Polynomial-Time Equivalences and Refined Algorithms for Longest Common Subsequence Variants

Yuichi Asahiro[a] Jesper Jansson[b] Guohui Lin[c]

Eiji Miyano[d] Hirotaka Ono[e] Tadatoshi Utashima[d]

[a] Kyushu Sangyo University
[b] Kyoto University
[c] University of Alberta
[d] **Kyushu Institute of Technology**
[e] Nagoya University

June 29, 2022 / CPM2022

Longest Common Subsequence Problem on Two Sequences

LCS

Input: A pair of sequences X and Y over the alphabet Σ.

Goal: Find a longest common subsequence $LCS(X, Y)$ of X and Y.

Example: Alphabet $\Sigma = \{a, c, g, t\}$

$$X = \langle t, g, a, c, t, c, t, g, t, g, c, a \rangle$$
$$Y = \langle t, g, c, t, c, a, g, t, g, c, a, c \rangle$$
$$LCS(X, Y) = \langle t, g, c, t, c, g, t, g, c, a \rangle$$

Proposition [Hirschberg '75][Needleman et al '70][Sankoff '72]

LCS can be solved in polynomial time.

Proof. DP works well. □

Four Variants of LCS

- In this talk, four variants of LCS are considered.
  - Every has two sequences (X, Y), plus some additional constraints as input.
  - Assume that $|X| = n$ and $|Y| = O(poly(n))$.
  - Every is NP-hard.

1. [Asahiro et al. COCOA 2019 & TCS 2020]
   Repetition-Bounded Longest Common Subsequence (RBLCS)
2. [Mincu et al. SPIRE 2018]
   Multiset Restricted Common Subsequence (MRCS)
3. [Castelli et al. CPM 2017 & TCS 2019]
   One-Side-Filled Longest Common Subsequence (1FLCS)
4. [Castelli et al. CPM 2017 & TCS 2019]
   Two-Side-Filled Longest Common Subsequence (2FLCS)

Our contributions

Result 1 (polynomial-time equivalence)

- Each of MRCS, 1FLCS and 2FLCS is polynomially equivalent to RBLCS.

Polynomial-time equivalence (polynomially equivalent)

- Let $\texttt{ALG}_A$ and $\texttt{ALG}_B$ be (exact exponential) algorithms for P_A and for P_B, respectively.
- We say that P_A and P_B are polynomially equivalent if
  - P_A can be solved by $\texttt{ALG}_B$ with some extra polynomial-time calculations; and
  - P_B can be solved by $\texttt{ALG}_A$ with some extra polynomial-time calculations.

Our contributions

Result 1 (polynomial-time equivalence)

- Each of MRCS, 1FLCS and 2FLCS is polynomially equivalent to RBLCS.

Result 2 (exact exponential algorithms)

- RBLCS can be solved in $O(1.415^n)$ time.

- $|X| = n$ and $|Y| = O(poly(n))$.
- From Results 1 and 2,
  MRCS, 1FLCS and 2FLCS can be also solved in $O(1.415^n)$ time.

Result 3 (approximation algorithm)

- There exists a 2-approximation algorithm for 2FLCS.

Repetition-Bounded LCS problem (RBLCS)

- Let $occ(W, \sigma)$ be the number of occurrences of $\sigma \in \Sigma$ in a sequence W.
- Let C_{occ} be an occurrence constraint on a solution sequence, i.e., a function $C_{occ} : \Sigma \to \mathbb{N}$ assigning an upper bound on the number of occurrences of each symbol in Σ.

RBLCS [Asahiro et al. COCOA 2019 & TCS 2020]

Input: A pair of sequences X and Y, and an occurrence constraint C_{occ}.

Goal: Find a longest common subsequence Z of X and Y such that $occ(Z, \sigma) \leq C_{occ}(\sigma)$ is satisfied for every $\sigma \in \Sigma$.

Example: RBLCS

$$X = \langle t, g, t, c, a, c, g, t, g, a, a, g \rangle$$
$$Y = \langle a, t, g, c, a, t, g, g, a, c, a, g, c \rangle$$
$$C_{occ}(a) = 1, C_{occ}(c) = 1, C_{occ}(g) = 2, C_{occ}(t) = 1$$

- $Z = \langle g, c, t, g, a \rangle$ of length 5 is an optimal solution since $occ(Z, a) = 1$, $occ(Z, c) = 1$, $occ(Z, g) = 2$, $occ(Z, t) = 1$
- Note that $\langle t, g, c, a, t, g, a, a, g \rangle$ of length 9 is an original LCS solution.

Repetition-Bounded LCS problem (RBLCS)

- Let $occ(W, \sigma)$ be the number of occurrences of $\sigma \in \Sigma$ in a sequence W.
- Let C_{occ} be an occurrence constraint on a solution sequence, i.e., a function $C_{occ} : \Sigma \to \mathbb{N}$ assigning an upper bound on the number of occurrences of each symbol in Σ.

RBLCS [Asahiro et al. COCOA 2019 & TCS 2020]

Input: A pair of sequences X and Y, and an occurrence constraint C_{occ}.

Goal: Find a longest common subsequence Z of X and Y such that $occ(Z, \sigma) \leq C_{occ}(\sigma)$ is satisfied for every $\sigma \in \Sigma$.

[Asahiro et al. COCOA 2019 & TCS 2020] previously proved

- NP-hard (APX-hard)
- RBLCS can be solved in $O(1.442^n)$ time by a DP-based algorithm.

New result

Result 2 (exact exponential algorithms)

- RBLCS can be solved in $O(1.415^n)$ time by using a smaller DP-table.

Multiset-Restricted Common Subsequence problem (MRCS)

MRCS [Mincu et al. SPIRE 2018]

Input: A pair of sequences X and Y, and a multiset $\mathcal{M}$.

Goal: Find a common subsequence Z of X and Y such that Z contains the maximum number of symbols from $\mathcal{M}$.

Example: MRCS

$$X = \langle t, g, t, c, a, c, g, t, g, a, a, g \rangle$$
$$Y = \langle a, t, g, c, a, t, g, g, a, c, a, g, c \rangle$$
$$\mathcal{M} = \{a, c, g, g, t\}$$

- $Z = \langle g, c, t, g, a \rangle$ of length 5 is an optimal solution since $|\mathcal{M}| = 5$ and Z has one a, one c, two g's, and one t.
- $Z' = \langle g, c, t, g, a, a, g \rangle$ of length 7 is another optimal solution since $|\mathcal{M} \cap Z| = 5$ and Z' also has one a, one c, two g's, and one t.
- Note that the solution value is at most $|\mathcal{M}|$.

Warm-up: Equivalence of RBLCS and MRCS

Simple Observation

- A multiset $\mathcal{M}$ of MRCS can be seen as an occurrence constraint C_{occ} of RBLCS.
- An occurrence constraint C_{occ} of RBLCS can be seen as a multiset $\mathcal{M}$ of MRCS.

MRCS

$$X = \langle t, g, t, c, a, c, g, t, g, a, a, g \rangle$$
$$Y = \langle a, t, g, c, a, t, g, g, a, c, a, g, c \rangle$$
$$\mathcal{M} = \{a, c, g, g, t\}$$
$$Z = \langle g, c, t, g, a \rangle$$

$$\Downarrow \quad \Uparrow$$

RBLCS

$$X = \langle t, g, t, c, a, c, g, t, g, a, a, g \rangle$$
$$Y = \langle a, t, g, c, a, t, g, g, a, c, a, g, c \rangle$$
$$C_{occ}(a) = 1, C_{occ}(c) = 1, C_{occ}(g) = 2, C_{occ}(t) = 1$$
$$Z = \langle g, c, t, g, a \rangle$$

Warm-up: Equivalence of RBLCS and MRCS

Theorem

- Consider a pair of a multiset $\mathcal{M}$ in an input for MRCS and an occurrence constraint C_{occ} of symbols in Σ in an input for RBLCS such that $C_{occ}(\sigma) = occ(\mathcal{M}, \sigma)$ for every $\sigma \in \Sigma$.

Then, the followings hold:

1. Given an optimal solution Z_R for an input (X, Y, C_{occ}) of RBLCS, we can obtain an optimal solution for an input $(X, Y, \mathcal{M})$ of MRCS in polynomial time.
2. Given an optimal solution $Z_{\mathcal{M}}$ for an input $(X, Y, \mathcal{M})$ of MRCS, we can obtain an optimal solution for an input (X, Y, C_{occ}) of RBLCS in polynomial time.

Namely,

(Part of) Result 1 (polynomial-time equivalence)

- MRCS is polynomially equivalent to RBLCS.

One-Side-Filled LCS problem (1FLCS)

1FLCS [Castelli et al. CPM 2017 & TCS 2019]

Input: A pair of a complete sequence X and an incomplete sequence Y, and a multiset $\mathcal{M}_Y$ of missing symbols.

Goal: Find a filling Y^* such that the length of $LCS(X, Y^*)$ is the longest among the length of $LCS(X, Y^+)$ over all fillings Y^+.

Example: 1FLCS

Input:

- (Complete) reference sequence $X = \langle a, c, a, g, t \rangle$;
- Incomplete sequence $Y = \langle g, c, g, a \rangle$; and
- Multiset $\mathcal{M}_Y = \{a, a, t\}$ of missing symbols.

Goal

- Find a omplete sequence $Y^* = \langle \underline{a}, g, c, \underline{a}, g, \underline{t}, a \rangle$ by filling missing symbols in $\mathcal{M}_Y$ to Y; and
- Find a LCS $LCS(X, Y^*) = \langle \underline{a}, c, \underline{a}, g, \underline{t} \rangle$ of two sequences X and Y^*.

Two-Side-Filled LCS problem (2FLCS)

2FLCS [Castelli et al. CPM 2017 & TCS 2019]

Input: A pair of incomplete sequences X and Y, and a pair of multisets $\mathcal{M}_X$ and $\mathcal{M}_Y$ of missing symbols.

Goal: Find two fillings X^* and Y^* such that the length of $LCS(X^*, Y^*)$ is the longest among the lengths of $LCS(X^+, Y^+)$ over all pairs of X^+ and Y^+.

Incomplete sequence $X = \langle g, t, c, a, c, t, g, a \rangle$

$\mathcal{M}_X = \{g, t\}$

Incomplete sequence $Y = \langle g, a, t, c, c, g, t, g \rangle$

$\mathcal{M}_Y = \{c, t, t\}$

Filling $X + \mathcal{M}_X$ $X^* = \langle \underline{t}, g, t, c, a, c, \underline{g}, t, g, a \rangle$

Filling $Y + \mathcal{M}_Y$ $Y^* = \langle \underline{t}, g, \underline{t}, \underline{c}, a, t, c, c, g, t, g \rangle$

$$LCS(X^*, Y^*) = \langle t, g, t, c, a, c, g, t, g \rangle$$

4 types of matches

- Let "match" be a common symbol of two fillings X^* and Y^*.

$$
\begin{aligned}
X &= \langle g, t, c, a, c, t, g, a \rangle \\
Y &= \langle g, a, t, c, c, g, t, g \rangle \\
\mathcal{M}_X &= \{g, t\} \\
\mathcal{M}_Y &= \{c, t, t\} \\
X^* &= \langle \underline{t}, g, t, c, a, c, \underline{g}, t, g, a \rangle \\
Y^* &= \langle \underline{t}, g, \underline{t}, \underline{c}, a, t, c, c, g, t, g \rangle \\
LCS(X^*, Y^*) &= \langle t, g, t, c, a, c, g, t, g \rangle
\end{aligned}
$$

Each match is one of the following 4 types:

- ($\mathcal{M}_X$-symbol, $\mathcal{M}_Y$-symbol)-match (e.g., 1st symbol "t")
- (X-symbol, Y-symbol)-match (e.g., 2nd symbol "g")
- (X-symbol, $\mathcal{M}_Y$-symbol)-match (e.g., 3rd symbol "t")
- ($\mathcal{M}_X$-symbol, Y-symbol)-match (e.g., 7th symbol "g")

Match Exchanging

Observation

- Every symbol $\sigma \in \mathcal{M}_Y$ (resp. $\mathcal{M}_X$) can be matched to σ at any position in X (resp. Y) without restrictions.

Match exchanging

- (X, Y)-match and ($\mathcal{M}_X$, $\mathcal{M}_Y$)-match can be exchanged to (X, $\mathcal{M}_Y$)-match and ($\mathcal{M}_X$, Y)-match.

$$
\begin{aligned}
X &= \langle \cdots a_X \cdots a_{\mathcal{M}_X} \cdots \rangle \\
Y^+ &= \langle \cdots a_Y \cdots a_{\mathcal{M}_Y} \cdots \rangle \\
&\Downarrow \\
X &= \langle \cdots a_X \; a_{\mathcal{M}_X} \cdots\cdots \rangle \\
Y^+ &= \langle \cdots a_{\mathcal{M}_Y} \; a_Y \cdots\cdots \rangle
\end{aligned}
$$

- By repeating match-exchanging methods, (X, Y)-match-free (or $(\mathcal{M}_X, \mathcal{M}_Y)$-match-free) sequence can be obtained.

Symbol Deleting

Symbol deleting

- A filling-procedure of a symbol $\sigma \in \mathcal{M}_Y$ into Y to match some σ in X can be seen as a deleting-procedure of the matched σ from X.

$$
\begin{aligned}
X &= \langle \cdots\ a\ g\ c \cdots \rangle \\
\mathcal{M}_Y &= \{\cdots, c, g, g, t, \cdots\} \\
Y^+ &= \langle \cdots\ t\ \underline{g}\ c\ \cdots \rangle \quad \text{(filling } g \in \mathcal{M}_Y \text{ between } t \text{ and } c \text{ in } Y\text{)} \\
&\Downarrow \\
X^- &= \langle \cdots\ a\ \Box\ c\ \cdots \rangle \quad \text{(deleting } g \text{ between } a \text{ and } c \text{ from } X\text{)} \\
\mathcal{M}_Y^- &= \{\cdots, c, \Box, g, t, \cdots\} \quad \text{(deleting one } g \text{ from } \mathcal{M}_Y\text{)} \\
Y &= \langle \cdots\ t\ c\ \cdots \rangle
\end{aligned}
$$

- Longest length on $(X, Y, \mathcal{M}_X, \mathcal{M}_Y)$ = Longest length on $(X^-, Y, \mathcal{M}_X, \mathcal{M}_Y^-)$ $+1$

Similarly,

- Longest length on $(X, Y, \mathcal{M}_X, \mathcal{M}_Y)$ = Longest length on $(X, Y^-, \mathcal{M}_X^-, \mathcal{M}_Y)$ $+1$

Polynomial-time equivalence of RBLCS and 1FLCS

Repetition-Bounded LCS problem

- Occurrence constraint on the output sequence

RBLCS [Asahiro et al. COCOA 2019 & TCS 2020]

Input: A pair of sequences X and Y, and an occurrence constraint C_{occ}.

Goal: Find a longest common subsequence Z of X and Y such that $occ(Z, \sigma) \leq C_{occ}(\sigma)$ is satisfied for every $\sigma \in \Sigma$.

One-Side-Filled LCS problem

- One complete, one incomplete sequences and missing symbols

1FLCS [Castelli et al. CPM 2017 & TCS 2019]

Input: A pair of a complete sequence X and an incomplete sequence Y, and a multiset $\mathcal{M}_Y$ of missing symbols.

Goal: Find a filling Y^* such that the length of $LCS(X, Y^*)$ is the longest among the length of $LCS(X, Y^+)$ over all fillings Y^+.

Polynomial-time equivalence of RBLCS and 1FLCS

Observations

- After all σ's in $\mathcal{M}_Y$ are matched in 1FLCS, the number of remaining unmatched σ's in X is $occ(X,\sigma) - occ(\mathcal{M}_Y,\sigma)$, which can be seen as the occurrence constraint $C_{occ}(\sigma)$ of the input (X, Y, C_{occ}) for RBLCS.
- The number $occ(X,\sigma) - C_{occ}(\sigma)$ of σ's in X for RBLCS can be seen as the number of σ's in $\mathcal{M}_Y$ for 1FLCS.

1FLCS

$$X = \langle a\ c\ a\ g\ c\ g\ a\ c\ t \rangle$$
$$\mathcal{M}_Y = \{c, c, g, t\}$$
$$Y = \langle a\ c\ g\ a\ g\ a\ c\ t \rangle$$

$$\Downarrow \qquad \Uparrow$$

RBLCS

$$X = \langle a\ \boxed{\mathrm{c}}\ a\ g\ \boxed{\mathrm{c}}\ \boxed{\mathrm{g}}\ a\ c\ \boxed{\mathrm{t}} \rangle$$
$$Y = \langle a\ c\ g\ a\ g\ a\ c\ t \rangle$$
$$C_{occ}(a) = 3,\ C_{occ}(c) = 1,\ C_{occ}(g) = 1,\ C_{occ}(t) = 0$$

Polynomial-time equivalence of RBLCS and 1FLCS

Lemma

- Given an input triple $(X, Y, \mathcal{M}_Y)$ of 1FLCS, we can construct an input triple (X, Y, C_{occ}) of RBLCS satisfying $C_{occ}(\sigma) = occ(X, \sigma) - occ(\mathcal{M}_Y, \sigma)$ for every $\sigma \in \Sigma$ in polynomial time.
- Given an input triple (X, Y, C_{occ}) of RBLCS, we can construct an input triple $(X, Y, \mathcal{M}_Y)$ of 1FLCS satisfying $occ(\mathcal{M}_Y, \sigma) = occ(X, \sigma) - C_{occ}(\sigma)$ for every $\sigma \in \Sigma$ in polynomial time.

1FLCS

$$X = \langle a\ c\ a\ g\ c\ g\ a\ c\ t \rangle$$
$$\mathcal{M}_Y = \{c, c, g, t\}$$
$$Y = \langle a\ c\ g\ a\ g\ a\ c\ t \rangle$$
$$\Downarrow \qquad \Uparrow$$

RBLCS

$$X = \langle a\ \boxed{\mathrm{c}}\ a\ g\ \boxed{\mathrm{c}}\ \boxed{\mathrm{g}}\ a\ c\ \boxed{\mathrm{t}} \rangle$$
$$Y = \langle a\ c\ g\ a\ g\ a\ c\ t \rangle$$
$$C_{occ}(a) = 3,\ C_{occ}(c) = 1,\ C_{occ}(g) = 1,\ C_{occ}(t) = 0$$

Polynomial-time equivalence of RBLCS and 1FLCS

- Consider a pair of inputs $(X, Y, \mathcal{M}_Y)$ for 1FLCS and (X, Y, C_{occ}) for RBLCS such that $C_{occ}(\sigma) = occ(X, \sigma) - occ(\mathcal{M}_Y, \sigma)$ holds for every $\sigma \in \Sigma$.
- Let $Z_F = LCS(X, Y, \mathcal{M}_Y)$ and Y^* be an optimal filling for 1FLCS.
- Let $Z_R = LCS(X, Y, C_{occ})$ be an optimal solution for RBLCS.

Theorem

1. Given an optimal solution Z_R for RBLCS, we can obtain an optimal solution for 1FLCS in polynomial time.
2. Given an optimal filling Y^* (or solution Z_F) for 1FLCS, we can obtain an optimal solution for RBLCS in polynomial time.

Namely,

(Part of) Result 1 (polynomial-time equivalence)

- 1FLCS is polynomially equivalent to RBLCS.

Polynomial-time equivalence of RBLCS and 2FLCS

2FLCS [Castelli et al. CPM 2017 & TCS 2019]

Input: A pair of incomplete sequences X and Y, and a pair of multisets $\mathcal{M}_X$ and $\mathcal{M}_Y$ of missing symbols.

Goal: Find two fillings X^* and Y^* such that the length of $LCS(X^*, Y^*)$ is the longest among the lengths of $LCS(X^+, Y^+)$ over all pairs of X^+ and Y^+.

- For an input $(X, Y, \mathcal{M}_X, \mathcal{M}_Y)$ of 2FLCS, consider an occurrence constraint $C_{occ}(\sigma) = \min\{occ(X, \sigma) - occ(\mathcal{M}_Y, \sigma), occ(Y, \sigma) - occ(\mathcal{M}_X, \sigma)\}$ for every $\sigma \in \Sigma$.

Theorem

Given an optimal solution Z_R of RBLCS on (X, Y, C_{occ}), we can obtain optimal fillings X^* and Y^* of 2FLCS on $(X, Y, \mathcal{M}_X, \mathcal{M}_Y)$ in polynomial time.

(Part of) Result 1 (polynomial-time equivalence)

- 2FLCS is polynomially equivalent to RBLCS.

Exact Algorithms for RBLCS

Result 2 (exact exponential algorithms)

- RBLCS can be solved in $O(1.415^n)$ time.

Proof.

- The basic ideas are very similar to the previous $O(1.443^n)$-time DP-based algorithm; but
- we can show the DP-table size can be reduced to $O(1.415^n)$ from $O(1.443^n)$. $\square$

Corollary

MRCS, 1FLCS, and 2FLCS can be also solved in $O(1.415^n)$ time.

Proof. Polynomial-time equivalences + $O(1.415^n)$-time algorithm for RBLCS. $\square$

Approximation Algorithm for 2FLCS

- Consider an input $(X, Y, \mathcal{M}_X, \mathcal{M}_Y)$ of 2FLCS.

$$
\begin{aligned}
X &= \langle g, t, c, a, c, t, g, a \rangle \\
Y &= \langle g, a, t, c, c, g, t, g \rangle \\
\mathcal{M}_X &= \{g, t\} \\
\mathcal{M}_Y &= \{t, c, t\}
\end{aligned}
$$

Algorithm ALG

1. Find $\langle t, c, t \rangle$ by scanning X from left to right, and construct a filling $Y^+ = \langle \mathcal{M}_Y \rangle \circ Y = \langle t, c, t \rangle \circ Y$.
2. Find $\langle g, t \rangle$ by scanning Y from left to right, and construct a filling $X^+ = X \circ \langle \mathcal{M}_X \rangle = X \circ \langle g, t \rangle$.
3. Find a LCS of two fillings X^+ and Y^+ (denoted by $LCS(X^+, Y^+)$).

Approximation Algorithm for 2FLCS

Theorem

Algorithm ALG is a polynomial-time 2-approximation algorithm for 2FLCS on an input $(X, Y, \mathcal{M}_X, \mathcal{M}_Y)$.

Proof.

- By repeating the symbol-deleting methods, we can show:

$$OPT \leq |LCS(X, Y)| + |\mathcal{M}_X| + |\mathcal{M}_Y|.$$

- Let ALG be the length obtained by our algorithm ALG. Namely,

$$ALG = |LCS(X \circ \langle \mathcal{M}_X \rangle, \langle \mathcal{M}_Y \rangle \circ Y)|.$$

- Therefore, $ALG \geq |LCS(X, Y)|$ and $ALG \geq |\mathcal{M}_X| + |\mathcal{M}_Y|$.
- The approximation ratio is:

$$\frac{OPT}{ALG} \leq \frac{|LCS(X,Y)| + |\mathcal{M}_X| + |\mathcal{M}_Y|}{\max\{|LCS(X,Y)|, |\mathcal{M}_X| + |\mathcal{M}_Y|\}} \leq \frac{2(|LCS(X,Y)| + |\mathcal{M}_X| + |\mathcal{M}_Y|)}{|LCS(X,Y)| + |\mathcal{M}_X| + |\mathcal{M}_Y|} = 2$$

Symbol Deleting (revisited)

Symbol deleting

- A filling-procedure of a symbol $\sigma \in \mathcal{M}_Y$ into Y to match some σ in X can be seen as a deleting-procedure of the matched σ from X.

$$X = \langle \cdots \ a \ g \ c \cdots \rangle$$
$$\mathcal{M}_Y = \{\cdots, c, g, g, t, \cdots\}$$
$$Y^+ = \langle \cdots \ t \ \underline{g} \ c \ \cdots \rangle \quad \text{(filling } g \in \mathcal{M}_Y \text{ between } t \text{ and } c \text{ in } Y\text{)}$$
$$\Downarrow$$
$$X^- = \langle \cdots \ a \ \Box \ c \ \cdots \rangle \quad \text{(deleting } g \text{ between } a \text{ and } c \text{ from } X\text{)}$$
$$\mathcal{M}_Y^- = \{\cdots, c, \Box, g, t, \cdots\}$$
$$Y = \langle \cdots \ t \ c \ \cdots \rangle$$

$$\begin{aligned} & \text{Longest length on } (X, Y, \mathcal{M}_X, \mathcal{M}_Y) \\ = \ & \text{Longest length on } (X^-, Y, \mathcal{M}_X, \mathcal{M}_Y^-) +1 \\ = \ & \text{Longest length on } (X^-, Y^-, \mathcal{M}_X^-, \mathcal{M}_Y^-) +2 \\ = \ & \cdots = \text{Longest length on } (X', Y', \emptyset, \emptyset) + |\mathcal{M}_X| + |\mathcal{M}_Y| \\ \leq \ & |LCS(X, Y)| + |\mathcal{M}_X| + |\mathcal{M}_Y| \end{aligned}$$

Summary

New results:

- Polynomial-time equivalences among RBLCS, MRCS, 1FLCS, and 2FLCS.
- $O(1.413^n)$-time algorithm for RBLCS, MRCS, 1FLCS, and 2FLCS.
- Polynomial-time 2-approximation algorithm for 2FLCS.

Previous results:

- $O(1.442^n)$-time algorithm for RBLCS.
- $O(|X||Y|(t+1)^{|\Sigma|})$-time algorithm for MRCS,
  - where t is the maximum multiplicity of symbols in $\mathcal{M}$, and
  - Σ is the alphabet set.
- $O(|X|^{|\Sigma|+2}|Y|)$-time algorithm for 1FLCS.
- $O(2^{O(k)}poly(|X|+|Y|+|\mathcal{M}_Y|))$-time algorithm for 1FLCS,
  - where k is the number of (X, $\mathcal{M}_Y$)-matches in $LCS(X, Y^*)$.
- Polynomial-time 1.667-approximation algorithm for 1FLCS.
- No results on algorithms for 2FLCS.