社団法人　情報処理学会　研究報告
IPSJ SIG Technical Report

2005－DBS－136（7）
2005－FI－79（7）
2005／5／19

半構造データ統合のための木構造の近似照合と結合手法

久保山　哲二† 　　申　吉浩†† 　安田　浩††

† 東京大学 国際・産学共同研究センター　†† 東京大学 先端科学技術研究センター

概要　インターネット上の XML や HTML 文書等の半構造データの増大にともない、膨大な半構造データを効率よく比較・照合するための手法や、複数の半構造データを統合するための手法が求められている。これまでに半構造データのための様々な照合・結合手法が提案されているものの、一般的なフレームワークが存在しないため同様の手法が独立に繰り返し提案されることも少なくない。本稿では、2 つの半構造データを結合するための一般的かつ理論的なフレームワークを提供する。本手法では、まず、木の編集距離の概念を用いて 2 つの木の近似照合を行い、2 つの木の間で類似しているノードの対応をとる。次に、対応のとれたノード同士を重ね合わせ、その他の全てのノードが、もとの木における階層構造を保つように新しい木を生成する。このような数理的なフレームワークによって、様々な照合・統合手法に統一的な観点を提供し、効率的な実装のための基礎を与える。

Approximate Tree Matching and Merging for Integrating Semistructured Data

Tetsuji KUBOYAMA†, Kilho SHIN††, and Hiroshi YASUDA††

† Center for Collaborative Research, The University of Tokyo
†† Research Center for Advanced Science and Technology, University of Tokyo

Abstract　With the rapid growth of semistructured data such as XML and HTML documents on the Internet, we need efficient methods for comparing, matching and integrating semistructured data. Although there have been diversity of these methods recently, no comprehensive framework has been available. In this paper, we formulate and provide a new framework for merging semistructured data. In this framework, we firstly find a set of node-to-node correspondences between two trees by approximate matching based on tree edit distance. Then, we merge two trees by overlaying these corresponding nodes, and locating the other nodes so that each ancestor relation in two trees is preserved. This mathematical framework gives a unifying view of matching and merging semistructured data, and is beneficial to efficient implementations.

1. はじめに

インターネット上に蓄積されている膨大な情報を系統的に抽出し、利用する技術の開発は、大量の情報資源をデータベースとして有効利用するために必須の技術である。本稿では、2 つの木構造の近似照合の結果を用いて、2 つの木を 1 つに結合するための一般的なフレームワークの提案を目的としている。これまでに半構造データ結合のための様々な手法が提案されている [1]〜[4]。しかし、一般的なフレームワークが存在しないために、同様の手法が独立に繰り返し提案されている。このような状況の中、適応領域に合った手法の選択や効率的なアルゴリズム設計のためにも、様々な手法を統一的に記述できる一般的なフレームワークが必要とされている。

本研究では、半構造データを一般的な木構造データとしてとらえ、木の編集距離による近似照合から、2 つの木を結合するフレームワークを提案する。本手法では、まず、木の編集距離の概念を用いて 2 つの木の近似照合を行い、2 つの

木の間で類似しているノードの対応をとる。この対応を木のマッピングという。次に、対応のとれたノード同士を重ね合わせ、その他の全てのノードが、もとの木における階層構造を保つように新しい木を生成する。

木の編集距離には、様々なクラスがあり [5]〜[7]、Zhang らによって提案された一般的な木の編集距離手法 [8], [9] では、結合した結果が木になることが保証されない。一方で、Jiang らにより提案された木のアラインメント [10] および、Lu らにより提案された less-constrained 編集距離 [11] が木の結合を可能にする最も一般的なクラスであることが、久保山らにより示されている [7]。また、Zhang らにより提案された constrained 編集距離 [12] による近似照合手法は、木のアラインメントのクラスに真に含まれることが知られており、このような既存の様々な木の近似照合の数だけ、木の結合手法も考えられる。本フレームワークにより、木のアラインメントのクラスに含まれる様々な近似照合手法から木を結合する一般的な方法が得られる。

以下、2 節では木の近似照合と結合を数理的に記述するための概念を導入し、3 節は、more-constrained という扱いやすい木のマッピングクラスを用いて実際に木を結合する手法を与えるための構成的な証明を示す。4 節では、より一般的な近似照合のクラスにこの証明を拡張する。最後に議論と本稿のまとめを行う。

2. 準　　備

2.1 木

本稿では、木を半順序集合上の構造として表現する。**厳密半順序**(strict partial order) を表すために、標準的な表記 $<$ を用いる。すなわち、非空有限集合 V について次の条件を満たすものとする。

（1）$\forall x,y,z \in V\ [x<y \ \wedge\ y<z \Rightarrow x<z]$　　(推移性),
（2）$\forall x \in V\ [x \not< x]$　　(非対称性).

また、任意の $x,y \in V$ について、表記 $x \leqq y$ により $x<y$ または $x=y$ であることを表す。

定義 1 (根つき木)**. 根つき木** $T=(V,<)$ は、**根**という最大要素 $r(T) \in V$ をもつ非空の有限厳密半順序集合で、かつ $\{y \in V | x \leqq y\}$ が任意の $x \in V$ について、全順序集合となるものである。

以降、根つき木を、単に木という。集合 V の要素を T の**ノード**と呼び、T のすべてのノードからなる集合を $V(T)$ で表す。木 T の**辺集合**を $E(T)=\{(x,y) \in V(T) \times V(T) | x < y \ \wedge\ \nexists z \in V(T)\ [x<z<y]\}$ により定義し、辺集合の要素を辺という。ノード x の**祖先**は、$x \leqq y$ となるようなノード y である。とくに、$x<y$ のとき、y を**真の祖先**という。ノード x の**親**とは、x の真の祖先のなかで最小のノードであり、$p(x)$ で表す。ノード x の **子集合**は、集合 $\{y | (y,x) \in E(T)\}$ であり、$\mathrm{ch}(x)$ で表す。子集合 $\mathrm{ch}(x)$ の要素を**子**という。また、T の極小ノードを**葉**という。木 T に含まれるノードの数を、木の大きさと定義し、$|T|$ で表す。

定義 2. 木 $T=(V,<)$ と集合 $V' \subseteqq V$ が与えられたとき、任意のノード $y \in V'$ について $y \leqq x$ となるようなノード $x \in V$ を V' の**共通祖先**(common ancestor) という。V' の共通祖先 x が、任意の V' の共通祖先の中で最小であるとき、**最小共通祖先**(least common ancestor) という。

$\mathrm{lca}(V')$ により、V' の最小共通祖先、$x \smile y$ により、$\mathrm{lca}(\{x,y\})$ をそれぞれ表す。

補題 1. 木のノード間では、次の性質が成り立つ。

（1）$x \smile x = x$,
（2）$x \smile y = y \smile x$,
（3）$(x \smile y) \smile z = x \smile (y \smile z)$,
（4）$x \leqq y \Leftrightarrow x \smile y = y$,
（5）$x \smile y < x \smile z \Rightarrow y \smile z = x \smile z$,
（6）$x \smile y = x \smile z \Rightarrow y \smile z \leqq x \smile y$.

2.2 木の近似照合とマッピング

2 つの木の間での近似照合の表現に、**マッピング**(tree mapping) という概念を用いる。木マッピングとは、2 つの木の間のノード間の対応関係を表す集合であり、次のように定義する。

定義 3 (マッピング)**.** 木 T_A から T_B への**マッピング** M とは、$V(T_A) \times V(T_B)$ の部分集合であり、次の条件を満たす集合である。

$$\forall (x_1,x_2),(y_1,y_2) \in M\ [x_1 \leqq y_1 \Leftrightarrow x_2 \leqq y_2].$$

この定義は、直感的には、ノードの対応が一対一であること、ノードの上下関係が保存されることを示している。

次のマッピングは Zhang により導入された。

定義 4 (Zhang [9])**.** マッピング M は、次の条件を満たすとき *constrained* であるという。

(C) $\forall (x_1,x_2),(y_1,y_2),(z_1,z_2) \in M$
$[z_1 < x_1 \smile y_1 \Leftrightarrow z_2 < x_2 \smile y_2]$

2.3 木の結合

あるマッピングが与えられたときの木の結合を定義する。

定義 5 (準同型)**.** 木 T_A から T_B への**準同型**(homomorphism) 写像とは、任意の $x,y \in V(T_A)$ に

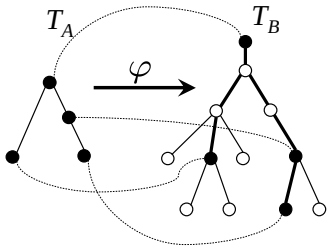


図 1　埋め込みの例

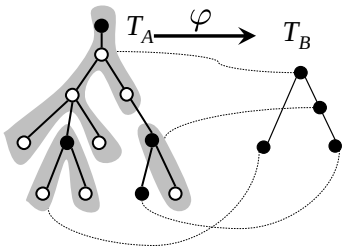


図 2　縮 退 の 例

ついて、$x < y$ ならば $\varphi(x) \leqq \varphi(y)$ となるような写像 $\varphi : V(T_A) \to V(T_B)$ と定義する。

写像 $\varphi : V(T_A) \to V(T_B)$ が、木 T_A から T_B への準同型写像であるとき、$\varphi : T_A \to T_B$ と表記する。

定義 6 (埋め込み)**.** 木 T_A と T_B について、次の条件を満たす準同型写像 $\varphi : T_A \to T_B$ を、T_A から T_B への**埋め込み**(embedding) という。

（1）φ は単射

（2）$\forall x, y \in V(T_A)\ [\varphi(x) < \varphi(y) \Rightarrow x < y]$

$\mathrm{red}(\varphi) = |V(T_B) \setminus \varphi(V(T_A))|$ を埋め込み $\varphi : T_A \to T_B$ の**冗長度**という。

図 1 に埋め込みの例を示す。

命題 2 (久保山ら [7])**.** 埋め込み $\varphi : T_A \to T_B$ とノード $x, y \in V(T_A)$ について、$\varphi(x) \smile \varphi(y) < \varphi(z)$ を満たす T_B のノードのうちで最小のノード $\varphi(z)$ は $\varphi(x \smile y)$ に等しい。また、次の性質は等価である。

（1）$\varphi(x) \smile \varphi(y) < \varphi(x \smile y)$

（2）$\varphi(x) \smile \varphi(y) \notin \varphi(V(T_A))$

系 3 (久保山ら [7])**.** 埋め込み $\varphi : T_A \to T_B$ について、$x \smile y < x \smile z$ ならば、$\varphi(x) \smile \varphi(y) < \varphi(x) \smile \varphi(z)$。

定義 7 (縮退)**.** 木 T_A と T_B について、次の条件を満たす準同型写像 $\varphi : T_A \to T_B$ を、T_A から T_B への**縮退**(degeneration) という。

（1）φ は全射

（2）$\forall x, y \in V(T_A)\ [\varphi(x) = \varphi(y) \Rightarrow \varphi(x \smile y) = \varphi(x)]$

（3）$\forall x, y \in V(T_A)$

$[\varphi(x) < \varphi(y) \Rightarrow \exists z \in V(T)[\varphi(y) = \varphi(z) \wedge x < z]]$

$\mathrm{Dup}(\varphi) = \{x \in V(T_A) | \varphi(x) = \varphi(p(x))\}$ を縮退 $\varphi : T_A \to T_B$ の**重複度**という。

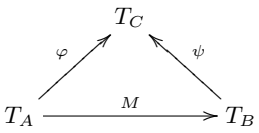
図 2 に縮退の例を示す。

定義 8 (木の結合)**.** 下記の条件を満たす 3 つ組 (T_C, φ, ψ) が存在するとき、木 T_A から T_B へのマッピング M は *alignable* であるという。

（1）$\varphi : T_A \to T_C$ は、埋め込み

（2）$\psi : T_B \to T_C$ は、埋め込み

（3）$\forall (x, y) \in M\ [\varphi(x) = \psi(y)]$



また、(T_C, φ, ψ) を M による木 T_A と T_B の**結合**といい、T_C を**結合木**という。

定義 9 (挿入)**.** 木 T_A と T_B について、$\mathrm{red}(\varphi) = 1$ であるような埋め込み $\varphi : T_A \to T_B$ を **挿入**という。とくに、ノード $x \neq r(U)$ について $\varphi(V(T_A)) = V(T_B) \setminus \{x\}$ であるような挿入 φ を x-挿入という。

ここで、木のノードをフィルタリングするための表記法を導入する。木 T について、$\pi(\xi) : V(T) \to \{\textsc{True}, \textsc{False}\}$ を述語変数 ξ をもつ 1 変数述語であるとする。

定義 10 (木の論理表記)**.** $T[\pi(\xi)] = (V[\pi(\xi)], <_\pi)$ を次のように定義する。

（1）$V[\pi(\xi)] = \{x | x \in V(T) \ \wedge \ \pi(x) = \textsc{True}\}$

（2）$\forall x, y \in V[\pi(\xi)]\ [x <_\pi y \Leftrightarrow x < y]$

$T[\pi(\xi)]$ は、根を持たなくてもよいことに注意されたい (いわゆる森でもよい)。

2.4　結合木の存在するマッピング条件

2 つの木の結合木が存在するための必要十分条件が次の定理により示されている。

定理 4 (久保山ら [7])**.** マッピング M が、次の条件を満たすとき、かつそのときに限り、木の結合が存在する。

(A)　$\forall (x_1, x_2), (y_1, y_2), (z_1, z_2) \in M$

$[x_1 \smile y_1 < x_1 \smile z_1 \Rightarrow y_2 \smile z_2 = x_2 \smile z_2]$.

すなわち、この条件を満たすマッピングが alignable マッピングである。

alignable マッピングは、less-constrained マッピングともよばれる。

命題 5. 条件 (A) と (A') は等価である。

(A') $\forall(x_1,x_2),(y_1,y_2),(z_1,z_2)\in M$
$[x_2\smile y_2 < x_2\smile z_2 \Rightarrow y_1\smile z_1 = x_1\smile z_1]$.

Proof. 対称性より、(A)⇒(A') を示せば十分。よって、$y_1\smile z_1 \neq x_1\smile z_1$ のとき、$x_2\smile y_2 \not< x_2\smile z_2$ となることを示す。ノード $y_1\smile z_1$ と $x_1\smile z_1$ が比較可能であることから、$y_1\smile z_1 < x_1\smile z_1$ のとき、(A) により $x_2\smile y_2 = x_2\smile z_2$ を得る。また、$y_1\smile z_1 > x_1\smile z_1$ のとき、(A) により $x_2\smile y_2 = y_2\smile z_2$ を得る。よって、$x_2\smile y_2 \geqq x_2\smile z_2$。 □

3. 結合木の構成的な存在証明

本節では、まず、新しいマッピング more-constrained マッピングを導入し、2 つの木は more-constrained マッピングにより統合可能であることを構成的証明により示す。

定義 11. マッピング M は、次の条件を満たすとき *more-constrained* であるという。

(M) $\forall(x_1,x_2),(y_1,y_2),(z_1,z_2)\in M$
$[x_1\smile y_1 = x_1\smile z_1 \Leftrightarrow x_2\smile y_2 = x_2\smile z_2]$

命題 6. マッピング M について、次の (M), (M'), (Q), (Q') は、いずれも等価である。

(M') $\forall(x_1,x_2),(y_1,y_2),(z_1,z_2)\in M$
$[x_1\smile y_1 < x_1\smile z_1 \Leftrightarrow x_2\smile y_2 < x_2\smile z_2]$

(Q) $\forall(x_1,x_2),(y_1,y_2),(z_1,z_2),(u_1,u_2)\in M$
$[x_1\smile y_1 = z_1\smile u_1 \Leftrightarrow x_2\smile y_2 = z_2\smile u_2]$

(Q') $\forall(x_1,x_2),(y_1,y_2),(z_1,z_2),(u_1,u_2)\in M$
$[x_1\smile y_1 < z_1\smile u_1 \Leftrightarrow x_2\smile y_2 < z_2\smile u_2]$

証明は省略する。

命題 7. $x\smile x' < x\smile y$ かつ $y\smile y' < x\smile y$ ならば、次の 2 つの性質が成り立つ。

（1）$x\smile y = x'\smile y'$

（2）M が $(x,\bar{x}),(y,\bar{y}),(x',\bar{x}'),(y',\bar{y}')$ を含む alignable マッピングならば、$\bar{x}\smile\bar{y} = \bar{x}'\smile\bar{y}'$

Proof. **(1):** $x\smile x' < x\smile y$ より $x'\smile y = x\smile y$。よって $x'\smile y' = x'\smile y = x\smile y$ を得る。

(2): M は alinable マッピングより、$x\smile x' < x\smile y$ から $\bar{x}'\smile\bar{y} = \bar{x}\smile\bar{y}$ を得る。また、(1) により $y\smile y' < x'\smile y'$ が成り立つ。よって $\bar{x}'\smile\bar{y}' = \bar{x}'\smile\bar{y} = \bar{x}\smile\bar{y}$ を得る。 □

定義 12. 木 T のノード集合 $X\subseteqq V(T)$ について、ノード $x\in X$ の真の祖先が X 中に存在するとき、その最小の祖先を $\mu_X(x)$ と表記する。すなわち、$\mu_X(x) = \min\{y\in X | y > x\}$ とする。

定義 13. 木 T_A から T_B へのマッピング M について、$M^* = \{(x\smile y, \bar{x}\smile\bar{y}) | (x,\bar{x}),(y,\bar{y}) \in M\} \cup \{(r(T_A), r(T_B))\}$ とおく。

一般の 2 項関係 R について、$R|_1 = \{x | (x,y)\in R\}$、$R|_2 = \{y | (x,y)\in R\}$ とする。

補題 8. 木 T_A から T_B へのマッピング $M = \{(a_1,b_1),(a_2,b_2),\ldots\}$ および、$B = M^*|_2$ について, x, y をそれぞれ次のようにおく。

- $b_i\smile b_j \leqq x < \mu_B(b_i\smile b_j)$
- $b_k\smile b_l \leqq y < \mu_B(b_k\smile b_l)$

このとき、$x \leqq y$ ならば, $b_i\smile b_j \leqq b_k\smile b_l$ かつ $\mu_B(b_i\smile b_j) \leqq \mu_B(b_k\smile b_l)$ となる。

Proof. $x\leqq y$ より、$\mu_B(b_i\smile b_j)$ と $\mu_B(b_k\smile b_l)$ は比較可能であり、$\mu_B(b_i\smile b_j) \leqq \mu_B(b_k\smile b_l)$ である。このとき、$b_i\smile b_j$ と $b_k\smile b_l$ が比較可能ならば、$b_i\smile b_j \leqq b_k\smile b_l$ となる。また、$b_i\smile b_j$ と $b_k\smile b_l$ が比較不可能ならば、補題 1 により、$(b_i\smile b_j)\smile(b_k\smile b_l)$ は、B の要素となる。よって、$\mu_B(b_k\smile b_l) = (b_i\smile b_j)\smile(b_k\smile b_l)$ となり、これは、仮定 $x\leqq y$ に反する。 □

対称性から、$A = M^*|_1$ についても、同様の補題が得られる。

定義 14. 互いに疎なノード $x,y\in V(T_A)\cup V(T_B)$ について、順序 $<_M$ を次のように定義する。

関係 $x <_M y$ が成り立つときかつそのときに限り、次の条件の少なくとも 1 つが満たされる。

（U1）$x,y\in V(T_A)\ \wedge\ x < y$

（U2）$x,y\in V(T_B)\ \wedge\ x < y$

（U3）$x\in V(T_A)\ \wedge\ y\in V(T_B)\ \wedge$
$\exists i,j[x\leqq a_i\smile a_j\ \wedge\ b_i\smile b_j\leqq y]$

（U4）$x\in V(T_B)\ \wedge\ y\in V(T_A)\ \wedge$
$\exists i,j\exists z\in V(T_B)$
$[b_i\smile b_j\leqq z < \mu_B(b_i\smile b_j)\wedge x\leqq z\wedge a_i\smile a_j < y]$

（U5）$y = r(T_B)$

条件 (U3) と (U4) では、$i=j$ となる場合もあることに注意。

関係 $<_M$ について、次の性質が成り立つ。

補題 9. 木 T_A から T_B へのマッピング M が more-constrained ならば、$T_C = (V(T_A)\cup V(T_B), <_M)$ は、根つき木であり、任意の $(x,\bar{x}),(y,\bar{y})\in M$ について、$p(x\smile y) = \bar{x}\smile\bar{y}$ となる。

Proof. この補題をいくつかの、claim に分け、証明を進める。以下の証明では、$A = M^*|_1$、$B = M^*|_2$ とする。

Claim 1. 互いに疎なノード $x,y\in V(T_A)\cup V(T_B)$ について、$x <_M y$ かつ $y <_M x$ となることはない。

Proof. ノード集合 B には、$r(T_B)$ の真の祖先となるノードは存在しないことから、任意の $x \in V(T_A)$ について、(U4) から、$r(T) <_M x$ となることはない。よって、x と y のいずれも $r(T_A)$ と等しくなることはない。

ここで、$x, y \in V(T_A)$ または $x, y \in V(T_B)$ のとき、(U1) と (U2) により、claim が成立する。よって、以下では $x \in V(T_A)$ かつ $y \in V(T_B)$ と仮定する。$x <_M y$ のとき、$x \leqq s_i \frown s_j \wedge y < b_i \frown b_j$ となる i, j が存在する。任意の k, l について、$a_k \frown a_l < a_i \frown a_j$ ならば、命題 6 により、$b_k \frown b_l < b_i \frown b_j$ となる。とくに、$\mu_B(b_k \frown b_l) \leqq b_i \frown b_j$ が成立する。よって、$z < \mu_B(b_k \frown b_l)$ となるような z は存在しないことから、(U4) により $y <_M x$ が否定された。 □

Claim 2. $x <_M y$ かつ $y <_M z$ ならば $x <_M y$。

Proof. 以下の場合それぞれについて証明する。

（a）$x, y, z \in V(T_A)$
（b）$x, y, z \in V(T_B)$
（c）$x \in V(T_A) \wedge y, z \in V(T_B)$
（d）$x, y \in V(T_A) \wedge z \in V(T_B)$
（e）$x \in V(T_A) \wedge y, z \in V(T_B)$
（f）$x, y \in V(T_A) \wedge z \in V(T_B)$
（g）$x, z \in V(T_A) \wedge y \in V(T_B)$
（h）$x, z \in V(T_A) \wedge y \in V(T_B)$

(a)、(b) の場合は明らか。

(c): (U3) から、$x <_M y$ ならば、$x \leqq a_i \frown a_j \wedge b_i \frown b_j \leqq y$ となる i, j が存在する。また、$y < z$ より $b_i \frown j_j \leqq y < z$ となり、(U3) から $x <_M z$ を得る。

(d): (c) と同様に証明できる。

(e): (U4) より $x <_M y$ ならば $b_i \frown b_j \leqq w < \mu_B(b_i \frown b_j) \wedge x \leqq w \wedge a_i \frown a_j < y$ となる i, j および $w \in V(T_B)$ が存在する。また $y < z$ $a_i \frown a_j < y < z$ となり、(U4) により $x <_M z$ が得られる。

(f): (d) と同様に証明できる。

(g): $x <_M y$ ならば、(U4) より $b_i \frown b_j \leqq w < \mu_B(b_i \frown b_j) \wedge x \leqq w \wedge a_i \frown a_j < y$ となる i, j および $w \in V(T_B)$ が存在する。また (U3) より、$y <_M z$ ならば、$y \leqq a_k \frown a_l \wedge b_k \frown b_l \leqq z$ となる k, l が存在する。まず、$a_i \frown a_j < a_k \frown a_l$ より命題 6 から $b_i \frown b_j < b_k \frown b_l$ を得る。特に、$\mu_B(b_i \frown b_j) \leqq b_k \frown b_l$ となる。ゆえに、$x \leqq w < \mu_B(b_i \frown b_j) \leqq b_k \frown b_l \leqq z$ から、$x < z$ を得る。

(h): (U3) から $x <_M y$ ならば、$x \leqq a_k \frown a_l \wedge b_k \frown b_l \leqq y$ とる k, l が存在し、(U4) から $y <_M z$ ならば、$b_i \frown b_j \leqq w < \mu_B(b_i \frown b_j) \wedge y \leqq w \wedge a_i \frown a_j < z$ となる i, j および $w \in V(T_B)$ が存在する。ここで、w を $(b_i \frown b_j) \frown (b_k \frown b_l)$ と同一視してもよい。すると、$(b_i \frown b_j) \frown (b_k \frown b_l) \in B$ より、w は $b_i \frown b_j$ そのものであることがわかる。さらに、命題 6 より $b_k \frown b_l \leqq b_i \frown b_j$ を得る。よって $a_k \frown a_l \leqq a_i \frown a_j$ より $x \leqq a_k \frown a_l \leqq a_i \frown a_j < z$ を得る。 □

Claim 3. 互いに疎なノード $x, y, z \in V(T_A) \cup V(T_B)$ について、$x <_M y$ かつ $x <_M z$ ならば、$y <_M z$ または $z <_M y$ となる。

Proof. 自明な場合を除外して、$y \neq r(T_B)$ と $z \neq r(T_B)$ と仮定する。よって、次の場合について証明すれば十分である。

（a）$x \in V(T_A)$ かつ $y, z \in V(T_B)$
（b）$x, y \in V(T_A)$ かつ $z \in V(T_B)$
（c）$x \in V(T_B)$ かつ $y, z \in V(T_A)$
（d）$x, y \in V(T_B)$ かつ $z \in V(T_A)$

(a): (U3) より $x \leqq a_i \frown a_j$, $x \leqq a_k \frown a_l$, $b_i \frown b_j \leqq y$ かつ $b_k \frown b_l \leqq z$ となる i, j, k, l が存在する。$a_i \frown a_j$ and $a_k \frown a_l$ は比較可能なことから、$a_i \frown a_j \leqq a_k \frown a_l$ と仮定する。命題 6 により $b_i \frown b_j \leqq b_k \frown b_l$ なので、y と z は比較可能である。

(b): (U3) より $x \leqq a_i \frown a_j$ かつ $b_i \frown b_j \leqq z$ となる i, j が存在する。よって、$b_k \frown b_l \leqq z < \mu_B(b_k \frown b_l)$ となる k, l が存在する。$b_i \frown b_j \leqq b_k \frown b_l$ より、命題 6 により、$a_i \frown a_j \leqq a_k \frown a_l$ が成立する。さらに、$x \leqq a_i \frown a_j \leqq a_k \frown a_l$ かつ $x < y$ より、$a_k \frown a_l$ と y は比較可能である。$y \leqq a_k \frown a_l$ のとき、(U3) から $y <_M z$ となり、$a_k \frown a_l < y$ のとき、(U4) から $z <_M y$ となる。

(c): (U4) から $a_i \frown a_j < y$, $a_k \frown a_l < z$, $b_i \frown b_j \leqq v < \mu_B(b_i \frown b_j)$, $b_k \frown b_l \leqq w < \mu_B(b_k \frown b_l)$, $x < v$ かつ $x < w$ となる i, j, k, l および $v, w \in V(T_B)$ が存在する。v, w が比較可能なことから、$b_i \frown b_j$ と $b_k \frown b_l$ も、補題 8 により比較可能である。よって、$a_i \frown a_j$ と $a_k \frown a_l$ も比較可能であり、y と z は比較可能である。

(d): (U4) より $a_i \frown a_j < z$, $b_i \frown b_j \leqq w < \mu_B(b_i \frown b_j)$, $x \leqq w$ となる i, j および、$w \in V(T_B)$ が存在する。今、k, l が $a_k \frown a_l < z \leqq \mu_A(a_k \frown a_l)$ かつ $a_i \frown a_j \leqq a_k \frown a_l$ を満たすとする。すると、$b_i \frown b_j \leqq b_k \frown b_l$ より、$w < \mu_B(b_k \frown b_l)$ を得る。ゆえに、y と $\mu_B(b_k \frown b_l)$ は比較可能である。$y < \mu_B(b_k \frown b_l)$ ならば、(U3) より $y <_M z$ となり、$\mu_B(b_k \frown b_l) \leqq y$ ならば、(U4) より $z <_M y$ となる。 □

$r(T_B)$ は、順序 $<_M$ における最大要素であることから、Claim 1. から Claim 3. は、$(V(T_A) \cup V(T_B), <_M)$ が根つき木であることを含意する。

最後に、次の claim を示す。

Claim 4. 任意の i, j について、$p(a_i \frown a_j) = b_i \frown b_j$ が成立する。

Proof. (U3) より $a_i \frown a_j <_M b_i \frown b_j$。ここで、$a_i \frown a_j <_M$

x となる任意の x について、$b_i \smile b_j \leqq_M x$ であることを示す。$a_i \smile a_j < x$ と (U4) により、$x \in V(T_A)$ かつ $b_i \smile b_j <_M x$ は明らか。$x \in V(T_B)$ を仮定すると、ある k,l が存在して、$a_i \smile a_j \leqq a_k \smile a_l$ かつ $b_k \smile b_l \leqq x$ となる。命題 6 より $a_i \smile a_j \leqq a_k \smile a_l$ であることから、$b_i \smile b_j \leqq b_k \smile b_l$ を得る。よって、$b_i \smile b_j \leqq x$. □

すべての claim を証明したことにより補題 9 の証明が完了した。 □

以降、 more-constrained マッピング M について、$(V(T_A) \cup V(T_B), <_M)$ を $T_A \cup_M T_B$ と表記する。

補題 10. 木 $T_A \cup_M T_B$ について、次の性質が成り立つ。
（1）自然な包含写像 $\varphi' : V(T_A) \to V(T_A) \cup V(T_B)$ は、T_A から $T_A \cup_M T_B$ への埋め込みである。
（2）自然な包含写像 $\psi' : V(T_B) \to V(T_A) \cup V(T_B)$ は、T_B から $T_A \cup_M T_B$ への埋め込みである。

Proof. 証明は容易なので省略する。 □

ここで、$\mathrm{Dup}(\eta) = \{a \smile a' | (a,b),(a',b') \in M\}$ となるような縮退 $\eta : T_A \cup_M T_B \to T_C$ が存在する。実際、$\pi(\xi) = \exists a, a', b, b' [(a,b) \in M \wedge (a',b') \in M \wedge \xi = a \smile a']$ とおき、$\eta = \mathcal{D}_{\pi(\xi)} : T_A \cup_M T_B \to (T_A \cup_M T_B)[\pi(\xi)]$ とおくことによって示せる。

命題 11. $\eta : T_A \cup_M T_B \to T_C$ を $\mathrm{Dup}(\eta) = \{a \smile a' | (a,b),(a',b') \in M\}$ を満たす縮退であるとする。$\varphi = \eta \circ \varphi'$ かつ $\psi = \eta \circ \psi'$ のとき、次の性質が成り立つ。
（1）φ と ψ は単射
（2）$\forall (a,b),(a',b') \in M\ [\varphi(a \smile a') = \psi(b \smile b')]$
（3）とくに、$\forall (a,b)\ [\varphi(a) = \psi(b)]$

Proof. 証明は容易なので省略する。 □

定理 12. $\eta : T_A \cup_M T_B \to T_C$ を $\mathrm{Dup}(\eta) = \{a \smile a' | (a,b),(a',b') \in M\}$ を満たす縮退であるとする。$\varphi = \eta \circ \varphi'$ かつ $\psi = \eta \circ \psi'$ のとき、(T_C, φ, ψ) は、マッピング M による結合木である。

Proof. 命題 11 より直ちに得られる。 □

ここで、補題 9 の条件を緩めることを考える。すなわち、マッピング M が more-constrained でなければならないという条件を、次の条件 (I1) と (I2) に緩めてみる。この条件で、補題 9 が成立するならば、alignable マッピングにおいても、結合木が構成的に得られると期待できる。
（I1）$a_1 \smile a_2 < a_1 \smile a_3 \Rightarrow b_1 \smile b_2 \leqq b_1 \smile b_3$.
（I2）$a_1 \smile a_2 = a_1 \smile a_3 \Rightarrow b_1 \smile b_2 < b_1 \smile b_3$.
この条件の下では、補題 9 の Claim 1 は成り立つことが証明できる。しかし、Claim 2 の (h) が成り立たないことがわかる。たとえば、$M = \{(a_i, b_i) | i \in \{1,2,3,4\}\}$ を次を満たすマッピングとする。

- すべての $a_i \smile a_j$ は $i = 1,2$ および $j = 3,4$ において、互いに等しい
- $a_1 \smile a_2 < a_1 \smile a_3$ かつ $a_3 \smile a_4 < a_1 \smile a_3$
- すべての $b_i \smile b_j$ は $i = 1,2$ および $j = 3,4$ において、互いに等しい
- $b_1 \smile b_2 < b_1 \smile b_3$ かつ $b_3 \smile b_4 = b_1 \smile b_3$.

このとき、M が (I1) と (I2) を満たすことは明らかである。しかし、$x \leqq a_1 \smile a_2$、 $b_3 \smile b_4 \leqq y < r(T_B)$、$a_3 \smile a_4 < z < a_1 \smile a_4$ とすると、$x <_M y$ かつ $y <_M z$ は成り立つが、$x <_M z$ は、成り立たない。

定義 15. M を木 T_A から木 T_B へのマッピングとする。互いに疎な対 $(a_i,b_i),(a_j,b_j) \in M$ は、次の条件が成り立つとき、T_A に関して**極小**であるという。

$$\forall (a_k,b_k),(a_l,b_l) \in M\ [a_k \smile a_l = a_i \smile a_j \ \wedge \\ b_k \smile b_l \leqq b_i \smile b_j \Rightarrow b_k \smile b_l = b_i \smile b_j]$$

定義 16. T_A におけるマッピング M の**結節**(knot) とは、次の条件を満たす 3 要素からなる M の部分集合 $\{(a_i,b_i),(a_j,b_j),(a_k,b_k)\}$ である。
（1）$a_i \smile a_j = a_j \smile a_k = a_k \smile a_i$
（2）a_i, a_j, a_k は、いずれも他方の祖先ではない
T_A における M の結節からなる集合を $\mathfrak{K}_{T_A}(M)$ とする。$\mathfrak{K}_{T_B}(M)$ についても、同様に定義する。

命題 13. $\{(a_i,b_i),(a_j,b_j),(a_k,b_k)\} \in \mathfrak{K}_{T_A}(M)$ とする。互いに疎な 2 つの対 $(x,\bar{x}),(y,\bar{y}) \in M$ が $x \smile y = a_i \smile a_j$ を満たすとき、$\{(x,\bar{x}),(y,\bar{y}),(a_l,b_l)\} \in \mathfrak{K}_{T_A}(M)$ となる k が存在する。

Proof. 一般性を失うことなく $x \neq a_i$、$y \neq a_i$、$y \neq a_j$ と仮定してよい。

$x \smile a_i = y \smile a_i$ のとき、$a_i \leqq x \smile y = a_i \smile a_j$ より $x \smile a_i = x \smile y$ が成り立つ。よって、$l = i$ とすると、命題が成り立つ。

ここで、必要ならば x と y を入れ替えて、$x \smile a_i < x \smile y$ と仮定する。仮定より $x \neq a_j$ と $x \neq a_k$ が直ちに得られる。よって、$x \smile a_j = x \smile a_k = x \smile y$ となる。$z \smile a_j = x \smile y$ ならば、$l = j$ で命題が成り立つ。$z \smile a_j < x \smile y$ ならば、$y \neq a_k$ かつ $z \smile a_k = x \smile y$ より、$l = k$ で命題が成り立つ。 □

定理 14. 任意の alignable マッピング $M \subset V(T_A) \times V(T_B)$ について、$\bar{M} = \{(\alpha(x), \beta(y)) | (x,y) \in M\}$ が more-constrained マッピングとなるような埋め込み $\alpha : T_A \to \bar{T}_A$ と $\beta : T_B \to \bar{T}_B$ が存在する。

Proof. $\{(\bar{a}_i,\bar{b}_i),(\bar{a}_j,\bar{b}_j),(\bar{a}_k,\bar{b}_k)\}$ を $\mathfrak{K}_{T_A}(M)$ の要素とする。また、命題 13 により, 対 $(\bar{a}_i,\bar{b}_i),(\bar{a}_j,\bar{b}_j)$ は、T_A に関して極小であると仮定してよい。

ここで、$\alpha' : T_A \to T'_A$ を $\alpha'(x) < \sigma$ かつそのときに限り $x \smile \bar{a}_i < \bar{a}_i \smile \bar{a}_j$ または $x \smile \bar{a}_j < \bar{a}_i \smile \bar{a}_j$ となるような、σ-挿入であるとする。

特に、$\alpha'(\bar{a}_i) \smile \alpha'(\bar{a}_j) = \sigma$ かつ $\alpha'(\bar{a}_i) \smile \alpha'(\bar{a}_k) = \alpha'(\bar{a}_j) \smile \alpha'(\bar{a}_k) = p(\sigma)$ が成り立つ。

証明の準備として次の 2 つの claim が満たされることを示す。

Claim 1. $M' = \{(\alpha'(x), y) | (x, y) \in M\}$ は、alignable マッピングである。

Proof. まず、任意の $(a_l, b_l) \in M$ $(i \in \{i, j, k\})$ について、$\alpha'(a_i) \smile \alpha'(a_j) < \alpha'(a_i) \smile \alpha'(a_k)$ ならば、$b_j \smile b_k = b_i \smile b_k$ であることを示す。

ここで、$a_i \smile a_j$ と $a_i \smile a_k$ が比較可能であることに留意すると、$a_i \smile a_k < a_i \smile a_j$ は決して成り立たない。なぜならば、成り立つとすると、系 3 により $\alpha'(a_i) \smile \alpha'(a_k) < \alpha'(a_i) \smile \alpha'(a_j)$ となる。

$a_i \smile a_j < a_i \smile a_k$ のとき、M が alignable マッピングとなるであることから、$b_j \smile b_k = b_i \smile b_k$ を得る。$a_i \smile a_j = a_i \smile a_k$ とすると、$a_j \smile a_k < a_i \smile a_j$ は決し成り立たない。なぜなら、成り立つとすると $\alpha'(a_j) \smile \alpha'(a_k) <$ となる$\alpha'(a_i) \smile \alpha'(a_j) <$ となる$\alpha'(a_i) \smile \alpha'(a_k)$ となる。

最後に、$a_i \smile a_j = a_i \smile a_k = a_j \smile a_k$ となる場合について考える。命題 2 より、$\alpha'(a_i) \smile \alpha'(a_j) = \sigma < p(\sigma)$ となる $= \alpha'(a_i) \smile \alpha'(a_k)$ となる。とくに、必要であれば a_i と a_j を入れ替えることにより、$a_i \smile \bar{a}_i < \bar{a}_i \smile \bar{a}_j$ かつ $a_j \smile \bar{a}_j < \bar{a}_i \smile \bar{a}_j$ と仮定してよい。すると、命題 7 により $b_i \smile b_j = \bar{b}_i \smile \bar{b}_j$ を得る。

対 $(\bar{a}_i,\bar{b}_i),(\bar{a}_j,\bar{b}_j)$ は極小であることから、$\bar{b}_i \smile \bar{b}_j = b_i \smile b_j \leqq b_i \smile b_k$ かつ $\bar{b}_i \smile \bar{b}_j = b_i \smile b_j \leqq b_j \smile b_k$ ととなる。ゆえに、$b_i \smile b_k = b_j \smile b_k$ を得る。 □

Claim 2. $|\mathfrak{K}_{T'_A}(M')| < |\mathfrak{K}_{T_A}(M)|$

Proof. $\{(\alpha'(a_i), b_i), (\alpha'(a_j), b_j), (\alpha'(a_k), b_k)\}$ を $\mathfrak{K}_{T'_A}(M')$ の要素とする。あるノード $x \in V(T_A)$ について、$\alpha'(a_i) \smile \alpha'(a_j) = \alpha'(a_j) \smile \alpha'(a_k) = \alpha'(a_k) \smile \alpha'(a_i) = \alpha'(x)$ のとき、$a_i \smile a_j = a_j \smile a_k = a_k \smile a_i = x$ が成り立つ。$\alpha'(a_i) \smile \alpha'(a_j) = \alpha'(a_j) \smile \alpha'(a_k) = \alpha'(a_k) \smile \alpha'(a_i) = \sigma$ のとき、$a_i \smile a_j = a_j \smile a_k = a_k \smile a_i = \bar{a}_i \smile \bar{a}_j$ が命題 7 により成り立つ。よって、$|\mathfrak{K}_{T'_A}(M')| \leqq |\mathfrak{K}_{T_A}(M)|$ となる。

さらに、$\{(\alpha'(\bar{a}_i), \bar{b}_i), (\alpha'(\bar{a}_j), \bar{b}_j), (\alpha'(\bar{a}_k), \bar{b}_k)\}$ が $\mathfrak{K}_{T'_A}(M')$ の要素ではないことから、この claim が満たされる。 □

$|\mathfrak{K}_S(M)|$ と $|\mathfrak{K}_T(M)|$ に関する帰納法により、埋め込み $\alpha : T_A \to \bar{T}_A$ と $\beta : T_B \to \bar{T}_B$ が存在し、$\bar{M}$ が alignable であり、$\mathfrak{K}_{\bar{T}_A}(\bar{M}) = \mathfrak{K}_{\bar{T}_B}(\bar{M}) = \emptyset$ となることが示せる (証明略)。

最後に、$\bar{M}$ が more-constrained であることを示す。$a_i \smile a_j < a_i \smile a_k$ かつ $(a_l, b_l) \in \bar{M}$ $l = i, j, k$ において、a_i, a_j, a_k のどの要素も他の要素の祖先ではないとする。$\bar{M}$ が、alignable マッピングであることから、$b_j \smile b_k = b_i \smile b_k$ が成り立つ。$b_i \smile b_j = b_i \smile b_k$ ならば、$\{(a_i b_i), (a_j, b_j), (a_k, b_k)\}$ は、$\mathfrak{K}_{\bar{T}_B}(\bar{M})$ の要素でなくてはならないが、これは矛盾である。 □

4. 木の結合アルゴリズム

定理 12 と定理 14 の証明過程は、木 T_A と T_B および、その alignable マッピング M が与えられたときに、結合木を構成するアルゴリズムを示している。

Step 1. 定理 14 の証明で言及した方法により M のすべての結節を解消する。

Step 2. 定理 12 の証明で言及した方法により結合木を構成する。

しかしながら、このアルゴリズムは解消しなくてもよい結節も解消してしまうという点において、効率がよくない。実際に解消すべき結節を**危険結節**(critical knot) といい、次のように定義する。

定義 17. 木 T_A に関するマッピング M の **危険結節**(critical knot) とは、次の条件を満たす 3 要素の集合 $\{(a_i, b_i), (a_j, b_j), (a_k, b_k)\} \subseteqq M$ である。

（1）a_i, a_j, a_k のいずれも、他の要素の祖先ではない

（2）$a_i \smile a_j = a_j \smile a_k = a_k \smile a_i$

（3）$b_i \smile b_j < b_i \smile b_k$

木 T_A に関する M の危険結節の集合を $\mathfrak{Ck}_{T_A}(M)$ と表記する。木 T_A に関しても同様に $\mathfrak{Ck}_{T_B}(M)$ を定義する。

命題 15. alignable マッピング M が、more-constrained であることと、$\mathfrak{Ck}_S(M) = \mathfrak{Ck}_T(M) = \emptyset$ であることは等価である。

Proof. more-constrained マッピングの定義から直ちに得られる。 □

よって、危険結節のみを対象としたアルゴリズムの概略は次のようになる。

Step 1. M の **極小**の危険結節を定理 14 の証明で言及した方法により解消する。

Step 2. $\mathfrak{C}k_S(M) = \emptyset$ または $\mathfrak{C}k_T(M) \neq \emptyset$ ならば **Step 1** へ戻る。

Step 3. 定理 12 の証明で言及した方法により結合木を構成する。

極小の危険結節の存在は、Step 2 を実行する際に必要であり、定理 14 の証明の Claim 1 を満たすためにも必要である。次の補題により、その存在が保証される。

補題 16. $\{(a_i, b_i), (a_j, b_j), (a_k, b_k)\}$ を木 T_A に関する危険結節とする。このとき、次の条件を満たす危険結節 $\{(\bar{a}_i, \bar{b}_i), (\bar{a}_j, \bar{b}_j), (\bar{a}_k, \bar{b}_k)\}$ が存在する。

（1）$a_i \smile a_j = \bar{a}_i \smile \bar{a}_j$

（2）対 $((\bar{a}_i, \bar{b}_i), (\bar{a}_j, \bar{b}_j))$ は、T_A に関して極小

Proof. $b_i \smile b_j < b_i \smile b_k$ が成り立つとする。また、一般性を失うことなく $\{(a_i, b_i), (a_j, b_j), (a_k, b_k)\}$ が極小であると仮定してよい。すなわち、$b'_i \smile b'_j \leqq b_i \smile b_k$ ならば、$b'_i \smile b'_j = b_1 \smile b_3$ となる。ここで、$\{(a'_i, b'_i), (a'_j, b'_j), (a'_k, b'_k)\}$ は、$a'_i \smile a'_j = a_i \smile a_j$ かつ $b'_i \smile b'_j < b'_i \smile t'_k$ であるような別の危険結節とする。

$(\bar{a}_i, \bar{b}_i), (\bar{a}_j, \bar{b}_j)$ を $\bar{a}_i \smile \bar{a}_j = a_i \smile a_j$ かつ $\bar{b}_i \smile \bar{b}_j < b_i \smile b_j$ であるような M 要素とする。

補題 1 により、$\bar{b}_i \smile b_i = \bar{b}_j \smile b_i = b_i \smile b_j$ と仮定してよい。とくに、$\bar{b}_i \smile \bar{b}_j < \bar{b}_i \smile b_i$ である。M が alignable マッピングであることから、$\bar{a}_i \smile a_i = \bar{a}_j \smile a_i$ である。一方、$\bar{a}_i \smile a_i \leqq \bar{a}_i \smile \bar{a}_j = a_i \smile a_j$ より、$\bar{a}_i \smile \bar{a}_j = \bar{a}_i \smile a_i = \bar{a}_j \smile a_i$ を得る。よって、$\{(\bar{a}_i, \bar{b}_i), (\bar{a}_j, \bar{b}_j), (a_i, b_i)\}$ は、危険結節であり、そのような $(\bar{a}_i, \bar{b}_i), (\bar{a}_j, \bar{b}_j)$ は存在しない。なぜなら、$\{(a_i, b_i), (a_j, b_j), (a_k, b_k)\}$ が極小であるからである。ゆえに、$(a_i, b_i), (a_j, \bar{b}_j)$ は T_A に関して極小である。 □

定理 14 の Claim 1 により、極小の危険結節を解消しても、alignable マッピングの性質は保たれることが保証されている。これに対して、極小な危険結節の解消手続きが、全体の危険結節の数を減らすとは限らず、逆に新たな危険結節を生成する可能性がある。よって、アルゴリズムの停止性は、危険結節の数による帰納法では証明できない。しかし、危険結節の解消手続きが、全体の結節の数を減らすことが、補題 16 と定理 14 の Claim 2 により保証される。よって、本アルゴリズムの停止性は、結節の数に関する帰納法により証明できることがわかる。

また、結合木は一般に一意には決まらない。そのため、どのような木を結合木として出力するかについては、応用領域に応じた評価関数の導入などが必要であり、計算量もこの部分に依存する。木の結合に必要な基本的な計算量は、マッピングに含まれる要素の数と最小共通祖先の計算量により決定される。

5. むすび

本稿では、2 つの木構造の近似照合の結果を用いて、2 つの木を 1 つに結合するための一般的なフレームワークを提案した。このフレームワークは、半構造データを木構造データとしてとらえ、木の編集距離による近似照合から、木を結合する。実際に、様々な近似照合のクラスから、木を結合できることを構成的証明により示した。今後、適用領域を定め、本手法を用いた具体的なアルゴリズムを提案することにより、本手法の有効性を示す予定である。

文　献

[1] R. L. Fontaine. Merging xml files: a new approach providing intelligent merge of xml data sets. In *Proc. of XML Europe 2002*, 2002.

[2] T. Lindholm. A 3-way merging algorithm for synchronizing ordered trees — the 3dm merging and differencing tool for xml. Master's thesis, Helsinki University of Technology, Dept. of Computer Science, 2001.

[3] K. Tufte and D. Maier. Aggregation and accumulation of xml data. *IEEE Data Engineering Bulletin*, 24(2):34–39, 2001.

[4] K. Tajima P. Buneman, S. Khanna and W.-C. Tan. Archiving scientific data. In *Proc. of the 2002 ACM SIGMOD international conference on Management of data*, pages 1–12, 2002.

[5] J.T.-L. Wang and K. Zhang. Finding similar consensus between trees: an algorithm and a distance hierarchy. *Pattern Recognition*, 34:127–137, 2001.

[6] G. Valiente. Tree edit distance and common subtrees. Technical Report LSI-02-20-R, Universitat Politecnica de Catalunya, Barcelona, Spain, 2002.

[7] T. Kuboyama, S. Kilho, and T. Miyahara. A theoretical analysis of tree edit distance measures. *Information Processing Society of Japan, Transactions on Mathematical Modeling and Its Applications*, 2005.

[8] K. Zhang and D. Shasha. Simple fast algorithms for the editing distance between trees and related problems. *SIAM Journal on Computing*, 18(6):1245–1262, December 1989.

[9] K. Zhang and T. Jiang. Some max snp-hard results concerning unordered labeled trees. *Information Processing Letters*, 49:249–254, 1994.

[10] T. Jiang, L. Wang, and K. Zhang. Alignment of trees — an alternative to tree edit. *Theoretical Computer Science*, 143:137–148, 1995.

[11] C. L. Lu, Z.-Y. Su, and G. Y. Tang. A new measure of edit distance between labeled trees. *LNCS*, 2108:pp. 338–348, 2001. COCOON 2001.

[12] K. Zhang. A constrained edit distance between unordered labeled trees. *Algorithmica*, 15:205–222, 1996.