ISSN: 0974-2115
www.jchps.com Journal of Chemical and Pharmaceutical Sciences

# Decode_HMM_MLCS: An efficient algorithm to identify Multiple Longest Common Subsequence (MLCS) in Large DNA Sequences by using Decoding HMM

**[1]B. Devika Rubi*, [2]Dr. L. Arockiam**
[1]Research Scholar, Research and Development Centre, Bharathiar University, Coimbatore
[2]Associate Professor, Department of Computer Science, St Joseph's College, Tiruchirappalli
***Corresponding author: E-Mail: deviraja@gmail.com**

**ABSTRACT**

Most of the methods in computational sequence analysis are essentially statistical methods. Identifying MLCS (Multiple Longest Common Subsequence) is a NP-hard (Non-Deterministic Polynomial time) sequence analysis problem. This can be solved using the decoding methodology of the probabilistic model called HMM (Hidden Markov Model). This paper proposes a statistical based algorithm called Decode_HMM_MLCS. It applies the key task of Decoding HMM i.e to determine "the most probable path of 'characters occurrence' in a given sequence", by using Viterbi algorithm. The decoding process of HMM primarily identifies the similar regions between the sequences. MLCS occurs only in these similar regions. The proposed algorithm Decode_HMM_MLCS identifies required MLCS in linear time and space complexity.

**KEY WORDS:** Longest Common Subsequence, Hidden Markov Model (HMM), HMM Decoding, Viterbi Algorithm.

## 1. INTRODUCTION

The challenge in computational sequence analysis is to organize, classify and parse the immense richness of sequence data. Most of the methods in computational sequence analysis are essentially statistical methods. These methods make use of probabilistic theory. Identifying MLCS (Multiple Longest Common Subsequence) is a NP-hard (Non-Deterministic Polynomial time) sequence analysis problem. This can be solved using the decoding methodology of the probabilistic model called HMM (Hidden Markov Model).

This paper is organized as given below. Section 2 defines HMM and MLCS methods. Section 3 proposes a new algorithm called Decode_HMM_MLCS to identify MLCS. Section 4 discusses about the implementation and analyses the proposed algorithm DECODE_HMM_MLCS. Section 5 provides the conclusion.

## 2. HMM AND MLCS METHODS

**2.1. Hidden Markov Model (HMM):** Hidden Markov Chain generates sequences in which the probability of a symbol depends on previous symbol. It can be represented in graphically as a collection of "States", each of which corresponds to a particular residue with arrows between the states (Rabiner and Juang, 1986; Stolcke and Omohundro, 1993).

The Markov Chain model (Fujiwara, 1994) for a DNA sequence with start and end positions is shown in Fig.1.

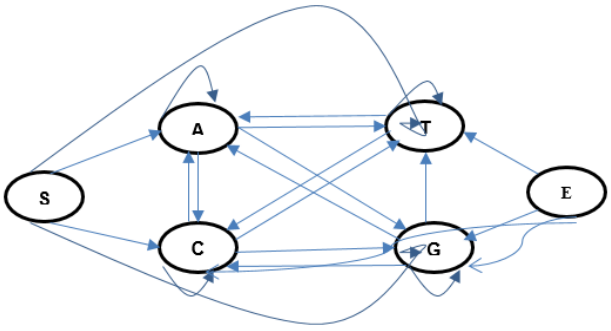


**Fig.1.Markov Chain Model for DNA sequence**

The key property of Markov chain is that the probability of each symbol $x_i$ depends only on the value of the preceding symbol $x_{i-1}$ and not on the entire previous sequence. This property helps to reduce the number of comparisons in sequence analysis.

The probability parameter of HMM are Transition Probability and Emission Probability (Durbin, 1998). The Transition Probability is defined as the probability of certain residue following another residue or one state following another state. This can be represented as

$a_{st} = p\ (x_i = t\ /\ x_{i-1}\ = s)$

In general, it is defined as $a_{ij} = x_{ij} / \sum\ x_{ik}$

The Emission Probability is defined as the probability that symbol "b" is seen when in state k.

i.e. $e_k\ (b) =\ p\ (x_i = b / \pi_i = k)$, where $\pi_i$ is the $i^{th}$ state in the path $\pi$.

The probability of emitting a symbol at a state is either 0 or 1. In general, the emission probability need not be 0 or 1. The sample DNA sequence is as shown in table 1.

**Table.1.Sample DNA sequence**

| Sample DNA Sequence for the GC-rich content<br>{ "A" ,"A", "G", "C", "G", "T", "G", "G", "G", "G", "C", "C", "C", "C", "G", "G" ,"C", "G", "A", "C", "A", "T", "G", "G", "G", "G", "T", "G", "T", "C" } |
|---|

The transition matrix for the sample sequence in table 2.

**Table.2.Transition Matrix for the sample Sequence**

| | AT – rich | GC-rich |
|---|---|---|
| AT - rich | 0.7 | 0.3 |
| GC - rich | 0.1 | 0.9 |

The emission matrix for the sample sequence in table 3.

**Table.3.Emission Matrix for the sample Sequence**

| | A | C | G | T |
|---|---|---|---|---|
| AT – rich | 0.39 | 0.10 | 0.10 | 0.41 |
| GC – rich | 0.10 | 0.41 | 0.39 | 0.10 |

Thus Hidden Markov Chain helps to model the DNA sequence and the transition and emission parameters help to evaluate the existence of a DNA sequence in the model.

**2.2. HMM Applications in recent biological research:** HSMM (Hilt, 2010) is a semi Hidden Markov model which incorporates the state duration information to identify genes. It also uses Baum-Welch algorithm for modelling the distribution on state intervals that are geometric. It also provides "Side" (extra) information for sequence segment homology maps.

Zhai (2010) identify motif using HMM. They use HMM implicitly modelling motif occurrence along genomic sequences. The estimation parameters are calculated using Poisson distribution for smaller number of occurrences of motif instances and normal approximation for larger number of occurrences of motif instances.

DNA base calling (Timp, 2012) is difficult in Nanopore sequencing (Timp, 2010), a promising third generation DNA sequencing. The difficulty is that resolution of signal/noise ratio is limited. The researchers use HMM Decoding by Viterbi algorithm to produce high accuracy in resolution, even with a poor signal/noise ratio.

Multi-Stream LSTM-HMM decoding (Wollmer, 2011) uses HMM decoding for noise robust keyword spotting in Automatic Speech Recognition (ASR) systems. ASR can be used in segment identification of DNA sequences.

**2.3. Multiple Longest Common Subsequence (MLCS):** The general description of DNA sequence is as follows: $A = \{a_1, a_2, \ldots a_{na}\}$ $B = \{b_1, b_2, \ldots b_{nb}\}$ $C = \{c_1, c_2, \ldots c_{nc}\}$where A, B, C represent the sequences and $a_i, b_i, c_i$ represent the basic units of the sequence, at positions i, whose elements are obtained from the set $V_q = \{ 0, 1 \ldots q - 1\}$. Typically,q = 4 and $V_4 = \{ a, c, g, t\}$ if A,B and C are DNA sequences.

A sequence $Z = \{z_1, z_2 \ldots z_n \}$ is called Multiple Longest Common Subsequence (MLCS) (Hirschberg, 1975), (Hirschberg, 1977), (Rick, 1994) and (Kumar & Rangan, 1987) of other sequences $A = \{ a_1, a_2 \ldots a_n \}$, $B = \{ b_1, b_2 \ldots b_n \} \ldots\ldots$ $K = \{k_1, k_2 \ldots k_n\}$ and A, B….K are the super sequences of Z denoted as $Z \subseteq \{A, B, \ldots K \}$, if there exists integers $1 \leq j_1 \leq j_2 \ldots \leq j_n \leq m$ such that $Z_1 \subseteq \{ a_{j1}, b_{j1}, c_{j1} \ldots k_{j1}\}$, $Z_2 \subseteq \{ a_{j2}, b_{j2}, c_{j2} \ldots k_{j2}\}$ $\ldots Z_n \subseteq \{ a_{jn}, b_{jn}, c_{jn} \ldots k_{jn}\} \leq m$

## 3. PROPOSED DECODE_HMM_MLCS ALGORITHM

The key task of HMM is to determine "the most probable path of characters occurrence in a given sequence". This can be obtained by experimenting/applying the given sequence into the Hidden Markov Model. This process is called as Decoding of HMM. This HMM is defined by the probability parameters called Transition and Emission matrices.

MLCS shares the longest common subsequence between two or more sequences. The key process to identify MLCS is identifying common sharing portion between the sequences. The decoding process of HMM primarily identifies the similar regions between the sequences. MLCS occurs only in these similar regions. Hence the required MLCS can be identified through the decoding process of HMM.

This algorithm contains three parts, namely (i) Transition and Emission matrices formation for the given set of sequences (ii) Apply Decoding of HMM to determine the most probable path for each sequence using Viterbi algorithm (Viterbi, 1967) (iii) Identification of MLCS using the results produced by part (ii). The pseudo code for proposed algorithm Decode_HMM_MLCS.

**Fig.2.Pseudeo code for Decode_HMM_MLCS ()**

```
//Decode_HMM_MLCS algorithm to find MLCS
Procedure Decode_HMM_MLCS()
{
// s1 are the sequences of length_n
{PS1_AA, PS1_AT, PS1_AG, … PS1_GG} = 0
// Are the 16 possible probabilities for the length_2 pattern in Seq_1

//Splitting the sequence s1 into consecutive sub_patterns of length_2
// To count the total number of occurrences for length_2 patterns
For i = 0 to n-1
{
X = s1.substring (i, i+1);
switch (x)
{
Case “AA” :
{ PS1_AA = PS1_AA + 1;
break;
}
Case “AT” :
{ PS1_AT = PS1_AT + 1;
break;
}
Case “AC” :
{ PS1_AC = PS1_AC + 1;
break;
}
Case “AG” :
{ PS1_AG = PS1_AG + 1;
break;
}
……
……
……
Case “GG” :
{ PS1_GG = PS1_AG + 1;
break;
}
} End Switch case
} End for i
//To calculate the total occurrence of each state
Row1_count = PS1_AA+ PS1_AT+PS1_AG+PS1_AC;
Row2_count = PS1_TA+ PS1_TT+PS1_TG+PS1_TC;
Row3_count = PS1_CA+ PS1_CT+PS1_CG+PS1_CC;
Row4_count = PS1_GA+ PS1_GT+PS1_GG+PS1_GC;
// To calculate Transition matrix for sequence s1
trans_matrix[0][0] = PS1_AA / Row1_count;
trans_matrix[0][1] = PS1_AT / Row1_count;
trans_matrix[0][2] = PS1_AC / Row1_count;
trans_matrix[0][3] = PS1_AG / Row1_count;
trans_matrix[1][0] = PS1_TA / Row2_count;
trans_matrix[1][1] = PS1_TT / Row2_count;
……
```

……
……
trans_matrix[3][2] = PS1_GG / Row4_count;
trans_matrix[3][3] = PS1_GC / Row4_count;
// To define emission matrix AT-rich, GC-rich content in the sequence
// Row-0 represents AT-rich, Row-1 represents GC rich for the
// columns nucleotides { A, C, G, T }
emission_matrix[0][0] = 0.39;
emission_matrix[0][1] = 0.1;
emission_matrix[0][2] = 0.1;
emission_matrix[0][3] = 0.41;
emission_matrix[1][0] = 0.1;
emission_matrix[1][1] = 0.41;
emission_matrix[1][2] = 0.39;
emission_matrix[1][3] = 0.1;
// Calling Viterbi algorithm
Viterbi_function(sequence, transitionmatrix, emissionmatrix);
} End Decode_HMM_MLCS

The pseudeo code of Decode_HMM_MLCS calculates the transition matrix for each sequence. The AT-rich row of emission matrix for the columns {A, C, G, T} is assigned with the values {0.39, 0.1, 0.1, 0.41} respectively, where the columns A and T have been assigned with higher probability_threshold. Similarly, in GC-rich row of emission matrix, columns G and C have been assigned with higher probability_threshold. Finally it calls Viterbi algorithm to generate the most probable path for characters occurrence in a given sequence.

## 4. ANALYSIS OF DECODE_HMM_MLCS

**4.1. Implementation details and Illustration of Decode_HMM_MLCS algorithm:** Decode_HMM_MLCS algorithm is implemented using "R" (ver 3.2.1) programming language on a Windows 10 machine with i7 Intel processor 2.33 GHZ, 16 GB RAM. The sample DNA sequences and the respective Decoding of sequences using Viterbi algorithm are given in Table 4. The transition and emission matrices for the sample sequences in Table 4 are given in Table 2 and Table 3.

**Table.4.Sample DNA Sequences with Decode_HMM_MLCS runtime results**

| |
|---|
| Sequence ID : 1 |
| Sequence :<br>"A", "T","C","G","G","G","G", "A", "T", "A", "T", "A", "T", "A", "G", "C","G", "C", "T", "C", "C", "C", "C", "G","A", "C", "A", "A", "A", "T", "C" |
| Decoding of Sequence # 1:<br>"Positions 1 - 2 → "AT-rich"<br>"Positions 3 - 8 → "GC-rich"<br>"Positions 9 - 14 → "AT-rich"<br>"Positions 15 - 26 → "GC-rich"<br>"Positions 27 - 29 → "AT-rich"<br>"Positions 30 - 30 → "GC-rich" |
| Sequence ID : 2 |
| Sequence :<br>"T", "G", "C", "T", "A" ,"T" ,"G" ,"G", "T", "C", "G", "A", "A", "T", "G","G", "G", "G", "C","T", "A", "A", "C", "C" ,"G" ,"A", "G", "G", "C", "G" |
| Decoding of Sequence # 2:<br>"Positions 1 – 1 → "AT-rich"<br>"Positions 2 - 4 → "GC-rich"<br>"Positions 5 - 6 → "AT-rich"<br>"Positions 7 - 12 → "GC-rich"<br>"Positions 13 - 14 → "AT-rich"<br>"Positions 15 - 20 → "GC-rich" |

| |
|---|
| "Positions 21 - 22 → "AT-rich"<br>"Positions 23 - 30 → "GC-rich" |

| |
|---|
| Sequence ID : 3 |
| Sequence :<br>"A", "C", "T", "G", "T" ,"T", "T", "T" ,"A", "G", "T", "C", "A", "G", "G", "G", "G", "C", "G", "C", "G", "T", "C", "C", "G", "G", "C", "A", "G", "C |
| Decoding of Sequence # 3:<br>"Positions 1 - 1 → "AT-rich"<br>"Positions 2 - 2 → "GC-rich"<br>"Positions 3 - 3 → "AT-rich"<br>"Positions 4 - 4 → "GC-rich"<br>"Positions 5 - 9 → "AT-rich"<br>"Positions 10 - 10 → "GC-rich"<br>"Positions 11 - 11 → "AT-rich"<br>"Positions 12 - 12 → "GC-rich"<br>"Positions 13 - 13 → "AT-rich"<br>"Positions 14 - 30 → "GC-rich" |

From the results of Decoding Sequences in Table 4 the DNA segments of length > 5 are sorted out and shown in Table 5.

**Table.5.Decode_HMM_MLCS runtime results (DNA Segment length > 5)**

| Sequence | DNA Segment | AT-rich | GC-rich | DNA Segment Length |
|---|---|---|---|---|
| 1 | 3,8 | | y | 6 |
| | 9,14 | Y | | 6 |
| | 15,26 | | y | 12 |
| 2 | 7,12 | | y | 6 |
| | 15,20 | | y | 6 |
| | 23,30 | | y | 8 |
| 3 | 14,30 | | y | 16 |

It can be inferred from the Table 5 that MLCS occurs only at the DNA segment regions { (15 – 26), (23 – 30), (14 – 30) } of sequenceID 1,2 and 3 respectively. Grpahical representation of this result is shown in Fig 3.

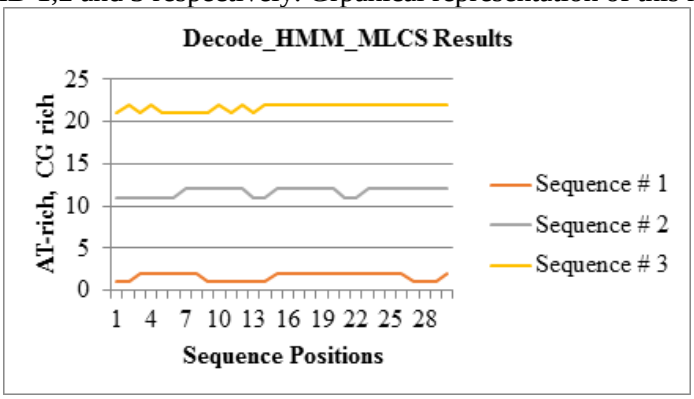


**Fig.3.Decode_HMM_MLCS results**

The Decode_HMM_MLCS runtime graphical results show that the DNA segment region 15 to 30 of the given three sequences contain GC-rich content. Hence, MLCS occurs only within this region. Thus by comparing only the similar regions, Decode_HMM_MLCS reduces the search space from the whole region to similar regions only. This leads to less number of comparisons and improves the time and space complexity.

**4.2. Time and Space complexity:** If "L" is the length of the sequence, then the time complexity for Decode_HMM_MLCS is defined as

$$T(n) = Q^2 + Q^2 + |Q^2| L,$$

Where Q is the number of states and $Q^2$ is the number of calculations needed to form the transition matrix, which is same as emission matrix. Decoding of sequence requires "$Q^2$ L" runtime. Hence, the time complexity for Decode_HMM_MLCS is O ($|Q^2|$ L). And the space complexity is O (| Q | L).

## 5. CONCLUSION

This paper proposes a statistical based algorithm called Decode_HMM_MLCS. HMM generates the current state symbol "$x_i$" with respect to the previous state symbol "$x_{i-1}$", in the forward direction. MLCS shares the longest common subsequence between two or more sequences. The key process to identify MLCS is identifying common sharing portion between the sequences. The decoding process of HMM primarily identifies the similar regions between the sequences. MLCS occurs only in these similar regions. Hence the required MLCS can be identified through the decoding process of HMM.

Decoding process of HMM requires linear time and space complexity. Hence the proposed algorithm Decode_HMM_MLCS require linear time and space complexity.

## REFERENCES

Durbin R, Eddy S.R, Krogh A, Mitchison G, Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids, s.l.:Cambridge University, 1998.

Fujiwara Y, Asogawa M, Konagaya A, Stoochastic motif extraction using hidden markov model, s.l., Proceedings of the Second International Conference on Intelligent Systems of Molecular Biology, 1994.

Hilt W.S, Jiang Z, Baribault C, Hidden Markov Model with Duration Side Information for Novel HMMD Derivation, with Application to Eukaryotic Gene Finding, EURASIP Journal on Advances in Singal Processing, 2010, 1 - 11.

Hirschberg D, A Linear Space Algorithm for Computing Maximal Common Subsequences, Comm. ACM, 18(6), 1975, 341 - 343.

Hirschberg D, Algorithms for the Longest Common Subsequence Problem, ACM, 24, 1977, 664 - 675.

Kumar S, Rangan C, A Linear Space Algorithm for the LCS Problem, Acta Informatica, 24, 1987, 353 - 362.

Rabiner L, Juang B, An introduction to Hidden Markov Models, IEEE ASSP Magazine, 1986, 4 - 16.

Rick, New Algorithms for the Longest Common Subsequence Problem, Bonn: Computer Science, University of Bonn, 1994.

Stolcke A, Omohundro S.M, Hidden Markov Model induction by Bayesian model merging, Advances in Neutral Information Processing Systems, 1993, 11 - 18.

Timp W, Comer J, Aksimentiev A, DNA Base-Calling from a Nanopore Using a Viterbi Algorithm, Biophysical Journal, 102, 2012, 37 - 39.

Timp W, Nanopore Sequencing: Electrical Measurements of the Code of Life, IEEE Transactions on Nanotechnology, 9(3), 2010, 281-294.

Viterbi A, Error bounds for convolutional codes and an asymptotically optimum decoding algorithm, IEEE Transaction on Information Theory, 13(2), 1967, 260 -269.

Wollmer M, Marchi E, Squaratini S, Schuller B, Mutli-stream LSTM-HMM decoding and histogram equalization for noise robust keyword spotting, Springer Science+Business Media , 5, 2011, 253 - 264.

Zhai Z, The Power of Detecting Enriched Patterns: An HMM Approach, Journal of Computational Biology, 17(4), 2010, 581-592.