Chapter 2: Order theoretic and FCA lattice definitions and notation

This chapter defines the basic lattice terminology and notation used throughout this text. The basic order-theoretic lattice definitions are well known and may be found in many standard mathematical texts, for example in Gratzer (1971). Building on the order-theoretic definitions the notation and definitions of Formal Concept Analysis (FCA) are then introduced (Ganter and Wille (1999)) and the basic FCA building block, the *formal concept lattice*, is defined.

Next an *EA-lattice* is defined. An EA-lattice is closely related to a FCA lattice and is described as a substantially equivalent lattice to a formal concept lattice. EA-lattices have some desirable features that make it suitable for the use in compressed pseudo-lattices (chapter 6). The different views of a lattice are also explored. Both the EA-lattice and the different views of a lattice are concepts that are frequently used in the rest of the chapters.

The chapter concludes by defining the *intent- and extent representative operations* on a lattice.

An important aspect of the intent- and extent representative operations is that they are defined specifically for use in concept sub-lattices, such as those formed by removing atoms and/or coatoms from formal concept lattices. In chapter 6 the notion of removing concepts from lattices is formalised and generalised into additional concept lattice operations and the definition of a data structure called a compressed pseudo-lattice. The removal of concepts from concept lattices have been proposed by other authors (refer to chapter 8 for a discussion) but the approach taken here is more general. The concept of an EA-lattice (section 2.5) is defined and compared to a formal concept lattice (section 2.3).

2.1 LIST AND SET NOTATION

In this text, sets are denoted by capitals e.g. S. whilst the set elements are in lower case e.g. x, y or z. A set is indicated by the notation $\{x, y, z\}$ or $\{a_1, a_2, \ldots, a_n\}$.

The cardinality of set S is denoted by $\|S\|$.

Ordered lists are shown as $\langle x, y, z\rangle$.

2.2 ORDER THEORETIC LATTICE DEFINITIONS

A *binary relation on a set* is an association between pairs of elements of the set.

Consider a set S and arbitrary elements x, y and z in S. A partial ordering relation, $\leqslant$, on S is a binary relation that is reflexive ($x \leqslant x$), antisymmetric ($x \leqslant y \wedge y \leqslant x \Rightarrow x = y$) and transitive ($x \leqslant y \wedge y \leqslant z \Rightarrow x \leqslant z$). The set S in conjunction with an associated partial ordering relation, $\leqslant$, is called a *partially ordered set*[1] or *poset* and is denoted by $\langle S, \leqslant \rangle$. For $x, y \in S$, $x \neq y$, x is said to *cover* y, denoted by $y \prec x$ when $y \leqslant x$ and there is no $z \in S$, $z \neq x$, $z \neq y$ such that $y \leqslant z$ and $z \leqslant x$.

When $y \prec x$, some texts refer to x as the *parent, predecessor, upper cover* or *upper neighbour* of y. Similarly y is referred to as the *child, successor, lower cover or lower neighbour* of x.

One way of visually representing a poset is by means of a directed graph called a *line diagram* in which elements of the poset form the nodes and a directed arc (or edge) is drawn from node y to x iff $y \prec x$. Line diagrams are often referred to as a *Hasse diagrams.* They provide a natural data structure for visually representing posets. By convention, instead of showing the direction of arcs explicitly in the line diagram, node x is shown above node y if $y \prec x$. By virtue of the transitivity of the partial ordering relation, line diagrams are directed acyclic graphs.

Two elements x, y of a poset $\langle S, \leqslant \rangle$ are called *comparable* if $y \leqslant x$ or $x \leqslant y$. If these conditions are not met they are said to be *not comparable*.

Consider a poset L. $x \in L$ is an *upper bound* of $H \subseteq L$ iff $y \leqslant x$ for all $y \in H$. Out of all the upper bounds of H in L, the *least upper bound* of H (if it exists) is called its *supremum* and is denoted by Sup(H) or Sup(L, H). Likewise, $x \in L$ is a *lower bound* of a set $H \subseteq L$ iff $x \leqslant y$ for all $y \in H$. The *greatest lower bound* of H (if it exists) is called its *infimum* and is denoted by Inf(H) or Inf(L, H). A poset $\langle L, \leqslant \rangle$ is a *lattice* iff Sup({x, y}) and Inf({x, y}) exist for all pairs $x, y \in L$. It is not difficult to show that if Sup(H) exists then it is unique, and likewise for Inf(H). Some texts refer to a supremum as a *join* and the infimum as a *meet.* Two or more elements of a poset are also said to *meet* at their infimum. A poset S is a *complete lattice* if the supremum and infimum exist for all subsets of S. It can be shown that all non-empty finite posets that are lattices are complete. A subset U of a complete lattice V that is closed under both suprema and infima is called a *complete sublattice.*

A complete lattice L has a largest element called the *unit element*, denoted by 1_L, and a smallest element called the *zero element*, denoted by 0_L. The elements in a lattice covering the zero element are often called *atoms* whilst the elements covered by the unit element are called *coatoms*.

The *upward closure* of any element c, indicated by UpwardClosure(L, c), is the set of elements greater than or equal to c in terms of the partial order. The *downward closure* of c is the set of elements that are less than or equal to c, and is indicated by DownwardClosure(L, c).

Figure 2.1 is the line diagram of the poset $\langle \{1, 2, 3, 4, 6, 8, 12, 24\}, | \rangle$ where $m|n$ means that m is a factor of (or divides) n. In the figure, 24 is the supremum of {3, 8} whilst 2 is the infimum of {8, 2}. It is easy to verify that both the supremum and infimum of any pair of elements exist, that in each case they are unique and this poset is therefore a lattice. The upward closure of 6 is {6, 12, 24} whilst its downward closure is {1, 2, 3, 6}. Since 24 is the largest element of the poset it is the unit element whilst 1, being the smallest element, is the zero element of the poset. Elements 2 and 3 are the atoms of the lattice whilst 8 and 12 are the coatoms.

[1] Ganter and Wille (1999) also refers to a partially ordered set as an ordered set but we avoid this terminology since it may cause confusion with that of a *completely ordered set* in which all the elements can be ordered from smallest to largest.

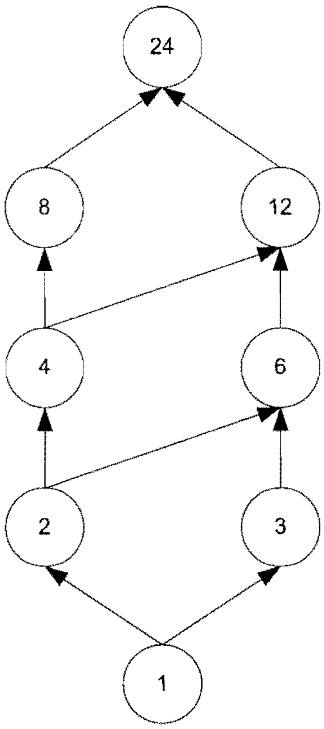


Figure 2.1: Lattice of $\langle\{1, 2, 3, 4, 6, 8, 12, 24\}, |\rangle$ *where* $m|n$ *means* m *is a factor of (or divides into)* n

In what follows the direction of the arcs will not be shown since there is no loss of generality in doing so.

Not all elements of a poset are necessarily related or comparable (hence the term *partial* order). A subset, S, of a poset whose elements are all comparable (i.e. $x \leqslant y$ or $y \leqslant x$ for all $x, y, \in S, x \neq y$) is called a *chain*. A subset, S, of a poset, none of whose elements are comparable is called an *anti-chain*.

The *width* of a finite poset is defined as the maximal size of an antichain in that poset. The *length* of a poset is defined as the supremum of the sizes of chains in the poset.

In a line diagram of a lattice, the nodes above a given element x are said to be *spanned* by x if there is a path from the x to the nodes. A node that spans a set of nodes is therefore a *lower bound* of the set of nodes. If for a node y there is a set of nodes with paths that end in y then y an *upper bound* of the set of lattice elements.

2.3 FCA DEFINITIONS

Consider a domain of discourse in which each element of a set of *objects*, $O = \{o_1, o_2, \ldots, o_j\}$, possesses one or more observable *attributes* from a set of attributes $A = \{a_1, a_2, \ldots, a_k\}$. We also refer to objects as *entities*, whilst attributes are sometimes referred to as *features* or *descriptors*. The triple $C = \langle O, A, I \rangle$, where I is a *binary relation* between O and A, $I \subseteq O \times A$, is referred to as a *context* and denotes this domain of discourse. The binary relation I, also called an *incidence relation*, identifies the attributes of each object. The notation oIa is used to indicate that object o possesses the attribute a. For any $E \subseteq O$ and $F \subseteq A$ the following operators are defined:

$E' = \{ a : A \mid (\forall o \in E)\ oIa \}$ – the set of attributes common to the objects in E

$F' = \{ o : O \mid (\forall a \in F)\ oIa \}$ – the set of objects common to the attributes in F

FCA studies posets known as *formal concept lattices* (also referred to as *Galois lattices*) that are induced by a binary relation over a pair of sets of objects and attributes. In FCA the context $C = \langle O, A, I \rangle$ is known as a *formal context*. A *formal concept* of a formal context is a couple $\langle E, F \rangle$ from $\mathcal{P}(O) \times \mathcal{P}(A)$ with $E \subseteq O$ and $F \subseteq A$ (where $\mathcal{P}(X)$ is the power set of the set X). In addition, the following property is satisfied:

$F = E'$ and $E = F'$

A formal concept (henceforth referred to simply as a concept) is thus a pair consisting of a set of related objects having some attributes in common and the set of precisely those attributes that all the objects have in common. E is also called the *extent* of the concept $c = \langle E, F \rangle$ while F is called the concept's *intent* denoted by Extent(c) and Intent(c) respectively. The set F with $F \subseteq A$ is the intent of some concept if and only if $(F')' = F$, in which case the concept of which F is the intent is precisely $\langle F', F \rangle$. Similarly $E \subseteq O$ is the extent of a concept $\langle E, E' \rangle$ iff $(E')' = E$. The *support* of a concept is defined as the number of objects in its extent.

The set of all formal concepts in a context can be shown (according to the basic theorem on concept lattices – see Ganter and Wille 1999) to constitute a lattice with respect to the partial ordering relation $\leq_c$ defined as:

$\langle E_1, F_1 \rangle \leq_c \langle E_2, F_2 \rangle$ iff $E_1 \subseteq E_2$ (or equivalently iff $F_1 \supseteq F_2$) for two concepts $c_1 = \langle E_1, F_1 \rangle$ and $c_2 = \langle E_2, F_2 \rangle$

This lattice is known as a *formal concept lattice* in FCA. Since only formal concepts are part of a concept lattice and since there is a direct relationship between E and E', and therefore either extent or intent of a concept in a formal concept lattice uniquely identifies the concept.

Incidence relation									
	nw	lw	ll	nc	1lg	2lg	mo	lb	sk
LE	✓	✓					✓		
BR	✓	✓					✓	✓	
FR	✓	✓	✓				✓	✓	
DG	✓		✓				✓	✓	✓
SW	✓	✓		✓	✓				
RD	✓	✓	✓	✓	✓				
BN	✓		✓	✓		✓			
MZ	✓		✓	✓	✓				

Entities/objects		Attributes	
LE	Leech	nw	Needs water
BR	Bream	lw	Lives in water
FR	Frog	ll	Lives on land
DG	Dog	nc	Needs chlorophyl
SW	Spike-weed	1lg	1 leaf germination
RD	Reed	2lg	2 leaf germination
BN	Bean	mo	Is motile
MZ	Maize	lb	Has limbs
		sk	Suckles young

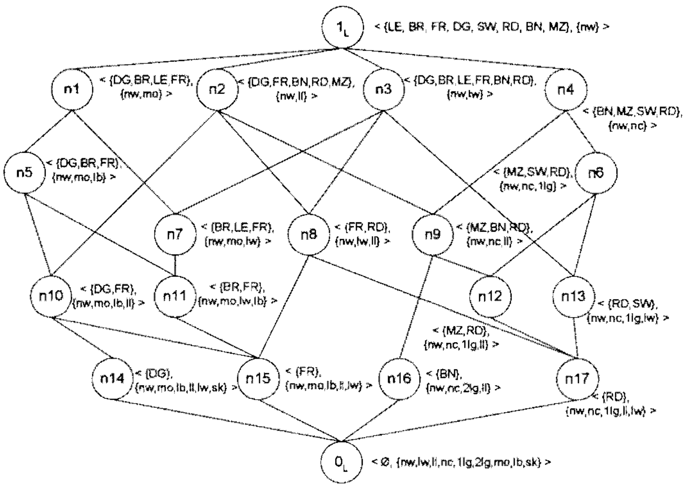


Figure 2.2: Context and formal concept lattice of the Living Context

A context can easily be represented by a *cross table*, i.e. by a matrix where the rows are labelled by the objects in the context and the columns by the attributes. A cross (or tick) in row g and column m indicates that the object g possesses the attribute m.

Figure 2.2 is the formal concept lattice of a small context. This simple context, called the Living Context, is taken from Ganter et al. (1986) and was originally used in a Hungarian educational film. The context is a simple ecological description of some living organisms.

Although very simplistic, it is useful for illustrative purposes. Each concept in the figure is numbered and is also labelled with its intent and extent.

The lattice shows a number of characteristics of the Living Context. For example the unit concept $1_L = \langle\{LE, BR, FR, DG, SW, RD, BN, MZ\}, \{nw\}\rangle$ has all the objects of the context in its extent but also has $\{nw\}$ as intent. This indicates that all the objects in the Living Context possess the attribute nw. Alternatively it can be said that all living things/objects (in the context) need water. By virtue of the partial ordering relation, $\leq_c$, it is clear that each concept y that is covered by x possesses at least all the attributes of x and at least one attribute in addition in its intent. Similarly the extent of x possesses at least the objects of the intent of y and at least one object in addition. Since there is no concept in the lattice above concept n_6 that has llg in its intent and both nw and nc are also contained in n_6's intent it shows that any object with llg as an attribute also has nc and nw as attributes. Concept n_4 with intent $\{nw, nc\}$ indicates that the converse is however not true, in that there are objects with nw and nc as attributes but not possessing llg for example BN.

A formal concept lattice is a useful structure since the formal concept lattice of a particular context contains encoded within its concepts all "meaningful" concepts in that only combinations of objects that actually have a particular set of attributes in common are grouped together in a concept. (This is as a result of the definition of formal concepts.) Similarly all groupings of attributes that does have common objects are represented in the lattice. Thus a set of attributes such as $\{nc, sk\}$ is not a "meaningful" set since there is no set of objects that has this particular and only this particular set of attributes in common. Another way of viewing this is that there is no evidence in the context to suggest that $\{nc, sk\}$ is a meaningful concept in the context. (Note however, that if the context is expanded in some way, this might not continue to be the case.) Similarly, the set of attributes $F = \{ll, lw\}$ is also not a grouping of attributes supported by the context. Using the operator defined earlier $F' = \{FR, RD\}$ and therefore the objects FR and RD are the only two having ll and lw in common. However applying the operator $(F')' = \{nw, ll, lw\}$ we see that whenever ll and lw are present for an object of the context, the attribute nw is *always* present. F is thus not a grouping of attributes supported by the evidence but $\{nw, ll, lw\}$ is and corresponds to concept n_8 of the lattice. This type of reasoning makes lattices a particularly useful tool in machine learning and in knowledge discovery in databases (KDD). Some of the many types of reasoning (e.g. abductive reasoning, inductive reasoning, unsupervised learning, supervised learning etc.) supported by a lattice are discussed in Oosthuizen (1994b).

2.4 WHY EA-LATTICES?

When algorithmically constructing formal concept lattices, for machine learning purposes based on using "real" or natural data, there are however a number of drawbacks. These drawbacks are the reason for defining EA-lattices, a class of lattices closely related to FCA lattices, in section 2.5.

- The same concept may simultaneously represent different objects and attributes in the context. This can happen when incrementally building the lattice and objects already represented in the lattice do not sufficiently differentiate the attributes or when there are duplicate objects (perhaps as a result of data errors or incomplete data). It is desirable to have each object and attribute represented by a separate concept because this correspond to the natural world where distinct objects are indeed acknowledged as being different although the initial information about their features might not be sufficient to indicate the precise nature of the differences. The same can be said in regard to differentiating features of objects. A "real world" example would be when two closely related books are described by a number of keywords. Because the texts of the books are so closely related the list of keywords describing each book could be the same for both books although they are still two separate books. In this example a concept lattice of books will typically be augmented with meta-information describing the location of the book in for example a library. If more than one object are represented by one concept a separate data structure needs to be created to store this meta-information. Should the different books be represented by different concepts, the lattice data structure could directly be used for this purpose without additional data structures. In the Living Context, concept n_{14} is clearly the most precise representative of the object DG. However, it is also the only representative of the attribute sk (sk occurs in no concept higher up in the lattice). Thus, despite the fact that object DG and attribute sk are different real-world concepts, they have the same representative in the formal concept lattice. The problem could conceivably be avoided by carefully choosing the attributes of each object or introducing more attributes, but this is not always possible.

- When incrementally building a lattice, the concepts corresponding to particular objects or attributes change as the lattice grows. When numbering the concepts in the data structure representing the lattice it is desirable to have the line diagram node corresponding to the particular object or attributes stay the same throughout the lifetime of the data structure. The node can then be used as an index into the data structure. A concept corresponds to an object if the extent of the concept contains only that object. A concept corresponds to an attribute if its intent contains only that attribute.

- The objects, attributes and so-called intermediate concepts of the lattice are not clearly partitioned in a formal concept lattice. In figure 2.2 it is difficult identifying the concepts corresponding to particular objects or attributes since they may be located on any level of the lattice. Indeed, as was pointed out above a concept, such as concept n_{14} may even "correspond" to both an object and an attribute.

- Attributes that are not present in any of the objects are not represented in a formal concept lattice. This can happen when the lattice is built incrementally and initially contains only a few objects, none of which contain the particular attributes. Similarly, if the lattice is being constructed by incrementally introducing new attributes, the objects without any attributes introduced into the lattice to date will not be represented in the lattice.

For the above reasons and for reasons related to the definition of a compressed pseudo-lattice (chapter 6), a related lattice called an *EA-lattice* or *entity attribute lattice* is defined. (Oosthuizen (1994b) and Kourie and Oosthuizen (1998) previously made mention of such lattices, but never formally defined them.) *Each* object and *each* attribute in an EA-lattice is represented as a separate concept that is not associated (or co-labelled with) with any other object or attribute. (Note that this need not be true in general for a formal concept lattice, as will be discussed below.) As a result, the concepts (excluding 1_L and 0_L) in an EA-lattice can be partitioned into three sets: the attribute concepts, the object concepts and the intermediate concepts respectively. (Note that such a partitioning is not necessarily possible in a formal concept lattice as, for example, when the extent of one attribute is a subset of another. The Living Context introduced in figure 2.2 is an example of a context where such a partition is not possible.) This corresponds to the real world in the sense that an object (or attribute) is acknowledged to be unique, even if there is initially insufficient evidence to support its uniqueness in the light of the data examined up until that time. Due to the one-to-one mapping from objects and attributes to concepts, it is thus permissible in an EA-lattice to talk of an *object concept* and an *attribute concept*.

2.5 EA-LATTICE DEFINITION

Assume that $\|A\| > 1$, $\|O\| > 1$. A concept $\langle E, F\rangle$, where $E \subseteq O$ and $F \subseteq A$, is called an *EA-formal concept* in a context $\langle O, A, I\rangle$ if any one of the following conditions are satisfied:

1. $\|E\| = 1$ and $F = E'$
2. $\|F\| = 1$ and $E = F'$
3. $E = \varnothing$ and $F = A$
4. $E = O$ and $F = \varnothing$
5. $F = E'$ and $E = F'$

The set of all EA-formal concepts from $\mathcal{P}(O) \times \mathcal{P}(A)$ in a formal context $C = \langle O, A, I\rangle$, is called an *EA-formal concept lattice*[2] (or simply an *EA-lattice*) with respect to the partial ordering relation $\leq_{EA}$[3] defined as:

$\langle E_1, F_1\rangle \leq_{EA} \langle E_2, F_2\rangle$ iff $E_1 \subseteq E_2$ or $F_1 \supseteq F_2$ for two concepts $c_1 = \langle E_1, F_1\rangle$ and $c_2 = \langle E_2, F_2\rangle$

In such a lattice, L, the *zero concept*, denoted by 0_L, corresponds to the EA-formal concept $\langle \varnothing, A\rangle$ (condition 3), whereas the *unit concept*, denoted by 1_L, corresponds to the EA-formal concept $\langle O, \varnothing\rangle$ (condition 4). *Attributes* are in the form $\langle F', F\rangle$ where F is a set containing only one element of A (condition 2). Objects are in the form $\langle E, E'\rangle$ where E is a set containing only one element of O (condition 1).

The lattice below is the corresponding EA-lattice of the Living Context applying the definition of an EA-lattice (for comparative purposes 0_L and 1_L are shown but will be excluded from further lattice diagrams). The same concept numbering of corresponding concepts in figure 2.2 is used to enable comparisons. Concepts that are exactly the same (in terms of their intents and extents) to that of the formal concept lattice are shaded.

[2] Note that some authors use the term 'lattice' interchangeably with 'formal concept lattices'. Here we distinguish between the order-theoretic term 'lattice' and 'sublattice' and the FCA terms 'formal concept lattice' and 'concept lattice' both of which are special 'order-theoretic lattices'. An EA-lattice is an order-theoretic lattice but not necessarily a formal concept lattice.

[3] It should be noted that $\leq_{EA}$ is a partial ordering relation only on EA-formal concepts and not on all possible concepts (this is also the case with $\leq_C$ and formal concepts).

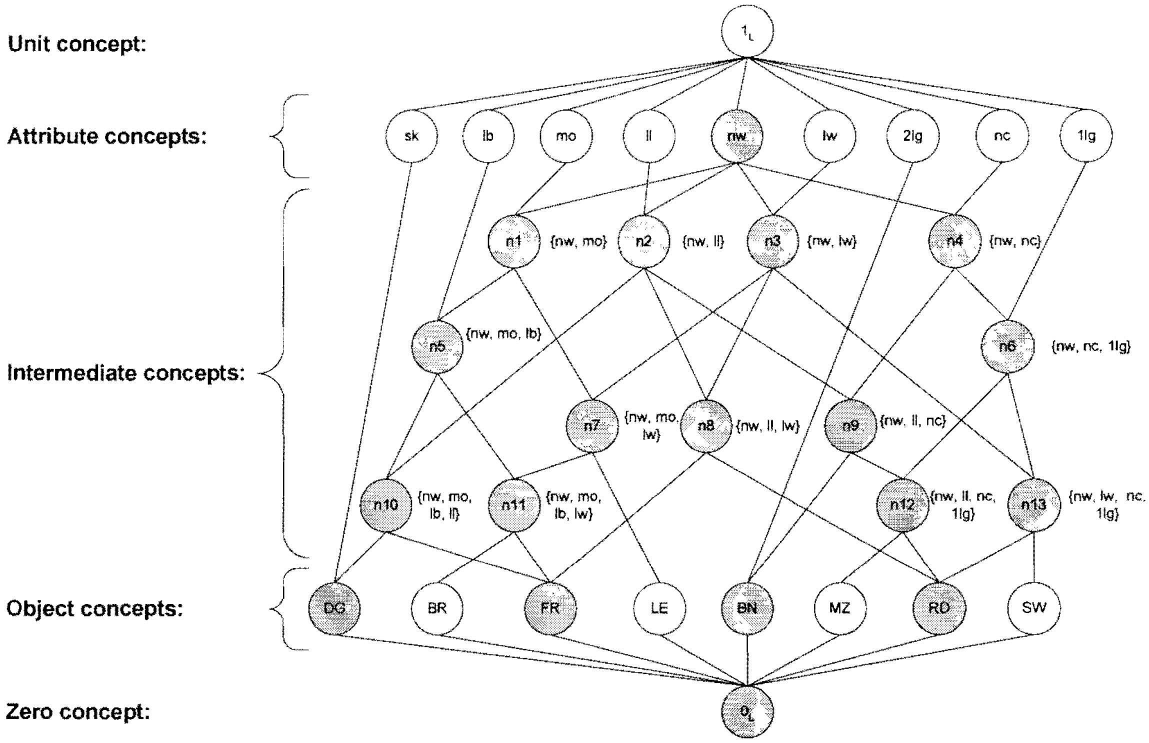


Figure 2.3: EA-lattice of the Living Context showing the partitioning of the concepts (compare to figure 2.2)

Note that an EA-lattice of a context differs only from the formal concept lattice if the concepts of conditions 1-4 are not generated by condition 5. This happens for example in a given context if an attribute f, is such that $F'' \neq F$, $F = \{f\}$. In such a case, $\langle F', F\rangle$ will be included in the context's EA-lattice, but not in the context's concept lattice in which case the concept $\langle F', F''\rangle$ will be labelled with the attribute f. Consider for example concepts MZ or sk in the EA-lattice in figure 2.3 and compare them to the formal concept lattice of the same context of figure 2.2. The concepts in the EA-lattice in figure 2.3 that correspond to the formal concepts in figure 2.2 are shaded and are identified by the fact that they have only one parent or child concept. In this example the zero concept is the same for the EA-lattice and formal concept lattice but not the unit concept. The EA-lattice is therefore essentially a generalisation of a formal concept lattice in the sense that it possibly contains a number of additional concepts. These additional concepts correspond **only** to objects (condition 1), attributes (condition 2), the unit concept (condition 3) or the zero concept (condition 4).

An *intermediate concept* of an EA-lattice is defined as a concept with more than one object in its extent and more than one attribute in its intent. Applying the same definition of intermediate concepts to formal concept lattices it can be seen from the definitions of both kinds of lattices of that the sets of intermediate concepts of the two types of lattices are identical.

It is important to note that an EA-lattice cannot use ordering relation $\leq_c$ since it is not antisymmetric for the given set of concepts. In the example above the antisymmetric property ($x \leq y \wedge y \leq x \Rightarrow x = y$) does not hold for $\leq_c$ in regard to the concept $nc = \langle\{BN, MZ, SW, RD\}, \{nc\}\rangle$ and $n_4 = \langle\{BN, MZ, SW, RD\}, \{nw, nc\}\rangle$. The slight modification of the partial ordering relation (from $\leq_c$ to $\leq_{EA}$) is therefore due to the fact that, in an EA-lattice, the intent or the extent of some attribute concepts and object concepts may be the same as some intermediate concepts. The supremum and infimum also need to be recomputed in terms of the revised partial ordering and may thus differ from those of a formal concept lattice. (The EA-lattice may for example have a different unit element compared to its corresponding formal concept lattice, as in the Living Context example). The EA-lattice is however for all practical purposes the same as a formal concept lattice (e.g. in lattice construction algorithms where only slight modifications are required).

From the definition it is easy to show that an EA-lattice has the following properties:

- There is one concept associated with each object e_i in the form $\langle E, E' \rangle$ where $E = \{e_i\}$. Similarly there is one associated concept for each attribute a_j in the form $\langle F', F \rangle$ where $F = \{a_j\}$. These concepts are called the *associated object concept* and the *associated attribute concept* respectively.
- Zero concept, $0_L = \langle \varnothing, A \rangle$ and unit concept, $1_L = \langle O, \varnothing \rangle$ are distinct from the object and attribute concepts and always have the same structure (i.e. empty extent and intent respectively).
- The EA-lattice, excluding the unit and zero concepts, can be partitioned into three sets corresponding to objects, attributes and intermediate concepts. The three sets correspond to the top, bottom and middle sections of the line diagram of a lattice (refer to figure 2.3).
- All EA-formal concepts in the formal concept lattice of a corresponding EA-lattice are contained in the EA-lattice. An EA-lattice thus has at least the same number of concepts as a formal concept lattice. Since the number of concepts in a large lattice is dominated by the intermediate concepts, the size of an formal concept and EA-lattice differs very little for large lattices. The difference in algorithmic complexity for their respective construction is also negligible (refer to chapter 5).
- The EA-lattice may contain a number of concepts in addition to those of the corresponding formal concept lattice. These additional concepts will correspond to the objects, attributes, unit concept or zero concepts, but there are no additional intermediate concepts.
- Each intermediate concept in an EA-lattice has at least two parent- and two children concepts. (Note that this is also true for non-atom and non-coatom concepts in boolean lattices but is not true in general for all FCA lattices such as the lattice in figure 2.2). Attributes concepts have only one parent (1_L) and only the attribute concepts generated by condition 5 have at least two child concepts. Other attribute concepts have one child concept. Similar dual observations can be made for object concepts.
- Each attribute has no parent concepts other than 1_L.
- Each object concept has no children concepts other than 0_L.
- Attribute concepts in the EA-lattice that do not correspond to a concept in the formal concept lattice have only one child concept. (Note that the corresponding concept in the formal concept lattice has only one parent.)
- Object concepts in the EA-lattice that do not correspond to a concept in the formal concept lattice have only one parent concept. (Note that the corresponding concept in the formal concept lattice has only one child concept.)
- In an EA-lattice the atoms and coatoms have a one-to-one relationship to the object- and attribute concepts of that lattice respectively.

In drawing the EA-lattice a number of conventions will be followed:

- The 1_L and 0_L concepts and associated cover relationships will not be shown so that the lattice is bounded from above by the attributes of the lattice and bounded from below by the objects of the lattice.
- The attribute and intermediate concepts are labelled with their intent only whilst object concepts are labelled with the object identifier itself.
- When convenient or to simplify the line diagram of a lattice, the labels of the concepts (especially the intermediate concepts) will be substituted for a concept number. From time-to-time a concept number and attribute list may also be used in conjunction especially for labelling intermediate concepts.

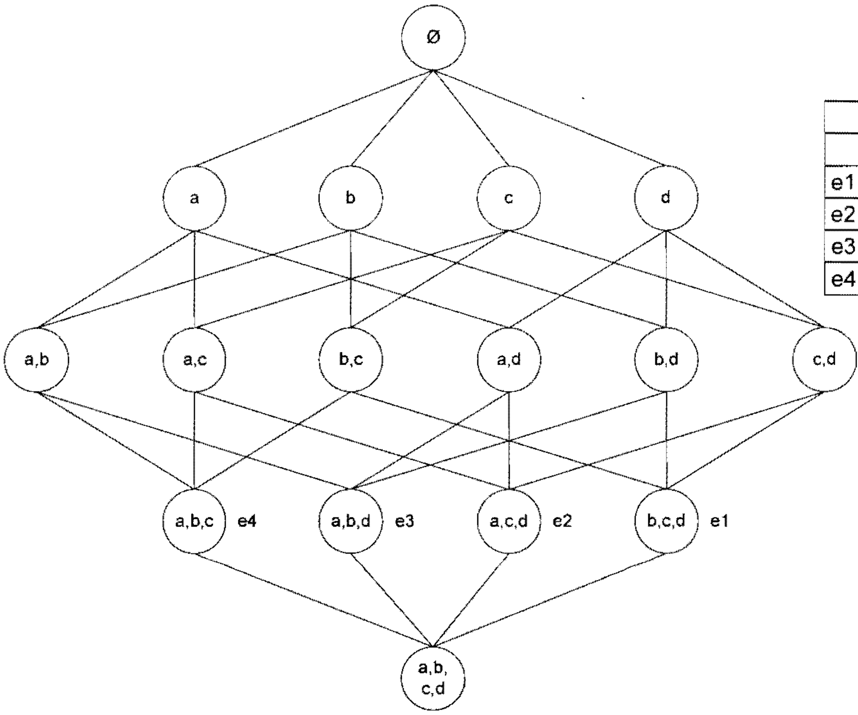
These conventions do not impact on the generality of the discussions or line diagrams since the complete line diagram and complete labels for concepts can be easily determined by inspecting the lattice using the definition of an EA-lattice (e.g. the intent of a concept is all the attributes contained in its upward closure).

As can be expected from the similarity in the definitions of the formal concept- and EA-lattices there is a direct mapping from the one to the other.

2.6 BOOLEAN LATTICES

A power-set lattice (Ganter and Wille (1999)) of a set of attributes A is the lattice $\langle \mathcal{P}(A), \subseteq \rangle$. A *Boolean lattice* is a lattice that is isomorphic to some power-set lattice. The figure below is the Boolean formal concept lattice of $A = \{a, b, c, d\}$. The corresponding incidence relation shows that a Boolean lattice is formed when there are as many objects as attributes and each object differs from each of the other objects by only one attribute. Since a Boolean lattice contains all concepts from $\mathcal{P}(O) \times \mathcal{P}(A)$ it follows that a Boolean lattice is both a formal concept- and an EA-lattice. The converse is however not true in general. Formal concept- and EA-lattices are not in general Boolean lattices.

Since a Boolean lattice contains all the possible concepts that can be formed with a given set of attributes, it also forms the theoretical upper limit of the size of a lattice with the given set of attributes. A Boolean lattice has exactly the same number of concepts as the elements of $\mathcal{P}(A)$. Since $\mathcal{P}(A)$ is exponential in terms of $||A||$ it follows that the number of concepts in a Boolean lattice is exponential in terms of $||A||$ and has $2^{||A||}$ elements (refer to chapter 5 for more formulas related to the various size aspects of a Boolean lattice).



Incidence relation				
	a	b	c	d
e1		✓	✓	✓
e2	✓		✓	✓
e3	✓	✓		✓
e4	✓	✓	✓	

Figure 2.4: Boolean lattice with four attributes and objects

2.7 AUGMENTED LATTICES

In applications such as machine learning and data mining that use formal concept lattices and EA-lattices it is common to augment or label the concepts of a lattice with additional meta-information that can be used in the application. These lattices are called augmented lattices (Kourie and Oosthuizen (1998)).

Examples of such meta-information could be:

- The support of a concept.
- A reference to a external database for object concepts.
- A name or descriptor for attribute and object concepts.
- Pre-computed upward and downward closures.

2.8 DIFFERENT VIEWS OF A LATTICE

In describing lattices, their properties and their construction, there are two different but essentially equal “views” of a lattice. One view of a lattice is to describe the lattice from a set theoretic point of view and the other from a graph theoretic point of view.

2.8.1 Set-theoretic view of lattices

The set-theoretic view of lattices emphasises the fact that a lattice consists of a set of concepts. Using the convention that we refer to a concept only by its intent (and assuming that there are no objects with the same intent), a formal concept- or an EA-lattice is essentially a set of sets (i.e. a set of concepts in which each concept is a set of attributes). In Kourie & Oosthuizen (1998), for example this view is taken.

Since concepts are merely sets, set operations such as union and intersection can be performed directly on (the intents of) concepts. The supremum of two concepts with intents A and B in a Boolean lattice is for example the concept with the intent $A \cap B$. The cover relationship between concepts is defined by set containment. One of the key aspects of the set-theoretic view is that the lattice is described in terms of its attributes (sometimes the objects) and since there are far fewer attributes than other (intermediate) concepts most computations are very efficient.

The disadvantage of the set-theoretic view is that we emphasise the set related properties of lattice concepts in favour of the graph-related properties especially the child-parent relationship of nodes in a graph. The advantage is however that lattice properties, operations and theorems are more easily proved because the lattice is essentially a construct defined in terms of sets, which have more easily manipulated mathematical relationships.

2.8.2 Graph-theoretic view of lattices

The graph-theoretic view of lattices emphasises the lattice as a collection of nodes (as opposed to sets of attributes) with the specific partial ordering relationship between them depicted by the arcs in the graph. The lattice is thus described in terminology such as "parent", "child", "upward closure", "downward link", "top", "bottom" etc. This greatly increases the understanding of the lattice concepts by newcomers to the subject since it refers to a graphic representation of the lattice namely the line diagram rather than the more abstract concept of a set of partially ordered sets of sets. In this view the intent and extent of concepts as well as the context can be inferred by closure operations. Although this view describes exactly the same lattice, different aspects of the lattice (in this case the line diagram representation) are promoted.

To distinguish between the two views we will follow the convention of referring to "concepts" when using the set-theoretic view and to "nodes" when using the graph theoretic view. Graph terminology such as "arcs", "parent node" and "child node" is also freely used when taking a graph theoretic view.

Often the graph properties of the lattice are emphasised in that nodes are numbered and referred to by number in-stead of their intent, extent or both. The intent and extent of the node is often not explicitly shown and must then be derived by the graph properties using closure operations.

The disadvantage of the graph-theoretic view is that the mathematical properties of the lattice may be obscured and compared to the set-theoretic view the proving of theorems is not straightforward (if one would be restricted to graph terminology only). An example of this is to be seen in Oosthuizen (1994b) where both the construction and application of lattices to machine learning from a graph-theoretic view are described. The graph-theoretic view is essentially a description of the lattice as a data structure consisting of nodes that have a number of node properties such as its extent, intent, child nodes and parent nodes.

In the set-theoretic view, a set of concepts can be proven to be a lattice by verifying that a unique infimum and supremum exist for any set of nodes. To do the same in the graph-theoretic view, the line diagram of the lattice may be inspected and the arcs leading upward or downward from a set of nodes are followed until they meet. If, for example, the arcs leading upward join at two or more nodes that are unrelated (i.e. the one is not a parent of the other), there does not exist a unique smallest upper bound and therefore the lattice property does not hold. A similar inspection should be done to verify the uniqueness of the greatest lower bound of each set of nodes. In figure 2.5 below both n_3 and n_5 are upper bounds for $\{n_7, n_8\}$ as can be seen by following the arcs in bold. However, n_3 and n_5 are unrelated. (In this example, it is of course assumed that there is in fact no top element – i.e. the assumption previously mentioned that the top is implicit has been lifted.) There is therefore not a unique least upper bound for $\{n7, n8\}$ and the graph therefore does not represent a lattice. Oosthuizen (1991) refers to the subgraph consisting of n_8, n_5, n_7, n_3 and the connecting arcs as a *quad* and describe the problem of algorithmically constructing a lattice as one in which an acyclic graph is constructed to connect all objects and attributes without any quads (i.e. every pair of nodes has a join and meet and that these should necessarily be unique). Note that any number of intermediate nodes can exist between the four "corner" nodes of a quad.

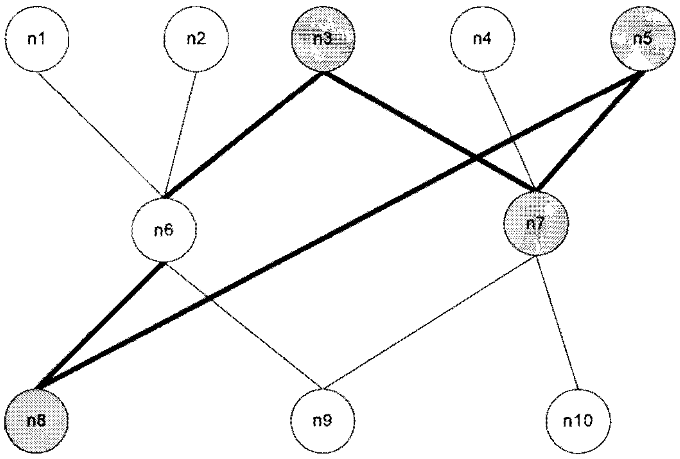


Figure 2.5: Example of a poset with non-unique infima and suprema of concepts n_3, n_5, n_7, n_8 (also called a quad)

Although the two viewpoints are essentially the same and there are many ways to combine the two views they represent the two main ways in which authors have chosen to describe lattices and especially the construction and application thereof. Authors such as Godin (1991) and Carpineto and Romano (1996b) tend to emphasise the set-theoretic view whilst Oosthuizen (1991) has favoured the graph-theoretic view in the description of their lattice construction algorithms.

Further on in this text it will be argued that the key to the AddAtom lattice construction algorithm is was developed using a more graph-theoretic view. The graph-theoretic view emphasises the use of the information that is contained in the lattice – the partial ordering of nodes and therefore the characteristics of the context from which the lattice was constructed.

In order to gain the advantages of both these views we will use both in various sections of this text. The set-theoretic view will be used to formally describe the lattice and its construction whilst the graph theoretic view will be used to informally describe the lattice construction concepts.

2.9 INTENT AND EXTENT OPERATIONS

In this section we define the *approximate intent representative* set (AIR) as well as the *exact intent representative* set (EIR) of a set of attributes with regards to a concept lattice or complete concept sublattice. These operations define in some sense a 'second-order meet operation' or 'second-order infimum' for concept lattices. The intent- and extent representative operations were originally defined in the context of compressed pseudo-lattices or concept sublattices where the infimum and supremum of some concepts in L have been removed (such lattices are under consideration in chapter 6). Since the removal of concepts from a lattice is the opposite of lattice construction there is a relationship between the two concepts. This relationship is explored in section chapter 6.

Let $Q \subseteq A$ be a set of attributes and H be the set of associated attribute concepts of Q in an EA-lattice L. The meet of the set of attribute concepts H, $\mathrm{Meet}(L, H)$ is a concept. If $\mathrm{Intent}(\mathrm{Meet}(L, H)) = Q$ then the $\mathrm{Meet}(L, H)$ is called an *exact meet* (or *exact infimum*) of the attributes in Q since it spans only the attribute concepts H and contains only elements of Q in its intent. If the meet contains additional attributes in its intent or alternatively spans attribute concepts other than H, then it is called an *approximate meet* (or *approximate infimum*) of the attributes. Similarly an *exact join* (or *exact supremum*) and *approximate join* (or *approximate supremum*) can be defined using a set G of object concepts associated with a set R of objects in stead of the set H of attribute concepts associated with the set Q.

Since the elements of the set of attributes, A, are not elements of an EA-lattice or concept lattice, L, the meet or infimum of a subset of A in L is not defined. Furthermore, since lattices of which concepts have been removed will be considered it is useful to define $\mathrm{Inf}'(L, Q)$, $Q \subseteq A$ as the maximal elements[4] of the set $\{x : L |\ \mathrm{Intent}(L, x) \supseteq Q\}$ (i.e. the set of greatest concepts that contains at least the attributes of Q in their intents). This function is closely related to the infimum or meet of a set of associated attribute concepts. In an EA-lattice $\mathrm{Inf}'(L, Q) = \mathrm{Inf}(L, H)$ where H is the set of attribute concepts associated with Q, but this function is also defined on lattices of which concepts (e.g. the associated attribute concepts themselves) have been removed and in which case $\mathrm{Inf}'(L, Q)$ may not be a single concept. Similarly $\mathrm{Sup}'(L, R)$ is the minimal elements of $\{x : L |\ \mathrm{Extent}(L, x) \supseteq R\}$ where R is a set of objects.

Consider a concept lattice or -sublattice, L. Let Q be a set of attributes, $Q \subseteq A$. Let S be the set of all elements of $\mathrm{Inf}'(L, F)$, $F \in \mathcal{P}(Q)$ (i.e. $S = \{x : L \mid \exists\, F \in \mathcal{P}(Q),\ x \in \mathrm{Inf}'(L, F)\}$). Exclude the zero concept from S. The *set of approximate intent representatives* of Q in L, denoted by $\mathrm{AIR}(L, Q)$, is the set of minimal concepts in S.

Now let T be that subset of S whose elements have intents that are not subsets of Q. The *set of exact intent representatives* of Q with respect to L, denoted by $\mathrm{EIR}(L, Q)$, is the set of minimal elements in $S - T$. If $T = \varnothing$ then clearly $\mathrm{EIR}(L, Q) = \mathrm{AIR}(L, Q)$.

From the definition it follows that the results of both $\mathrm{AIR}(L, Q)$ and $\mathrm{EIR}(L, Q)$ are anti-chains. They are said to be *infimum-dense*[5] and are therefore a concise way of representing Q. In the case of the exact intent representative there is a close relationship between Q and $\mathrm{EIR}(L, Q)$ since the intents of elements of $\mathrm{EIR}(L, Q)$ not only span the associated attribute concepts of Q, they constitute in fact, the minimal set of meets that do so. There is not necessarily such a direct mapping between Q and $\mathrm{AIR}(L, Q)$ in the sense that concepts in the intents of elements of $\mathrm{AIR}(L, Q)$ possibly contain attributes in addition to those in Q.

Dual extent operations for $\mathrm{AIR}(L, Q)$ and $\mathrm{EIR}(L, Q)$ can be defined as follows. R can be seen as a set of objects (instead of Q, the set of attributes) and Inf' and the maximal operations can be substituted by Sup' and minimal operations in the above definitions

---

[4] $x \in S$ is minimal, iff $\nexists y \in S$, $y \neq x$; such that $y \leq_{EA} x$. Similarly $x \in S$ is maximal, iff $\nexists y \in S$, $y \neq x$; such that $x \leq_{EA} y$.

[5] A set $X \subseteq Y$ is called infimum-dense in Y if every element from X can be represented as the infimum of a subset of Y.

respectively. The zero concept is replaced by the unit concept. This defines the set of *approximate extent representatives*, AER(L, R) and the set of *exact extent representatives*, EER(L, R). In an EA-lattice if Sup'(L, R) is non-trivial (i.e. if Sup'(L, R) is not 1_L), Sup'(L, R) = AER(L, R) = EER(L, R).

It is useful to define a further related set of operations, namely EIR(L, Q, c), $c \in L$. This is the set of *exact intent representatives of* Q *not less than* c. It corresponds identically to EIR(L, Q), except that in determining the minimal elements of S above, the downward closure of a designated concept, c, is specifically excluded from consideration. As a result, if c is in EIR(L, Q), then EIR(L, Q, c) contains no concepts that are less than or equal to c. In particular, if c = Meet(L, Q), then EIR(L, Q, c) is the set of concepts covering c in the sublattice L. The set of *exact extent representatives of* R *not greater than* c, EER(L, R, c) is defined similarly. It is easy to see that EIR(L, Q) = EIR(L, Q, 0_L). Similarly EER(L, R) = EER(L, R, 1_L). In the same way, the set of *approximate intent representatives of* Q *not less than* c, AIR(L, Q, c) is defined. The set of *approximate extent representatives of* R *not less than* c is similarly defined.

The operations defined here are collectively referred to as the *intent- or extent operations* of a concept lattice (or sublattice).

For example in figure 2.3 calculating AIR(L, {nw, nc, 1lg, 2lg}), S = {nw, 2lg, nc, 1lg, n4, n6, BN}. {BN, n_6} is the set of minimal elements of S and therefore AIR(L, {nw, nc, 1lg, 2lg}) = {BN, n_6}. In calculating EIR(L, {nw, nc, 1lg, 2lg}) we see that T = { BN } since BN also spans ll in addition. The minimal elements of S – T = { nw, 2lg, nc, 1lg, n_4, n_6 } is { n_6, 2lg} and therefore EIR(L, {nw, nc, 1lg, 2lg}) = { n_6 }. Chapter 6 provides more examples.

2.10 SUMMARY

In this chapter the basic building blocks and definitions that are key to formal concept analysis and that will be used in the rest of this text have been defined. This includes the notion of a lattice, sublattice, formal concept lattice and EA-lattice. A number of different operations have also been defined on these structures, this include the intent- and extent representative operations.