AENSI Journals

Australian Journal of Basic and Applied Sciences

ISSN:1991-8178

Journal home page: www.ajbasweb.com

Protein Sequence Alignment Using Dynamic Programming

M. Muriati and A.K.A. Zuriyati

Faculty of Computer, Media & Technology, TATI University College, 24000Terengganu, MALAYSIA

ARTICLE INFO

Article history:
Received 20 November 2013
Received in revised form 24 January 2014
Accepted 29 January 2014
Available online 5 April 2014

Keywords:
Protein sequence alignment; dynamic programming; bioinformatics

ABSTRACT

Bioinformatics is an emerging scientific discipline that uses information technology to organize, analyse and distribute biological information in order to answer complex biological questions. One of the complex biological questions is protein sequence alignment. Protein sequence alignment is a process to find similarity among protein sequences which can be identified as being related. The result usually being used to study the evolution, discover functional and structural information in protein data. A variety of computational algorithms have been applied to the sequence alignment problem. In this paper, we review the dynamic programming algorithm as one of the most popular technique used in the sequence alignment. The study showed the algorithm is guaranteed to find the best alignments. However, it has two major issues; time and scoring function. Due to these issues, some researchers are competing to each other to introduce new improvement over dynamic programming. The new variants of dynamic programming promise better algorithm but future work is still needed to ensure the algorithm are able to hold the growing biological data.

To Cite This Article: M. Muriati and A.K.A. Zuriyati., Protein Sequence Alignment Using Dynamic Programming. ***Aust. J. Basic & Appl. Sci.***, *8(4): 47-51, 2014*

INTRODUCTION

In Malaysia, Bioinformatics started in the early 1990s through individual initiatives within academia, offering introductory-level computational biology modules in seminars and workshops. The creation of a National Biotechnology Directorate in May 1995 and the launch of the Multimedia Super Corridor of the government generated a sufficient foundation for Bioinformatics to take root in the scientific community (Zeti *et al*., 2009).

One of the important issues in Bioinformatics is protein sequence alignment. It is a process to study a protein. Proteins are linear polymers built from series of 20 different amino acids identified by A, C, D, E, F, G, H, I, K, L, M, N, P, O, R, S, T, U, W and Y (Gill and Singh, 2011). Each protein differs according to the amount, type and arrangement of amino acids that make up its structure where the amino acids are encoded from three elements of DNA that consists of four alphabets, A (Adenine), C (Cytosine), G (Guanine) and T (Thymine) (Chao, 2006).

Modification that has occurred in the protein sequence when it is led by certain chemicals is one of the reasons why this issue is arise (Genome Research Institute, 2013). This modification caused the relevant information of the existing genetic change and cannot be used more as references. It makes errors in the protein sequence, creating non-functional and unidentified proteins (Oladele *et al*., 2009).

2. Protein Sequence Alignment:

Protein sequence alignment plays as the main rule in the validation process to study a protein. It is a process of comparing proteins within a species or between different species to find the similarities between protein functions. These similar regions provide important information like protein function, protein structure and the evolutionary of the sequences under study (Gill and Singh, 2011). This information is important especially for biologist who needs that information for developing better drugs, classifying the proteins, discover new functional and etc (Sharma, 2009).

The process is done by first comparing the unknown protein sequence with an identified protein sequence which can be taken from any available sequence database. It will search a series of character patterns that are in the same order of the sequences. Once the protein sequence is identified and validated, the protein sequence

Corresponding Author: M. Muriati, Department of Computer Networking, Faculty of Computer, Media & Technology, TATI University College, 24000 Terengganu, MALAYSIA.
E-mail: muriati@tatiuc.edu.my

information can be submitted to any available database for further references such as functional information, structural information and evolutionary information (Antler, 2011).

The identified proteins usually are stored in open access databases like SwissProt, Protein Information Resources (PIR), Protein Data Bank (PDB) and many other databases by their founder after several processes of validation and confirmation.

When aligning sequences, the most fundamental issue is to find the optimal structure by aligning two sequences across their entirety, which is referred to as pairwise global sequence alignment (Rouchka, 2006). Global alignments, which attempt to align every residue in every sequence, are most useful when the sequences in the query set are similar. The alignment attempts to match them to each other from end to end, even though parts of the alignment are not very convincing. A general global alignment technique is the Needleman-Wunsch algorithm, which is based on dynamic programming.

Dynamic Programming is a fundamental problem-solving technique that has been widely used for solving a broad range of search and optimization problems (Weimann, 2009). Dynamic programming is both a mathematical optimization method, and a computer programming method. In both contexts, it refers to simplifying a complicated problem by breaking it down into simpler subproblems in a recursive manner. These subproblems are then tackled one by one, so that the answers to small problems are used to solve the larger ones (Weimann, 2009). Dynamic programming for pairwise sequence alignment requires three steps that include initialization, matrix fill, and traceback.

In typical usage, protein sequence alignments use a scoring scheme to assign scores to amino-acid matches or mismatches, and a gap penalty for matching an amino acid in one sequence to a gap in the other. A simple scoring scheme use +1 as a reward for a match, -1 as the penalty for a mismatch, and ignore gaps.

3. Dynamic Programming:

Dynamic programming that performs a pairwise global sequence alignment was introduced by Needleman and Wunsch (1970). This technique is known as Needleman-Wunsch generally has three main steps; initialization, matrix fill and traceback. Initialization refers to the step of creating a matrix with m + 1 columns and n + 1 rows where m and n is the size of the aligned sequences. Matrix fill is a step to find the optimal score for each position in the matrix where the dynamic programming algorithm was applied by adding the current match score to the previously scored position (Needleman and Wunsch, 1970). The last step is traceback which is the process to determine the actual alignments that result in the maximum score. Fig. 1 shows all the main steps to align two protein sequence alignment globally using the dynamic programming.

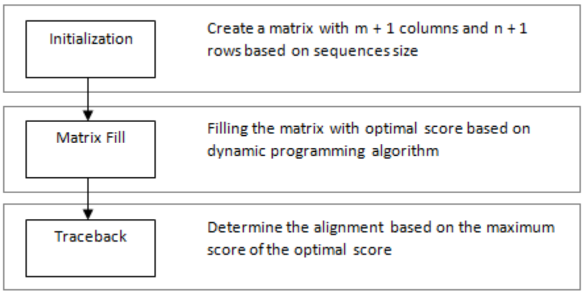

Initialization
Create a matrix with m + 1 columns and n + 1 rows based on sequences size
Matrix Fill
Filling the matrix with optimal score based on dynamic programming algorithm
Traceback
Determine the alignment based on the maximum score of the optimal score

Fig. 1: Main steps of Protein Sequence Alignment with Dynamic Programming.

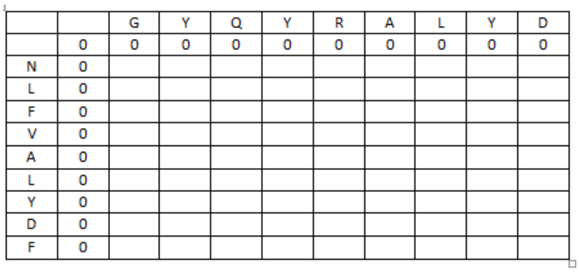
The following is a description of each step. The two sequences to be globally aligned are GYQYRALYD, a rat protein sequence (sequence 1) and NLFVALYDF, a human protein sequence (sequence 2) (BAliBASE, 2012). These two sequences were taken from BAliBASE, which is a benchmark alignment database for genomes and proteins (Thomson *et al*., 1999).

Initialization:

The first step is to create a matrix with m + 1 columns and n + 1 rows where m and n refer to the size of the sequences to be aligned. Since there is no gap penalty used, the first row and first column of the matrix filled with 0 as shown in Fig. 2.

Matrix Fil:

This step finds the maximum global alignment score by starting in the upper left in the matrix for each position in the matrix. The score is calculated using the dynamic programming approach as Eq. (1) below.



		G	Y	Q	Y	R	A	L	Y	D
	0	0	0	0	0	0	0	0	0	0
N	0									
L	0									
F	0									
V	0									
A	0									
L	0									
Y	0									
D	0									
F	0									

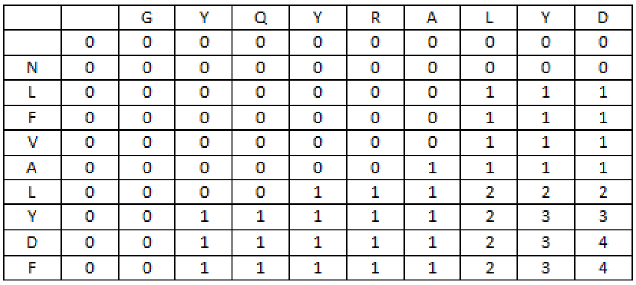
Fig. 2: The Initialization Process.

$$M_{i,j} = \text{Max} \begin{cases} M_{i-1,j-1} + S_{i,j} \\ M_{i,\,j-1} + \text{gap} \\ M_{i-1,\,j} + \text{gap} \end{cases} \qquad (1)$$

The first calculation takes the value of M(i-1,j-1) position and added with match or mismatch score. The second calculation takes the value of M(i, j-1) position and added with gap value. Since there is no gap penalty used, the score refers to the M(i, j-1) position value and the third calculation refers to the M(i-1, j) position value. Using this algorithm, the score at M(1,1) position in the matrix can be calculated as shown in Eq. (2).

$$M_{1,1} = \text{Max} \begin{cases} 0 + (-1) \\ 0 \\ 0 \end{cases} \qquad (2)$$

A value of 0 is then placed in M(1,1) position of the scoring matrix. The calculation process is continued until the last position the matrix. The final score for the matrix is shown as Fig. 3.

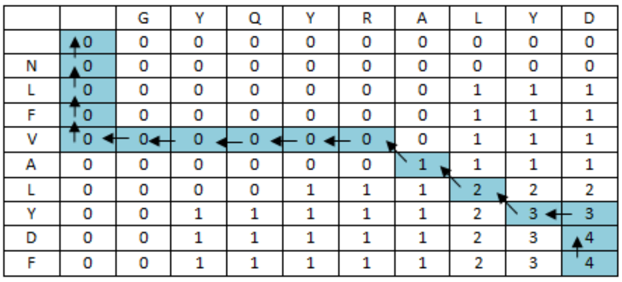


		G	Y	Q	Y	R	A	L	Y	D
	0	0	0	0	0	0	0	0	0	0
N	0	0	0	0	0	0	0	0	0	0
L	0	0	0	0	0	0	0	1	1	1
F	0	0	0	0	0	0	0	1	1	1
V	0	0	0	0	0	0	0	1	1	1
A	0	0	0	0	0	0	1	1	1	1
L	0	0	0	0	1	1	1	2	2	2
Y	0	0	1	1	1	1	1	2	3	3
D	0	0	1	1	1	1	1	2	3	4
F	0	0	1	1	1	1	1	2	3	4

Fig. 3: The Final Matrix Fill Process.

Traceback:

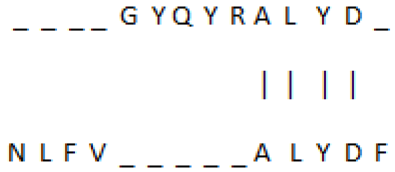
After the matrix fill step, the maximum score for the two test sequence is 4. The traceback process determines the actual alignment(s) that result in the maximum score. It starts at the end of the alignment, the position that leads to the maximal score. Traceback takes the current cell and looks to the neighbour cells that could be direct predecessors. This means it looks to the neighbour to the left, the diagonal neighbour and the neighbour above it. The algorithm for traceback chooses as the next cell in the sequence as one of the possible predecessors. If more than one predecessor exits, any can be chosen. This procedure follows as shown as in Fig. 4.



		G	Y	Q	Y	R	A	L	Y	D
	0	0	0	0	0	0	0	0	0	0
N	0	0	0	0	0	0	0	0	0	0
L	0	0	0	0	0	0	0	1	1	1
F	0	0	0	0	0	0	0	1	1	1
V	0	0	0	0	0	0	0	1	1	1
A	0	0	0	0	0	0	1	1	1	1
L	0	0	0	0	1	1	1	2	2	2
Y	0	0	1	1	1	1	1	2	3	3
D	0	0	1	1	1	1	1	2	3	4
F	0	0	1	1	1	1	1	2	3	4

Fig. 4: The Traceback Process.

The traceback is complete when it gets to the position in column 0 and row 0. Since this is an exponential problem, this algorithm will only produce a single result. One optimal alignment for GYQYRALYD and NLFVALYDF is shown in Fig. 5.



```
_ _ _ _ G Y Q Y R A L  Y D _
                  | |  | |
N L F V _ _ _ _ _ A L  Y D F
```

Fig. 5: The Optimal Alignment.

Issues in Dynamic Programming:

Dynamic programming has two major issues. The first one can be explained with one word: "Time". Dynamic programming is slow and time consuming because of the pre-processing stage it takes during the making of the matrix fill and the traceback alignment process. It is impractical for a large number of sequences (Manohar and Singh, 2012). Due to this issue, several new approaches were introduced such as progressive approach and iterative approach since 1988 (Manohar and Singh, 2012; Do and Katoh, 2002).

The second issue is the scoring function. As dynamic programming use simple scoring score, it cannot present the theory of evolution (Gotoh, 1982; Henikoff and Henikoff, 1991; Henikoff and Henikoff, 1992; Henikoff and Henikoff, 1993). The choice of a scoring function that reflects biological or statistical observations about known sequences is important to produce good alignments. In order to solve this problem, several meaningful scoring schemes were introduced in 1978. The schemes use a biological score which is based on the observed frequencies of such occurrences in alignments of related proteins. Two most commonly used are Percent Accepted Mutations (PAM) and BLOck SUbstitution Matrix (BLOSUM) (Deusdado and Carvalho, 2008).

Current Research:

Dynamic programming is guarantee to find the best alignment as it is aligns the entire sequences from beginning to end using whatever insertions or deletions are necessary [14]. Due to this advantage, there are researchers who are competing to produce a new improvement over dynamic programming. Study by Deusdado and Carvalho, Abu-Hashem *et al* and Gill and Singh are a few selection of current research that implement a new variant of dynamic programming for protein sequence alignment (Deusdado and Carvalho, 2008; Abu-Hashem *et al.*, 2011; Gill and Singh, 2011).

Deusdado and Carvalho (2008) proposed an algorithm which is based on distance series for optimal and near-optimal similarity discovery. In order to improve the processing time, their method included a filtering strategy in high score segments searched using a fast exact pattern-matching module. Performance test showed significant efficiency improvement over dynamic programming.

Abu-Hashem *et al* (2011) proposed a parallel design for dynamic programming which is an extension of N-Gram-Hirschberg (NGH) algorithm. This method is introduced in order to fasten the sequence alignment construction. It is divided into two levels and applied on two different architectures. From the experiment, the parallel algorithm showed in enhancement in the execution time but the speedup was a bit slow.

Gill and Singh (2011) proposed combination of dynamic programming with Fuzzy Logic. Fuzzy logic is used to measure the similarity of sequence based on fuzzy parameter. The score then calculated using dynamic programming. The experiment done by Gill and Singh showed the algorithm based on Fuzzy Logic is an efficient method.

Conclusion and Future Works:

Protein sequence alignment is one of complex biological questions in Bioinformatics field. It is a process to find the similarities among proteins under study which discovered the functional, evolution and structural information. One of the popular algorithms that have been applied to the sequence alignment is dynamic programming. Dynamic programming was introduced in 1970 to perform a pairwise global alignment by Needleman and Wunsch. This technique has three main steps such as initialization, matrix fill and traceback. Dynamic programming is guaranteed to find the optimal alignment. However, it has two major issues. It is impractical for a large number of sequences (took more time in the preprocessing stages) and cannot present the theory of evolution (used simple scoring scheme). Due to these issues, several new approaches and meaningful scoring scheme were introduced as well as the improvement over dynamic programming. The new variants of dynamic programming promise better algorithm which provide the optimal alignment that reflect biological data in appropriate processing time. However, detailed study of these algorithms is still needed to ensure those methods are able to accommodate the growing biological data. The study should focus not only the alignment process but also the suitable hardware used that support the process.

REFERENCES

Abu-Hashem, M.A., N.A. Abdul Rashid and R. Abdullah, 2011. "A Hybrid Distributed and Shared Memory Method for Fast HNGH Algorithm", International Journal of Advanced Computer Science, 1(6): 233-239.

Antler, C., 2012. "Investigating the cellular machinery: protein identification", Retrieved September 20, 2012, from http://www.scq.ubc.ca/investigating-the-cellular-machinery-protein-identification/BAliBASE. "Source of protein sequence" (online), Retrieved September 12, 2012, from http:/ /www-bio3d-igbmc.u-strasbg.fr/setminus balibase

Chao, K.M., 2006. "Basic Concepts of DNA, Proteins, Genes and Genomes", National Science Council, Taiwan.

Deusdado, S.A.D. and P.M.M. Carvalho, 2008. "SimSearch: A new variant of dynamic programming based on distance series for optimal and near-optimal similarity discovery in biological sequences", PRODEP. Genome Research Institute, 2013.

Gill, N. and S. Singh, 2011. "Biological Sequence Matching Using Fuzzy Logic", International Journal of Scientific & Engineering Research, 2(7).

Gill, N. and S. Singh, 2011." Multiple Sequence Alignment using Boolean Algebra and Fuzzy Logic: A Comparative Study", Int. J. Tech. Appl., 2(5): 1145-1152.

Gotoh, O., 1982. "An improved algorithm for matching biological sequences", J. Mol. Biol.

Henikoff, S. and J.G. Henikoff, 1991. "Automated assembly of protein blocks for database searching.", Nucleic Acids.

Henikoff, S. and J.G. Henikoff, 1992. "Amino acid substitution matrices from protein blocks", Proc Natl Acad Sci USA.

Henikoff, S. and J.G. Henikoff, 1993. "Performance evaluation of amino acid substitution matrices", Proteins.

Manohar, P. and S. Singh, 2012. "Protein Sequence Alignment: A Review", World Applied Programming, 2(3): 141-145.

Do, C.B. and K. Katoh, 2002. "Protein Multiple Sequence Alignment", Methods in Molecular Biology, 484.

Needleman, S.B. and C.D. Wunsch, 1970. "A general method applicable to the search for similarities in the amino acid sequence of two proteins", Journal of Molecular Biology, 48: 443-453.

Oladele, T.O., O.M. Bamigbola and C.O. Bewaji, 2009. "On Efficiency of sequence alignment algorithms", African Scientist, 10(1): 9-14.

Rouchka, E.C., 2006. "Aligning DNA Sequences Using Dynamic Programming", ACM New York, NY, USA: 9-9.

Sharma, K.R., 2009. "Bioinformatics Sequence Alignment and Markov Models", Mc Graw Hill.

Thomson, J.D., F. Plewniak and O. Poch, 1999. "BAliBASE: a benchmark alignment database for the evaluation of multiple alignment programs", Bioinformatics, 15(1): 87-88.

Weimann, O., 2009. "Accelerating Dynamic Programming", Massachusetts Institute of Technology, 136.

Zeti, A.M.H., M.S. Shamsir, K. Tajul-Arifin, A.F. Merican, R. Mohamed, S. Nathan, N.M. Mahadi, S. Napis and T.W. Tan, 2009. "Bioinformatics in Malaysia: Hope, Initiative, Effort, Reality, and Challenges", PLOS Computational Biology, 5(8).