NCICIT 2013: 1st National Conference on Intelligent Computing and Information Technology, November 21, CUET, Chittagong-4349, Bangladesh

Performance Evaluation of Warshall Algorithm and Dynamic Programming for Markov Chain in Local Sequence Alignment

Mohammad. I. Khan[1], Md. S. Kamal[2]
[1]Computer Science & Engineering., Chittagong University of Engineering and Technology, Bangladesh.
[2]Computer Science & Engineering., Chittagong University of Engineering and Technology, Bangladesh.
Email: muhammad_ikhancuet@yahoo.com[1], sarwar.saubdcoxbazar@gmail.com[2]

***Abstract*— Markov Chain is very effective in prediction basically in long data set. In DNA sequencing it is always very important to find the existence of certain nucleotides based on the previous history of the data set. We imposed the Chapman Kolmogorove equation to accomplish the task of Markov Chain. Chapman Kolmogorove equation is the key to help the address the proper places of the DNA chain and this is very powerful tools in mathematics as well as in any other prediction based research. It incorporates the score of DNA sequences calculated by various techniques. Our research utilize the fundamentals of Warshall Algorithm (WA) and Dynamic Programming (DP) to measures the score of DNA segments. The outcomes of the experiment are that Warshall Algorithm is good for small DNA sequences on the other hand Dynamic Programming are good for long DNA sequences. On the top of above findings, it is very important to measure the risk factors of local sequencing during the matching of local sequence alignments whatever the length.**

Keywords: Keywords:Hidden Markov Model, Chapman-Kolmogorov formula, Warshall Algorithm, Dynamic Programming, Score measurement.

I. INTRODUCTION

Markov Chain can be easily formulated in the state space for the simple model such as 0 for first Nucleotide Adenine (A), 1 for second Nucleotide Cytosine (C), 3 for Thiemann (T) and finally 4 for Guanine (G). For same data set it can be also possible by using second order Markov Chain value as {00,01,02,03,04.....}. Since the data sets in DNA sequences contain four fundamental bases, there should be 4^2 possible space states. But the complexity for higher order model is higher than a simple model and for this reason Markov process always holds the simple state space. For example if we ask to predict for 10000th Nucleotide in a sequence and we have to measure the 9999th Nucleotide in the same sequences.

On the same time it is possible to quantify the pairwise evolutionary distances, Hamming distance. If U is the total number of mismatches in an alignment of length l, then the Hamming distance for per 10000 sites is

$$H(U,l)=10000\frac{\vartheta}{l} \quad \text{.....(1)}$$

The equation 1 above works good when the DNA sequences space is discrete. To measure the discrete Markov Process for the A,G,C, and T Nucleotides the starting distribution will be as follows: $P^0=\rho C,\rho A,\rho G,\rho T$. The figure 1 below shows the basic Markov Process Transaction for Discrete system in DNA sequencing.

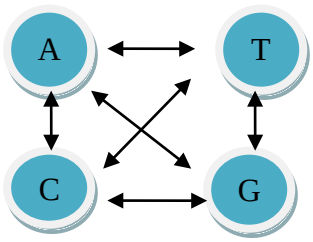

Figure 1: The transactional states for Markov chain.

II. MOTIVATION

According to the various algorithms, implemented through 2001 to 2005 as Ning [1], Kent [3,4] Schwartz [5] and Watanabe [6] examined the accuracy of the sequencing. In the age of information superhighway we know that the genome sequences may have continual or near about continual patterns in the given or collected data sets. As a result the outcomes might be same for many positions. On the contrary the mutations and **Indel** may generate incorrect judgments for sequencing and mapping whether it is local, global or pair-wise alignments or mapping. The algorithms above do not consider the situations regarding the repetitions of the patterns, mutations and incorrect mapping. Here we have noticed the system rejects the data sets, made the area

small and as consequences the calculations become complicated as well as wrong. According to the Ewing and Green [2] proposed a solution to overcome the ambiguity but this method is candidate of low-quality regions. Here we have implemented the Markov chain concept under Chapman – Kolmogorove equations to established probable sequenced positions. In the field of sequencing, there are lot of software's helping to detect the sequences and the frequently used systems are PolyPhred [7], SNP detector [8], and novoSNP [9]. But these three systems only detect genotype sample. They are unable to solve the dynamic patterns and sequences. In this research we have imposed the idea on prediction using Markov Process under the support of Dynamic Programming and Warshall Graph Algorithm. The fundamental step of prediction is that this Chapman Kolmogorove calculative prediction. We have compared DP and WA algorithm in the light of Local Sequence alignment with various length. The detection based depends on homolog finding. The homolog finding mainly depends on the database finding [10]. But the database finding is not always efficient due to the size and cost of the equipments. Training based sequencing is not able to identify the proper coding regions due to the lack of generalizations [11]. Besides, predictions are vulnerable with many false positives identifications and sensitivities [14].

III. DYNAMIC PROGRAMMING

Dynamic Programming decomposes the sequences into several parts and solved the problem recursively until it reaches to a particular condition. Sometimes decompositions becomes difficult and it hard to get a clear-cut solution. In that case we should investigate several possibilities and Recursive solution is one of the acceptable methods to overcome the problem. Basically, Dynamic Programming to choose the Maximum Matching Sub Sequences (MMSS).

Suppose
P =p_1, p_2 . . .p_n, and Q = q_1, q_2, . . . q_m, are two DNA sequences. The indel of two sequences is denoted by the weight M (g). At first we consider the best alignments as F (p, q) = max $\forall$ (p*,q*). Where, F is a function which relates the current alignments and new alignment. By using Dynamic Programming we can check the sequences F (p,q) recursively.
F (i,j) = F (p_1, p_2 . . .p_i q_1, q_2, . . . q_j)
Where F(0, O)=O, F(O,j)=F(-,q_1, q_2, . . . q_j) = M (j) and F(i, 0)= M(i). Then:
F(i,j)={F(i-1,j- l)+F(p_i,q_j), max { F (i - k , j) + M (k) } , max { F(i, j - I) + M(l)} . For local sequence alignment the function L(p, q)=max{F(p_u,p_{u+1},……p_v,q_x,q_{x+1},…….q_y): 1 ≤u ≤ v ≤n , l ≤x ≤y ≤m } .

IV. WARSHALL ALGORITHM ALIGNMENT

Warshall graph algorithm is a sequence path finding process which subgroups the entire data set into set of intermediate nodes along the path. To perform the decompositions, it is easy to label the nodes set form 1 to n. The decompositions helps to reduces the shortest paths along the sequences. By designing the total path under the variables I,J, and K we can say if there is path from I to J and J to K than we can say that there is a path from I to K. In a word we can say that the total path is
P[I,J,K]=Shortest path from I to J using the only intermediate node 1……K. the recursive process of solving the shortest path is as follows:

$$P[I,J,K] = MIN\{P[I,J,K-1], P[I,K,K-1] + P[K,J,K-1]\}..(1)$$

We can illustrate equation 1 as follows as algorithmic steps.

Sequencing Graph (Adjacent Matrix:ADM_R, n×n,)

1. Initialize Graph weight for all nodes as Weight: =ADM_R,(Weight=w_{ij})
2. Loop k=1;to n,
3. Inner loop J=1 to n;
4. Inner loop I=1;to n,
5. $W_{ij}=w_{ij} \vee (w_{ik} \wedge w_{kj})$
6. $W_{ij}=w_{ij} \vee w_{kj}$)
7. Return initial Weight

V.MARKOV CHAIN ALIGNMENT

Our motivation on mismatches identification according to the Chapman-Kolmogorove formula on Markov chain based stochastic matrix. The formula for a stochastic process with random variable X is X={X_t , t$\in T$ }. Where t= index and it indicate the time. X_t = State of the process .T= Index set constitute by time t.
Suppose n=0, 1, 2, 3….. And m=1, 2, 3…and i_0 ……i_m $\in$ E. E=All possible values that the random variable X_t can assumes. Then

$$\Pr\{X_{n+1} = i_1, \ldots, X_{n+m} = i_m \mid X_n = i_0\} = \Pr_{i_0 i_1} \cdot \Pr_{i_1 i_2} \cdot \ldots \cdot \Pr_{i_{m-1} i_m} \cdot \Pr\{X_{n+m} = j \mid X_n = i\}$$

$$= \sum_{k=0}^{\infty} \Pr\{X_{n+m} = j \mid X_{n+1} = k, X_n = i\} \Pr\{X_{n+1} = k \mid X_n = i\}$$

$$= \sum_{k=0}^{\infty} \Pr\{X_{n+m} = j \mid X_{n+1} = k\} \Pr\{X_{n+1} = k \mid X_n = i\}$$

$$= \sum_{k=0}^{\infty} \Pr_{ik} \Pr_{kj}$$

$$= \Pr_{ij}^{m}$$

In general,

$$\Pr_{ij}^{n+m} = \sum_{k \in E} \Pr_{ik}^{n} \Pr_{kj}^{m} \quad \text{for all } n, m \geq 0, \text{ all } i, j \in E.$$

VI. IMPLEMENTATION

We have implemented and experimented under the environments of Java with Integrated Development Environment (IDE) Netbeans. The object oriented implementation helped us to perform the nucleotides (A, C, T, and G) as a distinct object. In our previous work [1] we have improved the performance of [16] and noticed our RSAM algorithm is significantly better in the light of Speed, Complexity, Space, Sensitivity, Accuracy and risk. In this research we have compared all the above parameters under the light of Markov Process and Warshall Graph Algorithm. For Speed, Sensitivity and Accuracy we have measured referential value as best, average and low. Here we have checked the complexity, risk, accuracy and space for the first time and many local sequence alignment tools measured the sensitivity without any standard parameter. According to the MUMmer [17] termed the parameter '**q**' as the ratio between accurate aligned nucleotides pairs and total number of nucleotides in the given sequence. Total Column Score (TCS) is another aspect of MUMmer procedure. Again according to the AVID [18], where the authors considered the alignment pairs which have the score greater than the predefined threshold value. Instead of all of the methods above, we have concentrated towards the set operations under the complete machine learning process on exons and introns. Introns measurement are also essential part of the alignment to maintain proper checking instead of only one parameters checking (exons). For speed, sensitivity and accuracy the reference values have been checked according to the fuzzy manner, such as: best (H), average (M) and low (L).but according to the [16] there is no clue to compare the Sensitivity of the sequencing.

VII. RESULT

The outcomes of these two process, a few interesting changes have had observed. Chapman-Kolmogorov equation in Markov process is very efficient for any arbitrary predictions in any DNA segments or sequences. It is clearly noticed that Markov Process has protential and strong capabilities to handle the data set whatever the environment. In this case we also found that MP has significanlty better scpes for long data set. Figure 2 below shows the experimental outcomes for Markov Process.

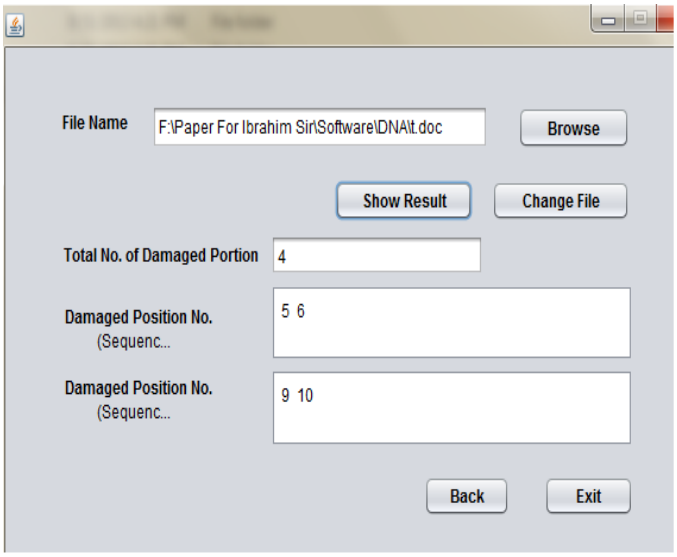

Figure 2: Markov Process out comes for data set.

On the contrary, Warshall Graph Algorithm, has limited scopes than that of Markov Process. But it has faster capabilites on short data set. The speed of the processing is sharply better than Markov Proceess. Other parameters such as complexity , Risk, Sensibity, Space and Accuracy are also signicicantly better than Markov Process due to its faster solving capabilties. Only pivotal drawback is that Warsehall Graph Algorithm work only for short length data set what ever the protein, DNA or RNA. Figure 3 below shows the performace Dynamic programming impact for Markov Process.

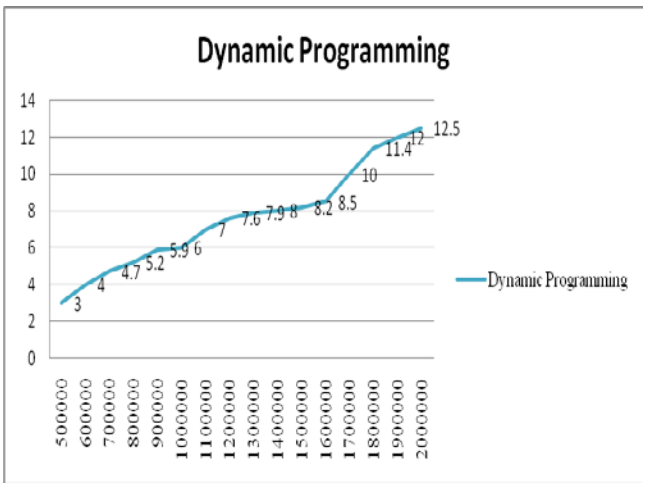

Figure 3: The impact of Dynamic Programming for Markov Process

Here the CPU Utilization time for long data set range from 1500000 to 2000000 requires are 8.2, 8.5, 10, 11.4,12, and 12.5. On the contrary, Warshall Graph algorithm takes more time for the same data set and the time values are 8.2, 9, 10.8, 12, 12.9 and 14. But for the previous data set whose lengths are less than 1500000, Warshall Graph Algorithm takes less time than Dynamic Programming. Figure 4 below shows the impact for Warshall Graph Algorithm.

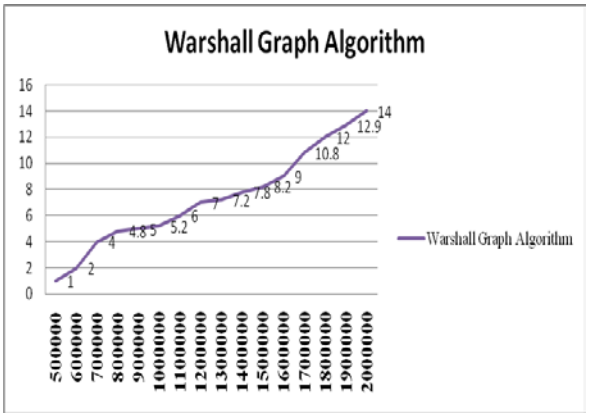

Figure 4: Impact of Warshall Graph algorithm

The reasons behind Dynamic programming requires more time to solve small data set is that it works for arbitrary probabilistic values where Warshall Graph Algorithm works deterministic path and values. The comparative results of these two methods are below at figure 5.

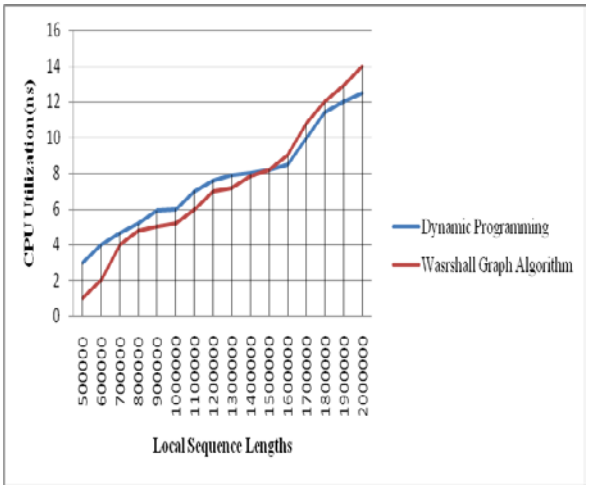

Figure 5:Comparitive Illustration of Dynamic Programming and Warshall Graph Algorithm.

VIII. CONCLUSION

Both Dynamic Programming and Warshall Graph Algorithm perform predictions of DNA base pair according to the process. Dynamic Programming has better capabilities to handle large data set due to its randomness. On the other side, Warshall Graph Algorithm works based on predefine values and path. That why Warshall Graph Algorithm has to check the entire path and values weather the path is short or long. That is the reason Dynamic Programming requires more time. We will find why Randomness causes more time and deterministic process is better for small data set in future work.

REFERENCES

[1]. Mohammad Ibrahim Khan and Md.Sarwar Kamal, ***"RSAM: AN INTEGRATED ALGORITHM FOR LOCAL SEQUENCE ALIGNMENT",*** Archives Des Sciences, Vol 66, No. 5;May 2013, ISSN 1661-464X,pp,395-412.

[2]. Ning Z., Cox A.J., Mullikin J.C. SSAHA: "A fast search method for large DNA databases", Genome Res.; 11:1725–1729, 2001.

[3]. Ewing B., Green P. "Base-calling of automated sequencer traces using *phred*. II. Error probabilities", Genome Res.;8:186–194,1998.

[4]. Kent W. J., "BLAT—the BLAST-Like Alignment Tool", Genome Res.**12**:656–664, 2002.

[5].Kent W.J., Sugnet C., Furey T., Roskin K., Pringle T., Zahler A. , Haussler D. ,"The human genome browser at UCSC", Genome Res.**12**:996–1006,2002.

[6]. Schwarz D.S., Hutvagner G., Du T., Xu Z., Aronin N., and Zamore P.D., "Asymmetry in the assembly of the RNAi enzyme complex", Cell **115:** 199,2003.

[7]. Watanabe T., Takeda A., Mise K., Okuno T., Suzuki T., Minami N., and Imai H., "Stage-specific expression of microRNAs during Xenopus development", FEBS Lett. **579:**318,2005.

[8]. Stephens M., Sloan J.S., Robertson P.D., Scheet P., Nickerson D.A., "Automating sequence-based detection and genotyping of SNPs from diploid samples", Nat. Genet.vol 38,pp.375-381, 2006.

[9]. Zhang J., Wheeler D.A., Yakub I., Wei S., Sood R., Rowe W., Liu P.P., Gibbs R.A., "Buetow K.H. SNPdetector: A software tool for sensitive and accurate SNP detection". PLoS Comput. Biol., 2005.

[10]. Weckx S., Del-Favero J., Rademakers R., Claes L., Cruts M., De Jonghe P., Van Broeckhoven C., De Rijk P. "novoSNP, a novel computational tool for sequence variation discovery", Genome Res.;15:436–442,2005.

[11]. Stephens M., Sloan J.S., Robertson P.D., Scheet P., Nickerson D.A., "Automating sequence-based detection and genotyping of SNPs from diploid samples", Nat. Genet.vol 38 (2006),pp.375-381.

[12]. Zhang J., Wheeler D.A., Yakub I., Wei S., Sood R., Rowe W., Liu P.P., Gibbs R.A., "Buetow K.H. SNPdetector: A software tool for sensitive and accurate SNP detection". PLoS Comput. Biol., 2005.

[13]. Weckx S., Del-Favero J., Rademakers R., Claes L., Cruts M., De Jonghe P., Van Broeckhoven C., De Rijk P. "novoSNP, a novel computational tool for sequence variation discovery", Genome Res.;15 (2005):436–442.

[14]. Yetisgen-Yildiz, M. & Pratt, W., "The effect of feature representation on Medline document classification", In AMIA Annual Symposium Proceedings, , American Medical Informatics Association, vol. 23 (2005), 849.

[16]H.Waqaar, A. Alex, and R. Bharath, An Efficient Algorithm for Local Sequence Alignment, 20-24, 2008**.**

[17] Delcher A.L., Kasif S., Fleischmann R.D., Peterson J., White O., and Salzberg S.L. 1999. Alignment of whole genomes. Nucleic Acids Res. 27: 2369-2376.

[18] Bray Nick, Dubchak Inna and Pachter Lior, Avid: A global alignment program, Genome Research. 2003 13: 97-102; 2002.