# Vertical and Horizontal Distances to Approximate Edit Distance for Rooted Labeled Caterpillars

Kohei Muraka, Takuya Yoshino and Kouichi Hirata
*Kyushu Institute of Technology, Kawazu 680-4, Iizuka 820-8502, Japan*

Keywords: Edit Distance, Rooted Labeled Caterpillar, Vertical Distance, Horizontal Distance, String Edit Distance, Multiset Edit Distance.

Abstract: A *rooted labeled caterpillar* (*caterpillar*, for short) is a rooted labeled tree transformed to a rooted path (called a *backbone*) after removing all the leaves in it and we can compute the *edit distance* between caterpillars in quartic time. In this paper, we introduce two *vertical distances* and two *horizontal distances* for caterpillars. The former are based on a *string edit distance* between the string representations of the backbones and the latter on a *multiset edit distance* between the multisets of labels occurring in all the leaves. Then, we show that these distances give both lower bound and upper bound of the edit distance and we can compute the vertical distances in quadratic time and the horizontal distances in linear time under the unit cost function.

## 1 INTRODUCTION

Comparing tree-structured data such as HTML and XML data for web mining or RNA and glycan data for bioinformatics is one of the important tasks for data mining. The most famous distance measure between *rooted labeled unordered trees* (*trees*, for short) is the *edit distance* (Tai, 1979). The edit distance is formulated as the minimum cost of *edit operations*, consisting of a *substitution*, a *deletion* and an *insertion*, applied to transform a tree to another tree. Unfortunately, the problem of computing the edit distance between trees is MAX SNP-hard (Zhang and Jiang, 1994), even if trees are binary or height 2 (Akutsu et al., 2013; Hirata et al., 2011).

A *caterpillar* (*cf.* (Gallian, 2007)) is a tree transformed to a rooted path after removing all the leaves in it. Recently, Muraka *et al.* (Muraka et al., 2018) have shown that we can compute the edit distance between caterpillars in $O(h^2\lambda^2)$ time, where $h$ is the maximum height and $\lambda$ is the maximum number of leaves in caterpillars. Hence, the problem is quartic-time tractable with respect to the maximum number of nodes, which is not efficient well.

As an efficient distance comparing caterpillars, histogram distances such as a *path histogram distance* (Kawaguchi et al., 2018), a *complete subtree histogram distance* (Akutsu et al., 2013; Yoshino et al., 2018) and an *LCA histogram distance* (Yoshino et al., 2018) have developed. Whereas these distances are metrics for caterpillars and we can compute them more efficiently (linear or quadratic time) than the edit distance (quartic time), they are incomparable with the edit distance in both theoretical and experimental.

In order to approximate the edit distance for caterpillars efficiently, in this paper, we introduce two *vertical distances* $d_V$ and $d_V^*$ based on a *string edit distance* and two *horizontal distances* $d_H$ and $d_H^*$ based on a *multiset edit distance*. Here, the multiset edit distance coincides with a famous *bag distance* (Deza and Deza, 2016) if we adopt a unit cost function.

Let $C_1$ and $C_2$ be caterpillars. Then, $d_V(C_1,C_2)$ is the string edit distance between the string representations of the backbones of $C_1$ and $C_2$, and $d_V^*(C_1,C_2)$ is the sum of $d_V(C_1,C_2)$, the multiset edit distance between the multisets on labels occurring in the leaves of the endpoints of the backbones in $C_1$ and $C_2$ and the costs of deleting the remained leaves in $C_1$ and inserting the remained leaves in $C_2$. Also $d_H(C_1,C_2)$ is the multiset edit distance between the multisets of labels occurring in all the leaves of $C_1$ and $C_2$, and $d_H^*(C_1,C_2)$ is the sum of $d_H(C_1,C_2)$, the cost of the correspondence between the roots of $C_1$ and $C_2$ and the costs of deleting nodes in the backbone in $C_1$ and inserting nodes in the backbone in $C_2$.

Then, we show that these distances provide the following lower bound and upper bound of the edit distance $\tau_{\text{TAI}}(C_1,C_2)$ between $C_1$ and $C_2$.

$$\max\{d_V(C_1,C_2),d_H(C_1,C_2)\} \\ \leq \tau_{\text{TAI}}(C_1,C_2) \leq \min\{d_V^*(C_1,C_2),d_H^*(C_1,C_2)\}.$$

Muraka, K., Yoshino, T. and Hirata, K.
Vertical and Horizontal Distances to Approximate Edit Distance for Rooted Labeled Caterpillars.
DOI: 10.5220/0007387205900597
In *Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods (ICPRAM 2019)*, pages 590-597
ISBN: 978-989-758-351-3

Furthermore, if we adopt the unit cost function, then we can compute $d_V(C_1,C_2)$, $d^*_V(C_1,C_2)$, $d_H(C_1,C_2)$ and $d^*_H(C_1,C_2)$ in $O(h^2)$ time, $O(h^2+\lambda)$ time, $O(\lambda)$ time and $O(\lambda+h)$ time, respectively. Hence, we can compute the vertical distances in quadratic time and the horizontal distances in linear time with respect to the number of nodes.

Finally, we give experimental results to evaluate the running time and the approximation for caterpillars in real data.

## 2 PRELIMINARIES

A *tree* $T$ is a connected graph $(V,E)$ without cycles, where $V$ is the set of vertices and $E$ is the set of edges. We denote $V$ and $E$ by $V(T)$ and $E(T)$. The *size* of $T$ is $|V|$ and denoted by $|T|$. We sometime denote $v \in V(T)$ by $v \in T$. We denote an empty tree $(\emptyset,\emptyset)$ by $\emptyset$. A *rooted tree* is a tree with one node $r$ chosen as its *root*. We denote the root of a rooted tree $T$ by $r(T)$.

Let $T$ be a rooted tree such that $r = r(T)$ and $u,v,w \in T$. We denote the unique path from $r$ to $v$, that is, the tree $(V',E')$ such that $V' = \{v_1,\ldots,v_k\}$, $v_1 = r$, $v_k = v$ and $(v_i,v_{i+1}) \in E'$ for every $i$ $(1 \leq i \leq k-1)$, by $UP_r(v)$.

The *parent* of $v(\neq r)$, which we denote by $par(v)$, is its adjacent node on $UP_r(v)$ and the *ancestors* of $v(\neq r)$ are the nodes on $UP_r(v)-\{v\}$. We say that $u$ is a *child* of $v$ if $v$ is the parent of $u$ and $u$ is a *descendant* of $v$ if $v$ is an ancestor of $u$. We denote the set of children of $v$ by $ch(v)$ and that $v$ is a ancestor of $u$ by $u \leq v$. We call a node with no children a *leaf* and denote the set of all the leaves in $T$ by $lv(T)$.

A *rooted path* $P$ is a rooted tree $(\{v_1,\ldots,v_n\},\{(v_i,v_{i+1}) \mid 1 \leq i \leq n-1\})$ such that $r(P) = v_1$. We call the node $v_n$ (the leaf of $P$) an *endpoint* of $P$ and denote it by $e(P)$.

The *degree* of $v$, denoted by $d(v)$, is the number of children of $v$, and the *degree* of $T$, denoted by $d(T)$, is $\max\{d(v) \mid v \in T\}$. The *height* of $v$, denoted by $h(v)$, is $\max\{|UP_v(w)| \mid w \in lv(T[v])\}$, and the *height* of $T$, denoted by $h(T)$, is $\max\{h(v) \mid v \in T\}$.

We say that $u$ is *to the left of* $v$ in $T$ if $pre(u) \leq pre(v)$ for the preorder number $pre$ in $T$ and $post(u) \leq post(v)$ for the postorder number $post$ in $T$. We say that a rooted tree is *ordered* if a left-to-right order among siblings is given; *unordered* otherwise. We say that a rooted tree is *labeled* if each node is assigned a symbol from a fixed finite alphabet $\Sigma$. For a node $v$, we denote the label of $v$ by $l(v)$, and sometimes identify $v$ with $l(v)$. In this paper, we call a rooted labeled unordered tree a *tree* simply.

**Definition 1** (Caterpillar (*cf.*, (Gallian, 2007))). We say that a tree is a *caterpillar* if it is transformed to a rooted path after removing all the leaves in it. For a caterpillar $C$, we call the remained rooted path a *backbone* of $C$ and denote it by $bb(C)$.

It is obvious that $r(C) = r(bb(C))$ and $V(C) = bb(C) \cup lv(C)$ for a caterpillar $C$, that is, every node in a caterpillar is either a leaf or an element of the backbone.

Next, we introduce a *tree edit distance* and a *Tai mapping*.

**Definition 2** (Edit operations (Tai, 1979)). The *edit operations* of a tree $T$ are defined as follows, see Figure 1.

1. *Substitution*: Change the label of the node $v$ in $T$.
2. *Deletion*: Delete a node $v$ in $T$ with parent $v'$, making the children of $v$ become the children of $v'$. The children are inserted in the place of $v$ as a subset of the children of $v'$. In particular, if $v$ is the root in $T$, then the result applying the deletion is a forest consisting of the children of the root.
3. *Insertion*: The complement of deletion. Insert a node $v$ as a child of $v'$ in $T$ making $v$ the parent of a subset of the children of $v'$.

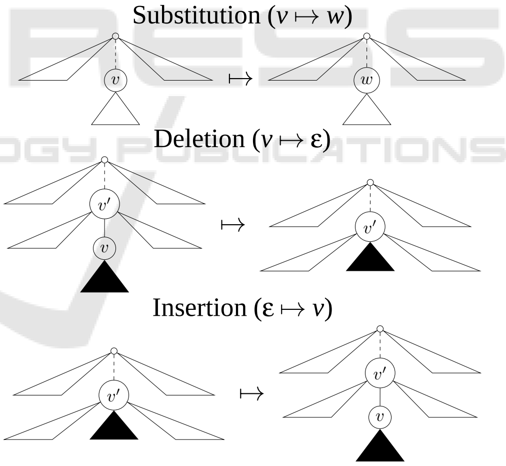

Substitution ($v \mapsto w$)
$v$
$\mapsto$
$w$
Deletion ($v \mapsto \varepsilon$)
$v'$
$v$
$\mapsto$
$v'$
Insertion ($\varepsilon \mapsto v$)
$v'$
$\mapsto$
$v'$
$v$

Figure 1: Edit operations for trees.

Let $\varepsilon \notin \Sigma$ denote a special *blank* symbol and define $\Sigma_\varepsilon = \Sigma \cup \{\varepsilon\}$. Then, we represent each edit operation by $(l_1 \mapsto l_2)$, where $(l_1,l_2) \in (\Sigma_\varepsilon \times \Sigma_\varepsilon - \{(\varepsilon,\varepsilon)\})$. The operation is a substitution if $l_1 \neq \varepsilon$ and $l_2 \neq \varepsilon$, a deletion if $l_2 = \varepsilon$, and an insertion if $l_1 = \varepsilon$. For nodes $v$ and $w$, we also denote $(l(v) \mapsto l(w))$ by $(v \mapsto w)$. We define a *cost function* $\gamma : (\Sigma_\varepsilon \times \Sigma_\varepsilon \setminus \{(\varepsilon,\varepsilon)\}) \mapsto \mathbf{R}^+$ on pairs of labels. We often constrain a cost function $\gamma$ to be a *metric*, that is, $\gamma(l_1,l_2) \geq 0$, $\gamma(l_1,l_2) = 0$ iff $l_1 = l_2$, $\gamma(l_1,l_2) = \gamma(l_2,l_1)$ and $\gamma(l_1,l_3) \leq \gamma(l_1,l_2)+\gamma(l_2,l_3)$. In particular, we call the cost function that $\gamma(l_1,l_2) = 1$ if $l_1 \neq l_2$ a *unit cost function*.

**Definition 3** (Edit distance (Tai, 1979))**.** For a cost function $\gamma$, the *cost* of an edit operation $e = l_1 \mapsto l_2$ is given by $\gamma(e) = \gamma(l_1, l_2)$. The *cost* of a sequence $E = e_1, \ldots, e_k$ of edit operations is given by $\gamma(E) = \sum_{i=1}^{k} \gamma(e_i)$. Then, an *edit distance* $\tau_{\text{TAI}}(T_1, T_2)$ between trees $T_1$ and $T_2$ is defined as follows:

$$\tau_{\text{TAI}}(T_1, T_2) = \min \left\{ \gamma(E) \;\middle|\; \begin{array}{l} E \text{ is a sequence} \\ \text{of edit operations} \\ \text{transforming } T_1 \text{ to } T_2 \end{array} \right\}.$$

**Definition 4** (Tai mapping (Tai, 1979))**.** Let $T_1$ and $T_2$ be trees. We say that a triple $(M, T_1, T_2)$ is a *Tai mapping* (a *mapping*, for short) from $T_1$ to $T_2$ if $M \subseteq V(T_1) \times V(T_2)$ and every pair $(v_1, w_1)$ and $(v_2, w_2)$ in $M$ satisfies the following conditions.

1. $v_1 = v_2$ iff $w_1 = w_2$ (one-to-one condition).
2. $v_1 \leq v_2$ iff $w_1 \leq w_2$ (ancestor condition).

We will use $M$ instead of $(M, T_1, T_2)$ when there is no confusion denote it by $M \in \mathcal{M}_{\text{TAI}}(T_1, T_2)$.

Let $M$ be a mapping from $T_1$ to $T_2$. Let $I_M$ and $J_M$ be the sets of nodes in $T_1$ and $T_2$ but not in $M$, that is, $I_M = \{v \in T_1 \mid (v, w) \notin M\}$ and $J_M = \{w \in T_2 \mid (v, w) \notin M\}$. Then, the *cost* $\gamma(M)$ of $M$ is given as follows.

$$\gamma(M) = \sum_{(v,w) \in M} \gamma(v, w) + \sum_{v \in I_M} \gamma(v, \varepsilon) + \sum_{w \in J_M} \gamma(\varepsilon, w).$$

**Theorem 1** (Tai, 1979)**.** $\tau_{\text{TAI}}(T_1, T_2) = \min\{\gamma(M) \mid M \in \mathcal{M}_{\text{TAI}}(T_1, T_2)\}$.

For computing the edit distance between trees, the following theorem is well-known.

**Theorem 2** (Akutsu et al., 2013; Hirata et al., 2011; Zhang and Jiang, 1994)**.** *Let $T_1$ and $T_2$ be trees. Then, the problem of computing $\tau_{\text{TAI}}(T_1, T_2)$ is* MAX SNP-*hard, even if both $T_1$ and $T_2$ are binary or height* 2.

On the other hand, Muraka *et al.* (Muraka et al., 2018) have recently shown the following theorem.

**Theorem 3** (Muraka et al., 2018)**.** *Let $C_1$ and $C_2$ be caterpillars, where $h = \max\{h(C_1), h(C_2)\}$ and $\lambda = \max\{|lv(C_1)|, |lv(C_2)|\}$. Then, we can compute $\tau_{\text{TAI}}(C_1, C_2)$ in $O(h^2\lambda^2)$ time.*

Finally, we introduce the notions of multisets. A *multiset* on $\Sigma$ is a mapping $S : \Sigma \to \mathbf{N}$. For a multiset $S$ on $\Sigma$, we say that $a \in \Sigma$ is an *element* of $S$ if $S(a) > 0$ and denote it by $a \in S$ (like as a standard set). The *cardinality* of $S$, denoted by $|S|$, is defined as $\sum_{a \in \Sigma} S(a)$.

Let $S_1$ and $S_2$ be multisets on $\Sigma$. Then, we define the *intersection* $S_1 \sqcap S_2$ and the *difference* $S_1 \setminus S_2$ are multisets satisfying that $(S_1 \sqcap S_2)(a) = \min\{S_1(a), S_2(a)\}$ and $(S_1 \setminus S_2)(a) = \max\{S_1(a) - S_2(a), 0\}$ for every $a \in \Sigma$. Note that $S_1 \setminus S_2 = S_1 \setminus S_1 \sqcap S_2$ and $|S_1 \setminus S_2| = |S_1 \setminus S_1 \sqcap S_2| = |S_1| - |S_1 \sqcap S_2|$.

# 3 VERTICAL AND HORIZONTAL DISTANCES FOR CATERPILLARS

Theorem 3 claims that the problem of computing $\tau_{\text{TAI}}(C_1, C_2)$ for caterpillars $C_1$ and $C_2$ is tractable in quartic time, which is not efficient well. In this section, we give simple and efficient approximation of $\tau_{\text{TAI}}(C_1, C_2)$ by using *vertical* and *horizontal distances*, respectively.

The vertical distance is based on a *string edit distance* (*cf.*, (Deza and Deza, 2016)) for the string representation of the backbones. For strings $s_1$ and $s_2$, we denote the string edit distance between $s_1$ and $s_2$ by $\sigma(s_1, s_2)$. For a rooted path $P = (\{v_1, \ldots, v_n\}, \{(v_i, v_{i+1}) \mid 1 \leq i \leq n-1\})$ such that $r(P) = v_1$, we define the *string representation* of $P$ as a string $l(v_1) \cdots l(v_n)$ and denote it by $s(P)$.

On the other hand, the horizontal distance is based on a *multiset edit distance*, which is defined as similar as another edit distance (*cf.*, Definition 3).

The *edit operations* of a multiset $S$ on $\Sigma$ are defined as those of a tree. Let $a, b \in \Sigma$ such that $S(a) > 0$ and $a \neq b$. Then, a *substitution* $(a \mapsto b)$ operates $S(a)$ to $S(a) - 1$ and $S(b)$ to $S(b) + 1$, a *deletion* $(a \mapsto \varepsilon)$ operates $S(a)$ to $S(a) - 1$ and an *insertion* $(\varepsilon \mapsto b)$ operates $S(b)$ to $S(b) + 1$. Also we assume a cost function $\gamma$ as in Section 2.

**Definition 5** (Multiset edit distance)**.** Let $S_1$ and $S_2$ be multisets on $\Sigma$ and $\gamma$ a cost function. Then, a *multiset edit distance* $\mu(S_1, S_2)$ between $S_1$ and $S_2$ is defined as follows.

$$\mu(S_1, S_2) = \min \left\{ \gamma(E) \;\middle|\; \begin{array}{l} E \text{ is a sequence} \\ \text{of edit operations} \\ \text{transforming } S_1 \text{ to } S_2 \end{array} \right\}.$$

For multisets $S_1$ and $S_2$ such that $|S_1| \leq |S_2|$ (*resp.*, $|S_1| > |S_2|$), we can consider an injection $\pi$ from $S_1$ to $S_2$ (*resp.*, from $S_2$ to $S_1$). For example, let $S_1$ and $S_2$ be multisets such that $S_1(a) = 3$, $S_1(b) = 0$, $S_2(a) = 2$ and $S_2(b) = 2$. Then, by regarding $S_1$ and $S_2$ as the sequences $[a^{(1)}, a^{(2)}, a^{(3)}]$ and $[a^{(1)}, a^{(2)}, b^{(1)}, b^{(2)}]$ (where the superscript denotes the order of the element), the function $\pi$ such that $\pi(a^{(1)}) = a^{(2)}$, $\pi(a^{(2)}) = b^{(2)}$ and $\pi(a^{(3)}) = a^{(1)}$ is an injection from $S_1$ to $S_2$. When $|S_1| \leq |S_2|$ (*resp.*, $|S_1| > |S_2|$), we denote the set of all the injections from $S_1$ to $S_2$ (*resp.*, from $S_2$ to $S_1$) by $\Pi_1$ (*resp.*, $\Pi_2$).

**Lemma 1.** *The following equation holds.*

$$\mu(S_1,S_2) = \begin{cases} \min\limits_{\pi\in\Pi_1}\left\{\sum\limits_{a\in S_1}\gamma(a,\pi(a)) + \sum\limits_{b\in S_2\setminus\pi(S_1)}\gamma(\varepsilon,b)\right\}, \\ \qquad \textit{if } |S_1|\le|S_2|, \\ \min\limits_{\pi\in\Pi_2}\left\{\sum\limits_{b\in S_2}\gamma(\pi(b),b) + \sum\limits_{a\in S_1\setminus\pi(S_2)}\gamma(a,\varepsilon)\right\}, \\ \qquad \textit{otherwise.} \end{cases}$$

*Proof.* Suppose that $|S_1|\le|S_2|$. By the minimality of Definition 5, an injection $\pi\in\Pi_1$ maps $a\in S_1$ to the same $a\in S_2$ as possible, that is, $\pi(a)=a$ with the cost $\gamma(a,\pi(a))=0$, and the remained $c\in S_1$ to $\pi(c)\in S_2$ with the cost $\gamma(c,\pi(c))$. Then, the sum of the costs is represented by $\sum_{a\in S_1}\gamma(a,\pi(a))$. Furthermore, every $b\in S_2\setminus\pi(S_1)$ is inserted with the cost $\sum_{b\in S_2\setminus\pi(S_1)}\gamma(\varepsilon,b)$. Hence, the total cost implies the first formula.

Suppose that $|S_1|>|S_2|$. By the minimality of Definition 5, an injection $\pi\in\Pi_2$ maps $b\in S_2$ to the same $b\in S_1$ as possible, that is, $\pi(b)=b$ with the cost $\gamma(\pi(b),b)=0$, and the remained $c\in S_2$ to $\pi(c)\in S_1$ with the cost $\gamma(\pi(c),c)$. Then, the sum of the costs is represented by $\sum_{b\in S_2}\gamma(\pi(b),b)$. Furthermore, every $a\in S_1\setminus\pi(S_2)$ is deleted with the cost $\sum_{a\in S_1\setminus\pi(S_2)}\gamma(a,\varepsilon)$. Hence, the total cost implies the second formula. □

If we adopt a unit cost function, then we can give the following simpler form of Lemma 1 which coincides with a *bag distance* (Deza and Deza, 2016) between multisets.

**Lemma 2.** *If $\gamma$ is a unit cost function, then the following statement holds.*

$$\mu(S_1,S_2) = \max\{|S_1\setminus S_2|,|S_2\setminus S_1|\}.$$

*Proof.* Suppose that $|S_1|\le|S_2|$. Then, by Lemma 1, it holds that:

$$\begin{aligned}&\sum_{a\in S_1}\gamma(a,\pi(a))\\ &= \underbrace{\sum_{a\in S_1\sqcap S_2}\gamma(a,a)}_{=0} + \sum_{a\in S_1\setminus S_1\sqcap S_2, b\in S_2\setminus S_1\sqcap S_2, a\neq b}\gamma(a,b)\\ &= |S_1\setminus S_1\sqcap S_2| = |S_1|-|S_1\sqcap S_2|.\end{aligned}$$

On the other hand, since $\pi$ is an injection, it holds that $\sum_{b\in S_2\setminus\pi(S_1)}\gamma(\varepsilon,b)=|S_2\setminus\pi(S_1)|=|S_2|-|S_1|$. As a result, it holds that $\mu(S_1,S_2)=|S_1|-|S_1\sqcap S_2|+|S_2|-|S_1|=|S_2|-|S_1\sqcap S_2|=|S_2\setminus S_1|$.

Furthermore, in this case, by the supposition that $|S_1|\le|S_2|$ and since $|S_2\setminus S_1|=|S_2\setminus S_1\sqcap S_2|=|S_2|-|S_1\sqcap S_2|$ and $|S_1\setminus S_2|=|S_1\setminus S_1\sqcap S_2|=|S_1|-|S_1\sqcap S_2|$, it holds that $|S_2\setminus S_1|\ge|S_1\setminus S_2|$. Hence, $|S_2\setminus S_1|=\max\{|S_1\setminus S_2|,|S_2\setminus S_1|\}$.

By using the same discussion, if $|S_1|>|S_2|$, then $\mu(S_1,S_2)=|S_1\setminus S_2|=\max\{|S_1\setminus S_2|,|S_2\setminus S_1|\}$. □

**Lemma 3.** *We can compute $\mu(S_1,S_2)$ in $O(m^2M)$ time, where $m=\min\{|S_1|,|S_2|\}$ and $M=\max\{|S_1|,|S_2|\}$. Furthermore, if we adopt the unit cost function, then we can compute $\mu(S_1,S_2)$ in $O(m+M)$ time.*

*Proof.* By Lemma 1 and by using the same technique based on the maximum weighted bipartite matching algorithm for the complete bipartite graph consisting of $S_1$ and $S_2$ (*cf.*, (Yamamoto et al., 2014; Zhang et al., 1996)), we can compute $\mu(S_1,S_2)$ in $O(m^2M)$ time. On the other hand, by Lemma 2, we can compute $\mu(S_1,S_2)$ in $O(m+M)$ time. □

Hence, we formulate vertical and horizontal distances between caterpillars. Here, we regard a set $L$ of leaves as a multiset of labels on $\Sigma$ occurring in $L$, which we denote by $\widetilde{L}$.

**Definition 6** (Vertical and horizontal distances)**.** For $i=1,2$, let $C_i$ be a caterpillar such that $r_i=r(C_i)$, $B_i=bb(C_i)$, $L_i=lv(C_i)$ and $E_i=ch(e(B_i))$. Then, we define two *vertical distances* $d_V$ and $d_V^*$ as follows.

$$\begin{aligned}d_V(C_1,C_2) &= \sigma(s(B_1),s(B_2)).\\ d_V^*(C_1,C_2) &= d_V(C_1,C_2)+\mu(\widetilde{E_1},\widetilde{E_2})\\ &\quad + \sum_{v\in L_1\setminus E_1}\gamma(v,\varepsilon) + \sum_{w\in L_2\setminus E_2}\gamma(\varepsilon,w).\end{aligned}$$

Also we define two *horizontal distances* $d_H$ and $d_H^*$ as follows.

$$\begin{aligned}d_H(C_1,C_2) &= \mu(\widetilde{L_1},\widetilde{L_2}).\\ d_H^*(C_1,C_2) &= d_H(C_1,C_2)+\gamma(r_1,r_2)\\ &\quad + \sum_{v\in B_1\setminus\{r_1\}}\gamma(v,\varepsilon) + \sum_{w\in B_2\setminus\{r_2\}}\gamma(\varepsilon,w).\end{aligned}$$

**Theorem 4.** *Let $C_1$ and $C_2$ be caterpillars. Then, the following statement holds.*

$$\begin{aligned}&\max\{d_V(C_1,C_2),d_H(C_1,C_2)\}\\ &\quad\le \tau_{\mathrm{TAI}}(C_1,C_2) \le \min\{d_V^*(C_1,C_2),d_H^*(C_1,C_2)\}.\end{aligned}$$

*Proof.* In order to show the left inequality, it is sufficient to show how the values of $d_V(C_1,C_2)$ and $d_H(C_1,C_2)$ change when $C_2$ is obtained by applying one edit operation to $C_1$.

If $C_2$ is obtained by substituting to an element in $bb(C_1)$, then it holds that $d_V(C_1,C_2)=1$ and $d_H(C_1,C_2)=0$. If $C_2$ is obtained by substituting to a leaf in $lv(C_1)$, then it holds that $d_V(C_1,C_2)=0$ and $d_H(C_1,C_2)=1$. If $C_2$ is obtained by deleting an element in $bb(C_1)$, then it holds that $d_V(C_1,C_2)=1$ and $d_H(C_1,C_2)=0$. If $C_2$ is obtained by deleting a leaf in $lv(C_1)$, then it holds that $d_V(C_1,C_2) = 0$ and $d_H(C_1,C_2) = 1$.

As a result, if $C_2$ is obtained by applying one edit operation to $C_1$, then both values of $d_V(C_1,C_2)$ and $d_H(C_1,C_2)$ change at most one. Hence, it holds that $d_V(C_1,C_2) \leq \tau_{\mathrm{TAI}}(C_1,C_2)$ and $d_H(C_1,C_2) \leq \tau_{\mathrm{TAI}}(C_1,C_2)$, which implies the left inequality.

On the other hand, it order to show the right inequality, by regarding the correspondences between $B_1$ and $B_2$ in $\sigma(s(B_1),s(B_2))$ and those between $L_1$ and $L_2$ in $\mu(\widetilde{L_1},\widetilde{L_2})$ as the pairs of $V(C_1) \times V(C_2)$, the set of correspondences between nodes in $d_V(C_1,C_2)$ and $d_H(C_1,C_2)$ form Tai mappings. Then, it is obvious that all the correspondences in $d^*_V(C_1,C_2)$ and $d^*_H(C_1,C_2)$ are one-to-one.

Since the correspondences in $d_V(C_1,C_2)$ preserve ancestor relation and every node in $E_i$ is a descendant of the node in $e(B_i)$ $(i = 1,2)$, all the correspondences in $d^*_V(C_1,C_2)$ preserve ancestor relation. Also, since every leaf in $L_i$ is an descendant of the root $r_i$ in $C_i$ $(i = 1,2)$, all the correspondences in $d^*_H(C_1,C_2)$ preserve ancestor relation.

As a result, all the correspondences in $d^*_V(C_1,C_2)$ and $d^*_H(C_1,C_2)$ form Tai mappings between $C_1$ and $C_2$, respectively, which implies that $\tau_{\mathrm{TAI}}(C_1,C_2) \leq d^*_V(C_1,C_2)$ and $\tau_{\mathrm{TAI}}(C_1,C_2) \leq d^*_H(C_1,C_2)$ by Theorem 1. Hence, the right inequality holds. □

**Theorem 5.** *Let $C_1$ and $C_2$ be caterpillars, where $h = \max\{h(C_1),h(C_2)\}$ and $\lambda = \max\{|lv(C_1)|,|lv(C_2)|\}$. Then, we can compute $d_V(C_1,C_2)$, $d^*_V(C_1,C_2)$, $d_H(C_1,C_2)$ and $d^*_H(C_1,C_2)$ in $O(h^2)$ time, $O(h^2+\lambda^3)$ time, $O(\lambda^3)$ time and $O(\lambda^3+h)$ time, respectively. Furthermore, if we adopt the unit cost function, then we can compute $d_V(C_1,C_2)$, $d^*_V(C_1,C_2)$, $d_H(C_1,C_2)$ and $d^*_H(C_1,C_2)$ in $O(h^2)$ time, $O(h^2+\lambda)$ time, $O(\lambda)$ time and $O(\lambda+h)$ time, respectively.*

*Proof.* It is obvious by Lemma 3 and since we can compute $\sigma(s(B_1),s(B_2))$ in $O(h^2)$ time (*cf.*, (Deza and Deza, 2016)). □

Hence, if we adopt the unit cost function, then we can compute the vertical distances of $d_V(C_1,C_2)$ and $d^*_V(C_1,C_2)$ in quadratic time and the horizontal distances of $d_H(C_1,C_2)$ and $d^*_H(C_1,C_2)$ in linear time.

## 4 EXPERIMENTAL RESULTS

In this section, we give experimental results to evaluate the inequality in Theorem 4 and the running time in Theorem 5 (under the unit cost function). Here, concerned with Theorem 4, we denote the lower bound distance $\max\{d_V,d_H\}$ of $\tau_{\mathrm{TAI}}$ by *lbd* and the upper bound distance $\min\{d^*_V,d^*_H\}$ of $\tau_{\mathrm{TAI}}$ by *ubd*. Also let $\mathit{diff} = \mathit{ubd} - \mathit{lbd}$.

In this paper, we use the real data illustrated from Table 1, which illustrates the number of caterpillars in N-glycans and all-glycans from KEGG[1], CSLOGS[2], dblp[3]. Here, #cat is the number of caterpillars and #data is the total number of data.

Table 1: The number of caterpillars in N-glycans and all-glycans from KEGG, CSLOGS and dblp.

| dataset | #cat | #data | % |
|---|---|---|---|
| N-glycans | 514 | 2,142 | 23.996 |
| all-glycans | 8,005 | 10,704 | 74.785 |
| CSLOGS | 41,592 | 59,691 | 69.679 |
| dblp | 5,154,295 | 5,154,530 | 99.995 |

We deal with caterpillars for N-glycans, all-glycans, CSLOGS and the largest 5,154 caterpillars (0.1%) in dblp (we refer to dblp$^-$). Table 2 illustrates the information of such caterpillars. Here, # is the number of caterpillars, *n* is the average number of nodes, *d* is the average degree, *h* is the average height, $\lambda$ is the average number of leaves and $\beta$ is the average number of labels.

Table 2: The information of caterpillars in N-glycans, all-glycans, CSLOGS and dblp$^-$.

| dataset | # | $n$ | $d$ | $h$ | $\lambda$ | $\beta$ |
|---|---|---|---|---|---|---|
| N-glycans | 514 | 6.40 | 1.84 | 4.22 | 2.18 | 4.50 |
| all-glycans | 8,005 | 4.74 | 1.49 | 3.02 | 1.72 | 2.84 |
| CSLOGS | 41,592 | 5.84 | 3.05 | 2.20 | 3.64 | 5.18 |
| dblp$^-$ | 5,154 | 41.74 | 40.73 | 1.01 | 40.73 | 10.62 |

First, Table 3 illustrates the running time to compute the vertical distances $d_V$ and $d^*_V$, the horizontal distances $d_H$ and $d^*_H$ and the edit distance $\tau_{\mathrm{TAI}}$ (Muraka et al., 2018) for all the pairs of caterpillars in Table 2.

Table 3: The running time of computing distances $d_V$, $d^*_V$, $d_H$, $d^*_H$ and $\tau_{\mathrm{TAI}}$ (sec).

| dataset | $d_V$ | $d^*_V$ | $d_H$ | $d^*_H$ | $\tau_{\mathrm{TAI}}$ |
|---|---|---|---|---|---|
| N-glycans | 0.15 | 0.26 | 0.17 | 0.19 | 635.97 |
| all-glycans | 20.35 | 48.08 | 29.98 | 20.35 | 57,011.10 |
| CSLOGS | 336.72 | 1,821.36 | 1,564.28 | 1,788.53 | — |
| dblp$^-$ | 2.86 | 149.17 | 137.20 | 143.22 | 6,363.79 |

[1] Kyoto Encyclopedia of Genes and Genomes, http://www.kegg.jp/

[2] CSLOGS: http://www.cs.rpi.edu/∼zaki/www-new/pmwiki.php/Software/Software

[3] dblp computer science bibliography: http://dblp.uni-trier.de/

Table 3 shows that, as the experimental evaluation of Theorem 5 (and 3), the running time of computing all the distances of $d_V$, $d_V^*$, $d_H$ and $d_H^*$ is much smaller than that of the edit distance $\tau_{\rm TAI}$, and the running time of computing the horizontal distance $d_H^*$ is smaller than that of the vertical distance $d_V^*$.

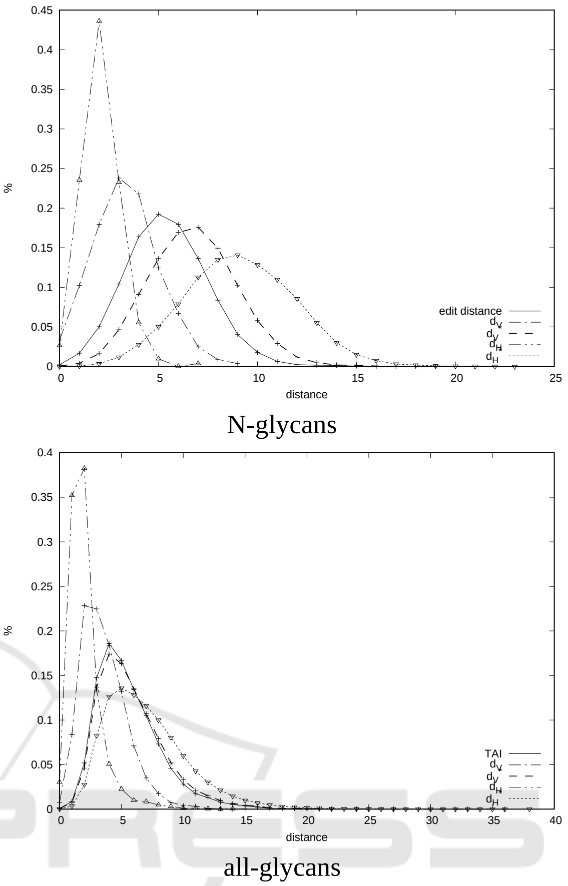
Note that, the reason why the running time of computing $d_V$ for dblp$^-$ is extremely small is that the height in every caterpillar in dblp$^-$ is either 1 or 2 and then the running time of $\sigma(s(B_1), s(B_2))$ is small. Also, the height of 88% in caterpillars for CSLOGS is from 1 to 3, which is the reason why the running time of computing $d_V$ is smaller than that of other distances for CSLOGS. Furthermore, in contrast to Theorem 5, the running time of computing $d_V$ and $d_V^*$ (in $O(h^2)$ and $O(h^2 + \lambda)$ time in theoretical) is not much larger than that of $d_H$ and $hd^*$ (in $O(\lambda)$ and $O(\lambda + h)$ time in theoretical), because we conjecture that the height in caterpillars for all the data is too small to influence the running time.

Next, we compare the distances of $d_V$, $d_V^*$, $d_H$, $d_H^*$ and $\tau_{\rm TAI}$. Figure 2 illustrates the distributions of the distances for N-glycans and all-glycans. Also Figure 3 and 4 illustrate the distributions of the distances to 10, from 10 to 30, from 30 to 100 and from 100, for CSLOGS and dblp$^-$, respectively. Since we cannot compute $\tau_{\rm TAI}$ for CSLOGS, Figure 3 presents the distances of $d_V$, $d_V^*$, $d_H$ and $d_H^*$. Since the vertical distance $d_V$ for more than 99% pairs of caterpillars in CSLOGS is 0 or 1, Figure 4 presents the distances of $d_V^*$, $d_H$, $d_H^*$ and $\tau_{\rm TAI}$

Figure 2 shows that the forms of all the distributions in are nearly normal, *lbd* is left to $\tau_{\rm TAI}$ and $\tau_{\rm TAI}$ is left to *ubd*. On the other hand, Figure 3 and 4 show that the forms of distributions are not normal, but concentrate small values. Figure 3 shows that more than 90% pairs of caterpillars for CSLOGS concentrate on the distances within 30, where the maximum values of $d_V$, $d_V^*$, $d_H$ and $d_H^*$ are 70, 579, 403 and 473, respectively. Also Figure 4 shows that more than 90% pairs of caterpillars for dblp$^-$ concentrate on the distances within 40, where the maximum values of $\tau_{\rm TAI}$. $d_V^*$, $d_H$ and $d_H^*$ are 746, 813, 745 and 746, respectively.

Figure 5 illustrates the scatter charts of *lbd*, *ubd* and $\tau_{\rm TAI}$ for N-glycans, all-glycans, CSLOGS and dblp$^-$. Here, the representation of $d_y/d_x$ means that the number of pairs of caterpillars with the distance $d_x$ is pointed at the $x$-axis and that with the distance $d_y$ at the $y$-axis.

Since the number of caterpillars in N-glycans is small, so the scatter charts in Figure 5 are sparse. For N-glycans and all-glycans, the difference between a pair of *ubd*, *lbd* and $\tau_{\rm TAI}$ is almost within 10. For CSLOGS and dblp$^-$, the difference is not large.



N-glycans

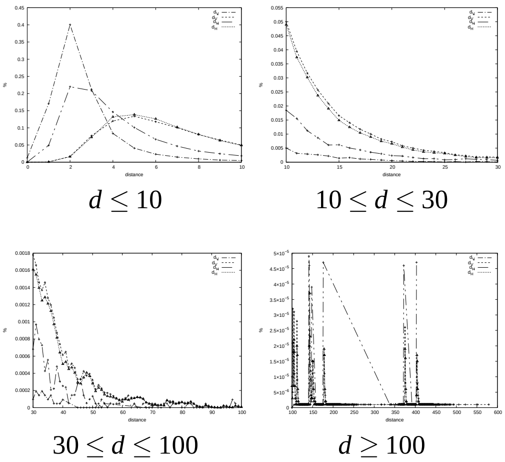
all-glycans

Figure 2: The distributions of distances for N-glycans and all-glycans.

In order to cofirm it in more detail, we evaluate how the lower bound distances and the upper bound distances approximate to the edit distance. Then, Table 4 illustrates the difference *diff* for N-glycans, all-glycans, dblp$^-$ and CSLOGS.

Table 4 shows that more than 93% of caterpillars for N-glycans satisfy that *diff* $\leq 5$, more than 94% of caterpillars for all-glycans satisfy that *diff* $\leq 4$, more than 99% of caterpillars for dblp$^-$ satisfy that *diff* $\leq 1$ and more than 92% of caterpillars for CSLOGS satisfy that *diff* $\leq 5$.

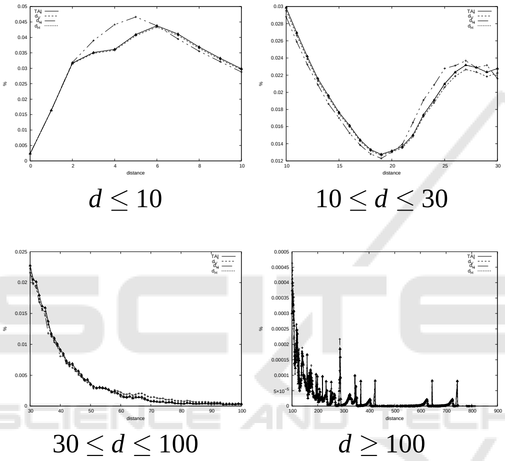
Hence, since more than 90% (*resp.*, 98%) of caterpillars satisfy that *diff* $\leq 5$ (*resp.*, *diff* $\leq 10$), we can conclude that $\max\{d_V, d_H\}$ and $\min\{d_V^*, d_H^*\}$ succeed to approximate $\tau_{\rm TAI}$ within 5 (*resp.*, 10). This result is important for the case that the running time of computing $\tau_{\rm TAI}$ is large as CSLOGS.



$d \leq 10$ $10 \leq d \leq 30$

$30 \leq d \leq 100$ $d \geq 100$

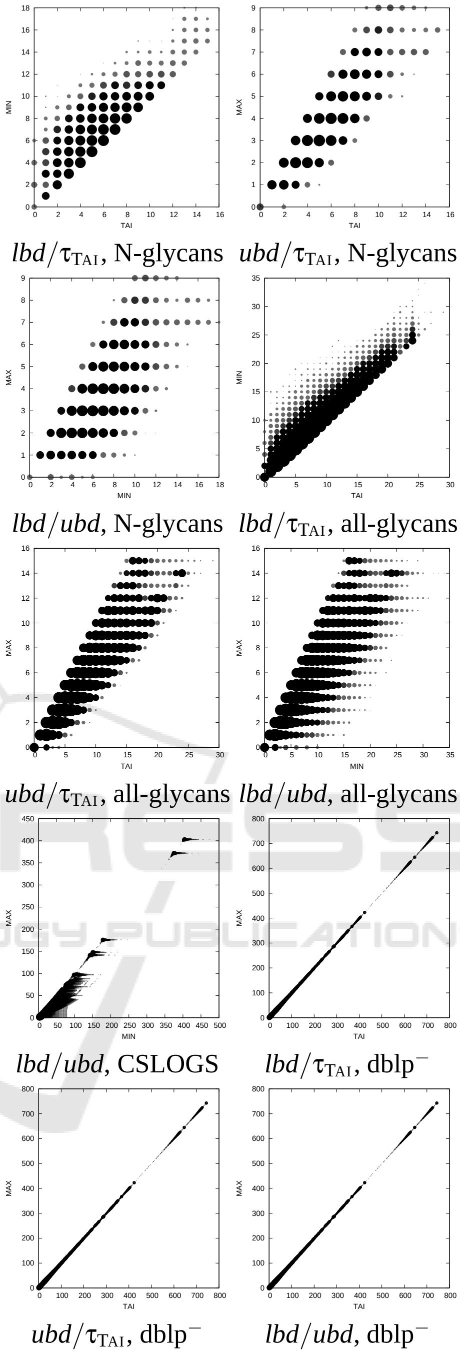
Figure 3: The distributions of distances for CSLOGS.



$d \leq 10$ $10 \leq d \leq 30$

$30 \leq d \leq 100$ $d \geq 100$

Figure 4: The distributions of distances for dblp$^-$.



$lbd/\tau_{\mathrm{TAI}}$, N-glycans $ubd/\tau_{\mathrm{TAI}}$, N-glycans

$lbd/ubd$, N-glycans $lbd/\tau_{\mathrm{TAI}}$, all-glycans

$ubd/\tau_{\mathrm{TAI}}$, all-glycans $lbd/ubd$, all-glycans

$lbd/ubd$, CSLOGS $lbd/\tau_{\mathrm{TAI}}$, dblp$^-$

$ubd/\tau_{\mathrm{TAI}}$, dblp$^-$ $lbd/ubd$, dblp$^-$

Figure 5: The scatter charts of of *lbd*, *ubd* and $\tau_{\mathrm{TAI}}$ for N-glycans, all-glycans, CSLOGS and dblp$^-$.

## 5 CONCLUSION

In this paper, we have formulated the *vertical distances* $d_V$ and $d_V^*$ and the *horizontal distances* $d_H$ and $d_H^*$ to approximate the edit distance $\tau_{\mathrm{TAI}}$. Then, we have shown the following inequality:

$$\max\{d_V, d_H\} \leq \tau_{\mathrm{TAI}} \leq \min\{d_V^*, d_H^*\}.$$

Furthermore, we have shown that, if we adopt the unit cost function, then we can compute $d_V$ and $d_V^*$ in quadratic time and $d_H$ and $d_H^*$ in linear time.

Finally, we have given the experimental results to evaluate the inequality and the running time for N-glycans, all-glycans, CSLOGS and dblp$^-$. Then, we can conclude that by combining $d_V$, $d_V^*$, $d_H$ and $d_H^*$, we can approximate to the edit distance well such that

$$\min\{d_V^*, d_H^*\} - \max\{d_V, d_H\} \leq 5$$

for more than 90% of caterpillars.

It is a future work to give experimental results for other data such as SwissProt, TPC-H, Auction, University, Protein and Nasa from UW XML Repository[4]. Note that, whereas the last four data contain no caterpillars, we can obtain many caterpillars by deleting the root (*cf.*, (Muraka et al., 2018)).

[4] UW XML Repository, http://aiweb.cs.washington.edu /research/projects/xmltk/xmldata/www/repository.html

Table 4: The difference *diff* for N-glycans, all-glycans, dblp$^-$ and CSLOGS.

N-glycans

| *diff* | # | % |
|---|---|---|
| 0 | 2,448 | 1.86 |
| 1 | 17,091 | 12.96 |
| 2 | 32,404 | 24.58 |
| 3 | 33,949 | 25.75 |
| 4 | 24,240 | 18.46 |
| 5 | 13,420 | 10.18 |
| 6 | 5,801 | 4.40 |
| 7 | 1,751 | 1.33 |
| 8 | 475 | 0.36 |
| 9 | 109 | 0.08 |
| 10 | 47 | 0.04 |
| 11 | 6 | 0.00 |

all-glycans

| *diff* | # | % |
|---|---|---|
| 0 | 1,105,515 | 3.47 |
| 1 | 11,619,644 | 34.46 |
| 2 | 10,547,139 | 33.10 |
| 3 | 4,633,275 | 14.54 |
| 4 | 2,108,501 | 6.62 |
| 5 | 1,001,311 | 3.14 |
| 6 | 458,637 | 1.44 |
| 7 | 203,334 | 0.64 |
| 8 | 110,184 | 0.35 |
| 9 | 49,385 | 0.16 |
| 10 | 20,461 | 0.06 |
| 11 | 6,999 | 0.02 |
| 12 | 2,393 | 0.01 |
| 13 | 801 | 0.00 |
| 14 | 350 | 0.00 |
| 15 | 147 | 0.00 |
| 16 | 30 | 0.00 |
| 17 | 18 | 0.00 |
| 18 | 8 | 0.00 |
| 19 | 3 | 0.00 |
| 20 | 1 | 0.00 |

dblp$^-$

| *diff* | # | % |
|---|---|---|
| 0 | 6,960,854 | 52.42 |
| 1 | 6,198,038 | 46.67 |
| 2 | 119,889 | 0.90 |
| 3 | 500 | 0.00 |

CSLOGS

| *diff* | # | % | *diff* | # | % |
|---|---|---|---|---|---|
| 0 | 10,513,132 | 1.22 | 8 | 8,791,664 | 1.02 |
| 1 | 174,777,470 | 20.21 | 9 | 5,472,715 | 0.63 |
| 2 | 301,960,142 | 34.91 | 10 | 3,612,677 | 0.42 |
| 3 | 175,761,327 | 20.32 | 11 | 2,667,528 | 0.31 |
| 4 | 90,141,737 | 10.42 | 12 | 2,046,998 | 0.24 |
| 5 | 42,955,474 | 4.97 | 13 | 1,567,370 | 0.18 |
| 6 | 23,342,365 | 2.70 | 14 | 1,247,637 | 0.14 |
| 7 | 14,094,693 | 1.63 | $\geq 15$ | 5,973,407 | 0.69 |

One of the reason that the approximation succeeds is that every node in a caterpillar is either an element of the backbone or a leaf, that is, $V(C) = bb(C) \cup lv(C)$. Also $d_V$ and $d_V^*$ are based on a string edit distance for $bb(C)$ and $d_H$ and $d_H^*$ are based on a multiset edit distance for $lv(C)$. When we can extend these distances to standard trees, it is necessary how to determine a backbone and to deal with internal nodes, which is a future work.

Concerned with the horizontal distances, we can consider the repetition of the bag distance between leaves after removing leaves from trees as possible. Then, it is a future work to analyze such a distance.

## ACKNOWLEDGMENTS

This work is partially supported by Grant-in-Aid for Scientific Research 17H00762, 16H02870 and 16H01743 from the Ministry of Education, Culture, Sports, Science and Technology, Japan.

## REFERENCES

Akutsu, T., Fukagawa, D., Halldórsson, M. M., Takasu, A., and Tanaka, K. (2013). Approximation and parameterized algorithms for common subtrees and edit distance between unordered trees. *Theoret. Comput. Sci.*, 470:10–22.

Deza, M. M. and Deza, E. (2016). *Encyclopedia of distances (4th ed.)*. Springer.

Gallian, J. A. (2007). A dynamic survey of graph labeling. *Electorn. J. Combin.*, 14:DS6.

Hirata, K., Yamamoto, Y., and Kuboyama, T. (2011). Improved MAX SNP-hard results for finding an edit distance between unordered trees. In *Proc. CPM'11 (LNCS 6661)*, pages 402–415.

Kawaguchi, T., Yoshino, T., and Hirata, K. (2018). Path histogram distance for rooted labeled caterpillars. In *Proc. ACIIDS'18 (LNAI 10751)*, pages 276–286.

Muraka, K., Yoshino, T., and Hirata, K. (2018). Computing edit distance between rooted labeled caterpillars. In *Proc. FedCSIS'18*, pages 245–252.

Tai, K.-C. (1979). The tree-to-tree correction problem. *J. ACM*, 26:422–433.

Yamamoto, Y., Hirata, K., and Kuboyama, T. (2014). Tractable and intractable variations of unordered tree edit distance. *Internat. J. Found. Comput. Sci.*, 25:307–329.

Yoshino, T., Muraka, K., and Hirata, K. (2018). LCA histogram distance for rooted labeled caterpillars. In *Proc. KDIR'18*, pages 307–314.

Zhang, K. and Jiang, T. (1994). Some MAX SNP-hard results concerning unordered labeled trees. *Inform. Process. Lett.*, 49:249–254.

Zhang, K., Wang, J., and Shasha, D. (1996). On the editing distance between undirected acyclic graphs. *Internat. J. Found. Comput. Sci.*, 7:43–58.