Alignment Algorithm for Predicting Three Dimensional Model: Identification of Anti-Stress Compounds Through In silico Studies

K Namitha1, Dr. Kunjam Nageswara Rao2

1, 2Department of CSSE, AU College of Engineering, Andhra University, A.P. India.

Abstract: Stress is a term that is commonly used today but has become increasingly difficult to define. It shares, to some extent, common meanings in both the biological and psychological sciences. Some medicinal plants, and herbal agents possessing anti-stress activity. For this study the medicinal plants used are Celastrus paniculatus, Withania somnifera, Convolvulus pluricaulis, Rauwolfia Serpentina. Among many terpenoids, flavones and alkaloids were identified which have anti-stress property. These were screened and docked to the protein like 5 Hydroxytryptamine receptor, Aryl Hydrocarbon receptor and Potassium voltage gatted channels which are expressed in Stress. Everyone knows that medicinal plants have disease curing properties and this is due to the compounds present in the extracts used for the treatment. So the identified compounds of different medicinal plants are used as medicine for Stress in Ayurvedha from previous literature. After identification using chemsketch software these compounds were designed and screened for anti-stress property. These were screened for the Anti-stress activity. The proteins responsible for stress in Homo sapiens were collected using uniprot. In this work we developed an algorithm for the alignment of stress proteins, which has two classes, the first one named Dynamic Programming and the second named Cell. The first class contains three methods that describe the steps of dynamic programming algorithm. The first method is named Initialization_Step, this method prepares the matrix a[i,j] that holds the similarity between arbitrary prefixes of the two sequences. The algorithm starts with shorter prefixes and uses previously computed results to solve the problem for larger prefixes. The second method named Get_Maxcomputes the value of the cell (j,i) by the Equation 1. The third method is named Traceback_Step. This method produced the alignment by traversing the cell matrix (N-1,M-1) back towards the initial entry of the cell matrix (1,1). Using this algorithm we aligned stress proteins to which three dimensional model structures has to be developed. The biological datasets were be selected for Phylogenetics studies using algorithm. Stress proteins structure were generated using MODELLER9V7 software. With the aid of the molecular mechanics and molecular dynamics methods, the final model is obtained and further assessed by PROCHECK and Verify 3D graph programs, which showed that the final refined model is reliable. Active sites were identified and used for docking with the compounds. Then the compounds were docked to the 5 Hydroxytryptamine receptor, Aryl Hydrocarbon receptor and Potassium voltage gatted channels in order find better inhibitor. Among 33 compounds 10 compounds showed best docking results for each protein.

Key words: Anti stress, Docking, Dynamic programming, Medicinal plants, Aryl hydrocarbon receptor.

Introduction

Very short or very similar sequences can be aligned by hand. However, most interesting problems require the alignment of lengthy, highly variable or extremely numerous sequences that cannot be aligned solely by human effort [1]. Instead, human knowledge is applied in constructing algorithms to produce high-quality sequence alignments, and occasionally in adjusting the final results to reflect patterns that are difficult to represent algorithmically (especially in the case of nucleotide sequences). Computational approaches to sequence alignment generally fall into two categories: global alignments and local alignments. Calculating a global alignment is a form of global optimization that "forces" the alignment to span the entire length of all query sequences [2]. By contrast, local alignments identify regions of similarity within long sequences that are often widely divergent overall. Local alignments are often preferable, but can be more difficult to calculate because of the additional challenge of identifying the regions of similarity. A variety of computational algorithms have been applied to the sequence alignment problem. These include slow but formally correct methods like dynamic programming. These also include efficient, heuristic algorithms or probabilistic methods designed for large-scale database search, that do not guarantee to find best matches [3].

Alignments are commonly represented both graphically and in text format. In almost all sequence alignment representations, sequences are written in rows arranged so that aligned residues
appear in successive columns. In text formats, aligned columns containing identical or similar characters are indicated with a system of conservation symbols. An asterisk or pipe symbol is used to show identity between two columns, other less common symbols include a colon for conservative substitutions and a period for semiconservative substitutions. Many sequence visualization programs also use color to display information about the properties of the individual sequence elements in DNA and RNA sequences, this equates to assigning each nucleotide its own color. In protein alignments, color is often used to indicate amino acid properties to aid in judging the conservation of a given amino acid substitution. For multiple sequences the last row in each column is often the consensus sequence determined by the alignment; the consensus sequence is also often represented in graphical format with a sequence logo in which the size of each nucleotide or amino acid letter corresponds to its degree of conservation [4].

Sequence alignments can be stored in a wide variety of text-based file formats, many of which were originally developed in conjunction with a specific alignment program or implementation. Most web-based tools allow a limited number of input and output formats, such as FASTA format and GenBank format and the output is not easily editable. Several conversion programs that provide graphical and/or command line interfaces are available, such as READSEQ and EMBASS. There are also several programming packages which provide this conversion functionality, such as BioPerl and BioRuby [5,6].

1. METHODOLOGY

Sequence alignment is the procedure of comparing two (pair-wise alignment) or more (multiple alignment) sequences by searching for a series of characters that are in the same order in all sequences. Two sequences can be aligned by writing them across a page in two rows. Identical or similar characters are placed in the same column, and non identical ones can either be placed in the same column as a mismatch or against a gap (-) in the other sequence. Sequences that are aligned in this manner are said to be similar. Sequence alignment is useful for discovering functional, structural, and evolutionary information in biological sequences. Consider the following DNA Sequences GACGGATTAG and GATCGGAATAG. Notice that when we align them one above the other:
GA-CGGATTAG
GATCGGAATAG

The only differences are marked with colors in the above sequences. Observe that the gap (-) is introduced in the first sequence to let equal bases align perfectly. The goal of this article is to present an efficient algorithm that takes two sequences and determine the best alignment between them. The total score of the alignment depends on each column of the alignment. If the column has two identical characters, it will receive value +1 (a match). Different characters will give the column value -1 (a mismatch). Finally a gap in a column drops down its value to -2 (Gap Penalty). The best alignment will be one with the maximum total score. The above alignment will give a total score: 9 × 1 + 1 × (-1) + 1 × (-2) = 6.

These parameters match, mismatch and gap penalty can be adjusted to different values according to the choice of sequences or experimental results. One approach to compute similarity between two sequences is to generate all possible alignments and pick the best one. However, the number of alignments between two sequences is exponential and this will result in a slow algorithm so, Dynamic Programming is used as a technique to produce faster alignment algorithm. Dynamic Programming tries to solve an instance of the problem by using already computed solutions for smaller instances of the same problem. Giving two sequences Seq1 and Seq2 instead of determining the similarity between sequences as a whole, dynamic programming tries to build up the solution by determining all similarities between arbitrary prefixes of the two sequences. The algorithm starts with shorter prefixes and uses previously computed results to solve the problem for larger prefixes.

Let M = size of Seq1 and N = size of Seq2 , the computation is arranged into an (N+1) × (M+1) array where entry (j,i) contains similarity between Seq2[1....j] and Seq1[1......i]. The algorithm computes the value for entry(j,i) by looking at just three previous entries:
(j-1,i-1) Diagonal Cell to entry (j,i)
(j-1,i) Above Cell to entry (j,i)
(j,i-1) Left Cell to entry (j,i)
The value of the entry \((j,i)\) can be computed by the following equation:

\[
\begin{align*}
\text{Equation (1.1)} \\
\{d_{j,i-1} \cdot \text{Gap} \} + \{d_{j-1,i-1} \cdot p(j,i)\}
\end{align*}
\]

where \(p(j,i)= +1\) if \(\text{Seq2}[j]=\text{Seq1}[i]\) (match Score) and \(p(j,i)= -1\) if \(\text{Seq2}[j]\neq\text{Seq1}[i]\).

The maximum value of the score of the alignment located in the cell \((N-1,M-1)\) and the algorithm will trace back from this cell to the first entry cell \((1,1)\) to produce the resulting alignment. If the value of the cell \((j,i)\) has been computed using the value of the diagonal cell, the alignment will contain the \(\text{Seq2}[j]\) and \(\text{Seq1}[i]\). If the value has been computed using the above cell, the alignment will contain \(\text{Seq2}[j]\) and a Gap ('-') in \(\text{Seq1}[i]\). If the value has been computed using the left cell, the alignment will contain \(\text{Seq1}[i]\) and a Gap ('-') in \(\text{Seq2}[j]\). The resulting alignment will produce completely by traversing the cell \((N-1,M-1)\) back towards the initial entry of the cell \((1,1)\).

### 2.1 Selection of Stress expressed proteins:

Selection of Stress expressed proteins from NCBI and Swiss-prot and searched for structures using BLAST. The best model was selected from Pdb. Then the selected 5 Hydroxytryptamine receptor, Aryl Hydrocarbon receptor and Potassium voltage gatted channels proteins were modified by removing unnecessary chains. Then the required chain was selected for SPDBV software. Then for the selected chain active site was identified.

### 2.2 Active site Identification

Active site of 5 Hydroxytryptamine receptor, Aryl Hydrocarbon receptor and Potassium voltage gatted channels were identified using CASTp server. A new program, CASTp, for automatically locating and measuring protein pockets and cavities, is based on precise computational geometry methods, including alpha shape and discrete flow theory. CASTp identifies and measures pockets and pocket mouth openings, as well as cavities. The program specifies the atoms lining pockets, pocket openings, and buried cavities; the volume and area of pockets and cavities; and the area and circumference of mouth openings.

### 2.3 Docking method

Docking with GOLD 3.0.1

GOLD (Genetic Optimization of Ligand Docking) a genetic algorithm (GA) based software, mainly utilizes an evolutionary strategy involving 3 genetic operators; cross overs, mutations and migrations. GOLD imports the partial flexibility to proteins and full flexibility to inhibitors. The compounds are docked into the active site of angiotensin, calcium channel and cholesterol absorption inhibitor and the interaction of these ligands with the active site residues are thoroughly studied using calculations of molecular mechanics. The parameters used for GA were population size (100), selection pressure (1.1), number of operations (10,000), number of island (1) and niche size. Operator parameters for crossover, mutation and migration were set to 100, 100 and 10 respectively. Default cutoff values of 3.0Å (dH-X) for hydrogen bonds and 6.0Å for vanderwaals were employed. The default algorithm speed was selected and the inhibitor binding site in angiotensin, calcium channel and cholesterol absorption inhibitor were defined within a 10Å radius with the centroid as HH atom of TYR51, HIS13, GLN8 respectively. The number of poses for each inhibitor was set 100, and early termination was allowed if the top three bound conformations of inhibitors were within 1.5Å RMSD. After docking, the individual binding poses of each inhibitor were observed and their interactions with the protein were studied. The best and most energetically favorable conformation of each inhibitor was selected.

### 2.3.1 GOLD Score fitness function

The four components \(\text{vig}, \text{Protein-ligand hydrogen bond energy (external H-bond)}; \text{Protein-ligand vanderwaals energy (external vdw)}; \text{Ligand internal vanderwaals energy (internal vdw)}; \text{Ligand intramolecular hydrogen bond energy (internal H-bond)}\) were considered for calculating the fitness function of GOLD score. The protein-ligand hydrophobic contact was encouraged by making an empirical correction by multiplying external vdw score with 1.375. The fitness function has been optimized for the prediction of ligand binding positions.
Gold Score = S (hb_ext) + S (vdw_ext) + S (hb_int) + S (vdw_int)

Where S (hb_ext) is the protein-ligand hydrogen bond score, S (vdw_ext) is the protein-ligand vanderwaals score, S (hb_int) is the score from intramolecular hydrogen bond in the ligand and S (vdw_int) is the score from intramolecular strain in the ligand.

3. RESULTS AND DISCUSSION

Using the Code

Code has two classes, the first one named Dynamic Programming and the second named Cell.

The first class contains three methods that describe the steps of dynamic programming algorithm. The first method is named Initialization_Step, this method prepares the matrix a[i,j] that holds the similarity between arbitrary prefixes of the two sequences. The algorithm starts with shorter prefixes and uses previously computed results to solve the problem for larger prefixes.

```csharp
public static Cell[,] Initialization_Step(string Seq1, string Seq2, int Sim, int NonSimilar, int Gap)
{
    int M = Seq1.Length; // Length + 1 // - AAA
    int N = Seq2.Length; // Length + 1 // - AAA
    Cell[,] Matrix = new Cell[N, M];
    // Initialize the first Row With Gap Penalty
    for (int i = 0; i < Matrix.GetLength(1); i++)
    {
        Matrix[0, i] = new Cell(0, i, i*Gap);
    }
    // Initialize the first Column With Gap Penalty
    for (int i = 0; i < Matrix.GetLength(0); i++)
    {
        Matrix[i, 0] = new Cell(i, 0, i*Gap);
    }
    // Fill Matrix with each cell has a value result from method Get_Max
    for (int j = 1; j < Matrix.GetLength(0); j++)
    {
        for (int i = 1; i < Matrix.GetLength(1); i++)
        {
            Matrix[j, i] = Get_Max(i, j, Seq1, Seq2, Matrix, Sim, NonSimilar, Gap);
        }
    }
    return Matrix;
}
```

The second method named Get_Max computes the value of the cell (j,i) by the Equation 1.1.

```csharp
public static Cell Get_Max(int i, int j, string Seq1, string Seq2, Cell[,] Matrix, int Similar, int NonSimilar, int GapPenality)
{
    Cell Temp = new Cell();
    int Sim;
    int Gap = GapPenality;
    if (Seq1[i] == Seq2[j])
    { Sim = Similar; }
    else
    { Sim = NonSimilar; }
    int M1, M2, M3;
    M1 = Matrix[j - 1, i - 1].CellScore + Sim;
    M2 = Matrix[j, i - 1].CellScore + Gap;
    M3 = Matrix[j - 1, i].CellScore + Gap;
    int max = M1 >= M2 ? M1 : M2;
    intMmax = M3 >= max ? M3 : max;
    if (Mmax == M1)
    { Temp = new Cell(j, i, M1, Matrix[j - 1, i - 1], Cell.PrevCellType.Diagonal); }
    else
    { if (Mmax == M2)
        { Temp = new Cell(j, i, M2, Matrix[j, i - 1], Cell.PrevCellType.Left); }
        else
        { if (Mmax == M3)
            { Temp = new Cell(j, i, M3, Matrix[j - 1, i], Cell.PrevCellType.Diagonal); }
            else
            { Temp = new Cell(j, i, M3, Matrix[j, i - 1], Cell.PrevCellType.Left); }
        }
    }
    return Temp;
}
```
The third method is named Traceback_Step. This method will produce the alignment by traversing the cell matrix(N-1,M-1) back towards the initial entry of the cell matrix (1,1).

```csharp
public static void Traceback_Step(Cell[,] Matrix, string Sq1, string Sq2, List<char> Seq1, List<char> Seq2)
{
    //List<char> Seq1 = new List<char>();
    //List<char> Seq2 = new List<char>();
    Cell CurrentCell = Matrix[Sq2.Length - 1, Sq1.Length - 1];

    while (CurrentCell.CellPointer != null)
    {
        if (CurrentCell.Type == Cell.PrevcellType.Diagonal)
        {
            Seq1.Add(Sq1[CurrentCell.CellColumn]);
            Seq2.Add(Sq2[CurrentCell.CellRow]);
        }
        if (CurrentCell.Type == Cell.PrevcellType.Left)
        {
            Seq1.Add(Sq1[CurrentCell.CellColumn]);
            Seq2.Add('-');
        }
        if (CurrentCell.Type == Cell.PrevcellType.Above)
        {
            Seq1.Add('-');
            Seq2.Add(Sq2[CurrentCell.CellRow]);
        }
        CurrentCell = CurrentCell.CellPointer;
    }
}
```

The second class in my code is named Cell. This class manipulates the cell of the matrix. Each cell has:

- A location indicated by the index of the row and index of the column
- A value that is represented by the score of the alignment
- A pointer to a previous cell that is used to compute the score of the current cell [Note: Pointer value "Diagonal, Above and Left"].

2. COMPOUND STRUCTURES

The anti stress compounds were identified from different medicinal plants like Celastrus paniculatus, Withania somnifera, Convolvulus pluricaulis, Rauvoifia Serpentina. These compounds were collected using pubchem and designed using chemsketch software. Totally 33 compounds were selected for screening. The structures are of the compounds are listed below and the properties were tabulated in Table 1.

Figure 1(1A-6A): Structures of the compounds identified from medicinal plants

1A: HYDROXYFLAVONE
2A: BAICALEIN
3A: CHRYSIN
4A: SCUTELLAREIN
5A: TANGERITIN
6A: WOGONIN

Table 1: Properties of the molecules

<table>
<thead>
<tr>
<th>Molecule</th>
<th>Molecular Formula</th>
<th>Molecular Weight</th>
<th>Molecular Refractivity cm³</th>
<th>Index of Refraction cm³</th>
<th>Density g/cc</th>
<th>Polarizability</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>C_{15}H_{10}O_{3}</td>
<td>238.23</td>
<td>66.08 ± 0.3</td>
<td>1.666 ± 0.02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>C_{15}H_{10}O_{3}</td>
<td>270.23</td>
<td>69.85 ± 0.3</td>
<td>1.732 ± 0.02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>C_{15}H_{10}O_{4}</td>
<td>254.23</td>
<td>67.97 ± 0.3</td>
<td>1.698 ± 0.02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>C_{15}H_{10}O_{6}</td>
<td>286.23</td>
<td>111.73 ± 0.3</td>
<td>1.767 ± 0.02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>C_{15}H_{10}O_{7}</td>
<td>432.42</td>
<td>114.34 ± 0.3</td>
<td>1.594 ± 0.02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>C_{17}H_{12}O_{3}</td>
<td>296.27</td>
<td>77.99 ± 0.3</td>
<td>1.674 ± 0.02</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

5 HYDROXYTRYPTAMINE RECEPTOR

Docking results

The concept of docking is important to determine the properties associated with protein-ligand interactions such as binding energy, electron distribution, hydrogen bond donor acceptor properties and hydrophobicity. In the present study, CASTp server was used to find the possible binding site of 5hydroxytryptamine receptor (Figure 2). From the binding site analysis it was observed that binding pockets are identified and the largest binding pocket was selected for the docking studies. The anti stress compounds were docked into proteins using GOLD 3.0.1 and all docking solutions for 5hydroxytryptamine receptor were ranked according to the GOLD fitness function. The docking results showed that all the Anti stress compounds are active 5hydroxytryptamine receptor inhibitors.

CAST P results

Active site Identification of 5Hydroxytryptamine receptor prediction by CASTp

After selecting receptor from PDB and isolated the A-chain in SPDBV, the possible binding sites of 5hydroxytryptamine receptor was searched based on the structural comparison of template and the model build and also with CASTp server and was shown in Figure 2 ,the residues are GLN8, GLU11, PHE12, MET72, LYS75, ASP78, THR79, SER81, GLU82, ILE85, ARG86, TYR138, VAL142.

Figure 3: shows Representing active site Pockets of the 5hydroxytryptamine receptor receptors shows highest area and volume.
Hydroxytryptamine receptor are given below and the docking energy values are tabulated in Table 2.

Fig 4 (1B-6B): Docking results of 5hydroxytryptamine receptor

<table>
<thead>
<tr>
<th>Ligand name</th>
<th>S(hb_ext)</th>
<th>S(vdw_ext)</th>
<th>S(hb_int)</th>
<th>S(int)</th>
</tr>
</thead>
<tbody>
<tr>
<td>methoxyquercetin</td>
<td>29.03</td>
<td>5.06</td>
<td>23.31</td>
<td>0</td>
</tr>
<tr>
<td>methoxycoumarin</td>
<td>22.15</td>
<td>0.00</td>
<td>17.36</td>
<td>0</td>
</tr>
<tr>
<td>anthrachinon</td>
<td>24.03</td>
<td>0.00</td>
<td>17.48</td>
<td>0.00</td>
</tr>
<tr>
<td>baicalin</td>
<td>27.16</td>
<td>0.00</td>
<td>23.54</td>
<td>0.00</td>
</tr>
<tr>
<td>campestrol</td>
<td>-59.71</td>
<td>0.00</td>
<td>21.25</td>
<td>0.00</td>
</tr>
<tr>
<td>cholestrol</td>
<td>-28.64</td>
<td>0.00</td>
<td>23.07</td>
<td>0.00</td>
</tr>
</tbody>
</table>

1B: 3 methoxy quercetin
2B: 6 methoxycoumarin
3B: Anthrachinon
4B: Anthrachinone
5B: Baicalin
6B: campestrol
29.86  6.00  21.02
0.00  -5.05  chrysin
28.59  0.15  24.87
0.00  -5.76  convolamine
30.35  11.11  20.03
0.00  -8.31  convolidine

5. CONCLUSION

Drugs with anti-stress activity need to be developed to avoid the serious consequences as a preventive curative measures. This analysis is very important in pathophysiology of the diseases caused by stress and to develop the medicine to manage the stress and its induced diseases. Docking results shows that out of 33 Compounds, randomly 10 compounds were shown best docking energy to the 5 Hydroxytryptamine receptor, Aryl Hydrocarbon receptor and Potassium voltage gatted channels proteins, By this we can say that the above docked compounds may show the antistress activity. These compounds were subjected to ADME studies to identify the activity. Toxicity and mutagenic studies are also performed. These can be synthesized further and used as antistress drugs in future.

6. REFERENCES