Vol.2011-BIO-24 No.1
2011/3/10

上下限付きパス頻度に基づく木状化合物の列挙

清水雅章[†1]　永持仁[†1]　阿久津達也[†1]

分子構造の部分情報が与えられたとき，それに基づく化合物の推定問題は生物情報学において重要な課題の一つであり，薬剤設計などの多くの分野に応用することが期待される．本研究では，部分的な分子構造として，長さが与えられた整数 $K \geq 0$ 以下である部分パスに関する特徴ベクトルの上限と下限が与えられたときに，この上下限制約を満たす木状の化合物を全て列挙する問題を考える．ここで，特徴ベクトルとは，化合物における原子のパスの出現頻度を表す特徴量である．与えられた特徴ベクトルの上限と下限の間に存在する全ての特徴ベクトル一つ一つに対して，石田ら（2008）による分枝限定法を用いたアルゴリズムを適用すれば，上下限制約を満たす木状の化合物を全て列挙することができる．しかし，上下限制約によっては非常に多くの計算を要することになる．そこでより高速に列挙を行うために，分枝限定法を用いた新しい厳密アルゴリズムを設計する．このアルゴリズムは，分枝操作として中野・宇野（2003）によるラベル付き木の列挙アルゴリズムを用いている．限定操作としては，特徴ベクトルの上限と下限が与えられているため，石田ら（2008）による限定操作をそのまま用いることはできない．そこで，特徴ベクトルの上限と下限に基づく特徴ベクトルによる限定操作，価数による限定操作，及びデタッチメントによる限定操作を提案する．

Enumerating Tree-like Chemical Graphs with Given Upper and Lower Bounds on Path Frequencies

Masaaki Shimizu,[†1] Hiroshi Nagamochi[†1] and Tatsuya Akutsu[†1]

Enumeration of chemical graphs satisfying given constraints is one of the fundamental problems in chemoinfomatics since they lead to a variety of useful applications including drug design. In this paper, we consider the problem of enumerating all tree-like chemical graphs from a given set of feature vectors, which is specified by a pair of upper and lower feature vectors, where a feature vector represents frequency of prescribed paths in a chemical compound to be constructed. This problem can be solved by applying the algorithm proposed by Ishida et al. to each single feature vector in the given set, but this method may take much computation time because in general there are many feature vectors in a given set. We propose a new exact branch-and-bound algorithm for the problem so that all the feature vectors in a given set are handled directly. In our algorithm, the branching operation is based on Nakano and Uno's enumeration algorithm on labeled trees. Since we cannot use the bounding operation proposed Ishida et al. due to upper and lower constraints, we introduce new bounding operations based on upper and lower feature vectors, a bond constraint, and a detachment condition.

1. 序　論

1.1 背　景

薬剤設計は生体生命情報学にとって重要な目的の一つであり，その重要なステップの一つとして，望ましい性質をもつ化合物を見分ける作業が挙げられる．近年，サポートベクターマシン（SVM）や，カーネル法に基づいた化合物の分類は広く研究されているが[5),7),14),18)]，それらに通じる考えは，化合物を特徴ベクトルに写像し，サポートベクターマシンを用いてその特徴ベクトル空間からの予測を行うことである[6)]．特徴ベクトルによる化合物の表現方法はいくつかあるが，ラベル付きのパス[14),18)] や部分構造[5),7)] の出現頻度に基づいたものが広く使われている．

現在，カーネル法を用いて，入力空間上で化合物を設計・最適化する手法が提案されている[2),3)]．この手法では，化合物は適当な関数によって特徴ベクトル空間上での一つの点として表わされ，その点からの元の入力空間への写像（原像）を求める．グラフに対する原像問題は，目的関数をうまく選ぶことができれば，所望の性質をもつような化合物を見つけることが期待され，薬剤の設計における候補化合物のスクリーニングへの応用が考えられる．

原像問題に関しては，様々な研究がなされてきた[2),3)]．しかし，これらはヒューリスティックスや確率的な手法に基づいており，厳密アルゴリズムが取り上げられるようになったのは最近のことである．阿久津と深川[1)] は原像問題を，長さ K 以下のラベル付きパスの出現頻度からグラフを推定する問題に定式化し，この問題は次数制約を持たせた平面的グラフにおいても NP 困難であることを示した．そのほか，永持[19)] は最大パス長が $K = 1$ のグラフ推定問題は，連結したデタッチメントを見つける問題として定式化することにより，多項式

†1 京都大学
Kyoto University

Vol.2011-BIO-24 No.1
2011/3/10

時間で解を得ることができることを証明した．

化合物の列挙に関する研究は，マススペクトルや NMR スペクトルを用いた構造決定の補助的役割などの実用的な目的で広く行われている[4),10),15),17)]．その中で，与えられた分子の特徴から，考えられる構造を列挙するような研究がいくつか行われているが[8),11)]，それらはパス頻度に基づいた特徴ベクトルによる原像問題と密接に関係している．また，最近の列挙に関する研究においては，グラフや木を効率よく列挙することが重要な役割を果たすことが多くなっている．

藤原ら[9)] は木状の化学構造の推定問題に対して，分枝限定法に基づくアルゴリズムを提案した．彼はこの中で，中野と宇野[20)] によって考案された左荷重（left heavy）である木を列挙し，同型なものの重複列挙を防いでいる．また，石田ら[12)] が従来のアルゴリズムに，永持[19)] によって考案されたデタッチメントによる限定操作を新たに導入し，アルゴリズムの高速化を図った．このアルゴリズムは誰でも利用出来るように，ウエブサーバーとして公開されている（http://sunflower.kuicr.kyoto-u.ac.jp/tools/enumol/）．

従来の研究では，与えられた特徴ベクトルに完全に一致する化合物を列挙していたが，本研究では，与えられた二つの特徴ベクトル，すなわち上限と下限の特徴ベクトルの「間」にある化合物を列挙する問題を考える．上限と下限の特徴ベクトルの間に存在する全ての特徴ベクトル一つ一つに対して，石田ら[12)] によるアルゴリズムを用いれば，この問題は解くことができる．しかし，一般に上限と下限の特徴ベクトルの間には多くの特徴ベクトルが存在するため，時間がかかることが予想される．そこで，より高速に列挙を行うための分枝限定法に基づくアルゴリズムを提案する．以下，本節の終わりまで問題の定式化を述べ，列挙のための標準形および分枝操作を第 2 節で与え，本研究で提案する限定操作を第 3 節で，実験結果と考察を第 4 節で述べる．

1.2 化合物列挙問題

本節では，用語の定義と化合物列挙問題の定式化を行う．同じ組み合わせの節点を結ぶ 2 本以上の枝を多重枝という．多重枝の存在を許すグラフのことを多重グラフと呼び，そうでないグラフのことを単純グラフと呼ぶ．特に，閉路や自己ループを持たない連結な多重グラフのことを多重木と呼ぶ．ラベルの集合を Σ としたとき，Σ^k を Σ に含まれるラベルの長さが k のすべての列の集合とし，$\Sigma^{\leq k} = \cup_{j=1}^{k}\Sigma^j$, $\mathcal{F}_k(\Sigma) = \{g : \Sigma^{\leq k+1} \to \mathbb{Z}_+\}$ と定義する．ここで，$\mathbb{Z}_+$ は非負整数の集合とする．

多重グラフ G の各節点 v に対し，$l(v)$ $(\in \Sigma)$ で表されるラベルが与えられているとき，G は Σ-ラベル付きグラフという．さらに，各節点に $val(l(v)) \in \mathbb{N}$ で表される価数が与えられているとき，G は (Σ, val)-ラベル付きグラフという．このとき，木状の化合物は閉路のない連結な (Σ, val)-ラベル付き多重グラフとして表現することができ，ラベル付けされたそれぞれの節点は原子を，節点間の枝の多重度はそれぞれ対応する原子間の結合の多重度を，節点の次数は対応する原子の価数を表す．パス P を $P = (v_0, v_1, \ldots, v_s)$ としたとき，ラベル列を $l(P) = (l(v_0), l(v_1), \ldots, l(v_s))$ と定義する．ラベル列 t に対し，$occ(t, G)$ は G に含まれるラベル列が t のパスの数とする．また，G の最大パス長が K の特徴ベクトルを，$f_K(G) = (occ(t, G))_{t \in \Sigma^{\leq K+1}}$ で定義する．**図 1** は化合物を表す (Σ, val)-ラベル付き多重木と，最大パス長が 1 の特徴ベクトルの例を示している．

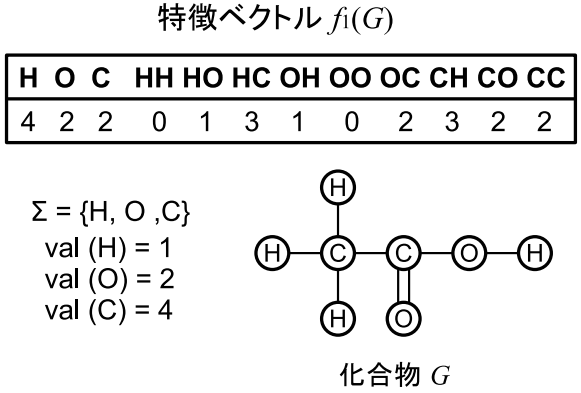

図 1 化合物と特徴ベクトル
Fig. 1 Compound and Feature vector.

与えられた一つの特徴ベクトルから，木状の化合物をすべて列挙する問題は，以下のように定式化される[9)]．

与えられたパス頻度に基づく木状化合物列挙（Enumerating Tree-like chemical graphs with given Path Frequency, ETPF） 与えられたラベル集合 Σ，自然数 K，特徴ベクトル $g \in \mathcal{F}_K(\Sigma)$ および価数 $val : \Sigma \to \mathbb{N}$ から $f_K(T) = g$ かつ，T に含まれる全ての節点 v に対し $deg(v) = val(l(v))$ を満たすような全ての (Σ, val)-ラベル付き多重木 T を出力せよ．もしそのような T が存在しなければ「解なし」と出力せよ．

Vol.2011-BIO-24 No.1
2011/3/10

また，多くの化合物において水素原子の占める割合が大きいことに注目して，問題 ETPF において，水素原子を取り除くことによる別のモデルを考えることもできる．このとき，枝の多重度を別の表現で表す必要があり，これには二つのモデルが提案されている[9),21)]．

本研究では，与えられた上限と下限に対応する二つの特徴ベクトルから，条件をみたす木状の化合物を全て列挙することを目的とする．それは問題 ETPF を元にして，以下のように定式化できる[21)]．なお，上限を表す特徴ベクトルを g_U，下限を表す特徴ベクトルを g_L と書き，二つの特徴ベクトル $g_1, g_2 \in \mathcal{F}_K(\Sigma)$ において，$g_1 \leq g_2$ であるとは g_1 の全ての成分の値が g_2 以下であると定義する．

与えられた上下限付きパス頻度に基づく木状化合物列挙（Enumerating Tree-like chemical graphs with given Upper and Lower bounds on path Frequencies, ETULF） 与えられたラベル集合 Σ，自然数 K，特徴ベクトル $g_U, g_L \in \mathcal{F}_K(\Sigma)$ および価数 $val : \Sigma \to \mathbb{N}$ から $g_L \leq f_K(T) \leq g_U$ かつ，T に含まれる全ての節点 v に対し $deg(v) = val(l(v))$ を満たすような全ての (Σ, val)-ラベル付き多重木 T を出力せよ．もしそのような T が存在しなければ「解なし」と出力せよ．

ここで本研究における上限，下限の特徴ベクトルに設ける条件について述べる．まず，パス $l(v), v \in \Sigma$ については上限と下限が等しいものとする．すなわち，各原子の数は固定される．それ以外のパスについては，上限の特徴ベクトルの成分が下限の特徴ベクトルの成分以上であるとする．すなわち，上限と下限が等しいとする部分については，$g_L = g_U$ であるという条件を課し，それ以外については，$g_L \leq g_U$ という条件を課す．

1.3 アルゴリズムの概略

1.2 節で述べたように，本研究では，価数と特徴ベクトルの条件を満たす多重木を列挙することを目的とするが，同型なものを重複して列挙するのを防ぐために，根付き木における標準形を導入する．木の列挙は，まず，空の木 $T = \emptyset$ に根となる節点を追加し，それから節点数が入力で与えられた節点数と等しくなるまで，木 T が標準形であるという条件を保ったまま節点を追加していく分枝操作[9),20)] を行う．この際，節点をひとつ追加するたびに，入力で与えられた上限と下限の特徴ベクトルの制約と価数の制約に反していないかを調べ，もし反していれば，それ以上節点を追加するのをやめる限定操作を行う．限定操作については，上限と下限の二つの特徴ベクトルが存在するため，石田ら[12)] が提案する限定操作をそのまま用いることはできない．そこで本研究では，上限と下限の二つの特徴ベクトルに基づく限定操作を新しく提案する．そして，木 T の節点数が入力で与えられた節点数と等しくなったとき，木 T の価数と特徴ベクトルが入力で与えられた条件を満たしていれば，T を解として出力する．なお，分枝操作と各限定操作については，それぞれ 2 節と 3 節で述べる．

2. ラベル付き多重木の標準形と分枝操作

この節では，藤原ら[9)] が導入した列挙のためのラベル付き多重木の標準形について述べ，それに基づいた分枝操作について述べる．

まず，以下の定理（Jordan[16)]）に基づいて根を導入する．

定理 1. 任意の n 個の節点をもつ単純木において，v^* を取り除いてできる部分木に含まれる節点が高々 $\lfloor (n-1)/2 \rfloor$ 個であるような節点 v^*，または e^* を取り除いてできる二つの部分木に含まれる節点はともに $n/2$ 個であるような枝 e^* のどちらか一方が一意に存在する．

定理 1 の v^*, e^* をそれぞれ，単重心（unicentroid），双重心（bicentroid）と呼ぶ．また，双重心が存在するのは n が偶数のときだけである．双重心の場合も単重心の場合とほぼ同様なので，以下では単重心の場合に限って説明を行う．

定理 1 に基づいて根を導入したら，次に木の形を一意に記述できる標準形を定義する．根付き木が平面に埋め込まれているとし，葉でない各節点の子は追加された順に左から右へ順序づけされるものとする．T を節点数が n の根付き木とするとき，左から深さ優先探索をしたときに訪れた順に節点を $v_0, v_1, \ldots, v_{n-1}$ としても一般性を失わない．このとき，深さラベル列[9),20)] を以下のように定義する．

$$L(T) = (d(v_0), l(v_0), d(v_1), l(v_1), \ldots, d(v_{n-1}), l(v_{n-1})).$$

ここで，$d(v_i), l(v_i)$ はそれぞれ v の深さとラベルである．このとき，任意に定めたラベルの大小関係に対して，深さラベル列が辞書式順で最大となるものを根付き木 T の標準形と定める．標準形の木 T から T の最も右の葉を取り除いてできる木 $P(T)$ を T の親と呼び，それに対応して T を $P(T)$ の子と呼ぶ．同様に先祖 $P(P(T))$ や $P(P(P(T)))$ も定義できる．

以下の補題が中野と宇野[20)] によって示されている．

補題 1. 最大の深さラベル列をもつ根付き単純木の埋め込みは左荷重である．

補題 2. 左荷重な木 T から，T の最も右の葉を取り除いて得られる木もまた左荷重である．

補題 1，2 より T の先祖は全て標準形となる．この標準形の木における親子関係により，家族木 $\mathcal{F}(n, m)$ を定義することができる．ここで，n は家族木における最大節点数で，m はラベル集合の要素数 $(= |\Sigma|)$ である．この家族木をなぞることによって，与えられた制約を

満たす木状化合物を列挙することができる．これは一つの節点，すなわち根から始め，生成される木の最右パス上の相応しい節点に新たな節点を追加していくことで実現できる．ここで，最右パス上のどの節点に新しい節点を追加すれば標準形の木が得られるかということは自明ではないが，最右パス上のどの深さの節点にどのラベルの節点を追加すればよいかは定数時間で決定することができ、これは藤原ら[9] によって示されている．

3. 限定操作

2 節では，現在得られている木の最右パス上に節点を追加していくことによって標準形の木を列挙することを述べた．そこで，この各反復毎に，それぞれ特徴ベクトル，価数，デタッチメント，多重度の制約を満たしているかを判定し，条件に反していれば，現在得られている木のそれ以上の探索を行わない．以下では，これら四つの限定操作について述べる．

3.1 特徴ベクトルカット

本研究では，上限と下限の二つの特徴ベクトルが存在するため，石田ら[12] が導入した特徴ベクトルによる限定操作をそのまま用いることはできないが，少しの改良を加えることによって，上限と下限の二つの特徴ベクトルが存在する場合においても，特徴ベクトルによる限定操作を導入することができる[21]．具体的には，現在探索中の木 T に対し，特徴ベクトル $f_K(T)$ と入力で与えられた上限の特徴ベクトル g_U について，

$$f_K(T) \leq g_U \tag{1}$$

を満たすかどうかを判定し，式 (1) に反した場合，T を破棄する．また，T の節点数が入力で与えられた節点数と一致する場合は，入力で与えられた上限の特徴ベクトル g_U と下限の特徴ベクトル g_L について，

$$g_L \leq f_K(T) \leq g_U \tag{2}$$

を満たすかどうかを判定する．もし，式 (2) に反した場合，T を破棄する．

3.2 価数カット

出力として得られる木においては，価数の制約を満たす必要がある．ここで，問題 ETULF においては, 入力に対する価数の制約は特徴ベクトルに依存しないため, 問題 ETPF と同じ制約となる．つまり問題 ETULF では, 藤原ら[9] が導入した価数による限定操作をそのまま用いることができる．

3.3 デタッチメントカット

デタッチメント[19] を用いた限定操作が石田ら[12] によって提案されている．これは入力で与えられた特徴ベクトルのパス長が 1 以下の情報と現在の木の情報から，特徴ベクトルのパス長が 1 以下の制約を満たす解が少なくとも一つ得られるかどうかを，デタッチメントというデータ構造を利用して判定するものである．

しかし，本研究においては特徴ベクトルが上限と下限の形として与えられているため，本研究で考える問題に，石田らによるデタッチメントを用いた限定操作をそのまま用いることはできない．そのため，上限と下限を考慮にいれた新しいデタッチメントによる限定操作を導入している[21]．以下で概略を述べる．

まず，現在の木および下限の特徴ベクトルのパス長が 1 以下の情報から，下限制約を満たすために最低限必要な各原子の結合の手の数を計算し，まだ使われていない手の数と比較する．もし，下限制約を満たすために必要な手の数が残っていなければ，そこで現在の木を破棄する．

次に，現在の木および上限の特徴ベクトルのパス長が 1 以下の情報から，特徴ベクトルのパス長が 1 以下の上下限制約を満たす解が少なくとも一つ得られるかどうかを，デタッチメントを利用して判定を行う．この判定は石田らが用いたデタッチメントによる限定操作を少し修正することにより行うことができる．もし，条件に反した場合，現在の木を破棄する．

3.4 多重度カット

ここでは，新しく提案する多重度による限定操作について述べる．問題 ETULF においては，特徴ベクトルにおいて単結合と多重結合は区別されていない．しかし，入力で与えられた各ラベルの節点数と価数および木状化合物という制約から，出力として得られる木における多重結合の数を見積もることができる．以下では多重結合の数を見積もる方法とそれを用いた限定操作について述べる．

$g(\ell)$ を特徴ベクトルから得られるラベルが $\ell \in \Sigma$ の節点数とする．問題 ETULF においては，全ての $g(\ell)$ は固定されているので，出力として得られる木の枝数を計算することができる．今，n を出力として得られる木の節点数とする．木における各枝の枝数を枝の多重度と定義した場合，出力として得られる木の枝数 e_m は，$val(\ell)$ をラベルが ℓ の節点の価数とするとき，以下のように計算することができる．

$$e_m = \frac{1}{2}\sum_{\ell \in \Sigma} val(\ell)g(\ell).$$

一方，多重枝を単純枝として扱った場合の出力として得られる木の枝数 e_s は，木状化合物を考えていることから

$$e_s = n - 1$$

Vol.2011-BIO-24 No.1
2011/3/10

となる．ここで，e_m と e_s の差

$$M = e_m - e_s$$

を考えると，M は多重結合を形成するために使われる枝の数を与える．

また，$T = (V, E)$ を多重木，m_e を $e \in E$ の多重度とするとき，多重木 T の多重度 $M(T)$ を以下のように定義する．

$$M(T) = \sum_{e \in E} (m_e - 1).$$

以下では，M と $M(T)$ に基づいた多重度カットについて説明する．

T を分枝操作において現在探索中の根付き多重木，$M(T)$ を T の多重度，$RP(T) = (r_0, r_1, \ldots, r_k)$ を T の最右パス，木 T_c を新しい節点 p を節点 r_i $(0 \leq i \leq k)$ に追加することによって得られる新しい根付き多重木とし，$RP(T_c)$ を T_c の最右パスとする．これにより，$RP(T_c)$ は $RP(T_c) = (r_0, r_1, \ldots, r_i, p)$ となるため，パス $\{(r_j, r_{j-1}), j = k, k-1, \ldots, i+1\}$ 上の枝の多重度を決定することができる．多重度を決定した後に，$M(T_c)$ の更新を行う．木 T_c における枝 (r_j, r_{j-1}) の多重度を $Mul(r_j, r_{j-1}|T_c)$ と書くとき，$M(T_c)$ を以下のように更新する．

$$M(T_c) := M(T) + \sum_{j=k}^{i+1} (Mul(r_j, r_{j-1}|T_c) - 1).$$

M は多重結合を形成するために使われる枝の数であったので，T_c において以下の条件を満たすかどうかを判定する．

$$M(T_c) \leq M. \tag{3}$$

もし，式（3）に反した場合，T_c を破棄する．**図 2** はその様子を表したものである．

4. 実験結果と考察

まず，問題 ETULF が石田ら[12] が提案した問題 ETPF を解くアルゴリズムによって解くことができることを示す．問題 ETULF について，上限と下限の特徴ベクトルの間に含まれる全ての特徴ベクトルの総数を N とすると，問題 ETULF は N 個の問題 ETPF の集合である．ゆえに，N 個の特徴ベクトル一つ一つに問題 ETPF を解くアルゴリズムを適用することによって，問題 ETULF を解くことができる．このアルゴリズムを Division とする．

そこで，問題 ETULF に対して，本研究で提案するアルゴリズム（Enumerate とする）を用いて解く方法と，アルゴリズム Division を用いて解く方法について，計算時間や探索節点数を比較するために計算実験を行った．また，特徴ベクトルに上限と下限を設けることによる実行可能解の数や探索節点数の変化を調べるため，上限と下限の幅を変化させて計算実験を行った．

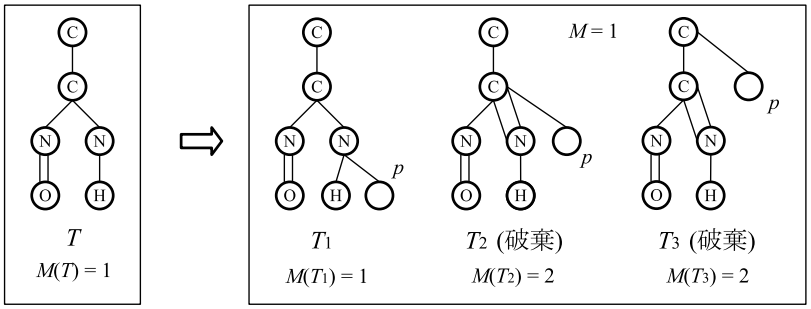

図 2 $M = 1$ の場合における多重度カットの様子
Fig. 2 An illustration of the multiplicity-cut procedure, where $M = 1$.

計算実験は PC（AMD Athlon Dual Core Processor 5050e）上で行った．問題例は，KEGG LIGAND データベース[13]（http://www.genome.jp/ligand/）から抽出したものである．ここで，ベンゼン環は価数が 6 の新たな原子であるとみなしている．本研究では，抽出した問題例を特徴ベクトル t の形で表現し，上限の特徴ベクトルと下限の特徴ベクトルのパス頻度の幅を $w \in \mathbb{Z}_+$ とする．つまり，ベクトル t の各成分値 $a > 0$ に対し，その値を $a + w$ としたものを上限の特徴ベクトル，逆に $a - w$（$a - w < 0$ なら $a := 0$）としたものを下限の特徴ベクトルとする．特に，$w = 0$ とすれば問題 ETULF の問題例は問題 ETPF の問題例に帰着される．

本研究における計算実験では，幅 w を 1 として，KEGG LIGAND データベースから抽出した問題例から問題 ETULF の問題例を生成し，アルゴリズム Enumerate とアルゴリズム Division に対し，計算時間，探索節点数，実行可能解の数を出力し比較を行った．また，幅 w を変化させて問題 ETULF の問題例を生成し，それに対してアルゴリズム Enumerate で解いた場合の計算時間，探索節点数，実行可能解の数を出力し比較を行った．

表 1 は，問題 ETULF に対する二つのアルゴリズム（Enumerate, Division）の比較実験の結果である．また，**表 2** は，幅 w を変化させた問題例に対して，アルゴリズム Enumerate を用いて解いた計算実験の結果である．ここで，入力例は KEGG LIGAND データベースにおいて用いられているエントリー番号であり，n は各問題における原子数，K は特徴ベクトルの最大パス長，w は幅，解は制限時間内に計算された実行可能解の数を表している．

Vol.2011-BIO-24 No.1
2011/3/10

制限時間は 1800 秒とし，各問題に対する計算時間，制限時間内に探索された探索節点数，制限時間内に計算された実行可能解の数を記した．

まず，アルゴリズム Division では $K \geq 2$ では全ての問題において時間内に計算が終了していない．これは，K が大きくなると，特徴ベクトルの総数が指数的に一気に増加するため，全ての問題を解くことが難しいためであると考えられる．一方，アルゴリズム Enumerate では K の値が大きくなるとより速く問題を解くことができることがわかる．よって，アルゴリズム Division に比べ，アルゴリズム Enumerate のほうが優れていることがわかる．

また，幅 w を大きくしていくと，探索空間は指数的に一気に大きくなる．アルゴリズム Enumerate において，w が 0 から 1 に増えるときは一気に探索節点数が増加するが，w が 2 以降の場合については，探索空間の増え方を考えれば探索節点の増加は緩やかで，計算時間も急激には増えていない．これは，アルゴリズム Enumerate が広大な探索空間に対しても有効であることを示している．

5. まとめと今後の課題

本研究では，与えられた二つの特徴ベクトルから，条件を満足する化合物を全て列挙する問題に対し，分枝限定法に基づいた厳密アルゴリズムを構成した．その結果，従来の方法を用いて本研究における問題を解く方法に比べて，探索節点数を大きく減らすことに成功し，計算時間も短縮することができ，多くの問題において全ての実行可能解を列挙することができた．しかし，与えられた特徴ベクトルに完全に一致する化合物を列挙する問題に対し，上限と下限の特徴ベクトルの間に存在する化合物を列挙することになるため，探索範囲が広大になり，どうしても探索節点数が多くなってしまう．今後の課題としては，上限と下限の特徴ベクトルの条件をうまく利用した限定操作の改良，もしくは新たな限定操作の導入によって探索節点数を減少させることが挙げられる．また，本研究で扱う問題においては入力の各原子数は固定されているので，各原子の個数と価数の情報をもとに，上限の特徴ベクトルの各成分がとりうる値の上限値を評価することができる．いま，与えられた特徴ベクトルの最大パス長が 1 の場合においては，最大パス長が 2 以降の場合よりも，問題を解くためにはるかに多くの時間がかかっていることがわかる．そこで各原子の個数と価数，および特徴ベクトルのパス長が 1 の部分の情報から，長さが 2 の各ラベル列について，パス頻度の上限と下限の値を評価し，上限と下限の特徴ベクトルにパス長が 2 の部分の情報を追加する．このようにして最大パス長が 1 の特徴ベクトルから最大パス長が 2 の特徴ベクトルを作り，問題を解けば，計算時間を短縮できることが期待できる．

表 1 問題 ETULF に対する二つのアルゴリズムによる比較実験
Table 1 Comparison of Enumerate and Division for ETULF.

入力例	n	K	w	Enumerate 時間	Enumerate 探索節点	Enumerate 解	Division 時間	Division 解
C00062 $C_6H_{14}N_2O_4$	26	1	1	1344.56	189,684,289	414,890	242.95	414,890
		2	1	3.58	400,501	44	T.O.	N.F.
		3	1	1.50	149,722	2	T.O.	N.F.
		4	1	0.37	35,810	1	T.O.	N.F.
		5	1	0.38	20,846	1	T.O.	N.F.
		6	1	0.28	15,582	1	T.O.	N.F.
		7	1	0.31	14,960	1	T.O.	N.F.
C03343 $C_{16}H_{22}O_4$	37	1	1	T.O.	344,075,147	N.F.	T.O.	N.F.
		2	1	8.39	845,760	25	T.O.	N.F.
		3	1	3.32	307,151	7	T.O.	N.F.
		4	1	1.22	99,945	1	T.O.	N.F.
		5	1	1.18	87,600	1	T.O.	N.F.
		6	1	0.95	60,194	1	T.O.	N.F.
		7	1	0.70	42,538	1	T.O.	N.F.
C07178 $C_{21}H_{28}N_2O_5$	46	1	1	T.O.	158,532,443	N.F.	T.O.	N.F.
		2	1	46.81	2,246,578	238	T.O.	N.F.
		3	1	1.63	67,855	3	T.O.	N.F.
		4	1	0.50	15,164	1	T.O.	N.F.
		5	1	0.39	11,543	1	T.O.	N.F.
		6	1	0.40	11,355	1	T.O.	N.F.
		7	1	0.36	9,920	1	T.O.	N.F.
C03690 $C_{24}H_{38}O_4$	61	1	1	T.O.	292,573,087	N.F.	T.O.	N.F.
		2	1	T.O.	181,053,717	N.F.	T.O.	N.F.
		3	1	T.O.	163,939,390	N.F.	T.O.	N.F.
		4	1	324.98	25,750,543	4	T.O.	N.F.
		5	1	263.35	18,718,379	2	T.O.	N.F.
		6	1	132.55	8,286,117	1	T.O.	N.F.
		7	1	53.12	3,058,895	1	T.O.	N.F.

注意
T.O. は制限時間 1,800 秒以内に計算が終了しなかったことを示す．

参 考 文 献

1) Akutsu, T. and Fukagawa, D.: Inferring a graph from path frequency, *Lecture Notes in Computer Science*, Vol.3537, pp.371–392 (2005).
2) Bakir, G.H., Weston, J. and Schölkopf, B.: Learning to find pre-images, *Advances*

表 2 問題 ETULF に対する幅を変化させた場合の比較実験
Table 2 Comparison of each width w for ETULF.

入力例	n	K	w	Enumerate 時間	探索節点	解
		2	0	0.51	55,196	6
		2	1	3.58	400,501	44
		2	2	7.58	835,509	503
C00062	26	2	3	10.84	1,163,548	2,351
$C_6H_{14}N_2O_4$		2	4	12.55	1,349,057	5,430
		2	5	13.29	1,431,075	9,852
		2	50	14.31	1,537,496	25,425
		2	0	0.34	35,952	9
		2	1	8.39	845,760	25
		2	2	48.27	4,815,369	41
C03343	37	2	3	149.83	14,781,738	305
$C_{16}H_{22}O_4$		2	4	377.01	37,435,878	40,732
		2	5	639.68	63,459,180	106,870
		2	50	1118.75	110,703,034	510,079
		2	0	2.33	111,781	16
		2	1	46.81	2,246,578	238
		2	2	96.52	4,715,072	1,375
C07178	46	2	3	152.18	7,420,060	6,824
$C_{21}H_{28}N_2O_5$		2	4	179.42	8,744,563	19,180
		2	5	199.66	9,677,513	29,891
		2	50	255.01	12,292,587	54,861
		5	0	19.50	1,482,017	2
		5	1	220.14	16,063,569	5
		5	2	439.12	33,037,741	32
C03690	61	5	3	684.88	52,207,745	178
$C_{24}H_{38}O_4$		5	4	1024.96	78,509,554	349
		5	5	1285.55	98,762,291	615
		5	50	T.O.	136,835,134	N.F.

注意
T.O. は制限時間 1,800 秒以内に計算が終了しなかったことを示す.

in Neural Information Processing Systems, Vol.16, pp.449–456 (2003).
3) Bakir, G.H., Zien, A. and Tsuda, K.: Learning to find graph pre-images, *Lecture Notes in Computer Science*, Vol.3175, pp.253–261 (2004).
4) Buchanan, B.G. and Feigenbaum, E.A.: DENDRAL and Meta-DENDRAL: their applications dimension, *Aritificial Intelligence*, Vol.11, pp.5–24 (1978).
5) Byvatov, E., Fechner, U., Sadowski, J. and Schneider, G.: Comparison of support vector machine and artificial neural network systems for drug/nondrug classification, *Journal of Chemical Information and Computer Sciences*, Vol. 43, pp. 1882–1889 (2003).
6) Cristianini, N. and Shawe-Taylor, J.: *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*, Cambridge University Press (2000).
7) Deshpande, M., Kuramochi, M., Wale, N. and Karypis, G.: Frequent substructure-based approaches for classifying chemical compounds, *IEEE Transactions on Knowledge and Data Engineering*, Vol.17, pp.1036–1050 (2005).
8) Faulon, J.L. and Churchwell, C.J.: The signature molecular descriptor. 2. Enumerating molecules from their extended valence sequences, *Journal of Chemical Information and Computer Sciences*, Vol.43, pp.721–734 (2003).
9) Fujiwara, H., Wang, J., Zhao, L., Nagamochi, H. and Akutsu, T.: Enumerating Tree-like Chemical Structures from Feature Vector, *IPSJ SIG Technical Reports*, Vol.2006, No.135, pp.111–118 (2006).
10) Funatsu, K. and Sasaki, S.: Recent advances in the automated structure elucidation system, CHEMICS. Utilization of two-dimensional NMR spectral information and development of peripheral functions for examination of candidates, *Journal of Chemical Information and Computer Sciences*, Vol.36, pp.190–204 (1996).
11) Hall, L.H. and Dailey, E.S.: Design of molecules from quantitative structure-activity relationship models. 3. Role of higher order path counts: path 3, *Journal of Chemical Information and Computer Sciences*, Vol.33, pp.598–603 (1996).
12) Ishida, Y., Zhao, L., Nagamochi, H. and Akutsu, T.: Improved algorithms for enumerating tree-like chemical graphs with given path frequency, *Genome Informatics*, Vol.21, pp.53–64 (2008).
13) Kanehisa, M., Goto, S., Furumichi, M., Tanabe, M., and Hirakawa, M.: KEGG for representation and analysis of molecular networks involving diseases and drugs, *Nucleic Acids Res*, Vol.38, pp.D355–D360 (2010).
14) Kashima, H., Tsuda, K. and Inokuchi, A.: Marginalized kernels between labeled graphs, *Proceedings of the 20th International Conference Machine Learning*, pp. 321–328 (2003).
15) Knop, J.V., Müeller, W.R., Ž. Jeričevič and Trinajstič, N.: Computer enumeration and generation of trees and rooted trees, *Journal of Chemical Information and Computer Science*, Vol.21, pp.91–99 (1981).
16) Kvasnička, V. and Pospichal, J.: Constructive Enumeration of Acyclic Molecules, *Collect Czech Chem Commun*, Vol.56, pp.1777–1802 (1991).
17) Lederberg, W.: Topologocal mapping of organic molecules, *Proceedings of the National Academy of Sciences*, Vol.53, pp.134–139 (1965).
18) Mahé, P., Ueda, N., Akutsu, T., Perret, J.L. and Vert, J.P.: Graph kernels for

Vol.2011-BIO-24 No.1
2011/3/10

molecular structure-activity relationship analysis with support vector machines, *Journal of Chemical Information and Modeling*, Vol.45, pp.939–951 (2005).
19) Nagamochi, H.: A detachment algorithm for inferring a graph from path frequency, *Algorithmica*, Vol.53, pp.207–224 (2009).
20) Nakano, S. and Uno, T.: Efficient Generation of Rooted Trees, *NII Technical Report* (NII-2003-005E, 2003).
21) 清水雅章：上下限付きパス頻度に基づく木状化合物の列挙，京都大学卒業論文 (2010).