Steps Towards Achieving Distributivity in Formal Concept Analysis

Alain Gély[1], Miguel Couceiro[2], and Amedeo Napoli[2]

[1] Université de Lorraine, CNRS, LORIA, F-57000 Metz, France
[2] Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France
{alain.gely,miguel.couceiro,amedeo.napoli}@loria.fr

Abstract. In this paper we study distributive lattices in the framework of Formal Concept Analysis (FCA). The main motivation comes from phylogeny where biological derivations and parsimonious trees can be represented as median graphs. There exists a close connection between distributive lattices and median graphs. Moreover, FCA provides efficient algorithms to build concept lattices. However, a concept lattice is not necessarily distributive and thus it is not necessarily a median graph.
In this paper we investigate possible ways of transforming a concept lattice into a distributive one, by making use Birkhoff's representation of distributive lattices. We detail the operation that transforms a reduced context into a context of a distributive lattice. This allows us to reuse the FCA algorithmic machinery to build and to visualize distributive concept lattices, and then to study the associated median graphs.

1 Context and Motivations

Formal Concept Analysis (FCA) has proved to be an effective tool in data analysis and knowledge discovery in several application domains [10,14]. Concept lattices provide a valuable support for several tasks, such as classification, information retrieval and pattern recognition. Besides lattices, trees and their extensions [4,5,13] are used in biology, notably, in phylogeny, for modeling inter-species filiations. In this domain, one of the main problems is to find evolution trees for representing existing species from accessible DNA fragments. When several trees are leading to the same inter-species filiations, the preferred ones are the most "parsimonious", i.e. the number of modifications such as mutations for example, is minimal for the considered species. However, several possible parsimonious trees may exist. Such a situation arises with inverse or parallel mutations, e.g., when a gene goes back to a previous state or the same mutation appears for two non-linked species. This asks for a generic representation of such a family of trees.

Bandelt [2,3] proposes the notion of *median graph* to overcome this issue, since he noticed that a median graph is capable of encoding all parsimonious trees. A median graph is a connected graph such that for any three vertices a, b, c, there is exactly one vertex x which lies on a shortest path between each pair of vertices in $\{a, b, c\}$. Alternatively, median graphs can be thought of as a

Paper published in Dmitry I. Ignatov, Lhouari Nourine (Eds.): CLA 2018, pp. 105–116, Department of Computer Science, Palacký University Olomouc, 2018.

generalization distributive lattices [8,15]. However, the extraction of such structures directly from data remained unaddressed.

Uta Priss [19,20] made a first attempt to use algorithmic machinery of FCA and the links between distributive lattices and median graphs, to analyze phylogenetic trees. However, not every concept lattice is distributive, and thus FCA alone does not necessarily outputs median graphs. In [20] Uta Priss sketches an algorithm to convert any lattice into a median graph. The key step is to transform any lattice into a distributive lattice.

In this article, we propose an algorithm supporting such a transformation that minimizes the changes introduced to the original lattice. Using the context of an initial concept lattice as input, the algorithm outputs the context of a distributive lattice, without necessarily building the lattice. Our approach relies on Birkhoff's representation of distributive lattices [6,7]. Moreover, we illustrate our approach with a generic example that reveals the difficulties of transforming of a concept lattice into a distributive lattice. We do not settle this issue entirely, but we propose major steps and an approach towards its solution.

The paper is organized as follows. In Sections 2 and 3 we recall the basic background and notation as well as some key results on distributive lattices. The transformation algorithm is presented in Section 4 and we discuss the strengths and limitations of our approach in Section 5.

2 Definitions and Notations

In this section we recall basic notions and notation needed throughout the paper. We will mainly adopt the formalism of [14], and we refer the reader to [11,12] for further background.

2.1 Partially Ordered Sets, Lattices and Homomorphisms

A *partially ordered set* (or *poset* for short) is a pair $(P, \leqslant)$ where P is a set and $\leqslant$ is a *partial order* on P, that is, a reflexive, antisymmetric and transitive binary relation on P. A poset $(P', \leqslant')$ is a *subposet* of $(P, \leqslant)$ if $P' \subseteq P$ and $\leqslant' \subseteq \leqslant$. For a subset $X \subseteq P$, let $\downarrow X = \{y \in P : y \leqslant x \text{ for some } x \in X\}$ and $\uparrow X = \{y \in P : x \leqslant y \text{ for some } x \in X\}$. If $X := \{x\}$, we use the notation $\downarrow x$ and $\uparrow x$ instead of $\downarrow \{x\}$ and $\uparrow \{x\}$, respectively. In this paper, we will only consider finite posets $(P, \leqslant)$ and, when there is no danger of ambiguity, we will refer to them by their underlying universes P.

A set $X \subseteq P$ is a (*poset*) *ideal* (resp. *filter*) if $X = \downarrow X$ (resp. $X = \uparrow X$). If $X = \downarrow x$ (resp. $X = \uparrow x$) for some $x \in P$, then X is said to be a *principal* ideal (resp. filter) of P. For $x, y \in P$, the greatest element of $\downarrow x \cap \downarrow y$ (resp. least element $\uparrow x \cap \uparrow y$) if it exists, is called the *infimum* (resp. *supremum*) of x and y. A *lattice* is a poset $(L, \leqslant)$ such that the infimum and the supremum of any pair $x, y \in L$ exist, and they are denoted respectively by $x \wedge y$ and $x \vee y$. A subset $X \subseteq L$ is a *sublattice* of L if for every $x, y \in X$ we have that $x \wedge y, x \vee y \in X$. As for posets, we will only consider finite lattices $(L, \leqslant)$ and we will refer to them by their underlying universes L.

In this finite setting, posets and lattices can be represented and clearly visualized by their Hasse-diagrams [12]. Also, the notions of infimum and supremum naturally extend from pairs to any subset of elements of a given lattice L. In this way, the notions of $\wedge$- and $\vee$-irreducible elements (that constitute the building blocks of lattices) can be defined as follows. For $x \in L$, let $x^* = \bigwedge(\uparrow x \backslash \{x\})$ and $x_* = \bigvee(\downarrow x \backslash \{x\})$. Then $x \in L$ is said to be a $\wedge$-*irreducible* element of L if $x \neq x^*$. Dually, x is said to be a $\vee$-*irreducible* element of L if $x \neq x_*$. We will denote the set of $\wedge$-irreducible elements and $\vee$-irreducible elements of L by $M(L)$ and $J(L)$, respectively. Observe that both $M(L)$ and $J(L)$ are posets when ordered by $\leqslant$.

We now recall the notions of poset and lattice homomorphisms.
Let $(P, \leqslant)$ and $(P', \leqslant')$ be two posets. A mapping $f\colon P \to P'$ is said to be a (*poset*) *homomorphism* if $x \leqslant y$ implies $f(x) \leqslant' f(y)$. In addition, if $f\colon P \to P'$ is injective (one-to-one), then it is called a (*poset*) *embedding*. If it is a bijection and an embedding such that, for every $x', y' \in P'$, $x' \leqslant' y'$ implies $f^{-1}(x') \leqslant f^{-1}(y')$, then it is called a (*poset*) *isomorphism*.

In the case of lattices, the notions of homomorphism, embedding and isomorphism become more stringent. Let $(L, \wedge, \vee)$ and $(L', \wedge', \vee')$ be two lattices. A mapping $f\colon L \to L'$ is said to be a (*lattice*) *homomorphism* if $f(x \wedge y) = f(x) \wedge' f(y)$ and $f(x \vee y) = f(x) \vee' f(y)$. In addition, if $f\colon L \to L'$ is injective, then it is called a (*lattice*) *embedding*. If it is a bijection and it is an embedding such that f^{-1} is also an embedding, then it is called a (*lattice*) *isomorphism*. When it is clear from the context, we will drop "(poset)" and "(lattice)" and simply refer to homomorphism, embedding and isomorphism.

It is noteworthy that the image $f(L)$ of a homomorphism $f\colon L \to L'$ is a sublattice of L', and that two isomorphic lattices have the same Hasse diagram. In particular, two lattices L and L' are isomorphic if and only if both (1) $J(L)$ and $J(L')$, and (2) $M(L)$ and $M(L')$ are isomorphic. In the case of distributive lattices, Birkhoff [7] showed that (1) suffice to guarantee that L and L' are isomorphic ($(J, \leqslant)$ and $(M, \leqslant)$ are isomorphic). The latter result is key ingredient in *Birkhoff's representation of distributive lattices* that we will discuss in Section 3, and that we will use in Section 4 to devise an algorithm to modify any finite lattice into an "optimal" distributive lattice containing it.

2.2 Formal Concept Analysis

Reduced Contexts, Concepts and Concept Lattices. We denote by $C = (O, A, I)$ a formal context where O is a set of objects, A a set of attributes and I an incidence relation between objects and attributes. In phylogenetic data, objects are usually species, attributes are mutations, and $(o, a) \in I$ –or oIa– when mutation a is spotted in species o.

Definition 1 (Galois connections). *For a set $X \subseteq O$, $Y \subseteq A$ we define:*

$$X' = \{y \in A \mid xIy \text{ for all } x \in X\}$$
$$Y' = \{x \in O \mid xIy \text{ for all } y \in Y\}$$

Then a formal concept is a pair (X, Y), where $X \subseteq O$, $Y \subseteq A$ and $X' = Y$ and $Y' = X$. X is the extent and Y is the intent of the concept. The set of all formal concepts ordered by inclusion of the extents –dually the intents– denoted by $\leqslant$ generates the concept lattice of the context $C = (O, A, I)$.

For $o \in O$, $\gamma o = (o'', o')$ denotes the concept introducing object o. For $a \in A$, $\mu a = (a', a'')$ denotes the concept introducing attribute a.

A *clarified context* is a context such that $x' = y'$ implies $x = y$ for any element of O and any element of A. Moreover, a clarified context is *reduced* iff it contains:

- no vertex $x \in O$ such that $x' = X'$ with $X \subseteq O$, $x \notin X$
- no vertex $x \in A$ such that $x' = X'$ with $X \subseteq A$, $x \notin X$

The reduced context is also called a *standard context.* Note that the standard context of lattice L is such that $O = J(L)$ and $A = M(L)$.

Arrow Relations.

Definition 2. *Let us consider a context (O, A, I), an object $o \in O$ and an attribute $a \in A$, then:*

- $o \nearrow a$ *iff* $(o, a) \notin I$ *and if* $a' \subseteq x'$, $a' \neq x'$ *then* $(o, x) \in I$
- $o \swarrow a$ *iff* $(o, a) \notin I$ *and if* $o' \subseteq x'$, $o' \neq x'$ *then* $(x, a) \in I$
- $o \swarrow\!\!\!\nearrow a$ *iff* $o \nearrow a$ *and* $o \swarrow a$

Stated differently, $o \swarrow a$ iff o' is maximal among all object intents which do not contain a. It can be shown that:

$$o \swarrow a \quad \Leftrightarrow \quad \gamma o \in J(L) \text{ and } \gamma o \wedge \mu a = (\gamma o)_*(\text{with } x_* = \bigvee(\downarrow x \backslash \{x\}))$$
$$o \nearrow a \quad \Leftrightarrow \quad \mu a \in M(L) \text{ and } \gamma o \vee \mu a = (\mu a)^*(\text{with } x^* = \bigwedge(\uparrow x \backslash \{x\}))$$

Arrow relations are related to irreducible elements in $J(L)$ and $M(L)$. In the following, we only consider arrow relations in reduced contexts.

An alternative equivalent definition of arrow relations is the following:

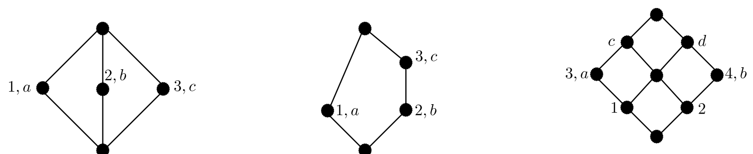
Definition 3. *Let L be a lattice, $j \in J(L)$ and $m \in M(L)$, then:*

- $j \nearrow m$ *iff* $\mu m \in max(L \backslash \uparrow \gamma j)$ *where* $max(.)$ *denotes the maximal elements.*
- $j \swarrow m$ *iff* $\gamma j \in min(L \backslash \downarrow \mu m)$ *where* $min(.)$ *denotes the minimal elements.*
- $j \swarrow\!\!\!\nearrow m$ *iff* $j \nearrow m$ *and* $j \swarrow m$.

$C = (J, M, I, \swarrow, \nearrow)$ is the reduced context with arrow relations. It can be represented by a table with (irreducible) objects in lines, (irreducible) attributes in columns, and in cell (j, m) (intersection of row j and column m):

- × if $j \leqslant m$ where $\leqslant$ is the partial ordering in the concept lattice,
- $\swarrow$ if $j \swarrow m$,
- $\nearrow$ if $j \nearrow m$,
- ⤢ if $j \swarrow m$ and $j \nearrow m$,
- otherwise an empty cell.

Fig. 1 shows three examples of reduced contexts with arrow relations $C = (J, M, I, \swarrow, \nearrow)$ and the corresponding concept lattices. The two first lattices on the left are respectively named M_3 and N_5 and they are the smallest non-distributive lattices. The third lattice on the right is a distributive lattice.



M_3 N_5 Distributive lattice

	a	b	c
1	×	⤢	⤢
2	⤢	×	⤢
3	⤢	⤢	×

	a	b	c
1	×	↙	⤢
2	⤢	×	×
3	↗	⤢	×

	a	b	c	d
1	×	⤢	×	×
2	⤢	×	×	×
3	×		×	⤢
4		×	⤢	×

Fig. 1. Three lattices and their reduced contexts with arrow relations.

3 Distributive Lattices and Their Representation

A lattice is *distributive* if $\wedge$ and $\vee$ are distributive one with respect to the over. Formally, a lattice L is distributive if for every $x, y, z \in L$, we have that one (or, equivalently, both) of the following identities holds:

$$(i)\ x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z), \quad (ii)\ x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z).$$

Distributive lattices appear naturally in any classification task or as computation and semantic models; see, e.g., [11,12,16,17]. This is partially due to the fact that *any distributive lattice can be thought of as a sublattice of a power-set lattice, i.e., the set $\mathcal{P}(X)$ of subsets of a given set X*. This result is a corollary to Birkhoff's representation of distributive lattices that we will further discuss in Subsection 3.2.

3.1 Characterization of Distributive Lattice

The distributive property of lattices has been equivalently described in several ways. We recall a few useful characterizations that we will use in the following sections of the paper.

Theorem 1. *A lattice L is distributive if and only if one (or, equivalently, all) of the following conditions hold:*

1. $(x \wedge y) \vee (y \wedge z) \vee (z \wedge x) = (x \vee y) \wedge (y \vee z) \wedge (z \vee x)$*;*
2. *L does not contain neither N_5 nor M_3 as sublattice;*
3. *the reduced context of L with arrow relations contains exactly one double-arrow $\swarrow\!\!\!\nearrow$ in each row and in each column, and no other arrows.*

The first characterization establishes a correspondence between distributive lattices and median algebras. Indeed, a median algebra is a structure (M, m) where M is a nonempty set and $m : M^3 \to M$ is an operation, called *median operation*, that satisfies the following conditions $m(a, a, b) = a$ and $m(m(a, b, c), d, e) = m(a, m(b, c, d), m(b, c, e))$, for every $a, b, c, d, e \in M$. It is not difficult to see that if L is distributive, then $m(a, b, c) = (a \wedge b) \vee (b \wedge c) \vee (c \wedge a)$ is a median operation. The connection to *median graphs* was established by Avann [1] who showed that every median graph is the Hasse diagram of a median algebra (thought of as a semilattice). For further background see, e.g., [2].

The second characterization describes distributive lattices in terms of two forbbiden structures, namely, M_3 and N_5 (see Fig. 1) that are, up to isomorphism, the smallest non distributive lattices. The third characterization is given in terms of formal contexts and it is also illustrated in Fig. 1: neither M_3 nor N_5 are distributive since

– for M_3, there are two double arrows by row and column;
– for N_5, there is one double arrow by row and column, but additional simple arrows.

3.2 Distributive Lattices and Ideal Lattices

Let $(P, \leqslant)$ be a poset and consider the set $\mathcal{O}(P)$ of ideals of P, i.e.,

$$\mathcal{O}(P) = \{\bigcup_{x \in X} \downarrow x \mid X \subseteq P\}.$$

It is well-known that for every poset P, the set $\mathcal{O}(P)$ ordered by inclusion is a distributive lattice, called *ideal lattice* of P, and that two posets P and P' are isomorphic if and only if $\mathcal{O}(P)$ and $\mathcal{O}(P')$ are isomorphic as lattices. Furthermore, the poset of $\vee$-irreducible elements of $\mathcal{O}(P)$ is

$$J(\mathcal{O}(P)) = \{\downarrow x \mid x \in P\}$$

and it is (order) isomorphic to P.

Dually, we saw in Subsection 2.1 that for any lattice L the set $J(L)$ of $\vee$-irreducible elements of L is a poset ordered by inclusion, and that if two lattices L and L' are isomorphic, then $J(L)$ and $J(L')$ are also isomorphic (as posets). Moreover, for any lattice L the set $\mathcal{O}(J(L))$ of ordered ideals of $J(L)$ is a distributive lattice that contains an isomorphic copy of L as a subposet. In particular, if L is isomorphic to $\mathcal{O}(J(L))$, then L must be distributive. The representation theorem of Birkhoff [6] states that the converse is true.

Birkhoff's Representation Theorem 1 *Let L be a (finite) distributive lattice and $J(L)$. Then the mapping $x \to \downarrow x \cap J(L)$ is a (lattice) isomorphism from L to $\mathcal{O}(J(L))$.*

As immediate consequences we have that every (finite) distributive lattice can be thought of as a sublattice of a powerset lattice or, equivalently, as a lattice of ideals of a poset. Figure 2 illustrates the latter assertion: on the left is a poset P, and on the right is the lattice of ideals of P. For an arbitrary lattice L,

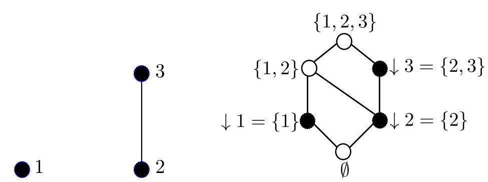


Fig. 2. Illustration of Birkhoff's Representation Theorem.

not necessarily distributive, there may be several lattices such that their poset of $\vee$-irreducible elements are isomorphic but only one of them is a distributive lattice [9,18]. Our goal is to make use of the previous results to present an algorithmic approach that receives a lattice L as input, and outputs an "optimal" distributive lattice L_d such that $(J(L), \leqslant)$ is isomorphic to$(J(L_d), \leqslant_d)$. Here, by "optimal" it should be understood "with the least number of modifications" (notably, insertions).

4 Proposal for Building a Distributive Lattice

From any lattice L, we want to obtain a distributive one L_d. Moreover, we want L_d to be "similar" to L. In this work, L_d is considered as similar to L if the posets of $\vee$-irreducible elements of L_d and of L are isomorphic. In this case, L can be embedded in L_d (L_d is a $\vee$-completion of L).

With this definition of "similar" (which can dually be applied to $\wedge$-irreducible elements), we can use Birkhoff Representation Theorem to compute L_d or its reduced context.

The main idea of algorithm 1 is to compute the context of L_d from the reduced context of L as input. Our approach relies on the underlying poset $(J, \leqslant)$ which is used to compute M_d.

Property 1. Algorithm 1 outputs the reduced context of the ideal lattice of $J(L)$.

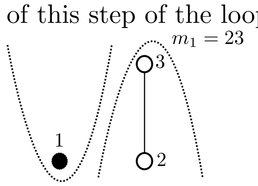
Proof. By construction, there is only one double-arrow by row and by column, and no other arrows. It follows that C_d is the context of a distributive lattice As discussed in section 2.1, this lattice is the ideal poset of $(J(L), \leqslant)$. It follows that $(J(L), \leqslant)$ and $(J(L_d), \leqslant_d)$ are isomorphic. □

Algorithm 1: Construction of context of distributive lattice.

Data: Reduced context $C(J, M, I)$
Result: Reduced context $C_d(J, M_d, I_d)$ of $(\mathcal{O}(J), \subseteq, \cap, \cup)$
$M_d \leftarrow \varnothing$
$I_d \leftarrow \varnothing$
foreach $j \in J$ **do**
  $\uparrow j \leftarrow \varnothing$
  foreach $i \in J$ **do**
    if $j' \subseteq i'$ **then** $\uparrow j \leftarrow \uparrow j \cup i$
  $M_d \leftarrow M_d \cup m_j$ // add a $\wedge$-irreducible element m_j such that $j \swarrow\!\!\!\nearrow m_j$
  $X \leftarrow J \setminus \uparrow j$ // elements of poset J which are not greater than j
  foreach $x \in X$ **do**
    $I_d \leftarrow I_d \cup (x, m_j)$

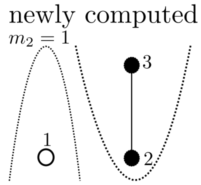
To illustrate the algorithm, we use N_5 context as input. At the beginning of the algorithm, the context C_d has $|J|$ rows but zero columns. Each step of the external loop computes m_j, a new $\wedge$-irreducible element of C_d such that $j \swarrow\!\!\!\nearrow m_j$.

Step 1. Computation of m_1 using $J \setminus \uparrow 1$. The algorithm computes the $\vee$-representation of m_1, the $\wedge$-irreducible element such that $1 \swarrow\!\!\!\nearrow m_1$. At the end of this step of the loop, C_d has a unique column which correspond to m_1.



	m_1
1	
2	×
3	×

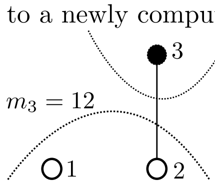
Step 2. Computation of m_2 using $J \setminus \uparrow 2$. The algorithm computes the $\vee$-representation of m_2, the $\wedge$-irreducible element such that $2 \swarrow\!\!\!\nearrow m_2$. At the end of this step of the loop, C_d has two columns which correspond to m_1 and to a newly computed element m_2.



	m_1	m_2
1		×
2	×	
3	×	

Step 3. Computation of m_3 using $J \setminus \uparrow 3$. The algorithm computes the $\vee$-representation of m_3, the $\wedge$-irreducible element such that $3 \swarrow\!\!\!\nearrow m_3$. At the end of this step of the loop, C_d has three columns which correspond to m_1, m_2 and

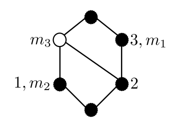
to a newly computed element m_3.



	m_1	m_2	m_3
1		×	×
2	×		×
3	×		

The whole context C_d for L_d is now computed. By construction, the only arrow relations are double arrows between j and m_j Below, L_d is drawn with black circles for concepts which were present in L and white circles for new concepts.

	m_1	m_2	m_3
1	⤢	×	×
2	×	⤢	×
3	×		⤢



5 Discussion and Conclusion

Motivated by the work of Priss [20] on the use of FCA on phylogenetic problems, we have proposed an algorithmic approach to compute the reduced context of a distributive lattice L_d from the reduced context of any lattice L, that ensures an order embedding from L into L_d that preserves $\wedge$. So, L_d can be considerated "not too far" from L and thus suitable for applications in phylogeny. In the remainder of this final section, we discuss some features of this algorithm.

First, we discuss an interpretation of the behavior of the algorithm for phylogenetic data. The algorithm computes $\wedge$-irreducible elements of L_d without any consideration for $\wedge$-irreducible elements of L but, as discussed in Subsection 3.2, this is not a problem. Now, for real data, two cases may appear:

1. $\mu m_j \in L$: in this case, we can use the initial label m of the object (this label may represent a particular gene mutation);
2. $\mu m_j \notin L$: this case suggests a gene mutation that is not spotted in the data, but that is necessary to provide a parsimonious tree.

Similarly, it is possible that $m \in M(L)$ but $m \notin M(L_d)$ but in any case, μm exists in L and L_d. This is the case when a mutation m is regarded as the infimum of other mutations.

Second, we propose an algorithm to build the context of a distributive lattice from any context. However, it is only a partial solution to the problem considered in [20]:

> "an algorithm for converting a concept lattice [into a median graph] consists of omitting the bottom node and then checking every principal filter for distributivity and turning it into a distributive lattice if it is not already one."

In the following, we discuss the whole process presented in [20]. We have proposed an algorithmic approach to "turning it into a distributive lattice if it is not already one". However, there is still some work to do as the suggestion in [20] does provide suitable solutions. This is illustrated by the example given in Figure 3.

	a	b	c	d	e
1	×	×		×	
2		×	×		×
3	×			×	
4			×		×
5				×	
6					×

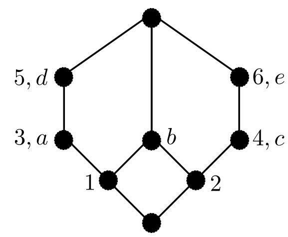


Fig. 3. Problematic context and associated concept lattice

Indeed, if we were to follow the steps suggested by Priss [20] on this example, the procedure would not provide a correct solution (i.e., a distributive lattice for principal filters). Consider a local approach on $\uparrow 1$ and $\uparrow 2$. The first step is to compute the reduced context of $\uparrow 1$ (since the example is symmetric for 1 and 2, we only give details for 1). The reduced context C^1 of $L^1 = \uparrow 1$ can be built from $C(J, M, I)$ by first observing that $1' = \{a, b, d\}$, which entails the following context:

	a	b	d
1	×	×	×
2		×	
3	×		×
4			
5			×
6			

and that reduces to:

	a	b	d
2		×	
3	×		×
5			×

Algorithm 1 then returns the context C_d^1 of a distributive lattice; similarly, Algorithm 1 returns context C_d^2 of $L^2 = \uparrow 2$.

C_d^1	m_2	m_3	m_5
2		×	×
3	×		×
5	×		

C_d^2	m_1	m_4	m_6
1		×	×
4	×		×
6	×		

Moreover, in the whole lattice, every $m \in M^1$ is greater than 1 and every $m \in M^2$ is greater than 2. Hence we obtain the left context for the whole lattice and the reduced context on the right:

	m_2	m_3	m_5	m_1	m_4	m_6
2		×	×	×	×	×
3	×		×			
5	×					
1	×	×	×		×	×
4				×		×
6				×		

	m_2	m_5	m_1	m_6
2		×	×	×
3	×	×		
5	×			
1	×	×		×
4			×	×
6			×	

The resulting lattice is presented in Figure 4.a; not every principal filter is distributive. The problem comes from the fact that the modified parts of the lattice belong to intersection of $\uparrow 1$ and $\uparrow 2$. The new added elements in a filter may belong to other filters, and this may "break" the consistency achieved in the other filters.

Now we applied this procedure in parallel for $\uparrow 1$ and $\uparrow 2$, and someone could argue that it should be iterated filter by filter until a fixed point is reached. Nevertheless, an optimal solution cannot be found through the general procedure suggested by Priss [20], since all filters must be considered simultaneously. In the present case, there exists an optimal solution with only one new concept. This solution is given in Figure 4.b

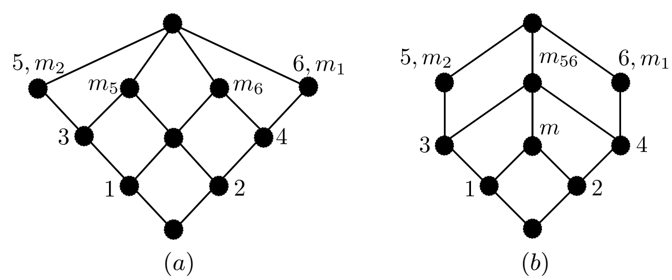


Fig. 4. (a) Solution obtained after a local approach and (b) optimal solution.

The difficulty of simultaneously considering all the filters should be studied and solved to deal with phylogenetic data. This entails to the two following open problems.

Problem 1 (Lattice version). Given a lattice L, propose an efficient algorithm to output a lattice L_d such that:

– for each atom x of L_d, $\uparrow x$ is a distributive lattice,
– there is an order embedding from L to L_d, and
– $|L_d| - |L|$ is minimal.

Problem 2 (Context version). Given the reduced context of a lattice L, propose an efficient algorithm to output the reduced context of a lattice L_d such that:

– for each atom x of L_d, $\uparrow x$ is a distributive lattice,

– there is an order embedding from L to L_d, and
– $|L_d| - |L|$ is minimal.

We are currently working on these two variations of the problem. The objective is to establish an operational bridge between FCA (concept lattices) and distributive lattices to allow the use of FCA algorithms in phylogeny.

References

1. Avann, S.P.: Median algebras. Proceedings of the American Mathematical Society **12**, 407–414 (1961)
2. Bandelt, H.J., Hedlíková, J.: Median algebras. Discrete mathematics **45**(1), 1–30 (1983)
3. Bandelt, H.J., Forster, P., Röhl, A.: Median-joining networks for inferring intraspecific phylogenies. Molecular biology and evolution **16**(1), 37–48 (1999)
4. Bertrand, P., Diatta, J.: Multilevel clustering models and interval convexities. Discrete Applied Mathematics **222**, 54–66 (2017)
5. Bertrand, P., Janowitz, M.F.: Pyramids and weak hierarchies in the ordinal model for clustering. Discrete Applied Mathematics **122**(1), 55–81 (2002)
6. Birkhoff, G.: On the combination of subalgebras. In: Mathematical Proceedings of the Cambridge Philosophical Society. vol. 29, pp. 441–464. Cambridge University Press (1933)
7. Birkhoff, G.: Lattice theory. In: Colloquium Publications, vol. 25. Amer. Math. Soc., 3. edn. (1967)
8. Birkhoff, G., Kiss, S.A.: A ternary operation in distributive lattices. Journal of Symbolic Logic **13**(1), 50–51 (1948)
9. Bordalo, G.H., Monjardet, B.: The lattice of strict completions of a poset. Electronic Notes in Discrete Mathematics **5**, 38–41 (2000)
10. Carpineto, C., Romano, G.: Concept Data Analysis: Theory and Applications. John Wiley & Sons (2004)
11. Caspard, N., Leclerc, B., Monjardet, B.: Ensembles Ordonnés Finis: Concepts, Résultats et Usages, vol. 60. Springer (2007)
12. Davey, B.A., Priestley, H.A.: Introduction to Lattices and Order. Cambridge university press (2002)
13. Diatta, J.: Galois weak hierarchies: Theoretical and computational issues. In: UK Workshop on Computational Intelligence. p. 3 (2005)
14. Ganter, B., Wille, R.: Formal Concept Analysis: Mathematical Foundations. Springer (1999)
15. Grätzer, G.: General Lattice Theory (Second Edition). Birkhäuser (1996)
16. Hopcroft, J.E., Motwani, R., Rotwani, Ullman, J.D.: Introduction to Automata Theory, Languages and Computability. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2nd edn. (2000)
17. Mattern, F.: Virtual time and global states of distributed systems. Parallel and Distributed Algorithms **1**(23), 215–226 (1989)
18. Nation, J., Pogel, A.: The lattice of completions of an ordered set. Order **14**(1), 1–7 (1997)
19. Priss, U.: Concept lattices and median networks. In: CLA. pp. 351–354 (2012)
20. Priss, U.: Representing median networks with concept lattices. In: ICCS. pp. 311–321. Springer (2013)