Order-embedded Complete Lattices

Bernhard Ganter

TU Dresden, Germany

Abstract. We study complete lattices which are contained in other complete lattices as suborders, but not necessarily as subsemilattices. We develop a representation of such lattices by means of implications, and show how to navigate them using a modification of the standard Next closure algorithm. Our approach is inspired by early work of Shmuely [8] and Crapo [1].

Keywords. Complete lattice, implication, fixed point.

1 Introduction

The interest in *concept lattices* [5] has stimulated the creation of algorithms for generating lattices, and the availability of fast algorithms may conversely have contributed to the popularity of concept lattices. Moreover, concept lattices have easy representations either by a binary relation or by a set of implications, both of which can conveniently be used as input for the algorithms.

Although all complete lattices are isomorphic to concept lattices, they sometimes come in a form for which the above mentioned algorithms are not easy to apply. There are, for example, many families of sets which form complete lattices when ordered by the subset relation $\subseteq$, but are neither closure nor kernel systems. We provide an "implicational" representation for such lattices and modify one of the standard algorithms accordingly.

Throughout the paper, $(L, \leq)$ will be some abstract complete lattice. The reader may assume, without much loss of generality, that $(L, \leq)$ is a powerset lattice $(\mathfrak{P}(M), \subseteq)$. We use the abstract setting because it is more transparent.

2 Monotone Functions

Definition 1 Let $(L, \leq)$ be a complete lattice. A function $\varphi : L \to L$ is **monotone**[1] if $x \leq y$ always implies $\varphi(x) \leq \varphi(y)$. A monotone function is called

idempotent	if $\varphi(x) = \varphi(\varphi(x))$	for all $x \in L$,
extensive	if $x \leq \varphi(x)$	for all $x \in L$,
contractive[2]	if $x \geq \varphi(x)$	for all $x \in L$,
tensive	if $\varphi(x) = \varphi(x \wedge \varphi(x))$	for all $x \in L$,
increasing[3]	if $\varphi(x) \leq \varphi(\varphi(x))$	for all $x \in L$, and
decreasing	if $\varphi(x) \geq \varphi(\varphi(x))$	for all $x \in L$.

$\Diamond$

[1] Synonyms are **order-preserving** and **isotone**.

[2] A synonym is **intensive**.

[3] Following Shmuely [8]. Tarski [9] uses "increasing" in the sense of "order-preserving".

© Dmitry I. Ignatov, Lhouari Nourine (Eds.): CLA 2018, pp. 153–165,
ISBN 978–80–244–5328–6, Department of Computer Science, Palacký University Olomouc, 2018.

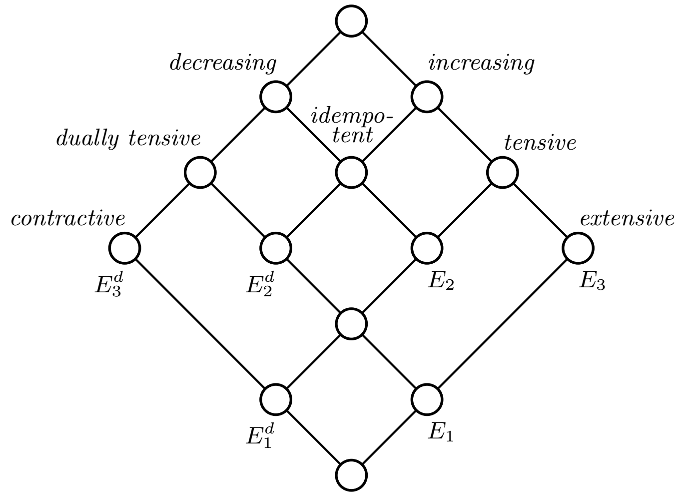

Fig. 1. The result of an attribute exploration [4] for monotone functions, see Proposition 1.

Proposition 1. *Figure 1 shows the logical hierarchy of the properties given in Definition 1. In particular, if $\varphi : L \to L$ is monotone, then the following statements hold (as well as their duals):*

1. *If φ is extensive, then φ is tensive.*
2. *If φ is tensive, then φ is increasing.*
3. *φ is idempotent iff φ is both increasing and decreasing.*
4. *If φ is idempotent and extensive, then φ is dually tensive.*

Moreover, there are examples of monotone functions falsifying other implications, as indicated in the diagram.

Proof. 1) If $x \leq \varphi(x)$, then $x \wedge \varphi(x) = x$ and thus $\varphi(x \wedge \varphi(x)) = \varphi(x)$. 2) From $x \wedge \varphi(x) \leq \varphi(x)$ we infer $\varphi(x) = \varphi(x \wedge \varphi(x)) \leq \varphi(\varphi(x))$. 3) is obvious. 4) If $x \leq \varphi(x)$ then $\varphi(x \vee \varphi(x)) = \varphi(\varphi(x)) = \varphi(x)$.

For the separating examples we use functions of the form $\varphi_{\mathcal{L}}$, to be defined in Proposition 6. $(L, \leq)$ is the powerset lattice of $\{a, b, c\}$ for E_1 and E_2 and of $\{a, b\}$ for E_3. E_1: $\mathcal{L} = \{\{a\} \to \{b, c\}, \{b\} \to \{c\}, \{c\} \to \{b\}\}$, E_2: $\mathcal{L} = \{\{a\} \to \{b\}, \{a, b, c\} \to \{a, b, c\}\}$ (see Example 1 below), E_3: $\mathcal{L} = \{\emptyset \to \{a\}, \{a\} \to \{b\}, \{b\} \to \{b\}\}$. E_1^d, E_2^d, E_3^d are dual to E_1, E_2, E_3. □

Definition 2 If $\varphi : L \to L$ is a mapping and $x \in L$, then we say that x is a **fixed** point of φ, iff $\varphi(x) = x$, and that x is a **closed** point of φ, iff $\varphi(x) \leq x$. ◊

Proposition 2. *Every fixed point is closed. If φ is monotone and increasing, and x is closed, then $\varphi(x)$ is fixed.*

Proof. The first statement is obvious. Suppose that x is closed, i.e., that $x \geq \varphi(x)$. Then $\varphi(x) \geq \varphi(\varphi(x)) \geq \varphi(x)$, since φ is monotone and increasing. We conclude that $\varphi(x) = \varphi(\varphi(x))$ and thus $\varphi(x)$ is fixed.

The proposition may suggest a pairing between fixed and closed elements. But note for example that when φ is the function which maps everything to the least element of $(L, \leq)$, then *every* element of $(L, \leq)$ is closed, but *only the least* element is fixed.

A function that is both idempotent and monotone is called a **closure operator** on $(L, \leq)$ if it is extensive, and is a **kernel operator** if contractive. The set of fixed points of a closure operator is called a **closure system**. It is well known that the closure systems are precisely the $\bigwedge$-subsemilattices. Each complete meet-subsemilattice of a complete lattice is itself a complete lattice, because the join operation can be expressed in terms of the meet operation: the join of a subset S equals the meet of all upper bounds of S. However, this join operation usually is not identical with the join in the original complete lattice. The meet-subsemilattice therefore is a complete lattice, but *not* a complete sublattice in general. In a closure system of sets, for example, the join of two elements is usually not given by their set union, but by the closure of this union. Thus a closure system, ordered by set inclusion, is a complete lattice, but not necessarily a sublattice.

The fixed points of a kernel operator are closed under arbitrary joins and thus form a $\bigvee$-subsemilattice, also called a **kernel system**. Again we get the second operation from the first, so that each kernel system also is a complete lattice.

This shows that closure systems are not the only subsets yielding order-embedded complete lattices. In fact, the following is well known[4]:

Lemma 1. *A subset of a complete lattice $(L, \leq)$, with the induced order, is a complete lattice if and only if it is the image of a monotone and idempotent function $\varphi : L \to L$.*

Proof. Suppose that $\mathcal{F} = \{\varphi(x) \mid x \in L\}$ for some monotone and idempotent function $\varphi : L \to L$. We claim that for any subfamily $\mathcal{S} \subseteq \mathcal{F}$ the element $\varphi(\bigwedge \mathcal{S})$ is the infimum of $\mathcal{S}$ in $\mathcal{F}$. Clearly $\bigwedge \mathcal{S} \leq s$ holds for every $s \in \mathcal{S}$. Since φ is monotone, we get that $\varphi(\bigwedge \mathcal{S}) \leq \varphi(s) = s$ for all $s \in \mathcal{S}$, which shows that $\varphi(\bigwedge \mathcal{S})$ is a lower bound of $\mathcal{S}$. But any lower bound b of $\mathcal{S}$ must satisfy $b \leq s$ for all $s \in \mathcal{S}$ and therefore $b \leq \bigwedge \mathcal{S}$. If $b \in \mathcal{F}$, then $b = \varphi(b) \leq \varphi(\bigwedge \mathcal{S})$, as desired.

For the converse suppose that $\mathcal{F} \subseteq L$ is a complete lattice and define a function $\varphi : L \to L$ by $\varphi(x) := \sup_{\mathcal{F}}\{f \in \mathcal{F} \mid f \leq x\}$ (where $\sup_{\mathcal{F}}$ denotes the supremum in $\mathcal{F}$). This function is clearly idempotent and monotone, and its image is $\mathcal{F}$. □

Lemma 1 adds a kind of converse to the celebrated Knaster-Tarski theorem [6,9], which states that the set of fixed points of any monotone function on a complete lattice is itself a complete lattice:

[4] Crapo [1] cites Duffus and Rival [2], while Shmuely [8] cites older notes by Crapo.

Corollary 1. *A subset $\mathcal{F} \subseteq L$ of a complete lattice $(L, \leq)$, with the induced order, is a complete lattice if and only if $\mathcal{F}$ is the set of fixed points of some monotone function.*

Two details from the proof of Lemma 1 will be used later. We list them as separate propositions:

Proposition 3. *Let φ be an idempotent and monotone function on a complete lattice $(L, \leq)$, and let $\bigvee$, $\bigwedge$ denote the supremum and infimum operation of $(L, \leq)$, respectively.*

In the complete lattice of fixed points of φ, the supremum and infimum of a set $\mathcal{S}$ are given by

$$\varphi(\bigvee \mathcal{S}) \qquad \textit{and} \qquad \varphi(\bigwedge \mathcal{S}).$$

The second part of the proof of Lemma 1 is stronger than necessary: the function which was used is not only monotone and idempotent, but has an additional property:

Proposition 4. *The function which was used in the proof of Lemma 1,*

$$x \mapsto \varphi(x) := \sup_{\mathcal{F}}\{f \in \mathcal{F} \mid f \leq x\},$$

is tensive.

Proof. If $f \leq x$ and $f \in \mathcal{F}$, then $f \leq \varphi(x)$ and so $f \leq x \wedge \varphi(x)$. Thus

$$\{f \in \mathcal{F} \mid f \leq x\} \subseteq \{f \in \mathcal{F} \mid f \leq x \wedge \varphi(x)\},$$

which implies that $\varphi(x) \leq \varphi(x \wedge \varphi(x))$. Since φ is monotone, we conclude equality. □

A simple consequence of the Knaster-Tarski result which we will use is

Proposition 5. *If $(L, \leq)$ is a complete lattice, $\varphi : L \to L$ is monotone, and $x \in L$ is an element for which $x \leq \varphi(x)$, then there is a least fixed point of φ that is greater or equal to x.*

Proof. Note that since φ is monotone, the set $\uparrow x := \{y \in L \mid x \leq y\}$ is mapped into itself by φ: when $y \geq x$, then $\varphi(y) \geq \varphi(x) \geq x$. But $\uparrow x$ is a complete lattice as well, to which the Knaster-Tarski result can be applied. So there is a least fixed point of φ in $\uparrow x$. □

Lemma 2. *If $\varphi : L \to L$ is monotone and increasing, then for each $x \in L$ there is a least closed element $\overline{\varphi}(x) \geq x$, and there is a least fixed element $\widehat{\varphi}(x) \geq \varphi(x)$. If φ is tensive, then so is $\widehat{\varphi}$.*

Proof. For the first claim define a function $\rho(x) := x \vee \varphi(x)$. Note that the fixed points of ρ are precisely the closed points of φ. Clearly ρ is monotone and extensive, so by Proposition 5 there is a least fixed point y of ρ which is greater or equal to x.

The second claim follows again from Proposition 5, assuming that the function φ is increasing.

Finally, assume that φ is tensive. By definition, $\widehat{\varphi}(x \wedge \varphi(x))$ is the least fixed point of φ greater or equal to $\varphi(x \wedge \varphi(x))$. But when φ is tensive, the latter equals $\varphi(x)$, and therefore $\widehat{\varphi}(x \wedge \varphi(x)) = \widehat{\varphi}(x)$. But since $\varphi(x) \leq \widehat{\varphi}(x)$, we get $\widehat{\varphi}(x) = \widehat{\varphi}(x \wedge \varphi(x)) \leq \widehat{\varphi}(x \wedge \widehat{\varphi}(x)) \leq \widehat{\varphi}(x)$, which concludes the proof. □

Note that the function $\overline{\varphi}$, defined in Lemma 2, is a closure operator, and that $\widehat{\varphi}$ has the same fixed points as φ.

Lemma 3. *If φ is monotone and increasing, then for all $x \in L$*

$$\widehat{\varphi}(x) = \overline{\varphi}(\varphi(x)).$$

Proof. $\widehat{\varphi}(x)$ is fixed and therefore closed, and contains $\varphi(x)$, thus $\widehat{\varphi}(x) \geq \overline{\varphi}(\varphi(x))$. It remains to show that $\widehat{\varphi}(x) \leq \overline{\varphi}(\varphi(x))$. Proposition 2 yields that $\varphi(\overline{\varphi}(\widehat{\varphi}(x)))$ is fixed and less or equal to $\overline{\varphi}(\varphi(x))$. The proof is complete if we show that this fixed element contains $\varphi(x)$, because that forces it to be equal to $\widehat{\varphi}(x)$ (which is the *least* such fixed point). But from $\varphi(x) \leq \overline{\varphi}(\varphi(x))$ and the fact that φ is increasing and monotone we conclude that $\varphi(x) \leq \varphi(\varphi(x)) \leq \varphi(\overline{\varphi}(\varphi(x)))$. □

3 Implications

There is a simple way of constructing such monotone and increasing functions without reference to an embedded lattice. It relies on *implications*. An **implication** over L is just an ordered pair[5] of elements $x, y \in L$, denoted $x \to y$. We say that a lattice element z **respects** an implication $x \to y$ if $x \not\leq z$ or $y \leq z$.

Proposition 6. *Let $\mathcal{L}$ be a set of implications over L. The function $\varphi_{\mathcal{L}} : L \to L$, defined as*

$$\varphi_{\mathcal{L}}(x) := \bigvee\{a \vee b \mid a \leq x, a \to b \in \mathcal{L}\},$$

is monotone and tensive. Conversely, if $\varphi : L \to L$ is monotone and tensive, then $\varphi = \varphi_{\mathcal{L}}$ for

$$\mathcal{L} = \{x \wedge \varphi(x) \to \varphi(x) \mid x \in L\}.$$

Proof. When $x \leq y$, then $\{a \to b \in \mathcal{L} \mid a \leq x\} \subseteq \{a \to b \in \mathcal{L} \mid a \leq y\}$, and thus $\varphi_{\mathcal{L}}(x) \leq \varphi_{\mathcal{L}}(y)$. So $\varphi_{\mathcal{L}}$ is monotone. For the other claim, note that if $a \to b \in \mathcal{L}$ and $a \leq x$, then $a \leq \varphi_{\mathcal{L}}(x)$ and thus $a \leq x \wedge \varphi_{\mathcal{L}}(x)$. It follows that $\varphi_{\mathcal{L}}(x \wedge \varphi_{\mathcal{L}}(x)) \geq \varphi_{\mathcal{L}}(x)$. Monotonicity of $\varphi_{\mathcal{L}}$ yields equality and concludes the proof that $\varphi_{\mathcal{L}}$ is tensive.

For the converse we claim that we have $\varphi_{\mathcal{L}}(y) = \varphi(y)$ for all $y \in L$. Since $y \wedge \varphi(y) \leq y$ and $y \wedge \varphi(y) \to \varphi(y) \in \mathcal{L}$, we get that $\varphi_{\mathcal{L}}(y) \geq \varphi(y)$. If $x \wedge \varphi(x) \leq y$ holds for some x, then $\varphi(x) = \varphi(x \wedge \varphi(x)) \leq \varphi(y)$ (since φ is monotone and tensive), and therefore $\varphi_{\mathcal{L}}(y) \leq \varphi(y)$. This proves $\varphi_{\mathcal{L}} = \varphi$. □

[5] The notion abstracts that of an *attribute implication* in Formal Concept Analysis. Note that in our approach implication sets are *not* assumed to be closed under the Armstrong rules.

Example 1. The separating example E_2 of Figure 1 was defined in the proof of Proposition 1 as the monotone function $\varphi_{\mathcal{L}}$ on the power set of $\{a, b, c\}$ given by the implication set

$$\mathcal{L} := \{\{a\} \to \{b\}, \{a, b, c\} \to \{a, b, c\}\}.$$

According to the definition in Proposition 6, this function has the following values:

x	$\emptyset$	$\{a\}$	$\{b\}$	$\{c\}$	$\{a,b\}$	$\{a,c\}$	$\{b,c\}$	$\{a,b,c\}$
$\varphi_{\mathcal{L}}(x)$	$\emptyset$	$\{a,b\}$	$\emptyset$	$\emptyset$	$\{a,b\}$	$\{a,b\}$	$\emptyset$	$\{a,b,c\}$

It is easy to check that the function is monotone, idempotent, and tensive. But it is not dually tensive, since $\varphi_{\mathcal{L}}(\{a, c\}) = \{a, b\} \neq \varphi_{\mathcal{L}}(\{a, c\} \cup \varphi_{\mathcal{L}}(\{a, c\})) = \varphi_{\mathcal{L}}(\{a, b, c\}) = \{a, b, c\}$.

Following Definition 2 we call an element $x \in L$ **fixed** under a set $\mathcal{L}$ of implications, if $x = \varphi_{\mathcal{L}}(x)$, and **closed**, if $x \geq \varphi_{\mathcal{L}}(x)$. It is easy to see that the latter is equivalent to the standard definition in Formal Concept Analysis, where an element is called *closed* under $\mathcal{L}$ when it respects all implications in $\mathcal{L}$. The corresponding closure operator is often denoted $x \mapsto \mathcal{L}(x)$. Here we write $\overline{\varphi}_{\mathcal{L}}$, as it is suggested by Lemma 2. The following is a corollary to that lemma.

Corollary 2. *Let $\mathcal{L}$ be a set of implications over L. Then the function $\widehat{\varphi}_{\mathcal{L}} : L \to L$, defined by*

$$\widehat{\varphi}_{\mathcal{L}}(x) \textit{ is the least fixed point of } \varphi_{\mathcal{L}} \textit{ greater or equal to } \varphi_{\mathcal{L}}(x),$$

is idempotent, monotone, and tensive. Moreover,

$$\widehat{\varphi}_{\mathcal{L}}(x) = \overline{\varphi}_{\mathcal{L}}(\varphi_{\mathcal{L}}(x)) \textit{ for all } x \in L.$$

A very welcome consequence of this corollary is that $\widehat{\varphi}_{\mathcal{L}}$ can efficiently be computed, for example by the LinClosure algorithm (see Algorithm 15 in [4]).

It is actually possible to give a (kind of) explicit representation of $\widehat{\varphi}_{\mathcal{L}}$ in terms of $\varphi_{\mathcal{L}}$: If $(L, \leq)$ is finite, then

$$\widehat{\varphi}_{\mathcal{L}}(x) := \varphi_{\mathcal{L}}(x) \vee \varphi_{\mathcal{L}}(\varphi_{\mathcal{L}}(x)) \vee \varphi_{\mathcal{L}}(\varphi_{\mathcal{L}}(\varphi_{\mathcal{L}}(x))) \vee \ldots . \quad (*)$$

Without the finiteness condition it may be necessary to apply $\varphi_{\mathcal{L}}$ "transfinitely often", as the following example shows.

Example 2. Let $L := \mathbb{N} \cup \{\infty_1, \infty_2\}$, where the integers are in the natural order, ∞_1 is greater than all integers and $\infty_1 < \infty_2$. Moreover, let

$$\mathcal{L} := \{0 \to 1,\ 1 \to 2,\ 2 \to 3, \ldots\} \cup \{\infty_1 \to \infty_2\}.$$

For $x := 0$ we get $\varphi_{\mathcal{L}}(x) = 1$, $\varphi_{\mathcal{L}}(\varphi_{\mathcal{L}}(x)) = 2$, and so on. Applying formula $(*)$ yields $\widehat{\varphi}_{\mathcal{L}}(0) = 1 \vee 2 \vee 3 \vee \ldots = \infty_1$. But this is no fixed point, since $\varphi_{\mathcal{L}}(\infty_1) = \infty_2$.

Example 3. Let $M := \{a, b, c, d\}$, $(L, \leq) := (\mathfrak{P}(M), \subseteq)$, and

$$\mathcal{L} := \{\{a\} \to \{a\}, \{b\} \to \{b\}, \{b, c, d\} \to \{b, c, d\}, \{a, b\} \to \{c\}\}.$$

Then, for example, $\widehat{\varphi}_{\mathcal{L}}(\{a, c, d\}) = \{a\}$ and $\widehat{\varphi}_{\mathcal{L}}(\{a, b, d\}) = \{a, b, c\}$. $\widehat{\varphi}_{\mathcal{L}}$ has six fixed points; $\emptyset, \{a\}, \{b\}, \{a, b, c\}, \{b, c, d\}$, and $\{a, b, c, d\}$. These six sets, ordered by $\subseteq$, form a complete lattice which is neither a $\bigcap$- nor a $\bigcup$-subsemilattice of the powerset lattice.

We claim that our construction based on implications is universal in the sense, that every embedded complete lattice is obtained. This is shown in the theorem below.

Theorem 1. *For every monotone function $\varphi : L \to L$ on a complete lattice $(L, \leq)$ there is a set $\mathcal{L}$ of implications over L such that φ and $\widehat{\varphi}_{\mathcal{L}}$ have the same fixed points.*

Proof. The set $\mathcal{F}$ of fixed points of any monotone function is, according to Knaster and Tarski (see Corollary 1), a complete lattice. For each such complete lattice $(\mathcal{F}, \leq)$ we therefore need to find a suitable set of implications. Let $\sup_L$ denote the supremum in $(L, \leq)$ and $\sup_{\mathcal{F}}$ denote the supremum in $(\mathcal{F}, \leq)$. We choose

$$\mathcal{L} := \{\sup_L \mathcal{S} \to \sup_{\mathcal{F}} \mathcal{S} \mid \mathcal{S} \subseteq \mathcal{F}\}$$

and prove that the fixed points of $\widehat{\varphi}_{\mathcal{L}}$ are precisely the elements of $\mathcal{F}$:

First suppose that $e \in \mathcal{F}$. The $\varphi_{\mathcal{L}}(e) = e$ because, first of all, $e \to e \in \mathcal{L}$, and secondly e respects all implications in $\mathcal{L}$: if $\sup_L \mathcal{S} \leq e$ for some set $\mathcal{S} \subseteq \mathcal{F}$, then $e \geq s$ for all $s \in \mathcal{S}$ and, since $(\mathcal{F}, \leq)$ is a complete lattice, $e \geq \sup_{\mathcal{F}} \mathcal{S}$. Conversely, if $e = \widehat{\varphi}_{\mathcal{L}}(e)$ is a fixed point of $\widehat{\varphi}_{\mathcal{L}}$, then let $\mathcal{S} := \{f \in \mathcal{F} \mid f \leq e\}$ be the set of all $\mathcal{F}$-elements below e. Since e respects the implication $\sup_L \mathcal{S} \to \sup_{\mathcal{F}} \mathcal{S}$, we get $\sup_{\mathcal{F}} \mathcal{S} \leq e$. But whenever the premise of an implication in $\mathcal{L}$ is below e, it must be below $\sup_L \mathcal{S}$. Therefore $e = \varphi_{\mathcal{L}}(e) = \sup_{\mathcal{F}} \mathcal{S}$ and thus $e \in \mathcal{F}$. □

As an immediate consequence of Theorem 1 we get

Corollary 3. *The subsets of a complete lattice which are, with the induced order, complete lattices themselves, are precisely the sets of fixed elements under some set of implications.*

$\widehat{\varphi}_{\mathcal{L}}$ is usually not extensive, while the closure operator $\overline{\varphi}_{\mathcal{L}}$ is. That condition can easily be achieved by including $\{x \to x \mid x \in L\}$ into the list of implications (it actually suffices to do this for a join-dense set of elements), so all closure operators are of the form $\widehat{\varphi}_{\mathcal{L}}$ for a suitable set $\mathcal{L}$.

But *kernel operators* can be represented as well: If $\mathcal{F}$ is a kernel system, then $\widehat{\varphi}_{\mathcal{L}}$ is the corresponding kernel operator when $\mathcal{L} := \{f \to f \mid f \in \mathcal{F}\}$. More generally, if $\mathcal{F}$ is an arbitrary family of sets then the so defined function $\widehat{\varphi}_{\mathcal{L}}$ is the kernel operator for the kernel system generated by $\mathcal{F}$.

Is it possible to find, for a given function, a suitable implication set without reference to the embedded lattice of fixed points? The next proposition gives an answer. However, we shall learn from Example 4 that this is not always practical.

Proposition 7. *If $\varphi : L \to L$ is monotone, idempotent, and tensive, then the set*

$$\mathcal{L} := \{x \wedge \varphi(x) \to \varphi(x) \mid x \in L\}$$

is such that $\widehat{\varphi}_{\mathcal{L}} = \varphi$.

Proof. From Proposition 6 we get that $\varphi = \varphi_{\mathcal{L}}$. But when φ is idempotent, then $\varphi(x) \to \varphi(x) \in \mathcal{L}$, which makes $\varphi(x)$ a fixed point of $\varphi_{\mathcal{L}}$, and we conclude $\varphi = \varphi_{\mathcal{L}} = \widehat{\varphi}_{\mathcal{L}}$. □

To summarize: Every embedded complete lattice is the image of some function which is monotone, idempotent and tensive (Proposition 4). These are precisely the functions which can be described by means of implications as in Corollary 2. Implications can easily be found for any given such function (Proposition 7).

4 The Next Fixed Point Algorithm

Many years ago the author suggested a simple algorithm [3] for finding all closed sets of a given closure operator φ on a (finite, linearly ordered) set G. One starts with the closure $A := \varphi(\emptyset)$ of the empty set and then repeats the procedure shown in Figure 2, using the output of each application as the input of the next one, until it returns $\perp$. The algorithm is extremely useful for browsing and

```
for all g ∈ G in reverse order do
        if g ∈ A then A := A \ {g}
        else
              B := φ(A ∪ {g})
              if g is the smallest element of B \ A then return B
return ⊥.
```

Fig. 2. The NEXT CLOSURE algorithm, from [4]

navigating in closure systems. And since it is so simple, many variations and generalizations have been invented, see [4].

It is easy to generalize the algorithm to closure operators on complete lattices, not only powerset lattices. It therefore seems natural to ask if a modification of NEXT CLOSURE can be used for generating all images of any given idempotent, monotone, and tensive function. Unfortunately, the answer is "no", unless some additional information is provided. Our pessimism is prompted by the following example:

Example 4. Let $A \subseteq L$ be an antichain in a complete lattice $(L, \leq)$, let 0_L and 1_L be the least and the greatest element of $(L, \leq)$, and let f be an element of

A. The function

$$\varphi(x) := \begin{cases} f & \text{if } x = f \\ 1_L & \text{if } a < x \text{ for some } a \in A \\ 0_L & \text{else} \end{cases}$$

is idempotent, monotone, and tensive.

In this example it is tedious to determine the fixed points by repeated invocation of φ. Since the number of antichains may be exponential[6] in the size of L, but nevertheless may be large on average, it seems difficult to find an algorithm which determines the fixed point f reasonably fast. Stronger assumptions are needed.

Proposition 8. *Let $(L, \leq)$ be a complete lattice, let $\varphi : L \to L$ be a monotone and idempotent function and let $G \subseteq L$ be a finite set that is join-dense in the complete lattice formed by the images of φ. Endow G with an arbitrary linear order.*

Compute a sequence of sets, starting with $A := \{g \in G \mid g \leq \varphi(\emptyset)\}$, and then repeatedly invoking the algorithm in Figure 3, always using the previous output as the next input, until $\bot$ is reached. For each set B in this sequence, $\varphi(\bigvee B)$ is a fixed point of φ, and all fixed points occur exactly once.

```
for all g ∈ G in reverse order do
        if g ∈ A then A := A \ {g}
        else
              B := {h ∈ G | h ≤ φ(⋁(A ∪ {g}))}
              if g is the smallest element of B \ A then return B
return ⊥.
```

Fig. 3. The NEXT FIXED POINT algorithm

Proof. Each fixed point f of φ is uniquely determined by its projection

$$\Pi(f) := \{g \in G \mid g \leq f\}$$

to the join-dense set G, because it can be obtained as the join of these elements: $f = \varphi(\bigvee(\Pi(f)))$ (recall from Proposition 3 that $S \mapsto \varphi(\bigvee S)$ is the join operation in the fixed point lattice). These projection sets form a closure system on G, for which $F \mapsto \Pi(\varphi(\bigvee F))$ is the closure operator. □

[6] For example, the Dedekind numbers in case that L is a powerset lattice.

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

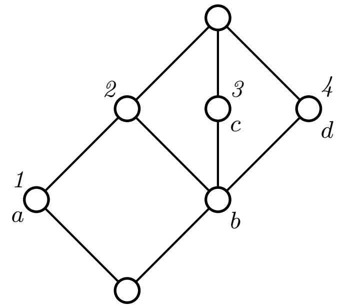

Fig. 4. A formal context and its concept lattice. Of its 66 embedded complete lattices, 20 are complete sublattices.

Example 5. We illustrate our findings by calculating the closed relations of the formal context in Figure 4. Closed relations are subrelations with the property that every formal concept of the subrelation is a formal concept of the original formal context [5]. They are in 1-1-correspondence with the complete sublattices of the concept lattice. The formal context has 20 closed relations, 18 of which are shown in Figure 5. The two missing ones are the empty relation and the full incidence of the formal context itself. Note that these relations (including

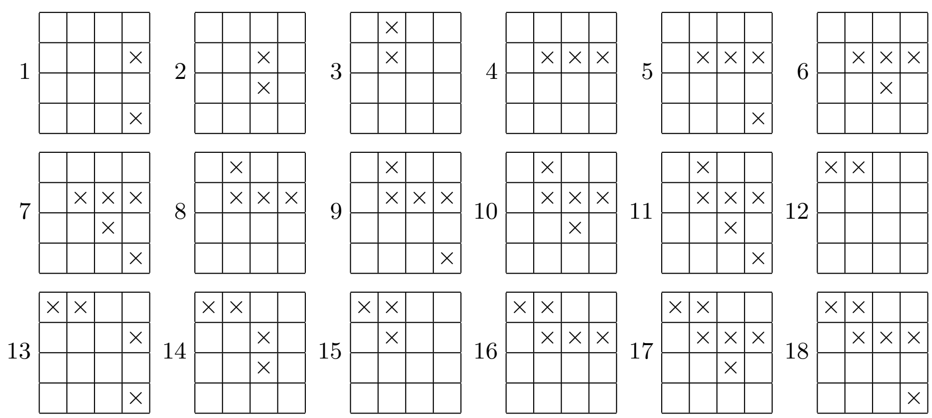

Fig. 5. The non-trivial closed relations of the formal context in Figure 4

the trivial ones) are not closed under intersection nor under union. The union of relations R_1 and R_2 is not closed, nor is the intersection of R_3 and R_4 closed. However, when ordered by set inclusion $\subseteq$, these 20 relations form a complete lattice which is isomorphic to the lattice of all complete sublattices. This lattice of closed relations is contained in the lattice $(\mathfrak{P}(\{a, b, c, d\} \times \{1, 2, 3, 4\}), \subseteq)$ of all relations between these two sets as a suborder, but not as a sublattice.

The closed relations R_1, R_2, R_3, R_4, and R_{12} are of the form $A \times B$ for some nontrivial formal concept (A, B) and represent the complete sublattice with exactly one nontrivial element. These are the join-irreducible closed relations. Together, they form a join-dense set. For reasons that become clear later we reverse the order and work with

$$G := \{R_{12} < R_4 < R_3 < R_2 < R_1\}.$$

The function φ will be given by eight implications, five of which are of the form $X \to X$. Three more are derived from the condition that a sublattice must be closed under join and meet. So $\varphi := \widehat{\varphi}_{\mathcal{L}}$ for

$$\begin{aligned}\mathcal{L} := &\{R_1 \to R_1, R_2 \to R_2, R_3 \to R_3, R_4 \to R_4, R_{12} \to R_{12}\} \\ \cup\ &\{R_1 \cup R_2 \to R_4, R_1 \cup R_3 \to R_4, R_2 \cup R_3 \to R_4\}.\end{aligned}$$

If Algorithm 3 is used for this operator φ and is started with the empty relation, it produces all closed relations in the order of Figure 5, terminating with the full incidence relation of the context in Figure 4.

We give one intermediate step of the algorithm in detail, namely the step from R_2 to R_3:

R_2 contains none of the other relations in G, so the Next fixed point algorithm is invoked with $A := \{R_2\}$. The largest element of G is R_1, which is not in A, so $B := \{h \in G \mid h \leq \varphi(\bigvee(A \cup \{R_1\}))\}$ must be computed. $A \cup \{R_1\} = \{R_1, R_2\}$, and the join $\bigvee$ is the union $\bigcup$ of relations. We obtain $\bigvee(A \cup \{R_1\}) = R_1 \cup R_2$, which is not a closed relation. But $\mathcal{L}$ contains three implications the premise of which is contained in $R_1 \cup R_2$, and we find that $\varphi_{\mathcal{L}}(R_1 \cup R_2) = \varphi(R_1 \cup R_2) = R_1 \cup R_2 \cup R_4 = R_7$ and, since R_7 contains no further elements of G, $B = \{R_1, R_2, R_4\}$. However, R_1 is *not* the smallest element of $B \setminus A$ (the smallest element is R_4), so this iteration step does not return a result. The next iteration has $A = \{R_1\}$ and $g = R_1$, so R_1 is simply removed from A. Then $A = \emptyset$ and $g = R_3$ result in $B = \{R_3\}$, which is returned as the next closed relation.

For this particular example, the set of all 20 sublattices of the lattice in Figure 5 is easily determined by hand. In general, a concept lattice can be much larger than its formal context. Working with the formal context then may be more efficient.

How to find a join-dense set, as it is required in Proposition 8 ? There is an easy answer when the function φ is given by implications.

Proposition 9. *Let $\varphi := \widehat{\varphi}_{\mathcal{L}}$ for some set $\mathcal{L}$ of implications. Then the set*

$$\{\varphi(a) \mid a \to b \in \mathcal{L}\}$$

is join-dense in the lattice of fixed points of φ.

Proof. Any fixed point of $\widehat{\varphi}_{\mathcal{L}}$ by definition also is a fixed point of $\varphi_{\mathcal{L}}$. So if $\widehat{\varphi}_{\mathcal{L}}(f) = f$ then

$$f = \varphi_{\mathcal{L}}(f) = \bigvee\{a \vee b \mid a \leq f, a \to b \in \mathcal{L}\}.$$

But since $a \vee b \leq \varphi(a)$ whenever $a \to b \in \mathcal{L}$, we get that $f = \bigvee\{\varphi(a) \mid a \to b \in \mathcal{L}, a \leq f\}$, which proves the claim. □

Example 6. The set $\mathcal{L}$ in Example 3 consists of four implications, and we get

$$\{\{a\}, \{b\}, \{b, c, d\}, \{a, b, c\}\} = \{\widehat{\varphi}_{\mathcal{L}}(\{a\}), \widehat{\varphi}_{\mathcal{L}}(\{b\}), \widehat{\varphi}_{\mathcal{L}}(\{b, c, d\}), \widehat{\varphi}_{\mathcal{L}}(\{a, b\})\}$$

as a join-dense set according to Proposition 9. However, $\{a, b, c\}$ is not join-irreducible, because the supremum of $\{a\}$ and $\{b\}$ is, using Proposition 3,

$$\widehat{\varphi}_{\mathcal{L}}(\{a\} \vee \{b\}) = \widehat{\varphi}_{\mathcal{L}}(\{a\} \cup \{b\}) = \widehat{\varphi}_{\mathcal{L}}(\{a, b\}) = \{a, b, c\}.$$

5 Discussion

Apart from closure and kernel systems, there are many "lattices of sets", i.e., families of sets which form complete lattices, when ordered by set inclusion. More generally we have studied subsets of arbitrary complete lattices which, endowed with the induced order, are complete lattices themselves. We have shown that each such complete lattice can be described by a set of implications, in a way which is very similar to the standard one in Formal Concept Analysis. The NEXT CLOSURE algorithm can be tweaked to work with this representation, so that we were able to give an algorithm for generating such lattices.

The reader may wonder why we did not use the even more general operator

$$x \mapsto \bigvee\{b \mid a \leq x, a \to b \in \mathcal{L}\}, \qquad x \in L,$$

which also is monotone. But such operators are no longer tensive in general, not even increasing. Actually, it is easy to see that *every* monotone function φ can so be represented (choose $\mathcal{L} := \{a \to \varphi(a) \mid a \in L\}$). Such operators are more difficult to handle, and we see no possibility of using LINCLOSURE here. But the fixed point sets of such functions describe the same as we have treated with tensive functions: all embedded complete lattices.

Much more important is the question if embedded complete lattices have a natural and useful interpretation. The work of Shmuely [8] gives interesting hints. Her *$u-v$-connections* generalize Galois connections and are closely related to what we construct. One might hope that these can be derived from formal contexts with additional, meaningful structure.

Acknowledgements

The author thanks Bogdan Chornomaz for his helpful remarks, and the reviewers for their careful proofreading.

References

1. Crapo, H.: Ordered sets: retracts and connections. Journal of Pure and Applied Algebra **23**(1), 13–28 (1982)
2. Duffus, D., Rival, I.: Retracts of partially ordered sets. Journal of the Australian Mathematical Society **27**(4), 495–506 (1979)
3. Ganter, B.: Two basic algorithms in concept analysis. FB4–Preprint 831, TH Darmstadt (1984), reprinted in [7]
4. Ganter, B., Obiedkov, S.: Conceptual exploration. Springer (2016)
5. Ganter, B., Wille, R.: Formal Concept Analysis: Mathematical Foundations. Springer (1998)
6. Knaster, B.: Un théorème sur les fonctions d'ensembles. Ann. Soc. Polon. Math. **6**, 133–134 (1928)
7. Kwuida, L., Sertkaya, B. (eds.): Formal Concept Analysis. Proceedings ICFCA 2010, Lecture Notes in Computer Science, vol. 5986. Springer (2010)
8. Shmuely, Z.: Increasing and decreasing operators on complete lattices. Journal of Combinatorial Theory, Series A **21**(3), 369–383 (1976)
9. Tarski, A.: A lattice-theoretical fixpoint theorem and its applications. Pacific J. Math. **5**(2), 285–309 (1955)