of the potential peaks are interpreted as

$$k \approx [\,\mathbf{N}\cos\vartheta\frac{S}{D} \pm \sin\vartheta\frac{M}{D}\,]_{\text{integer}} \qquad (13)$$

where the corresponding width is omitted. However, Eq. (13) doesn't represents necessarily large peaks. The first few of them may be suppressed, but some potential peaks from Eq. (13) may match both conditions Eq. (11) and Eq. (12). Consequently, the same process has to be repeated for the transposed array (NM→)MN. The weave-numbers of the peaks are now

$$k' \approx [\,\mathbf{N}\sin\vartheta\frac{S}{D} \pm \cos\vartheta\frac{N}{D}\,]_{\text{integer}} \qquad (14)$$

since transposing changes the array orientation per $\pi/2$.

D and ϑ candidate values can be approximated by combining the possible values for $D/\cos\vartheta$ from Eq. (13) with the ones for $D/\sin\vartheta$ from Eq. (14). Moreover, the comparison with the conditions Eq. (11) and Eq. (12) and the sets of the suppressed peaks will enable to extract the D and ϑ values with high precision. Now, the size of the pattern χ is determined by the frequency of measuring the peaks at k and k' and their width. Knowing L, θ and χ the pattern can be localized easily by splitting up the total S array into smaller fragments and running the same quantum algorithm again in the smaller domains.

V. Complexity Analysis

The complexity of the Smith-Waterman algorithm can be estimated following the three steps presented in Section III. To summarize, the computing process is divided into three phases: the initialization, the filling of the matrix and the traceback. The time complexity of the initialization is $O(M+N)$ because it is needed to set zero all the rows and the columns. During the filling of the matrix phase each cell is traversed and a constant number of operations is performed in each cell. The time complexity for this part is $O(MN)$. In the traceback, it is required to find the maximum cell by traversing the entire matrix, making the time complexity for the traceback $O(MN)$. Thus the total time complexity of the Smith-Waterman algorithm is $O(M+N) + O(MN) + O(MN) = O(MN)$.

However, in reality, due to the adopted gap penalty system, gaps of different sizes would all have different penalties. When computing the score of each cell, instead of finding the maximum of three adjacent cells, the number of cells to the right or down which also are included in the gap have to be found. Thus, it increases the time complexity to $O(M^2N)$. Since this algorithm fills a single matrix of size MN and stores at most N positions for the traceback, the total space complexity of this algorithm is $O(MN)+O(N)=O(MN)$.

Given the physical realization of the black box proposed in [1], it is possible to quantify the total number of fundamental manipulations. The adopted architecture uses a focused light beam which passes $n+m$ controlled refractors, made by a non-linear Kerr media, which change its direction by definite angles φ_j (with $j\in\bullet$) if the control qubit is $|1\rangle$. Based on this fabrication, the preparation of the initial state in Eq. (7) takes $\log_2 S$ times since the Hadamard gate is applied s times. The black box itself takes $\log_2 S$ times, too. The QFT in Eq. (6) requires $O(\log_2^2 S)$ steps for obtaining the exact result and is even faster $O(\log_2 S)$ if we measure the outcome immediately afterwards. So, the size of the proposed algorithm in the limit $S \to \infty$ while ρ, $\Delta\rho$ and x remain finite is estimated only to a few $O(S^0)$ queries of the black box in order to find a pattern of

a given size with high probability. However, in the case that the complete data set is loaded into a quantum memory may represent a drawback as it would involve about $O(S)$ operations.

Even adopting the classical version of pattern recognition, the fast Fourier transform implements $O(S\log_2 S)$ operations which is exponentially slower.

VI. Conclusion

A sophisticated matrix completion and presentation of the matches between two sequences of data elements allows the local sequence alignment to be accomplished in a more efficient way even for large scale sequences. A comparison complexity analysis with the Smith-Waterman algorithm showed that the proposed method outperforms the classical Smith-Waterman algorithm. The QFT solved efficiently the problem of feature selection by extracting the relevant quantities from the unstructured data set, but the necessity to load the complete data set into a quantum memory remains a problem.

References

[1] R. Schützhold, "Pattern recognition on a quantum computer," Phys. Rev. A., vol. 67, 062311, 2002.

[2] C.A. Trugenberger, "Quantum pattern recognition," Quantum Information Processing, vol.1 iss. 6, pp. 471-493, 2002.

[3] P. W. H. Pinkse, S. A. Goorden, M. Horstmann, B. Škorić, and A. P. Mosk, "Quantum pattern recognition," In 2013 Conference on and International Quantum Electronics Conference Lasers and Electro-Optics Europe (CLEO EUROPE,/IQEC), pp.1, IEEE, Munich, Germany, 12-16 May 2013.

[4] R. Zhou, Q. Ding, "Quantum pattern recognition with probability 100%," International Journal of Theoretic Physics, vol. 47, iss. 5, pp.1278-1285, 2008.

[5] S.B. Needleman, C.D. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins," Journal of Molecular Biology, vol. 48, pp. 443–453, 1970.

[6] T.F. Smith and M.S. Waterman, "Identification of common molecular subsequences," J. Mol. Biol., vol. 147, pp. 195-197, 1981.

[7] M.A. Nielsen and I. L. Chuang (2000), Quantum Computation and Quantum Information*, Cambridge University Press (Cambridge)*.

[8] L. Hales, S. Hallgren, "An improved quantum Fourier transform algorithm and applications," Proceedings of the 41st Annual Symposium on Foundations of Computer Science, pp. 515 – 525, 12-14 Nov. 2000.

[9] D.J. Lipman, W.R. Pearson, "Rapid and sensitive protein similarity searches," Science. vol. 227, iss. 4693, pp. 1435–41, 1985.

[10] Rognes Torbjorn, "Faster Smith–Waterman database searches with inter-sequence SIMD parallelisation," BMC Bioinformatics, vol. 12, iss. 221, 2011.

[11] S. Altschul, W. Gish, W. Miller, E. Myers, D. Lipman, "Basic local alignment search tool," Journal of Molecular Biology., Vol. 215, iss. 3, pp. 403–410, 1990.

[12] Osamu Gotoh, "An improved algorithm for matching biological sequences," J. Mol. Biol., vol 162, pp. 705-708, 1982.

[13] S.F. Altschul and B.W. Erickson, "Optimal sequence alignment using affine gap costs," Bulletin of Mathematical Biology, vol. 48, pp. 603-616, 1986.

[14] M. Webb and E. Miller, "Optimal alignments in linear space," Computer applications in the biosciences, vol. 4, pp.11-17, 1988.