Vol.2019-ICS-195 No.5
2019/3/18

属性情報の図示に基づく概念束分解手法の比較

石榑 隼人[1,a)]　武藤 敦子[1]　森山 甲一[1]　犬塚 信博[1]

概要：形式概念分析は概念を数学的に扱い，データ分析を行う手法である．形式概念分析では，データから概念束と呼ばれる構造を計算し，これをデータ中の属性情報と共に図示することで，データの理解を助ける．しかし，データのサイズが増加すると，概念束は急速に複雑になる．これに対処するために，概念束の分解を行う様々な手法が提案されている．本稿では，様々な分解手法の利用法を検討するために，分解後の概念束を図示した場合に，元の概念束の情報をどの程度読み取れるかを比較した．

Comparison of Decomposition Methods for Concept Lattices Based on Visualization with Attributes

1. はじめに

近年，データ量の増加や計算機の高性能化を背景として，大量のデータから有用な情報を抽出するデータマイニングが注目されている．データマイニングに関する手法の一つに形式概念分析 [1] がある．これは束論の応用として Wille が提案した手法であり，概念やその構造を数理的に扱うことでデータ分析を行う．形式概念分析で扱われる概念は形式概念と呼ばれる．また，形式概念がなす構造は概念束と呼ばれる．形式概念分析では概念束をハッセ図などで図示し，概念束の構造を観察することで，データの構造や属性間の関係などを理解することができる．形式概念分析は，画像データベースの構造の可視化 [2] やソースコードの分析 [3] などで利用されている．

一方でデータの増大に従い形式概念の数が急速に大きくなり，概念束の構造が複雑になることが，形式概念分析の問題として知られている．概念束が大きく複雑になると，図示によるデータの理解は困難となる．このため，複雑な概念束に対処するために，概念束の簡素化や概念束の分解などの方法がある．概念束の簡素化は，大きく複雑な概念束を単純化し，小さくする手法である．一方で概念束の分解は，大きく複雑な概念束を複数に分けることで扱いやすくする手法である．概念束の分解はソフトウェア工学の分野で特に利用されている [4], [5]．概念束の分解手法として，様々な手法が提案されている．その多くは代数的な性質を基に定義されているために，分解が行えるための条件がある手法が多いが，分解後の概念束は分解前の概念束の性質をよく表す．本稿では，概念束を図示した場合に，分解後の概念束から分解前の概念束の情報をどの程度読み取れるかについて，分解手法を比較した．

2. 関連研究

[6] は，概念束の分解を含む多くの手法について分類を行った．[7] は，complete congruence relation を利用する分解手法について，分解前の概念束と分解後の概念束の性質を比較した．これらの研究では，分解手法間の比較は行われていない．また，[8] は，三種類の分解手法のアルゴリズムを紹介し，適用例を示した．この研究では，分解手法間の違いについては，計算量以外は明確に述べられていない．本稿では，概念束の図示を行った場合について，様々な分解手法の性質を比較し，どの分解手法が適切であるかを考える．

3. 形式概念分析

3.1 概念束

形式概念分析では，形式文脈と呼ばれるデータから形式概念と概念束を得ることを基礎として，データ分析を行う．以降では，集合は全て有限集合だと仮定する．形式文脈 (G, M, I) は，対象集合 G，属性集合 M，G と M の間

1　名古屋工業大学
Nagoya Institute of Technology, Gokiso-cho, Showa-ku, Nagoya, Aichi, 466–8555, Japan
a)　h.ishigure.488@nitech.jp

表 1　形式文脈
Table 1　Formal Context

	卵生（m_1）	母乳　（m_2）	言葉（m_3）
ハト（g_1）	×		
カモノハシ（g_2）	×	×	
ネコ（g_3）		×	
ヒト（g_4）		×	×

の二項関係 $I \subseteq G \times M$ から構成される．対象 $g \in G$ と属性 $m \in M$ に対して，$(g, m) \in I$ である時，g は m を持つという．これを gIm と書く．**表 1** は形式文脈の例である．表中の × は対象が属性を持つことを表す．

ここで，対象の集合 $X \subseteq G$ と属性の集合 $Y \subseteq M$ に対して，式 (1), (2) の写像を定義する．

$$X \mapsto X^I = \{m \in M \mid \forall g \in X[gIm]\} \tag{1}$$
$$Y \mapsto Y^I = \{g \in G \mid \forall m \in Y[gIm]\} \tag{2}$$

集合のサイズが 1 である場合，すなわち $X = g$ のような場合には，X^I を g^I の様に書くことがある．このとき，形式概念が以下の様に定義される．

定義 1 (形式概念)**.** 形式文脈 (G, M, I) 及び対象の集合 $A \subseteq G$ と属性の集合 $B \subseteq M$ に対して，組 (A, B) が $A^I = B, B^I = A$ を共に満たす時，これを (G, M, I) の形式概念という．この時，A, B をそれぞれ外延，内包と呼ぶ．また，(G, M, I) の形式概念全ての集合を $\mathfrak{B}(G, M, I)$ と書く．

形式概念がなす構造を概念束と呼ぶ．概念束は以下の様に定義される．

定義 2 (概念束)**.** 形式文脈 (G, M, I) の二つの形式概念 $(A_i, B_i), (A_j, B_j)$ に対して，以下のように半順序が定義される．

$$(A_i, B_i) \geq (A_j, B_j) \iff A_i \supseteq A_j \iff B_i \subseteq B_j$$

この順序により $(\mathfrak{B}(G, M, I), \leq)$ は束をなす．これを概念束と呼ぶ．

以降は，$\mathfrak{B}(G, M, I)$ と $(\mathfrak{B}(G, M, I), \leq)$ を区別しない．また，概念束において，上限，下限は以下の様になる．

定理 1. 概念束 $(\mathfrak{B}(G, M, I), \leq)$ において，下限，上限は次式の様に定まる．

$$\bigwedge_{t \in T}(A_t, B_t) = \left(\bigcap_{t \in T} A_t, \left(\bigcup_{t \in T} B_t\right)^{II}\right)$$
$$\bigvee_{t \in T}(A_t, B_t) = \left(\left(\bigcup_{t \in T} B_t\right)^{II}, \bigcap_{t \in T} B_t\right)$$

図 1 は表 1 で表される形式文脈の概念束を，ハッセ図で図示したものである．一つのノードは一つの形式概念を表す．属性 m のラベルは形式概念 $\left(m^I, m^{II}\right)$ に付与される．以降ではこの形式概念を $\mu_I m$ と書く．この形式概念は m を含む形式概念の中で，最上位の形式概念である．対象の数は属性の数に比べて膨大である場合が多いため，対象のラベルは省略することが多い．このとき，形式概念の内包は，それ自身かそれより上位の形式概念に与えられたラベルで示される．例えば，図 1 において，m_3 のラベルが付いた形式概念の内包は，$\{m_2, m_3\}$ である．また，最大元の内包は $\emptyset$ である．

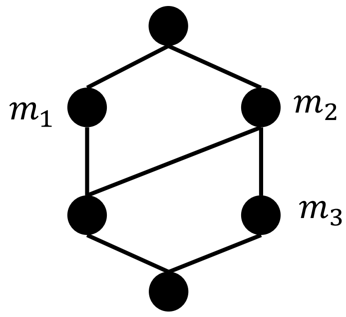

図 1　表 1 の概念束
Fig. 1　Concept Lattice of Table 1

3.2　属性間の含意関係

形式概念分析では属性間の含意関係から，データに対しての考察を得ることが多い．属性間の含意関係は，次のように定義される．

定義 3 (属性間の含意関係)**.** (G, M, I) を形式文脈とする．属性集合 $P, Q \subseteq M$ が，任意の対象 $g \in G$ に対して $P \not\subseteq g^I$ または $Q \subseteq g^I$ を満たすとき，$A \rightarrow B$ を (G, M, I) の含意関係と呼ぶ．

定義より，$A \rightarrow B$ は，任意の対象が A の全ての属性を持つならば，B の全ての属性も持つことを表す．また，$P \rightarrow Q$ が成立することと，全ての形式概念の内包が P を含むならば Q も含むことは同値である．表 1 の形式文脈では，$m_1, m_3 \rightarrow m_2$ という含意関係が成立する．

また，含意関係は図示された概念束から読み取ることもできる．概念束 $L = \mathfrak{B}(G, M, I)$ において成立する含意関係の集合を $\mathfrak{L}(L)$ とする．このとき，属性集合 $P, Q \subseteq M$ に対して，式 (3) が成立する．

$$P \rightarrow Q \in \mathfrak{L}(L) \Leftrightarrow \bigwedge_{m \in P} \mu_I m \leq \bigwedge_{m \in Q} \mu_I m \tag{3}$$

すなわち P に含まれる属性のラベルがある形式概念全ての下限が，Q に含まれる属性のラベルがある形式概念全ての下限より下にある時，$P \rightarrow Q$ が成立する．図 1 において，m_1 のラベルが付いた形式概念と m_3 のラベルが付いた形式概念の上限は最小元であり，これは m_2 のラベルが付いた形式概念より下位であるため，含意関係 $m_1, m_3 \rightarrow m_2$ が成立することが概念束から読み取れる．

4.　概念束の分解

形式文脈のサイズが大きくなると，概念束は急速に大き

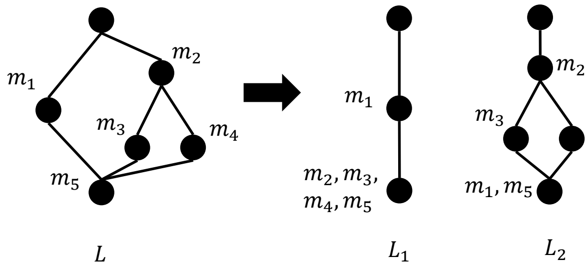

図 2　horizontal decomposition
Fig. 2　Horizontal Decomposition

く複雑になる．これに対処するため，概念束を分解する様々な手法がある．ここでは，horizontal decomposition [9]，atlas decomposition [10]，属性集合の分割，subdirect decomposition [10] について紹介する．これらの手法は概念束の分解を求めるのが比較的容易であるため，利用しやすい．ただし，属性集合の分割以外には，分解を行えるための条件がある．

4.1　horizontal decomposition

horizontal decomposition は，概念束を横に並べるように分解する手法である．具体的には，概念束の最大元と最小元を取り除いて分割した後，それぞれに新たな最大元と最小元を加えることで，概念束の分解を行う．本稿では，新たに加える最大元と最小元は，分解前の概念束の最大元と最小元とする．

定義 4 (horizontal decomposition)**.** 概念束 L に対して，$\top, \bot$ を L の最大元，最小元とする．$L \setminus \{\top, \bot\}$ のハッセ図をグラフとみなした場合の連結成分を $X_1, X_2, \ldots, X_n$，$L_i = X_i \cup \{\top, \bot\}$ としたとき，$L_1, L_2, \ldots, L_n$ を L の horizontal decomposition と呼ぶ．

図 2 は horizontal decomposition の例である．これのみでは分解できない場合に，事前に概念束に対して処理を行う場合があるが，本稿ではこのような処理を行わない場合についてのみ評価を行う．

概念束 L の horizontal decomposition $L_1, L_2, ..., L_n$ は，定義より式 (4), (5) の性質を満たす．

$$\forall i \in \{1, 2, \ldots, n\}[L_i \subseteq L] \tag{4}$$

$$\bigcup_{i=1}^{n} L_i = L \tag{5}$$

また，分解後のそれぞれの束 L_i は元の概念束 L の完備部分束となる．このとき，$J = \bigcup_{(A,B)\in L_i} A \times B$ と書くと，L_i は形式文脈 (G, M, J) の概念束でもある．したがって，分解後の束は概念束としての性質を持つため，図示により内包や含意関係が読み取れる．

また，属性 $m \in M$ が $\mu_I m \in L_i$ である場合，明らかに $\mu_J m = \mu_I m$ である．$\mu_I m \notin L_i$ である場合は，定義より m を含む L_i の形式概念は $\bot$ のみである．したがって，この場合は $\mu_J m = \bot$ である．

表 2　表 1 の block relation
Table 2　Block Relation of Table 1

	m_1	m_2	m_3
g_1	×	×	×
g_2	×	×	×
g_3		×	×
g_4		×	×

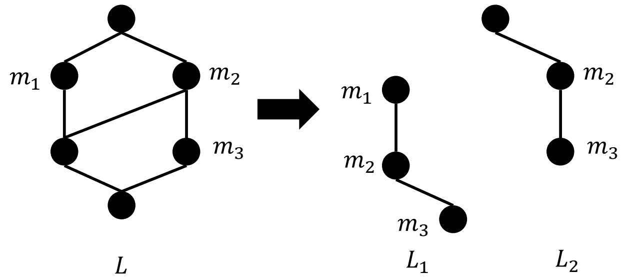

図 3　atlas decomposition
Fig. 3　atlas decomposition

4.2　atlas decomposition

atlas decomposition は，概念束の全体像と詳細を分けることで，概念束の構造をわかりやすくするための分解手法である．[10] では束上の二項関係を用いて定義しているが，本稿ではそれと同等な block relation を用いた定義を示す．

定義 5 (block relation)**.** 形式文脈 (G, M, I) の block relation $J \subseteq G \times M$ は以下の性質を満たす二項関係 J である．

- $I \subseteq J$
- 任意の対象 $g \in G$ に対して g^J は (G, M, I) の内包
- 任意の属性 $m \in M$ に対して m^J は (G, M, I) の外延

表 2 は表 1 で表される形式文脈の block relation の例である．block relation J の形式概念 (A, B) に対して，$\mathfrak{B}(A, B, I \cap A \times B)$ をブロックと呼ぶ．ブロックは，必ず $\mathfrak{B}(G, M, I)$ において，二つの形式概念 $(A, A^I), (B^I, B)$ に挟まれた形式概念の集合となる．

定義 6 (atlas decomposition)**.** 概念束 $\mathfrak{B}(G, M, I)$ に対して，(G, M, I) のある block relation J のブロック全ての集合を atlas decomposition と呼ぶ．また，$\mathfrak{B}(G, M, J)$ を factor lattice と呼ぶ．

factor lattice は元の概念束の大まかな全体像を表し，ブロックは概念束の一部に関しての詳細を表す．**図 3** は，表 2 の block relation による atlas decomposition を示す．

block relation の性質から，概念束 L の atlas decomposition $L_1, L_2, \ldots, L_n$ は式 (6), (7) を満たす．

$$\forall i \in \{1, 2, \ldots, n\}[L_i \subseteq L] \tag{6}$$

$$\bigcup_{i=1}^{n} L_i = L \tag{7}$$

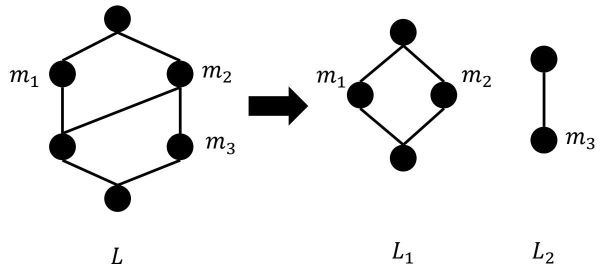

図 4　属性集合の分割
Fig. 4　Partition of the Set of Attributes

4.3　属性集合の分割

概念束の図示の手法の一つである nested line diagram では，属性集合を複数に分割した後，それぞれ得られる形式文脈から概念束を求め，これらを組み合わせることで，概念束を図示する．本稿では，分割した形式文脈から得られる概念束を，概念束の分解として扱う．

属性集合の分割は次の性質を満たす．

定理 2. (G, M, I) を形式文脈，$M = \bigcup_{i=1}^{n} M_i$ とする．このとき，任意の $i = 1, 2, \ldots, n$ に対して，$J = I \cap G \times M_i$ とおくと，次の写像は全射である．

$$\begin{aligned} \mathfrak{B}(G, M, I) &\to \mathfrak{B}(G, M_i, J) \\ (A, B) &\mapsto \left((B \cap M_i)^J, B \cap M_i\right) \end{aligned}$$

分解前の概念束の形式概念一つは，分解後の概念束それぞれに対して対応する形式概念を持つ．すなわち，分解前の形式概念は分解後の形式概念の組み合わせで表現できる．

図 4 は属性集合の分割の例である．ここでは，属性集合を $\{m_1, m_2\}, \{m_3\}$ の二つに分割している．

4.4　subdirect decomposition

定義 7 (complete congruence relation)**.** 完備束 V に対して，以下の性質を満たす同値関係 Θ を complete congruence relation と呼ぶ．

$$x_t \Theta y_t \text{ for } t \in T \Rightarrow \left(\bigwedge_{t \in T} x_t\right) \Theta \left(\bigwedge_{t \in T} y_t\right)$$
$$x_t \Theta y_t \text{ for } t \in T \Rightarrow \left(\bigvee_{t \in T} x_t\right) \Theta \left(\bigvee_{t \in T} y_t\right)$$

また，$x \in V$ が属する Θ の同値類を $[x]\Theta$ と書くとき，以下のように定義される順序により同値類全ての集合がなす束を factor lattice と呼ぶ．

$$[x]\Theta \leq [y]\Theta \Leftrightarrow x\Theta(x \wedge y)$$

subdirect decomposition は，complete congruence relation の集合として定義される．

定義 8 (subdirect decomposition)**.** 束 V 上の complete congruence relation の集合 $\Theta_t, t \in T$ が，以下の条件を満たすとき，これらを V の subdirect decomposition と呼ぶ．

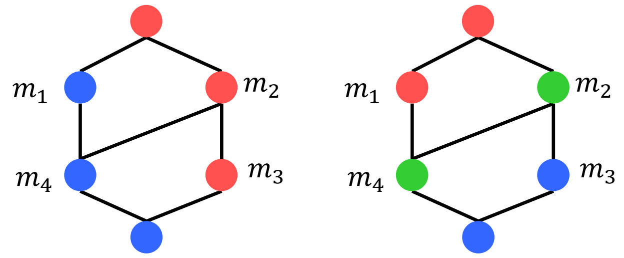

図 5　complete congruence relation
Fig. 5　Complete Congruence Relation

$$\bigcap_{t \in T} \Theta_t = \{(x, x) \mid x \in V\}$$

定義から，各 complete congruence relation から同値類を一つずつ選んだとき，選んだ全ての同値類全てに含まれる V の元は高々一つである．すなわち，V の任意の元は factor lattice の元の集合で表すことができる．

定義 9 (compatible subcontext)**.** (G, M, I) を形式文脈，$(H, N, I \cap H \times N)$ をその subcontext とする．(G, M, I) の任意の形式概念 (A, B) に対して，$(A \cap H, B \cap N)$ が $(H, N, I \cap H \times N)$ の形式概念であるとき，$(H, N, I \cap H \times N)$ を compatible と呼ぶ．

compatible subcontext $(H, N, I \cap H \times N)$ に対して，式 (8) で定義される写像 $\Pi_{H,N} : \mathfrak{B}(G, M, I) \to \mathfrak{B}(H, N, I \cap H \times N)$ は全射である．

$$\Pi(A, B) := (A \cap H, B \cap N) \qquad (8)$$

概念束 $\mathfrak{B}(G, M, I)$ 上の complete congruence relation Θ に対して，式 (9) を満たす compatible subcontext $(H, N, I \cap H \times N)$ が存在する．

$$(A_1, B_1)\Theta(A_2, B_2) \Leftrightarrow B_1 \cap N = B_2 \cap N \qquad (9)$$

このとき，$\mathfrak{B}(H, N, I \cap H \times N)$ は Θ の factor lattice と同型である．したがって，compatible subcontext の概念束の集合を subdirect decomposition とみなすことができる．本稿では，subdirect decomposition $\Theta_t, t \in T$ に対して，各 Θ_t に対応する compatible subcontext の概念束を考え，これらについて評価する．

図 5 は概念束上の二つの complete congruence relation を表す．図中で色が同じ形式概念同士は同値である．また，図 6 は，図 5 による subdirect decomposition を示す．subdirect decomposition では，元の概念束に含まれる属性が，分解後の概念束には現れない場合がある．この例では，属性 m_4 は，分解後の二つの概念束のどちらにも入ることがない．

5.　分解手法の評価

5.1　問題設定

本稿では，分解後の概念束を図示した場合に，分解前の

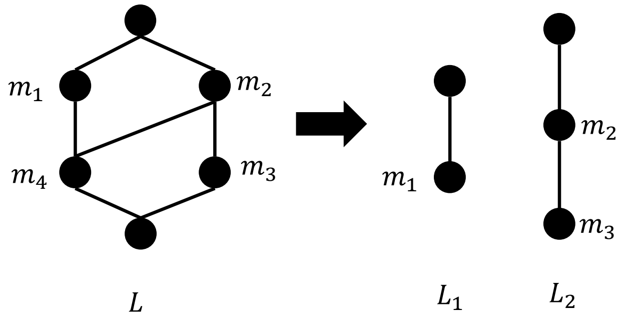

図 **6** subdirect decomposition
Fig. 6 Subdirect Decomposition

概念束の内包，含意関係をどの程度読み取れるかに関して，分解手法の評価を行う．概念束を図示した場合，そこから内包と含意関係を読み取ることができるため，分解後の概念束の内包と含意関係は得られるとして考える．以下では，分解前の概念束を L ，分解後の概念束を $L_1, L_2, ..., L_n$ ，L_i に含まれる属性の集合を M_i と書く．また，概念束 L に含まれる内包の集合を $\mathrm{intents}(L)$ と書く．

分解手法の評価の際には，まず分解後の概念束一つのみから読み取れる分解前の概念束の情報について考える．その後，分解後の概念束全てから読み取れる情報についても，簡単に考える．

5.2 horizontal decomposition

まず，分解後のある概念束 L_i について考える．内包については，式 (4) より以下が成立する．

$$B \in \mathrm{intents}(L_i) \Rightarrow B \in \mathrm{intents}(L) \tag{10}$$

逆は明らかに成立しない．

また，概念束 L の horizontal decomposition で成立する含意関係は，分解後の概念束 L_i でも成立する．逆に言えば， L_i で成立しない含意関係は L でも成立しない．

$$P \to Q \in \mathfrak{L}(L) \Rightarrow P \to Q \in \mathfrak{L}(L_i) \tag{11}$$

Proof. $P \to Q \in \mathfrak{L}(L)$ と仮定すると，任意の形式概念 $(A, B) \in L$ について， $B \subseteq P \Rightarrow B \subseteq Q$ である．また，式 (4) より，任意の i について $L_i \subseteq L$．したがって， L_i の任意の形式概念 (A, B) について， $B \subseteq P \Rightarrow B \subseteq Q$ である．また， $M_i = M$ より， $P \cup Q \in M_i$ である．以上より， $P \to Q \in \mathfrak{L}(L_i)$ が成立する． □

逆は常には成立せず，以下の様な条件が必要である．

$$P \to Q \in \mathfrak{L}(L_i) \,\mathrm{and}\, \forall m \in P[\mu_J m \neq \bot] \Rightarrow P \to Q \in \mathfrak{L}(L) \tag{12}$$

ここで， (G, M, J) は L_i に対応する形式文脈とする．したがって， L_i で成立する含意関係の一部は L でも成立する．

Proof. 左辺が成立すると仮定する．このとき，任意の $m \in P$ について， $\mu_I m = \mu_J m$ であり，かつ L_i は L の完備部分束であるため，$\bigwedge_{m\in P} \mu_I m = \bigwedge_{m\in P} \mu_J m$ となる．また，任意の $m \in Q$ について， $\mu_J m$ は $\mu_I m$ または $\bot$ と一致するため，$\bigwedge_{m\in P} \mu_J m$ は $\bigwedge_{m\in P} \mu_I m$ または $\bot$ と一致する．すなわち， $\bigwedge_{m\in Q} \mu_J m \leq \bigwedge_{m\in Q} \mu_I m$ である．したがって， 以下が成立する．

$$\begin{aligned}\bigwedge_{m\in P} \mu_I m &= \bigwedge_{m\in P} \mu_J m \\ &\leq \bigwedge_{m\in Q} \mu_J m \\ &\leq \bigwedge_{m\in Q} \mu_I m\end{aligned}$$

これは $P \to Q \in \mathfrak{L}(L)$ と同値である． □

分解後の概念束全てが与えられている場合，分解後の各概念束の最大元と最小元を重ね合わせることで，分解前の概念束の構造が得られる．また，属性のラベルについても復元可能である．したがって，分解前の概念束の内包と含意関係は全て正確に得られる．

5.3 atlas decomposition

atlas decomposition では，horizontal decomposition と同様に， $L_i \subseteq L$ である．したがって，分解後の概念束一つについて，式 (13) が成立する．

$$B \in \mathrm{intents}(L_i) \Rightarrow B \in \mathrm{intents}(L) \tag{13}$$

horizontal decomposition と同様に，逆は成立しない．

各 L_i には含まれない属性がありうるため，含意関係に関しては，式 (14) の性質が成立する．

$$P \to Q \in \mathfrak{L}(L) \,\mathrm{and}\, P \cup Q \subseteq M_i \Rightarrow P \to Q \in \mathfrak{L}(L_i) \tag{14}$$

したがって atlas decomposition でも， L で成立しない含意関係は L_i から読み取ることができる．

一方で， L で成立する含意関係は，式 (15) から分かる．

$$\exists S \subseteq L_i \left[S \neq \emptyset \,\mathrm{and}\, \textstyle\bigcup_{(A,B)\in S} B = P\right] \,\mathrm{and}\, P \to Q \in \mathfrak{L}(L_i) \Rightarrow P \to Q \in \mathfrak{L}(L) \tag{15}$$

Proof. 左辺を仮定する．また，(H, M_i, J) を L_i に対応する形式文脈とする．式 (3) より，$\bigwedge_{m\in P} \mu_J m \leq \bigwedge_{m\in Q} \mu_J m$ である．ここで， $L_i \subseteq L$ であり， $\mu_I m, \mu_J m$ はそれぞれ L, L_i で m を含む最上位の形式概念であるため，$\mu_J m \leq \mu_I m$ となる．また，L において，$(A, B) = \bigwedge_{m\in B}$ であり，また $\bigcup_{(A,B)\in S} B = P$ であるため，$\bigwedge S = \bigwedge_{m\in P} \mu_I m$ である．同様に，L_i においても，$\bigwedge S = \bigwedge_{m\in P} \mu_J m$ となる．ここで，L_i は L の部分束でかつ $S \neq \emptyset$ なので，L, L_i において $\bigwedge S$ は一致する．したがって， $\bigwedge_{m\in P} \mu_I m = \bigwedge_{m\in P} \mu_J m$

となる．ここから，以下が成立する．

$$\begin{aligned}\bigwedge_{m\in P}\mu_I m &= \bigwedge_{m\in P}\mu_J m\\ &\le \bigwedge_{m\in Q}\mu_J m\\ &\le \bigwedge_{m\in Q}\mu_I m\end{aligned}$$

これは右辺と同値である． □

また，分解後の概念束全てを用いた場合，内包を比較して適切に重ね合わせることで，分解前の概念束を復元できる．したがって，atlas decomposition では horizontal decomposition と同様に，分解前の概念束の内包と含意関係が全て導出できる．

5.4 属性集合の分割

定理 2 より， L_i の内包について，式 (16) が成立する．

$$B \in \mathrm{intents}(L_i) \Leftrightarrow \exists D \in \mathrm{intents}(L)[D \cap M_i = B] \quad (16)$$

すなわち， L_i からは， L の内包について部分的に知識が得られる．一方で， L_i には含まれない属性があるため，L_i のみからは L の内包全体は得られない．

また，含意関係について，式 (17) が成立する．

$$P \to Q \in \mathfrak{L}(L) \text{ and } P \cup Q \subseteq M_i \Leftrightarrow P \to Q \in \mathfrak{L}(L_i) \quad (17)$$

Proof. まず， 左辺 ⇒ 右辺 について考える．$J = I \cap G \times M_i$ とおくと，L_i に対応する形式文脈は (G, M_i, J) と書ける．左辺を仮定すると， L の任意の内包 B について， $P \not\subseteq B$ または $Q \subseteq B$ である．また， $P \cup Q \subseteq M_i$ である．したがって，式 (16) より， L_i の任意の内包についても，同じことが成立する．すなわち， $P \to Q \in \mathfrak{L}(L)$ が成立する．

次に， 右辺 ⇒ 左辺 について考える．左辺を仮定すると， $P \to Q \in \mathfrak{L}(L_i)$ より， $P \cup Q \subseteq M_i$ である．また L_i の任意の内包 B について， $P \not\subseteq B$ または $Q \subseteq B$ である．このとき，式 (16) より， L の任意の内包についても，同じことが成立する．したがって， $P \to Q \in \mathfrak{L}(L)$ が成立する． □

次に分割後の概念束を全て用いた場合を考える．定理 2 より，各 L_i から内包を一つずつ取り出して和集合をとることで，L の内包の候補が得られる．しかし，このようにして作られる内包の候補は，必ずしも L の内包ではない．

含意関係についても， M_i ごとに含まれる属性の含意関係は各 L_i から得られるが，複数の M_i にまたがる属性の含意関係は得られない．

5.5 subdirect decomposition

式 (8) の写像は全射であり，この写像は属性集合については，定理 2 と同じである．したがって，属性集合の分割と同様に，式 (18), (19) の性質が成立する．

$$B \in \mathrm{intents}(L_i) \Leftrightarrow \exists D \in \mathrm{intents}(L)[D \cap M_i = B] \quad (18)$$

$$P \to Q \in \mathfrak{L}(L) \text{ and } P \cup Q \subseteq M_i \Leftrightarrow P \to Q \in \mathfrak{L}(L_i) \quad (19)$$

分解後の概念束全てを用いた場合でも，属性集合の分割と同様に， L の内包と含意関係について，全ての情報は読み取れない．また，subdirect decomposition では，$\bigcup_i \in \{1, 2, ..., n\} M_i = M$ が成立しない場合があるため，この場合には， $\left(\bigcup_{i\in\{1,2,\ldots,n\}} M_i\right) \setminus M$ に含まれる属性については全く知識が得られない．

5.6 考察

horizontal decomposition と atlas decomposition において，分解後の概念束一つからでも一部の含意関係は分解前の概念束でも成立することが読み取れる．ここで，式 (12), (15) の条件を比較する．式 (15) において， S が L_i の最小元を含むならば $P = M_i$ である．このとき，$Q \subseteq P$ なので， $P \to Q$ はどのような概念束でも成立する．したがって， S が L_i の最小元を含まない場合のみを考える．このとき，horizontal decomposition において，$P \to Q$ が式 (15) の条件を満たすならば，式 (12) の条件も満たす．一方で， atlas decomposition で式 (12) の条件を満たす場合でも，$P \to Q$ が分解前の概念束で成立するとは限らない．また， 式 (12) の条件では， P は任意の場合に判定可能だが，式 (15) の条件では， P が複数の内包の和集合である場合しか判定できない．

したがって含意関係を重要視した場合には，atlas decomposition より horizontal decomposition が適していると考えられる．しかし，概念束がそれぞれの手法で分解可能か，どのように分解されるかは，形式文脈がどのようであるかに依存する．また，どちらの手法も分解後の概念束を全て利用すれば，元の概念束の内包と含意関係が全て得られるが，分解後の概念束全てのサイズの合計は元の概念束とあまり変わらないため，この場合は分解の恩恵を得られない．

一方で，属性集合の分割と subdirect decomposition では，分解後の概念束一つからでも全てからでも，分解前の概念束について得られる情報はあまり変わらない．さらに属性集合の分割は，subdirect decomposition と異なり，分解可能であるための制約がない．そのため，今回の目的では subdirect decomposition より属性集合の分割の方が適していると考えられる．また，この二つの手法は分解後の形式概念の組み合わせで，分解前の形式概念を表すため，分解後の概念束のサイズが小さくなりやすい．したがって，分解後の概念束全てを組み合わせて利用するのが妥当

だと考えられる．しかし，属性集合の分割によって得られる情報は，分割方法に大きく依存するため，どのように適切な分割を行うかを考える必要がある．

6. おわりに

本稿では概念束の様々な分解手法について，分解後の概念束を図示した場合に，分解前の概念束の情報が得られるかという観点で評価した．その結果，atlas decomposition より horizontal decomposition，subdirect decomposition より属性集合の分割がこの目的に適していることがわかった．

今後の課題として，実データを用いた分解可能性や分解後に読み取れる情報の評価がある．また，属性集合の分割に対して，適切な分割を行うための手法を考える必要がある．

参考文献

[1] Wille, R.: Restructuring Lattice Theory: An Approach Based on Hierarchies of Concepts, *Ordered Sets*, Springer, pp. 445–470 (1982).

[2] 澤勢一史，延原　肇：大規模画像群のための形式概念分析に基づく束構造可視化システム，知能と情報，Vol. 21, No. 1, pp. 32–40 (2009).

[3] Kazato, H., Hayashi, S., Oshima, T., Miyata, S., Hoshino, T. and Saeki, M.: Extracting and Visualizing Implementation Structure of Features, *Software Engineering Conference (APSEC), 2013 20th Asia-Pacific*, Vol. 1, IEEE, pp. 476–484 (2013).

[4] Lindig, C. and Snelting, G.: Assessing Modular Structure of Legacy Code Based on Mathematical Concept Analysis, *Proceedings of the 19th International Conference on Software Engineering*, pp. 349–359 (1997).

[5] Bhatti, M. U., Anquetil, N., Huchard, M. and Ducasse, S.: A Catalog of Patterns for Concept Lattice Interpretation in Software Reengineering, *Proceedings of the 24th International Conference on Software Engineering & Knowledge Engineering*, pp. 118–123 (2012).

[6] Priss, U. and Old, L. J.: Data Weeding Techniques Applied to Roget's Thesaurus, *Knowledge Processing and Data Analysis*, Springer, pp. 150–163 (2011).

[7] Viaud, J.-F., Bertet, K., Missaoui, R. and Demko, C.: Using congruence relations to extract knowledge from concept lattices, *Discrete Applied Mathematics*, Vol. 249, pp. 135–150 (2018).

[8] Funk, P., Lewien, A. and Snelting, G.: Algorithtms for Concept Lattice Decomposition and their Application, Technical report, Technische Universität Braunschweig (1995).

[9] Snelting, G.: Reengineering of Configurations Based on Mathematical Concept Analysis, *ACM Transactions on Software Engineering and Methodology*, Vol. 5, No. 2, pp. 146–189 (1996).

[10] Ganter, B. and Wille, R.: *Formal Concept Analysis: Mathematical Foundations*, Springer (1999).