Available online at www.sciencedirect.com

SCIENCE DIRECT®

Information Processing Letters 99 (2006) 203–207

Information Processing Letters

www.elsevier.com/locate/ipl

# A linear space algorithm for computing a longest common increasing subsequence

Yoshifumi Sakai

*Graduate School of Agricultural Science, Tohoku University, Tsutsumidori-amamiyamachi, Aoba-ku, Sendai-shi, 981-8555 Japan*

Received 21 September 2005; received in revised form 6 January 2006

Available online 9 June 2006

Communicated by M. Yamashita

**Abstract**

Let $X$ and $Y$ be sequences of integers. A common increasing subsequence of $X$ and $Y$ is an increasing subsequence common to $X$ and $Y$. In this note, we propose an $O(|X| \cdot |Y|)$-time and $O(|X| + |Y|)$-space algorithm for finding one of the longest common increasing subsequences of $X$ and $Y$, which improves the space complexity of Yang et al. [I.H. Yang, C.P. Huang, K.M. Chao, A fast algorithm for computing a longest common increasing subsequence, Inform. Process. Lett. 93 (2005) 249–253] $O(|X| \cdot |Y|)$-time and $O(|X| \cdot |Y|)$-space algorithm, where $|X|$ and $|Y|$ denote the lengths of $X$ and $Y$, respectively.


*Keywords:* Algorithms; Longest common subsequence; Longest increasing subsequence

## 1. Introduction

We consider the longest common increasing subsequence (LCIS) problem, whose goal is to find one of the longest increasing subsequences common to all given sequences of integers. This problem is a simple generalization of a classic computer science problem of finding one of the longest increasing subsequences (LISs) of a single sequence of integers [2,8]. As well, the LCIS problem is closely linked to another classic computer science problem: finding one of the longest common subsequences (LCSs) of all given sequences [5–7]. For a sequence $S_0$ in which no element appears more than once, the LCS problem for sequences $S_0, S_1, \ldots, S_k$ coincides with the LCIS problem for integer sequences $X_1, \ldots, X_k$, where each $X_i$ is obtained from $S_i$ by replacing each element in $S_i$ identical with the $s$th element in $S_0$ by the integer $s$, and deleting all elements in $S_i$ not appearing in $S_0$. The relationship between the LCIS problem and the LCS problem can be applied when computing the alignment of whole genomes [4].

In this note, we only focus on the LCIS problem for two sequences. For any sequence $S$, we use $|S|$ to denote the length of $S$, and $S[s]$ to denote the $s$th element of $S$. That is, $S = S[1]S[2] \cdots S[|S|]$. A subsequence of a sequence $S$ is a sequence $S[s_1]S[s_2] \cdots S[s_t]$ for any length $0 \leqslant t \leqslant |S|$ and any indices $1 \leqslant s_1 < s_2 < \cdots < s_t \leqslant |S|$. For integer sequences, $X$ and $Y$, and integers, $l$ and $u$, an $(X, Y, l, u)$-*common increasing sequence* ($(X, Y, l, u)$-*CIS*) is a common subsequence $Z$ of $X$ and $Y$ such that $l < Z[1] < Z[2] < \cdots < Z[|Z|] < u$. The LCIS problem is to find one of the longest such $(X, Y, l, u)$-CISs for any given integer sequences, $X$ and $Y$, and any integers, $l$ and $u$.

*E-mail address:* sakai@biochem.tohoku.ac.jp (Y. Sakai).

0020-0190/$ – see front matter 
doi:10.1016/j.ipl.2006.05.005

Recently, Yang et al. [9] proposed an $O(|X| \cdot |Y|)$-time and $O(|X| \cdot |Y|)$-space algorithm for the LCIS problem, which improves the time complexity of the straightforward algorithm based on the relationship between the LCIS problem and the LCS problem mentioned earlier. In this note, we propose an $O(|X| \cdot |Y|)$-time and $O(|X| + |Y|)$-space algorithm for the LCIS problem, which improves the space complexity of Yang et al.'s algorithm. Subsequently to [9], several faster algorithms were obtained for the LCIS problem in special cases, for example, where the number of pairs $\langle x, y\rangle$ such that $X[x] = Y[y]$ is relatively small [3], and where the length of the LCIS of $X$ and $Y$ is relatively small [1]. In particular, Brodal el al.'s algorithm [1] has the same linear space complexity as our algorithm, although the time complexities of these algorithms are incomparable when the length of the LCIS of $X$ and $Y$ is unrestricted.

## 2. The algorithm

In this section, we use the following notations and terminology: For a sequence $S$ and an index $0 \leqslant s \leqslant |S|+1$, $S[1..s]$ denotes the prefix of $S$ with length $s$, and $S[s..|S|]$ denotes the suffix of $S$ with length $|S| + 1 - s$. For sequences $S$ and $T$, $S \cdot T$ denotes the concatenation $S[1]S[2]\cdots S[|S|]T[1]T[2]\cdots T[|T|]$. For a sequence $S$, the *head* element of $S$ is $S[1]$, and the *tail* element of $S$ is $S[|S|]$.

The algorithm proposed in this note is based on Hirschberg's divide-and-conquer method of solving the LCS problem in linear space [6]. In order to apply the method, we need the following definitions which will be used to divide the LCIS problem into two subproblems. For integer sequences, $X$, $Y$ and $Z$, we say $\langle x_1, y_1\rangle\langle x_2, y_2\rangle \cdots \langle x_{|Z|}, y_{|Z|}\rangle$ is an *index sequence of* $\langle X, Y\rangle$ *representing* $Z$, if $1 \leqslant x_1 < x_2 < \cdots < x_{|Z|} \leqslant |X|$, $1 \leqslant y_1 < y_2 < \cdots < y_{|Z|} \leqslant |Y|$, and $X[x_z] = Y[y_z] = Z[z]$ for any index $1 \leqslant z \leqslant |Z|$. Let the *center* of an index sequence $\langle x_1, y_1\rangle\langle x_2, y_2\rangle \cdots \langle x_{|Z|}, y_{|Z|}\rangle$ of $\langle X, Y\rangle$ representing nonempty $Z$ be $\langle x_z, y_z\rangle$ such that

$$z = \begin{cases} 1 & \text{if } y_1 > \lceil |Y|/2\rceil, \\ \max\{k \mid y_k \leqslant \lceil |Y|/2\rceil\} & \text{otherwise.} \end{cases}$$

Let the center of an empty index sequence be empty.

Based on the divide-and-conquer method [6], we first prove the following lemma.

**Lemma 1.** *Assume that, for any integer sequences, X and Y, and any integers, l and u, the center of an index sequence of* $\langle X, Y\rangle$ *representing one of the longest* $(X, Y, l, u)$*-CISs can be computed in* $O(|X| \cdot |Y|)$ *time and* $O(|X| + |Y|)$ *space. Then, for any integer sequences, X and Y, and any integers, l and u, one of the longest* $(X, Y, l, u)$*-CISs can be computed in* $O(|X| \cdot |Y|)$ *time and* $O(|X| + |Y|)$ *space.*

**Proof.** Since the lemma holds when an empty sequence is the only $(X, Y, l, u)$-CIS, we only consider the case where there exists at least one nonempty $(X, Y, l, u)$-CIS. Let $\langle x, y\rangle$ be the center of any index sequence of $\langle X, Y\rangle$ representing any longest $(X, Y, l, u)$-CIS. Let $X_L = X[1..x-1]$, $Y_L = Y[1..\min(y, \lceil Y/2\rceil) - 1]$, $X_U = X[x+1..|X|]$, and $Y_U = Y[\max(y, \lceil Y/2\rceil) + 1..|Y|]$. Furthermore, let $Z_L$ be any longest $(X_L, Y_L, l, X[x])$-CIS, and $Z_U$ be any longest $(X_U, Y_U, X[x], u)$-CIS.

We first show that the concatenation $Z_L \cdot X[x] \cdot Z_U$ is one of the longest $(X, Y, l, u)$-CISs. Since the tail integer of nonempty $Z_L$ is less than $X[x]$, and $X[x]$ is less than the head integer of nonempty $Z_U$, $Z_L \cdot X[x] \cdot Z_U$ is an $(X, Y, l, u)$-CIS. On the other hand, from the definition of $\langle x, y\rangle$, there exists a longest $(X, Y, l, u)$-CIS $Z$ and an index sequence $\langle x_1, y_1\rangle\langle x_2, y_2\rangle \cdots \langle x_{|Z|}, y_{|Z|}\rangle$ of $\langle X, Y\rangle$ representing $Z$ whose center $\langle x_z, y_z\rangle$ is $\langle x, y\rangle$. Let $y_0 = 0$ and $y_{|Z|+1} = |Y| + 1$. Then, since $Z[1..z-1]$ is a longest $(X_L, Y[1..y-1], l, X[x])$-CIS and $y_{z-1} \leqslant \min(y, \lceil Y/2\rceil) - 1$, $Z[1..z-1]$ is a longest $(X_L, Y_L, l, X[x])$-CIS, which implies that $|Z_L| = |Z[1..z-1]|$. Similarly, since $Z[z+1..|Z|]$ is a longest $(X_U, Y[y+1..|Y|], X[x], u)$-CIS and $y_{z+1} \geqslant \max(y, \lceil Y/2\rceil) + 1$, $|Z_U| = |Z[z+1..|Z|]|$. Therefore, $|Z_L \cdot X[x] \cdot Z_U| = |Z|$. Recall that $Z_L \cdot X[x] \cdot Z_U$ is an $(X, Y, l, u)$-CIS and that $Z$ is a longest $(X, Y, l, u)$-CIS. Thus, $Z_L \cdot X[x] \cdot Z_U$ is a longest $(X, Y, l, u)$-CIS.

Since we assume that $\langle x, y\rangle$ can be computed in $O(|X| \cdot |Y|)$ time and $O(|X| + |Y|)$ space, and both the lengths of $Y_L$ and $Y_U$ are at most $|Y|/2$, if $Z_L$ can be recursively computed in $O(|X_L| \cdot |Y_L|)$ time and $O(|X_L| + |Y_L|)$ space, and if $Z_U$ can be recursively computed in $O(|X_U| \cdot |Y_U|)$ time and $O(|X_U| + |Y_U|)$ space, then it is easy to verify that $Z_L \cdot X[x] \cdot Z_U$ can be computed in $O(|X| \cdot |Y|)$ time and $O(|X| + |Y|)$ space. □

Before explaining how to compute the center of an index sequence of $\langle X, Y\rangle$ representing one of the longest $(X, Y, l, u)$-CISs in $O(|X| \cdot |Y|)$ time and $O(|X| + |Y|)$ space, we need to prove some additional lemmas.

Fix integer sequences, $X$ and $Y$, and integers, $l$ and $u$, arbitrarily. For indices $1 \leqslant x \leqslant |X|$, $1 \leqslant y \leqslant |Y|$, a length $1 \leqslant k \leqslant |X|$ and an integer $a$, let $\mathcal{W}_y^x(k, a)$ be the set of all index sequences $W$ of $\langle X, Y\rangle$ representing any $(X[1..x], Y[1..y], l, u)$-CIS $Z$ such that $|Z| = k$,

$Z[k]=a$, $x_k \leqslant x$ and $y_k \leqslant y$, where $W[k]=\langle x_k, y_k\rangle$. Furthermore, let

$$K^x[y]=\max\bigl(\{k \mid \mathcal{W}^x_y(k, Y[y]) \neq \emptyset\} \cup \{0\}\bigr),$$
$$K_y[x]=\max\bigl(\{k \mid \mathcal{W}^x_y(k, X[x]) \neq \emptyset\} \cup \{0\}\bigr),$$
$$L^x_y[k]=\min\bigl(\{a \mid \mathcal{W}^x_y(k, a) \neq \emptyset\} \cup \{\infty\}\bigr).$$

In other words, $K^x[y]$ (resp. $K_y[x]$) denotes the length of the longest $(X[1..x], Y[1..y], l, u)$-CIS whose tail integer is $Y[y]$ (resp. $X[x]$), while $L^x_y[k]$ denotes the least tail integer of any $(X[1..x], Y[1..y], l, u)$-CIS whose length is $k$. As we will see later, these values play the important roles in computing inductively the center of an index sequence of $\langle X, Y\rangle$ representing one of the longest $(X, Y, l, u)$-CISs. Let $K^0[y]=K_0[x]=0$ and $L^0_y[k]=\infty$. We define three conditions $C_1$, $C_2$ and $C_3$ as follows:

($C_1$) $l < X[x] < u$ and $X[x]=Y[y]$,
($C_2$) $k=K_y[x]$ and $L^{x-1}_y[k] \geqslant X[x]$,
($C_3$) $y \leqslant \lceil |Y|/2\rceil$ or $K_y[x]=1$.

Then we have the following inductive lemmas.

**Lemma 2.** *For any indices* $1 \leqslant x \leqslant |X|$ *and* $1 \leqslant y \leqslant |Y|$,

$$K^x[y]=\begin{cases}\min\{k \mid X[x] \leqslant L^{x-1}_y[k]\} & \textit{if } C_1,\\ K^{x-1}[y] & \textit{otherwise},\end{cases}$$

*and*

$$K_y[x]=\begin{cases}K^x[y] & \textit{if } C_1,\\ K_{y-1}[x] & \textit{otherwise.}\end{cases}$$

**Proof.** Assume $C_1$. It follows from $X[x]=Y[y]$ that $K_y[x]=K^x[y]$. Hence, it suffices to show that $K_y[x]=\min\{k \mid X[x] \leqslant L^{x-1}_y[k]\}$. If $L^{x-1}_y[k] < X[x]$, then for any $W \in \mathcal{W}^{x-1}_y(k, L^{x-1}_y[k])$, the concatenation $W \cdot \langle x, y\rangle$ is in $\mathcal{W}^x_y(k+1, X[x])$. Therefore, it follows from $\mathcal{W}^{x-1}_y(k, L^{x-1}_y[k]) \neq \emptyset$ that $\mathcal{W}^x_y(k+1, X[x]) \neq \emptyset$, and hence, $K_y[x] \geqslant k+1$. Conversely, if $K_y[x] \geqslant k+1$, then for any $W \in \mathcal{W}^x_y(K_y[x], X[x])$, the prefix $W[1..k]$ is in $\mathcal{W}^{x-1}_y(k, X[x'])$, where $W[k]=\langle x', y'\rangle$. Therefore, it follows from $\mathcal{W}^x_y(K_y[x], X[x]) \neq \emptyset$ that $\mathcal{W}^{x-1}_y(k, X[x']) \neq \emptyset$, and hence, $L^{x-1}_y[k] < X[x]$ because $X[x'] < X[x]$. Thus, $K_y[x] \leqslant k$ if and only if $X[x] \leqslant L^{x-1}_y[k]$, which implies that $K_y[x]$ is the least $k$ such that $X[x] \leqslant L^{x-1}_y[k]$.

Assume $\neg C_1$. Then, $\langle x, y\rangle$ is not the tail element of any index sequence of $\langle X, Y\rangle$ representing any $(X[1..x], Y[1..y], l, u)$-CIS. Therefore, $\mathcal{W}^x_y(k, Y[x])=\mathcal{W}^{x-1}_y(k, Y[x])$ for any length $1 \leqslant k \leqslant |X|$, and hence, $K^x[y]=K^{x-1}[y]$. Similarly, $\mathcal{W}^x_y(k, X[x])=\mathcal{W}^x_{y-1}(k, X[x])$ for any length $1 \leqslant k \leqslant |X|$, and hence, $K_y[x]=K_{y-1}[x]$. □

**Lemma 3.** *For any indices* $1 \leqslant x \leqslant |X|$, $1 \leqslant y \leqslant |Y|$, *and any length* $1 \leqslant k \leqslant |X|$,

$$L^x_y[k]=\begin{cases}X[x] & \textit{if } C_2,\\ L^{x-1}_y[k] & \textit{otherwise.}\end{cases}$$

**Proof.** Assume $C_2$. It follows from $k=K_y[x]$ and $\mathcal{W}^x_y(K_y[x], X[x]) \neq \emptyset$ that $L^x_y[k]=L^x_y[K_y[x]] \leqslant X[x]$. On the other hand, $L^x_y[k]$ is equal to either $L^{x-1}_y[k]$ or $X[x]$. Therefore, it follows from $L^{x-1}_y[k] \geqslant X[x]$ that $L^x_y[k] \geqslant X[x]$. Thus, $L^x_y[k]=X[x]$.

Assume $\neg C_2$. Since $L^x_y[k]$ is equal to either $L^{x-1}_y[k]$ or $X[x]$, $L^{x-1}_y[k] \neq X[x]$ implies that $L^x_y[k]=L^{x-1}_y[k]$. Thus, it suffices to show that, if $k \neq K_y[x]$, then $L^{x-1}_y[k] \neq X[x]$. If $k < K_y[x]$, then for any $W \in \mathcal{W}^x_y(K_y[x], X[x])$, the prefix $W[1..k]$ is in $\mathcal{W}^{x-1}_y(k, X[x'])$, where $W[k]=\langle x', y'\rangle$. Therefore, it follows from $\mathcal{W}^x_y(K_y[x], X[x]) \neq \emptyset$ that $\mathcal{W}^{x-1}_y(k, X[x']) \neq \emptyset$, which implies that $L^{x-1}_y[k] < X[x]$ because $X[x'] < X[x]$. On the other hand, if $k > K_y[x]$, then $\mathcal{W}^x_y(k, X[x])=\emptyset$. Therefore, it follows from $\mathcal{W}^{x-1}_y(k, X[x]) \subseteq \mathcal{W}^x_y(k, X[x])$ that $\mathcal{W}^{x-1}_y(k, X[x])=\emptyset$, and hence, $L^{x-1}_y[k] \neq X[x]$. □

Next, we inductively define the values, $I_y[x]$ and $J^x_y[k]$, which will be shown to be the center of an index sequence in $\mathcal{W}^x_y(K_y[x], X[x])$, and the center of an index sequence in $\mathcal{W}^x_y(k, L^x_y[k])$, respectively. Note that, since $\max\{k \mid L^{|X|}_{|Y|}[k] < \infty\}$ is equal to the length of the longest $(X, Y, l, u)$-CISs, $J^{|X|}_{|Y|}[\max\{k \mid L^{|X|}_{|Y|}[k] < \infty\}]$ gives us the center of an index sequence of $\langle X, Y\rangle$ representing one of the longest $(X, Y, l, u)$-CISs, which is what we want to compute. For indices $1 \leqslant x \leqslant |X|$, $1 \leqslant y \leqslant |Y|$, and a length $1 \leqslant k \leqslant |X|$, let

$$I_y[x]=\begin{cases}\langle x, y\rangle & \text{if } C_1 \wedge C_3,\\ J^{x-1}_y[K_y[x]-1] & \text{if } C_1 \wedge \neg C_3,\\ I_{y-1}[x] & \text{otherwise,}\end{cases}$$

and

$$J^x_y[k]=\begin{cases}I_y[x] & \text{if } C_2,\\ J^{x-1}_y[k] & \text{otherwise,}\end{cases}$$

where $I_0[x]$ and $J^0_y[k]$ are empty. Then, we have the following lemma.

**Lemma 4.** *For any indices* $1 \leqslant x \leqslant |X|$ *and* $1 \leqslant y \leqslant |Y|$, *if* $K_y[x] \geqslant 1$, *then there exists* $W \in \mathcal{W}^x_y(K_y[x],$

*$X[x]$) whose center is $I_y[x]$, and for any length $1 \leqslant k \leqslant |X|$, if $L_y^x[k] < \infty$, then there exists $W \in \mathcal{W}_y^x(k, L_y^x[k])$ whose center is $J_y^x[k]$.*

**Proof.** We use the induction method.

Assume that $K_y[x] \geqslant 1$ and $C_1 \wedge C_3$. From $C_1$, there exists $W \in \mathcal{W}_y^x(K_y[x], X[x])$ whose tail element is $\langle x, y\rangle$. From $C_3$, $\langle x, y\rangle$ is the center of $W$.

Assume that $K_y[x] \geqslant 1$ and $C_1 \wedge \neg C_3$. It follows from $K_y[x] \geqslant 1$ and $\neg C_3$ that $K_y[x] \geqslant 2$. Therefore, for any $W \in \mathcal{W}_y^x(K_y[x], X[x])$, the prefix $W[1..K_y[x]-1]$ is in $\mathcal{W}_y^{x-1}(K_y[x]-1, X[x'])$, where $W[K_y[x]-1] = \langle x', y'\rangle$. Hence, it follows from $\mathcal{W}_y^x(K_y[x], X[x]) \neq \emptyset$ that $\mathcal{W}_y^{x-1}(K_y[x]-1, X[x']) \neq \emptyset$, which implies that $L_y^{x-1}[K_y[x]-1] < X[x]$ because $X[x'] < X[x]$. Thus, based on the induction assumption, there exists $W' \in \mathcal{W}_y^{x-1}(K_y[x]-1, L_y^{x-1}[K_y[x]-1])$ whose center is $J_y^{x-1}[K_y[x]-1]$. It follows from $L_y^{x-1}[K_y[x]-1] < X[x]$ and $C_1$ that the concatenation $W' \cdot \langle x, y\rangle$ is in $\mathcal{W}_y^x(K_y[x], X[x])$. Furthermore, from $\neg C_3$, $W' \cdot \langle x, y\rangle$ has the same center of $W'$.

Assume that $K_y[x] \geqslant 1$ and $\neg C_1$. It follows from Lemma 2 that $K_{y-1}[x] = K_y[x]$. Therefore, $K_{y-1}[x] \geqslant 1$, and hence, based on the induction assumption, there exists $W \in \mathcal{W}_{y-1}^x(K_{y-1}[x], X[x])$ whose center is $I_{y-1}[x]$. On the other hand, from $\neg C_1$, $\langle x, y\rangle$ is not the tail element of any index sequence in $\mathcal{W}_y^x(K_y[x], X[x])$, which implies that $\mathcal{W}_y^x(K_y[x], X[x]) = \mathcal{W}_{y-1}^x(K_y[x], X[x])$. Thus, it follows from $K_{y-1}[x] = K_y[x]$ that $\mathcal{W}_y^x(K_y[x], X[x]) = \mathcal{W}_{y-1}^x(K_{y-1}[x], X[x])$, and hence, $W$ is in $\mathcal{W}_y^x(K_y[x], X[x])$.

Assume that $L_y^x[k] < \infty$ and $C_2$. Then, $K_y[x] = k \geqslant 1$, and hence, based on the induction assumption, there exists $W \in \mathcal{W}_y^x(K_y[x], X[x])$ whose center is $I_y[x]$. On the other hand, it follows from Lemma 3 that $L_y^x[k] = X[x]$. Therefore, from $K_y[x] = k$, $\mathcal{W}_y^x(k, L_y^x[k]) = \mathcal{W}_y^x(K_y[x], X[x])$, and hence, $W$ is in $\mathcal{W}_y^x(k, L_y^x[k])$.

Assume that $L_y^x[k] < \infty$ and $\neg C_2$. It follows from Lemma 3 that $L_y^{x-1}[k] = L_y^x[k]$. Therefore, $L_y^{x-1}[k] < \infty$, and hence, based on the induction assumption, there exists $W \in \mathcal{W}_y^{x-1}(k, L_y^{x-1}[k])$ whose center is $J_y^{x-1}[k]$. On the other hand, it follows from the definition that $\mathcal{W}_y^{x-1}(k, L_y^{x-1}[k]) \subseteq \mathcal{W}_y^x(k, L_y^{x-1}[k])$. Therefore, from $L_y^{x-1}[k] = L_y^x[k]$, $\mathcal{W}_y^{x-1}(k, L_y^{x-1}[k]) \subseteq \mathcal{W}_y^x(k, L_y^x[k])$, and hence, $W$ is in $\mathcal{W}_y^x(k, L_y^x[k])$. □

Now we are ready to show the following lemma.

**Lemma 5.** *For any integer sequences, $X$ and $Y$, and any integers, $l$ and $u$, the center of an index sequence of $\langle X, Y\rangle$ representing one of the longest $(X, Y, l, u)$-CISs can be computed in $O(|X| \cdot |Y|)$ time and $O(|X| + |Y|)$ space.*

**Proof.** For any indices $0 \leqslant x \leqslant |X|$ and $1 \leqslant y \leqslant |Y|$, let $H_y^x$ be a $(4|X|+1)$-tuple

$$\big\langle K^x[y],\ K_y[1], \dots, K_y[x], K_{y-1}[x+1], \dots, K_{y-1}[|X|], L_y^x[1], \dots, L_y^x[|X|], I_y[1], \dots, I_y[x], I_{y-1}[x+1], \dots, I_{y-1}[|X|], J_y^x[1], \dots, J_y^x[|X|]\big\rangle.$$

For example, if $X = 4\ 1\ 3$, $Y = 3\ 1\ 7\ 2\ 4\ 3$, $l = 0$ and $u = 5$, then we have

$$H_1^0 = \langle 0,\ 0,0,0,\ \infty,\infty,\infty,\ \text{empty},\text{empty},\text{empty},\ \text{empty},\text{empty},\text{empty}\rangle,$$

$$H_1^1 = \langle 0,\ 0,0,0,\ \infty,\infty,\infty,\ \text{empty},\text{empty},\text{empty},\ \text{empty},\text{empty},\text{empty}\rangle,$$

$$H_1^2 = \langle 0,\ 0,0,0,\ \infty,\infty,\infty,\ \text{empty},\text{empty},\text{empty},\ \text{empty},\text{empty},\text{empty}\rangle,$$

$$H_1^3 = \langle 1,\ 0,0,1,\ 3,\infty,\infty,\ \text{empty},\text{empty},\langle 3,1\rangle,\ \langle 3,1\rangle,\text{empty},\text{empty}\rangle,$$

$$H_2^0 = \langle 0,\ 0,0,1,\ \infty,\infty,\infty,\ \text{empty},\text{empty},\langle 3,1\rangle,\ \text{empty},\text{empty},\text{empty}\rangle,$$

$$\vdots$$

$$H_6^1 = \langle 0,\ 1,1,1,\ 4,\infty,\infty,\ \langle 1,5\rangle,\langle 2,2\rangle,\langle 3,1\rangle,\ \langle 1,5\rangle,\text{empty},\text{empty}\rangle,$$

$$H_6^2 = \langle 0,\ 1,1,1,\ 1,\infty,\infty,\ \langle 1,5\rangle,\langle 2,2\rangle,\langle 3,1\rangle,\ \langle 2,2\rangle,\text{empty},\text{empty}\rangle,$$

$$H_6^3 = \langle 2,\ 1,1,2,\ 1,3,\infty,\ \langle 1,5\rangle,\langle 2,2\rangle,\langle 2,2\rangle,\ \langle 2,2\rangle,\langle 2,2\rangle,\text{empty}\rangle.$$

From the definition of $L_y^x[k]$ and Lemma 4, if $L_{|Y|}^{|X|}[1] < \infty$, then $J_{|Y|}^{|X|}[\max\{k \mid L_{|Y|}^{|X|}[k] < \infty\}]$ is the center of an index sequence of $\langle X, Y\rangle$ representing one of the longest $(X, Y, l, u)$-CISs, otherwise, an empty sequence is the only $(X, Y, l, u)$-CIS. Therefore, it suffices to show that $H_{|Y|}^{|X|}$ can be computed in $O(|X| \cdot |Y|)$ time and $O(|X| + |Y|)$ space. We will show that, only using a memory space of size $O(|X|)$ that can take the value of any $H_y^x$, together with access to $X$ and $Y$, $H_{|Y|}^{|X|}$ can be computed in $O(|X| \cdot |Y|)$ time by successively updating the content of the memory space in order of $H_1^0, H_1^1, \dots, H_1^{|X|}, H_2^0, H_2^1, \dots, H_{|Y|}^{|X|}$ (the same order of the example above), which will complete the proof of the lemma.

Recall that $K^0[y] = K_0[x] = 0$, $L_y^0[k] = \infty$, and both $I_0[x]$ and $J_y^0[k]$ are empty. Hence, $H_1^0$ can be computed in $O(|X|)$ time. Also, for any $2 \leqslant y \leqslant |Y|$, $H_y^0$ can be obtained from $H_{y-1}^{|X|}$ in $O(|X|)$ time by only initializing $K^0[y]$, $L_y^0[1], \ldots, L_y^0[|X|]$ and $J_y^0[1], \ldots, J_y^0[|X|]$. On the other hand, for any $1 \leqslant x \leqslant |X|$ and $1 \leqslant y \leqslant |Y|$, it follows from Lemmas 2 and 3 and the definitions of $I_y[x]$ and $J_y^x[k]$ that $H_y^x$ can be obtained from $H_y^{x-1}$ by only replacing $K^{x-1}[y]$, $K_{y-1}[x]$, $L_y^{x-1}[K_y[x]]$, $I_{y-1}[x]$ and $J_y^{x-1}[K_y[x]]$ with $K^x[y]$, $K_y[x]$, $L_y^x[K_y[x]]$, $I_y[x]$ and $J_y^x[K_y[x]]$, respectively. Note that all the five values, $K^x[y]$, $K_y[x]$, $L_y^x[K_y[x]]$, $I_y[x]$ and $J_y^x[K_y[x]]$, can be computed only from $X[x]$, $Y[y]$ and $H_y^{x-1}$. Since it follows from the definition that $K^{x-1}[y] \leqslant K^x[y]$, $K^y[x]$ can be found in $O(K^x[y] - K^{x-1}[y] + 1)$ time by repeatedly increasing $k$ from $K^{x-1}[y]$ until $X[x]$ becomes less than or equal to $L_y^{x-1}[k]$. After $K^x[y]$ is found, the other four values can be computed in constant time. Therefore, $H_y^x$ can be obtained from $H_y^{x-1}$ in $O(K^x[y] - K^{x-1}[y] + 1)$ time, and hence, for each $1 \leqslant y \leqslant |Y|$, $H_y^{|X|}$ can be obtained from $H_y^0$ in $O(|X|)$ time because $K^{|X|}[y] \leqslant |X|$. Thus, successively updating $H_y^x$ in order of $H_1^0, H_1^1, \ldots, H_1^{|X|}, H_2^0, H_2^1, \ldots, H_{|Y|}^{|X|}$, we can finally obtain $H_{|Y|}^{|X|}$ in $O(|X| \cdot |Y|)$ time. □

From Lemmas 1 and 5, we immediately have the following theorem.

**Theorem 1.** *For any integer sequences, $X$ and $Y$, and any integers, $l$ and $u$, one of the longest $(X, Y, l, u)$-CISs can be computed in* $O(|X| \cdot |Y|)$ *time and* $O(|X| + |Y|)$ *space.*

## References


[1] G.S. Brodal, K. Kaligosi, I. Katriel, M. Kutz, Faster algorithms for computing longest common increasing subsequences, BRICS RS-05-37, Department of Computer Science, University of Aarhus, Ny Munkegade, building 540, DK-8000 Aarhus C, Denmark, December 14, 2005.

[2] S. Bespamyatnikh, M. Segal, Enumerating longest increasing subsequences and patience sorting, Inform. Process. Lett. 76 (2000) 7–11.

[3] W.T. Chan, Y. Zhang, S.P.Y. Fung, D. Ye, H. Zhu, Efficient algorithms for finding a longest common increasing subsequence, in: Lecture Notes in Comput. Sci., vol. 3827, Springer, Berlin, 2005, pp. 655–674.

[4] A.L. Delcher, S. Kasif, R.D. Fleischmann, J. Peterson, O. White, S.L. Salzberg, Alignment of whole genomes, Nucl. Acids Res. 27 (1999) 2376–2396.

[5] J.Y. Guo, F.K. Hwang, An almost-linear time and linear space algorithm for the longest common subsequence problem, Inform. Process. Lett. 94 (2005) 131–135.

[6] D.S. Hirschberg, A linear space algorithm for computing maximal common subsequences, Comm. ACM 18 (6) (1975) 341–343.

[7] W.J. Masek, M.S. Paterson, A fast algorithm computing string edit distance, J. Comput. System Sci. 20 (1) (1980) 18–31.

[8] C. Schensted, Longest increasing and decreasing subsequences, Canad. J. Math. 12 (1961) 179–191.

[9] I.H. Yang, C.P. Huang, K.M. Chao, A fast algorithm for computing a longest common increasing subsequence, Inform. Process. Lett. 93 (2005) 249–253.