International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 - 5161
©2014 INPRESSCO®, All Rights Reserved
Available at http://inpressco.com/category/ijcet

Research Article

PDA to Identify Palindrome Subsequence Problem in DNA Sequence

Anupama B S[Ȧ*] and Prasanna K B[Ḃ]

[Ȧ]Information Science and Engineering, Channabasaveshwara Institute of Technology, Tumkur, India
[Ḃ]Computer Science and Engineering, Channabasaveshwara Institute of Technology, Tumkur, India

Accepted 20 Nov 2014, Available online 30 Dec 2014, **Vol.4, No.6 (Dec 2014)**

Abstract

DNA molecules contains the strings consisting of four symbols namely 1) A-adenine,2)C-cytosine , 3) G - guanine and 4) T –thymine. Since the bio-molecular structures can be defined in terms of sequence of symbols (i.e., strings) there exists a correlation between formal model and bi-molecular structure. DNA palindromes appear frequently and are widespread in human cancers. Identifying them could help advance the understanding of genomic instability (Choi Charles Q, 2005; Tanaka, Hisashi; et al, 2003). The Palindrome subsequences detection problem is therefore an important issue in computational biology. In this paper we presented a Push down Automata Model (PDA) to identify all palindrome subsequence that is present in the DNA sequence.

Keywords: *The PushDown Automata, subsequence algorithm, DNA base sequence.*

1. Introduction

The very basic unit of the human genome is a single DNA nucleotide. This nucleotide is extremely small and is made up of minuscule atoms, which creates a challenge for even an advanced microscope to be used for detection. Researchers still, however, need to be able to determine the sequence of bases in DNA that make up the human genome. As such, DNA sequencing has been developed but the process itself is seemingly complex one. DNA sequencing involves the determination of the order of DNA bases. You may be wondering what makes these bases so important. In a strand of DNA, there are some simple units known as nucleotides. These nucleotides have a 'backbone' that consists of sugars and a phosphate group. The DNA bases can be one of four kinds and they are attached to these sugars. These bases hold the important and unique genetic information for your body. These bases are: Adenine (A). Thymine (T), Cytosine (C) and
Guanine (G)

They explained how the DNA chains travelled in opposite directions, with the bases on the inside of the helix and the phosphates on the outside. They emphasised the importance of how these two chains are held together by the purine and pyrimidine bases. A purine pairs with a pyrimidine base, which means that one base from a chain is bonded to a single base from the other chain. Ultimately, this means that the two bases lie side-by-side. If you recall the four bases mentioned earlier, you can perhaps understand now how the bases are very specific in how they pair with another base. An adenine base only pairs with thymine and a guanine base only pairs with cytosine. This understanding of the structure of DNA is particularly important because it led to the realization that if there is an adenine on one side of the pair, then the other base must be thymine. Similarly, if we know there is guanine on one side of the chain, it is paired with cytosine. What this essentially means is that if there is a set sequence of bases on one side of the chain, the other side is automatically determined.

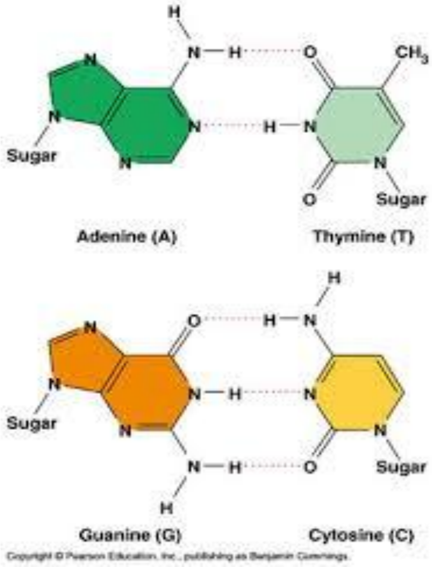

Figure 1: DNA base pairs

2. Pushdown Automata (PDA)

Pushdown automata are used in theories about what can be computed by machines. They are more capable than finite-

*Corresponding author: **Anupama B S**

state machines but less capable than Turing machines .Pushdown automata differ from finite state machines in two ways:

1) They can use the top of the stack to decide which transition to take.
2) They can manipulate the stack as part of performing a transition.

Pushdown automata choose a transition by indexing a table by input signal, current state, and the symbol at the top of the stack. This means that those three parameters completely determine the transition path that is chosen. Finite state machines just look at the input signal and the current state: they have no stack to work with. Pushdown automata add the stack as a parameter for choice.

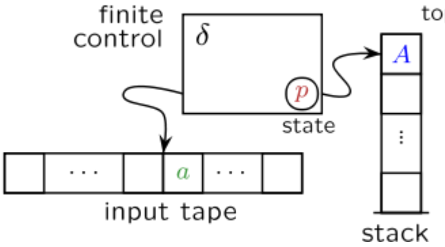

Figure 2: Block Diagram of PDA

PDA is formally defined as a 7-tuple:

$$M = (Q,\ \Sigma,\ \Gamma,\ \delta,\ q_0,\ Z,\ F)$$

where

- **Q** is a finite set of *states*
- Σ is a finite set which is called the *input alphabet*
- **Γ** is a finite set which is called the *stack alphabet*
- δ is a finite subset of **Q×(∑Ụє)× Γ to Q× Γ***, the *transition relation*.
- $q_0 \in Q$ is the *start state*
- $Z \in \Gamma$ is the *initial stack symbol*
- $F \subseteq Q$ is the set of *accepting states*

3. Proposed model

A string is a sequence of symbols from an alphabet set ∑. A string is a sequence of symbols from an alphabet set . For a string S = s1s2…sn of length n, let si denote the ith symbol in S. A subsequence of S is obtained by deleting zero or more (not necessarily consecutive) symbols form S. A palindrome is a string of the form ww^R where w is a non-empty substring and w^R is the reverse of w. For example, TT and GCAACG are palindromes. There are many various classic computing problems in finding palindromes of a string. For example, Manacher discovered an on-line sequential algorithm that finds all initial palindromes in a string (Manacher D *et al*, 1975). Porto and Barbosa gave an algorithm to find long approximate palindromes (Proto A. H *et al*, 2002). In computational molecular biology, finding out the palindrome subsequences in DNA sequence is an important issue (Gusfied D. *et al*, 1997). However, as far as we know, there is no article discussing about how to verify the all palindrome subsequences in computation model.

3.1 K-Palindrome Subsequence Algorithm

In this algorithm, we find all palindrome subsequences form one palindrome subsequence to the longest palindrome subsequence. Given a string S of length n, let U_k be the set of K- palindrome where 1≤ K≤ n/2.

Step 1: We use incidence matrix to find all matched pair (i,j) where 1≤ i<j≤n. and add them into U_1 ,because each matched pair is 1-palindrome subsequence .

Step 2 : We generate Uk from Uk-1 and U1 where 1≤ K≤ n/2. . For all k-1-palindrome subsequences in Uk-1, we take a k -1-palindrome subsequence (i1, j1) … (ik-1, jk-1) form Uk-1 and we check all 1-palindromes from U1 whether there is a 1-palindrome (i', j') which satisfies the rule i' > ik-1 and j' < jk-1. If it is satisfied, we combine the k-1-palindrome (i1, j1) … (ik-1, jk-1) with the 1-palindrome (i', j') to be k-palindrome (i1, j1) … (ik-1, jk-1) (i', j') and add it into the set Uk. Until the Un/2 is generated, we can get the set U = U1 U2 … Un/2 which contains all palindrome subsequences of S.

Ex : Given a string S = ACGATGTAC, We now illustrate the whole procedure in detail,

S1	S2	S3	S4	S5	S6	S7	S8	S9
A	C	G	A	T	G	T	A	C

Step 1: We use incidence matrix to find all matched pairs (i, j) where 1≤ i<j≤n. Table 1 The incidence matrix for this string S = ACGATGTAC.

Table 1: Incidence Matrix for the String S

9	C									
8	A									0
7	T								0	0
6	G							0	0	0
5	T						0	1	0	0
4	A					0	0	0	1	0
3	G				0	0	1	0	0	0
2	C			0	0	0	0	0	0	1
1	A		0	0	1	0	0	0	1	0
Si	/	A	C	G	A	T	G	T	A	C
Sj		1	2	3	4	5	6	7	8	9

After generating above the incident matrix, we can get the U1 = {(1, 4), (1, 8), (2, 9), (3, 6), (4, 8), (5, 7)} and corresponding string s1={ AA,CC,GG,,TT } , U2 = { (1, 8) (3, 6), (1, 8) (5, 7), (2, 9) (3, 6), (2, 9) (4, 8), (2, 9) (5, 7), (4, 8) (5, 7)}, and corresponding strings are s2={ AGGA,ACCA, CGGC, CAAC,CTTC, ATTA } and similarly for U3={ (2, 9) (4, 8) (5, 7) and the set s3={ CATTAC }.

Now consider a set S=S1U S2U S3 i , e S ={ AA,CC , GG,,TT, AGGA,ACCA, CGGC, CAAC,CTTC, ATTA, CATTAC }. Above gene sequence present in the S resembles the context-free language L= { WW^R / W∈{a, b} * } .

By using the Pushdown automata to verify that the language of the form WW^R is a palindrome. The formal machine for this language as shown below.

$\delta(q_o, a, Z_o)= \{ (q_o , aZ_o) \}$
$\delta(q_o, b, Z_o)= \{ (q_o , bZ_o) \}$
$\delta(q_o, a, a)= \{ (q_o , aa) , (q_1 ,\epsilon) \}$
$\delta(q_o, a, b)= \{ (q_o , ab) \}$
$\delta(q_o, b, a)= \{ (q_o , ba) \}$
$\delta(q_o, b, b)= \{ (q_o , bb) , (q_1 ,\epsilon) \}$
$\delta(q_1, a, a)= \{ (q_1 ,\epsilon) \}$
$\delta(q_1, b, b)= \{ (q_1 ,\epsilon) \}$
$\delta(q_o, \epsilon , Z_o)=\{ (q_1 ,\epsilon) \}$
$\delta(q_1, \epsilon , Z_o)=\{ (q_2 , Z_o) \}$

We can prove this by following theorem.

Theorem: The PDA for L= WW^R accepts a string x by final state if and only if x is of the form WW^R .

Proof: *(if-part)* : If the string is of the form wwR then there exists a sequence of IDs that leads to a final state:
$(q_o, ww^R , Z0)$ |---* $(q_o ,w^R ,wZ0)$ |---* $(q_1,w^R,wZ0)$ |---* $(q_1, \epsilon ,Z0)$ |---* $(\mathbf{q}_2, \epsilon ,Z0)$
(only-if part); Proof by induction on |x|

Conclusion

Today's large-scale sequencing projects would be impossible without automatic sequencing machines**.** So the DNA molecules become separated into different bands according to their size. Here we find palindrome subsequence and verified that sequence by automatic machine. It is an effective tool for further research.

References

Choi, Charles Q (2005) DNA palindromes found in cancer. The Scientist

Tanaka, Hisashi; Bergstrom, Donald A; Yao, Meng-Chao and Tapscott, Stephen J (2006) Large DNA palindromes as a common form of structural chromosome aberrations in human cancers.

Human CellKurose and Ross (2004), Computer Networking: A top-down approach featuring the Internet, 3rd ed., Addison-Wesley.

Manacher, D. (1975) A new Linear-Time “On-Line” Algorithm for Finding the Smallest Initial Palindrome of a String. J. Assoc. Comput.

Proto, A. H. L. and Barbosa V. C. (2002) Finding Approximate Palindromes in Strings. Pattern Recognition

Gusfied, D. (1997) Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology, Cambridge University Press, New York.