

© Imprinted Paper

A Tool for Pair-Wise Alignment Algorithm

Allam Appa Rao
College of Engineering (Autonomous)
Andhra University
Visakhapatnam, India
Email: allamapparao@gmail.com

Management Review: An International Journal

Volume 2, Number 1, June 30, 2007

Pages: 28-40

ISSN: 1975-8480

Management Review: An International Journal

Management Review: An International Journal
Volume 2, Number 1, June 30, 2007
Pages: 28-40
ISSN: 1975-8480

Sang Hyung Ahn
Editor-In-Chief
Seoul National University
© 2007 by INFORMS Korea Chapter
Email: KINFORMS@Korea.com

A Tool for Pair-Wise Alignment Algorithm

Allam Appa Rao
College of Engineering (Autonomous)
Andhra University
Visakhapatnam, India
Email: allamapparao@gmail.com

ABSTRACT

Bioinformatics is the application of computational techniques to the management and analysis of biological information. The relationship between a query sequence, commonly termed as probe and other sequence, known as subject can be quantified and their similarity can be assessed. This similarity can be used to identify the evolutionary relationship between a newly determined sequence and a known gene family. When the degree of similarity is low, the relationship must remain competitive, until evidence has been gathered. The purpose of this research is to implement various methods for Pair-Wise alignment techniques and design a tool with a good user -interface for aligning two sequences and output the score of the alignment. This research uses the various pair wise alignment algorithms, like Needleman-Wunsch global alignment, Smith-Waterman algorithm to find the optimum alignment (including gaps) of two sequences. Dynamic programming methods ensure the optimal global alignment by exploring all possible alignments and choosing the best. The user will be provided a good user-interface for giving the input sequences and opting for the required algorithm. The algorithm uses the BLOSUM50 substitution matrix for computation, and outputs the Functional matrix(F-matrix), Optimal Alignment and the score. The total alignment score is

calculated as a function of the identity between the aligned residues and the gap penalties incurred. The input sequences can be taken from a file stored on the disk.

Keywords: *Bioinformatics, Pair-Wise, BLOSUM50, Needleman-Wunsch, Smith- Waterman, Dynamic Programming*

INTRODUCTION

Determination of protein/peptide sequences is a basic requirement for biomedical research, including cancer research. It is absolutely essential for characterising and identifying proteins or peptides. Imagine you are a Biologist, who has discovered an unknown peptide, perhaps theoretically translated from a nucleotide sequence, or isolated from a gel, which can be sequenced. In this process is to look for similarities with already discovered peptide sequences/proteins. The purpose of this research is to implement various methods for pair-wise alignment techniques and design a tool with a good user -interface for aligning two sequences and output the score of the alignment. The sequence alignment is a linear comparison of amino-acid sequence in which insertions are made in order to bring equivalent positions in adjacent sequences into the correct register.

BACKGROUND

Most biological databases consist of long strings of nucleotides (guanine, adenine, thymine, cytosine and uracil) and/or amino acids (threonine, serine, glycine, etc.). Each sequence of nucleotides or amino acids represents a particular gene or protein (or section thereof), respectively. Sequences are represented in

shorthand, using single letter designations. This decreases the space necessary to store information and increases processing speed for analysis. While most biological databases contain nucleotide and protein sequence information, there are also databases which include taxonomic information such as the structural and biochemical characteristics of organisms. The power and ease of using sequence information has however, made it the method of choice in modern analysis. Not only can computers be used to store and organize sequence information into databases, but they can also be used to analyze sequence data rapidly. The evolution of computing power and storage capacity has, so far, been able to outpace the increase in sequence information being created. Theoretical scientists have derived new and sophisticated algorithms which allow sequences to be readily compared using probability theories. These comparisons become the basis for determining gene function, developing phylogenetic relationships and simulating protein models. The physical linking of a vast array of computers in the 1970's provided a few biologists with ready access to the expanding pool of sequence information. This web of connections, now known as the Internet, has evolved and expanded so that nearly everyone has access to this information and the tools necessary to analyze.

IMPLEMENTATION PROCESS

The implementation part has three phases. They are

1. Input

The two sequences are read from the user and are feed to the selected algorithm. The sequences may be taken from a file stored on the disk also.

2. Alignment

The sequences are aligned for global and local alignment using Needle-man and Wunsch algorithm, Smith-Waterman algorithm, repeated matches with simple gap costs, and overlap matches with simple gap costs. The functional matrices are computed along with the score and optimal alignment.

3. Output

The optimal alignment, score and the functional matrices are printed on their respective fields.

ALIGNMENT ALGORITHMS

Given a scoring scheme, we need to have an algorithm that computes the highest-scoring alignment of two sequences. We will discuss alignment algorithms based on dynamic programming. Dynamic programming algorithms play a central role in computational sequence analysis. They are guaranteed to find the optimal scoring alignment. However, for large sequences they can be too slow and heuristics (such as BLAST, FASTA, MUMMER etc) are then used that usually perform very well, but will miss the best alignment for some sequence pairs. Depending on the input data, there are a number of different variants of alignment that are considered, among them global alignment, local alignment and overlap alignment. We will use two short amino acid sequences for illustration: HEAGAWGHEE and PAWHEAE. To score the alignment we will use the BLOSUM50 matrix and a gap cost of $d = 8$. (Later, we will also use affine gap costs.) Here they are arranged to show a matrix of corresponding BLOSUM50 values:

Table 1. A Matrix of Corresponding BLOSUM50 Values

	H	E	A	G	A	W	G	H	E	E
P	-2	-1	-1	-2	-1	-1	-2	-2	-1	-1
A	-2	-1	5	0	5	-1	0	-2	-1	-1
W	-3	-3	-3	-3	-1	15	-3	-3	-3	-3
H	10	0	-2	-2	-2	-1	-3	10	0	0
A	-2	-1	-5	0	5	-3	0	-2	-1	-1
E	0	6	-1	-3	-1	-3	-3	0	6	0

Gap penalties

Gaps are undesirable and thus penalized. The standard cost associated with a gap of length g is given either by a linear score.

Or an affine score

$$y(g) = -g^d$$

$$y(g) = -d \cdot (g-1)e,$$

where d is the gap open penalty and e is the gap extension penalty. Usually, $e < d$ with the result that less isolated gaps are produced, as shown in the following comparison:

Global alignment: Needleman-Wunsch algorithm

For obtaining the best global alignment of two sequences, the Needleman-Wunsch algorithm is applied as a dynamic program to solve this program.

Idea: Build up an optimal alignment using previous solutions for optimal alignments of smaller substrings.

Given two sequences $x = (x_1, x_2, \dots, x_j)$ and $y = (y_1, y_2, \dots, y_j)$. We will compute a matrix. $F: \{1, 2, \dots, n\} \times \{1, 2, \dots, m\} \Rightarrow \mathbb{R}$, in which $F(i, j)$ equals the best score of the alignment of the two prefixes (x_1, x_2, \dots, x_i) and (y_1, y_2, \dots, y_j) . This will be done recursively by setting $F(0,0) = 0$ and then computing $F(i, j)$ from $F(i-1, j-1)$, $F(i-1, j)$ and $F(i, j-1)$.

The Recursion

There are three ways in which an alignment can be extended up to (i, j) . We obtain $F(i, j)$ as the largest score arising from these three options. This is applied repeatedly until the whole matrix $F(i, j)$ is filled with values. To complete the description of the recursion, we need to set values of $F(i, 0)$ and $F(0, j)$ for $i \neq 0$ for $j \neq 0$. The final value $F(n, m)$ contains the score of the best global alignment between x and y . To obtain an alignment corresponding to this score, we must find the path of choices that the recursion made to obtain the score. This is called a trace back.

Algorithm

Input: two sequences X and Y
Output: optimal alignment and score α
Initialization:
 Set $F(i, 0) := -i \cdot d$ for all $i = 0, 1, 2, \dots, n$
 Set $F(0, j) := -j \cdot d$ for all $j = 0, 1, 2, \dots, m$
For $i = 1, 2, \dots, n$ **do:**
 For $j = 1, 2, \dots, m$ **do:**
 Set $F(i, j) := \max \begin{cases} F(i-1, j-1) + s(x_i, y_j) \\ F(i-1, j) - d \\ F(i, j-1) - d \end{cases}$
 Set backtrace $T(i, j)$ to the maximizing pair (i', j')
 The score is $\alpha := F(n, m)$
 Set $(i, j) := (n, m)$
repeat
 if $T(i, j) = (i-1, j-1)$ **print** $\begin{pmatrix} x_i \\ y_j \end{pmatrix}$
 else if $T(i, j) = (i-1, j)$ **print** $\begin{pmatrix} x_i \\ _ \end{pmatrix}$ **else print** $\begin{pmatrix} _ \\ y_j \end{pmatrix}$
 Set $(i, j) := T(i, j)$
until $(i, j) = (0, 0)$.

Complexity

We need to store $(n+1) \times (m+1)$ numbers. Each number takes a constant number of calculations to compute: three sums and a max. Hence, the algorithm requires $O(nm)$ time and memory. For biological sequence analysis, we prefer algorithms that have time and space requirements that are linear in the length of the sequences. Quadratic time algorithms are a little slow, but feasible. $O(n^3)$ algorithms are only feasible for very short sequences. Something to think about: if we are only interested in the best score, but not the actual alignment, then it is easy to reduce the space requirement to linear.

Local alignment: Smith-Waterman algorithm:

Global alignment is applicable when we have two similar sequences that we want to align from end-to-end, e.g. two homologous genes from related species. Often, however, we have two sequences x and y and we would like to find the best match between substrings of both. For example, we may want to find the position of a fragment of DNA in a genomic sequence. The best scoring alignment of two substrings of x and y is called the best local alignment. The Smith-Waterman local alignment algorithm is obtained by making two simple modifications to the global alignment algorithm.

In the main recursion, we set the value of $F(i, j)$ to zero, if all attainable values at position (i, j) are negative:

$$F(i, j) = \max \left\{ \begin{array}{l} 0, \\ F(i-1, j-1) + s(x_i, y_j), \\ F(i-1, j) - d, \\ F(i, j-1) - d. \end{array} \right.$$

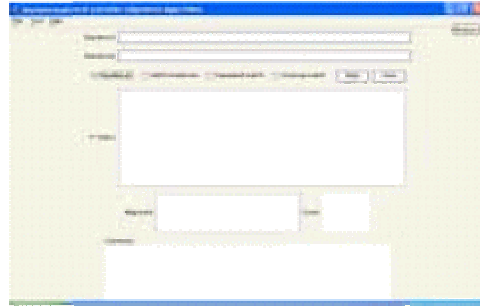
The value $F(i, j) = 0$ indicates that we should start a new alignment at (i, j) . This is because, if the best alignment up to (i, j) has a negative score, then it is better to start a new one, rather than to extend the old one. Note that, in particular, we have $F(i, 0) = 0$ and $F(0, j) = 0$ for all $i = 0, 1, 2, \dots, n$ and $j = 0, 1, 2, \dots, m$.

Instead of starting the trace back at (n, m) , we start it at the cell with the highest score, $\text{argmax } F(i, j)$. The trace back ends upon arrival at a cell with score 0, which corresponds to the start of the alignment. For this algorithm to work, we require that the expected score for a random match is negative, i.e. that where q_a and q_b are the probabilities for seeing the symbol a or b at any given position, respectively. Otherwise, matrix entries will tend to be positive, producing long matches between random sequences.

Algorithm

Input: two sequences X and Y
Output: optimal alignment and score α
Initialization:
 Set $F(i, 0) := -j \cdot d$ for all $i = 0, 1, 2, \dots, n$
 Set $F(0, j) := -j \cdot d$ for all $j = 0, 1, 2, \dots, m$
For $i = 1, 2, \dots, n$ **do:**
 For $j = 1, 2, \dots, m$ **do:**
 Set $F(i, j) := \max \begin{cases} F(i-1, j-1) + s(x_i, y_j) \\ F(i-1, j) - d \\ F(i, j-1) - d \end{cases}$
 Set backtrace $T(i, j)$ to the maximizing pair (i', j')
 The score is $\alpha := F(n, m)$
 Set $(i, j) := (n, m)$
repeat
 if $T(i, j) = (i-1, j-1)$ **print** $\begin{pmatrix} x_i \\ y_j \end{pmatrix}$
 else if $T(i, j) = (i-1, j)$ **print** $\begin{pmatrix} x_i \\ - \end{pmatrix}$ **else print** $\begin{pmatrix} - \\ y_j \end{pmatrix}$
 Set $(i, j) := T(i, j)$
until $(i, j) = (0, 0)$.

Results



The User -Interface.

The following sequences were taken as an example for implementing the algorithm.

SEQUENCE 1:

```
AGGCTCAGAACGCGTCCAGAAATCAGGGGAAGGAGACCCCTAT
CTGTCCTTCTTCTGGAAGAG CTGGAAA
```

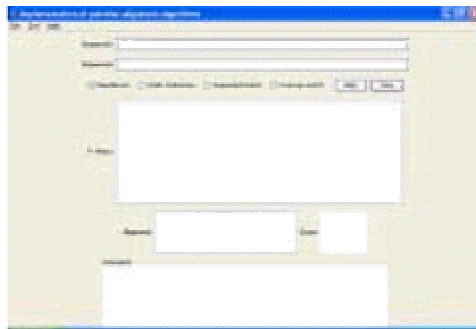
SEQUENCE 2:

```
ATGGGTGACT GGGGCTTCCT GGAGAAGTTG CTGGACCAGG
CCAGGAGCA CTCGACCGTG
```

The output screens for various algorithms are shown below.

- a) Optimal Alignment Using Needleman - Wunch Algorithm:
- b) Optimal Alignment Using Smith – Waterman Algorithm
- c) Optimal Alignment Using Repeated - Match Algorithm:
- d) Optimal Alignment Using Overlap - match algorithm:

Clearing the previous sequences and their results. A predefined example



CONCLUSION AND FUTURE RESEARCH

Bio-Informatics is the study of complex biological information using computational techniques. The role of the computers in Bio-Informatics is required for their processing speed of complex data and for their problem solving power. Bio-Informatics include Molecular Biology, Bio-Physics and Computer Science, Mathematics and Statistics. This requires solidarity among Biologists, Mathematicians, Engineers and Computer scientists to provide effective solutions for scientific problems. Accelerating Biological research projects using computer databases and algorithms. In this project we have implemented the various pair-wise alignment algorithms like Needleman and Wunsch algorithm, Smith-Waterman algorithm and designed a tool for the user through which he can input the sequences that are to be aligned, select a particular algorithm and compute the optimal alignment along with the functional matrix and score. The results are displayed on the

screen.

REFERENCES

- Altschul, S., F. Stephen, L.M. Thomas, A.A. Schaffer, J. Zhang, Z.Zhang, W. Miller, and D.J. Lipman (1997) Gapped BLAST and PSI-BLAST: A New Generation of Protein Database Search Programs, *Nucleic Acids Research*, 25: 3389-3402.
- Bilu, Y., P. K. Agarwal, and R. Kolodny (2006) Faster algorithms for optimal multiple sequence alignment based on pairwise comparisons, *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 3(4): 408-422.
- Bonizzoni, P. and G. Delia Vedova (2001) The Complexity of MultipleSequence Alignment with Sp-Score that Is a Metric, *Theoretical Computer Science*, 259(1-2): 63-79.
- Eppstein, D., Z. Galil, R. Giancarlo, and G.F. Italiano (1992) Sparse Dynamic-Programming: I. Linear Cost-Functions, *ACM*, 39(3): 519-545.
- Huang, Y-M and C. Bystroff (2005) Improved pairwise alignment of proteins in Twilight structure predictions, *Proceedings of the 2005 IEEE Computational Systems Bioinformatics Conference Workshops (CSBW'05)*.
- Naveed, T., I. S. Siddiqui, S. Ahmed, Parallel Needleman-Wunsch Algorithm for Grid. Available <http://www.gridbus.org/~alchemi/files/Parallel%20Needleman%20Algo.pdf>
- Rao, Allam Appa, G. Prakash Gupta, M. Rajesh Babu, P. Sateesh Chandra, D.V. Phaneendra Teja,(2006) A Java Based Tool For Implementing The Pair-Wise Alignment Algorithms” Working Paper.
- Thorne, J. L, H. Kishino, J. Felsenstein (1991) An Evolutionary Model for Maximum Likelihood Alignment of DNA Sequences. *Journal of Molecular Evolution*, 33:114-124.
- Wang, Bin (2002) “Implementation of a dynamic programming algorithm for DNA Sequence alignment on the Cell Matrix Architecture “[online], Utah State University, Logan, Utah.

Available:

<http://www.cellmatrix.com/entryway/products/pub/wang2002.pdf>

Whelan, S, P. Lio, and N. Goldman (2001) Molecular phylogenetics: state-of-the-art methods for looking into the past. *Trends in Genetics*, 17: 262-272.

Appendix:

Sample Project Code in Java

```
import java.io. *; import
java.awt. *; import java.awt.
event. *; import
javax.swing. *; import
java.util. *; abstract class
Substitution { public
int score;

void buildscore(String residues,
int[][] residuescores) { // Allow
lowercase and uppercase
residues (ASCII code <= 127):
score = new int[127][127]; for
(int i=0; i<residues.length();
i++) { char res 1 =
residues.charAt(i);
for(int j=0; j<=i; j++){ char
res2 = residues.charAt(j);
score[res1][res2] =
score[res2][res1]
= score[res1][res2+32] =
```

```
score[res2+32][res1] =  
score[res1+32][res2] =  
score[res2][res1+32] =  
score[res1+32][res2+32] =  
score[res2+32][res1+32] =  
residuescores[i][j]; }  
  
abstract public String getResiduesQ; }  
class BlosumSO extends Substitution {  
    private String residues = "ARNDCQEGHILKMFPSTWYV";  
    public String getResiduesQ  
    { return residues; }  
}
```

Management Review: An International Journal

Editorial Policy

Management Review: An International Journal (MRIJ) publishes intellectual findings to academics and practitioners in profit and non-profit organizations as well as local and global institutions on all aspects of managerial issues. MRIJ promotes the findings of sharing knowledge, exchanging experience and creating new ideas between academics and practitioners. MRIJ encourages all manuscripts of multi-disciplinary and cross-functional approaches with theoretical and empirical, technical and non-technical, and cases studies related to managerial issues in certain individual organizations, societies, countries.

Manuscript Submission

Your manuscript should be original contents that are not copyrighted, published, accepted for publication by any other journal, or being reviewed to any other journal while being reviewed by the Journal. Your manuscripts should be formatted with Century 12 points, double-spaced, left-aligned, 2.5 inches of top, 1.5 left and right, and 2 bottom margins on international standard (letter) size. The manuscript size may be between seven and fifteen pages. Manuscripts should follow generally accepted manuscripts printing guidelines. All manuscripts should be electronically submitted to the managing editor at kinforms@korea.com.

Management Review: An International Journal

Editor-In-Chief

Sang Hyung Ahn

Seoul National University

Managing Editor

Chang Won Lee

Jinju National University

Associate Editors

Soo Wook Kim

Seoul National University

Jae Hyung Min

Sogang University

Dennis G. Severance

University of Michigan-Ann Arbor

Seung Chul Kim

Hanyang University

David C. J. Ho

Oklahoma State University

Sungmin Kang

The Catholic University of Korea

Chang Whan Lee

Ajou University

Herbert Meyr

Technical University of Darmstadt

Dong Seol Shin

KIMI

Sung Ku Cho

Dongguk University

Marc Schniederjans

University of Nebraska-Lincoln

Joong-Soon Kim

Keimyung University

Zhao Xiande

Chinese University of Hong Kong

Kwang Tae Park

Korea University

Naoki Ando

Nagasaki Prefectural University

Bowon Kim

KAIST

Do Hoon Kim

Kyunghee University

Copyright © Management Review: An International Journal

INFORMS Korea Chapter

Seoul National University

56-1 Shilim Dong, Kwanak Ku

Seoul 151-742, Korea

Printed in Korea

ISSN: 1975-8480 · Volume 2 · Issue 1 · Summer 2007