解　説

# 構造をもつものの距離と類似度†

田　中　栄　一††

## 1. まえがき

長さが必ずしも等しくない二つの文字列を比較する問題は，遺伝子を比較して系統発生図を推定する問題，符号理論，音声認識などの分野から出てきたものであるが，「1960 年代から 1970 年代初期にかけておよそ1ダースの論文が独立に発表され，それらは似たような問題の定義をし，似たような結果を得ていた」[1] 科学史上でも珍しい出来事であった．やがて，文字列比較から，図形，木，グラフを対象とするように拡張されてきた．これら一群の構造をもつものを比較する研究の発端は，上記の生物学，計算機科学，音声認識などで解決を迫られた具体的な問題から始まっているが，少し広い立場からみると，次のような問題の中に位置づけられる．いま比較する文字列，図形，木，グラフなどを対象と呼ぶことにし，対象 $A$ が対象 $B$ の部分構造であるとき $A \subset B$，$A$ と $B$ の距離を $D(A, B)$，類似度を $S(A, B)$ と書くことにする．

（1） $A$ と $B$ は等しいか（$A=B$ か）．グラフの同型判定問題 (graph isomorphism)[2] はこの問題の代表的なものである．

（2） $A$ は $B$ の一部分か（$A \subset B$ か）．ある綴りがテキストにあるかどうかの探索 (string search)[3]，高次元対象の探索問題[4]，部分木判定問題 (subtree isomorphism)[5]，部分グラフ判定問題 (subgraph isomorphism)[6]．

（3） $A$ と $B$ の（最大）共通部分を列挙する（$A \supset a$, $B \supset b$, かつ $a=b$ となる $a$，$b$ を求める）．最長共通文字列の探索 (longest common subsequence)[7]，複数の化合物の共通構造を列挙する[8]．

これらの問題に近似の概念が入ると，問題は次のようになる．

（4） $A$ は $B$ にどの程度似ているか，あるいは似ていないか（$S(A, B)$, $D(A, B)$ の定義）．系列間距離[9]，図形間距離[10]，木の間の距離[11]，グラフ間距離[12]．

（5） $B$ の中にある $A$ と似ている部分を列挙する（$B \supset b$, $D(A, b) \leqq k$（$k$ は非負）となる $b$ を列挙する）．綴り $A$ と似た綴りをテキストから探し列挙する[13]．グラフ $A$ と似た構造をグラフ $B$ から探し列挙する[14]．

（6） $A$ と $B$ に類似構造があれば列挙する（$A \supset a$, $B \supset b$, $D(a, b) \leqq k$ となる $a$，$b$ を列挙する）．複数の化合物の類似構造の列挙[15]．

文献は多いので例をあげるにとどめた．（4）～（6）は明らかに（1）～（3）の一般化になっている．（4）は（5），（6）を考えるときの基礎になるので，本文では主に（4）について述べることにする．

次に距離と類似度についての考え方を整理しておこう．距離関数 $D$ は次の公理を満たす．

D1. $D(A, B) \geqq 0$, $A=B$ のときに限り符号が成立する．

D2. $D(A, B)=D(B, A)$.

D3. $D(A, B)+D(B, C) \geqq D(A, C)$.

最近，D2 が成立しない場合も習慣的に距離と呼ぶことがある．ちなみに交通網でも一方通行があるところでは D2 は成立しない．ところで，距離公理は空間の性質がいたるところで等しい，すなわち一様であるときに満たされるが，局所的に性質が異なっている空間では成立しない．たとえば，文字パターンの作る空間は意味を考えると一様ではない．文字が2次元のユークリッド空間上に書かれているとして，太と犬の「丶」は，次の「丶」とは異なる．前者の「丶」は意味があるが，後者のそれは雑音であるから，性質が一様であるユークリッド空間上に書かれたものも，文字パターンとなると空間の性質は一様ではなくなる．伸縮整合でペナルティ関数を用いたり，文脈に依存する測度を考え局所を強調したりすると距離関数を導入できなくなる．そこで，このような対象に対して定義される測度 $M(A, B)$ は距離公理を満たさないことが多い．しかし，その測度がなんらかの意味で有用であれ

† Structural Distances and Similarities by Eiichi TANAKA (Department of Electronics, Faculty of Engineering, Kobe University).

†† 神戸大学工学部電子工学科

ば，類似度と呼ぶことが多い．類似度関数$S$が満たすべき公理として，次のものがあげられている[16]．

S1. $S(A,B)=S(B,A)$.

S2. $S(A,A)\leqq S(A,B)$.

あるいは

S2′. $S(A,A)\geqq S(A,B)$.

S2のとき類似度は前向き (forward) である，S2′のとき後向き (backward) であるという．S1もD2と似た事情にある．類似度のもつ数学的条件はまことにゆるい．それぞれの類似度の定義が個々の問題にとってどのような意味があるかが重要である．**2.** で1次元の文字列，**3.** で2次元の文字列，すなわち図形，**4.** で木，**5.** でグラフについて述べることにする．

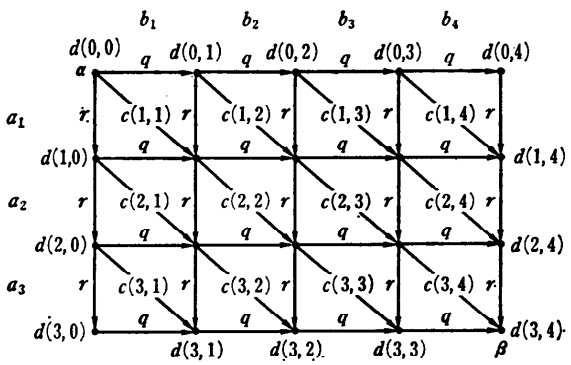
## 2. 1次元記号列間の距離と類似度

### 2.1 1次元記号列間の距離

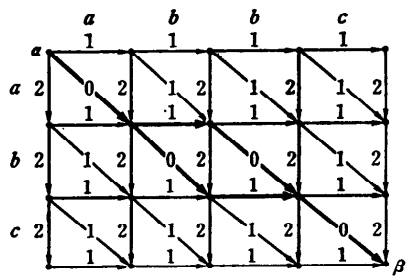
いま，$T$を記号の集合，$T$の記号からできる二つの記号列を $A=a_1a_2\cdots a_m$, $B=b_1b_2\cdots b_n$ とする．$m=n$ かつ $T=\{0,1\}$ のとき，$A$，$B$間の距離としてのハミング距離はよく知られている．Levenshtein[17]は，必ずしも $m=n$ とは限らない場合，ビット誤り（置換）の他に，不用なビットの挿入やビットの脱落も起こるとして，$A$から$B$へ変換するのに要する置換，挿入，脱落操作の最少数を$A$，$B$間の距離と定義した．このような距離についての報告は多い[9),19)~22)]．$T$が2記号以上でも，また置換，挿入，脱落のコスト（それを$p$，$q$，$r$とする）が変わっていても本質的には変わらないので，ここではそのような一般化をした場合[22)]の距離をレーベンシュタイン距離（略してLD）と呼ぶことにする．$A$，$B$間のLD $D(A,B)$は，$a_i=b_j$ のとき $c(i,j)=0$, $a_i\neq b_j$ のとき $c(i,j)=p$ として次の手順で計算できる．

アルゴリズム 1

```
begin
d(0,0) :=0;
{長さ 1,2,…,m の列 a1, a1a2, …, a1…am を
 脱落変換のみで空列にする際のコスト}
for i=1 to m do d(i,0) :=d(i-1, 0)+r;
{空列より長さ 1,2,…,n の列 b1, b1b2, …, b1
…bn を挿入変換のみで生成する際のコスト}
for j=1 to n do d(0,j) :=d(0,j-1)+q;
 for i=1 to m do
  for j=1 to n do
{a1…ai を b1…bj に変換するコスト d(i,j) を求める}
   begin
{a1…ai-1 を b1…bj-1 に変換するコストと ai を bj
に置換するコストの合計}
   d1 :=d(i-1,j-1)+c(i,j);
{a1…ai-1 を b1…bj に変換するコストと ai が脱落す
るコストの合計}
   d2 :=d(i-1,j)+r;
{a1…ai を b1…bj-1 に変換するコストと bj を挿入す
るコストの合計}
   d3 :=d(i,j-1)+q;
{最小コストの変換を採用}
   d(i,j) :=min (d1,d2,d3);
   end;
{a1…am を b1…bn に変換するコスト}
D(A,B) :=d(m,n);
end
```

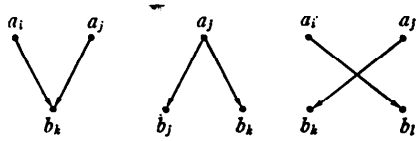



(a) $D(a_1a_2a_3, b_1b_2b_3b_4)$ を計算する図




(b) $D(abc, abbc)$ を計算する図

図-1




(a) $i<j$ (b) $j<k$ (c) $i<j$ かつ $k<l$

**図-2** 禁止された写像

図-1(a)のダイアグラムの辺に与えられたコストを辺の距離とみたとき，アルゴリズム1は$\alpha$から$\beta$に至る最短距離を求めることと同じである．$p=q=1$, $r=2$, $A=abc$, $B=abbc$ のときの例を図-1(b)に示す．太線の経路が最短距離になるから，$B=abbc$ の第2文字あるいは第3文字が挿入したと考えればよい．この計算の時間複雑さは $O_T(mn)$，空間複雑さは $O_S(mn)$ であるが，後者は工夫して $O_S(m)$ とすることができる．LDは，$q\neq r$ のとき，すなわち挿入と脱落のコストが等しくないときは，必ずしも $D(A,B)=D(B,A)$ とはならない．LDの定義の背後には次の条件がある．$a_i$ が $b_j$ に置換したとき，$a_i=b_j$ あるいは $a_i\neq b_j$ にかかわらず，$a_i$ が $b_j$ に写像していると考えると，$A$から$B$への一つの変換は$A$から$B$への一つの写像でもある．写像していない$A$の記号は脱落したものであり，写像されていない$B$の記号は挿入したものである．いま，$A$から$B$への写像で，$a_i$ が $b_j$ に写像しているとき $(i,j)$ と書き，$(i,j)$ の集合 $Ms$ は次の条件を満たしているとする．

$(i_1,j_1), (i_2,j_2)\in Ms$ について

(1) $i_1=i_2$ iff $j_1=j_2$,

(2) $i_1<i_2$ iff $j_1<j_2$.

この条件は，図-2のような多対1写像，1対多写像，交鎖写像を禁止している．このような制限は，どのような構造の変形を許すかを明らかにしているので重要である．LDはこのような写像の下での距離である．タイプの打鍵誤りの訂正のために隣接記号間の交鎖写像を許す変換も考えられている[23]．また，記号が属性をもっている場合の距離もある[24]．

### 2.2 1次元記号列間の類似度

音声では母音の発声が長くなったり短くなったりすることがある．このような変形に影響されにくい測度に伸縮整合 (elastic matching) による類似度がある[25]．二つの音声パターンの時間標本化を行い，その特徴ベクトルの系列を $A=a_1a_2\cdots a_m$, $B=b_1b_2\cdots b_n$ とする．

$$c(a_i,b_j)=|a_i-b_j|$$

とする．伸縮整合による類似度 $E(A,B)$ は次の手順で計算できる．

アルゴリズム 2

```
begin
e(1,1) := c(a1, b1);
for i=1 to m do
  for j=1 to n do
    begin
      e1 := e(i-1, j)+c(ai, bj);
      e2 := e(i-1, j-1)+2c(ai, bj);
      e3 := e(i, j-1)+2c(ai, bj);
      e(i, j) := min (e1, e2, e3);
    end;
  E(A, B) := e(m, n);
end
```

伸縮整合にはいろいろな変形がある[26]．$a_i, b_j$ を特徴ベクトルではなく，記号 $a_i, b_j$ のとき，LDでのコスト関数 $c(a_i,b_j)$ を用いると記号列の伸縮整合になる．伸縮整合による測度は距離公理を満足しない[19]．

LDでは，記号列の記号間に相関がないものと考えていた．しかし音声のように調音のために発音が前後の発音の影響を受ける場合もある．たとえば，無声破裂音に挟まれた [ə] はしばしば消失するために [m∧lt ə play] は [m∧ltplay] と発音される．これを $\lambda$ を空記号として，

[ə]→$\lambda$/[無声破裂音]_[無声破裂音]

と書く．このような状況を反映させた測度を作るために，$a_i$ が $b_j$ に置換するのは $a_i$ の文脈が $K(a_i)$ (たとえば，$a_{i-1}\_a_{i+1}$) のときであるとし，そのコスト関数を $c(a_i,b_j:K(a_i))$ と書き，これを文脈依存コスト関数と呼ぶことにする．文脈依存コスト関数を用いて求めた系列間の測度を文脈依存類似度という[27]．たとえば，$\alpha_i$, $\beta_i$ などを系列として，$c(a_i,b_j:\alpha_ia_i\beta_i)$ $c(a_i,\lambda:\alpha_i'a_i\beta_i')$, $c(\lambda,b_j:\alpha_i''a_i\beta_i'')$ を定めると，これらは文脈下での置換，脱落，挿入のコスト関数である．$A$から$B$への文脈依存類似度 $S(A,B)$ は次の手順で計算できる．

アルゴリズム 3

```
begin
s(0, 0) := 0;
for i=1 to m do
  s(i, 0) := s(i-1, 0)+c(ai, λ : αiaiβi);
for j=1 to n do
  s(0, j) := s(0, j-1)+c(λ, bj : λa1a2···);
```

```
for i=1 to m do
  for j=1 to n do
    begin
      s1 :=s(i-1, j)+c(a_i, λ : α_i a_i β_i);
      s2 :=s(i-1, j-1)+c(a_i, b_j : α_i' a_i β_i');
      s3 :=s(i, j-1)+c(λ, b_j : α_i'' λ β_i'');
      s(i, j) :=min {s1, s2, s3} ;
    end;
```

文脈類似度を用いると，LD では分類できなかった系列集合の分類ができるようになることがある．

## 3. 図形間の距離と類似度

図形間の距離や類似度にはパターン認識の立場からいろいろなものがあるが，本文では前章の記号列間の距離と類似度の延長線にあるものについて述べる．

### 3.1 図形間の距離

1次元記号列間の距離は2次元記号列，すなわち図形，の間の距離に拡張できる[10),28)]．図形 $A$ の位置 $(i, j)$ の画素を $a(i, j)$ とし，$a(i, \bar{j})$, $a(\bar{i}, j)$ を次のように定義する (図-3)．

$$a(i, \bar{j})=a(i, 1)a(i, 2)\cdots a(i, j),$$
$$a(\bar{i}, j)=a(1, j)a(2, j)\cdots a(i, j).$$

$A(I, J)$ は大きさ $I*J$ の図形を表すものとし，二つの図形 $A(I, J)$, $B(M, N)$ を考える．まぎらわしくないときは $A(I, J)$ を単に $A$ と書く．$A$ の画素 $a(i, j)$ から $B$ の画素 $b(m, n)$ への写像を $(i, j, m, n)$ と書くと．$A$ から $B$ への写像は $(i, j, m, n)$ の集合 $M_P$ で表されるが，$M_P$ は次のような条件を満たしているものとする．

$(i_1, j_1, m_1, n_1)$, $(i_2, j_2, m_2, n_2)\in M_P$ について

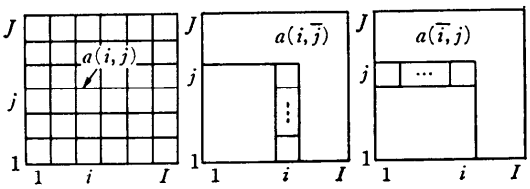
(1) $i_1=i_2$ & $j_1=j_2$ iff $m_1=m_2$ & $n_1=n_2$,

(2) $i_1<i_2$ & $j_1=j_2$ iff $m_1<m_2$ & $n_1=n_2$,

(3) $i_1=i_2$ & $j_1<j_2$ iff $m_1=m_2$ & $n_1<n_2$,

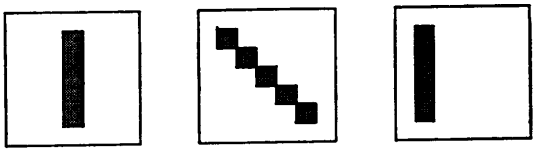
(4) $i_1<i_2$ & $j_1<j_2$ iff $m_1<m_2$ & $n_1<n_2$.

$A$ から $B$ への距離 $D(A, B)$ は次式を回帰的に用いて計算できる．




(a) $a(i, j)$ (b) $a(i, \bar{j})$ (c) $a(\bar{i}, j)$

図-3



(a) A (b) B (c) C

図-4

$$\bar{D}(i, j, m, n)$$
$$=\min\{\bar{D}(i-1, j, m, n)+j*r,$$
$$\bar{D}(i, j-1, m, n)+i*r,$$
$$\bar{D}(i, j, m-1, n)+n*q,$$
$$\bar{D}(i, j, m, n-1)+m*q,$$
$$\bar{D}(i-1, j, m-1, n)+d(a(i, \bar{j}), b(m, \bar{n})),$$
$$\bar{D}(i, j-1, m, n-1)+d(a(\bar{i}, j), b(\bar{m}, n))\}.$$

ここで，

(i) $\bar{D}(i, j, m, n)=i*j*r$ $(m=n=0$ のとき$)$,
$\bar{D}(i, j, m, n)=m*n*q$ $(i=j=0$ のとき$)$,

(ii) LD の計算 $d(a(\bar{i}, j), b(\bar{m}, n))$, $d(a(i, \bar{j}), b(m, \bar{n}))$ において，画素 $a(i, j)$ が画素 $b(m, n)$ に変換するときのコスト $c(i, j, m, n)$ は

$$c(i, j, m, n)=\begin{cases} o, & a(i, j)=b(m, n) \text{ のとき} \\ p, & a(i, j)\neq b(m, n) \text{ のとき} \end{cases}$$

と表す．

このとき，

$$D(A, B)=\bar{D}(I, J, M, N)$$

である．この距離は LD の自然な拡張になっている．計算の複雑さは，少し工夫すると，$O_T(IJMN)$, $O_S(IJMN)$ とすることができる．

### 3.2 図形間の類似度

1次元ベクトル列間の伸縮整合類似度[25)]を2次元に拡張しようとしたものに磯道・小川[29)]がある．ここでは，伸縮の度合によるペナルティを考慮している．パターン認識の立場からみると，図-4の三つの図形間の距離が $D(A, B)\neq 0$ かつ $D(A, C)=0$ であることが望ましい．すなわち，異なった図形間の距離は零ではなく，平行移動の関係にある図形間の距離は零にしたい．しかし，3.1 で述べた距離では，いかにコストを調整しても，$D(A, C)=0$ としようとすると，$D(A, B)=0$ となってしまう．これを実現するためには，1次元の場合と同様に文脈依存コスト関数を作って，文脈依存類似度を定義すればよい[30)]．

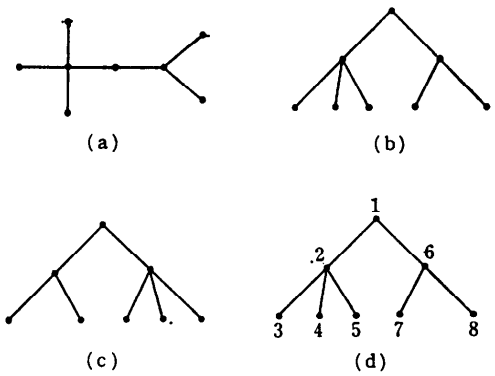



図-5 木

## 4. 木の間の距離

木は閉路のないグラフのことであるが，根のない木，さらに，根があり節点に順序のついた木などを考えることもある．図-5 の(a)～(d)はいずれも閉路のないグラフであるが，根のある木では(b)と(c)は同じ木を表し，根があり節点に順序のついた木(d)では(b)と(c)は異なった木になる．

### 4.1 節点の置換，挿入，脱落操作による距離

根があり，節点に順序とラベルがついている木を考える．いま節点の順序はプリオーダとする(図-5(d))．木 $T_A$ の節点 $a_i$ から木 $T_B$ の節点 $b_j$ への写像があるとき $(i, j)$ と書くと，$T_A$ から $T_B$ への写像は $(i, j)$ の集合 $M_t$ で表される．$M_t$ は次の条件を満たしているものとする．

$(i_1, j_1), (i_2, j_2) \in M_t$ について，

(i) $i_1 = i_2$ iff $j_1 = j_2$,

(ii) $i_1 < j_2$ iff $j_1 < j_2$,

(iii) $j_1$ が $j_2$ の祖先のとき，かつそのときにのみ $i_1$ は $i_2$ の祖先である．

この写像を Tai 写像と呼び，Tai 写像の下で定義された木の間の距離を Tai 距離と呼ぶことにする[20]．$N_A$, $D_A$, $L_A$ を $T_A$ の節点数，深さ，葉の数とすると，計算手数が $O_T(N_AN_BD_AD_B)$, $O_S(N_AN_BD_AD_B)$ の下降形計算法[31]，$O_T(N_AN_BL_AL_B)$, $O_S(N_AN_BL_AL_B)$ の上昇形計算法がある[32]．

いま，$T_A$ から $T_B$ への写像 $M_t$ があるとする．$T_3$, $T_4$ はそれぞれ $T_1$, $T_2$ の像を覆う最小の部分木とする．Tai 写像では $T_1$ と $T_2$ が分離していても，$T_3$ と $T_4$ が分離しているとは限らない (図-6(c))．この分離条件を満たす写像を考える．いま，$i$ を根とする部分木 $T(i)$ で最も大きい順序数をもつ節点を $el(i)$ と書く．また，$T(i)$ の像を覆いそれ以外の像をもたない最小の部分木の根を $Ri$ と書く．Tai 写像の条件(iii)を次のように書き換えたもの

(iii)′ $i_1$, $i_2$ に対して，$Ri_1$, $Ri_2$ が定まるとき

$el(i_1) < i_2$ iff $el(Ri_1) < Ri_2$

を構造を保存する写像 (structure preserving mapping: SPM) といい，この写像に基づく距離を構造を保存する写像に基づく距離 (SPM 距離と略す) という．写像と逆写像が共に SPM であるとき，強構造保存写像 (strongly structure preserving mapping: SSPM) という[33]．SSPM に基づく距離も定義できる[34),35]．SPM 距離の計算の複雑さは $O_T(N_AN_B^2)$,

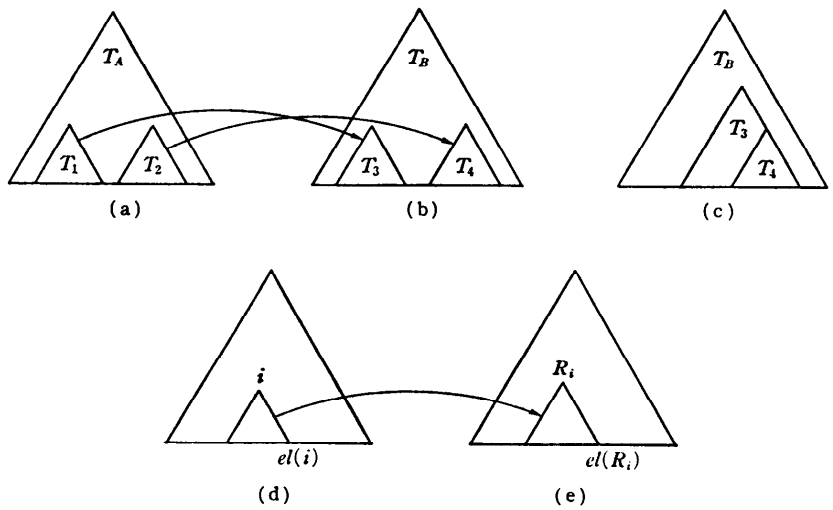



図-6 木の間の写像

$O_S(N_AN_B)$, SSPM 距離のそれは $O_T(N_AN_B)$, $O_S(N_AN_B)$ である．根から節点への距離を節点のレベルとし，同じレベルの節点を根とする部分木を比較して距離を計算するものもある[36]．SPM は節点の挿入に対して，逆写像が SPM のときは節点の脱落に対して自然な写像である．そこで，SPM であるか，あるいは逆写像が SPM であるかのいずれかが成立しているときの測度を考えると，三角公理を満たさない．これを弱構造保存写像に基づく類似度というが，文献 37) の測度は弱構造保存写像に基づく類似度が距離になる特殊な場合である[32]．

### 4.2 節点の挿入，脱落，分割，融合操作に基づく距離

節点の挿入，脱落，分割，融合操作を $\alpha_1$, $\beta_1$, $\alpha_2$, $\beta_2$ とする．たとえば図-7 では，$\alpha_1$, $\beta_1$ は $f$ の挿入，脱落，$\alpha_2$ は $b$ の $b$ と $b'$ への分割，$\beta_2$ は $b$ と $b'$ の融合を表す．この木の変換に要する操作の最小数で木の間の距離を定義したものもある[38]．計算の複雑さは $O_T(N_AN_B^2)$ である．この木の距離の性質についてはよく分かっていない．この距離を波形の相関を求める問題に応用している[39]．図-8(a)の二つの波形を木 $T_1$, $T_2$ で表すと，木は波のおよその形を表している．このような木の間の葉の対応を求めると図-8(b)のような波形の対応が得られる．図-8(c)は実在の波形に適用した例で，地震波の伝播などを調べるのに用いられている．このほかに，5つの操作を基に誤り訂正木オートマトンで木と木言語の距離を計算するものがある[40]．

### 4.3 その他の木の距離

根はあるが節点に順序がついていない木についての距離も考えられている[41]．また2分木について，隣接する部分木の交換操作を基に定義した木の間の距離もある[42]．図-9(a)の $T_1$ を $T_2$ あるいは $T_3$ に変換する操作を枝 $e$ に関する最近接交換 (nearest neighbour interchange : nni) と呼び，$T_A$ から $T_B$ を得るのに必要な最少の nni

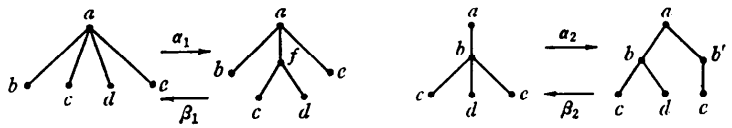
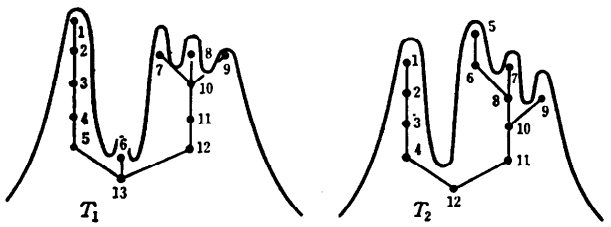
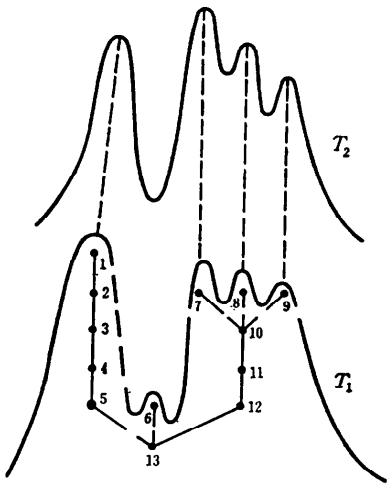


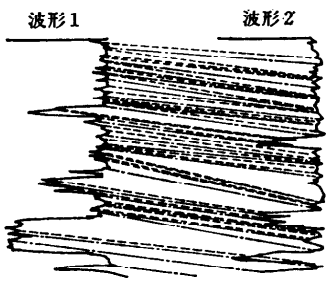

図-7 木の変換操作




(a) 波形を木で表す

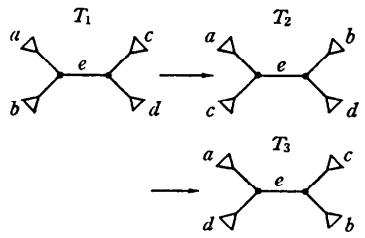



(b) 波形の対応

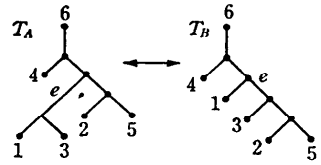



(c) 波形の対応

図-8 波形と木




(a) 最近接交換 (△は部分木)




(b) 最近接交換の例

**図-9**

の回数を $T_A$ から $T_B$ への距離とする. **図-9**(b)にnni の例を示す. この距離は進化の系統樹推定に関するもので報告も多い[43].

## 5. グラフ間の距離

グラフ間の距離に関する報告は多くが属性関係グラフ (attributed relational graph) に関するものである. $V_N$ を節点の集合, $V_B$ を枝の集合とし, $V_N$ あるいは $V_B$ の要素は $(s, x)$ で表す. ここで, $s$ は構文記号, $x=(x_1, x_2, \cdots, x_n)$ は $s$ に付随する属性からなる意味ベクトルである. $V=V_N \cup V_B$ 上の属性関係グラフは $G=(N, B, \mu, \varepsilon)$ で表される.

(1) $N$ は節点の有限集合

(2) $B \subseteq N \times N$ は枝の集合

(3) $\mu: N \to V_N$ は節点のラベル関数

(4) $\varepsilon: B \to V_B$ は枝のラベル関数

たとえば, **表-1** のような構文記号を用いて**図-10**(a)の台形は**図-10**(b)のようなグラフで表せる. ここで, $(c, 106)$ は構文記号 $c$ の属性は角度が 106° であり, $(E, 5)$ は辺の長さが 5 であることを示す[12].

グラフ間の距離の研究はグラフの同型判定問題の拡張でもある. グラフの構造は変わらないがラベルが異なっている二つのグラフの同型判定[44], 属性関係グラフ上での同型・準同型の判定[45], 文献 5) の部分同型判定への拡張[14]などがある. グラフ間の距離を正面から取り扱おうとしたものに, グラフ文法に基づくもの[47], 節点や枝の置換, 挿入, 脱落操作を基に定義したもの[12], 有向グラフ間の距離を考えたもの[48),49] などがある. ここでは **2.**～**4.** の内容と関係が深い文献 12) の定義を述べることにする. 節点 $v$ に対して $dn(v)$, $in(v)$ を $v$ の脱落のコスト, 挿入のコストとし, 節点 $v_1$ が $v_2$ に置換するコストを $sn(v_1, v_2)$ とする. 同様に枝の置換のコストを $sb(b_1, b_2)$ とする. $db(b)$, $ib(b)$ は $b$ の脱落, 挿入のとき $b$ の両端の節点の脱落や挿入はないときのコスト, $db'(b)$, $ib'(b)$ は $b$ の脱落, 挿入のとき少なくとも $b$ の両端の節点の一つは脱落, 挿入するときのコストとする. 二つの節点を $s=(\sigma, (x_1, x_2, \cdots, x_n))$, $t=(\tau, (y_1, y_2, \cdots, y_n))$ とし, $q_i$ を重み関数とするとき, 構文記号と意味ベクトルの両者を考慮した節点の置換のコストを,

$$sn(\sigma, \tau)+\sum_{i}^{n} q_i |x_i - y_i|$$

とする. 二つの属性関係グラフを $G_A=(N_A, B_A, \mu_A, \varepsilon_A)$, $G_B=(N_B, B_B, \mu_B, \varepsilon_B)$ とする. \$ を特殊な記号とし,

$\overline{N}_A = N_A \cup \{\$\}$, $\overline{N}_B = N_B \cup \{\$\}$, $\$ \notin N_A \cup N_B$.

のとき, 写像 $f: \overline{N}_A \to \overline{N}_B$ を次のように定める.

(1) $f(\$) \neq \$$

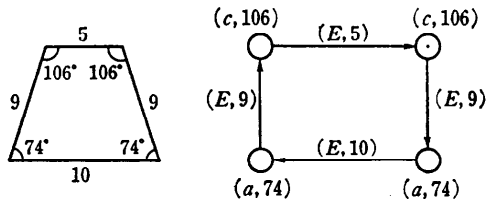
(2) 任意の $n, n' \in N_A$, $f(n)$ と $f(n') \in N_B$ に対して $n \neq n' \to f(n) \neq f(n')$.

(3) 任意の $n_2 \in N_B$ に対して, $f(n_1)=n_2$ となる $n_1 \in \overline{N}_A$ が存在し, 任意の $n_1 \in N_A$ に対して, $f(n_1)=n_2$ となる $n_2 \in \overline{N}_B$ が存在する.

$f(n_1)=n_2 \in N_B$ のとき節点 $n_1$ は節点 $n_2$ に置換され, $f(n_1)=\$$ のとき節点 $n_1$ は脱落し, $f^{-1}(n_2)=\$$ のとき

**表-1**

| 構文記号 | 意 味 |
|---|---|
| a | $\alpha<90°$ |
| b | $\alpha=90°$ |
| c | $90°<\alpha<180°$ |
| d | $180°<\alpha<270°$ |
| e | $\alpha=270°$ |
| f | $270°<\alpha<360°$ |




(a) 図A (b) 図Aのグラフ表現

**図-10** 図のグラフ表現

節点 $n_2$ は挿入されたことになる．写像 $f$ に対して定まるコストを cost ($f$)，$G_A$ から $G_B$ への可能な写像を $f_1, f_2, \cdots, f_m$ とする．最適近似整合 $f*$ は，次のようなコストが最小の写像である．

$$\text{cost}(f*)=\min_{f_1 \cdots f_m}\{\text{cost}(f)\}.$$

次の条件が満足されるとき，cost $(G_A, G_B)$=cost $(f*)$ は距離になる．

（1） 全ての構文記号 $x$ に対して，$sn(x,x)=sb(x,x)=0$. $x\neq y$ のとき，$sn(x,y)>0$, $sb(x,y)>0$.
挿入，脱落のコストは全て正である．

（2） 任意の $x$，$y$ に対して，$sn(x,y)=sn(y,x)$, $dn(x)=in(x)$, $db'(x)=ib'(x)$. 枝に対しても同様である．

（3） 任意の三つの変換 $t_1$, $t_2$, $t_3$ のコストが $c_1$, $c_2$, $c_3$ で，$t_2$ と $t_3$ の合成が $t_1$ であるとき，$c_1 \leqq c_2+c_3$ が成立する．

距離の計算は状態空間探索法で求める．次の条件をもつ部分集合 $S \subseteq \bar{N}_A \times \bar{N}_B$ は一つの状態を表す．

$$(x,y)\in S \Rightarrow f_s(x)=y.$$

初期状態は $S=\phi$，最終状態は $N_A$ と $N_B$ の要素が全て現れた状態である．状態 $S$ を展開して新しい状態 $S'$ を作る．$S=\{(x_1,y_1),\cdots,(x_n,y_n)\}$ の展開とは，$S$ に新しい $(x,y)\in N_A\times N_B$ を加えることである．ここで $x\notin\{x_1,\cdots,x_n\}-\{\$\}$, $y\notin\{y_1,\cdots,y_n\}-\{\$\}$, $(x,y)\neq(\$,\$)$. 図-12 は図-11 の二つのグラフに関する状態展開である．丸印の数字はコストで，コスト関数は次のように定めている．

$$dn(x)=2,\quad db(x)=2,\quad ib(x)=2,$$
$$sn(x,y)=sn(y,x)=1,\quad db'(x)=1.$$

上記の距離の定義の基になっている写像は文字列，図形，木で考慮されていたトポロジが入っていない．グラフ間の写像にトポロジを入れようとした例[15]をみよう．$G_A$ から $G_B$ への写像 $M_Q$ は $(i,j)$ を要素とする写像である．ここで，$i$ は $G_A$ の節点，$j$ は $G_B$ の節点である．このとき，$M_Q$ は次の条件を満たすものとする．

（1） $(i_1,j_1),(i_2,j_2)\in M_Q$ について
$i_1=i_2$ iff $j_1=j_2$.

（2） $(i_1,j_1),(i_2,j_2),(i_3,j_3)\in M_Q$ について
$i_1$, $i_2$, $i_3$ を結ぶ道があり，$j_1$, $j_2$, $j_3$ を結ぶ道があるとき，
$i_2\in P(i_1,i_3)$ iff $j_3\in P(j_1,j_3)$.

ここで，$P(i_1,i_3)$ は $i_1$ から $i_3$ に至る全ての道にある全ての節点の集合である．
この写像は，$G_A$ を張る木 $T_A$ から $G_B$ を張る木 $T_B$ への写像を考えていることになる．

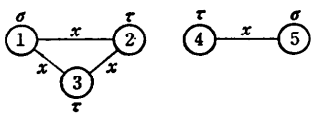



図-11 二つのグラフ

## 6. むすび

構造をもつものの距離や類似度は誤り訂正構文解析とも関係がある．文法 $G$ で生成される言語を $L(G)$ とし，文 $x$ に誤りがあるために $x\notin L(G)$ であるとする．$x$ に最も近い文 $y$ を $L(G)$ から選び，$y$ の構文解

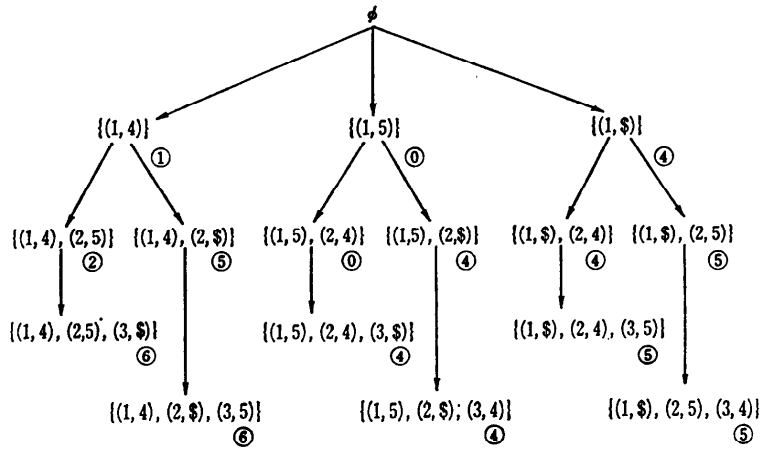



図-12 図-11 のグラフに関する状態展開

析木を作ることを$x$の誤り訂正構文解析というが，どのような基準で近いかを明示しないと，誤り訂正の意味が分からなくなる．これに関して$G$が系列を生成する文法であるとき，$x$をLDあるいは文脈類似度の意味で最適な文$y$に訂正する誤り訂正構文解析法[50)~52)]が，また，$G$が木文法であるとき，Tai距離あるいはSSPM距離の意味で最適な誤り訂正構文解析法[53),54)]が作られている．

構造をもつものの距離や類似度はパターン認識や情景解析のほかに，化合物の検索[55)]や構造-活性関係(structure-activity relationship)の解析[56)]などでもやがて必要になる．

## 参 考 文 献


1) Sankoff, D. and Kruskal, J. B.: Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison, Addison-Wesley Publishing Company (1983).
2) Corneil, D. G. and Gotlieb, C. C.: An Efficient Algorithm for Graph Isomorphism, JACM, Vol. 17, pp. 51-64 (1970).
3) Knuth, D., Morris, J. and Pratt, V.: Fast Pattern Matching in String, SIAM J. Comput., Vol. 6, pp. 323-350 (1977).
4) Zhu, R. F. and Takaoka, T.: The Extension of the Aho-Corasick Algorithm to Multiple Rectangular Pattern Arrays of Different Sizes and N-Dimensional Cases, J. Inf. Process., Vol. 11, pp. 271-277 (1988).
5) Lingas, A. and Karpinski, M.: Subtree Isomorphism is NC Reducible to Bipartite Perfect Matching, Information Processing Letters, Vol. 30, pp. 27-32 (1989).
6) Ullmann, J. R.: An Algorithm for Subgraph Isomorphism, JACM, Vol. 23, pp. 31-42 (1976).
7) Aho, A. V., Hirschberg, D. S. and Ullman, J. D.: Bounds on the Complexity of the Longest Common Subsequence Problem, JACM, Vol. 23, pp. 1-12 (1976).
8) Takahashi, Y., Satoh, Y., Suzuki, H., Abe, H. and Sasaki, S.: Recognition of Largest Common Structural Fragment among a Variety of Chemical Structures, Analytical Sciences, Vol. 3, pp. 23-28 (1986).
9) Wagner, R. A. and Fischer, M. J.: The String-to-String Correction Problem, JACM, Vol. 21, pp. 168-173 (1974).
10) Moore, R. K.: A Dynamic Programming Algorithm for the Distance between Two Finite Areas, IEEE Trans. PAMI, Vol. PAMI-1, pp. 86-88 (1979).
11) Tai, K. C.: The Tree-to-Tree Correcting Problem, JACM, Vol. 26, pp. 422-433 (1979).
12) Bunke, H. and Allermann G.: Inexact Graph Matching for Structural Pattern Recognition, Pattern Recognition Letters, Vol. 1, pp. 245-253 (1983).
13) Landau, G. M. and Vishkin, U.: Fast String Matching with k Differences, J. Comput. Syst. Sciences, Vol. 37, pp. 63-78 (1988).
14) Tsai, W. H. and Fu, K. S.: Subgraph Error-Correcting Isomorphism for Syntactic Pattern Recognition, IEEE Trans. SMC, Vol. SMC-13, pp. 48-62 (1983).
15) 前原，池田，田中：グラフの部分整合アルゴリズム，情報処理学会第40回全国大会，5K-1 (1990).
16) Batagelj, V.: Similarity Measures between Structured Objects, MATH/CHEM/COMP 1988 (Graovac, A. 編) Elsevier, pp. 25-39 (1989).
17) Levenshtein, V. I.: Binary Codes with Correction of Deletions, Insertions and Substitutions of Symbols, Dokl. Akad. Nauk SSSR, Vol. 163, pp. 845-848 (1965).
18) Bunke, H. and Sanfeliu, A. (ed.): Syntactic and Structural Pattern Recognition, Theory and Applications, World Scientific (1990).
19) Abe, K. and Sugita, N.: Distances between Strings of Symbols—Review and Remarks, Proc. 6th ICPR, Munich, pp. 172-174 (1982).
20) Bunke, H.: String Matching for Structural Pattern Recognition, 2) の第5章, pp. 119-144 (1990).
21) Kruskal, J. B.: An Overview of Sequence Comparison, 1) の第1章, pp. 1-44 (1983).
22) 奥田，田中，笠井：距離概念の拡張による語の誤り訂正法，昭47電気関係学会東海支部連合大会，18A-B-6 (1972).
Okuda, T., Tanaka, E. and Kasai, T.: A Method for the Correction of Garbled Words Based on the Levenshtein Metric, IEEE Trans. Comput., Vol. C-25, pp. 172-178 (1976).
23) Lowrance, R. and Wagner, R. A.: An Extention of the String-to-String Correction Problem, JACM, Vol. 22, pp. 177-183 (1975).
24) Tsai, W. H. and Yu, S. S.: Attributed String Matching with Merging for Shape Recognition, IEEE Trans. PAMI, Vol. PAMI-7, pp. 453-462 (1985).
25) 迫江，千葉：動的計画法を利用した音声の時間正規化に基づく連続単語認識，日本音響学会誌，Vol. 29, pp. 483-490 (1971).
26) Sakoe, H. and Chiba, S.: Dynamic Programming Algorithm Optimization for Spoken Word Recognition, IEEE ASSP, Vol. ASSP-26, pp. 43-49 (1978).
27) Tanaka, E.: A String Correction Methods

Based on the Context-Dependent Similarity, Syntactic and Structural Pattern Recognition, ed. by Ferrate, G. et al. Springer-Verlag (1988).
28) 田中，菊池：図形間の距離，電子通信学会論文誌，Vol. 63-D, pp. 1018-1025 (1980).
29) 磯道，小川：動的計画法によるパターン・マッチング，情報処理，Vol. 16, pp. 15-22 (1975).
30) Tanaka, E.: A Two Dimensional Context-Dependent Similarity Measure, Trans. IECE, Vol. E 68, pp. 667-673 (1985).
31) 青木：木と木の距離を求める下降型アルゴリズム，電子通信学会論文誌，Vol. J 66-D, pp. 49-56 (1983).
32) 大森：A Unified View on Tree Metrics, 宇都宮大学大学院工学研究科情報工学専攻修士論文 (1987).
33) 田中，田中：木の間の距離とその計算法，電子通信学会論文誌，Vol. J 65-D, pp. 511-518 (1982).
34) Lu, S. Y. and Fu, K. S.: A Tree-to-Tree Distance and Its Application to Cluster Analysis, IEEE Trans. PAMI, Vol. PAMI-1, pp. 219-224 (1979).
35) 田中：強構造保存写像に基づく木の間の距離とその計算法，電子通信学会論文誌，Vol. J 67-D, pp. 722-723 (1984).
36) Selkow, S. M.: Tree-to-Tree Editing Problem, Information Processing Letters, Vol. 6, pp. 184-186 (1977).
37) 中林，鎌田：2分木間の距離とその計算アルゴリズム，電子通信学会論文誌，Vol. J 66-D, pp. 455-462 (1983).
38) Lu, S. Y.: A Tree-Matching Algorithm Based on Node Splitting and Merging, IEEE Trans. PAMI, Vol. PAMI-6, pp. 249-256 (1984).
39) Cheng, Y. C. and Lu, S. Y.: Waveform Correlation by Tree Matching, IEEE Trans. PAMI, Vol. PAMI-7, pp. 299-305 (1985).
40) Lu, S. Y. and Fu. K. S.: Error-Correcting Tree Automata for Syntactic Pattern Recognition IEEE Trans. Computers, Vol. C-27, pp. 1040-1053 (1978).
41) Wlihelm, R.: A Modified Tree-to-Tree Correction Problem, Information Processing Letters, Vol. 12, pp. 127-132 (1981).
42) Culik II, K. and Wood, D.: A Note on Some Tree Similarity Measures, Information Processing Letters, Vol. 15, pp. 39-42 (1982).
43) Felsenstein, J.: Numerical Methods for Inferring Evolutionary Trees, The Quarterly Review of Biology, Vol. 57, pp. 379-404 (1982).
44) Tsai, W. H. and Fu. K. S.: Error-Correcting Isomorphisms of Attributed Relational Graphs for Pattern Analysis, IEEE Trans. SMC, Vol. SMC-9, pp. 747-768 (1979).
45) Shapiro, L. G. and Haralick, R. M.: Structural Descriptions and Inexact Matching, IEEE Trans. PAMI, Vol. PAMI-3, pp. 504-519 (1981).
46) Shapiro, L. G. and Haralick, R. M.: Matching Relational Structures Using Discrete Relaxation, 18) の第7章, pp. 179-195 (1990).
47) Sanfeliu, A. and Fu. K. S.: A Distance Measure between Attributed Relational Graphs for Pattern Recognition, IEEE Trans. SMC, Vol. SMC-13, pp. 353-362 (1983).
48) Eshera, M. A. and Fu. K. S.: A Graph Distance Measure for Image Analysis, IEEE Trans. SMC, Vol. SMC-14, pp. 398-408 (1984).
49) Kaul, M.: Computing the Minimum Error Distance of Graphs in $O(n^3)$ Time with Precedence Graph Grammars, Syntactic and Structural Pattern Recognition, ed. Ferrate et al., Springer-Verlag (1988).
50) Tanaka, E.: Parsing and Error-Correcting Parsing for String Grammars, 18) の第3章, pp. 55-83 (1990).
51) Ikeda, M. and Tanaka, E.: An Error-Correcting Parser for a Context-Free Language Based on the Context-Dependent Similarity, Syntactic and Structural Pattern Recognition, ed. by Ferrate, G. et al. Springer, pp. 19-32 (1988).
52) Ikeda, M., Tanaka, E. and Kakusho, O.: An Error-Correcting Parser Based on the Context-Dependent Similarity, Syntactic and Structural Pattern Recognition, ed. by Ferrate, G. et al. Springer (1988).
53) 青木，松浦：Tai の距離に基づいた文脈自由木言語の下降形誤り訂正構文解析法，電子通信学会論文誌，Vol. J 66-D, pp. 1015-1022 (1983).
54) 松浦：木言語の構文解析法及び誤り訂正構文解析法，宇都宮大学大学院工学研究科情報工学専攻修士論文，第6章 完全文脈自由木言語の誤り訂正構文解析法，pp. 65-83 (1984).
55) Balaban, A. T.: Applications of Graph Theory in Chemistry, J. Chem. Inf. Comput. Sci., Vol. 25, pp. 334-343 (1985).
56) Jurs, P. C., Stouch, T. R., Czerwinski, M. and Narvaez, J. N.: Computer-Assisted Studies of Molecular Structure-Biological Activity Relationships, J. Chem. Inf. Comput. Sci., Vol. 25, pp. 296-308 (1985).

(平成2年4月16日受付)