論 文

Technical Papers

AGMアルゴリズムの高速化と立体構造解析への適用

Fast Apriori-based Graph Mining Algorithm and application to 3-dimensional Structure Analysis

西村 芳男 Yoshio Nishimura
大阪大学産業科学研究所
The Institute of Scientific and Industrial Research, Osaka University
nishimura@ar.sanken.osaka-u.ac.jp

鷲尾 隆 Takashi Washio
（同　上）
washio@ar.sanken.osaka-u.ac.jp

吉田 哲也 Tetsuya Yoshida
（同　上）
yoshida@ar.sanken.osaka-u.ac.jp

元田 浩 Hiroshi Motoda
（同　上）
motoda@ar.sanken.osaka-u.ac.jp

猪口 明博 Akihiro Inokuchi
日本アイ・ビー・エム株式会社東京基礎研究所
IBM Japan, Ltd. Tokyo Research Laboratory
inokuchi@jp.ibm.com

岡田 孝 Takashi Okada
関西学院大学情報メディア教育センター
Center for Information & Media Studies, Kwansei Gakuin University
okada@kwansei.ac.jp

keywords: graph structured data, apriori algorithm, 3-dimensional structure, physiological activity, chemistry

Summary

Apriori-based Graph Mining (AGM) algorithm efficiently extracts all the subgraph patterns which frequently appear in graph structured data. The algorithm can deal with general graph structured data with multiple labels of vartices and edges, and is capable of analyzing the topological structure of graphs. In this paper, we propose a new method to analyze graph structured data for a 3-dimensional coordinate by AGM. In this method the distance between each vertex of a graph is calculated and added to the edge label so that AGM can handle 3-dimensional graph structured data. One problem in our approach is that the number of edge labels increases, which results in the increase of computational time to extract subgraph patterns. To alleviate this problem, we also propose a faster algorithm of AGM by adding an extra constraint to reduce the number of generated candidates for seeking frequent subgraphs. Chemical compounds with dopamine antagonist in MDDR database were analyzed by AGM to characterize their 3-dimensional chemical structure and correlation with physiological activity.

1. はじめに

近年，計算機の計算能力が急速に発達し，その能力を最大限に活用するデータマイニングが産業および科学技術分野で注目されている．医療や化学の分野では，新薬開発によって多くの新規化合物が合成され，私たちの生活や健康に役立っている．しかし，その化合物の有効性や有害性の調査は，巨額の費用と長い年月をかけた臨床実験が必要であり，すべての化合物の調査を行うのは経済的，時間的に困難である．そこで，化合物の分子構造から薬としての有効性や有害性を表す生理活性を事前に予測し，新薬開発を支援することは極めて意義が高い．

医療や化学の分野では化合物をグラフとして表現する．そして，データベースにはこのような多くの化合物の情報が蓄えられており，データベースに部分グラフとして含まれる特徴的なフラグメントを抽出するさまざまな手法が提案されている．フラグメントの抽出はNP完全である部分グラフ同型問題を含むため，実用時間での抽出は非常に困難な問題である．例えば，WARMR[Dehaspe 98]のように一階述語論理を用いた帰納論理プログラミング(ILP)はグラフを強力に表現できる．しかし，この手法の探索空間は非常に広いため，実用時間で抽出できるフラグメントはごく小さなものである．そのため，フラグメントを抽出する手法はSUBDUE[Cook 94]やGBI[Yoshida 95, Motoda 97]のように計算時間の観点からGreedy探索を用いるものが多い．また，完全探索を用いる手法で

は，抽出される部分グラフがループや枝分かれを含まないフラグメントに限定される MolFea [DeRaedt 01]，連結グラフと呼ばれるすべての原子が繋がっている状態にある一部のフラグメントに限定される FSG[Kuramochi 01], gFSG[Kuramochi 02], gSpan[Yan 02] など制限のあるものが多い．筆者らは，グラフデータベースに誘導部分グラフとして含まれるすべてのフラグメントを完全探索で抽出する AGM アルゴリズム [Inokuchi 00, Inokuchi 02] を開発した．AGM アルゴリズムは頂点及び辺が複数の種類（ラベル）を持つ有向・無向グラフから特徴的パターンを抽出する．この特徴的パターンは連結グラフに限定されず，互いに連続せず分離した位置にある誘導部分グラフも抽出可能である．

しかし，上記の手法はいずれもデータベースからグラフで表現された部分パターンを抽出するものであり，医療や化学の分野で需要が高い立体構造の部分パターン抽出には利用できない．上記の手法の中で，gFSG[Kuramochi 02] はデータベースに含まれる立体構造の情報を用いて部分パターン抽出を行う．しかし，この手法は従来の FSG で抽出されるグラフ表現の部分パターンをいくつかの立体構造情報を使用してフィルタリングしたものにすぎず，立体構造の部分パターンを抽出することはできない．

本稿では，AGM アルゴリズムを使用して立体分子構造を解析するための手法を提案する．提案手法では AGM アルゴリズムで使用するグラフデータと頂点が 3 次元座標で表現された立体構造データを扱うことが可能であり，このデータの部分構造に含まれる特徴的な立体構造パターンを抽出する．この手法では立体構造の情報をグラフの辺ラベルに追加するため，多くの辺ラベルを持つ．そこで 2 章では従来の AGM アルゴリズムの概略を説明し，3 章で今回新たに提案する効率化手法とテストデータによる効率化手法の評価実験について述べる．ここで効率化を行った AGM アルゴリズムをベースにして，4 章では立体構造解析を行う提案手法を説明し，5 章では提案手法による特徴的パターン抽出を実際の市販薬物データベースを用いて行う．6 章では本研究と関連のある研究を紹介する．

2. AGM アルゴリズム

2·1 AGM アルゴリズムの概要

AGM アルゴリズムは，Apriori アルゴリズム [Agrawal 94] をグラフデータに拡張したアルゴリズムであり，グラフデータベース GD が与えられたとき，Apriori アルゴリズムと同様にユーザが指定した最小支持度 (minsup) と呼ばれる閾値を使用して，GD の中に最小支持度を上回る支持度で誘導部分グラフとして含まれるグラフのみを効率よく抽出するアルゴリズムである．グラフ G_s の支持度 $sup(G_s)$ は，

$$sup(G_s) = \frac{G_s\text{を誘導部分グラフとして含むグラフの数}}{GD\text{ に含まれるグラフの総数}}$$

で定義され，グラフ G_s の支持度 $sup(G_s)$ が最小支持度を上回る場合，グラフ G_s を多頻度グラフと呼ぶ．

```
// GD:グラフデータベース
// F_k:頂点数 k の多頻度グラフの集合
// Ĉ_{k+1}:頂点数 k の多頻度グラフを合成したものの集合
// C_k:頂点数 k の多頻度グラフの候補の集合
// minsup:最小支持度 (閾値)
1) F_1 = { Frequent subgraph of size=1} ;
2) for(k = 1; F_k ≠ ∅; k++) do begin
3)    Ĉ_{k+1} =apriori-gen-join(F_k);
4)    C_{k+1} =apriori-gen-prune(Ĉ_{k+1});
5)    count(GD, C_{k+1});
6)    F_{k+1} ={ c_{k+1} ∈ C_{k+1} | sup(G(c_{k+1})) ≥ minsup};
7) end
8) Answer= ⋃_k F_k;
```

図 1 AGM アルゴリズム

AGM アルゴリズムの探索はグラフの頂点数をレベルとして，頂点数が 1 の多頻度グラフから逐次的に頂点数が多い多頻度グラフをレベルワイズに抽出する．図 1 に AGM アルゴリズムの概略を示す．はじめに，頂点数が 1 の多頻度グラフをデータベースより抽出し，それを F_1 に代入する．次に，関数 apriori-gen-join では，頂点数が k の多頻度グラフから頂点数 $k+1$ の多頻度グラフの候補を生成し，それを $\hat{C}_{k+1}$ に代入する．次に，関数 apriori-gen-prune では $\hat{C}_{k+1}$ に格納されている多頻度グラフの各候補について，多頻度グラフであるための必要条件を調べる．この条件を調べることで多頻度グラフの候補の数を絞りこむ．絞りこみで残った多頻度グラフの候補のみを C_{k+1} に格納する．次に，関数 count ではグラフデータベース GD にアクセスして，C_{k+1} の各要素の支持度を求める．C_{k+1} の各要素の支持度が最小支持度を上回る場合は，そのグラフを多頻度グラフとし，それを F_{k+1} に格納する．以上の操作を F_k が空集合になるまで繰り返し，グラフデータベース GD に含まれる多頻度グラフをすべて抽出する．

2·2 AGM アルゴリズムの詳細

AGM アルゴリズムで扱うグラフは頂点，辺に種類を表すラベルを持ち，以下のように定義される．頂点の集合 $V(G)$，辺の集合 $E(G)$，頂点のラベル集合 $L_V(V(G))$，辺のラベル集合 $L_E(E(G))$ が

$$\begin{aligned} V(G) &= \{\ v_1, v_2, ..., v_k\ \}\ , \\ E(G) &= \{\ e_h = (v_i, v_j) | v_i, v_j \in V(G), i \neq j\}\ , \\ L_V(V(G)) &= \{lb(v_i) | \forall v_i \in V(G)\ \}\ , \\ L_E(E(G)) &= \{lb(e_h) | \forall e_h \in E(G)\ \} \end{aligned}$$

と与えられたとき，グラフ G は

$$G = (V(G), E(G), L_V(V(G)), L_E(E(G)))$$

と表現される．ここで，頂点の数 $|V(G)| = k$ をグラフ G の大きさとする．$lb(v_i)$ および $lb(e_h)$ はそれぞれ 頂点 v_i

のラベル，辺 e_h のラベルである．頂点ラベル $lb(v_i)$ および辺ラベル $lb(e_h)$ にはそれぞれ $num(lb(v_i))$,$num(lb(e_h))$ によって自然数を以下のように割り当てる．

§1 ラベル間の順序関係

グラフデータベースが与えられたとき，それに含まれる頂点ラベル lb_i を持つ頂点の数を $avg(lb_i)$ とする．$avg(lb_i)$ が少ないものから自然数を昇順に割り当てると，生成されるグラフ数は少なくなる [猪口 01]．つまり，

$$\text{if } avg(lb_i) < avg(lb_j) \text{ then } num(lb_i) < num(lb_j) \quad \text{for } i,j = 1,...,|L_V(V(G))|, i \neq j$$

とする．辺のラベル lb_i に割り当てる自然数も同様に，

$$\text{if } avg(lb_i) < avg(lb_j) \text{ then } num(lb_i) < num(lb_j) \quad \text{for } i,j = 1,...,|L_E(E(G))|, i \neq j$$

とする．

§2 隣接行列

頂点数 k のグラフ $G = (V(G), E(G), L_V(V(G)), L_E(E(G)))$ が与えられたとき，隣接行列 X_k の (i,j) 要素 $x_{i,j}$ は

$$x_{i,j} = \begin{cases} num(lb(e_h)) & \text{if } e_h = (v_i, v_j) \in E(G) \\ 0 & \text{if } (v_i, v_j) \notin E(G) \end{cases}$$

で与えられる．さらに，グラフ G は頂点ラベルに割り当てられた自然数によって，

$$num(lb(v_i)) \leq num(lb(v_{i+1})) \text{ for } i = 1,...,k-1$$

の条件を満たすように行と列をソートする．

2·3 多頻度グラフの候補生成 (join 部)

多頻度グラフの候補生成は，頂点数が k の多頻度グラフを合成して頂点数 $k+1$ の多頻度グラフの候補を作成する join 部と，合成された多頻度グラフの候補が多頻度グラフになるための必要条件を満たすかどうかを調べる prune 部の 2 つの部分から成り立つ．join 部では以下の条件を満たすように多頻度グラフの候補 $G(Z_{k+1})$ を順に生成していく．

条件 1 頂点数が k の多頻度グラフを 2 つ考え，その隣接行列を X_k，Y_k とする． X_k，Y_k の k 行及び k 列以外の要素が全て等しいとき，すなわち各グラフの第 k 頂点を除いてグラフ表現が等しいとき，以下のように X_k，Y_k を結合し，頂点数 $k+1$ の 隣接行列 Z_{k+1} を生成する．

$$X_k = \begin{pmatrix} X_{k-1} & \boldsymbol{x}_1 \\ \boldsymbol{x}_2^T & 0 \end{pmatrix}, Y_k = \begin{pmatrix} X_{k-1} & \boldsymbol{y}_1 \\ \boldsymbol{y}_2^T & 0 \end{pmatrix}$$

$$Z_{k+1} = \begin{pmatrix} X_{k-1} & \boldsymbol{x}_1 & \boldsymbol{y}_1 \\ \boldsymbol{x}_2^T & 0 & z_{k,k+1} \\ \boldsymbol{y}_2^T & z_{k+1,k} & 0 \end{pmatrix}$$

ここで，X_{k-1} は頂点数 $k-1$ のグラフの隣接行列，$\boldsymbol{x}_i, \boldsymbol{y}_i (i = 1,2)$ は $(k-1) \times 1$ の縦ベクトルである．$G(X_k)$, $G(Y_k)$ をそれぞれ $G(Z_{k+1})$ の第 1 生成グラフ，第 2 生成グラフと呼ぶ．

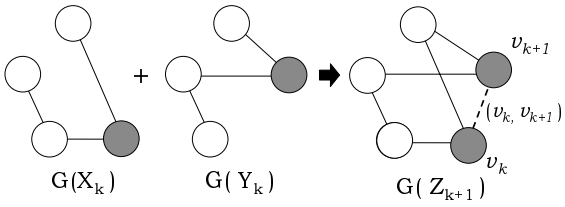


図 2 多頻度グラフの候補生成例

条件 2 $i = 1, \cdots, k-1$ として，生成される $G(Z_{k+1})$ の頂点ラベルには以下の条件がある．

$$\begin{aligned} lb(v_i \in V(G(Z_{k+1}))) &= lb(v_i \in V(G(X_k))) \\ &= lb(v_i \in V(G(Y_k))), \\ num(lb(v_i \in V(G(X_k)))) &\leq num(lb(v_{i+1} \in V(G(X_k)))), \\ lb(v_k \in V(G(Z_{k+1}))) &= lb(v_k \in V(G(X_k))), \\ lb(v_{k+1} \in V(G(Z_{k+1}))) &= lb(v_k \in V(G(Y_k))), \\ num(lb(v_k \in V(G(X_k)))) &\leq num(lb(v_k \in V(G(Y_k)))) \end{aligned}$$

条件 1, 条件 2 による多頻度グラフの候補生成の例を図 2 に示す．$G(X_k)$，$G(Y_k)$ の各頂点は，白色が同じグラフ表現 $G(X_{k-1})$ の頂点，黒色が各グラフの k 頂点を表す．このとき，生成される $G(Z_{k+1})$ の頂点 v_k と頂点 v_{k+1} はそれぞれ $G(X_k)$，$G(Y_k)$ の k 頂点であるため，この 2 頂点間の辺 (v_k, v_{k+1}) は $G(X_k)$，$G(Y_k)$ より作成できない．つまり，隣接行列 Z_{k+1} の $(k, k+1)$ 要素 $z_{k,k+1}$ および $(k+1, k)$ 要素 $z_{k+1,k}$ は X_k, Y_k から決定することはできない．

条件 3 そこで，隣接行列 Z_{k+1} は以下の条件を満たすものすべてが作られる．すなわち，

$$\begin{gathered} \hat{C}_{k+1} \leftarrow G(Z_{k+1}) \\ \text{where } z_{k,k+1} = lb1 \text{ and } z_{k+1,k} = lb2, \\ \forall\ lb1, 0 \leq lb1 \leq |L_E(E(G))| \text{ and} \\ \forall\ lb2, 0 \leq lb2 \leq |L_E(E(G))| \end{gathered}$$

である．有向グラフの場合は $(|L_E(E(G))|+1)^2$ 個のグラフが生成される．無向グラフの場合は $z_{k,k+1} = z_{k+1,k}$ であるため，$(|L_E(E(G))|+1)$ 個のグラフが生成される．

条件 4 ここでグラフ $G(X_k)$ と $G(Y_k)$ の第 k 頂点のラベルが等しい場合，$G(Y_k), G(X_k)$ をそれぞれ第 1 生成グラフ，第 2 生成グラフとして 2 つのグラフを結合した場合，このグラフは冗長である．そこで，このような冗長な生成を避けるため，以下の関係にある場合にのみグラフを結合する．

CODE(第 1 生成グラフ) $\leq$ CODE(第 2 生成グラフ)

以上の 4 つの条件のもとで生成されるグラフを**正規形** (normal form) と呼ぶ．

2·4 多頻度グラフの候補生成 (prune 部)

前節の join 部で合成された多頻度グラフの候補 $G(Z_{k+1}) \in \hat{C}_{k+1}$ が多頻度グラフであるための必要条件は，$G(Z_{k+1})$

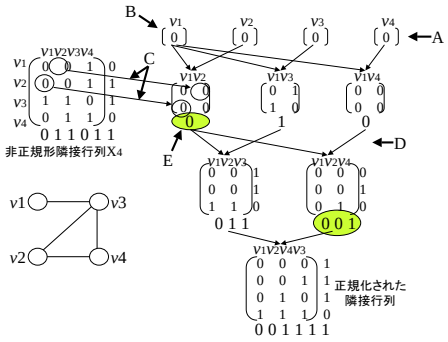


図 3 正規化の例

の全ての誘導部分グラフが多頻度グラフであることである．そこで，この必要条件と等価である以下の必要条件を調べる．

誘導部分グラフの必要条件

グラフ $G(Z_{k+1})$ が多頻度グラフであるための必要条件は，$G(Z_{k+1})$ の第 i 頂点 $(1 \leq i \leq k-1)$ を除去してできるグラフが全て多頻度グラフであることである．

先にも述べたように，このアルゴリズムでは正規形の隣接行列しか探索生成しないために，第 i 頂点を開放除去したグラフの隣接行列が正規形でなければ，それが多頻度グラフであるかを過去の探索から容易にチェックする事ができない．よって，非正規形の隣接行列を正規化する手法が必要である．

正規化の具体例を図 3 の非正規形の隣接行列 X_4 の正規化で示す．(A) はじめに頂点が 1 つからなる X_4 の部分グラフの隣接行列を考える．(B) 多数ある正規形の中で，最終的に 1 つ正規形が見つかれば十分なので，結合の組み合わせを限定し，v_1 を元にして結合を行う．(C) 結合により得られない情報，例えば，v_1, v_2 からなる隣接行列の (1,2) 要素，(2,1) 要素は元の隣接行列 X_4 の x_{12} 及び x_{21} から補う．(D) 次に頂点数が 2 の隣接行列の結合を行う．(E) このとき隣接行列のコードが最小の行列を第 1 生成行列とする．ここではコードが 0 の隣接行列が 2 つあるが，どちらか一方を選択する．以下，順に繰り返し，非正規形の隣接行列 X_4 を再構築し正規化された行列を得る．

上記の方法によって頂点を除去した全てのグラフが過去の探索結果から多頻度グラフであることを確定できれば $G(Z_{k+1})$ は多頻度グラフの候補となり，C_{k+1} に格納される．

全ての多頻度グラフの候補を取り出した後，実際にデータベースをスキャンして，それらの支持度を求める．しかし，異なる正規形のグラフでも同型グラフが存在する場合があるため，支持度を計算する前に正準形を求める処理が必要となる．正準形を求める処理と支持度の計算方法については文献 [Inokuchi 00] を参照されたい．C_{k+1} の各要素について多頻度グラフの候補の支持度を計算して，その支持度が最小支持度を上回る場合には，その多頻度グラフの候補を多頻度グラフとして，F_{k+1} に格納する．つまり，

$$\text{if } \forall c_{k+1} \in C_{k+1} \text{ and } sup(G(c_{k+1})) \geq minsup$$
$$\text{then } F_{k+1} \leftarrow c_{k+1}$$

である．

3. AGM アルゴリズムの効率化

3·1 AGM アルゴリズムの高速化

2·3 節の図 2 のように，頂点数が k の多頻度グラフ $G(X_k)$ と $G(Y_k)$ を結合し，頂点数が $k+1$ の多頻度グラフの候補 $G(Z_{k+1})$ を生成する場合を考える．グラフ $G(Z_{k+1})$ の要素 $z_{k,k+1}$ と $z_{k+1,k}$ は X_k, Y_k から決定することができないため，辺のラベル数 $|L_E(E(G))|$ に応じて条件 3 で示された数のグラフ $G(Z_{k+1})$ が生成される．そして，生成されたグラフ $G(Z_{k+1})$ に含まれるすべての誘導部分グラフが多頻度グラフであることを確認するために，それと等価な必要条件を確認している．この方法では合成されたすべてのグラフ $G(Z_{k+1})$ について，その誘導部分グラフを正規化する必要があるため多くの計算時間を要する．

そこで，$G(Z_{k+1})$ の頂点 v_k, v_{k+1} とその頂点間の辺 $e_{k,k+1} = (v_k, v_{k+1})$ から構成される頂点数 2 のグラフ $G(Z_{S2})$ に着目する．$G(Z_{k+1})$ が多頻度グラフになるためには，その誘導部分グラフである $G(Z_{S2})$ も多頻度グラフであることが必要条件の 1 つである．つまり，$G(Z_{S2})$ が多頻度グラフでない場合は，$G(Z_{k+1})$ も多頻度グラフになり得ない．そのため，このような合成を行わない．そこで，$k \geq 2$ では条件 3 の代わりに以下の条件 3' を使用する．

条件 3'

$$\text{If } k \geq 2 \text{ and } \forall\ G(Z_{S2}) \in F_2 \text{ then } \hat{C}_{k+1} \leftarrow G(Z_{k+1}),$$
$$\text{where } V(G(Z_{S2})) = \{v_k, v_{k+1}\}$$

この条件 3' は F_2 にあるグラフ $G(Z_{S2})$ のみを合成するため，そのグラフの頂点 v_k, v_{k+1} と辺 $e_{k,k+1}$ の各ラベルも限定されている．そのため，条件 3' は条件 3 よりも制約の厳しい条件である．提案手法はこの条件 3' と条件 1, 条件 2, 条件 4 を用いて多頻度グラフの候補 $G(Z_{k+1})$ を生成するものである．この提案手法を AGM' アルゴリズムと呼ぶ．AGM' アルゴリズムではこの 4 つの条件によって作成されたグラフを改めて正規形と定義する．

AGM' アルゴリズムで合成された $G(Z_{k+1}) \in \hat{C}_{k+1}$ は 2·4 節で述べた必要条件を確認するために $G(Z_{k+1})$ の誘導部分グラフを正規化する必要がある．しかし，AGM' アルゴリズムで合成される多頻度グラフの候補数 $|\hat{C}_{k+1}|$ は，従来の AGM アルゴリズムで合成されるものよりも要素数が少ないため，正規化の計算時間を短縮できる．

また，条件 3' は必要条件の一つであるため，2･4 節ですべての必要条件が確認された後に残るグラフの数 $|C_{k+1}|$ は，AGM アルゴリズム，AGM' アルゴリズムともに同じである．

3･2 AGM' アルゴリズムの計算時間評価

本研究の提案手法を C 言語で実装し，CPU が PentiumIII 1GHz, メモリが 1.5GB 搭載された計算機を使用して，評価実験を行った．

実験で用いたグラフデータベースは表 1 に示すパラメータとそのデフォルト値を基準にランダムに作成する．まず，平均 $|T|$，分散 1 のガウス分布を用いて各グラフデータのサイズを決める．各グラフデータ中の頂点のラベルは等確率で決定する．次に存在確率 p をもとに頂点間に辺を結ぶ．辺のラベルも等確率で決定する．

しかし，この方法でランダムに作成したグラフデータベースは多頻度グラフになる共通した誘導部分グラフを含んでいない．そこで，グラフデータベースが下記の基本パターンを共通した誘導部分グラフとして持つように，グラフデータベースの情報を上書きする．基本パターンはグラフデータベースと同様に，表 1 に示すパラメータから平均サイズ $|I|$ のものを L 個作る．データベースの各グラフは，1 つの基本パターンを誘導部分グラフとして持つようにグラフデータベースの情報を上書きする．どの基本パターンをどの場所に埋め込むかはランダムに決めている．このようにして得られたグラフデータベースを使用して部分パターンを抽出し，その計算時間を評価した．

図 4 は辺のラベル数 $|L_E|$, 図 5 は頂点のラベル数 $|L_V|$, 図 6 はグラフデータの平均頂点数 $|T|$ を変化させた場合の，多頻度グラフの候補生成に要する計算時間を評価した結果である．AGM' アルゴリズムは AGM アルゴリズムの条件 3 の代わりに条件 3' を用いた提案手法を表す．また，$|\hat{C}_{sum}|$ は合成されたグラフの数，$|C_{sum}|$ は正規形の多頻度グラフ候補の数を頂点数が 1 から最大のものまですべてを合計したものとする．表 2 は辺のラベル数 $|L_E|$, 表 3 は頂点のラベル数 $|L_V|$ 表 4 はグラフデータの平均頂点数 $|T|$ を変化させた場合の，$|\hat{C}_{sum}|$,$|C_{sum}|$ の生成数を示す．なお，$|C_{sum}|$ の生成数は AGM アルゴリズムと AGM' アルゴリズムの両手法とも同じであるため，一つにまとめている．

表 1 人工データのパラメータとそのデフォルト値

パラメータ	意味	デフォルト値
D	GD に含まれるグラフ数	1000
$\|T\|$	グラフデータの平均頂点数	15
L	基本パターンの数	8
$\|I\|$	基本パターンの平均頂点数	7
$\|L_V\|$	頂点ラベルの種類数	5
$\|L_E\|$	辺ラベル種類数	5
p	頂点間に辺が存在する確率	4%
$minsup$	最小支持度	10%

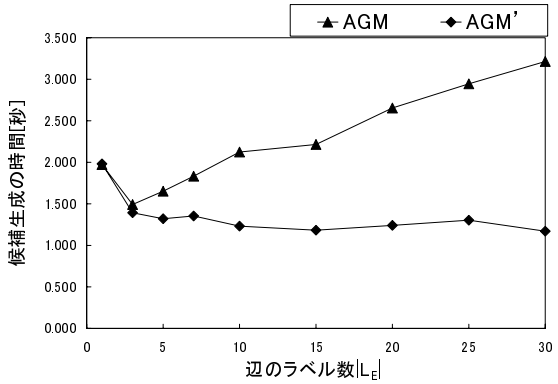


図 4 $|L_E|$ v.s. 多頻度グラフの候補生成の計算時間

表 2 $|L_E|$ v.s. $|\hat{C}_{sum}|$,$|C_{sum}|$

	$\|\hat{C}_{sum}\|$		$\|C_{sum}\|$
$\|L_E\|$	AGM	AGM'	
1	68,950	68,950	14,532
3	103,932	92,280	11,854
5	140,838	103,217	12,389
7	171,824	114,644	14,039
10	217,965	104,657	14,661
15	237,200	101,940	16,875
20	297,696	108,151	18,092
25	330,096	102,529	20,608
30	363,816	87,164	20,588

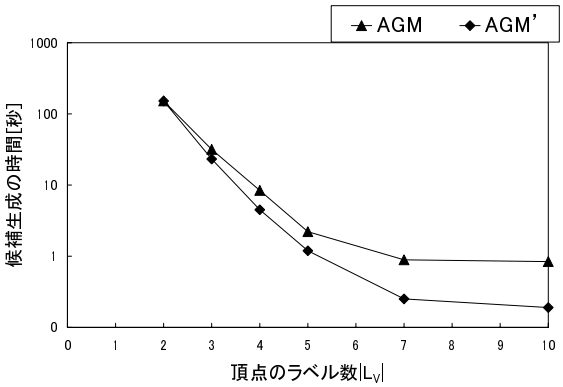


図 5 $|L_V|$ v.s. 多頻度グラフの候補生成の計算時間

表 3 $|L_V|$ v.s. $|\hat{C}_{sum}|$,$|C_{sum}|$

	$\|\hat{C}_{sum}\|$		$\|C_{sum}\|$
$\|L_V\|$	AGM	AGM'	
2	4,331,264	4,310,981	36,506
3	1,522,880	1,079,426	28,572
4	658,476	262,046	33,582
5	237,200	101,940	16,875
6	160,030	20,863	6,686
7	104,192	17,282	5,308
8	99,034	7,526	4,170
9	114,920	6,294	4,714
10	95,552	7,771	5,762

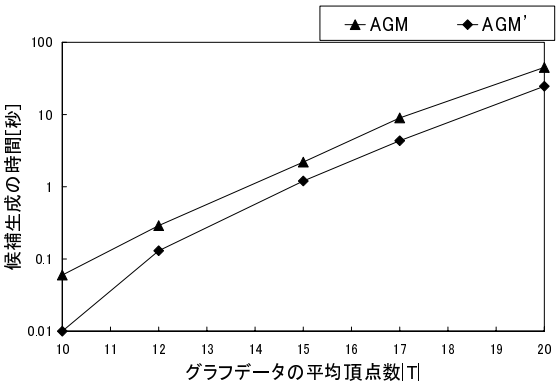


図 6 $|T|$ v.s. 多頻度グラフの候補生成の計算時間

表 4 $|T|$ v.s. $|\hat{C}_{sum}|$,$|C_{sum}|$

	$|\hat{C}_{sum}|$		$|C_{sum}|$
$|T|$	AGM	AGM'	
10	8,640	1,146	917
12	43,520	13,193	5,089
15	237,200	101,940	16,875
17	666,320	287,525	29,814
20	2,323,824	1,141,953	53,902

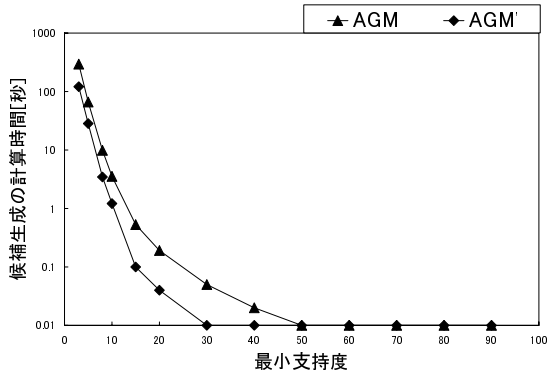


図 7 $p = 1\%$ の minsup v.s. 多頻度グラフの候補生成の計算時間

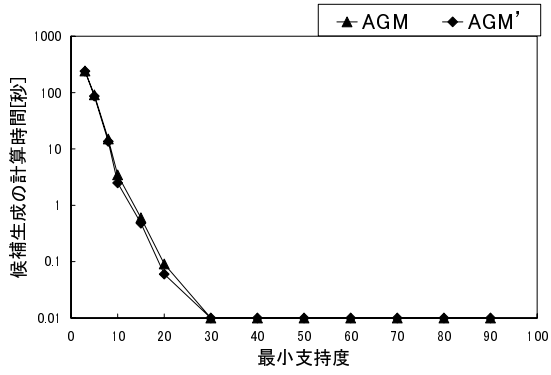


図 8 $p = 30\%$ の minsup v.s. 多頻度グラフの候補生成の計算時間

AGM アルゴリズムと AGM' アルゴリズムを比較すると，辺のラベル数 $|L_E|$ および頂点のラベル数 $|L_V|$ が少ない場合に，AGM' アルゴリズムは多頻度グラフの候補生成における候補の生成数 $|\hat{C}_{sum}|$ と計算時間をほとんど削減できていない．これはラベル数が少なく，グラフデータベースに多くの多頻度グラフが含まれているためである．しかし，辺のラベル数 $|L_E|$ および頂点のラベル数 $|L_V|$ が多い場合には，AGM' アルゴリズムは多頻度グラフの候補生成において，生成数 $|\hat{C}_{sum}|$ と計算時間を削減できている．これは，AGM' アルゴリズムの条件 3' は辺ラベルとその両端の 2 つの頂点ラベルの組み合わせに制約があるため，ラベルの種類数が多くなるほどその制約は厳しくなるためと考えられる．

また，図 6 のようにグラフデータの平均頂点数 $|T|$ を増やした場合にも，AGM' アルゴリズムの条件 3' は多頻度グラフの候補生成にかかる計算時間を削減できている．これは，平均頂点数がより大きなグラフデータに対しても，AGM' アルゴリズムが計算時間を削減することが期待できることを示している．

図 7 は各頂点間に存在する辺の割合を $p = 1\%$ に, 図 8 は $p = 30\%$ に固定し，最小支持度 $minsup$ を変化させた場合の多頻度グラフの候補生成に要する計算時間を評価した結果である．この 2 つの場合を比較すると，AGM' アルゴリズムが有効な場合は頂点間の辺の存在する確率 p が低いグラフ (疎グラフ) でかつ，最小支持度 $minsup$ が低い時である．

4. AGM' アルゴリズムを用いた立体構造パターン抽出

4·1 立体構造データの表現方法

AGM アルゴリズムはグラフ表現のデータを扱い，データベースから部分パターンを抽出するが，立体構造の情報を含むグラフデータを扱うことはできない．そこで，本稿では立体構造のグラフデータを扱い，データベースに共通して現れる多頻度の部分パターンを抽出する手法を提案する．提案手法では図 9 に示すようなグラフの各頂点が 3 次元座標で表された立体構造を扱う．この立体構造から各頂点間の距離を表す距離行列を計算し，計算された距離を離散化する．この距離行列の計算と離散化の方法については以下で詳しく述べる．提案手法では，離散化した立体構造の情報をグラフの新たな辺ラベルとして扱うため，辺ラベル数が多数になる．そこで，立体構造パターンの抽出はラベル数が多い場合に高速な AGM' アルゴリズムを用いる．

距離行列の計算は，頂点が 3 次元座標で表された立体構造から各頂点間の距離を計算し，立体構造を図 10 の左側に示す距離行列 [Kato 97] で表現する．距離を用いて立体構造を解析する手法を用いる利点は，距離以外の立体構造データを使用しないため，立体構造モデルの回

転や移動など座標系の影響を受けないこと，立体構造の同一性の判定は，距離行列の各要素を比較することで実現できるため，容易に判定が行えることである．

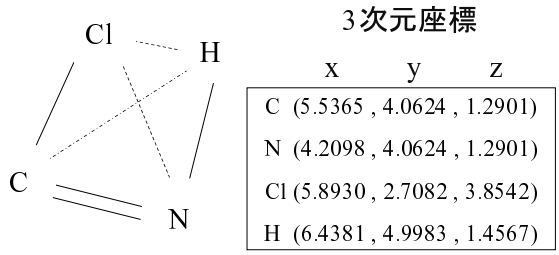


図 9 3 次元座標で表現される立体構造

4·2 離散化による前処理方法

AGM' アルゴリズムで扱えるグラフは，C, H などの頂点ラベルや単結合, 二重結合などの辺ラベルのように離散的なラベルで表現されることを前提としているため，頂点間の距離のように連続値をそのままラベルとして扱うことはできない．そこで，本稿では連続値からなる距離値をある閾値で区切って離散化を行い，離散値に変換する．例えば，図 10 左側の距離行列の各要素 dist を以下の閾値によって離散化した場合，図 10 右側となる．

$$\text{離散値} = \begin{cases} a & (1.2 \leq dist < 1.8) \\ b & (1.8 \leq dist < 2.4) \\ c & (2.4 \leq dist < 3.0) \end{cases} \quad (1)$$

ここで離散化した距離値は図 11 に示すように，隣接行列の辺ラベルの情報に追加し，AGM' アルゴリズムで解析可能なグラフデータに変換する．提案手法は距離行列の計算とその離散化を行い，AGM' アルゴリズムで立体構造の部分パターンを抽出する手法である．提案手法を AGM'-3D と呼ぶ．

5. 立体構造パターン抽出と生理活性の相関解析

ここでは，MDDR(MDL Drug Data Report)[MDL 01] データベースに含まれるドーパミンアンタゴニスト活性を持つ化合物データを対象とし，MDDR 3D データベース所載の 3 次元座標値を立体構造データとして使用し，AGM'-3D で立体構造の部分パターンを抽出した．MDDR データベースに含まれるドーパミンアンタゴニスト活性は作用する受容体によって, D1, D2, D3, D4 の

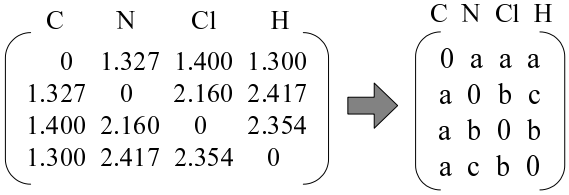


図 10 距離行列と離散化

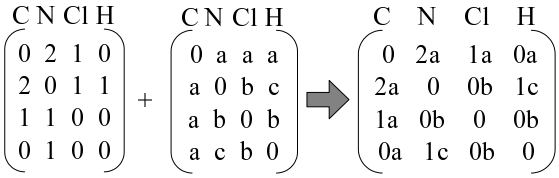


図 11 隣接行列に辺の情報を追加

表 5 検証用データと学習用データの化合物数

	D1	D2	D3	D4	合計
検証用データ	20	40	24	60	144
学習用データ	153	390	230	514	1287
合計	173	430	254	574	1431

4 種類あり，その活性をもつ化合物はそれぞれ 173 個,430 個,254 個,574 個である．このデータから，表 5 のように全体の 10%にあたる 144 個の化合物を検証用データとして除外し，残り 1287 個の化合物からなる学習用データを対象として AGM'-3D で立体構造の部分パターンを抽出した．

次に，学習用データから AGM'-3D で抽出した立体構造の部分パターン，原子団寄与法で計算された LogP 値を用いて立体構造とドーパミンアンタゴニスト活性の相関解析を行った．LogP 値は分子と分子の相互作用の大小を考える重要な目安であり，薬学の分野の定量的構造活性相関 (QSAR) において必ず用いられる物性値であり，化学での理解度の容易さのために，LogP 値も使用して解析した．学習用データの相関解析で得られた分類規則は，検証用データでこの分類誤差を評価した．

5·1 ドーパミン化合物からの立体構造パターン抽出

学習用データに含まれる 1287 個の化合物を対象として，立体構造の部分パターン抽出を行った．学習用データから部分パターンを抽出する前に，仮想リンクの追加と距離行列の離散化を行った．まず最初に，それぞれの方法について説明する．

仮想リンクは実際には結合が存在しない辺に，仮想的な辺のラベルをつけたものである．例えば，図 12 の場合は C と H の原子間に結合はないが，2 つの結合を通じてつながっているため，P2 という仮想的な辺ラベルを追加する．同様に Cl と H は 3 つの結合で接続しているので，P3 の辺ラベルを追加する．同様の方法で結合のない他の原子間にも仮想リンクを追加する．学習用データの化合物では，単結合や二重結合で接続している原子間の距離がほぼ決まっており，それによって仮想的な辺ラベル P2 や P3 の原子間距離も特定されやすい．そのため，学習用データの化合物には P2 から P33 の仮想リンクを追加した．

次に，距離行列の各要素の距離を離散化した．距離行列の距離は MDDR 3D データベース所載の 3 次元座標値から計算したものである．離散化の閾値の設定方法は，

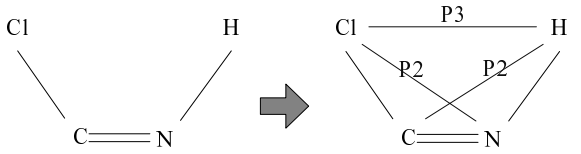


図 12 仮想リンクの追加

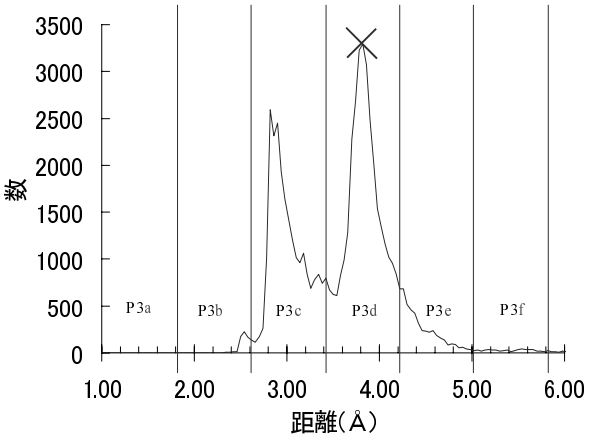


図 13 ヒストグラムと離散化

図 13 を用いて説明する．この図は学習用データに含まれる仮想的な辺ラベル P3 について，横軸の距離を 0.04 Åごとに区切ったヒストグラムであり，縦軸はデータ数を取っている．離散化閾値の設定方法は，このヒストグラムで最大値となる×印のついた点（モード）が基準になる．モードは辺ラベルごとに異なった値になる．各辺ラベルごとにモードを基準にして± 0.4 Åの点に最初の閾値を設定し，残りの閾値は一定間隔 0.8 Åごとに設定した．閾値の間隔はすべての辺ラベルで同じ値 0.8 Åに設定した．辺ラベルごとに異なったモードを基準に閾値を設定するのは，ヒストグラムの分布が異なっているためである．閾値の間隔が一定なのは，それが抽出する立体構造パターンに密接にかかわっているためであり，同一の立体構造とみなせる距離値 0.8 Åを離散化に使用した [Kato 97].

仮想リンクの追加と距離行列の離散化の 2 つの前処理によって，辺が持つ情報はもとの辺ラベル，仮想リンク，離散化された距離の 3 つになる．辺のラベルにはこの 3 つの情報によって新たに自然数を割り当てる．割り当てる自然数は 2･2･1 節で説明した辺ラベルの出現数によってきまる．表 6 は学習用データの辺ラベルに新たに割り当てられた自然数の一例である．単結合の辺ラベルは，仮想リンクを持たず距離が 1.14〜1.94 Åのものには 105 の自然数を，距離が 1.94〜2.74 Åのものには 64 の自然数を割り当てた．結合がない辺ラベルの場合は仮想リンク p2 と仮想リンク p3 の場合で距離の閾値が異なるが，同様の方法で自然数を割り当てた．このように前処理を行った学習用データのグラフは，原子の種類を表す頂点のラベル数が 12，結合の種類を表す辺のラベル数が 108 であった．

表 6 辺ラベルに割り当て直された自然数の一例

辺ラベル	仮想リンク	距離 (Å)	$num(lb)$
単結合	−−	1.14 〜 1.94	105
		1.94 〜 2.74	64
結合なし	p2	2.10 〜 2.90	108
		2.90 〜 3.70	79
	p3	2.62 〜 3.42	100
		3.42 〜 4.22	106
		4.22 〜 5.02	80

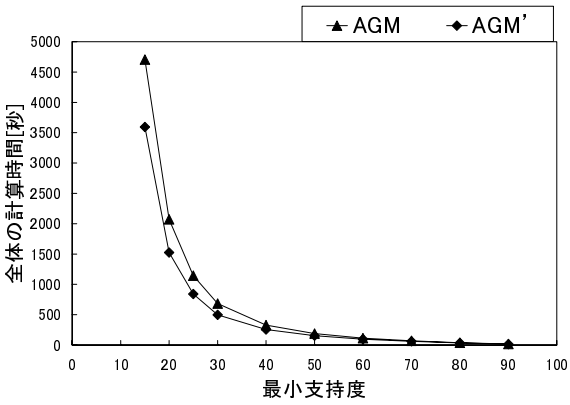


図 14 AGM と AGM' の計算時間

評価実験は立体構造のパターン抽出に要する計算時間を AGM アルゴリズムと AGM' アルゴリズムの場合で比較した．実験には 3･2 節と同じ，CPU が PentiumIII 1GHz, メモリが 1.5GB 搭載された計算機を使用した．図 14 は最小支持度を変化された場合の各アルゴリズムの計算時間を示す．この結果，AGM' アルゴリズムは AGM アルゴリズムより約 30%高速であることが確認できた．これは，AGM' アルゴリズムは辺のラベル数が多いと条件 3' による探索空間削減の効果が大きくなることの現れである．

本実験では多数の部分構造パターンが抽出されたが，活性に影響があると思われるものを χ^2 検定 [Brin 97] で評価した．表 7 は抽出されたある部分構造パターンに対して，学習用データを活性の種類と部分構造パターンを含むか含まないかによって分割し，分割された各ブロックのデータ数を数えるための分割表である．N は学習用データの総数で，C_1 は部分構造パターンを含む学習用データ数，C_2 は部分構造パターンを含まない学習用データ数，$O_j(j=1,\cdots,4)$ は活性が Dj を持つ学習用データ数，$C_{1,j}(j=1,\cdots,4)$ は活性が Dj を持つデータ

表 7 χ^2-検定の分割表

	D1	D2	D3	D4	合計
部分構造を含む	C_{11}	C_{12}	C_{13}	C_{14}	C_1
部分構造を含まない	C_{21}	C_{22}	C_{23}	C_{24}	C_2
学習用データ数	O_1	O_2	O_3	O_4	N

表 8 部分構造パターン (1) の分割表

	D1	D2	D3	D4	合計
部分構造を含む	91	196	129	148	564
部分構造を含まない	62	194	101	366	723
学習用データ数	153	390	230	514	1287

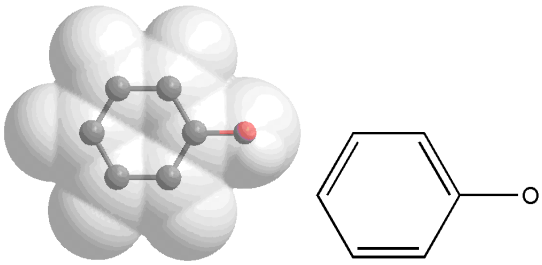


図 15 抽出された部分構造パターン (1)

の中で部分構造パターンを含む化合物数，$C_{2,j}$ は活性が Dj を持つデータの中で部分構造パターンを含まない化合物数である．学習用データから部分構造パターンを抽出したため，$O_1 = 153, O_2 = 390, O_3 = 230, O_4 = 514, N = 1287$ である．このとき，各部分構造パターンを含む学習用データの活性分布とすべての学習用データの活性分布を χ^2 検定で統計的に適合するかどうかを調べる式が以下の式 (2) である．

$$\chi^2 = \sum_{i,j} \frac{(C_{i,j} - E_{i,j})^2}{E_{i,j}}, E_{i,j} = \frac{C_i \times O_j}{N} \quad (2)$$

$C_{1,j}$ と $C_{2,j}$ の j に含む比率が O_j のそれと離れているほど χ^2 値は大きくなり，活性に影響のある部分構造パターンの特徴が抽出されていると考えられる．ここでは，実験で抽出した部分構造パターンの場合，自由度 3 の χ^2 分布になるため 有意水準が 1%となるχ^2 値は 11.345 である．

ここでは χ^2 値が大きな例を図 15, 図 16, 図 17 に示す．図 15, 図 16 は左側に部分構造パターンの立体構造モデルを示し，右側にグラフ表現を化学の表記方法に従って炭素原子を省略して示している．図 15 の部分構造パターン (1) は $C_{11} = 91, C_{12} = 196, C_{13} = 129, C_{14} = 148$ であり，表 8 の分割表が得られる．この分割表から式 (2) で χ^2 値を計算すると 82.0 になる．図 16 のパターン (2) は

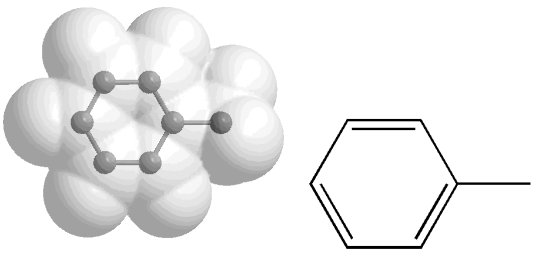
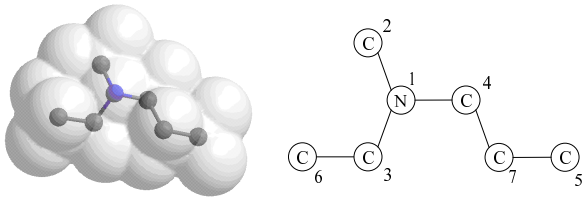


図 16 抽出された部分構造パターン (2)



辺ラベル／仮想リンク	距離 (Å)	辺 (頂点 ID のペア)
単結合/−−	1.14 ～ 1.94	(1,2)(1,3)(1,4) (3,6)(4,7)(5,7)
結合なし/p2	2.10 ～ 2.90	(1,6)(1,7)(2,3) (2,4)(3,4)(4,5)
結合なし/p3	2.62 ～ 3.42	(2,6)(3,7)
	3.42 ～ 4.22	(1,5)(2,7)(4,6)
結合なし/p4	3.94 ～ 4.74	(3,5)(6,7)
	4.74 ～ 5.54	(2,5)
結合なし/p5	5.34 ～ 6.14	(5,6)

図 17 抽出された部分構造パターン (3)

$C_{11} = 84, C_{12} = 263, C_{13} = 192, C_{14} = 398$ で χ^2 値が 49.7 である．パターン (1) の分子は活性が D4 の化合物に含まれる数が少なく，パターン (2) の分子は D3，D4 の化合物に多く含まれる．2 つのパターンはベンゼン環に接続している原子が 1 つ異なるだけであるが，それぞれの活性の特徴を表している．

また，図 17 は左側に部分構造パターンの立体構造モデルを示し，右側にグラフ表現を示している．パターン (3) はフレキシブル (flexible) な立体構造であるため，原子間の距離も図に示す．右側のグラフ表現では各原子に頂点 ID を割り当て，この頂点 ID のペアで辺を表現し，下側の表で辺の距離を示している．例えば，頂点 ID が 1 の N 原子と頂点 ID が 2 の C 原子間の辺は，単結合で距離が 1.14～1.94 Åであることを示す．このパターン (3) は $C_{11} = 25, C_{12} = 160, C_{13} = 83, C_{14} = 184$ で χ^2 値が 31.3 である．D1 の化合物に含まれる数が少なく，D2 の化合物に多く含まれるため，これらの活性に影響を与えるものと考えられる．

5·2 立体構造パターンと生理活性の相関解析

AGM'-3D によって得られる 3 次元構造の特徴を，AGM アルゴリズムによるグラフ表現の場合と比較して論じるため，分類規則学習法では標準的な手法である C4.5[Quinlan 93] を使用して，学習用データからドーパミンアンタゴニスト活性の分類規則を作成し，新規化合物を想定した検証用データでこの分類誤差の測定を行った．C4.5 は各化合物のベクトル情報から目的とするクラス属性（本実験ではドーパミンアンタゴニスト活性レベル）を分類する手法である．化合物のデータについて，実験 1 の AGM'-3D で抽出された 111 個の各パターンを部分構造に持つかどうか判定し，それぞれに「ある」,「なし」の記号を割り

表 9 C4.5 によるデータ分類の分布 (AGM'-3D)

C4.5 による分類→	D1	D2	D3	D4
D1	11	7		2
D2	1	29	5	5
D3		2	18	4
D4		7	1	52

当て，各化合物をベクトルで表す．このベクトル形式のデータと LogP 値を属性に，ドーパミンアンタゴニスト活性をクラスに設定して C4.5 の入力とした．この場合，各ドーパミンアンタゴニスト活性の分類誤差は 23.6 %となった．C4.5 によるデータの分類分布を表 9 に示す．横行は実際の活性の種類，縦列は C4.5 による分類結果を表し，対角部分の個数が多いほうが精度が高いことを表す．

また，同様の実験を AGM アルゴリズムに適用した．最小支持度 30%以上の化合物に含まれる部分パターンは，AGM'-3D よりも多い 275 個のパターンを抽出した．抽出したパターンは C4.5 の属性として使用した．この結果，分類誤差は 18.8 %となった．C4.5 によるデータの分類分布を表 10 に示す．

AGM アルゴリズムと AGM'-3D を用いて部分構造パターンの属性項目を生成し C4.5 で分類規則を得る両手法を，表 11 に示す部分構造パターン数と分類誤差について比較した．この場合，AGM'-3D の結果の方が分類精度が低い．これは，C4.5 で属性として使用した部分構造パターン数が AGM'-3D の場合は少ないためと思われる．

AGM'-3D で抽出した部分構造パターン数が少ないのは以下の 2 つ理由が考えられる．AGM'-3D で使用する辺ラベルは AGM アルゴリズムの辺ラベルを距離によってさらに細かく区切られたものである． AGM アルゴリズムのパターン数とそれに対応する AGM'-3D の立体構造のパターン数は 1 対複数になるが，部分構造パターンの抽出は同じ最小支持度を基準にして行ったため，これが 1 番目の理由と考えられる．2 番目の理由は，離散化の問題である．AGM'-3D はもともとは連続値である距離を離散化して取り扱っているため，距離が近く実際の立体構造も似ている化合物が，離散化によって異なる物質と判断される可能性がある．

これら 2 つの問題を回避するために，最小支持度を小さく設定してより多くのパターンを抽出すること，創薬へのヒントにするために化学の背景知識を利用して距離を離散化することが今後の課題である．

6. 関 連 研 究

グラフデータからの特徴的な部分パターン抽出は医学や化学の分野でも研究がされており，化合物に特化した手法が多く提案されている．その代表的な手法は化合物の分子構造骨格を解析するもの [Bemis 96, Bemis 99]

表 10 C4.5 によるデータ分類の分布 (AGM)

C4.5 による分類→	D1	D2	D3	D4
D1	12	8		
D2	1	33	2	4
D3		1	21	2
D4	1	5	3	51

表 11 AGM'-3D と AGM の結果比較

	部分パターン数	分類誤差
AGM'-3D	111	23.6%
AGM	275	18.8%

などグラフ表現の部分パターンを抽出するものであり，立体構造の部分パターンは抽出できない．また，立体構造の情報を扱う手法では，Voronoi 平面を使用して解析を行うもの [Chuman 98, Chuman 00] が提案されている．この手法は前処理で立体構造を Voronoi 平面で表現し，Voronoi 平面上で各原子が隣接しているかの情報に変換する．そのため，抽出される部分パターンの各原子は Voronoi 平面上で隣接しているかの情報しか得られない．

また，化合物の立体構造の類似性を検索する手法は多く提案されている．これらの手法はある 1 つの化合物と立体構造が類似したものをデータベースから検索する手法であり，類似性の判定には提案手法と同様に原子間の距離を用いるもの [Kato 97, Kato 01] が多い．しかし，これらの手法は類似性の検索には有効であるが，立体構造のパターン抽出は困難であると考えられる．

7. お わ り に

本稿では AGM アルゴリズムを高速化した AGM' アルゴリズムと立体構造パターンを抽出する AGM'-3D 手法を提案した．AGM'-3D 手法では立体構造の情報をグラフの新たな辺ラベルとして扱うため，辺ラベルの数が多数になる．このためラベル数が多い場合に有効な AGM' アルゴリズムによっての高速化が得られた．実際の市販薬物データベースを用いて立体構造パターンを抽出し，立体構造と生理活性の相関解析の応用に有効であることが確認できた．

今後の課題は，最小支持度を小さく設定してより多くのパターンを抽出すること，化学の背景知識を利用し距離を離散化することである．

謝 辞

本研究において，化学データの提供および御指導してくださった豊橋技術科学大学知識情報工学系の高橋 由雅教授に感謝いたします．

◇ 参 考 文 献 ◇

[Agrawal 94] Agrawal, R. and Srikant, R.: Fast Algorithms for Mining Association Rules, in Bocca, J. B., Jarke, M., and Zaniolo, C. eds., *Proc. of the 20th Very Large Data Bases Conference*, pp. 487–499, Morgan Kaufmann (1994)

[Bemis 96] Bemis, G. W. and Murcko, M. A.: The Properties of Known Drugs. 1. Molecular Frameworks, *J. Med. Chem.*, Vol. 39, pp. 2887 –2893 (1996)

[Bemis 99] Bemis, G. W. and Murcko, M. A.: The Properties of Known Drugs. 2. Side Chains, *J. Med. Chem.*, Vol. 42, pp. 5095 –5099 (1999)

[Brin 97] Brin, S., Motwani, R., and Silverstein, C.: Beyond market baskets: generalizing association rules to correlations, Proc. of ACM SIGMOD Conference (1997)

[Chuman 98] Chuman, H., Karasawa, M., and Fujita, T.: A Novel Three-Dimensional QSAR Procedure: Voronoi Field Analysis, *Quant. Struct.-Act. Relat.*, Vol. 17, pp. 313–326 (1998)

[Chuman 00] Chuman, H., Karasawa, M., Sasaki, M., Nagashima, U., Nishimura, K., and Fujita, T.: Three-Dimensional Structure-Activity Relationships of Synthetic Pyrethroids:2.Tree-Dimentional and Classical QSAR Studies, *Quant. Struct.-Act. Relat.*, Vol. 19, pp. 455–467 (2000)

[Cook 94] Cook, D. J. and Holder, L. B.: Substructure Discovery Using Minimum Description Length and Background Knowledge, *Journal of Artificial Intelligence Research*, Vol. 1, pp. 231–255 (1994)

[Dehaspe 98] Dehaspe, L., Toivonen, H., and King, R. D.: Finding frequent substructures in chemical compounds, in *Proc. of the 4th International Conference on Knowledge Discovery and Data Mining*, pp. 30–36 (1998)

[DeRaedt 01] DeRaedt, L. and Kramer, S.: The Levelwise Version Space Algorithm and its Application to Molecular Fragment Finding, in *Proc. of the 17th IJCAI*, pp. 853–859 (2001)

[Inokuchi 00] Inokuchi, A., Washio, T., and Motoda, H.: An Apriori-Based Algorithm for Mining Frequent Substructures from Graph Data, in *Proc. of the 4th European Conference on Principles and Practice of Knowledge Discovery in Databases*, pp. 13–23 (2000)

[Inokuchi 02] Inokuchi, A., Washio, T., Nishimura, Y., and Motoda, H.: General Framework for Mining Frequent Structures in Graphs, in *Proc. of the International Workshop on Active Mining*, pp. 23–30 (2002)

[Kato 97] Kato, H. and Takahashi, Y.: An Approach to Three-Dimensional Motif Finding in Proteins, Proc. Genome Informatics Workshop IV, pp.315-324 (1993)

[Kato 01] Kato, H. and Takahashi, Y.: Automated identification of three-dimensional common structural features of proteins, *J. Chem. Software*, Vol. 7, No. 4, pp. 161–170 (2001)

[Kuramochi 01] Kuramochi, M. and Karypis, G.: Frequent Subgraph Discovery, in *Proc. of the 1st IEEE International Conference on Data Mining*, pp. 313–320 (2001)

[Kuramochi 02] Kuramochi, M. and Karypis, G.: Discovering Frequent Geometric Subgraphs, in *Proc. of the 2002 IEEE International Conference on Data Mining*, pp. 258–265 (2002)

[MDL 01] MDL Drug Data Report, MDL, ver2001.1 (2001)

[Motoda 97] Motoda, H. and Yoshida, K.: Machine Learning Techniques to Make Computers Easier to Use, in *Proc. of the 15th International Joint Conference on Artficical Intelligence*, Vol. 2, pp. 1622–1631 (1997)

[Quinlan 93] Quinlan, J. R.: C4.5:Programs For Machine Learning (1993)

[Yan 02] Yan, X. and Han, J.: gSpan: Graph-Based Substructure Pattern Mining, in *Proc. of the 2002 IEEE International Conference on Data Mining*, pp. 721–724 (2002)

[Yoshida 95] Yoshida, K. and Motoda, H.: CLIP: Concept Learning from Inference Pattern, *Artificial Intelligence*, Vol. 75, No. 1, pp. 63–92 (1995)

[猪口 01] 猪口 明博, 鷲尾 隆, 元田 浩：Apriori-Based Graph Mining アルゴリズムの効率化, 第 15 回 人工知能学会全国大会 (2001)

〔担当委員：栗原　聡〕

2003 年 2 月 7 日　受理

著 者 紹 介

西村　芳男(学生会員)

2001 年大阪大学工学部通信工学科卒業．現在，同大学院工学研究科通信工学専攻博士前期課程在学中．グラフ構造データからのデータマイニング・知識発見に関する研究に興味を持つ．

鷲尾　　隆(正会員)

1960 年生．1983 年東北大学工学部原子核工学科卒業．1988 年東北大学大学院原子核工学専攻博士課程修了．工学博士．1988 年から 1990 年にかけてマセチューセッツ工科大学原子炉研究所客員研究員．1990 年（株）三菱総合研究所入社．1996 年退社．現在，大阪大学産業科学研究所助教授（知能システム科学研究部門）原子力システムの異常診断手法に関する研究，定性推論に関する研究を経て，現在は人工知能の基礎研究，特に科学的知識発見，データマイニングなどの研究に従事．1988 年 2 月計測自動制御学会学術奨励賞受賞，1995 年 8 月人工知能学会全国大会優秀論文賞受賞，他 2 件．1996 年 3 月日本原子力学会論文賞受賞，1996 年 12 月人工知能学会研究奨励賞受賞，他 1 件．著書に “*Expert Systems Applications within the Nuclear Industry*”, American Nuclear Society,「知能工学概論」: 第 2 章エージェント（共著，廣田　薫　編，昭晃堂）など．AAAI，人工知能学会，計測自動制御学会，情報処理学会，日本ファジイ学会，各会員．

吉田　哲也(正会員)

1991 年東京大学工学部航空工学科卒業．1992 年から 1993 年にかけてエジンバラ大学大学院留学．1994 年から 1995 年にかけてカリフォルニア大学バークレー校交換留学．1997 年東京大学大学院博士課程修了．工学博士．同年，大阪大学大学院基礎工学研究科助手．現在，大阪大学産業科学研究所助手（知能システム科学研究部門）．主に機械学習，知識獲得，データマイニングなどの研究に興味を持つ．人工知能学会，情報処理学会など各会員．

元田　　浩(正会員)

1965 年東京大学工学部原子力工学科卒業．1967 年同大学院原子力工学専攻修士課程終了．同年，日立製作所に入社．同社中央研究所，原子力研究所，エネルギー研究所，基礎研究所を経て平成 7 年退社．現在，大阪大学産業科学研究所教授 (知能システム科学研究部門，高次推論研究分野)．原子力システムの設計，運用，制御に関する研究，診断型エキスパート・システムの研究を経て，現在は人工知能の基礎研究，とくに機械学習，知識獲得，知識発見，データマイニングなどの研究に従事．工学博士．日本ソフトウェア科学会理事，人工知能学会理事，同編集委員会委員，日本認知科学会編集委員会委員，Knowledge Acquisition (Academic Press) 編集委員，IEEE Expert 編集委員を歴任．Artificial Intelligence in Engineering (Elsevier Applied Science) 編集委員，International Journal of Human-Computer Studies (Academic Press) 編集委員，Knowledge and Information Systems: An International Journal (Springer-Verlag), Intelligent Data Analysis: An Internartional Journal (IOS Press) 編集委員．1970 年日本原子力学会奨励賞，1977，1984 年日本原子力学会論文賞，1990，1993, 2001 年人工知能学会論文賞受賞．1997 年人工知能学会研究奨励賞受賞，1998，1999 年人工知能学会全国大会優秀論文賞受賞，2000 年人工知能学会業績賞受賞．人工知能学会，情報処理学会，日本ソフトウェア科学会，日本認知科学会，AAAI, IEEE Computer Society，各会員．

猪口 明博（正会員）

1998 年大阪大学工学部通信工学科卒業．2000 年同大学大学院工学研究科通信工学専攻博士前期課程修了．同年年日本アイ・ビー・エム株式会社入社．現在，同社東京基礎研究所に勤務．データマイニング，知識発見，機械学習に関する研究に興味を持つ．2002 年 JCAC(Journal of Computer Aided Chemistry) 論文賞受賞．

岡田 孝（正会員）

1971 年東京大学工学部合成化学科卒業．1976 年大阪大学大学院化学工学専攻博士課程修了．工学博士．同年より関西学院大学情報処理研究センターにて専任講師、助教授を経て教授．2003 年より同大学理工学部情報科学科教授．電子状態理論の研究，化学構造の類似性に関する研究を経て，現在はデータマイニングシステムの開発および化学構造と生理活性間の相関関係を中心とする分野での知識発見研究に従事．人工知能学会，情報処理学会，化学会，コンピュータ化学会，OR 学会，AAAI, ACM, IEEE Computer Society, ACS, 各会員．