Problem statement: The 2D-LCS Problem*

Marko Djukanović

April 14, 2025

1 The 2D-Longest Common Subsequence Problem

1.1 Introduction

The Longest Common Subsequence (LCS) problem is a fundamental optimization problem arising in bioinformatics that rates the structural similarity between two or more biological sequences. It has applications in computational biology, data compression, text editing, etc.

The literature is known for many linear generalizations of the LCS problem. The first inherently multidimensional generalization of the LCS problem is provided in [1] by introducing the concept of *multidimensional sequences*, so-called 2D-sequences. Comparing such sequences is of practical interest in, for example, quantifying the similarity of two images represented by respective matrices of pixels.

1.2 Task

In the 2D-LCS problem, given are two two-dimensional matrices or multidimensional objects. The task is to rate the (structural) similarity between a pair of two-dimensional objects. To do so, one may seek identical symbols in both matrices, preserving their order. This will not always result in a sub-matrix, but rather the symbols common to both matrices, which preserve the order relation between them in both matrices.

1.3 Problem statement

Given is matrix A of dimension $n \times n$ and matrix B of dimension $m \times m$ over a finite alphabet Σ.
The objective is to find a **maximum** domain size 1-1 mapping $f = (f_1, f_2)\colon [n]\times$

*A contribution to the Working group 1 of the Cost Action ROAR-NET

$[n] \mapsto [m] \times [m]$ so that $A[i,j] = B[f((i,j)) = (f_1(i), f_2(j))]$ for each $(i,j) \in Dom(f)$ and for each two elements $(i,j), (i',j') \in Dom(f)$, the following conditions hold:

1. $i < i'$ iff $f_1(i) < f_1(i')$
2. $j < j'$ iff $f_2(j) < f_2(j')$
3. $i = i'$ iff $f_1(i) = f_1(i')$
4. $j = j'$ iff $f_2(j) = f_2(j')$

A feasible *solution* is therefore any (1-1) mapping f from $[n]^2$ to $[m]^2$ satisfying Conditions 1–4.

The objective value of a solution f: domain size of a feasible solution (mapping), i.e. the value $|Dom(f)|$.

It is proven that the 2D-LCS problem is indeed $\mathcal{NP}$-hard, see [1].

1.4 Instance data file

Each *problem instance* is represented by two matrices of characters of dimensions $n \times n$ and $m \times m$, respectively. These two numbers stating the dimensions may be written in the first line of the input file. The subsequent n lines are in charge for storing row elements of the. The last m lines store the elements of the second matrix row-by-row.

The extension of the file can be *.txt*, *.csv*, or some other similar (textual) format.

1.5 Solution file

A solution can be stored in a textual-type file where in the first line elements of the domain are stored, given as a sequence of pairs. The second line keeps respective co-domain values of the solution also as a sequence of pairs.

1.6 Example

For the shake of clarity, given is an example in the subsequent section.

1.6.1 Instance

Input file:

```
3 3
A C C
B D A
A A C
A A C
D B C
B D D
```

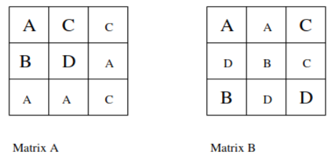

Figure 1: The symbols in bold denote a solution to the instance[1]

Visually, the instance correspond to the matrices shown in Figure 1.

1.6.2 Solution

An optimal solution to the problem problem instance given above is a function $f : X \to Y$ with

$$X = \{(1,1),(1,2),(2,1),(2,2)\}, Y = \{(1,1),(1,3),(3,1),(3,3)\}.$$

So, the objective value of f is equal to 4. The output file looks as follows.

```
(1, 1), (1, 2), (2, 1), (2, 2)
(1, 1), (1, 3), (3, 1), (3, 3)
Size: 4
```

1.6.3 Explanation

One can easily verify that the solution is feasible (after directly checking Conditions 1-4). It can be verified that one cannot choose all symbols of a row (or a column) to be elements of any feasible function f. Thus, an upper bound of the optimal solution must be less or equal to 4.

References

[1] Amir, A., Hartman, T., Kapah, O., Shalom, B. R., & Tsur, D. (2007, October). Generalized LCS. In International Symposium on String Processing and Information Retrieval (pp. 50-61). Berlin, Heidelberg: Springer Berlin Heidelberg (doi: https://doi.org/10.1007/978-3-540-75530-2_5)

[1] The example borrowed from [1]