Chapter 7
Global Common Sequence Alignment Using Dynamic Window Algorithm

Lalit Kumar Behera

Abstract The DNA sequencing technology is evolving day by day. It results in the creation of large repository of sequence data. One important aspect of analyzing these sequence data is sequence alignment. In this research paper, a method of DNA sequence alignment with the help of a dynamic window algorithm has been discussed. The dynamic window will help in producing the comparative score of different alignment scheme resulting into the best acceptable alignments efficiently.

Keywords DNA · Sequence alignment · Dynamic algorithm

Introduction

In the past decade, a lot of research work has been done in the field of DNA sequence alignment. As a result, a large number of alignment algorithms and related software are available at present [1]. Basically, a sequence alignment problem deals with matching the nitrogen base of a sequence with a reference sequence. The processes help in finding out the similar functional regions that have great importance in the field of bioinformatics and its related field.

DNA Sequencing and Alignment

DNA is generally a long sequence of base pairs, polymer of repeating subunits called nucleotide. Each nucleotide contains five-carbon sugar, a phosphate group, and a base. Different bases are adenine (A), guanine (G), thymine (T), cytosine (C), and uracil (U). Nitrogen base combines with the five-carbon sugar to form nucleoside. Nucleoside in turn combines with phosphate groups to form nucleotide [2].

L. K. Behera (✉)
Department of Statistics, Center for IT, Utkal University, Odisha, India
e-mail: whomelalit@gmail.com

J. K. Mandal et al. (eds.), *Proceedings of the International Conference on Computing and Communication Systems*, Lecture Notes in Networks and Systems 24, https://doi.org/10.1007/978-981-10-6890-4_7

During the sequencing, DNA is represented as sequence of these nitrogen bases. Various methods are there for sequencing the DNA like Sanger's method, Maxam & Gilbert method, Pyro-sequencing, etc. [3]. Apart from these conventional methods, there are many other interdisciplinary approaches like graphical representation techniques [4], Voss representation, 2-bit binary, the 4-bit binary, the paired nucleotide, the 12-letter alphabet, the digital Z-signals, and the phase-specific Z-curve [5].

Sequence alignment is the simultaneous matching of two or more nucleotides. There are different algorithms available for this purpose. Broadly, those can be classified as algorithms based on hash tables, algorithms based on suffix trees and algorithms based on merge sorting [6]. Optimal alignment methods aim to maximize the score or minimize the cost of the process. But in case of dynamic programming approach, the time complexity of the algorithm is proportional to length of the sequence in exponential terms [7, 8]. In such approach, if the gap cost is nonlinear then complexity is extremely more [9].

Proposed Method

In this research paper, two DNA sequences have been aligned using a dynamic window algorithm. The major functions associated with this algorithm are LengthEqualiser (X, Y), dynamic window (p, q, X, Y), Modify-Y (i, j, X, Y), Modify-X (i, j, X, Y), and score (hit, c).

Algorithm of length equalizer function

```
Length_Equaliser (X,Y)
1.If |X| != |Y|
2.    do  NIG ← ||X| ~ |Y||
3.          if  |X| < |Y|
4.               append NIG to X
5.               else
6.               append NIG to Y
```

The function Length Equalizer takes the two sequences as arguments and tries to make them the same length by introducing the gaps at proper place. NIG represents the number of introductory gaps. X and Y are the two sequences,| | represents their cardinality.

Algorithm of Dynamic window function

```
1. P←1
2. q←1
3.       Dynamic_Window (p,q,X,Y)
4.         {seq← ø
5.           for  i← p to ( i- |X|)
6.           for  j← q to  (j- |Y|)
7.                 do  {seq ← ø + X(i)
8.                         i ←i+1
9.                         j←j+1
10.                    }
11.        hit ←i-p
12.           Return  seq to List , hit to Score( )
13.        }
14.            Modify-Y (i,j,X,Y)
15.            Modify-X (i,j,X,Y)
```

The dynamic window function tries to match the possible common sequences. The dynamic nature of the window is maintained with the help of sub-functions—Modify-Y (i, j, X, Y) and Modify-X (i, j, X, Y). Apart from this, the function sends the hits calculated to the score function to calculate the score of the process and keeps the different common sub-sequences obtained.

Algorithm of Modify-Y (i,j,X,Y) function

```
Modify-Y (i,j,X,Y)
1.{
2. for m ← j to m ≤ |Y|
3.       do {
4.                while ( X(i) != Y(m) )
5.                            c ← j +1
6.                            m ← m+1
7.            }
8.                      Insert gap (c-i) before X(I)
9.                       X(i) ← X(c)
10.                  P←i
11.                q←i
12.               X← X (i, |X|+c-i)
13.            Length_Equaliser (X,Y)
14.         Dynamic_Window (p,q,X,Y)
15.    Return  c to Score(hit, c )
16.  score (hit, c)
17. }
```

Algorithm of Modify-X (i,j,X,Y) function

```
Modify-X (i,j,X,Y)
1.{
2. for n ← i to m ≤ |X|
3.       do {
4.                while ( Y(j) != X(n) )
5.                            c ← i +1
6.                            n ← n+1
7.            }
8.                      Insert gap (c-i) before Y(j)
9.                       Y(j) ← Y(c)
10.                  P←j
11.                q←j
12.               Y← Y (i, |Y|+c-i)
13.            Length_Equaliser (X,Y)
14.         Dynamic_Window (p,q,X,Y)
15.    Return  c to Score(hit, c )
16.  Score (hit, c)
17. }
```

These two Modify functions are accountable for the dynamic nature of the window. First of all, these functions search the right position for the shifting of a base by calculating the number of gaps required which is represented by "c". Then after introducing the gaps in the either sequence X OR Y, they revoke the Length_Equaliser (X, Y) and Dynamic_Window (p, q, X, Y) which work recursively. They also return the c-value to Score function.

```
Algorithm of  Score (hit, c)

Score (hit, c)
   1. {
   2. Cost- hit ←hit x weight-of- base
   3. Penalty ← c  x  weight-of- gap
   4. Score % ←( Cost- hit ~ Penalty) / |X|
   5. }
```

The Score () function takes hit from Dynamic_Window (p, q, X, Y) and c from Modify-X (i, j, X, Y) or Modify-Y (i, j, X, Y) and calculates the score. Here, the cost of hit is calculated using the cost of each base in the sequence. Similarly, the penalty is calculated using the cost of a single gap and cumulative number of gaps introduced. Figure 7.1 represents the schematic view of the working of the algorithm.

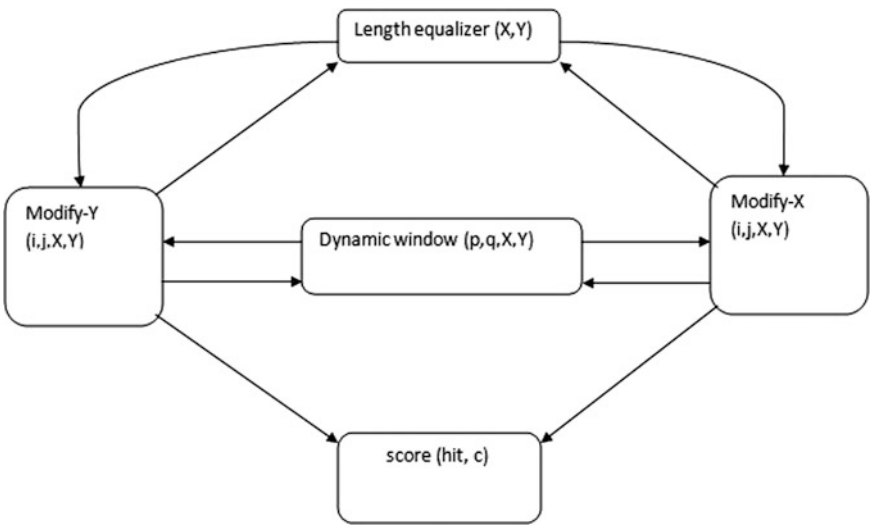

Fig. 7.1 Schematic view of the working of the dynamic window algorithm

Table 7.1 The coding sequence of the exon1 of beta-globin gene of eight different species

Species	Coding sequence
Human (92 bases)	ATGGTGCACCTGACTCCTGAGGAGAAGTCTGCCGTTACTGCCCTGT
	GGGGCAAGGTGAACGTGGATGAAGTTGGTGGTGAGGCCCTGGGCAG
Opossum (92 bases)	ATGGTGCACTTGACTTCTGAGGAGAAGAACTGCATCACTACCATCT
	GGTCTAAGGTGCAGGTTGACCAGACTGGTGGTGAGGCCCTTGGCAG
Gallus (92 bases)	ATGGTGCACTGGACTGCTGAGGAGAAGCAGCTCATCACCGGCCTCT
	GGGGCAAGGTCAATGTGGCCGAATGTGGGGCCGAAGCCCTGGCCAG
Lemur (92 bases)	ATGACTTTGCTGAGTGCTGAGGAGAATGCTCATGTCACCTCTCTGT
	GGGGCAAGGTGGATGTAGAGAAAGTTGGTGGCGAGGCCTTGGGCAG
Rat (92 bases)	ATGGTGCACCTAACTGATGCTGAGAAGGCTACTGTTAGTGGCCTGT
	GGGGAAAGGTGAACCCTGATAATGTTGGCGCTGAGGCCCTGGGCAG
Rabbit (90 bases)	ATGGTGCATCTGTCCAGTGAGGAGAAGTCTGCGGTCACTGCCCTGTG
	GGGCAAGGTGAATGTGGAAGAAGTTGGTGGTGAGGCCCTGGGC
Goat (86 bases)	ATGCTGACTGCTGAGGAGAAGGCTGCCGTCACCGGCTTCTGGGGCA
	AGGTGAAAGTGGATGAAGTTGGTGCTGAGGCCCTGGGCAG
Chimpanzee (105 bases)	ATGGTGCACCTGACTCCTGAGGAGAAGTCTGCCGTTACTGCCCTGTGGGGCAA
	GGTGAACGTGGATGAAGTTGGTGGTGAGGCCCTGGGCAGGTTGGTATCAAGG

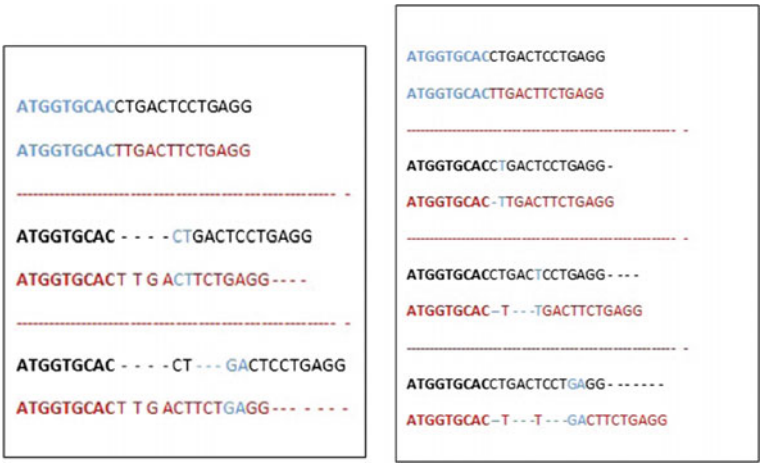

Fig. 7.2 Dynamic_Window algorithm functioning; left side represents the working of Modify-Y (i, j, X, Y) function; right side represents the working of Modify-X (i, j, X, Y) function

Result and Discussion

The performance of the dynamic window algorithm has been analyzed by different combinations of the following DNA sequences of β-globin genes of eight species given in Table 7.1 [10]. A part of the algorithm functioning is given in Fig. 7.2.

As a result of the working of the Dynamic_Window algorithm on a sub-sequences of Human and Opossum β-globin genes, it has been found that the hits are 11, gaps are 7 using Modify-Y (i, j, X, Y) function whereas the hits are 12, gaps are 7 using Modify-X (i, j, X, Y) function. By taking the cost of a base as +1 and that of a gap as −1, the score % in first case is 13.79 (left side in Fig. 7.2) whereas in second case, it is 17.29 (right side in Fig. 7.2). The common sequences in first case in the List are {ATGGTGCAC, CT, GA} and in the second case are {ATGGTGCAC, T, GA}.

Conclusion

In this study, the common sequence alignment using dynamic window algorithm has been proposed. The structure and the working of the algorithm have been discussed. The algorithm has been used for different sets of subsequences of β-globin genes and the score % has been obtained. The dynamic nature is clear from the comparative analysis of the score percentages.

References

1. Bioinformatics: sequence and genome analysis, David W. Mount, Cold Spring Harbor Laboratory Press.
2. Fundamental Molecular Biology, Lizabeth A. Allison, Blackwell Publishing Ltd.
3. A review of DNA sequencing techniques, Lilian T. C. Franca et al, Quarterly Reviews of Biophysics 35, 2 (2002), pp. 169–200. 2002 Cambridge University Press.
4. DB-Curve: a novel 2D method of DNA sequence visualization and representation, Chemical Physics Letters 367 (2003) 170–176.
5. Advanced Numerical Representation of DNA Sequences, 2012 International Conference on Bioscience, Biochemistry and Bioinformatics, IPCBEE vol. 3 1(2012) © (2012) IACSIT Press, Singapore.
6. A survey of sequence alignment algorithms for next-generation sequencing, Heng Li and Nils Homer, Briefings in bioinformatics.vol 11. no 5. 473–483, 2010.
7. Bioinformatics: sequence, structure & databanks, A practical approach, D. Higgins, W. Taylor, Oxford university press.
8. Sequence alignment and Markov's Model, K.R. Sharma, The Mc-Graw Hill.
9. Mind the Gaps: Evidence of Bias in Estimates of Multiple Sequence Alignments, Tanya Golubchik etal, Mol. Biol. Evol. 24(11):2433–2442. 2007.
10. Directed graphs of DNA sequences and their numerical characterization, Chun Li, Journal of Theoretical Biology 241 (2006) 173–177.