Parallel Smith-Waterman Algorithm for Gene Sequencing

Deepa. B. C[1], Nagaveni. V[2]

[1]Student, Department of CSE, Acharya Institute of Technology, Bangalore, Karnataka, India

[2]Assistant professor, Department of CSE, Acharya Institute of Technology, Bangalore, Karnataka India

Abstract: Smith-Waterman Algorithm represents a highly robust and efficient parallel computing system development for biological gene sequence. The research work here gives a deep understanding and knowledge transfer about exiting approach for gene sequencing and alignment using Smith-waterman their strength and weaknesses. Smith-Waterman algorithm calculates the local alignment of two given sequences used to identify similar RNA, DNA and protein segments. To identify the enhanced local alignments of biological gene pairs Smith-Waterman algorithm uses dynamic programming approach. It is proficient in finding the optimal local alignment considering the given scoring system.

Keywords: *Bioinformatics, Smith-waterman, Needleman-Wunsch Algorithm, Smith-Waterman algorithm, Dynamic programming.*

*****

I. Introduction

Gene sequencing problem are one of the major issues for researchers regarding with optimized system model that could help optimum processing and efficiency without introduction overheads in terms of memory and time. Bioinformatics and computational biology is a latest multidisciplinary field which explains many aspects of the fields of computer science, while computational biology harnesses computational approach and technologies to respond biological questions conveniently. In the present days scenario the approaches of genomics have played a vital role in optimizing parallel dealing out systems. Genome is an emerging field, constantly presenting many new challenges to researchers in both biological and computational aspect of application. Sequence comparison is a very essential and important operation in Bioinformatics. Sequence alignment algorithms detect similar or identical parts between two sequences called the query sequence and the reference sequence. The global and local alignments are the most prevalent kinds of sequence alignment. In global alignment problem finds the superior counterpart between the whole sequences. On the other hand, local alignment algorithms must find the superior counterpart between parts of the sequences. Gene sequences consider the order of DNA nucleotides, in a genome, the order of Cytosine, Adenine, Guanine, and Thymine that form an organism's DNA. The human gene is made up of more than three billion of these genetic letters. A gene sequence does contain some clues about wherever genes are, even if scientists be just learning to infer these clues. The study of the entire genome sequence will help to understand how the genome as a whole works, how genes work together to increase, growth and maintenance of a whole organism. Taking into account that sequences can have up to 109 nucleotides each, the time and memory required to solve this problem in a sequential manner is unfeasible. This guide to the parallelization of the algorithm based on powerful parallel architectures.

II. Literature Survey

A. Sequence comparison

The most challenging and important tasks in bioinformatics is the sequence database searching from [3], where the fast growing amounts to genetic-sequence information available represents a regular challenge to developers of software and hardware database searching and management. The size of EMBL/DDBJ nucleotide database has been doubling every 15 months. The rapid growth of the genetic sequence information is possibly exceeding the growth in computer power in spite of the fact that computing resource also have been increasing exponentially for many years. When looking for sequences in a database similarly to a given analysis sequence, the search program calculate an alignment score for every database. Using the optimal algorithm for database searches is quite slow.

B. Methods for sequence alignment

The basic and most important work in Bioinformatics field is scanning biological sequence database and finding similarity among protein and DNA sequences. To resolve this, Needleman-Wunsch (NW) algorithm has been implemented. Needleman-Wunsch algorithm include comparison between two entire sequences, hence processing time becomes insupportable due to exponential growth speed and large amount of biological sequence database. To overcome from this problem, a reconfigurable accelerator for Smith-Waterman algorithm is presented from [1], where in the accelerator, a modified equation is projected to develop mapping efficiency of a processing which provides more than 330 speedup when compared to a standard desktop platform with the 4GB memory and 2.8GHz Xeon processor and it has a 50% progress on the peak performance of a traditional implementation without the two special techniques.

C. Space and Time Optimal Parallel Sequencing

The space and time optimal parallel algorithm used for the pairwise sequence alignment problem from [14], where this problem in computational Bioinformatics can be solved by $O(mn)$ time and $O(m+n)$ space sequentially, where m and n are the lengths of the sequences to be aligned. Pairwise sequence alignment uses the parallel space optimal algorithm which takes optimal $O(m+n/p)$ space and sub-optimal $O((m+n)^2/p)$, where p is the total number of processors. Alternatively, the mainly time optimal and space economical takes $O(m+n/p)$ space, but $O(mn/p)$ time. To achieve both time and space optimally this gap can be closed by presenting an algorithm which requires only *O(mn/p)* time and *O((m+n)/p)* space. This algorithm is also used for other alignment problems in computational biology including synthetic alignment and local alignments**.**

D. Multiple Sequence Alignment

Multiple sequence alignment presents a dynamic programming solution to simulate copies of automaton from [10], where composed a weighted finite automaton from a specified regular expression in looking for data sequencing with maximum score comprised of the usual expressions. The researchers simplify this approach: 1) Introduce a differentiation of the problem for multiple sequences, stated by the standard expression inhibited multiple sequence alignment. 2) Developed a robust technique for the situation when the sequential alignments needs are advocated for containing certain query sequence of standard expressions.

E. Regular Expression Constrain Multiple Sequence Alignment

Smith-Waterman algorithm is a classical algorithm for pairwise sequence alignment. In this paper [11], introduce a dissimilarity of the problem for multiple sequences, specifically the regular expression constrain multiple sequence alignment, and presented an algorithm for it. For the case of the problem when the alignments are required to contain a given sequence of regular expression this algorithm has been developed.

F. Performance Improvement Of The Smith-Waterman Algorithm

In this paper [15], aims at providing a new approach to improve the performance of the Smith-Waterman using partially custom hardware. In this approach customized hardware is used to accelerate the computationally intensive part of the algorithm. Rather than implementing the entire algorithm in hardware. Implementing specific small part of the algorithm in hardware results in a 35.82 times speedup relative to its software equivalent.

G. Aligning Two Sequences Within A Specified Diagonal Band

Aligning two sequences within a specified diagonal band it requires only O(NW) computational time and *O(N)* gap [5], where *N* is the length of the two sequences and W is the size of the band. The local and global algorithms can be used to calculate the scores. Local alignments are formed by finding the beginning and end of a best local alignment in the band and followed by global alignment algorithm between those points. This algorithm has been produced into FASTA program, it has decreased the amount of memory which is required to calculate local alignment from O(NW) to O(N) and it has also decreased the time which is required for every sequence to calculate optimized scores in a protein sequence database by 40%.

H. Global Alignment

The ruling global alignment algorithm is the Needleman-Wunsch algorithm. This approach finds an optimal global alignment between two DNA sequences (database and query sequence) over their total length. These two sequences are associated with each other so that matches and mismatches are easy to identify. The maximum match is a number dependent upon the similarity of the sequences. By considering the smallest unit of consequence comparisons are made, a pair of amino acids, one from every protein. The global approach appeared to be insufficient for the new methods of DNA analysis, where smaller sequences were compared to one large sequence. Global alignment is hardly used nowadays for this reason.

I. Parallel Implementation of Smith-Waterman Algorithm

In this paper [4], a parallel execution methodology of the Smith-Waterman algorithm is obtainable. This method provide magnificent speedup more than the traditional sequential implementation, while retain the same level of sensitivity. This approach reduce the time complexity from *O(mn)* to *O(m+n)* for a single sequence comparison and from *O(mnk)* to *O(m+nk)* for multiple sequence pairs comparison. This highlights the urgent needs of cost-effective and efficient comparison mechanism. The parallel implementation methodology presented here provides the framework of reducing time complexity, while maintaining the same level of sensitivity for this important task.

III. PROPOSED ALGORITHM (SMITH-WATERMAN ALGORITHM)

Smith-Waterman algorithm calculates the local alignment of two sequences. It guarantees to find out the best possible local alignment taking into account the specified scoring system. This includes a substitution matrix and a gap-scoring method. Scores consider match, mismatch and substitution. To measure the comparison between two sequences, a score be calculate as follows: given an alignment between sequences S_0 and S, the following values be assigned, for each column:

1) *ma=+5* (match);
2) *mi= -3* (mismatch);
3) *G= -4* (gap).

Gap score or gap penalty

Dynamic programming algorithm applies gap penalties to maximize the biological importance. Gap penalty be subtract for each gap that has been introduced. When there is a insertion or deletion the gap score define a penalty given to alignment. By introducing and extending gaps into one of both sequences, more optimal alignment can be obtained.

Assumed scoring schemas

In both the sequences if the nucleotide or amino acids are same then the match score is assumed $(S_{i,\ j})$ as +5. It added to the diagonally located cell of the current cell such as *i, j* position. Suppose the residues are not same, the mismatch score is taken as -3. This score is added to the diagonally positioned cell of the existing cell. The gap penalty score is taken as -4 and this score is added to the left and above positioned cells of the current cell. These scores are not exceptional, they can be user define too, however the gap penalty and mismatch value should be negative values.

Procedure of the Algorithm

i. Initialization of matrix and consider the two sequences A and B.
ii. Matrix filling with the suitable scores. The two sequences are set in a matrix form by means of A+ 1

column and B+1 row. The value in the first row and first column are set to zero.

$$M_{i,j} = Max \begin{cases} M_{i-1,j-1} + S_{i,j}, \\ M_{i,j-1}+W, \\ M_{i-1,j} + W, \\ 0 \end{cases}$$

The second and essential step of the algorithm is filling the entire matrix. To fill each and every cell it is important to know the diagonal values.

iii. Trace back the sequence for an appropriate alignment. The final step for the proper alignment is trace backing; before that the maximum score obtained in the entire matrix has to be detected for the local alignment of the sequences.
It is likely to those maximum scores can be present in one or more than one cell**,** in such case there may be option of two or more alignments, and the best alignment can be obtained by scoring it.

Example

Consider the sequences: CGTGAATTCAG and GACTTAC with the proposed algorithm the table will begin to fill from the position (1, 1), the first entry in the first row.

	-	C	G	T	G	A	A	T	T	C	A	G
-	0	0	0	0	0	0	0	0	0	0	0	0
G	0											
A	0											
C	0											
T	0											
T	0											
A	0											
C	0											

Fig.1 Initialization of matrix

Variable used:

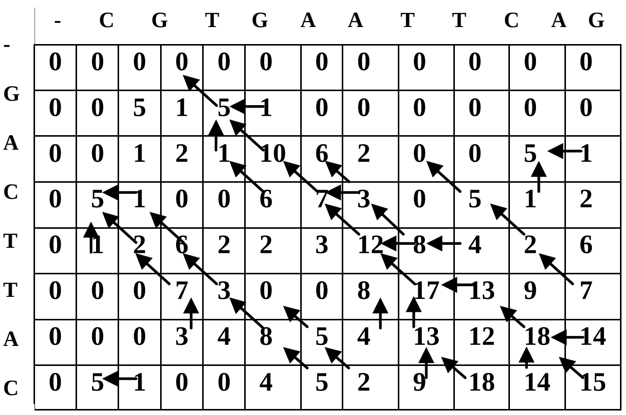
i, j describe row and column.
M is the matrix value of the essential cell
S is the score of the cell *($s_{i,j}$)*
W is the gap alignment

Matrix filling

The first residue in both the sequence is 'C' and 'G', the matching and mismatching score is added to the neighboring value which is located diagonally.
The score schema equation can be shown as follows:

M_{11} = Max [$M_{0,0}$ + $S_{1,1}$, $M_{1,0}$+ W, $M_{0,\ 1+}$W, 0]
= Max [0(-3), 0 + (-4), 0 + (-4), 0]
= Max [-3, -4, -4, 0]
= 0

After filling the matrix, keep the pointer back to the cell from where the maximum score is obtained. In the similar way all the values of the matrix of the cell is to be filled.



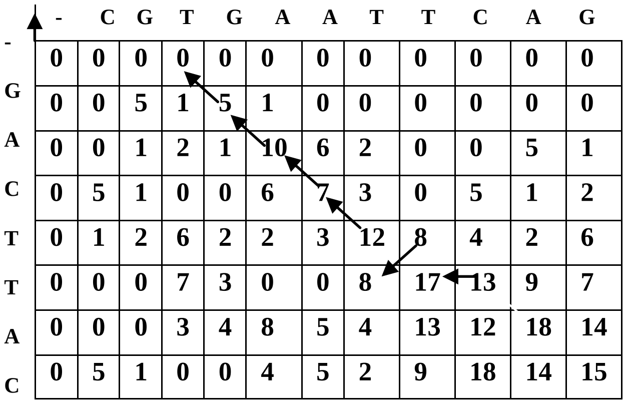
	-	C	G	T	G	A	A	T	T	C	A	G
-	0	0	0	0	0	0	0	0	0	0	0	0
G	0	0	5	1	5	1	0	0	0	0	0	0
A	0	0	1	2	1	10	6	2	0	0	5	1
C	0	5	1	0	0	6	7	3	0	5	1	2
T	0	1	2	6	2	2	3	12	8	4	2	6
T	0	0	0	7	3	0	0	8	17	13	9	7
A	0	0	0	3	4	8	5	4	13	12	18	14
C	0	5	1	0	0	4	5	2	9	18	14	15

Fig.2 matrix filling with back pointers

With one or more pointers every cell is back pointed from where the maximum score is obtained.

Trace Backing the Sequence for an Optimal Alignment

The maximum score in the matrix is 18. So the trace back begin from the position which has the highest value, pointing back with the pointers, consequently find out the possible predecessor, then go to next predecessor and continue until it reach the score 0.



	-	C	G	T	G	A	A	T	T	C	A	G
-	0	0	0	0	0	0	0	0	0	0	0	0
G	0	0	5	1	5	1	0	0	0	0	0	0
A	0	0	1	2	1	10	6	2	0	0	5	1
C	0	5	1	0	0	6	7	3	0	5	1	2
T	0	1	2	6	2	2	3	12	8	4	2	6
T	0	0	0	7	3	0	0	8	17	13	9	7
A	0	0	0	3	4	8	5	4	13	12	18	14
C	0	5	1	0	0	4	5	2	9	18	14	15

Fig.3 Trace back of possible Alignment

As a result a local alignment is obtained and the possible alignment is:

```
G  A  A  T  T C  A
|  |  |  |  |    |
G  A  A  T  T  -  A
+  +  -  +  +  -  +
5  5  3  5  5  4  5
```

Fig.4 scoring for best alignment

The best alignment is arranged with a score, for matching as +5, mismatch as -3 and gap penalty as -4, by summing up the score it giving the equivalent as 18, so one can predict that it is a best alignment.

Problem Statement

Genome sequencing problems are one of the main issues for researchers to develop an optimized system model that could facilitate the optimum processing and efficiency without affecting the performance of memory and time. This study is oriented towards developing such type of system while taking into consideration of the dynamic programming approach called a Smith-Waterman algorithm.

Proposed Solution

The predominant factor which has been optimized with its optimum possibility is Smith-Waterman algorithm which functions in a unique parallel alignment rather being in conventional serial approaches. Smith-Waterman algorithm is a dynamic programming approach that could accomplish the higher rate sequencing with parallel scheme. The dynamic programming approach uses a table or matrix to preserve values and avoid re-computation.

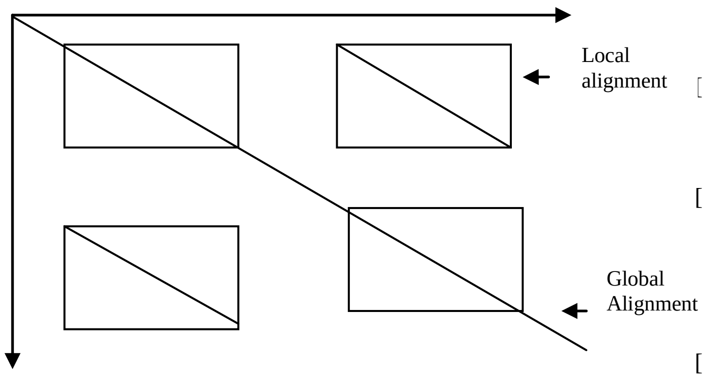


Fig.4 Comparison between Global and Local Alignments.

Local alignment are performed everywhere, in every direction. When trying to find the best local alignments corresponding to a global alignment, a sub matrix is created with highest positive score for all alignments above.

Conclusion

In this paper, we have discussed different approaches such as Needleman-Wunsch algorithm and Smith-Waterman algorithm. The Smith-Waterman algorithm presents a kind of dynamic programming approach for identifying the enhanced local sequencing alignments of biological gene pairs. The local alignment reduces the running time and increases accuracy of the sequence matching within two sequences. The parallel programming approach is a basic process in Bioinformatics for biological sequence database and sequence alignment. The improvement of parallel sequencing reduces the system memory and time.

References

[1] P. Zhang, X. Liu, N. Sun and X. Jiang, “A Reconfigurable Accelerator for Smith-Waterman Algorithm,” IEEE Trans. Circuits and Systems, vol. 54, no. 12, pp. 1077-1081, Dec. 2007.

[2] De Giusti Armado.E, Rucci Enzo, “Parallel Smith-Waterman Algorithm for DNA Sequences Comparison on different Cluster Architectures” {erucci, degiusti, HUUUfrancoch}@lidi.info.unlp.edu.arUH Universidad nacional de la Plata. Argentina.

[3] J.M.Correa, R.P.Jacobi, Boukerche, A.C.M.A. de Melo, and A.F.Rocha, “Reconfigurable Architecture for Biological Sequence Comparison in Reduced Memory Space”, Proc. IEEE Int’l parallel and Distributed Processing Symp. (IPDPS), pp. 1-8, 2007.

[4] Meng-Lai Yin, Hsien-yu Liao, Cheng.Y, “A parallel Implementation of the Smith-Waterman Algorithm for massive sequences searching”, Engineering in Medicine and Biology Society, 26th Annual International Conference of the IEEE, vol.2, pp.2817, 2820, 1-5 Sept. 2004.

[5] William R.Pearson, Kun-Mao Chao and Webb Miller, “ Aligning two sequences within a specified diagonal band”, Department of computer science, university park, PA 16802 and department of biochemistry, university of virgina.

[6] Abdullah.R Rashid, N.A.A, Talib, A.Z.H, Ali.Z, “Fast Dynamic Programming Based Sequence Alignment Algorithm”, Distributed Frameworks for Multimedia Applications, 2006. The 2nd International Conference on vol-no-pp.1,7,May 2006.

[7] Al-Ars.Z, Hasan.L, "An efficient and high performance linear recursive variable expansion implementation of the smith-waterman algorithm," Engineering in Medicine and Biology Society, Annual International Conference of the IEEE3-6 Sept. 2009; pp.3845-3848.

[8] S. Needleman and C. Wunch, “A general method applicable to the search for similarities in the amino acid sequences of two proteins," Journal of Molecular Biology, vol. 48, no. 3, pp. 443{453, 1970.

[9] W.Miller and X.Huang, “A time-efficient linear-space local similarity algorithm,” Adv.Appl.Math” vol. 12, no. 3, pp. 337{357, 1991.

] F. Guinand, “Parallelism for computational molecular biology," in ISThmus 2000 Conference on Research and Development for the Information Society, Poznan, Poland, 2000.

[11] Arslan A.N; "Multiple Sequence Alignment Containing a Sequence of Regular Expressions"; Computational Intelligence in Bioinformatics and Computational Biology, 2005. Proceedings of the IEEE Symposium on. 14-15 Nov. 2005, pp.1-7.

[12] Matthijs. Geers, Faith Han Caglayan, Roelof Willem Heij, “low-cost smith waterman acceleration” DELFT University of technology august 2013.

[13] Brian Hang; Wai Yang; “A Parallel Implementation of Smith-Waterman Sequence Comparison Algorithm”; December 6, 2002.

[14] S. Rajko and S. Aluru, “Space and Time Optimal Parallel Sequence Alignments,” IEEE Trans. Parallel Distributed Systems, vol. 15, no. 12, pp. 1070-1081, Dec. 2004.

[15] Laiq Hasan, Zaid Al-Ars, “Performance improvement of the Smith-Waterman Algorithm” Delft University of technology computer Engineering Laboratory.