On 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

In this paper we consider an approach to solve the longest common subsequence problem. This approach is based on constructing logical models for the problem.

Keywords: the longest common subsequence problem, logical model, satisfiability problem, NP-complete

1 Introduction

Algorithms on sequences of symbols are used in many very different applications, groups (see e.g. [1]), rings (see e.g. [2, 3]), semigroups (see e.g. [4, 5]), bioinformatics (see e.g. [6]), robotics (see e.g. [7, 8]), compression (see e.g. [9, 10]), etc. These algorithms now form a fundamental part of computer science. Algorithms on sequences of symbols received a lot of attention recently (see e.g. [11, 12, 13, 14]). In particular, computationally hard problems is extensively studied (see e.g. [15, 16, 17, 18]). One of the most important problems in analysis of sequences is the longest common subsequence (LCS) problem (see e.g. [18]). An approach to solve this problem is described in [18]. This approach is based on constructing a logical model for the longest common subsequence problem.

In papers [18, 19, 20, 21, 22, 23] the authors considered some algorithms to solve different logical models (see also [24, 25, 26, 27, 28, 29, 30]). In particular,
we have obtained an explicit reduction from LCS to PSAT in [18]. In this paper we consider explicit reductions from LCS to SAT and 3SAT.

In our reductions we use Boolean functions $\xi$ and $\tau$ from [18]. Let

$$\xi' = \land_{1 \leq i[1] \leq m, 1 \leq i[2] \leq m} (\land_{1 \leq l \leq |\Sigma|} ((\neg y[i[1], j[1], l] \lor \neg y[i[2], j[2], l]) \land \neg x[i[1], j[1], l] \lor x[i[2], j[2], l]) \land ($$ $$\neg y[i[1], j[1], l] \lor \neg y[i[2], j[2], l]) \lor x[i[1], j[1], l] \lor \neg x[i[2], j[2], l]).$$

**Theorem 1.** $\xi \Leftrightarrow \xi'$. 
**Proof.** Note that $\alpha \Rightarrow \beta \Leftrightarrow \neg\alpha \lor \beta$. Hence

$$(y[i[1], j[1], p] \land y[i[2], j[2], p]) \Rightarrow x[i[1], j[1], l] = x[i[2], j[2], l] \quad (1)$$

is equivalent to

$$\neg(y[i[1], j[1], p] \land y[i[2], j[2], p]) \lor x[i[1], j[1], l] = x[i[2], j[2], l]. \quad (2)$$

Since $\neg(\alpha \land \beta) \Leftrightarrow \neg\alpha \lor \neg\beta$, in view of (2), we obtain that (1) is equivalent to

$$\neg y[i[1], j[1], p] \lor \neg y[i[2], j[2], p]) \lor x[i[1], j[1], l] = x[i[2], j[2], l]. \quad (3)$$

Note that $\alpha = \beta \Leftrightarrow (\neg\alpha \lor \beta) \land (\alpha \lor \neg\beta)$. Hence (3) is equivalent to

$$\neg y[i[1], j[1], p] \lor (\neg y[i[2], j[2], p]) \lor (\neg x[i[1], j[1], l] \lor x[i[2], j[2], l]) \land (\neg x[i[1], j[1], l] \lor x[i[2], j[2], l]). \quad (4)$$

In view of $\alpha \lor (\beta \land \gamma) \Leftrightarrow (\alpha \lor \beta) \land (\alpha \lor \gamma)$, it is easy to see that (4) is equivalent to

$$\neg y[i[1], j[1], p] \lor \neg y[i[2], j[2], p] \lor \neg x[i[1], j[1], l] \lor x[i[2], j[2], l]) \land$$

$$\neg y[i[1], j[1], p] \lor y[i[2], j[2], p] \lor x[i[1], j[1], l] \lor \neg x[i[2], j[2], l]). \quad (5)$$

Hence (1) is equivalent to (5). Therefore, $\xi \Leftrightarrow \xi'$. 

Theorem 1 gives us an explicit reduction from LCS to SAT.

If $|\Sigma| = 1$, then

$$\varphi'[i, j] = (x[i, j, 1] \lor z_1[1, j, 1] \lor z_1[i, j, 2])) \land (x[i, j, 1] \lor \neg z_1[1, j, 1] \lor z_1[i, j, 2])$$

$$\land (x[i, j, 1] \lor z_1[1, j, 1] \lor \neg z_1[i, j, 2]) \land (x[i, j, 1] \lor \neg z_1[1, j, 1] \lor z_1[i, j, 2])$$

$$\land \land_{1 \leq l[1] \leq |\Sigma|, 1 \leq l[2] \leq |\Sigma|, 1 \leq j[1] \leq |\Sigma|, 1 \leq j[2] \leq |\Sigma|}((\neg x[i, j, l[1]] \lor \neg x[i, j, l[2]] \lor z_2[i, j, l[1], l[2]]) \land$$

$$\neg x[i, j, l[1]] \lor \neg x[i, j, l[2]] \lor \neg z_2[i, j, l[1], l[2]]).$$
If $|\Sigma| = 2$, then

$$\varphi'[i, j] = (x[i, j, 1] \lor x[i, j, 2] \lor z_3[i, j, 1]) \land (x[i, j, 1] \lor x[i, j, 2] \lor \neg z_3[i, j, 1]) \land$$

$$\bigwedge_{1 \leq l \leq |\Sigma|, 1 \leq (l[1] \lor l[2]) \leq |\Sigma|, l[1] \neq l[2]}((\neg x[i, j, l[1]] \lor \neg x[i, j, l[2]] \lor z_4[i, j, l[1], l[2]]) \land$$

$$\neg x[i, j, l[1]] \lor \neg x[i, j, l[2]] \lor \neg z_4[i, j, l[1], l[2]]).$$

If $|\Sigma| = 3$, then

$$\varphi'[i, j] = (\lor_{1 \leq l \leq |\Sigma|} x[i, j, l]) \land$$

$$\bigwedge_{1 \leq l \leq |\Sigma|, 1 \leq (l[1] \lor l[2]) \leq |\Sigma|, l[1] \neq l[2]}((\neg x[i, j, l[1]] \lor \neg x[i, j, l[2]] \lor z_5[i, j, l[1], l[2]]) \land$$

$$\neg x[i, j, l[1]] \lor \neg x[i, j, l[2]] \lor \neg z_5[i, j, l[1], l[2]]).$$

If $|\Sigma| = 4$, then

$$\varphi'[i, j] = (x[i, j, 1] \lor x[i, j, 2] \lor z_6[i, j, 1]) \land (\neg z_6[i, j, 1] \lor x[i, j, 3] \lor x[i, j, 4]) \land$$

$$\bigwedge_{1 \leq l \leq |\Sigma|, 1 \leq (l[1] \lor l[2]) \leq |\Sigma|, l[1] \neq l[2]}((\neg x[i, j, l[1]] \lor \neg x[i, j, l[2]] \lor z_7[i, j, l[1], l[2]]) \land$$

$$\neg x[i, j, l[1]] \lor \neg x[i, j, l[2]] \lor \neg z_7[i, j, l[1], l[2]]).)$$

If $|\Sigma| > 4$, then

$$\varphi'[i, j] = (x[i, j, 1] \lor x[i, j, 2] \lor z_8[i, j, 1]) \land$$

$$\bigwedge_{3 \leq l \leq |\Sigma| - 2}(-z_8[i, j, l - 2] \lor x[i, j, l] \lor z_8[i, j, l - 1]) \land$$

$$(-z_8[i, j, |\Sigma| - 3] \lor x[i, j, |\Sigma| - 1] \lor x[i, j, |\Sigma|]) \land$$

$$\bigwedge_{1 \leq l \leq |\Sigma|, 1 \leq (l[1] \lor l[2]) \leq |\Sigma|, l[1] \neq l[2]}((\neg x[i, j, l[1]] \lor \neg x[i, j, l[2]] \lor z_9[i, j, l[1], l[2]]) \land$$

$$\neg x[i, j, l[1]] \lor \neg x[i, j, l[2]] \lor \neg z_9[i, j, l[1], l[2]]).$$

Let $\varphi' = \bigwedge_{1 \leq i \leq m, 1 \leq j \leq |S|} \varphi'[i, j]$. Let

$$\psi'[1, i, j] = \bigwedge_{1 \leq k \leq |\Sigma|, 1 \leq l \leq |\Sigma|}((-y[i, j, l[1]] \lor \neg y[i, j, l[2]] \lor z_{10}[i, j, l[1], l[2]]) \land$$

$$\bigwedge_{1 \leq l[2] \leq k, l[1] \neq l[2]}(-y[i, j, l[1]] \lor \neg y[i, j, l[2]] \lor \neg z_{10}[i, j, l[1], l[2]]),$$

$$\psi'[1] = \bigwedge_{1 \leq i \leq m} \psi'[1, i, j],$$

$$\psi'[2, i, j, l] = \bigwedge_{1 \leq j \leq |S|}((-y[i, j, l] \lor \neg y[i, j, l[1], l] \lor z_{11}[i, j, j[1], l]) \land$$

$$j[1] \neq j$$

$$\bigwedge_{1 \leq i \leq |S|}(-y[i, j, l] \lor \neg y[i, j, l[1], l] \lor \neg z_{11}[i, j, j[1], l]),$$
\[ \psi'[2] = \land_{1 \leq i \leq m} \land_{1 \leq j \leq |S_i|} \land_{1 \leq l \leq k} \]

\[ \psi'[3, i, j, l] = \land_{1 \leq i \leq |S_i|} ((\neg y[i, j, l] \lor \neg y[i, j[1], l[1]] \lor z_{12}[i, j, j[1], l, l[1]]) \land \\
\neg y[i, j[1], l[1]] \lor \neg y[i, j[j[1], l[1]] \lor z_{12}[i, j, j[1], l, l[1]])) \land \\
(\neg y[i, j, l] \lor \neg y[i, j[1], l[1]] \lor \neg z_{12}[i, j, j[1], l, l[1]]), \]

If \( |S_i| = 1 \), then

\[ \psi'[4, i, l] = (y[i, 1, l] \lor z_{13}[i, 1, l] \lor z_{13}[i, 2, l]) \land (y[i, 1, l] \lor \neg z_{13}[i, 1, l] \lor z_{13}[i, 2, l]) \land \\
(y[i, 1, l] \lor z_{13}[i, 1, l] \lor \neg z_{13}[i, 2, l]) \land (y[i, 1, l] \lor \neg z_{13}[i, 1, l] \lor \neg z_{13}[i, 2, l]). \]

If \( |S_i| = 2 \), then

\[ \psi'[4, i, l] = (y[i, 1, l] \lor y[i, 2, l] \lor z_{14}[i, 2, l]) \land (y[i, 1, l] \lor y[i, 2, l] \lor \neg z_{14}[i, 2, l]). \]

If \( |S_i| = 3 \), then \( \psi'[4, i, l] = \psi[4, i, l] \). If \( |S_i| = 4 \), then

\[ \psi'[4, i, l] = (y[i, 1, l] \lor y[i, 2, l] \lor z_{15}[i, 1, l]) \land (\neg z_{15}[i, 1, l] \land y[i, 3, l] \land y[i, 4, l]). \]

If \( |S_i| > 4 \), then

\[ \psi'[4, i, l] = (y[i, 1, l] \lor y[i, 2, l] \lor z_{16}[i, 1, l]) \land \\
(\land_{3 \leq j \leq |S_i| - 3} (\neg z_{16}[i, j - 2, l] \lor y[i, j, l] \lor z_{16}[i, j - 1, l])) \land \\
(\neg z_{16}[i, |S_i| - 3, l] \lor y[i, |S_i| - 1, l] \lor y[i, |S_i|, l]). \]

Let \( \psi''[4] = \land_{1 \leq i \leq m, 1 \leq j \leq k} \psi'[4, i, l] \). Let

\[ \xi'' = \land_{1 \leq i \leq |S_i|} \land_{1 \leq j \leq |S_i|} ( (\neg y[i[1], j[1], p] \lor \neg y[i[2], j[2], p]) \lor \\
\neg z_{17}[i[1], j[1], i[2], j[2], p, l]) \land \\
(\neg z_{17}[i[1], j[1], i[2], j[2], p, l] \lor \neg x[i[1], j[1], l] \lor x[i[2], j[2], l]) \land \\
(\neg y[i[1], j[1], p] \lor \neg y[i[2], j[2], p] \lor z_{18}[i[1], j[1], i[2], j[2], p, l]) \land \\
(\neg z_{18}[i[1], j[1], i[2], j[2], p, l] \lor x[i[1], j[1], l] \lor \neg x[i[2], j[2], l])). \]
Let $\tau' = \varphi' \land (\land_{1 \leq q \leq 4} \psi'[q]) \land \xi''$.

**Theorem 2.** $\tau \Leftrightarrow \tau'$.

**Proof.** Since
\[\alpha \lor \beta \Leftrightarrow (\alpha \lor z) \land (\beta \lor \neg z)\] (6)
where $z$ is a new variable, it is easy to see that $\psi[4] \Leftrightarrow \psi'[4]$ and $\xi' \Leftrightarrow \xi''$. Note that
\[\alpha \Leftrightarrow (\alpha \lor z) \land (\alpha \lor \neg z).\] (7)
Hence $\psi[1] \Leftrightarrow \psi'[1]$, $\psi[2] \Leftrightarrow \psi'[2]$, and $\psi[3] \Leftrightarrow \psi'[3]$. In view of (6) and (7), it is easy to check that $\varphi \Leftrightarrow \varphi'$. Therefore, $\tau \Leftrightarrow \tau'$.

Theorem 2 gives us an explicit reduction from LCS to 3SAT.

We have obtained explicit reductions from LCS to some variants of satisfiability, SAT and 3SAT.

There is a well known site on which solvers for SAT are posted [31]. They are divided into two main classes, stochastic local search algorithms and improved exhaustive search algorithms. All solvers allow the conventional format for recording DIMACS boolean function in conjunctive normal form and solve the corresponding problem [31]. In addition to the solvers the site also represented a large set of test problems in the format of DIMACS. This set includes a randomly generated problems of 3SAT.

We have created a generator of natural instances for LCS. Also, we have used test problems from [31].

We have used algorithms from [31]. Also, we have designed our own genetic algorithm for SAT which based on algorithms from [31].

We have used heterogeneous cluster based on three clusters (Cluster USU, Linux, 8 calculation nodes, Intel Pentium IV 2.40GHz processors; umt, Linux, 256 calculation nodes, Xeon 3.00GHz processors; um64, Linux, 124 calculation nodes, AMD Opteron 2.6GHz bi-processors) [32].

Each test was runned on a cluster of at least 100 nodes. The maximum solution time was 15 hours. The average time to find a solution was 18.7 minutes. The best time was 29 seconds.

**References**


On the longest common subsequence problem


[31] URL: http://people.cs.ubc.ca/~hoos/SATLIB/


Received: July, 2012