ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 8, August 2015

Gene Sequencing Parallelization Using Smith-Waterman Algorithm

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

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

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

Abstract: Genomics is the determination of the complete DNA sequence of an organism. The objective of modern human genomics is preventive, predictive and individual medicine. In agriculture, the goal is the production of foods with improved production characteristics and increasingly beneficial consumer traits. Post-genomics refers to the biological processes that follow from DNA sequence. Efficient sequence alignment is one of the most important and challenging activities in bioinformatics. To perform and accelerate sequence alignment activities several algorithms have been proposed. 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 existing approach for gene sequencing and alignment using Smith-Waterman algorithm their strength and weaknesses. Smith-Waterman algorithm calculates the local alignment of two given sequences used to identify similar RNA, DNA nucleotides. 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 systems.

Keywords: Smith-Waterman, Dynamic Programming, Smith-Waterman Algorithm (SW), Needleman-Wunsch Algorithm (NW).

I. INTRODUCTION

Gene sequencing problem are one of the major issues for researchers to come up with optimized system model that could facilitate optimum processing and efficiency without introduction overheads in terms of memory and time [1]. Bioinformatics and computational biology is a new multidisciplinary field which brings together many aspects of the fields of computer science, while computational biology harnesses computational approaches and technologies to answer biological questions conveniently [1]. In the present days scenario the approaches of genomics have played a vital role in optimizing parallel processing systems. Genomics is an emerging field, constantly presenting many new challenges to researches in both biological and computational aspects of applications. Sequence comparison is an essential and basic operation in Bioinformatics.

Sequence alignment algorithms distinguish comparable or indistinguishable parts between two sequences called the reference sequence and query sequence. The local and global sequence alignments are the most predominant sorts of alignment of sequence. In global alignment issue finds the prevalent counterpart between the entire sequences. Then again**,** local sequence alignment algorithm must find the superior counterpart among parts of the sequences. Gene sequencing consider the organize of DNA nucleotides, or bases, in a genome the order of cytosine, adenine, guanine and thymine that forms an organism DNA. The human gene consists of more than three billion of these genetic letters.

A gene sequence contains a small number of hints about where genes are, notwithstanding the way that researchers are just figuring out how to interpret these intimation. Researchers the same trust that having the ability to study the whole genome sequencing will help them to perceive how the genome all in all functions, how genes work together to direct the development, advancement, upkeep of a whole organism. Considering that sequences can have up to 109 nucleotides each, the time and memory required to solve this problem in a sequential manner is impracticable.

This leads to the parallelization of the algorithm based on powerful parallel architectures. Sequence alignment is a method of arranging the sequences of RNA, DNA amino acids of proteins to find region of match which may be an outcome of efficient, structural relationships among the sequences. To compare the sequence quantify the similarity or dissimilarity between two or more sequences and find out where they are similar or different. Sequence comparison and alignment is an efficient and basic operation in Bioinformatics. There are generally two methods of sequence alignment they are global alignment and local Alignment. Global Alignment compares two sequences throughout their lengths. The sequences must be aligned from start to end. Local Alignment identify region of high match in sequences which are often widely different overall.

Dynamic programming is used for optimal alignment of two sequences. It finds the alignment in a more quantitative way by giving some scores for matches and mismatches (Scoring matrices), rather than only applying dots. By searching the highest scores in the matrix, alignment can be accurately obtained. The Dynamic Programming solves the original problem by dividing the

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 8, August 2015

problem into smaller independent sub problems. These techniques are used in many different aspects of computer science. Needleman-Wunsch and Smith-Waterman algorithms for sequence alignment are defined by dynamic programming approach.

Smith Waterman algorithm was first proposed by Temple F. Smith and Michael S. Waterman in 1981. The algorithm explains the local sequence alignment, it gives conserved regions between the two sequences, and one can align two moderately overlap sequences, and it is possible to align the sub-sequence of the sequence to itself. These are the major advantages of alignment of Local Sequence Alignment. This algorithm mainly differs within two aspects from the Needleman-Wunsch algorithm. initial part being, local alignment differ by having only a negative score for the mismatch and when the matrix value becomes negative it has to be set to zero (need to take maximum value of the scorings compared by means of zero). They are predefined scoring matrix for nucleotide sequences.

In this paper the presented work emphasizes on the development of a highly robust parallelization approach for gene sequencing and thus coming up with a highly efficient parallel computing approach that could facilitate optimum performance, here it is required to have a complete knowledge transfer about the normal processing and its relation with parallel architecture and computing strategies. Therefore, in this section a brief of these systems has been provided. To study gene function and regulation, nucleic acids.

II. METHODOLOGY

The development of a highly robust parallelization approach for gene sequencing and thus coming up with a highly efficient parallel computing approach that could facilitate optimum performance, here it is required to have a complete knowledge transfer about the normal processing and its relation with parallel architecture and computing strategies.

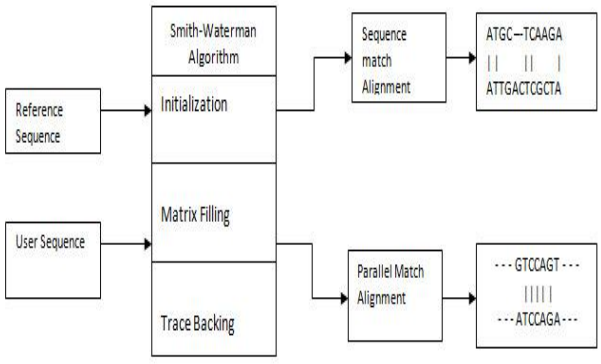

Fig.1 System Architecture

Fig.1 shows the structural behaviour of system. In this architecture, the reference sequence and user sequence are given as input, to the Smith-Waterman algorithm. The Smith-Waterman algorithm calculates the maximum score and time taken for the alignment of both sequential alignment and parallel alignment.

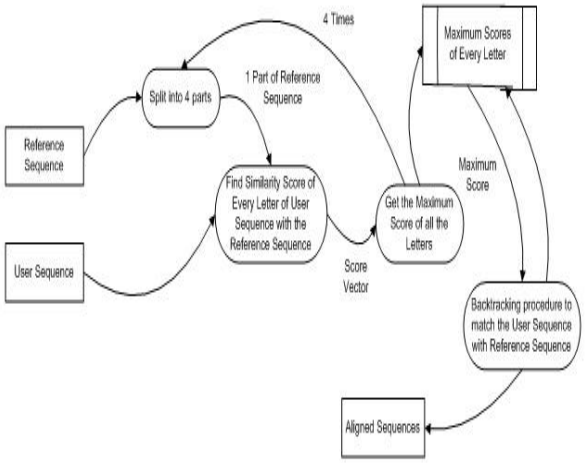

Fig.2 Data Flow Diagram

The inputs of the system are reference sequence and user sequence, where each sequence is calculated with respect to sequential alignment and parallel alignment. The length of the sequence is calculated. In sequential alignment the reference sequence is aligned with user sequence where the alignment is done with initial sequence, where the score and time is calculated. After sequence alignment the parallel alignment is performed. In parallel alignment the input sequence is divided into four parts and the similarity score is calculated of each part. Finally the maximum score and time is displayed. The final result is sequence alignment with maximum score and time consumption.

Proposed system

The Smith-Waterman algorithm is a database search algorithm developed by T.F. Smith and M.S. Waterman, and based on an earlier model appropriately named Needleman and Wunsch after its original creators. The S-W Algorithm implements a technique called dynamic programming, which takes alignments of any length, at any location, in any sequence, and determines whether an optimal alignment can be found. Based on these calculations, scores or weights are assigned to each character-to-character comparison: positive for exact matches/substitutions, negative for insertions/deletions. In weight matrices, scores are added together and the highest scoring alignment is reported.

Dynamic programming can be rather illustrated in a comfortable way. Dynamic programming applies itself on smaller chunks of problem and then puts them all together to form a integrative and absolute final solution to the entire problem.

It is superior to the BLAST and FASTA algorithms because it searches a maximum amount of possibilities, making it a more sophisticated and sensitive technique. However, an ambiguity which slows down the process is initiated between letters and thus the comparison slows down the process significantly.

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 8, August 2015

Instead of looking at an entire sequence at once, the S-W algorithm compares multi-lengthen segments, looking for whichever segment maximizes the scoring measure. The algorithm itself is recursive in nature:
The process for this is straight forward in the sense we are converting one string (sequence of letters) into another string by performing certain operations on the individual characters that make up that string. Let us start up the concept briefly which says about the operations such as inserting a character or deleting a character from the source string, or we can re-forge a character in the first string with a character from the second string. Thus, an insertion into one string results in the simultaneous deletion from the second string which happens to trigger simultaneously to minimize the time complexity. The similarity of two strings is simply the value of the alignment between the two strings that maximizes the total alignment value (optimal alignment value), or the highest score given

The project is divided into the following modules based on the functionalities and operations.

Reference Sequence Module
In reference sequence module, the set of reference input sequence is given. The length of the sequence is calculated in this module.

User Sequence Module
In user sequence module, the set of query sequence is given. The length of the query sequence is calculated in this module.

Sequential Match Sequence
Sequential match module takes the input as sequences from reference and user sequence module and produces the output as matched sequence which is aligned sequentially. It calculates the maximum similarity score, gap and time consumed.

Parallel Match Sequence
Parallel match module takes the input as sequences from reference and user sequence module and produces the output a parallel match sequence using parfor ( ) function. It also calculates the maximum similarity score, gap and time taken to match the sequences.

Algorithm used

Input: Reference sequence and User sequence

Step 1: In this step give it the filenames
d1→ Reference Sequence
d2→ User Sequence

Build the matrix row by row
for i = 2 to c1+1
for j = 2 to c2+1

Step2: Initialize max to the first of the three terms.
temp(1) = h(i-1,j-1)+similarity score of each character;
temp(2) = h(i-1,j)-delta;
temp(3) = h(i,j-1)-delta;
temp(4) = 0;
h(i,j) = finding maximum score;

Step 3: Score in (i, j) stems from a match and mismatch.
i_i(i,j) = i-1;
i_j(i,j) = j-1;

Step 4: Search h for the maximal score.
hMax = h(i,j);
iMax = i;
jMax = j;

Step 5: Backtracking from hMax.
score = hMax;
currentI = iMax;
currentJ = jMax;
nextI = i_i(currentI,currentJ);
nextJ = i_j(currentI,currentJ);

Output: Time consumed by Sequential matched sequence and Parallel matched sequence.

III.RESULTS AND DISCUSSIONS

In the project, the results and discussions are used for comparing the existing system with the proposed model by using the performance analysis plot as shown in Fig. 2. In this below figure, the performance analysis with existing and proposed system by comparing with time consumption by both sequential match sequence and parallel match sequence**.** The parallel match yields more efficient result than the sequential match.

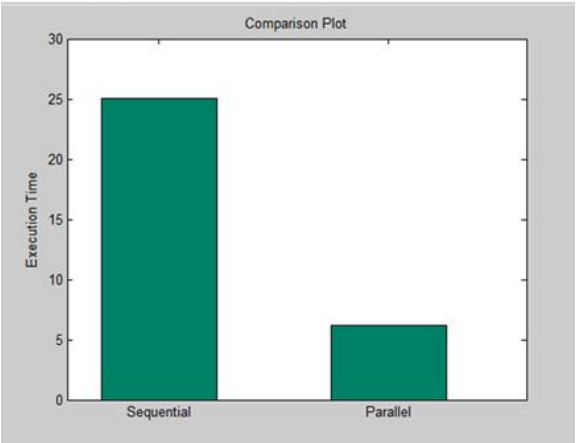

Fig. 2. Performance of the Proposed Work

III. CONCLUSION

In this paper, we discussed the sequential and parallel approach. The Smith-Waterman Algorithm presents a type of dynamic programming method for identifying the improved local sequencing alignments of biological gene pairs. The improvements of parallel sequencing make the efficiency of the system to estimate distance matrices and query matching with neighbour matrices. Because of its highest optimum sensitivity for accomplishing local alignments, this kind of parallel programming approach is a fundamental process in bioinformatics computation, comprising exploration for biological sequence database and sequence alignment. The local alignment increases

DOI 10.17148/IJARCCE.2015.4805

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 8, August 2015

accuracy of the sequence matching within two sequences, reducing running time as well as a good challenge in bioinformatics computing. The time taken for sequence comparison by parallel alignment of Smith-Waterman Algorithm is less compared to the time taken by sequential alignment.

The proposed methods and the previous approaches have their own advantages and disadvantages. For example Needleman-Wunsch algorithm and discriminative model have their own complexities. The future development is to test the complexity of each model so that a framework can be designed this can enhance added efficiency.
Since efficiency is an important measure dynamic programming can be utilize for huge set length of the strings the time for the execution increases which in turn increases the time complexity of the system thus minimizing the efficiency. Hence as a further enhancement a new novel algorithm can be expressed so that efficiency as a feature can be improvised

REFERENCES

[1] Ananth Prabhu G, Dr.Ganesh Aithal, "Automatic parallelization for Parallel Architectures using Smith-Waterman Algorithm" volume 3, issue 9, April 2014.
[2] "Global Alignment of two Sequences Needleman-wunsch Algorithm" www.amritavlab.com
[3] F. Guinand, "Parallelism for computational molecular biology," in ISThmus 2000 Conference on Research and Development for the Information Society, Poznan, Poland, 2000.
[4] Rajesh Mehra, Sonali Vijan, " Biological Sequence Alignment for Bioinformatics Application using MATLAB" Electronics Engineering, NITTTR, Chandigarh.
[5] Azzedine Boukerche, Jan M. Correa, Alba Cristina M, A de Melo, Ricardo.P.Jacobi, "A Hardware Accelerator for the Fast Retrival of DIALIGN Biological Sequence Alignments in Linear Space" IEEE Transactions on Computers, vol.59, No-6, pp.808-821, 2010.
[6] Jane B. Reece and Neil A. Campbell. Campbell biology/Jane B Reece.[et al]. Pearson Australia Frenchs Forest, N.S.W, 9th ed. Edition, 2012. ISBN 9781442531765.
[7] Laiq Hasan and Zaid Al-Ars, "An overview of hardware-based Acceleration of Biological Sequence Alignment" Computational Biology and Applied Bioinformatics, pages 187-202, 2011.
[8] S.B.Needleman and C.D. Wunsch, "A general method Applicable to the search for Similarities in the amino acid sequence of two proteins" Journal of Molecular Biology, 48(3):443-453, 1970.
[9] Temple F Smith and Michael S Waterman, "Comparison of Biosequences" Advances in applied Mathematics, 2(4):482-489.
[10] L.D'Antonio, "Incorporating bioinformatics in an algorithms course," in Proceedings of the 8th annual conference on Innovation and Technology in Computer Science Education, vol. 35 (3), 2003, pp. 211{214.
[11] Needleman S., Wunsch C., A general method applicable to the search for similarities in the amino acid sequences of two proteins, Journal of Molecular Biology, Vol. 48, 1970, pp. 443–453.
[12] Smith T.F., Waterman M.S., Identification of common molecular sub sequences, J Mol Biol, Vol. 147, 1981, pp. 195–197.
[13] Bellman R., Dynamic programming, Princeton University Press, Princeton, N. J. 1957.
[14] Sellers P.H., Pattern recognition in genetic sequences by mismatch density, Bull. Math. Biol., Vol. 46, 1984, pp. 501–514.
[15] Pearson W.R., Flexible sequence similarity searching with the FASTA3 program package, Methods Mol Biol. Vol. 132, 2000, pp. 185–219.

DOI 10.17148/IJARCCE.2015.4805