Available online at www.sciencedirect.com

ScienceDirect

Materials Today: Proceedings XX (2016) XXX–XXX

www.materialstoday.com/proceedings

PMME 2016

Micro Sequence Identification of DNA Data Using Pattern Mining Techniques

A.Surendar[a] Sadulla Shaik[b] N.Usha Rani Rani[c]

[a,b]Assistant Professor,School of Electronics,Vignan's university,Guntur- 522213,India
[c] Professor and Head, School of Electronics,Vignan's university,Guntur- 522213,India

Abstract

A rapid development of NGS(Next generation Sequencing) technologies can be able to produce large amounts of sequence data,which is leading in to a wide range of new applications. Sequence matches between a translated nucleotide sequence and a well known biological protein can able to provide a strong evidence for the presence of homologous coding region, and such similarities often be identified even between a distantly related genes. Common techniques often restrict indels in the alignment to improve time,speed, whereas more Flexible aligners are too slow for large-scale applications. Moreover, many current aligners are becoming inefficient as generated reads grow ever larger. To perform such high dimensional process, it requires a special hardware implementation & designing, such implementation can also increases a complexity of efficiency and hardware. The Field Programmable Gate Array is the well-known to design and we propose a high efficient algorithm for sequence detection in any of bioinformatics data. Unlike previous methods, the proposed pattern matching algorithm can identifies the sequence of each factor on the basis of their occurrences. This method can computes the multi-level similarity measure with an available sequences. Based on the multi-level sequence similarity measure computed a single sequence of bioinformatics data can be identified. The proposed method produces efficient result in sequence searching and detection and improves the hardware utilization which in terms reduces the time complexity as well.

Selection and Peer-review under responsibility of International Conference on Processing of Materials, Minerals and Energy (July 29th – 30th) 2016, Ongole, Andhra Pradesh, India.

Keywords: Bioinformatics; Pattern Matching; Sequence Identification; Dynamic Programming; Hardware Acceleration.

1. Introduction

Bioinformatics is an interdisciplinary research area that is the platform between the biological and computational sciences. The bioinformatics community is doing an research in many subfields , such as DNA,RNA,gene structure prediction, phylogenetic trees, protein docking (2D, 3D) etc., but the most promising one is sequence similarity

2214-7853 © 2016 Elsevier Ltd. All rights reserved.
Selection and Peer-review under responsibility of International Conference on Processing of Materials, Minerals and Energy (July 29th – 30th) 2016, Ongole, Andhra Pradesh, India.

analysis or sequence alignment[1]. Identifying similar sequence in high dimensional data is not such easier and requires some special attention. In most common terms sequence alignment may be defined ned as an arrangement of two or more DNA, RNA or Protein sequences to highlight the regions of their similarity.

1.1. Hardware Based Matching

Bioinformatics applications are characterized by immense computational loads and extremely huge datasets. At the same time, technologies such as reconfigurable computing are reaching at an level of maturity, modern field-programmable gate array devices offer substantial hardware resources. Reconfigurable computing, also known as FPGA (Field Programmable Gate Array)[5] computing, is the field in which algorithms are mapped directly to an FPGA hardware resources. Despite clock speeds that are typically 1/10th of those in general-purpose computing, by exploiting parallelism at all levels, speedups of one to three orders of magnitude can be achieved vs. software executing the same algorithms. The cost per computation and watts per computation are also quite favorable for reconfigurable computing [7], and it is worth to examining the specifics of this form of computers as a platform for bioinformatics applications.

The New powerful FPGA based platforms have emerged during a last two years, ones that combine general-purpose computers and other is FPGAs[17]. These platforms highlight on the high-speed data transmission between the FPGA device and the CPU’s main memory[2], the availability of a conventional CPU and the usage of the network for I/O, thus offering integrated solutions for the execution of I/O- and memory-intensive problems, in which the FPGAs form a tightly coupled coprocessor to the conventional one.

The contributions of this work include:

- The presentation of many bioinformatics algorithms that have been mapped on FPGA stand-alone Platforms. [6]
- The presentation of FPGA technology barriers that is needs to be overcome to reconfigurable technology, which can offer usable, high performance bioinformatics systems.[7]
- The presentation of some case studies for bioinformatics algorithms on the modern high-end FPGA-based Platforms, which show the benefits of new generation FPGA platforms.

2. Related Work

2.1. Blast Algorithm

BLAST[3] is known for its wide use in Bioinformatics. Basic Local Alignment Search Tool is used to find similarities between genetic sequences and sequence databases. It follows a experimental approach based on Smith Waterman algorithms. It locates a best of possible local alignments and It is well known for its statistical significance.

The inputs of this algorithm are the genetic sequence database and a query which has to be found in the database. The outputs of the algorithm are the positions of the areas of these two strings that have similarity, as well the score of these similarities. The quality of each pair-wise alignment is represented as a score and the scores are ranked. Scoring matrices are used to calculate the score of the alignment base by base (DNA) or amino acid by amino acid (protein). The alignment score will be the sum of the scores for each position. The significance of each alignment is computed as E-value. The lower the E value, the more significant is the score and the sequences are homologues for low E values. Each of these pairs, comprising of a database area and a query area, is called a High Score Pair (HSP). The score has significant value for biologist because it is used to compute several variables, of which the e-value is the most important.

Depending on the query and database data types, each BLAST implementation can be classified into many types. Some types are given in Table 1.

Sl.No	Algorithm	Query	Database
1.	BLASTp	Amino acid	Protein
2.	BLASTn	Nucleotide	Nucleotide
3.	nBLASTp	Nucleotide translated	protein
4.	tBLASTn	Amino acid	Nucleotide translated
5.	tBLASTx	Nucleotide translated	Nucleotide translated

Table 1: Variations of BLAST algorithm

The database and query are separated in small substrings known as words. After the word list generation, the database sequences are searched for an exact match between any words of the word list found in the database is called hit. When these words are separately pattern matched among database and query, the patterns searching is extended in both directions with an aim of maximizing the alignment score S. The BLAST algorithm extends the initial word hit to a High scoring Segment Pair[4] (HSP). The BLAST algorithm was designed by balancing speed and increased sensitivity for distant sequence relationships. BLAST emphasizes regions of local alignment to discover relationships among sequences which share only remote regions of similarity.

Reconfigurable computing was used for speeding up of BLAST elements. NCBI databank contains millions of sequences. The hardware implementation should match a sequence with NCBI database, which grows rapidly in size [15]. The input output (I/O) operations of FPGA have been found to be a bottleneck in implementing the BLAST algorithm in previous versions of FPGA configurations due to enormous amount of input data to be analyzed. Recent FPGAs provide embedded blocks of RAM which offers flexibility in design and faster memory access time. Virtex-II Pro and Virtex Pro has a transceiver named ROCKET I/O transceiver which can allow transfer rates of 10.3125 Gb/s. It has been found to work efficiently under smaller transfer rates of 8 Gb/s [16].

2.2. Aho-Corasick Algorithm (ACA)

Aho-Corasick[8] String Matching algorithm is developed by Alfred V. Aho and Margaret J. Corasick belongs to class of string matching algorithm that can be able to finds a elements of a finite set of strings with in an input text. It matches all patterns at a same time. The algorithm defines a finite state machine resembling like a digital tree with essential links between the various internal nodes. These links allows fast transitions between failed pattern matches to other branches of a tree that shares common prefix. It specializes in locating all occurrences of any of a finite number of keywords in a string of text. It is consists of constructing an finite state pattern matching machine from a keywords and then using a pattern matching machine[10] to process the text string in a single pass..

When there is a search for cat in a tree that only contains cart, the search would be a failure when it reaches a node with prefix value ca. Hence the search allows the automaton to transit between pattern matches without the need for backtracking. The time complexity of the algorithm is linear with the length of the patterns (Lp), the length of the searched text (Ls) and the number of output matches (Lo).

Time complexity = Lp + Ls + Lo

Aho-Corasick algorithm was implemented in Virtex IV fx100 with speed grade-12. The FX series device offered better RAM/logic ratio compared to the other devices in the Virtex IV series as the architecture is constrained only by the amount of block RAM and not the logic [11].

2.3. Bloom filter:

Bloom filter[12] is an space efficient probabilistic data structure that can be used to test whether an element is a member of an larger set. This compact representation is the payoff for allowing a small rate of false positives and false negative in membership queries; that queries might be incorrectly recognize an element as member of the set which can be made negligible by the intensive design effort.this bloom filter was implemented on FPGAs for query and data retrieval applications of computer network security and high speed searching. Counting k-mers substrings of length 'k' in DNA sequence data is an essential for the component of many methods in bioinformatics, including for DNA, RNA,genome and transcriptome assembly for metagenomic sequencing, and for error correction of sequence reads. Using a Bloom filter, a probabilistic data structure that stores all the observed k-mers implicitly reduced memory requirements.

2.4. Content Addressable Memory (CAM):

A content-addressable memory (CAM)[13] is a critical device for applications involving communication networks, local area network bridges/switches, databases, lookup tables, and tag directories, due to its high-speed data search capability. Bloom filter and CAM are methods which have more hardware compatibility and higher degree of parallelism [9]. CAM based architectures were implemented for high data intensive search applications using Xilinx Virtex5LX85T [9]. Hardware compatible architectures like Bloom filter and CAM are need to be explored further for FPGA implementation of bioinformatics applications[14].

3. Proposed Work

3.1. Micro Sequence Identification Using Pattern Mining:

The micro sequence identification approach reads the input sequence and for each class of amino acid sequence, the method generates number of pattern from 2 to N, where N is the size of sequence. For each class according to the patterns being generated, the method computes the multi-level sequence similarity measure (MLSSM) which represents how far the input sequence is close to the sequences belongs to the different classes. The entire process can be split into number stages namely Multi Level Pattern Generation, Multi-Level sequence similarity measure computation, Sequence Identification.

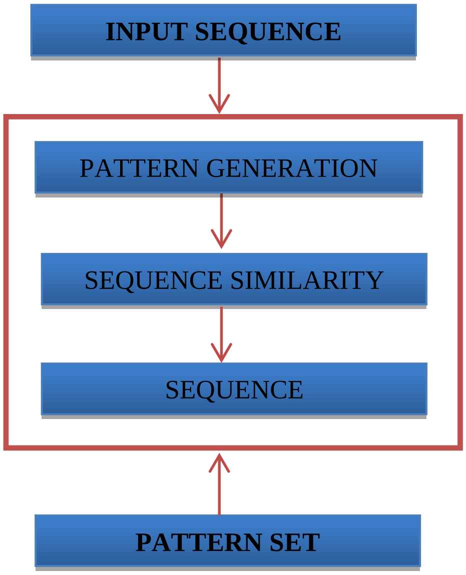

Fig. 1. Architecture of Multi Level Micro Sequence Identification

The Figure 1, shows the architecture of multi level micro sequence identification and shows the components of the proposed approach.

3.2. Multi Level Pattern Generation:

The multi level pattern is the combination of different pattern for different dimension. For given DNA sequence or Amino acid sequence with dimension D, the method generates N number of Dn dimensional sequence from 2 to D. For each class of DNA sequence, the method generates number of different sequence pattern at different levels. The pattern is generated from different dimension and generated pattern will be used to compute the sequence similarity measure.

```
Pseudo Code of MLP Generation:
Input: DNA Sequence Ds
Output: Pattern Set Ps.
Start
    Compute the size of sequence ss.
    Ss = $\int size(Ds)$
    For each dimension Di from SS
        Generate Pattern Pi.
        Pi = $\int_{i=1}^{ss} Subset(Ds, Di)$
        Add to pattern set Ps.
        Ps = $\sum(Pk \in Ps) \cup Pi$
    End
Stop.
```

The above discussed algorithm generates the multi level pattern set from given DNA sequence and generated sequence will be used to compute the sequence similarity measure. For example, from the given Amino Acid sequence "MEKLLDEVLAPGGPYNLTVGSWVRDHVRSIVEGAWEVR", the pattern generation approach can generate the following patterns as follows:

According to the representation and the class of amino acids the input sequence is represented as follows: "CBCCCBCCCCAACCCCCCACCCCCACCACCBCC".

CB	CC	CCC	CCCC	CBCCCB	CCAACCCC	CCACCACCBCC	CCCCACCCCCACCA
CC	CC	CCC	ACCC	CCCCAA	CCACCCCC	CBCCCBCCCCAA	CBCCCBCCCCAACCC
BC	AC	ACC	CACC	CCCCCC	ACCACCBC	CCCCCCACCCCC	CCCACCCCCACCACC
CC	CA	CCA	ACCB	ACCCCC	CBCCCBCCC	CBCCCBCCCCAA	CBCCCBCCCCAACCCC
CC	CC	CCA	ACCA	ACCACC	CAACCCCCC	CCCCCCACCCCC	CCACCCCCACCACCBC
AA	BC	CCB	CBCCC	CBCCCBC	ACCCCCACC	CBCCCBCCCCAA	CBCCCBCCCCAACCCCC
CC	CBC	BCC	BCCCC	CCCAACC	ACCCCCCACCC	CCCCCCACCCCC	CBCCCBCCCCAACCCCC
CC	CCB	CBCC	AACCC	CCCCACC	CBCCCBCCCC	CBCCCBCCCCAA	CBCCCBCCCCAACCCCC

Table 2: Example pattern generated

The Table 2 shows the set of patterns being generated from the above discussed algorithm.

CBCCCBCCCCAACCCCCCACCCC
CBCCCBCCCCAACCCCCCACCCCC
CBCCCBCCCCAACCCCCCACCCCCA
CBCCCBCCCCAACCCCCCACCCCCAC
CBCCCBCCCCAACCCCCCACCCCCACC
CBCCCBCCCCAACCCCCCACCCCCACCA
CBCCCBCCCCAACCCCCCACCCCCACCAC
CBCCCBCCCCAACCCCCCACCCCCACCACC
CBCCCBCCCCAACCCCCCACCCCCACCACCBCC

Table 3: Example pattern generated

The Table 3, shows the example pattern being generated by the proposed algorithm .

The Table 2 and Table 3, shows the set of patterns being identified from the starting position and similarly the patterns can be generated from the remaining dimensions which produce enormous number of patterns. The method generates such patterns and will be used to compute the sequence similarity measure.

3.3. Multi Level Sequence Similarity Measure:

The multi level sequence similarity measure shows the similarity of the sequences at different levels and the number of levels is depending on the dimension of the sequence. For each level using the pattern set being generated, the method computes the sequence similarity measure. For each dimension the method computes the similarity measure and then finally the method computes the multi level similarity measure which will be used to identify the sequence.

```
Pseudo Code of MLSSM:
Input: Pattern Set Ps, Pattern Pi
Output: MLSSM.
Start
    For each level l from Ps
        Compute similarity.
        MLSSM = $\int_{i=1}^{Levels} \int_{i=1}^{size(Ps)} \sum Ps(j,l) == Pi$
    End
    MLSSM = $\frac{MLSSM}{size(Levels)}$
Stop
```

The above discussed algorithm computes the multi level sequence similarity at each level and finally computes the cumulative sequence similarity.

3.4. Sequence Identification:

At this stage, the method generates the sequence set for the given sequence and based on the pattern set being generated the method computes the multi level sequence similarity. For each level of the DNA sequence, the method computes the sequence similarity and finally the method computes the cumulative sequence similarity measure. For each class the method maintains different sequence and the method computes multi level sequence similarity for each class. Based on the sequence similarity measure the method selects a single class and identifies the most sequence similar.

```
Pseudo Code of Sequence Identification:
Input: DNA sequence Ds, Pattern Set Ps
Output: Sequence S.
Start
    Pattern set Dps = Multi Level Pattern Generation(Ds).
    For each class Ci
        For each Level l
            Compute MLSSM.
            MLSSM = $\int_{i=1}^{size\,(DPs)} \sum MLSSM\,(Ps, Dps(i)))$
        End
        MLSSM = $\frac{\sum MLSSMi}{size\,(Ps)}$
    End
    Choose the class with maximum MLSSM.
    Choose the sequence with maximum MLSSM.
Stop
```

The above discussed algorithm computes the multi level sequence similarity measure and selects the class and sequence with maximum sequence similarity measure.

4. Results and Discussion:

The proposed multi level micro sequence has been implemented and evaluated for its efficiency using the Model Simulator and has been evaluated using the FPGA test bench. The method has been validated for its efficiency using various DNA sequence and Amino Acid sequence. The efficiency of the method has been validated by computing the sequence detection accuracy and the time complexity produced.

DATA SET	SIZE
GENIE	793
UCI	2500
dbGap	4300

Table 1: Details of data set used

The Table 1, shows the details of data set being used to evaluate the performance of the proposed approach. The method has been validated for its efficiency using different data sets and the method has been validated for its efficiency in sequence identification and its time complexity.

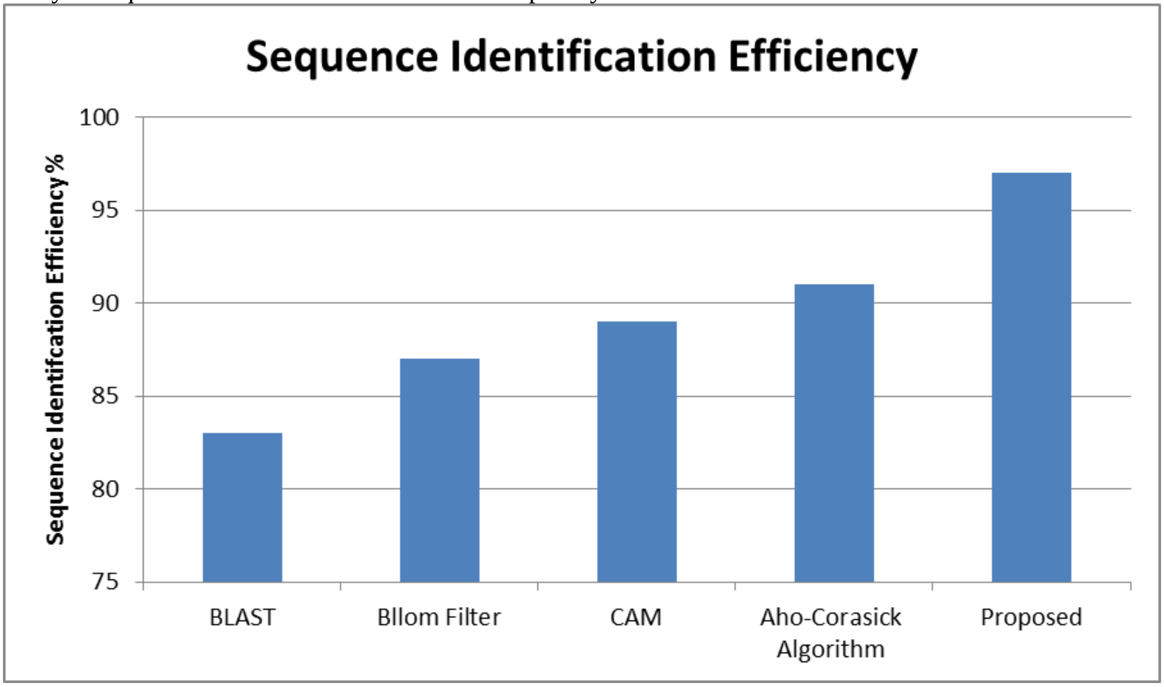

Fig 2: Comparison of sequence identification efficiency

The Figure 2, shows the comparison of sequence identification efficiency produced by different methods and it shows that the proposed method has produces higher efficiency than other methods.

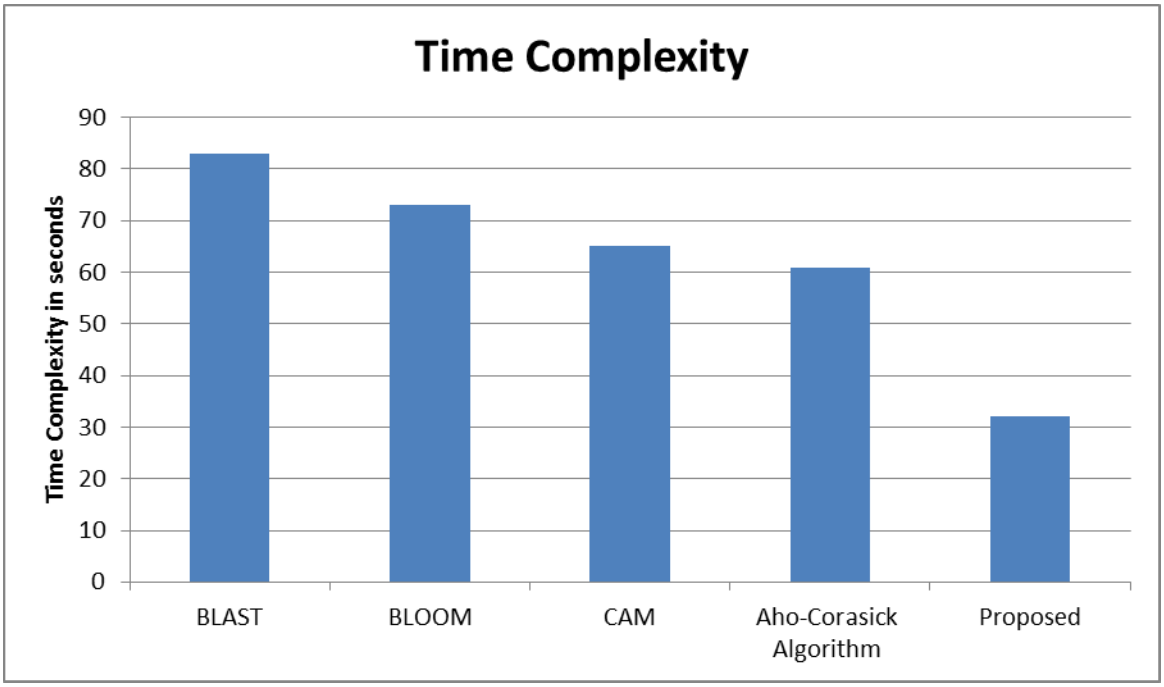

Figure 3: Comparison of time complexity

The Figure 3, shows the comparison of time complexity produced by different methods in identifying the sequence and the result shows that the proposed method has produced less time complexity than other methods.

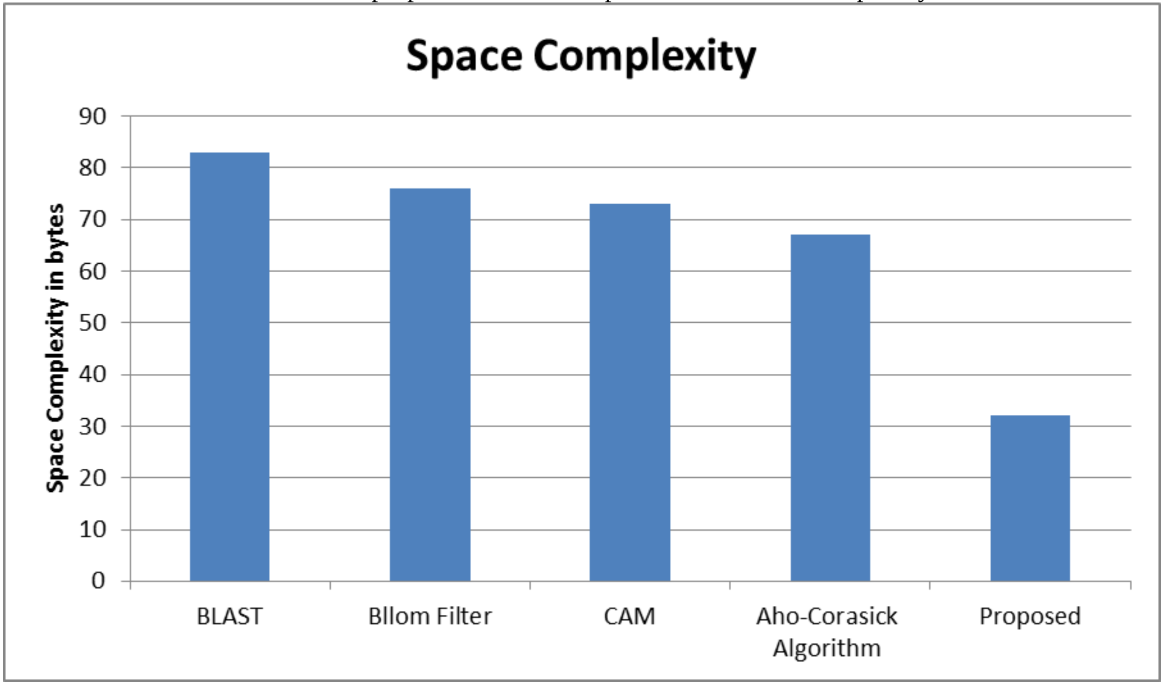

Figure 4: Comparison of space complexity

The figure 4, shows the comparison of space occupied by the different methods and it shows that the proposed method has produced less space complexity than other methods.

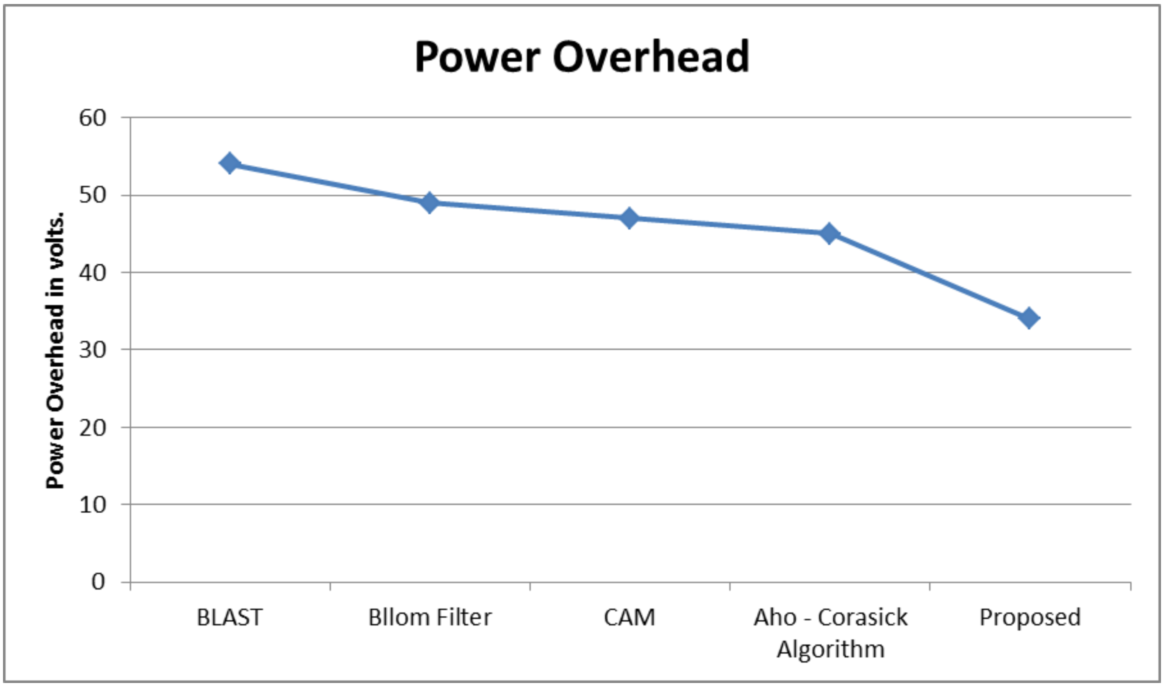

Figure 5: Comparison of power overhead

The figure 5, shows the comparative results on power overhead produced by different methods and the result shows that the proposed method has produced less power overhead than other methods.

5. Conclusion:

In this paper, we propose a micro sequence identification using pattern mining technique. First the method generates number of patterns or sequences from the dimension 2 to the dimension N. The patterns are generated at each dimension and with varying size of dimension. The generated patterns are stored in the pattern set and for the input sequence the method generates the similar set of pattern set. Based on generated pattern set, the method computes the sequence similarity at each level and finally a cumulative similarity value is computed. Based on the value of multi level sequence similarity value the method selects the sequence as the result. The proposed method identifies the DNA sequence in efficient manner and reduces the time complexity.

References

[1] F.N.Muhamad,R.B.Ahmad,S.Mohd.Asi,M.N.Murad, Jurnal Teknologi ,"Reducing the search space and time complexity of Needleman wunsch algorithm and smith waterman algorithm for DNA sequence alignment ",ISSN 2180–3722, 77:20 (2015) 137–146.

[2] Sean R Eddy, "A memory-efficient dynamic programming algorithm for optimal alignment of a sequence to an RNA secondary structure",BMC Bioinformatics,

[3] Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ: Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. Nucl. Acids Res. 1997,25:3389-3402

[4] Pearson WR, Lipman DJ"Improved tools for biological sequence comparison", Proc. Natl. Acad. Sci. USA 1988, 85:2444-2448.

[5] Ali Khajeh-Saeed , Stephen Poole , J. Blair Perot "Acceleration of the Smith–Waterman algorithm using single and multiple graphics processors", Journal of Computational Physics 229 (2010) 4247–4258,pp 4247 – 4258.

[6] A Surendar, M Arun, C Bagavathi "Evolution of Reconfigurable Based Algorithms for Bioinformatics Applications: An Investigation"- Int. J. Life Sci. Bt & Pharm. Res, 2013.

[7] A Surendar, M Arun, PS Periasamy “Hardware Based Algorithms for Bioinformatics Applications--A Survey.” International Journal of Applied Engineering Research, 2013.

[8] A Surendar, M Arun, PS Periasamy “A parallel reconfigurable platform for efficient sequence alignment” African Journal of Biotechnology, 2014

[9] Arun M and Krishnan A, “Functional Verif icat ion of Signature Detect ion Architectures for High Speed Network Applications”, International Journal of Automation and Computing, Springer, Vol. 9, No. 4, pp. 395-402, 2011.

[10] R. Valli Suseela “An Efficient Retouched Bloom Filter Based Word-Matching Stage Of Blastn” International Journal of Engineering and Scientific Research. 1(1), 2014, 25 – 31.

[11] Van Hoa Nguyen and Dominique Lavenier “parallel local alignment search tool for database comparison”, BMC Bioinformatics,BMC Bioinformatics 2009, 10:329 doi:10.1186/1471-2105-10-329.

[12] GenBank Statistics at NCBI [Online].

[13] Available: http://www.ncbi. nlm.nih.gov/genbank/genbankstats.html

[14] Gabriel F Villoente, Mark Oliver L Ouano, Mary Grace C Dy Jongco, and Emilyn B Escabarte , “FPGA Based Agrep for DNA Microarray Sequence Searching”, International Conference on Computer Engineering and Applications, IACSIT Press, Vol. 2, 2011.

[15] M. Sujithra and G. Padmavathi, 2015. A Survey of Biometric Iris Recognition: Security, Techniques and Metrics. Asian Journal of Information Technology, 14: 192-199.

[16] Yongchang Zheng, Masood ur-Rehman , Ting F. Zhu, Rapid identification of multi-strain HBV infection in patient by high-throughput DNA sequencing, Springer, Quantitative Biology , Volume 3, Issue 2, pp 103-106, 2015.

[17] Krishna, VVS Vijay, et al. "Design and implementation of an automatic beverages vending machine and its performance evaluation using Xilinx ISE and Cadence." 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT). 2013.