The Longest Common Subsequence Problem

Anna Gorbenko

Department of Intelligent Systems and Robotics
Ural Federal University
620083 Ekaterinburg, Russia
gorbenko.ann@gmail.com

Vladimir Popov

Department of Intelligent Systems and Robotics
Ural Federal University
620083 Ekaterinburg, Russia
Vladimir.Popov@usu.ru

Abstract

Algorithms on sequences of symbols have been studied for a long time and now form a fundamental part of computer science. One of the very important problems in analysis of sequences is the longest common subsequence problem. For the general case of an arbitrary number of input sequences, the problem is \textit{NP}-hard. We describe an approach to solve this problem. This approach is based on constructing a logical model for the longest common subsequence problem.

Keywords: the longest common subsequence problem, logical model, satisfiability problem, \textit{NP}-complete

1 Introduction

Algorithms on sequences of symbols have been studied for a long time and now form a fundamental part of computer science. One of the very important problems in analysis of sequences is the longest common subsequence (LCS) problem. The LCS problem is to find the longest subsequence common to all sequences in a set of sequences (often just two).

The computational problem of finding the longest common subsequence of a set of \( k \) strings has been studied extensively over the last thirty years (see [1, 2, 3] and references). This problem has many applications. When \( k = 2 \), the longest common subsequence is a measure of the similarity of two
strings and is thus useful in molecular biology, pattern recognition, and text compression [4, 5, 6]. The version of LCS in which the number of strings is unrestricted is also useful in text compression [5], and is a special case of the multiple sequence alignment and consensus subsequence discovery problem in molecular biology [7, 8, 9].

The k-unrestricted LCS problem is NP-complete [5]. If the number of sequences is fixed at k with maximum length n, their longest common subsequence can be found in $O(n^{k-1})$ time, through an extension of the pairwise algorithm [3]. Suppose $|S_1| = n$ and $|S_2| = m$, the longest common subsequence of $S_1$ and $S_2$ can be found in time $O(nm)$ [10, 11, 12].

2 Problem Definition

Given two sequences $S$ and $T$ over some fixed alphabet $\Sigma$, the sequence $T$ is a subsequence of $S$ if $T$ can be obtained from $S$ by deleting some letters from $S$. Notice that the order of the remaining letters of $S$ bases must be preserved. The length of a sequence $S$ is the number of letters in it and is denoted as $|S|$. For simplicity, we use $S[i]$ to denote the $i$th letter in sequence $S$, and $S[i, j]$ to denote the substring of $S$ consisting of the $i$th letter through the $j$th letter.

Given sequences $S_i, 1 \leq i \leq m$, over some fixed alphabet $\Sigma$, the LCS problem asks for a longest sequence $T$ that is a subsequence of $S_i$ for all $i \in \{1, 2, \ldots, m\}$.

In the decision version LCS can be formulated as following.

LCS:

\textbf{Instance:} A fixed alphabet $\Sigma$, sequences $S_i, 1 \leq i \leq m$, and positive integer $k$.

\textbf{Question:} Is there a sequence $T$, $|T| \geq k$, that is a subsequence of $S_i$ for all $i \in \{1, 2, \ldots, m\}$?

3 Logical model

The propositional satisfiability problem (PSAT) is a core problem in mathematical logic and computing theory. Propositional satisfiability is the problem of determining if the variables of a given boolean function can be assigned in such a way as to make the formula evaluate to true. PSAT was the first known NP-complete problem, as proved by Stephen Cook in 1971.

Encoding problems as Boolean satisfiability and solving them with very efficient satisfiability algorithms has recently caused considerable interest. From a practical point of view, to further investigations of the LCS of interest to find explicit reductions to the satisfiability problem. Such reduction will provide a basis for construction of an effective intelligent algorithm for LCS. In
this paper we consider reductions from LCS to PSAT.

Let

\[ \varphi[i, j] = (\bigvee_{1 \leq l \leq |\Sigma|} x[i, j, l]) \land (\bigwedge_{1 \leq l[1] \leq |\Sigma|, 1 \leq l[2] \leq |\Sigma|, l[1] \neq l[2]} (\neg x[i, j, l[1]] \lor \neg x[i, j, l[2]])), \]

\[ \varphi = \bigwedge_{1 \leq i \leq m, 1 \leq j \leq |S_i|} \varphi[i, j], \]

\[ \psi[1, i, j] = \bigwedge_{1 \leq l[1] \leq k, 1 \leq l[2] \leq k, l[1] \neq l[2]} (\neg y[i, j, l[1]] \lor \neg y[i, j, l[2]]), \]

\[ \psi[1] = \bigwedge_{1 \leq i \leq m, 1 \leq j \leq |S_i|} \psi[1, i, j], \]

\[ \psi[2, i, j, l] = \bigwedge_{1 \leq j[1] \leq |S_i|, j[1] \neq j} (\neg y[i, j, l] \lor \neg y[i, j[1], l]), \]

\[ \psi[2] = \bigwedge_{1 \leq i \leq m, 1 \leq j \leq |S_i|, 1 \leq l \leq k} \psi[2, i, j, l], \]

\[ \psi[3, i, j, l] = \bigwedge_{1 \leq j[1] \leq |S_i|, 1 \leq l[1] \leq k, j[1] < j} (\neg y[i, j, l] \lor \neg y[i, j[1], l[1]]), \]

\[ \psi[3] = \bigwedge_{1 \leq i \leq m, 1 \leq j \leq |S_i|, 1 \leq l \leq k} \psi[3, i, j, l], \]

\[ \psi[4, i, l] = \bigvee_{1 \leq j \leq |S_i|} y[i, j, l], \]

\[ \psi[4] = \bigwedge_{1 \leq i \leq m, 1 \leq l \leq k} \psi[4, i, l], \]
\[ \xi[i[1], i[2], j[1], j[2], p] = \bigwedge_{1 \leq l \leq |\Sigma|} (y[i[1], j[1], p] \land y[i[2], j[2], p]) \rightarrow \\
 x[i[1], j[1], l] = x[i[2], j[2], l], \]

\[ \xi = \bigwedge_{1 \leq i[1] \leq m} \bigwedge_{1 \leq i[2] \leq m} \bigwedge_{i[1] \neq i[2]} \bigwedge_{1 \leq j[1] \leq |S_{i[1]}|} \bigwedge_{1 \leq j[2] \leq |S_{i[2]}|} \bigwedge_{1 \leq p \leq k} \]

\[ \tau = \varphi \land (\bigwedge_{1 \leq q \leq 4} \psi[q]) \land \xi. \]

**Theorem.** Given sequences \( S_i, 1 \leq i \leq m, \) over some fixed alphabet \( \Sigma, \) and positive integer \( k. \) There is a sequence \( T, |T| \geq k, \) that is a subsequence of \( S_i \) for all \( i \in \{1, 2, \ldots, m\}, \) if and only if \( \tau \) is satisfiable.

**Proof.** Suppose that there is a sequence \( T, |T| \geq k, \) that is a subsequence of \( S_i \) for all \( i \in \{1, 2, \ldots, m\}. \) Clearly, we can assume that \( |T| = k. \) In this case,

\[ T = S_1[j[1, 1]]S_1[j[1, 2]] \ldots S_1[j[1, k]] = \]
\[ S_2[j[2, 1]]S_2[j[2, 2]] \ldots S_2[j[2, k]] = \]
\[ \cdots \]
\[ S_m[j[m, 1]]S_m[j[m, 2]] \ldots S_m[j[m, k]] \]

for some

\[ 1 \leq j[1, 1] < j[1, 2] < \ldots < j[1, k] \leq |S_1|, \]
\[ 1 \leq j[2, 1] < j[2, 2] < \ldots < j[2, k] \leq |S_2|, \]
\[ \cdots \]
\[ 1 \leq j[m, 1] < j[m, 2] < \ldots < j[m, k] \leq |S_m|. \]

Let \( \Sigma = \{a_1, a_2, \ldots, a_p\}; \) \( x[i, j, l] = 1 \) where \( 1 \leq i \leq m, 1 \leq j \leq |S_i|, 1 \leq l \leq |\Sigma|, S_i[j] = a_t \); \( x[i, j, l] = 0 \) where \( 1 \leq i \leq m, 1 \leq j \leq |S_i|, 1 \leq l \leq |\Sigma|, S_i[j] \neq a_t \); \( y[i, j, l] = 0 \) where \( 1 \leq i \leq m, 1 \leq j \leq |S_i|, 1 \leq l \leq k, j \notin \{j[i, 1], j[i, 2], \ldots, j[i, k]\}; \) \( y[i, j, l] = 0 \) where \( 1 \leq i \leq m, 1 \leq j \leq |S_i|, 1 \leq l \leq k, j = j[i, r], l \neq r; \) \( y[i, j, l] = 1 \) where \( 1 \leq i \leq m, 1 \leq j \leq |S_i|, 1 \leq l \leq k, j = j[i, r], l = r. \)

Suppose that \( S_i[j] = a_{i_0}. \) In this case, \( x[i, j, l_0] = 1. \) Therefore,

\[ \bigvee_{1 \leq l \leq |\Sigma|} x[i, j, l] = 1. \quad (1) \]
Suppose that \( j \psi = 1 \). Since \( l \leq 1 \leq |\Sigma| \), \( 1 \leq l \leq |\Sigma| \), \( l \neq 2 \). In view of (1), hence we get \( \varphi[i, j] = 1 \). Therefore, by arbitrariness of \( i \) and \( j \) we obtain that \( \varphi = 1 \).

If \( j \notin \{j[i, 1], j[i, 2], \ldots, j[i, k]\} \), then \( y[i, j, l] = 0 \) for all \( l \). Hence \( -y[i, j, l] = 1 \). So, \( \psi[1, i, j] = 1 \). If \( j = j[i, r] \) and \( l \neq r \), then \( y[i, j, l] = 0 \). In view of \( l \neq 2 \), we obtain \( -y[i, j, l] \neq 1 \). Therefore, \( \psi[1] = 1 \).

Consider \( \psi[2, i, j, l] \). If \( y[i, j, l] = 0 \) or \( y[i, j, l] = 1 \), then \( \psi[2, i, j, l] = 1 \).

Suppose that \( y[i, j, l] = 1 \) and \( y[i, j, l] = 1 \). By definition, if \( y[i, j, l] = 1 \), then \( j = j[i, r] \), \( l = r \). Similarly, \( j[1] = j[i, r] \), \( l = r \). Hence \( j = j[1] \). Therefore, \( \psi[2] = 1 \).

Consider \( \psi[3, i, j, l] \). If \( y[i, j, l] = 0 \) or \( y[i, j, l] = 1 \), then \( \psi[3, i, j, l] = 1 \).

Since \( \varphi = \psi[1] = \psi[2] = \psi[3] = \psi[4] = \xi = 1 \), it is clear that \( \tau = 1 \).

Now suppose that \( \tau = 1 \). By definition, \( \varphi = \psi[1] = \psi[2] = \psi[3] = \psi[4] = \xi = 1 \). Since \( \varphi = 1 \), it is easy to see that \( \varphi[i, j] = 1 \) for all \( 1 \leq i \leq m \), \( 1 \leq j \leq |S_i| \). Hence \( -y[i, j, l] \neq 1 \) for all \( 1 \leq l \leq |\Sigma| \), \( 1 \leq l \leq |\Sigma| \) such that \( l \neq 2 \). Therefore, for given \( i \) and \( j \) there is no more than one value of \( l \) such that \( x[i, j, l] = 1 \). In view of \( \varphi[i, j] = 1 \), we have (1). Hence there exists at least one value of \( l \) such that \( x[i, j, l] = 1 \).

Therefore, we can assume that \( S_i[j] = a_i \) where \( 1 \leq i \leq m \), \( 1 \leq j \leq |S_i| \), \( 1 \leq l \leq |\Sigma| \). Hence \( -y[i, j, l] \neq 1 \) for all \( 1 \leq i \leq m \), \( 1 \leq j \leq |S_i| \). Hence \( -y[i, j, l] \neq 1 \) for all \( 1 \leq l \leq |\Sigma| \), \( 1 \leq l \leq |\Sigma| \) such that \( l \neq 2 \). Therefore, for given \( i \) and \( j \) there is no more than one value of \( l \) such that \( y[i, j, l] = 1 \). Since \( \psi[2] = 1 \), it is easy to see that \( \psi[2, i, j, l] = 1 \) for all \( 1 \leq i \leq m \), \( 1 \leq j \leq |S_i| \), \( 1 \leq l \leq k \).
Hence \( -y[i,j,l] \lor -y[i,j[1],l] = 1 \) for all \( 1 \leq j[1] \leq |S_i| \) such that \( j[1] \neq j \).

Therefore, for given \( i, j, \) and \( l \) either \( y[i,j,l] = 0 \) or

\[
y[i,j,l] = 1 \land ( \bigwedge_{j[1] \neq j} y[i,j[1],l] = 0).
\]

(2)

Since \( \psi[3] = 1 \), it is easy to see that \( \psi[3,i,j,l] = 1 \) for all \( 1 \leq i \leq m, 1 \leq j \leq |S_i|, 1 \leq l \leq k \).

Hence \( -y[i,j,l] \lor -y[i,j[1],l[1]] = 1 \) for all \( 1 \leq j[1] \leq |S_i|, 1 \leq l[1] \leq k \) such that \( j[1] < j \) and \( l[1] \geq l \). Therefore, for given \( i, j, \) and \( l \) either \( y[i,j,l] = 0 \) or

\[
y[i,j,l] = 1 \land ( \bigwedge_{j[1] < j, l[1] \geq l} y[i,j[1],l[1]] = 0).
\]

(3)

Since \( \psi[4] = 1 \), it is easy to see that \( \psi[4,i,l] = 1 \) for all \( 1 \leq i \leq m, 1 \leq l \leq k \).

Hence for given \( i \) and \( l \) there exists at least one value of \( j \) such that \( y[i,j,l] = 1 \).

Since for given \( i \) and \( j \) there is no more than one value of \( l \) such that \( y[i,j,l] = 1 \), we can suppose that

\[
y[i,j[i,1],l[i,1]] = 1, y[i,j[i,2],l[i,2]] = 1, \ldots, y[i,j[i,p[i]],l[i,p[i]]] = 1
\]

and \( y[i,j,l] = 0 \) for given \( i \) where \( 1 \leq p[i] \leq |S_i|, (i,l) \notin \{(i,1],l[i,1]), (j[i,2],l[i,2]), \ldots, (j[i,p[i]],l[i,p[i]])\} \).

Since for given \( i \) and \( l \) there exists at least one value of \( j \) such that \( y[i,j,l] = 1 \), we can suppose that \( p[i] \geq k \). In view of (2), it is clear that \( p[i] \leq k \). So, \( p[i] = k \). Without loss of generality, we can suppose that

\( j[i,1] < j[i,2] < \ldots < j[i,k] \).

In view of (3), it is easy to see that if \( j[i,t_1] < j[i,t_2] \), then \( l[i,t_1] < l[i,t_2] \).

Therefore, (4) can be represented as follows:

\[
y[i,j[i,1],1] = 1, y[i,j[i,2],2] = 1, \ldots, y[i,j[i,k],k] = 1.
\]

Let

\[
T = S_1[j[1,1]]S_1[j[1,2]]\ldots S_1[j[1,k]].
\]

Since \( \xi = 1 \), it is easy to see that

\[
S_1[j[1,1]]S_1[j[1,2]]\ldots S_1[j[1,k]] =
S_2[j[2,1]]S_2[j[2,2]]\ldots S_2[j[2,k]] = \ldots
S_m[j[m,1]]S_m[j[m,2]]\ldots S_m[j[m,k]].
\]

So, in view of theorem, we obtain an explicit reduction from LCS to PSAT.
4 Conclusion

In papers [13, 14, 15, 16, 17, 18, 19] the authors considered some algorithms to solve logical models. Our computational experiments have shown that these algorithms can be used to solve the logical model for LCS.

References


Received: May, 2012