Admissibility in Finitely Generated Quasivarieties

Inauguraldissertation
der Philosophisch-naturwissenschaftlichen Fakultät
der Universität Bern

vorgelegt von

Christoph Röthlisberger
von Langnau im Emmental BE

Leiter der Arbeit:

Prof. Dr. George Metcalfe
Mathematisches Institut der Universität Bern

Admissibility in Finitely Generated Quasivarieties

Inauguraldissertation
der Philosophisch-naturwissenschaftlichen Fakultät
der Universität Bern

vorgelegt von

Christoph Röthlisberger

von Langnau im Emmental BE

Leiter der Arbeit:

Prof. Dr. George Metcalfe
Mathematisches Institut der Universität Bern

Von der Philosophisch-naturwissenschaftlichen Fakultät angenommen.

Der Dekan:

Bern, 10. Dezember 2013

Prof. Dr. Silvio Decurtins

Acknowledgement

First and foremost, I would like to express my gratitude to my advisor, George Metcalfe for his great support. Thank you for your belief in my abilities to finish the thesis, for your helpful guidance and inspiring advice – I could not imagine a better supervisor than you.

Furthermore, I would like to thank the University of Bern for giving me the possibility to complete my thesis. I acknowledge financial support from Swiss National Science Foundation Grant 20002_129507.

Special thanks also go to the following people: Markus Sprenger for providing me with the source code of the Algebra Workbench, Leonardo Cabrer for all the instructive discussions and his friendship, Jürg Schmid for motivating me to do a PhD, Lukas Gerber for answering my questions when starting at the University of Bern, Stefan Bachmann for explaining Delphi programming to me, Rosalie Iemhoff for taking the time to read and review my work.

Finally, I would like to express my love and gratitude to my beloved wife Rosana and my boys Julian, Noah and Ben, for their understanding and ongoing support during my studies.

Overall I praise God, the Almighty, for loving me.

Contents

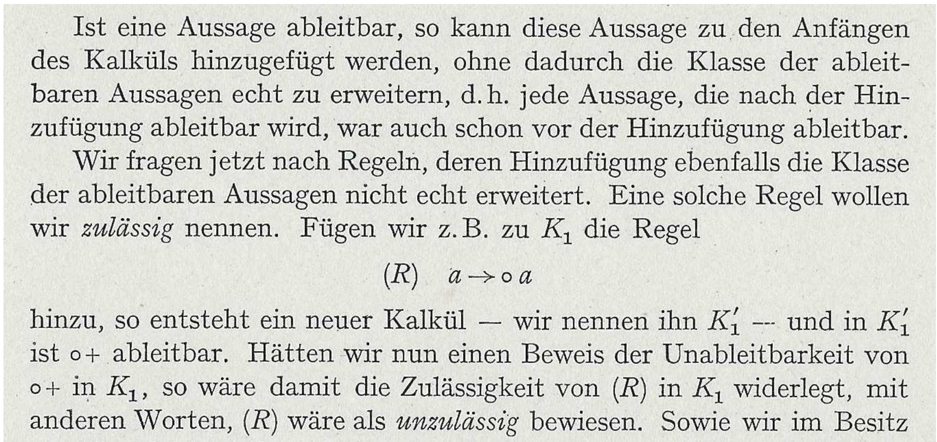
Chapter 1

Introduction

A rule $\varphi_1, \ldots, \varphi_n \,/\, \varphi$, understood as "if φ_1 and ... and φ_n, then φ", of a logic L is said to be *admissible* in L if it can be added to the logic without producing any new theorems (in particular, every derivable rule is admissible). Intuitively, adding an admissible rule to a logical system may change the internal structure of the system, but does not affect its output. Equivalently, the rule $\varphi_1, \ldots, \varphi_n \,/\, \varphi$ is *admissible* in the logic L, if for any substitution σ, whenever $\sigma(\varphi_1), \ldots, \sigma(\varphi_n)$ are theorems of L, also $\sigma(\varphi)$ is a theorem of L.

Admissibility often plays a substantial role in proving properties of logical systems. For example, establishing the completeness of a proof system usually involves showing that a certain rule is admissible. In particular, cut-elimination proofs verify that the cut-rule is admissible with respect to the given proof system without cut, leading in some cases to decidability, complexity and interpolation results (see, e.g., [74, 4]). Moreover, admissible rules like the cut rule can be used to shorten proofs, and other admissible rules might be used to improve proof search. Admissibility is also closely related to the topic of unification (see, e.g., [99, 42, 43, 44, 46, 1]); in particular, a formula φ is unifiable in a logic L if and only if $\varphi \,/\, \bot$ is not admissible in L (assuming that $\bot$ is in the language).

The notion of admissibility was defined by Lorenzen in the 1950s [69, page 19] (see Figure 1.1). However, particular admissible rules were already stud-



Ist eine Aussage ableitbar, so kann diese Aussage zu den Anfängen des Kalküls hinzugefügt werden, ohne dadurch die Klasse der ableitbaren Aussagen echt zu erweitern, d. h. jede Aussage, die nach der Hinzufügung ableitbar wird, war auch schon vor der Hinzufügung ableitbar.

Wir fragen jetzt nach Regeln, deren Hinzufügung ebenfalls die Klasse der ableitbaren Aussagen nicht echt erweitert. Eine solche Regel wollen wir *zulässig* nennen. Fügen wir z. B. zu K_1 die Regel

$$(R) \quad a \to \circ a$$

hinzu, so entsteht ein neuer Kalkül — wir nennen ihn K_1' — und in K_1' ist $\circ +$ ableitbar. Hätten wir nun einen Beweis der Unableitbarkeit von $\circ +$ in K_1, so wäre damit die Zulässigkeit von (R) in K_1 widerlegt, mit anderen Worten, (R) wäre als *unzulässig* bewiesen. Sowie wir im Besitz

Figure 1.1: Excerpt from *Einführung in die operative Logik und Mathematik* where admissible (german: *zulässig*) rules are defined, Lorenzen 1955.

ied by Gentzen [41, e.g.,page 13] and Johansson [63, page 128] in the context of sequent calculi and minimal logic, respectively, twenty years earlier.

All admissible rules of classical propositional logic CPC are derivable, i.e., CPC is *structurally complete* (see [83]). For many other logics this is not the case. The most famous example is intuitionistic propositional logic IPC where, e.g., the Kreisel-Putnam rule is admissible, but not derivable:

$$\neg\varphi \to (\psi \vee \chi) \ \ / \ \ (\neg\varphi \to \psi) \vee (\neg\varphi \to \chi).$$

The notion of structural completeness was introduced by Pogorzelski [83] and has been studied for many-valued logics (in particular, Gödel and Łukasiewicz logics) [36, 107, 108, 109, 30], modal and intermediate logics [85, 32, 99] and substructural logics [82]. Also algebraic characterizations have been given for structural completeness (see, e.g., [86, 12, 9]).

In 1975, Friedman posed the question as to whether "There is a decision procedure for determining whether a figure A / B represents a valid rule of inference in the intuitionistic propositional calculus (where A, B are formulae in the propositional calculus)." ([39, Question 40]), i.e., whether there is a *decision procedure* for admissible rules in IPC. Rybakov answered this question positively not only for IPC, but also for the modal logic S4 in [96, 98].

Concrete proof systems for checking admissibility in modal and intermediate logics have also been provided (see, e.g., [45, 59, 58, 6]).

Another question of interest is to find (possibly small) sets of rules characterizing the admissible rules of a given logic which is not structurally complete. More formally, a set of admissible rules of a given logic L is called a *basis for the admissible rules of* L, if every admissible rule of L is derivable from this set in L. Rybakov showed in particular that there is no finite basis for the admissible rules of IPC [97]. Iemhoff [56] and Rozière [95] subsequently proved, independently, that an elegant infinite set of rules conjectured by De Jongh and Visser provides a basis. Bases for admissible rules have also been found for other logics, in particular modal logics [100, 60, 5], intermediate logics [57, 31], Łukasiewicz logics [61, 62] and other fuzzy logics [30], fragments of the substructural logic R-Mingle [73] and classes of De Morgan algebras [26].

The focus of this work is on admissibility in *finite-valued* logics. At the beginning of the twentieth century, Łukasiewicz introduced the three-valued logic $Ł_3$ to handle future contingents such as "tomorrow it will rain" [70]. This and further investigations of finite and infinite valued Łukasiewicz logics [71, 72] together with the work of Post [84], which introduced other logics to tackle questions of functional completeness, stimulated further research on finite-valued logics. Since then many different finite-valued logics have been defined to treat statements which can have more than just the two truth values *true* and *false*. These additional truth values typically stand for uncertain, vague, undefined or senseless statements. Famous finite-valued logics were introduced, e.g., by Gödel [48], Bochvar [23], Kleene [66] and Belnap [8].

Checking the *derivability* of rules in finite-valued logics is decidable and has been investigated extensively in the literature. In particular, general methods for generating proof systems to check derivability such as tableaux, resolution and multisequent calculi, have been developed, as have standard optimization techniques for these systems such as lemma generation and indexing (see, e.g., [28, 29, 54, 110, 3, 2]). However, checking the *admissibility* of rules in finite-valued logics is not so well-understood. Although the problem is decidable, a naive approach leads to computationally unfeasible procedures even for very small logics. A central goal of this thesis is to obtain a general and more feasible method to check admissibility in finite-valued logics. These techniques can then be useful for improving proof systems to check derivability in the logics or understanding their properties.

Even though the motivation comes from *logic*, the theory developed in this thesis makes use of the notions and methods of *universal algebra*. Many well-known logics are algebraizable in the sense of [21], i.e., they correspond to some *quasivariety* (their algebraic semantics), and hence results obtained in universal algebra can be translated back into the logical context. A logical rule corresponds to a *quasiequation*, i.e., to a finite set of equations implying another equation. A quasiequation

$$\{\ \varphi_1 \approx \psi_1,\ \ldots,\ \varphi_n \approx \psi_n\ \}\ \Rightarrow\ \varphi \approx \psi$$

is called *admissible* in a class of algebras $\mathcal{K}$ if every $\mathcal{K}$-unifier of the premises is a $\mathcal{K}$-unifier of the conclusion, where a $\mathcal{K}$-unifier of an equation $\varphi \approx \psi$ is a substitution σ such that $\sigma(\varphi) \approx \sigma(\psi)$ is valid in $\mathcal{K}$.

The starting point for this work is the observation that for a finite set of finite algebras $\mathcal{K}$, checking admissibility in the quasivariety $\mathbb{Q}(\mathcal{K})$ is the same as checking validity in the free algebra $\mathbf{F}_{\mathcal{K}}(n)$, where n is the maximal cardinality of the algebras in $\mathcal{K}$ (see Theorem 3.9 and Corollary 2.19). This algebra $\mathbf{F}_{\mathcal{K}}(n)$ is finite (see Lemma 2.12), hence checking admissibility in $\mathbb{Q}(\mathcal{K})$ is decidable. But in some cases, even for small n and a small set of small algebras $\mathcal{K}$, the size of $\mathbf{F}_{\mathcal{K}}(n)$ is very large. An implementation of a derived proof system to check validity for such algebras would not be practical. However, sometimes $\mathcal{K}$-admissibility corresponds to validity in other, quite small algebras. We aim to discover these small algebras using features of the free algebra. It turns out that every subalgebra of the free algebra $\mathbf{F}_{\mathcal{K}}(n)$ for which there are homomorphisms onto the algebras in $\mathcal{K}$, may also be used for checking $\mathcal{K}$-admissibility. Unfortunately, these subalgebras of $\mathbf{F}_{\mathcal{K}}(n)$ are not always the smallest algebras with this property. Therefore we provide an algorithm which finds the *minimal* (with respect to the standard multiset ordering) set of algebras satisfying the requirements (see Algorithm 3.1). Using this algorithm MinGenSet we are then able to characterize structural completeness and the related property of almost structural completeness.

These algorithms have been implemented in the tool TAFA, which has then been used to obtain admissibility results (some known and some new) for a wide range of (classes of) algebras. In particular, after showing that all two element algebras are structurally complete, we describe admissibility for all three element groupoids and lattices with up to five elements. We also provide bases for admissible quasiequations for De Morgan and Kleene algebras.

We proceed as follows. First, *Chapter 2* recalls some required notions and results from universal algebra. Then *Chapter 3* develops the theoretical core of the thesis, including results on minimal generating sets for quasivarieties, characterizations for admissibility, structural and almost structural completeness and algorithms to find sets of algebras to check admissibility. *Chapter 4* presents admissibility related results for well known classes of algebras, including, e.g., the proof that all two element algebras are structurally complete, a study of all three element groupoids (see also Appendix A) and bases for admissible quasiequations for the quasivarieties of Kleene and De Morgan lattices and algebras. *Chapter 5* describes the system TAFA, a tool for studying admissibility in finite algebras as well as solving general algebraic problems like calculating subalgebras, different kinds of morphisms, products, congruences and their lattices and checking properties like being subdirectly irreducible. Finally, *Chapter 6* provides a summary of the contribution of the thesis to the theory of admissible rules and lists some ideas for further work.

Chapter 3 presents joint work with George Metcalfe of which substantial parts have been published as [76] and [77]. *Section 4.4* also presents joint work with George Metcalfe that has appeared in [75]. The rest of *Chapter 4* and the whole of *Chapter 5* is my own work, some of which has appeared in [92, 93].

Chapter 2

Preliminaries

This chapter introduces some basic definitions and known results of Universal Algebra that we will need to develop the theoretical machinery of the following chapters. We refer to [25] and [49] for further details.

2.1 Algebras

A *language* is a set of operation symbols $\mathcal{L}$ with a nonnegative integer $\mathrm{ar}(*)$ assigned to each operation symbol $* \in \mathcal{L}$, called the *arity* of $*$. We say that $*$ is *n-ary* if $\mathrm{ar}(*) = n$ for some operation symbol $* \in \mathcal{L}$ (*nullary*, *unary* or *binary* if n is 0, 1 or 2, respectively). An *$\mathcal{L}$-algebra* $\mathbf{A}$ (*algebra*, if the language is clear from the context) is an ordered pair consisting of a nonempty set A (the *universe* of $\mathbf{A}$) and an n-ary operation $*^{\mathbf{A}} \colon A^n \to A$ corresponding to each n-ary operation symbol $*$ of $\mathcal{L}$ (as usual, calling nullary operations *constants*). We often omit superscripts when describing the operations of an algebra. Sometimes we write $\mathcal{F}$ for the set of operations on an algebra $\mathbf{A}$ and represent the algebra as follows:

$$\mathbf{A} \ := \ \langle A, \mathcal{F} \rangle.$$

An algebra $\mathbf{A}' := \langle A, \mathcal{F}' \rangle$ is called a *reduct* of the algebra $\mathbf{A} := \langle A, \mathcal{F} \rangle$ if $\mathcal{F}' \subseteq \mathcal{F}$. The $\mathcal{L}$-algebra $\mathbf{A}$ is said to be *finite* if A is a finite set and $\mathcal{L}$ consists of finitely many operation symbols with finite arity.

We use the letters x_1, x_2, …, y_1, y_2, …, z_1, z_2, …, sometimes without indices, to denote countably infinitely many *variables*. For a set X of variables and a language $\mathcal{L}$, the set $\mathrm{Tm}_{\mathcal{L}}(X)$ of *$\mathcal{L}$-terms over X* is inductively defined as usual: every variable $x \in X$ is an $\mathcal{L}$-term over X and if $\varphi_1, \dots, \varphi_n$ are $\mathcal{L}$-terms over X and the operation symbol $* \in \mathcal{L}$ has arity n, then also $*(\varphi_1, \dots, \varphi_n)$ is an $\mathcal{L}$-term over X. We call members of $\mathrm{Tm}_{\mathcal{L}}(\{x_1, x_2, \dots\})$ *$\mathcal{L}$-terms* and denote them by $\mathrm{Tm}_{\mathcal{L}}(\omega)$ or just $\mathrm{Tm}_{\mathcal{L}}$. We usually omit brackets where convenient and use the *infix notation* for binary operation symbols, e.g., we write $x * y$ instead of $*(x, y)$. The $\mathcal{L}$-terms over X build the universe of the *term algebra over X*. The operations of $\mathbf{Tm}_{\mathcal{L}}(X)$ are defined as expected, i.e., for each n-ary $* \in \mathcal{L}$, $\varphi_1, \dots, \varphi_n \in \mathrm{Tm}_{\mathcal{L}}(X)$,

$$*^{\mathbf{Tm}_{\mathcal{L}}(X)}(\varphi_1, \dots, \varphi_n) \ := \ *(\varphi_1, \dots, \varphi_n).$$

Let $\varphi(x_1, \dots, x_n)$ be an $\mathcal{L}$-term over some set X and $\mathbf{A}$ an $\mathcal{L}$-algebra. We define a map $\varphi^{\mathbf{A}} \colon A^n \to A$ called the *term operation* on $\mathbf{A}$ corresponding to φ (we often omit the superscripts for convenience) as follows:

- If φ is a variable x_i $(0 \leq i \leq n)$, then $\varphi^{\mathbf{A}}(a_1, \dots, a_n) \ := \ a_i$.

- If φ is of the form $*(\varphi_1(x_1, \dots, x_n), \dots, \varphi_k(x_1, \dots, x_n))$ for an operation symbol $* \in \mathcal{L}$ of arity k, then

$$\varphi^{\mathbf{A}}(a_1, \dots, a_n) \ := \ *^{\mathbf{A}}(\varphi_1^{\mathbf{A}}(a_1, \dots, a_n), \dots, \varphi_k^{\mathbf{A}}(a_1, \dots, a_n)).$$

The n-ary *i-th projection* p_i^n is defined by $p_i^n(x_1, \dots, x_n) := x_i$ and the n-ary *constant operation* c_a^n is defined by $c_a^n(x_1, \dots, x_n) := a$. Let $*$ be an n-ary operation and $\diamond_1, \dots, \diamond_n$ m-ary operations. Then the *composition of* *the operations* $*$ *and* $\diamond$ is defined as the m-ary operation

$$*[\diamond_1, \dots, \diamond_n](x_1, \dots, x_m) \ := \ *(\diamond_1(x_1, \dots, x_m), \dots, \diamond_n(x_1, \dots, x_m)).$$

The *clone of operations of* $\mathbf{A}$, denoted $\mathrm{Clo}\,\mathbf{A}$, is the smallest set of operations on A which contains all projections p_i^n $(n \in \mathbb{N}, 0 \leq i \leq n)$, the operations of $\mathbf{A}$ and is closed under compositions. We write $\mathrm{Clo_n}\,\mathbf{A}$ to denote the set of n-ary members of $\mathrm{Clo}\,\mathbf{A}$. We say that two algebras $\mathbf{A}_1$ and $\mathbf{A}_2$ over the universe A are *clone equivalent* if $\mathrm{Clo}\,\mathbf{A}_1 = \mathrm{Clo}\,\mathbf{A}_2$, and write $\mathbf{A}_1 \approx_{clo} \mathbf{A}_2$. We say that an n-ary operation $*(x_1, \dots, x_n)$ is *definable by the set of operations* $\mathcal{F}$ of an algebra $\mathbf{A} := \langle A, \mathcal{F} \rangle$ if there exists an $*' \in \mathrm{Clo}\,\mathbf{A}$ such that $*(a_1, \dots, a_n) = *'(a_1, \dots, a_n)$ for any $a_1, \dots, a_n \in A$.

Example 2.1. *Let* $\mathbf{A} := \langle \{0, 1\}, \wedge, \neg \rangle$ *be an algebra with* $x \wedge y := \min\{x, y\}$ *and* $\neg x := 1 - x$*. Then the unary operation* c_0^1 *is definable by* $\{\wedge, \neg\}$*, e.g., by* $x \wedge \neg x$*, while the nullary operation* c_0^0 *is not definable by* $\{\wedge, \neg\}$.

Note that two algebras $\mathbf{A}_1 := \langle A, \mathcal{F} \rangle$ and $\mathbf{A}_2 := \langle A, \mathcal{G} \rangle$ are clone equivalent if and only if every operation in $\mathcal{F}$ is definable by $\mathcal{G}$ and vice versa.

Example 2.2. *Let* $\mathbf{G}_{16} := \langle G_{16}, \circ \rangle$ *and* $\mathbf{G}_{17} := \langle G_{17}, * \rangle$ *be two* $\mathcal{L}$*-algebras with universe* $\{a, b, c\}$ *and the binary operations defined as follows (see Appendix A):*

$\circ$	a	b	c
a	a	a	a
b	a	a	a
c	b	c	a

$*$	a	b	c
a	a	a	a
b	a	a	a
c	b	c	b

$\mathbf{G}_{16}$ *and* $\mathbf{G}_{17}$ *are clone equivalent since for any* $x, y \in \{a, b, c\}$*, we have that*

$$\begin{aligned} x \circ y &= (x * (x * (y * x))) * y, \quad \textit{and} \\ x * y &= x \circ ((x \circ y) \circ (x \circ x)). \end{aligned}$$

Let $\mathbf{A}$ and $\mathbf{B}$ be two $\mathcal{L}$-algebras. $\mathbf{B}$ is said to be a *subalgebra* of $\mathbf{A}$, written $\mathbf{B} \leq \mathbf{A}$, if $B \subseteq A$ and every operation of $\mathbf{B}$ is the restriction of the corresponding operation of $\mathbf{A}$ to the universe B.

A map $h\colon A \to B$ is called a *homomorphism from* $\mathbf{A}$ *to* $\mathbf{B}$, written $h\colon \mathbf{A} \to \mathbf{B}$, if it is compatible with all operations, i.e., for all $a_1, \ldots, a_n \in A$ and every n-ary operation $* \in \mathcal{L}$,

$$h(*^{\mathbf{A}}(a_1, \ldots, a_n)) \;=\; *^{\mathbf{B}}(h(a_1), \ldots, h(a_n)).$$

The algebra $\mathbf{C}$ with universe $C := \{h(a) : a \in A\} \subseteq B$ and the restrictions of the operations of $\mathbf{B}$ to C as operations is called a *homomorphic image of* $\mathbf{A}$. $\mathbf{A}$ is called the *prehomomorphic image* of $\mathbf{C}$ in this case. The *kernel* of a homomorphism $h\colon \mathbf{A} \to \mathbf{B}$ is defined by $\ker h := \{(a_1, a_2) \in A^2 : h(a_1) = h(a_2)\}$. We often call injective homomorphisms *embeddings* and a homomorphism $h\colon \mathbf{A} \to \mathbf{B}$ that is bijective is called an *isomorphism*. We say that $\mathbf{A}$ is *isomorphic to* $\mathbf{B}$ if there is an isomorphism from $\mathbf{A}$ to $\mathbf{B}$, and write $\mathbf{A} \cong \mathbf{B}$.

The *direct product* of the $\mathcal{L}$-algebras $\{\mathbf{A}_i\}_{i \in I}$ for an index set I has universe $\prod_{i \in I} A_i$ and its operations are defined coordinate-wise, i.e., for an n-ary operation symbol $* \in \mathcal{L}$, the i-th coordinate is defined as follows:

$$*^{\prod_{i \in I} \mathbf{A}_i}(a_1, \ldots, a_n)(i) \;:=\; *^{\mathbf{A}_i}(a_1(i), \ldots, a_n(i)).$$

2.2 Varieties and Quasivarieties

An $\mathcal{L}$-*equation* is a pair of $\mathcal{L}$-terms, written $\varphi \approx \psi$. An $\mathcal{L}$-*clause* is defined as an ordered pair Σ, Δ of finite sets of $\mathcal{L}$-equations, written $\Sigma \Rightarrow \Delta$, and called an $\mathcal{L}$-*quasiequation* if $|\Delta| = 1$ and an $\mathcal{L}$-*negative clause* if $\Delta = \emptyset$. As usual, if the language is clear from the context, we may omit the prefix $\mathcal{L}$.

Let us fix $\mathcal{K}$ to be a class of $\mathcal{L}$-algebras, noting that often in what follows $\mathcal{K}$ will consist of a finite set of $\mathcal{L}$-algebras $\mathbf{A}_1, \ldots, \mathbf{A}_n$, and in this case we typically omit brackets. For a finite set of $\mathcal{L}$-equations $\Sigma \cup \Delta$, we say that the set Σ is $\mathcal{K}$-*satisfiable* if $\Sigma \subseteq \ker h$ for some $\mathbf{A} \in \mathcal{K}$ and homomorphism $h\colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{A}$, and the $\mathcal{L}$-clause $\Sigma \Rightarrow \Delta$ is $\mathcal{K}$-*valid* (or, $\mathcal{K}$ *satisfies* the $\mathcal{L}$-

clause $\Sigma \Rightarrow \Delta$), written $\Sigma \models_{\mathcal{K}} \Delta$ (or $\models_{\mathcal{K}} \Delta$, if $\Sigma = \emptyset$), if for every $\mathbf{A} \in \mathcal{K}$ and homomorphism $h \colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{A}$,

$$\Sigma \ \subseteq\ \ker h \qquad \text{implies} \qquad \Delta \cap \ker h \ \neq\ \emptyset.$$

The class $\mathcal{K}$ is said to be *axiomatized* by a set of $\mathcal{L}$-clauses Λ if $\mathcal{K}$ is the class of $\mathcal{L}$-algebras $\mathbf{A}$ such that all $\mathcal{L}$-clauses in Λ are $\mathbf{A}$-valid, i.e., $\mathbf{A} \in \mathcal{K}$ if and only if $\Sigma \models_{\mathbf{A}} \Delta$ for all $\Sigma \Rightarrow \Delta \in \Lambda$. The class of $\mathcal{L}$-algebras $\mathcal{K}$ is called an *$\mathcal{L}$-universal class*, *$\mathcal{L}$-variety*, *$\mathcal{L}$-quasivariety* or *$\mathcal{L}$-antivariety* if it is axiomatized by a set of $\mathcal{L}$-clauses, $\mathcal{L}$-equations, $\mathcal{L}$-quasiequations or $\mathcal{L}$-negative clauses, respectively. The universal class $\mathbb{U}(\mathcal{K})$, variety $\mathbb{V}(\mathcal{K})$, quasivariety $\mathbb{Q}(\mathcal{K})$ and antivariety $\mathbb{V}^{-}(\mathcal{K})$ *generated by* $\mathcal{K}$ are the smallest universal class, variety, quasivariety and antivariety containing $\mathcal{K}$, respectively. $\mathcal{K}$ is called the *generating set* in this cases. If $\mathcal{K}$ is a finite set of finite $\mathcal{L}$-algebras, these classes are called *finitely generated*.

Moreover, let $\mathbb{H}$, $\mathbb{I}$, $\mathbb{S}$, $\mathbb{P}$, $\mathbb{P}_U$, $\mathbb{P}_U^*$ and $\mathbb{H}^{-1}$ be the class operators (mapping classes of algebras to classes of algebras) of taking homomorphic images, isomorphic images, subalgebras, products, ultraproducts, non-empty ultraproducts[1] and prehomomorphic images, respectively. E.g., $\mathbf{A} \in \mathbb{H}(\mathcal{K})$ if $\mathbf{A}$ is a homomorphic image of some $\mathbf{B} \in \mathcal{K}$. Birkhoff proved in his famous HSP theorem [17] that the equational classes, i.e., varieties, are exactly the classes which are closed under $\mathbb{H}$, $\mathbb{S}$ and $\mathbb{P}$. Tarski refined this result to $\mathbb{V}(\mathcal{K}) = \mathbb{HSP}(\mathcal{K})$ in [105]. Similar results were also obtained for other syntactically defined classes of algebras:

Theorem 2.3. *Let $\mathcal{K}$ be a class of $\mathcal{L}$-algebras.*

(a) $\mathbb{U}(\mathcal{K}) = \mathbb{ISP}_U(\mathcal{K})$ *([25, Theorem V.2.20]).*

(b) $\mathbb{V}(\mathcal{K}) = \mathbb{HSP}(\mathcal{K})$ *([17, Theorem 6] and [105, Theorem]).*

(c) $\mathbb{Q}(\mathcal{K}) = \mathbb{ISPP}_U(\mathcal{K})$ *([51, Theorem]).*

[1] We refer to Section IV.6 in [25] for a proper definition of ultraproducts since they do not play any special role when considering finite sets of finite algebras.

(d) $\mathbb{V}^{-}(\mathcal{K}) = \mathbb{H}^{-1}\mathbb{S}\mathbb{P}_U^*(\mathcal{K})$ *([50, Theorem 1.2]).*

If $\mathcal{K}$ is a finite set of finite algebras, then $\mathbb{P}_U(\mathcal{K}) \subseteq \mathbb{I}(\mathcal{K})$ (see [25, Lemma IV.6.5]); hence $\mathbb{U}(\mathcal{K}) = \mathbb{IS}(\mathcal{K})$, $\mathbb{Q}(\mathcal{K}) = \mathbb{ISP}(\mathcal{K})$ and $\mathbb{V}^{-}(\mathcal{K}) = \mathbb{H}^{-1}\mathbb{S}(\mathcal{K})$. Note furthermore, that all varieties and quasivarieties contain trivial algebras since empty products are allowed (contrary to, e.g., [25]).

Example 2.4. *A* Boolean algebra *is an algebra* $\mathbf{B} := \langle B, \wedge, \vee, \neg, \bot, \top\rangle$ *such that* $\langle B, \wedge, \vee\rangle$ *is a distributive lattice (see Section 2.3) and the following hold:* $x \wedge \bot \approx \bot$, $x \vee \top \approx \top$, $x \wedge \neg x \approx \bot$, $x \vee \neg x \approx \top$. *If* $\mathbf{B_2}$ *is the two element Boolean algebra, then* $\mathbb{Q}(\mathbf{B_2}) = \mathbb{Q}(\mathbf{B_2} \times \mathbf{B_2})$ *since* $\mathbf{B_2} \in \mathbb{P}(\mathbf{B_2})$ *and* $\mathbf{B_2} \in \mathbb{IS}(\mathbf{B_2} \times \mathbf{B_2})$*, but* $\mathbb{U}(\mathbf{B_2} \times \mathbf{B_2}) \nsubseteq \mathbb{U}(\mathbf{B_2})$ *since* $\mathbf{B_2} \times \mathbf{B_2} \notin \mathbb{IS}(\mathbf{B_2})$.

It is crucial to note that equations are preserved by the class operators defining universal classes, varieties and quasivarieties.

Lemma 2.5 ([25, Lemma II.11.3])**.** *Let* $\mathcal{K}$ *be a class of* $\mathcal{L}$*-algebras. Then* $\mathcal{K}$, $\mathbb{I}(\mathcal{K})$, $\mathbb{H}(\mathcal{K})$, $\mathbb{S}(\mathcal{K})$, $\mathbb{P}(\mathcal{K})$, $\mathbb{P}_U(\mathcal{K})$ *and* $\mathbb{P}_U^*(\mathcal{K})$ *satisfy the same equations.*

Proof. The cases $\mathcal{K}$, $\mathbb{I}(\mathcal{K})$, $\mathbb{H}(\mathcal{K})$, $\mathbb{S}(\mathcal{K})$, $\mathbb{P}(\mathcal{K})$ are covered in [25]. For the ultraproducts note that $\mathbb{P}_U^*(\mathcal{K}) \subseteq \mathbb{P}_U(\mathcal{K}) \subseteq \mathbb{HP}(\mathcal{K})$. □

2.3 Lattices and Congruences

A *lattice* is an algebra $\mathbf{L} := \langle L, \wedge, \vee\rangle$, where $\wedge$ and $\vee$ (called *meet* and *join*) are binary operations satisfying the following equations:

$$\begin{array}{rrcl}
\textit{commutativity} & x \wedge y & \approx & y \wedge x \\
 & x \vee y & \approx & y \vee x \\
\textit{associativity} & x \wedge (y \wedge z) & \approx & (x \wedge y) \wedge z \\
 & x \vee (y \vee z) & \approx & (x \vee y) \vee z \\
\textit{idempotency} & x \wedge x & \approx & x \\
 & x \vee x & \approx & x \\
\textit{absorption} & x \wedge (x \vee y) & \approx & x \\
 & x \vee (x \wedge y) & \approx & x.
\end{array}$$

A *distributive* lattice is a lattice admitting the equations of

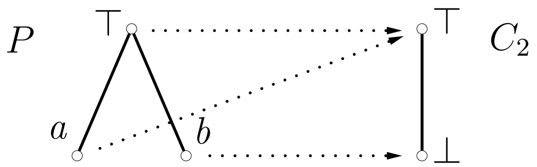
$$\begin{array}{lrcl} \textit{distributivity} & x \wedge (y \vee z) & \approx & (x \wedge y) \vee (x \wedge z) \\ & x \vee (y \wedge z) & \approx & (x \vee y) \wedge (x \vee z), \end{array}$$

and a *bounded lattice* $\mathbf{L}$ is an algebra $\langle L, \wedge, \vee, \bot, \top \rangle$ such that $\langle L, \wedge, \vee \rangle$ is a lattice and the constants $\bot$ and $\top$ satisfy the equations $x \vee \bot \approx x$ and $x \wedge \top \approx x$. A lattice $\mathbf{L}$ is *complete* if for every subset $B \subseteq L$ the meet $\bigwedge B$ and join $\bigvee B$ exist. An element a of a complete lattice $\mathbf{L}$ is called *completely meet irreducible (meet irreducible)* if for any (finite) subset $B \subseteq L$, $a = \bigwedge B$ implies $a \in B$. The notion of *join irreducibility* is defined dually.

Besides this algebraic definition of a lattice there exists a corresponding order theoretic definition. To establish this connection we first have to introduce the notion of a partially ordered set. A *partially ordered set*, *poset* for short, is a set P together with a binary relation $\leq$ (the *partial order*), written $\langle P, \leq \rangle$, such that for all $a, b, c \in P$ the following hold: $a \leq a$ (*reflexivity*), $a \leq b$ and $b \leq a$ imply $a = b$ (*antisymmetry*) and $a \leq b$ and $b \leq c$ imply $a \leq c$ (*transitivity*). An *upper (lower) bound* of a subset A of a poset P is an element $b \in P$ such that $a \leq b$ ($b \leq a$) for all $a \in A$. A *least upper (greatest lower) bound* is an upper (lower) bound $m \in P$ such that for all other upper (lower) bounds b, $m \leq b$ ($b \leq m$). We now define a *lattice* as a poset P such that any two elements $a, b \in P$ have a least upper bound and a greatest lower bound. These two definitions of lattices are equivalent in the following sense: If $\mathbf{L}$ is a lattice by the first definition, we define a partial order $\leq$ on L by $a \leq b$ iff $a = a \wedge b$. If P is a lattice by the second definition, we define the binary operations $\wedge$ and $\vee$ to be the greatest lower bound and the least upper bound, respectively.

We say that b *covers* a in the poset P, denoted $a \prec b$, if $a \leq c \leq b$ for any $c \in P$ implies $a = c$ or $b = c$. We usually draw finite posets using *Hasse Diagrams*: a circle "$\circ$" represents an element of the poset and whenever $a \prec b$, we draw the b-circle above the a-circle and connect them with a line.

Example 2.6. *The Hasse Diagram depicted below shows two posets $\langle P, \leq \rangle$ and $\langle C_2, \leq \rangle$. P is not a lattice since $a, b \in P$ do not have a lower bound.*



Interpreted as algebras $\mathbf{P} := \langle P, \wedge, \vee\rangle$ *(setting* $a \wedge b := b$*) and* $\mathbf{C}_2 := \langle C_2, \wedge, \vee\rangle$ *we can define a homomorphism* h *as indicated by the dotted arrows in the picture. Note that* $\mathbf{C}_2$ *satisfies the* $\wedge$*-distributivity law* $x \wedge (y \vee z) \approx (x \wedge y) \vee (x \wedge z)$*, while* $\mathbf{P}$ *does not (e.g.,* $a \wedge (b \vee a) = a \wedge \top = a \neq \top = b \vee a = (a \wedge b) \vee (a \wedge a)$*). Hence prehomomorphisms do not preserve equations (compare with Lemma 2.5).*

An *equivalence relation* θ on a set A is a subset $\theta \subseteq A \times A$, such that for all $a, b, c \in A$, $\langle a, a\rangle \in \theta$ (*reflexivity*), $\langle a, b\rangle \in \theta$ implies $\langle b, a\rangle \in \theta$ (*symmetry*) and $\langle a, b\rangle, \langle b, c\rangle \in \theta$ implies $\langle a, c\rangle \in \theta$ (*transitivity*). For the elements $a \in A$ we define $a/\theta := \{b \in A : \langle a, b\rangle \in \theta\}$, the *equivalence class modulo* θ (sometimes just $[a]$ if it is clear which equivalence relation we mean). The *quotient of* A *by* θ, denoted A/θ, is the collection of the equivalence classes of θ, i.e., $A/\theta := \{a/\theta : a \in A\}$. A *congruence* on an $\mathcal{L}$-algebra $\mathbf{A}$ is an equivalence relation θ on A satisfying for each n-ary operation symbol $*$ of $\mathcal{L}$ and $a_1, \ldots, a_n, b_1, \ldots, b_n \in A$:

$$\langle a_1, b_1\rangle \in \theta, \ldots, \langle a_n, b_n\rangle \in \theta \quad \text{implies} \quad \langle *^{\mathbf{A}}(a_1, \ldots, a_n), *^{\mathbf{A}}(b_1, \ldots, b_n)\rangle \in \theta.$$

The congruences of $\mathbf{A}$, denoted $\mathrm{Con}(\mathbf{A})$, form a complete lattice $\mathbf{Con}(\mathbf{A}) := \langle \mathrm{Con}(\mathbf{A}), \wedge, \vee\rangle$ with bottom element $\Delta_{\mathbf{A}} := \{\langle a, a\rangle : a \in A\}$ and top element $\nabla_{\mathbf{A}} := \{\langle a, b\rangle : a, b \in A\}$, where the meet of two congruences θ_1, θ_2 on $\mathbf{A}$ is just the intersection $\theta_1 \cap \theta_2$ and the join of θ_1 and θ_2 is the intersection of all congruences containing $\theta_1 \cup \theta_2$.

Given $\theta \in \mathrm{Con}(\mathbf{A})$, the *quotient algebra of* $\mathbf{A}$ *by* θ is the $\mathcal{L}$-algebra $\mathbf{A}/\theta$ with universe A/θ and operations defined for each n-ary operation symbol $* \in \mathcal{L}$ by

$$*^{\mathbf{A}/\theta}(a_1/\theta, \ldots, a_n/\theta) \ := \ *^{\mathbf{A}}(a_1, \ldots, a_n)/\theta.$$

For an algebra $\mathbf{A}$ and a congruence $\theta \in \mathrm{Con}(\mathbf{A})$, the *natural homomorphism* $\nu_\theta \colon \mathbf{A} \to \mathbf{A}/\theta$ (sometimes just ν for convenience) sends each element of A to its congruence class, i.e., $\nu_\theta(a) := a/\theta$.

We finish this section by stating that term operations behave as the operations of an algebra with respect to congruences and homomorphisms:

Lemma 2.7 ([25, Theorem II.10.3])**.** *Let $\mathbf{A}$, $\mathbf{B}$ be $\mathcal{L}$-algebras, $\varphi(x_1, \ldots, x_n)$ an n-ary $\mathcal{L}$-term and $a_1, \ldots, a_n, b_1, \ldots, b_n \in A$.*

(a) *If $\theta \in \mathrm{Con}(\mathbf{A})$ and $\langle a_i, b_i \rangle \in \theta$ for $1 \leq i \leq n$, then $\langle \varphi^{\mathbf{A}}(a_1, \ldots, a_n), \varphi^{\mathbf{A}}(b_1, \ldots, b_n) \rangle \in \theta$.*

(b) *If $h \colon \mathbf{A} \to \mathbf{B}$ is a homomorphism, then $h(\varphi^{\mathbf{A}}(a_1, \ldots, a_n)) = \varphi^{\mathbf{B}}(h(a_1), \ldots, h(a_n))$.*

2.4 Subdirect Representations

An $\mathcal{L}$-algebra $\mathbf{A}$ is called a *subdirect product* of the family $(\mathbf{A}_i)_{i \in I}$ if there exist surjective homomorphisms $f_i \colon \mathbf{A} \to \mathbf{A}_i$ for each i of the index set I such that the induced homomorphism

$$f \colon \mathbf{A} \to \prod_{i \in I} \mathbf{A}_i, \qquad f(x)(i) \ := \ f_i(x),$$

is an embedding. In this case, f is called a *subdirect representation of* $\mathbf{A}$ and the members of $(\mathbf{A}_i)_{i \in I}$ are called *subdirect components* (for this representation). If $\mathcal{K}$ is a class of $\mathcal{L}$-algebras and $\mathbf{A}_i \in \mathcal{K}$ for all $i \in I$, then $\mathbf{A}$ is called a *$\mathcal{K}$-subdirect product of the algebras* $\mathbf{A}_i$*, $i \in I$* and f is called a *$\mathcal{K}$-subdirect embedding*. $\mathbf{A}$ is called *$\mathcal{K}$-subdirectly irreducible* if for every $\mathcal{K}$-subdirect embedding $f \colon \mathbf{A} \to \prod_{i \in I} \mathbf{A}_i$, $\mathbf{A}$ is isomorphic to $\mathbf{A}_i$ for some $i \in I$.

The well known Subdirect Decomposition Theorem for equational classes [19, Theorem 2] also holds for more general classes, including quasivarieties:

Theorem 2.8 ([27, Corollary 6])**.** *Let $\mathcal{Q}$ be a quasivariety and $\mathbf{A} \in \mathcal{Q}$. Then $\mathbf{A}$ is a $\mathcal{Q}$-subdirect product of $\mathcal{Q}$-subdirectly irreducible members of $\mathcal{Q}$.*

Moreover, the $\mathbb{Q}(\mathcal{K})$-subdirectly irreducible algebras always embed into a generating algebra $\mathbf{A} \in \mathcal{K}$ of the quasivariety:

Lemma 2.9 ([49, Proposition 3.1.6])**.** *Let $\mathbb{Q}(\mathcal{K})$ be a finitely generated quasivariety and $\mathbf{A}$ a $\mathbb{Q}(\mathcal{K})$-subdirectly irreducible algebra. Then $\mathbf{A} \in \mathbb{IS}(\mathcal{K})$.*

Theorem 2.10 ([18, Theorem VI.11])**.** *Let $\mathbf{A}$ be a subdirect product of the family $(\mathbf{A}_i)_{i\in I}$. Then there exist for $i \in I$, congruences $\theta_i \in \mathrm{Con}(\mathbf{A})$ such that $\mathbf{A}_i \cong \mathbf{A}/\theta_i$ and $\bigcap_{i\in I} \theta_i = \Delta_{\mathbf{A}}$.*

Conversely, let $(\theta_i)_{i\in I}$ be a family of congruences on $\mathbf{A}$. Then the quotient $\mathbf{A}/(\bigcap_{i\in I} \theta_i)$ is a subdirect product of the family $(\mathbf{A}/\theta_i)_{i\in I}$.

We now translate this theorem to $\mathcal{Q}$-subdirect representations of an algebra $\mathbf{A}$, where $\mathcal{Q}$ is a quasivariety containing $\mathbf{A}$. This establishes the relationship between $\mathcal{Q}$-subdirect representations of $\mathbf{A}$ and families of $\mathcal{Q}$-congruences on $\mathbf{A}$. The set of *$\mathcal{Q}$-congruences on* $\mathbf{A}$ is defined as

$$\mathrm{Con}_{\mathcal{Q}}(\mathbf{A}) \ := \ \{\theta \in \mathrm{Con}(\mathbf{A}) : \mathbf{A}/\theta \in \mathcal{Q}\}.$$

Corollary 2.11. *Let $\mathcal{Q}$ be a quasivariety and $\mathbf{A} \in \mathcal{Q}$.*

(a) *Let $\mathbf{A}$ be a $\mathcal{Q}$-subdirect product of the family $(\mathbf{A}_i)_{i\in I}$. Then there exist for $i \in I$, $\mathcal{Q}$-congruences $\theta_i \in \mathrm{Con}_{\mathcal{Q}}(\mathbf{A})$ such that $\mathbf{A}_i \cong \mathbf{A}/\theta_i$ and $\bigcap_{i\in I} \theta_i = \Delta_{\mathbf{A}}$.*

Conversely, let $(\theta_i)_{i\in I}$ be a family of $\mathcal{Q}$-congruences on $\mathbf{A}$ with $\bigcap_{i\in I} \theta_i = \Delta_{\mathbf{A}}$. Then $\mathbf{A}$ is a $\mathcal{Q}$-subdirect product of the family $(\mathbf{A}/\theta_i)_{i\in I}$.

(b) *$\mathbf{A}$ is $\mathcal{Q}$-subdirectly irreducible iff it is trivial or the bottom element $\Delta_{\mathbf{A}}$ of $\mathrm{Con}_{\mathcal{Q}}(\mathbf{A})$ is completely meet-irreducible.*

Proof. (a) Suppose that $\mathbf{A}$ is a $\mathcal{Q}$-subdirect product of the family $(\mathbf{A}_i)_{i\in I}$ and hence obviously also a subdirect product of the family $(\mathbf{A}_i)_{i\in I}$. So there exist for $i \in I$, congruences $\theta_i \in \mathrm{Con}(\mathbf{A})$ such that $\mathbf{A}_i \cong \mathbf{A}/\theta_i$ and $\bigcap_{i \in I} \theta_i = \Delta_{\mathbf{A}}$ by Theorem 2.10. But then $\theta_i \in \mathrm{Con}_{\mathcal{Q}}(\mathbf{A})$ for all $i \in I$ since $\mathbf{A}_i \in \mathcal{Q}$ for all $i \in I$ by assumption and $\mathcal{Q}$ is closed under isomorphisms. For the other direction consider a family of $\mathcal{Q}$-congruences $(\theta_i)_{i \in I}$ on $\mathbf{A}$ with $\bigcap_{i \in I} \theta_i = \Delta_{\mathbf{A}}$. By Theorem 2.10, $\mathbf{A}/(\bigcap_{i \in I} \theta_i)$ is a subdirect product of the family $(\mathbf{A}/\theta_i)_{i \in I}$. But since $\bigcap_{i \in I} \theta_i = \Delta_{\mathbf{A}}$ and $\mathbf{A}/\theta_i \in \mathcal{Q}$ for all $i \in I$ by assumption, $\mathbf{A}$ is a $\mathcal{Q}$-subdirect product of the family $(\mathbf{A}/\theta_i)_{i \in I}$.

(b) Define $J := \mathrm{Con}_{\mathcal{Q}}(\mathbf{A}) \setminus \{\Delta_{\mathbf{A}}\}$. Suppose for a contradiction that $\mathbf{A}$ is non-trivial and $\mathcal{Q}$-subdirectly irreducible and that $\bigcap J = \Delta_{\mathbf{A}}$. $\mathbf{A}$ is a $\mathcal{Q}$-subdirect product of the algebras $\{\mathbf{A}/\theta : \theta \in J\}$ by (a). Since $\mathbf{A}$ is subdirectly irreducible, there is an isomorphism $f \colon \mathbf{A} \to \mathbf{A}/\theta$ for some $\theta \in J$, which contradicts $\Delta_{\mathbf{A}} \notin J$. For the other direction suppose that $\bigcap J \neq \Delta_{\mathbf{A}}$ and let $f \colon \mathbf{A} \to \prod_{i \in I} \mathbf{A}_i$ be a subdirect representation of $\mathbf{A}$. By (a) there exist for $i \in I$, $\mathcal{Q}$-congruences $\theta_i \in \mathrm{Con}_{\mathcal{Q}}(\mathbf{A})$ such that $\mathbf{A}_i \cong \mathbf{A}/\theta_i$ and $\bigcap_{i \in I} \theta_i = \Delta_{\mathbf{A}}$. So by assumption $\theta_i = \Delta_{\mathbf{A}}$ for some $j \in I$, hence $\mathbf{A} \cong \mathbf{A}/\theta_j$ and $\mathbf{A}$ is subdirectly irreducible. □

Note, moreover, that the number of congruences needed to obtain a subdirect representation of a finite algebra $\mathbf{A}$ is at most $|A|$, the maximal number of coatoms of the congruence lattice $\mathrm{Con}(\mathbf{A})$.

2.5 Free Algebras

Given a class of $\mathcal{L}$-algebras $\mathcal{K}$ and a set of variables X such that either $X \neq \emptyset$ or $\mathcal{L}$ contains at least one constant symbol, the term algebra $\mathbf{Tm}_{\mathcal{L}}(X)$ exists and admits the following congruence:

$$\Psi_{\mathcal{K}}(X) \ := \ \bigcap \{\theta \in \mathrm{Con}(\mathbf{Tm}_{\mathcal{L}}(X)) : \mathbf{Tm}_{\mathcal{L}}(X)/\theta \in \mathbb{IS}(\mathcal{K})\}.$$

Following [25], we let $\overline{X} := X/\Psi_{\mathcal{K}}(X)$ and define the *free algebra of* $\mathcal{K}$ *over* $\overline{X}$ by:

$$\mathbf{F}_{\mathcal{K}}(\overline{X}) \ := \ \mathbf{Tm}_{\mathcal{L}}(X)/\Psi_{\mathcal{K}}(X).$$

Then $\mathbf{F}_{\mathcal{K}}(\overline{X})$ has the *universal mapping property* for $\mathcal{K}$ over $\overline{X}$, i.e., for each $\mathbf{A} \in \mathcal{K}$, any map from $\overline{X}$ to A extends to a homomorphism from $\mathbf{F}_{\mathcal{K}}(\overline{X})$ to $\mathbf{A}$ (see [25, Theorem II.10.10]).

Note that $\mathbf{F}_{\mathcal{K}}(\overline{X}) \cong \mathbf{F}_{\mathcal{K}}(\overline{Y})$ whenever $|X| = |Y|$. Also $|\overline{X}| = |X|$ if $\mathcal{K}$ contains at least one non-trivial algebra (in this case we write x for $\overline{x} \in \overline{X}$). Hence we may consider for each cardinal κ, the (unique up to isomorphism) *free algebra of* $\mathcal{K}$ *on* κ *generators* $\mathbf{F}_{\mathcal{K}}(\kappa)$, where $\mathbf{F}_{\mathcal{K}}(\kappa_1)$ is a subalgebra of $\mathbf{F}_{\mathcal{K}}(\kappa_2)$ for cardinals $\kappa_1 \leq \kappa_2$.

It is crucial for us that for a finite set of finite algebras[2] $\mathcal{K}$, the free algebra $\mathbf{F}_{\mathcal{K}}(n)$ is finite for all $n \in \mathbb{N}$:

Lemma 2.12 ([17, Corollary 2]). *For any set of finite* $\mathcal{L}$*-algebras* $\mathcal{K} := \{\mathbf{A}_1, \ldots, \mathbf{A}_m\}$ *and* $n \in \mathbb{N}$*:*

$$|F_{\mathcal{K}}(n)| \ \leq \ \prod_{i=1}^{m} |A_i|^{|A_i|^n}.$$

We will sometimes need the fact that a free algebra is contained in the corresponding quasivariety and variety.

Lemma 2.13 ([25, Theorem II.10.12]). *Suppose that* $\mathbf{Tm}_{\mathcal{L}}(X)$ *exists. Then for a nonempty class* $\mathcal{K}$ *of* $\mathcal{L}$*-algebras,*

$$\mathbf{F}_{\mathcal{K}}(\overline{X}) \ \in \ \mathbb{ISP}(\mathcal{K}).$$

Note that if $\mathcal{K}$ is a universal class or an antivariety, then $\mathbf{F}_{\mathcal{K}}(\overline{X}) \in \mathcal{K}$ does not hold in general.

Theorem 2.14 ([25, Theorem II.11.4]). *Let* $\mathcal{K}$ *be a class of* $\mathcal{L}$*-algebras and* $\varphi, \psi \in \mathrm{Tm}_{\mathcal{L}}(x_1, \ldots, x_n)$*. Then the following are equivalent:*

(1) $\models_{\mathcal{K}} \varphi \approx \psi$.

(2) $\models_{\mathbf{F}_{\mathcal{K}}(\overline{X})} \varphi \approx \psi$.

[2] This was extended in [15, Theorem 2.8] to members of finitely generated varieties.

(3) $\varphi^{\mathbf{F}_{\mathcal{K}}(\overline{X})}(\overline{x}_1, \ldots, \overline{x}_n) = \psi^{\mathbf{F}_{\mathcal{K}}(\overline{X})}(\overline{x}_1, \ldots, \overline{x}_n)$.

(4) $\langle \varphi, \psi \rangle \in \Psi_{\mathcal{K}}(X)$.

It follows that the free algebras of two classes of algebras $\mathcal{K}_1$ and $\mathcal{K}_2$ are the same if and only if $\mathcal{K}_1$ and $\mathcal{K}_2$ satisfy the same equations. Define $\mathrm{Eq}_{\mathcal{K}}(X) := \{\varphi \approx \psi : \varphi, \psi \in \mathrm{Tm}_{\mathcal{L}}(X) \text{ and } \models_{\mathcal{K}} \varphi \approx \psi\}$.

Corollary 2.15. *Let $\mathcal{K}_1, \mathcal{K}_2$ be classes of $\mathcal{L}$-algebras. Then $\mathbf{F}_{\mathcal{K}_1}(\overline{X}) = \mathbf{F}_{\mathcal{K}_2}(\overline{X})$ iff $\mathrm{Eq}_{\mathcal{K}_1}(X) = \mathrm{Eq}_{\mathcal{K}_2}(X)$. In particular, $\mathbf{F}_{\mathbb{V}(\mathcal{K}_1)}(\omega) = \mathbf{F}_{\mathbb{V}(\mathcal{K}_2)}(\omega)$ iff $\mathbb{V}(\mathcal{K}_1) = \mathbb{V}(\mathcal{K}_2)$.*

Combining this with Lemma 2.5 we immediately get:

Corollary 2.16. *Let $\mathcal{K}$ be a class of $\mathcal{L}$-algebras and κ a cardinal number with $\kappa \leq \omega$. Then*

$$\mathbf{F}_{\mathcal{K}}(\kappa) \;=\; \mathbf{F}_{\mathbb{U}(\mathcal{K})}(\kappa) \;=\; \mathbf{F}_{\mathbb{V}(\mathcal{K})}(\kappa) \;=\; \mathbf{F}_{\mathbb{Q}(\mathcal{K})}(\kappa).$$

Also, Lemma 2.13 and Theorem 2.14 imply that the free algebra of a variety $\mathcal{V}$ on infinitely many generators generates $\mathcal{V}$.

Corollary 2.17. *If $\mathcal{V}$ is a variety, then $\mathcal{V} = \mathbb{V}(\mathbf{F}_{\mathcal{V}}(\omega))$.*

If $\mathcal{Q}$ is a finitely generated quasivariety, then only finitely many generators are needed to generate $\mathbb{Q}(\mathbf{F}_{\mathcal{Q}}(\omega))$.

Theorem 2.18 ([99, Lemma 4.1.10])**.** *Let $\mathcal{K}$ be a finite set of finite $\mathcal{L}$-algebras $\{\mathbf{A}_1, \ldots, \mathbf{A}_n\}$ and $m := \max\{|A| : \mathbf{A} \in \mathcal{K}\}$. Then $\mathbb{Q}(\mathbf{F}_{\mathbb{Q}(\mathcal{K})}(\omega)) = \mathbb{Q}(\mathbf{F}_{\mathbb{Q}(\mathcal{K})}(m))$.*

Hence we obtain, using Lemma 2.12, Corollary 2.16 and $\mathbb{P}_U(\mathbf{A}) \subseteq \mathbb{I}(\mathbf{A})$ for a finite algebra $\mathbf{A}$ (see Section 2.2), the following useful corollary:

Corollary 2.19. *Let $\mathcal{K}$ be a finite set of finite $\mathcal{L}$-algebras $\{\mathbf{A}_1, \ldots, \mathbf{A}_n\}$ and $m := \max\{|A| : \mathbf{A} \in \mathcal{K}\}$. Then $\mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega)) = \mathbb{ISPP}_U(\mathbf{F}_{\mathcal{K}}(m)) = \mathbb{ISP}(\mathbf{F}_{\mathcal{K}}(m))$.*

Chapter 3

Finitely Generated Quasivarieties

In this chapter, we address issues of admissibility in finitely generated quasivarieties: that is, quasivarieties generated by a finite number of finite algebras. We start by defining minimal generating sets for a finitely generated quasivariety $\mathcal{Q}$: minimal sets of algebras $\mathcal{K}$ (with respect to some multiset ordering) with $\mathbb{Q}(\mathcal{K}) = \mathcal{Q}$ (*Section 3.1*). Due to the fact that the considered quasivarieties $\mathbb{Q}(\mathcal{K})$ are finitely generated, we are able to present an algorithm which calculates a minimal generating set for $\mathbb{Q}(\mathcal{K})$, given $\mathcal{K}$ (see Algorithm 3.1). *Sections 3.2, 3.4 and 3.5* provide useful characterizations of unification, structural completeness and almost structural completeness, respectively, and *Section 3.3* presents an algorithm to build a proof system for checking admissibility in finitely generated quasivarieties. *Section 3.6* takes a closer look at clone equivalences to prove that if the clones of operations of two algebras $\mathbf{A}_1, \mathbf{A}_2$ are the same, the free algebras $\mathbf{F}_{\mathbf{A}_1}(n)$ and $\mathbf{F}_{\mathbf{A}_2}(n)$ and the corresponding minimal generating sets are isomorphic up to a translation (inside the clone) of their languages. *Section 3.7* explains how the algebraic ideas presented in this chapter can be transferred to finite-valued logics and *Section 3.8* finally gives a concrete example of how a proof system for checking admissibility can be automatically generated.

3.1 Minimal Generating Sets

If $\mathcal{Q}$ is a quasivariety, a quasiequation $\Sigma \Rightarrow \varphi \approx \psi$ is $\mathcal{Q}$-admissible if and only if $\Sigma \models_{\mathbf{F}_{\mathcal{Q}}(\omega)} \varphi \approx \psi$ (see Theorem 3.9). I.e., to check the $\mathcal{Q}$-admissibility of a quasiequation, we have to check whether this quasiequation holds in the quasivariety generated by the free algebra $\mathbf{F}_{\mathcal{Q}}(\omega)$. When $\mathcal{Q}$ is finitely generated there is an $n \in \mathbb{N}$ such that $\mathbf{F}_{\mathcal{Q}}(n)$ generates the same quasivariety (see Theorem 2.18). Since this algebra $\mathbf{F}_{\mathcal{Q}}(n)$ is finite (see Lemma 2.12), it is possible to generate a proof system with a tool such as MUltlog [101] or $_3T^AP$ [7] to check the validity of the quasiequation $\Sigma \Rightarrow \varphi \approx \psi$ (e.g., using MUltseq [47]). See Section 3.8 for an example system.

It is natural to ask for the "smallest" set of algebras generating the quasivariety $\mathbb{Q}(\mathbf{F}_{\mathcal{Q}}(\omega))$. But first we have to determine a suitable measure for comparison. It turns out that a good choice for comparing the cardinalities of the algebras is the multiset well-ordering defined in [34]. Recall that a *multiset* over a set S is an ordered pair $\langle S, f\rangle$ where f is a function $f\colon S \to \mathbb{N}$. The multiset $\langle S, f\rangle$ is called *finite* if $\{x \in S : f(x) > 0\}$ is finite. We will write a finite multiset over S as $[a_1, \ldots, a_n]$ where $a_1, \ldots, a_n \in S$ may include repetitions. For a well-ordered set $\langle S, \leq\rangle$, the *multiset ordering* $\leq_m$ on the set $M(S)$ of finite multisets over S is defined by $\langle S, f\rangle \leq_m \langle S, g\rangle$ if $f(x) > g(x)$ implies that for some $y \in S$, $y > x$ and $g(y) > f(y)$. Intuitively, $\mathcal{M}_1 \leq_m \mathcal{M}_2$ holds for two multisets $\mathcal{M}_1$ and $\mathcal{M}_2$ if $\mathcal{M}_1$ can be obtained from $\mathcal{M}_2$ by replacing its elements with a finite number (possibly zero) of strictly smaller elements of $\mathcal{M}_1$.

Example 3.1. *Let $\mathcal{M}_1 := [1, 1, 3, 3, 3, 7]$ and $\mathcal{M}_2 := [2, 8]$ be multisets over $\mathbb{N}$. Then $\mathcal{M}_1 \leq_m \mathcal{M}_2$ since 8 can be replaced by $3, 3, 3, 7$ and 2 by $1, 1$ to obtain $\mathcal{M}_1$ from $\mathcal{M}_2$.*

We are now able to compare sets of algebras by comparing the corresponding multisets of cardinalities using $\leq_m$. A set of finite $\mathcal{L}$-algebras $\mathcal{A} := \{\mathbf{A}_1, \ldots, \mathbf{A}_n\}$ will be called a *minimal generating set* for the quasivariety $\mathbb{Q}(\mathbf{A}_1, \ldots, \mathbf{A}_n)$ if for every set of finite $\mathcal{L}$-algebras $\mathcal{B} := \{\mathbf{B}_1, \ldots, \mathbf{B}_k\}$:

$$\mathbb{Q}(\mathcal{A}) \;=\; \mathbb{Q}(\mathcal{B}) \quad \text{implies} \quad [|A_1|, \ldots, |A_n|] \;\leq_m\; [|B_1|, \ldots, |B_k|].$$

The smallest free algebra $\mathbf{F}_{\mathcal{K}}(n)$, $n \in \mathbb{N}$, that generates the quasivariety $\mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$ is called the *minimal generating free algebra for* $\mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$. By Theorem 2.18 such a free algebra must exist.

Although it may seem counter-intuitive to say that twenty algebras with three elements are an improvement over one single four element algebra, the measure $\leq_m$ is a good choice for comparing generating sets. Checking a quasiequation with r variables in a finite algebra $\mathbf{A}$ requires checking $|A|^r$ assignments of variables to elements of $\mathbf{A}$. But then checking validity in $\{\mathbf{A}_1, \ldots, \mathbf{A}_n\}$ will involve checking fewer assignments of variables than checking validity in $\{\mathbf{B}_1, \ldots, \mathbf{B}_k\}$ if $[|A_1|, \ldots, |A_n|] \leq_m [|B_1|, \ldots, |B_k|]$ for quasiequations with sufficiently many variables[1].

Given a finitely generated quasivariety, we would like to calculate a minimal generating set. To do this, we need the following decomposition lemma:

Lemma 3.2. *Let $\mathcal{K}$ be a class of $\mathcal{L}$-algebras and suppose that $\mathcal{K}'$ is obtained from $\mathcal{K}$ by either (a) replacing $\mathbf{A} \in \mathcal{K}$ with $\mathbf{A}_1, \ldots, \mathbf{A}_n$ where $\mathbf{A}$ is a $\mathbb{Q}(\mathcal{K})$-subdirect product of $\mathbf{A}_1, \ldots, \mathbf{A}_n$, or (b) replacing $\mathbf{A}, \mathbf{B} \in \mathcal{K}$ with $\mathbf{B}$ where $\mathbf{A} \in \mathbb{IS}(\mathbf{B})$. Then $\mathbb{Q}(\mathcal{K}) = \mathbb{Q}(\mathcal{K}')$.*

Proof. Assume that $\mathcal{K}$ is a class of $\mathcal{L}$-algebras. If $\mathbf{A}$ is a $\mathbb{Q}(\mathcal{K})$-subdirect product of $\mathbf{A}_1, \ldots, \mathbf{A}_n$, then $\mathbf{A} \in \mathbb{ISP}(\mathbf{A}_1, \ldots, \mathbf{A}_n) \subseteq \mathbb{Q}(\mathbf{A}_1, \ldots, \mathbf{A}_n)$ and $\mathbf{A}_1, \ldots, \mathbf{A}_n \in \mathbb{Q}(\mathcal{K})$. Hence $\mathbb{Q}(\mathcal{K}) = \mathbb{Q}(\mathcal{K}')$, where $\mathcal{K}'$ is obtained from $\mathcal{K}$ by replacing $\mathbf{A}$ with $\mathbf{A}_1, \ldots, \mathbf{A}_n$. On the other hand, if $\mathbf{A} \in \mathbb{IS}(\mathbf{B})$, then $\mathbf{A} \in \mathbb{IS}(\mathcal{K} \setminus \{\mathbf{A}\}) \subseteq \mathbb{Q}(\mathcal{K} \setminus \{\mathbf{A}\})$ and hence $\mathbb{Q}(\mathcal{K}) = \mathbb{Q}(\mathcal{K}')$, where $\mathcal{K}'$ is obtained from $\mathcal{K}$ by replacing $\mathbf{A}, \mathbf{B} \in \mathcal{K}$ with $\mathbf{B}$. □

In particular, replacing each algebra $\mathbf{A}$ in a finite set $\mathcal{K}$ of finite algebras with

[1] In the example above where we have twenty three element algebras or one single four-element algebra, we need at least eleven variables to see the advantage with respect to $\leq_m$, since $20 \cdot 3^r < 4^r$ for $r \geq 11$.

the $\mathbb{Q}(\mathcal{K})$-subdirectly irreducible algebras in some $\mathbb{Q}(\mathcal{K})$-subdirect representation of $\mathbf{A}$, then removing any algebra that embeds into another algebra in the set, produces a minimal generating set for $\mathbb{Q}(\mathcal{K})$ that is unique up to isomorphism.

Theorem 3.3. *Suppose that $\mathcal{Q} := \mathbb{Q}(\mathbf{A}_1, \ldots, \mathbf{A}_n)$ where $\mathbf{A}_i$ is a finite $\mathcal{Q}$-subdirectly irreducible algebra for $i \in \{1, \ldots, n\}$ and $\mathbf{A}_i \notin \mathbb{IS}(\mathbf{A}_j)$ for $j \neq i$. Then $\{\mathbf{A}_1, \ldots, \mathbf{A}_n\}$ is the unique minimal generating set for $\mathcal{Q}$ up to isomorphism.*

Proof. Let $\mathcal{Q} := \mathbb{Q}(\mathbf{A}_1, \ldots, \mathbf{A}_n)$ where $\mathbf{A}_i$ is a finite $\mathcal{Q}$-subdirectly irreducible algebra for $i \in \{1, \ldots, n\}$ and $\mathbf{A}_i \notin \mathbb{IS}(\mathbf{A}_j)$ for $j \neq i$. Suppose for a contradiction that $\mathcal{Q} := \mathbb{Q}(\mathbf{B}_1, \ldots, \mathbf{B}_k)$ and $[|B_1|, \ldots, |B_k|] <_m [|A_1|, \ldots, |A_n|]$. Without loss of generality, we can suppose that $\mathbf{B}_j$ is $\mathcal{Q}$-subdirectly irreducible for $j \in \{1, \ldots, k\}$; otherwise, by Theorem 2.8 and Lemma 3.2, $\mathbf{B}_j$ can be replaced with the $\mathcal{Q}$-subdirectly irreducible components of a $\mathcal{Q}$-subdirect representation of $\mathbf{B}_j$ and we obtain a smaller (according to $\leq_m$) generating set of algebras for $\mathcal{Q}$.

It follows that there exists a largest $r \in \mathbb{N}$ such that there are strictly more occurrences of r in $[|A_1|, \ldots, |A_n|]$ than in $[|B_1|, \ldots, |B_k|]$, and for each $r' > r$, the number of occurrences of r' in $[|A_1|, \ldots, |A_n|]$ and $[|B_1|, \ldots, |B_k|]$ are equal. Each $\mathbf{A}_i$ is finite and $\mathcal{Q}$-subdirectly irreducible, and hence by Lemma 2.9, embeds into some $\mathbf{B}_j$ where $|A_i| \leq |B_j|$. If every $\mathbf{A}_i$ of size r embeds into, and is hence isomorphic to, a $\mathbf{B}_j$ of size r, then (by the pigeonhole principle) there must be two isomorphic algebras in $\{\mathbf{A}_1, \ldots, \mathbf{A}_n\}$, a contradiction. Hence, suppose without loss of generality that $\mathbf{A}_1$ embeds into $\mathbf{B}_1$ with $|A_1| = r$ and $|B_1| > r$. But notice now that $\mathbf{B}_1$ is also $\mathcal{Q}$-subdirectly irreducible and hence embeds into some $\mathbf{A}_i$ with $i \in \{2, \ldots, n\}$. So $\mathbf{A}_1 \in \mathbb{IS}(\mathbf{A}_i)$, a contradiction.

Finally, consider any minimal generating set $\{\mathbf{B}_1, \ldots, \mathbf{B}_k\}$ for $\mathcal{Q}$, and suppose for a contradiction that $\mathbf{B}_i \notin \mathbb{I}(\mathbf{A}_1, \ldots, \mathbf{A}_n)$ for some $i \in \{1, \ldots, k\}$. Then by Lemma 2.9, $\mathbf{B}_i$ properly embeds into $\mathbf{A}_j$ for some $j \in \{1, \ldots, n\}$. But also by Lemma 2.9, $\mathbf{A}_j$ embeds into $\mathbf{B}_d$ for some $d \in \{1, \ldots, k\} \setminus \{i\}$.

It follows that $\mathbf{B}_i$ can be embedded into the strictly larger algebra $\mathbf{B}_d$. But then $\{\mathbf{B}_1, \ldots, \mathbf{B}_k\}$ is not a minimal generating set for $\mathcal{Q}$, a contradiction. $\square$

Hence, to calculate a minimal generating set for a finitely generated quasivariety $\mathbb{Q}(\mathcal{K})$, we should find $\mathbb{Q}(\mathcal{K})$-subdirect products with $\mathbb{Q}(\mathcal{K})$-irreducible components of the algebras in $\mathcal{K}$. Recalling the connection between subdirect representations of a given algebra $\mathbf{A}$ and sets of congruences on $\mathbf{A}$ (see Theorem 2.10), it would appear to be a good idea to calculate the set $\mathrm{Con}_{\mathbb{Q}(\mathcal{K})}(\mathbf{A})$, the universe of a sublattice of the lattice of congruences $\mathrm{Con}(\mathbf{A})$ (see [49, Corollary 1.4.11]). It is known that the problem of finding the congruence closure for a given equivalence relation on a finite algebra, i.e., the smallest congruence containing this equivalence, can be solved in polynomial time[2]. The problem of finding the $\mathcal{Q}$-congruence closure of an equivalence relation on a finite algebra with respect to a finitely generated quasivariety $\mathcal{Q}$ appears to be much harder, however. Instead, we use here the following characterization of $\mathcal{Q}$-subdirectly irreducible algebras without needing to calculate the $\mathcal{Q}$-congruence lattice.

Lemma 3.4. *Let $\mathcal{K}$ be a finite set of finite $\mathcal{L}$-algebras and $\mathbf{A} \in \mathbb{Q}(\mathcal{K})$. Then the following are equivalent:*

(1) $\mathbf{A}$ *is $\mathbb{Q}(\mathcal{K})$-subdirectly irreducible.*

(2) $\bigcap\{\theta \in \mathrm{Con}(\mathbf{A}) \setminus \{\Delta_{\mathbf{A}}\} : \mathbf{A}/\theta \in \mathbb{IS}(\mathcal{K})\} \neq \Delta_{\mathbf{A}}$.

Proof. For convenience, let

$$\Theta \;:=\; \{\theta \in \mathrm{Con}(\mathbf{A}) \setminus \{\Delta_{\mathbf{A}}\} : \mathbf{A}/\theta \in \mathbb{IS}(\mathcal{K})\} \;\subseteq\; \mathrm{Con}_{\mathbb{Q}(\mathcal{K})}(\mathbf{A}).$$

(1) $\Rightarrow$ (2) We proceed contrapositively. If $\bigcap \Theta = \Delta_{\mathbf{A}}$, then by Corollary 2.11(a), $\mathbf{A}$ is a $\mathbb{Q}(\mathcal{K})$-subdirect product of algebras in $\{\mathbf{A}/\theta : \theta \in \Theta\}$. But also by Corollary 2.11(b), since $\Delta_{\mathbf{A}} \notin \Theta$, $\mathbf{A}$ is not $\mathbb{Q}(\mathcal{K})$-subdirectly irreducible.

[2] This was used in [33] to provide a polynomial time algorithm for calculating a subdirect representation of a finite algebra. We refer to [24] for the definitions of complexity classes.

(2) ⇒ (1) Again, we proceed contrapositively and assume that $\mathbf{A}$ is not $\mathbb{Q}(\mathcal{K})$-subdirectly irreducible. By combining Theorem 2.8 and Corollary 2.11, there exist $(\theta_i)_{i\in I} \subseteq \mathrm{Con}_{\mathbb{Q}(\mathcal{K})}(\mathbf{A}) \setminus \{\Delta_{\mathbf{A}}\}$ such that $\bigcap_{i\in I} \theta_i = \Delta_{\mathbf{A}}$ and $\mathbf{A}$ is a $\mathbb{Q}(\mathcal{K})$-subdirect product of $\mathbb{Q}(\mathcal{K})$-subdirectly irreducible algebras $\mathbf{A}/\theta_i$ ($i \in I$). But then we have $\mathbf{A}/\theta_i \in \mathbb{IS}(\mathcal{K})$ for each $i \in I$ by Lemma 2.9, so $(\theta_i)_{i\in I} \subseteq \Theta$, and hence $\bigcap \Theta = \Delta_{\mathbf{A}}$. □

We now have all the ingredients necessary to describe the algorithm MinGenSet (see Algorithm 3.1) that calculates the (unique up to isomorphism) minimal generating set for a finitely generated quasivariety.

Theorem 3.5. *For a finite set $\mathcal{K}$ of finite $\mathcal{L}$-algebras,* MinGenSet*($\mathcal{K}$) returns the (unique up to isomorphism) minimal generating set for $\mathbb{Q}(\mathcal{K})$.*

Proof. Let $\mathcal{Q} := \mathbb{Q}(\mathcal{K})$. By Theorem 3.3, it suffices to find a set of $\mathcal{Q}$-subdirectly irreducible algebras that generates $\mathcal{Q}$, where no member of the set embeds into another member of the set. The algorithm proceeds by considering each $\mathbf{A} \in \mathcal{K}$ in turn. First, the congruence lattice $\mathrm{Con}(\mathbf{A})$ is generated (line 10) by checking for all equivalence relations if they are congruences. Next, the congruences $\theta \in \mathrm{Con}(\mathbf{A}) \setminus \{\Delta_{\mathbf{A}}\}$ such that $\mathbf{A}/\theta$ embeds into $\mathbf{A}$ or some other member of $\mathcal{K}$ are collected in sets S_1 and S_2, respectively. If $\bigcap(S_1 \cup S_2) \neq \Delta_{\mathbf{A}}$, then $\mathbf{A}$ is $\mathcal{Q}$-subdirectly irreducible by Lemma 3.4, so the algorithm proceeds to the next algebra in $\mathcal{K}$. Otherwise $\bigcap(S_1 \cup S_2) = \Delta_{\mathbf{A}}$ and by Lemma 3.4, $\mathbf{A}$ is not $\mathcal{Q}$-subdirectly irreducible. In this case, for each $\theta \in S_1 \setminus S_2$, the algebra $\mathbf{A}/\theta$ is added to $\mathcal{K}$ (line 15) and $\mathbf{A}$ is removed from $\mathcal{K}$ (line 17). Note that since the cardinalities of the added algebras are strictly smaller than the cardinality of the removed algebra, the new set of algebras is smaller according to the multiset ordering $\leq_m$. Hence this procedure is terminating. Moreover, the resulting finite set of finite algebras must generate the quasivariety $\mathcal{Q}$ (by Lemma 3.2), contain only $\mathcal{Q}$-subdirectly irreducible algebras, and not contain any algebra that embeds into another member of the set (lines 22–26). Hence MinGenSet($\mathcal{K}$) is the minimal generating set of $\mathbb{Q}(\mathcal{K})$ up to isomorphism by Theorem 3.3. □

Algorithm 3.1 MinGenSet($\mathcal{K}$): For a finite set $\mathcal{K}$ of finite algebras, return the minimal generating set of $\mathbb{Q}(\mathcal{K})$.

```
function MinGenSet($\mathcal{K}$)
    declare $S_1, S_2, C$ : set
    declare $\mathcal{M}$ : list
    declare $\mathbf{A}$ : algebra
    declare $i$ : integer
    $\mathcal{M} \leftarrow \text{list}(\mathcal{K})$
    $i \leftarrow 1$
    while $i \leq \text{length}(\mathcal{M})$ do
        $\mathbf{A} \leftarrow \mathcal{M}[i]$
        $C \leftarrow \text{Con}(\mathbf{A}) \setminus \{\Delta_{\mathbf{A}}\}$
        $S_1 \leftarrow \{\theta \in C : \mathbf{A}/\theta \text{ embeds into } \mathbf{A}\}$
        $S_2 \leftarrow \{\theta \in C : \mathbf{A}/\theta \text{ embeds into some } \mathcal{M}[j] \neq \mathbf{A}\}$
        if $\bigcap(S_1 \cup S_2) = \Delta_{\mathbf{A}}$ then
            for all $\theta$ in $S_1 \setminus S_2$ do
                add $\mathbf{A}/\theta$ to $\mathcal{M}$
            end for
            remove $\mathbf{A}$ from $\mathcal{M}$
        else
            $i \leftarrow i + 1$
        end if
    end while
    for all $\mathbf{A}$ in $\mathcal{M}$ do
        if $\mathbf{A}$ embeds into some $\mathcal{M}[j] \neq \mathbf{A}$ then
            remove $\mathbf{A}$ from $\mathcal{M}$
        end if
    end for
    return set($\mathcal{M}$)
end function
```

We remark that although the algorithm MinGenSet does not need to calculate the $\mathcal{Q}$-congruence lattice, already calculating the congruence lattice of a finite algebra can take exponential time EXPTIME. Moreover, the algorithm repeatedly checks for embeddings, which is in general an NP-complete problem (see [78, 13]).

3.2 Unification

For a class of $\mathcal{L}$-algebras $\mathcal{K}$, a set of $\mathcal{L}$-equations Σ is said to be *$\mathcal{K}$-unifiable*, if there is a homomorphism $\sigma\colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{Tm}_{\mathcal{L}}$ (often called a *substitution*) such that $\models_{\mathcal{K}} \sigma(\varphi) \approx \sigma(\psi)$ for all $\varphi \approx \psi \in \Sigma$. In this case we call σ a *$\mathcal{K}$-unifier* of Σ and say that it *$\mathcal{K}$-unifies* Σ.

Note that a finite set Σ of $\mathcal{L}$-equations is $\mathcal{K}$-unifiable if and only if the $\mathcal{L}$-negative clause $\Sigma \Rightarrow \emptyset$ is not $\mathcal{K}$-admissible. Or equivalently, when $\mathcal{K}$ contains a non-trivial algebra, if and only if the $\mathcal{L}$-quasiequation $\Sigma \Rightarrow x \approx y$ with x, y not occurring in Σ is not $\mathcal{K}$-admissible.

We will prove in Theorem 3.7 that for checking $\mathcal{K}$-unifiability for a given set of $\mathcal{L}$-equations Σ, it suffices to find a smallest subalgebra $\mathbf{C}$ of the free algebra $\mathbf{F}_{\mathcal{K}}(1)$, noting that this is $\mathbf{F}_{\mathcal{K}}(0)$ if the language $\mathcal{L}$ contains constants. Then Σ is $\mathcal{K}$-unifiable if and only if Σ is $\mathbf{C}$-valid, and indeed there is no smaller algebra with this property. First, however, we prove a useful lemma:

Lemma 3.6. *Let $\mathcal{K}$ and $\mathcal{K}'$ be classes of $\mathcal{L}$-algebras. Then the following are equivalent:*

(1) Σ *is $\mathcal{K}$-unifiable iff* Σ *is $\mathcal{K}'$-satisfiable.*

(2) $\mathbb{V}^{-}(\mathcal{K}') = \mathbb{V}^{-}(\mathbf{F}_{\mathcal{K}}(\omega))$.

Proof. Recall (from Section 2.2) that $\mathbb{V}^{-}(\mathcal{K}') = \mathbb{V}^{-}(\mathbf{F}_{\mathcal{K}}(\omega))$ is equivalent to the condition that an $\mathcal{L}$-negative clause $\Sigma \Rightarrow \emptyset$ is $\mathcal{K}'$-valid iff it is $\mathbf{F}_{\mathcal{K}}(\omega)$-valid. However, $\Sigma \Rightarrow \emptyset$ is $\mathcal{K}'$-valid iff Σ is not $\mathcal{K}'$-satisfiable and $\Sigma \Rightarrow \emptyset$ is $\mathbf{F}_{\mathcal{K}}(\omega)$-valid iff Σ is not $\mathbf{F}_{\mathcal{K}}(\omega)$-satisfiable. For the equivalence of (1) and (2), it suffices therefore to show that Σ is $\mathbf{F}_{\mathcal{K}}(\omega)$-satisfiable iff Σ is $\mathcal{K}$-unifiable. Suppose first that $h\colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{F}_{\mathcal{K}}(\omega)$ satisfies Σ. Then any homomorphism $\sigma\colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{Tm}_{\mathcal{L}}$ defined such that $\sigma(x) \in h(x)$ for each variable x is a $\mathcal{K}$-unifier of Σ. Conversely, if σ is a $\mathcal{K}$-unifier of Σ, then the homomorphism $h\colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{F}_{\mathcal{K}}(\omega)$ defined by $h(x) := \sigma(x)/\Psi_{\mathcal{K}}(\omega)$ for each variable x satisfies Σ as required. □

Theorem 3.7. *Let $\mathcal{K}$ be a class of $\mathcal{L}$-algebras and $\mathbf{A} \in \mathbb{S}(\mathbf{F}_{\mathcal{K}}(\omega))$.*

(a) *Σ is $\mathcal{K}$-unifiable iff Σ is $\mathbf{A}$-satisfiable.*

(b) *If $\mathbf{A}$ is a smallest finite subalgebra of $\mathbf{F}_{\mathcal{K}}(\omega)$ and $\mathcal{K}'$ is a class of $\mathcal{L}$-algebras such that Σ is $\mathcal{K}$-unifiable iff Σ is $\mathcal{K}'$-satisfiable, then $|A| \leq |B|$ for each $\mathbf{B} \in \mathcal{K}'$.*

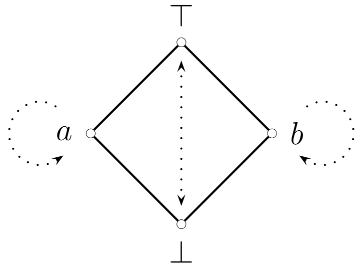
Proof. (a) By assumption, $\mathbf{A} \in \mathbb{V}^{-}(\mathbf{F}_{\mathcal{K}}(\omega))$, so $\mathbb{V}^{-}(\mathbf{A}) \subseteq \mathbb{V}^{-}(\mathbf{F}_{\mathcal{K}}(\omega))$. But also, since $\mathbf{A} \in \mathbb{S}(\mathbf{F}_{\mathcal{K}}(\omega)) \subseteq \mathbb{V}(\mathbf{F}_{\mathcal{K}}(\omega)) = \mathbb{V}(\mathcal{K})$ by Corollary 2.16 and $\mathbf{F}_{\mathcal{K}}(\omega) = \mathbf{F}_{\mathbb{V}(\mathcal{K})}(\omega)$ has the universal mapping property for $\mathbb{V}(\mathcal{K})$ over countably infinitely many generators, we obtain a homomorphism $h\colon \mathbf{F}_{\mathcal{K}}(\omega) \to \mathbf{A}$ defined by $h(x) := a$ for every variable x for some fixed $a \in A$. But then $h[\mathbf{F}_{\mathcal{K}}(\omega)]$ is a subalgebra of $\mathbf{A}$. Hence $\mathbf{F}_{\mathcal{K}}(\omega) \in \mathbb{H}^{-1}\mathbb{S}(\mathbf{A}) \subseteq \mathbb{V}^{-}(\mathbf{A})$. So $\mathbb{V}^{-}(\mathbf{F}_{\mathcal{K}}(\omega)) \subseteq \mathbb{V}^{-}(\mathbf{A})$ and the result follows by Lemma 3.6.

(b) Let $\mathbf{A}$ be a smallest finite subalgebra of $\mathbf{F}_{\mathcal{K}}(\omega)$ and suppose that $\mathcal{K}'$ is a class of $\mathcal{L}$-algebras such that Σ is $\mathcal{K}$-unifiable iff Σ is $\mathcal{K}'$-satisfiable. Then by Lemma 3.6 and part (a), $\mathbb{V}^{-}(\mathcal{K}') = \mathbb{V}^{-}(\mathbf{F}_{\mathcal{K}}(\omega)) = \mathbb{V}^{-}(\mathbf{A})$. Hence if $\mathbf{B} \in \mathcal{K}' \subseteq \mathbb{V}^{-}(\mathcal{K}') = \mathbb{V}^{-}(\mathbf{F}_{\mathcal{K}}(\omega)) = \mathbb{V}^{-}(\mathbf{A}) = \mathbb{H}^{-1}\mathbb{S}\mathbb{P}^{*}_{U}(\mathbf{A}) = \mathbb{H}^{-1}(\mathbf{A})$, then clearly $|A| \leq |B|$. □

Example 3.8. De Morgan algebras *are algebras $\langle A, \wedge, \vee, \neg, \bot, \top\rangle$ such that $\langle A, \wedge, \vee, \bot, \top\rangle$ is a bounded distributive lattice satisfying the following equations:*

$$\begin{array}{rrcl} \textit{involution} & \neg\neg x & \approx & x \\ \textit{De Morgan laws} & \neg(x \wedge y) & \approx & \neg x \vee \neg y \\ & \neg(x \vee y) & \approx & \neg x \wedge \neg y. \end{array}$$

The variety DMA *of De Morgan algebras is generated as a quasivariety by the De Morgan algebra* $\mathbf{D_4}$ *illustrated below (the negation is indicated by the dotted arrows):*



Since there are constants in the language of $\mathbf{D_4}$*, the smallest algebra for checking* DMA*-unifiability is the two element ground algebra* $\mathbf{F}_{\mathbf{D_4}}(0)$*: i.e., the two element Boolean algebra. That is, checking unifiability amounts to checking classical satisfiability. E.g.,* $x \land \neg x \approx x \lor \neg x$ *is not* DMA*-unifiable, since in the two element Boolean algebra,* $\top \land \neg \top \neq \top \lor \neg \top$ *and* $\bot \land \neg \bot \neq \bot \lor \neg \bot$*. The case of the "constant-free" variety* DML *of De Morgan lattices, generated as a quasivariety by* $\mathbf{D}_\mathbf{4}^\ell := \langle \{\bot, a, b, \top\}, \land, \lor, \neg \rangle$*, is not so immediate. However, there is also a smallest two element subalgebra of* $\mathbf{F}_{\mathbf{D}_\mathbf{4}^\ell}(\omega)$ *with elements corresponding to* $x \land \neg x$ *and* $x \lor \neg x$*. So checking* DML*-unifiability amounts again to checking classical satisfiability.*

3.3 Admissibility

Let $\mathcal{K}$ be a class of $\mathcal{L}$-algebras. An $\mathcal{L}$-quasiequation $\Sigma \Rightarrow \varphi \approx \psi$ is called $\mathcal{K}$-admissible if every $\mathcal{K}$-unifier σ of Σ also $\mathcal{K}$-unifies $\varphi \approx \psi$. More formally, $\Sigma \Rightarrow \varphi \approx \psi$ is *$\mathcal{K}$-admissible* (or *admissible in $\mathcal{K}$*) if for every homomorphism $\sigma \colon \mathbf{Tm}_\mathcal{L} \to \mathbf{Tm}_\mathcal{L}$:

$$\models_\mathcal{K} \sigma(\varphi') \approx \sigma(\psi') \text{ for all } \varphi' \approx \psi' \in \Sigma \quad \text{implies} \quad \models_\mathcal{K} \sigma(\varphi) \approx \sigma(\psi).$$

Actually, $\mathcal{K}$-admissible quasiequations are simply the quasiequations that are valid in $\mathbf{F}_\mathcal{K}(\omega)$. We integrate this fact, which was already proven in [99, Theorem 1.4.5], into the following characterization theorem.

Theorem 3.9 ([99, Theorem 1.4.5] and [26, Theorem 2]). *Let $\mathcal{K}$ be a class of $\mathcal{L}$-algebras and $\Sigma \cup \{\varphi \approx \psi\}$ a finite set of $\mathcal{L}$-equations. Then the following are equivalent:*

(1) $\Sigma \Rightarrow \varphi \approx \psi$ *is $\mathcal{K}$-admissible.*

(2) $\Sigma \Rightarrow \varphi \approx \psi$ *is $\mathbb{Q}(\mathcal{K})$-admissible.*

(3) $\Sigma \models_{\mathbf{F}_{\mathcal{K}}(\omega)} \varphi \approx \psi$.

(4) $\mathbb{V}(\mathcal{K}) = \mathbb{V}(\{\mathbf{A} \in \mathbb{Q}(\mathcal{K}) : \Sigma \models_{\mathbf{A}} \varphi \approx \psi\})$.

Proof. (1) $\Rightarrow$ (2) Suppose $\Sigma \Rightarrow \varphi \approx \psi$ is $\mathcal{K}$-admissible and $\models_{\mathbb{Q}(\mathcal{K})} \sigma(\varphi') \approx \sigma(\psi')$ for all $\varphi' \approx \psi' \in \Sigma$ and some homomorphism $\sigma\colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{Tm}_{\mathcal{L}}$. By Lemma 2.5, $\models_{\mathcal{K}} \sigma(\varphi') \approx \sigma(\psi')$ for all $\varphi' \approx \psi' \in \Sigma$ and since $\Sigma \Rightarrow \varphi \approx \psi$ is $\mathcal{K}$-admissible also $\models_{\mathcal{K}} \sigma(\varphi) \approx \sigma(\psi)$. Again by Lemma 2.5, $\models_{\mathbb{Q}(\mathcal{K})} \sigma(\varphi) \approx \sigma(\psi)$ and hence $\Sigma \Rightarrow \varphi \approx \psi$ is $\mathbb{Q}(\mathcal{K})$-admissible.

(2) $\Rightarrow$ (1) is similar.

(1) $\Rightarrow$ (3) Suppose that $\Sigma \Rightarrow \varphi \approx \psi$ is $\mathcal{K}$-admissible and let $g\colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{F}_{\mathcal{K}}(\omega)$ be a homomorphism such that $\Sigma \subseteq \ker g$. We define a map σ that sends each variable x to a member of the equivalence class $g(x)$. By the universal mapping property of $\mathbf{Tm}_{\mathcal{L}}$, this extends to a homomorphism $\sigma\colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{Tm}_{\mathcal{L}}$. But since $\nu(\sigma(x)) = g(x)$ for each variable x (ν is the natural homomorphism for the congruence $\Psi_{\mathcal{K}}(\omega)$), we obtain $\nu \circ \sigma = g$. But then $\Sigma \subseteq \ker(\nu \circ \sigma)$, so for each $\varphi' \approx \psi' \in \Sigma$, we have $\nu(\sigma(\varphi')) = \nu(\sigma(\psi'))$ and therefore $\models_{\mathcal{K}} \sigma(\varphi') \approx \sigma(\psi')$. Hence by assumption, $\models_{\mathcal{K}} \sigma(\varphi) \approx \sigma(\psi)$, and $g(\varphi) = \nu(\sigma(\varphi)) = \nu(\sigma(\psi)) = g(\psi)$ as required.

(3) $\Rightarrow$ (1) Suppose that $\Sigma \models_{\mathbf{F}_{\mathcal{K}}(\omega)} \varphi \approx \psi$ and let $\sigma\colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{Tm}_{\mathcal{L}}$ be a homomorphism such that $\models_{\mathcal{K}} \sigma(\varphi') \approx \sigma(\psi')$ for each $\varphi' \approx \psi' \in \Sigma$ and hence $\nu(\sigma(\varphi')) = \nu(\sigma(\psi'))$. By assumption, $\nu(\sigma(\varphi)) = \nu(\sigma(\psi))$. Hence $\models_{\mathcal{K}} \sigma(\varphi) \approx \sigma(\psi)$ as required.

We define $\mathcal{Q}' := \{\mathbf{A} \in \mathbb{Q}(\mathcal{K}) : \Sigma \models_{\mathbf{A}} \varphi \approx \psi\}$ for the rest of the proof.

(3) $\Rightarrow$ (4) Suppose that $\Sigma \models_{\mathbf{F}_{\mathcal{K}}(\omega)} \varphi \approx \psi$. Then $\mathbf{F}_{\mathcal{K}}(\omega) \in \mathcal{Q}'$ and, using Corollary 2.17 and Lemma 2.13, $\mathbb{V}(\mathcal{K}) = \mathbb{V}(\mathbf{F}_{\mathcal{K}}(\omega)) \subseteq \mathbb{V}(\mathcal{Q}')$, hence $\mathbb{V}(\mathcal{K}) = \mathbb{V}(\mathcal{Q}')$ since $\mathcal{Q}' \subseteq \mathbb{Q}(\mathcal{K}) \subseteq \mathbb{V}(\mathcal{K})$.

(4) ⇒ (2): Suppose $\mathbb{V}(\mathcal{K}) = \mathbb{V}(\mathcal{Q}')$ and let $\sigma\colon \mathbf{Tm}_\mathcal{L} \to \mathbf{Tm}_\mathcal{L}$ be a homomorphism such that $\models_{\mathbb{Q}(\mathcal{K})} \sigma(\varphi') \approx \sigma(\psi')$ for all $\varphi' \approx \psi' \in \Sigma$. Since $\Sigma \models_{\mathcal{Q}'} \varphi \approx \psi$ and $\mathcal{Q}' \subseteq \mathbb{Q}(\mathcal{K})$, $\models_{\mathcal{Q}'} \sigma(\varphi) \approx \sigma(\psi)$ and by assumption, $\models_\mathcal{K} \sigma(\varphi) \approx \sigma(\psi)$. Hence by Lemma 2.5, $\models_{\mathbb{Q}(\mathcal{K})} \sigma(\varphi) \approx \sigma(\psi)$ as required. □

Example 3.10. *The following quasiequations, expressing meet and join semidistributivity for $\mathcal{L} := \{\wedge, \vee\}$ are satisfied by all free lattices (see [64, Lemma 2.6]), and are therefore admissible in the variety of lattices.*

$$\begin{array}{rclcrcl} x \wedge y & \approx & x \wedge z & \Rightarrow & x \wedge y & \approx & x \wedge (y \vee z) \\ x \vee y & \approx & x \vee z & \Rightarrow & x \vee y & \approx & x \vee (y \wedge z). \end{array}$$

Given a class $\mathcal{K}$ of $\mathcal{L}$-algebras, we are interested in determining when the $\mathcal{K}$-admissibility of quasiequations coincides with their $\mathcal{K}'$-validity in another class of $\mathcal{L}$-algebras $\mathcal{K}'$. By Theorem 3.9, this is the case exactly when $\mathbb{Q}(\mathcal{K}') = \mathbb{Q}(\mathbf{F}_\mathcal{K}(\omega))$. The next result provides a further useful characterization of this situation.

Theorem 3.11. *Let $\mathcal{K}$ be a class of $\mathcal{L}$-algebras and $\Sigma \cup \{\varphi \approx \psi\}$ a finite set of $\mathcal{L}$-equations. Then the following are equivalent:*

(1) $\Sigma \Rightarrow \varphi \approx \psi$ *is $\mathcal{K}$-admissible iff* $\Sigma \models_{\mathcal{K}'} \varphi \approx \psi$.

(2) $\mathbb{Q}(\mathcal{K}') = \mathbb{Q}(\mathbf{F}_\mathcal{K}(\omega))$.

(3) $\mathcal{K}' \subseteq \mathbb{Q}(\mathbf{F}_\mathcal{K}(\omega))$ *and* $\mathcal{K} \subseteq \mathbb{V}(\mathcal{K}')$.

Proof. (1) ⇔ (2) follows directly from Theorem 3.9.

(2) ⇒ (3) Suppose that $\mathbb{Q}(\mathcal{K}') = \mathbb{Q}(\mathbf{F}_\mathcal{K}(\omega))$. Then clearly $\mathcal{K}' \subseteq \mathbb{Q}(\mathbf{F}_\mathcal{K}(\omega))$. Moreover, $\mathbb{V}(\mathcal{K}') = \mathbb{V}(\mathbb{Q}(\mathcal{K}')) = \mathbb{V}(\mathbb{Q}(\mathbf{F}_\mathcal{K}(\omega))) = \mathbb{V}(\mathbf{F}_\mathcal{K}(\omega)) = \mathbb{V}(\mathcal{K})$, so $\mathcal{K} \subseteq \mathbb{V}(\mathcal{K}')$.

(3) ⇒ (2) Suppose that $\mathcal{K}' \subseteq \mathbb{Q}(\mathbf{F}_\mathcal{K}(\omega))$ and $\mathcal{K} \subseteq \mathbb{V}(\mathcal{K}')$. Then clearly $\mathbb{Q}(\mathcal{K}') \subseteq \mathbb{Q}(\mathbf{F}_\mathcal{K}(\omega))$. But also $\mathbb{V}(\mathcal{K}) \subseteq \mathbb{V}(\mathcal{K}') \subseteq \mathbb{V}(\mathbb{Q}(\mathbf{F}_\mathcal{K}(\omega))) = \mathbb{V}(\mathbf{F}_\mathcal{K}(\omega)) = \mathbb{V}(\mathcal{K})$. That is, $\mathbb{V}(\mathcal{K}) = \mathbb{V}(\mathcal{K}')$. Hence $\mathbf{F}_\mathcal{K}(\omega) = \mathbf{F}_{\mathcal{K}'}(\omega) \in \mathbb{Q}(\mathcal{K}')$ and $\mathbb{Q}(\mathbf{F}_\mathcal{K}(\omega)) \subseteq \mathbb{Q}(\mathcal{K}')$. □

For checking $\mathcal{K}$-admissibility, we make use of a known result for finitely generated quasivarieties (see [99, Lemma 4.1.10]), obtained here as a corollary of Theorem 3.11:

Corollary 3.12. *Let $\mathcal{K}$ be a finite set of finite $\mathcal{L}$-algebras with $n := \max\{|A| : \mathbf{A} \in \mathcal{K}\}$.*

(a) $\mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega)) = \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(n))$.

(b) $\Sigma \Rightarrow \varphi \approx \psi$ *is $\mathcal{K}$-admissible iff* $\Sigma \models_{\mathbf{F}_{\mathcal{K}}(n)} \varphi \approx \psi$.

Proof. Observe first that each $\mathbf{A} \in \mathcal{K}$ is a homomorphic image of $\mathbf{F}_{\mathcal{K}}(n)$. That is, define any surjective map from the n generators of $\mathbf{F}_{\mathcal{K}}(n)$ to A; this extends to a surjective homomorphism from $\mathbf{F}_{\mathcal{K}}(n)$ onto $\mathbf{A}$ since $\mathbf{F}_{\mathcal{K}}(n)$ has the universal mapping property for $\mathcal{K}$ over n generators. So $\mathcal{K} \subseteq \mathbb{V}(\mathbf{F}_{\mathcal{K}}(n))$ and, since also $\mathbf{F}_{\mathcal{K}}(n) \in \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$, (a) and (b) follow by Theorem 3.11. □

Hence, since the finitely generated free algebra $\mathbf{F}_{\mathcal{K}}(n)$ is finite when $\mathcal{K}$ is a finite set of finite algebras (see Lemma 2.12), checking $\mathcal{K}$-admissibility of quasiequations is decidable. However, even when $\mathcal{K}$ consists of a small number of small algebras, free algebras on a small number of generators can be quite large. For example, the free algebra $\mathbf{F}_{\mathbf{D_4}}(2)$ (see Example 3.8) has 168 elements. We therefore seek smaller algebras or finite sets of smaller algebras that also generate $\mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$ as a quasivariety. In fact, since $\mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$ is finitely generated, we may apply the multiset ordering $\leq_m$ and seek a minimal generating set of finite algebras for this quasivariety that is unique up to isomorphism. One strategy would therefore be to apply the algorithm MinGenSet directly to $\mathbf{F}_{\mathcal{K}}(n)$. However, this method is not feasible for large free algebras, since it involves the computationally labor-intensive task of building the congruence lattice of $\mathbf{F}_{\mathcal{K}}(n)$. Instead, we make use of the following corollary of Theorem 3.11:

Corollary 3.13. *Let $\mathcal{K}$ be a class of $\mathcal{L}$-algebras and $\mathcal{K}' \subseteq \mathbb{S}(\mathbf{F}_{\mathcal{K}}(\omega))$ such that $\mathcal{K} \subseteq \mathbb{H}(\mathcal{K}')$.*

(a) $\mathbb{Q}(\mathcal{K}') = \mathbb{Q}(\mathbf{F}_\mathcal{K}(\omega))$.

(b) $\Sigma \Rightarrow \varphi \approx \psi$ *is* $\mathcal{K}$*-admissible iff* $\Sigma \models_{\mathcal{K}'} \varphi \approx \psi$.

Hence, given a class $\mathcal{K}$ of $\mathcal{L}$-algebras, we might seek a set $\mathcal{K}'$ of smallest subalgebras (according to $\leq_m$) of the free algebra $\mathbf{F}_\mathcal{K}(\omega)$ such that $\mathcal{K} \subseteq \mathbb{H}(\mathcal{K}')$ to reduce the complexity of checking admissibility. Note, however, that this set $\mathcal{K}'$ might not be the minimal generating set for $\mathbb{Q}(\mathbf{F}_\mathcal{K}(\omega))$.

Example 3.14. *Consider the algebra* $\mathbf{G_{106}} := \langle \{a, b, c\}, \circ \rangle$ *with the binary operation* $\circ$ *defined as follows (see also Appendix A):*

$\circ$	a	b	c
a	a	a	a
b	a	b	b
c	a	c	b

The minimal generating free algebra for $\mathbb{Q}(\mathbf{F_{G_{106}}}(\omega))$ *has two generators and ten elements. There are* 21 *subalgebras of* $\mathbf{F_{G_{106}}}(2)$ *which are pre-homomorphic images of* $\mathbf{G_{106}}$*, out of* 93 *subalgebras in total (see filled dots in Figure 3.1). The two smallest subalgebras* $\mathbf{A}$ *with* $\mathbf{G_{106}} \in \mathbb{H}(\mathbf{A})$ *have four elements, but* MinGenSet$(\mathbb{Q}(\mathbf{F_{G_{106}}}(2)))$ *consists of two algebras* $\mathbf{B}_1 := \langle \{a, b\}, \circ \rangle$, $\mathbf{B}_2 := \langle \{a, b\}, \circ \rangle$ *with*

$\circ^{\mathbf{B}_1}$	a	b
a	a	b
b	b	b

$\circ^{\mathbf{B}_2}$	a	b
a	b	a
b	b	b

and hence $\mathbf{A}$ *is not the best choice with respect to the multiset ordering* $\leq_m$.

We combine the idea of decomposition via the algorithm MinGenSet and the search for subalgebras of the minimal generating free algebra that still generate the quasivariety, using Corollary 3.13, into the algorithm AdmAlgs (see Algorithm 3.2). This algorithm calculates the (unique up to isomor-

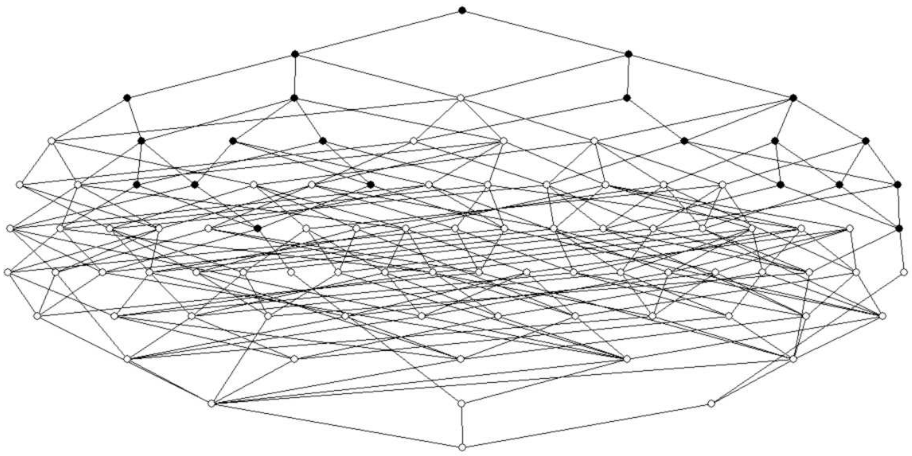


Figure 3.1: Lattice of subuniverses of the algebra $\mathbf{F}_{\mathbf{G_{106}}}(2)$.

phism) minimal generating set for $\mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$ for a finite set $\mathcal{K}$ of finite $\mathcal{L}$-algebras.

Theorem 3.15. *For a finite set $\mathcal{K}$ of finite $\mathcal{L}$-algebras,* AdmAlgs*($\mathcal{K}$) returns the (unique up to isomorphism) minimal generating set for $\mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$.*

Proof. Let $\mathcal{K}$ be a finite set of finite $\mathcal{L}$-algebras. When AdmAlgs is applied to $\mathcal{K}$, first $\mathcal{D} := \text{MinGenSet}(\mathcal{K})$ is calculated, which typically is a small set of small algebras with $\mathbb{Q}(\mathcal{K}) = \mathbb{Q}(\mathcal{D})$ (see Theorem 3.5). We know by Theorem 2.18 that $\mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega)) = \mathbb{Q}(\mathbf{F}_{\mathcal{D}}(n))$ where $n := \max\{|D| : \mathbf{D} \in \mathcal{D}\}$. By Corollary 3.13 it even suffices that the free algebras are prehomomorphic images of the algebras in $\mathcal{D}$. Such free algebras are calculated in line 7 for each $\mathbf{A} \in \mathcal{D}$ by the procedure[3] $\text{Free}(\mathbf{A}, \mathcal{D})$, which returns the smallest free algebra $\mathbf{F}_{\mathcal{D}}(n)$, $n \leq |A|$, with $\mathbf{A} \in \mathbb{H}(\mathbf{F}_{\mathcal{D}}(n))$. (The procedure begins by checking the smallest free algebra $\mathbf{F}_{\mathcal{D}}(0)$ or $\mathbf{F}_{\mathcal{D}}(1)$, then increases the number of generators one at a time.) The algorithm then searches for progressively smaller subalgebras of $\mathbf{F}_{\mathcal{D}}(m)$ which have $\mathbf{A}$ as a homomorphic image. More precisely, the procedure $\text{SubPreHom}(\mathbf{A}, \mathbf{B})$ searches for a proper subalgebra of $\mathbf{B}$ that is a homomorphic image of $\mathbf{A}$, returning $\mathbf{B}$ if no such algebra exists (line 9). This process terminates with a (hopefully reasonably small)

[3] We obviously do not calculate the same free algebra twice in the implementation.

Algorithm 3.2 AdmAlgs($\mathcal{K}$): For a finite set $\mathcal{K}$ of finite algebras, return the minimal generating set of $\mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$.

```
function AdmAlgs($\mathcal{K}$)
    declare $\mathcal{A}, \mathcal{D}$ : set
    declare $\mathbf{B}, \mathbf{B}'$ : algebra
    $\mathcal{D} \leftarrow$ MinGenSet($\mathcal{K}$)
    $\mathcal{A} \leftarrow \emptyset$
    for all $\mathbf{A} \in \mathcal{D}$ do
        $\mathbf{B} \leftarrow$ Free($\mathbf{A}, \mathcal{D}$)
        $\mathbf{B}' \leftarrow$ SubPreHom($\mathbf{A}, \mathbf{B}$)
        while $\mathbf{B}' \neq \mathbf{B}$ do
            $\mathbf{B} \leftarrow \mathbf{B}'$
            $\mathbf{B}' \leftarrow$ SubPreHom($\mathbf{A}, \mathbf{B}$)
        end while
        add $\mathbf{B}$ to $\mathcal{A}$
    end for
    return MinGenSet($\mathcal{A}$)
end function
```

algebra which is added to a set $\mathcal{A}$. Again using Corollary 3.13, $\mathbb{Q}(\mathcal{A}) = \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$. Finally, the procedure MinGenSet is applied to $\mathcal{A}$ to get the minimal generating set of $\mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$ by Theorem 3.5. □

3.4 Structural Completeness

We now turn our attention to classes of algebras for which admissibility and validity of quasiequations coincide. More formally, a class $\mathcal{K}$ of $\mathcal{L}$-algebras is said to be *structurally complete* if for any $\mathcal{L}$-quasiequation $\Sigma \Rightarrow \varphi \approx \psi$:

$$\Sigma \Rightarrow \varphi \approx \psi \text{ is } \mathcal{K}\text{-admissible} \quad \text{iff} \quad \Sigma \models_{\mathcal{K}} \varphi \approx \psi.$$

We say that $\mathbf{A}$ is *structurally complete* if $\{\mathbf{A}\}$ is structurally complete.

Using Theorem 3.9, this is true if and only if $\mathbb{Q}(\mathcal{K}) = \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$ and leads to the following useful characterization, which also includes the equivalent condition proved by Berman [9]:

Theorem 3.16 ([9, Proposition 2.3])**.** *Let $\mathcal{K}$ be a class of $\mathcal{L}$-algebras. Then the following are equivalent:*

(1) *$\mathcal{K}$ is structurally complete.*

(2) $\mathbb{Q}(\mathcal{K}) = \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$.

(3) *$\mathcal{Q}' \subset \mathbb{Q}(\mathcal{K})$ for some quasivariety $\mathcal{Q}'$ implies $\mathbb{V}(\mathcal{Q}') \subset \mathbb{V}(\mathcal{K})$.*

Proof. (1) ⇔ (2) Let $\mathcal{K}$ be a class of $\mathcal{L}$-algebras and $\Sigma \Rightarrow \varphi \approx \psi$ any $\mathcal{L}$-quasiequation. By the definition of structural completeness and Theorem 3.9, $\Sigma \models_{\mathcal{K}} \varphi \approx \psi$ iff $\Sigma \Rightarrow \varphi \approx \psi$ is $\mathcal{K}$-admissible iff $\Sigma \models_{\mathbf{F}_{\mathcal{K}}(\omega)} \varphi \approx \psi$ as required.

(2) ⇒ (3) Suppose that $\mathbb{Q}(\mathcal{K}) = \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$ and, for a contradiction, that $\mathcal{Q}' \subset \mathbb{Q}(\mathcal{K})$ and $\mathbb{V}(\mathcal{Q}') = \mathbb{V}(\mathcal{K})$ for some quasivariety $\mathcal{Q}'$. But then $\mathbb{Q}(\mathcal{K}) \subseteq \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega)) = \mathbb{Q}(\mathbf{F}_{\mathcal{Q}'}(\omega)) \subseteq \mathcal{Q}'$ by Corollaries 2.15 and 2.16, a contradiction.

(3) ⇒ (2) Assume that $\mathbb{V}(\mathcal{Q}') \subset \mathbb{V}(\mathcal{K})$ for every quasivariety $\mathcal{Q}' \subset \mathbb{Q}(\mathcal{K})$. Using Lemma 2.13 it is clear that $\mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega)) \subseteq \mathbb{Q}(\mathcal{K})$. Suppose for a contra-

diction that $\mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega)) \subset \mathbb{Q}(\mathcal{K})$, so $\mathbb{V}(\mathbf{F}_{\mathcal{K}}(\omega)) \subset \mathbb{V}(\mathcal{K})$ by assumption, which is a contradiction by Corollaries 2.16 and 2.17. □

This provides a method for establishing structural completeness for quasivarieties. A quasivariety $\mathcal{Q}$ is structurally complete if each member of a class of algebras generating $\mathcal{Q}$ as a quasivariety can be embedded into the free algebra $\mathbf{F}_{\mathcal{Q}}(\omega)$, since then any quasiequation failing in one of the generating algebras also fails in $\mathbf{F}_{\mathcal{Q}}(\omega)$. More precisely:

Theorem 3.17 ([30, Theorem 3.3])**.** *Let $\mathcal{K}$ be a class of $\mathcal{L}$-algebras and suppose that for each $\mathbf{A} \in \mathcal{K}$, there is a map $g^{\mathbf{A}} \colon A \to \mathrm{Tm}_{\mathcal{L}}$ such that $\nu \circ g^{\mathbf{A}}$ embeds $\mathbf{A}$ into $\mathbf{F}_{\mathcal{K}}(\omega)$, where ν is the natural homomorphism (see Section 2.3). Then $\mathcal{K}$ is structurally complete.*

Proof. Let $\mathcal{K}$ be a class of $\mathcal{L}$-algebras and suppose that each $\mathbf{A} \in \mathcal{K}$ embeds into $\mathbf{F}_{\mathcal{K}}(\omega)$. $\mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega)) \subseteq \mathbb{Q}(\mathcal{K})$ by Lemma 2.13. On the other hand $\mathbf{A} \in \mathbb{IS}(\mathbf{F}_{\mathcal{K}}(\omega)) \subseteq \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$ for each $\mathbf{A} \in \mathcal{K}$, hence $\mathbb{Q}(\mathcal{K}) \subseteq \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$ and $\mathcal{K}$ is structurally complete by Theorem 3.16. □

Combining this result with Corollary 2.19 we obtain:

Corollary 3.18. *Let $\mathcal{K}$ be a finite set of finite $\mathcal{L}$-algebras and suppose that for each $\mathbf{A} \in \mathcal{K}$, there is a map $g^{\mathbf{A}} \colon A \to \mathrm{Tm}_{\mathcal{L}}$ such that $\nu \circ g^{\mathbf{A}}$ embeds $\mathbf{A}$ into $\mathbf{F}_{\mathcal{K}}(m)$, where $m := \max\{|A| : \mathbf{A} \in \mathcal{K}\}$. Then $\mathcal{K}$ is structurally complete.*

Example 3.19. *Consider the variety BA of Boolean algebras, generated as a quasivariety by the two element Boolean algebra $\mathbf{B_2}$ (see Example 2.4). Define $g(0) := \bot$ and $g(1) := \top$. Then $\nu \circ g$ is a homomorphism embedding $\mathbf{B_2}$ into $\mathbf{F}_{\mathsf{BA}}(0)$ and hence BA is structurally complete.*

If a quasivariety $\mathcal{Q}$ is not structurally complete, then the question arises of how to characterize the $\mathcal{Q}$-admissible quasiequations. Let $\mathcal{Q}$ and $\mathcal{Q}'$ be quasivarieties for a language $\mathcal{L}$ and let Λ be a set of $\mathcal{L}$-quasiequations. Suppose that $\mathbf{A} \in \mathcal{Q}'$ if and only if both $\mathbf{A} \in \mathcal{Q}$ and each quasiequation in Λ holds in $\mathbf{A}$. Then Λ *axiomatizes $\mathcal{Q}'$ relative to $\mathcal{Q}$*. In particular, if $\mathbb{Q}(\mathbf{F}_{\mathcal{Q}}(\omega))$

is axiomatized by Λ relative to $\mathcal{Q}$, then we call Λ a *basis* for the admissible quasiequations of $\mathcal{Q}$.

Example 3.20. *Rozière [94] (see also [95]) and Iemhoff [56] proved independently that the the set $\{V_n : n = 1, 2, \dots\}$ of quasiequations V_n forms a basis for the admissible quasiequations of the variety of Heyting algebras, where V_n is defined as*

$$(\bigwedge_{i=1}^{n}(x_i \to y_i) \to (x_{n+1} \vee x_{n+2})) \vee z \approx \top \Rightarrow \bigvee_{j=1}^{n+2}(\bigwedge_{i=1}^{n}(x_i \to y_i) \to x_j) \vee z \approx \top.$$

Since $\mathbb{Q}(\mathbf{F}_{\mathcal{Q}}(\omega)) \subseteq \mathcal{Q}$ for any quasivariety $\mathcal{Q}$, finding a basis for the admissible quasiequations of $\mathcal{Q}$ essentially involves finding a set of quasiequations that are admissible in $\mathcal{Q}$ and that axiomatize a structurally complete quasivariety relative to $\mathcal{Q}$. More precisely:

Theorem 3.21. *Let $\mathcal{Q}$ and $\mathcal{Q}'$ be $\mathcal{L}$-quasivarieties and let Λ be a set of $\mathcal{L}$-quasiequations axiomatizing $\mathcal{Q}'$ relative to $\mathcal{Q}$. Suppose that $\mathcal{Q}'$ is structurally complete and that each quasiequation in Λ is admissible in $\mathcal{Q}$. Then Λ is a basis for the $\mathcal{Q}$-admissible quasiequations.*

Proof. It suffices to show that $\mathcal{Q}' = \mathbb{Q}(\mathbf{F}_{\mathcal{Q}}(\omega))$. If each quasiequation in Λ is admissible in $\mathcal{Q}$, then by Theorem 3.9, each quasiequation in Λ holds in $\mathbf{F}_{\mathcal{Q}}(\omega)$. Hence $\mathbf{F}_{\mathcal{Q}}(\omega) \in \mathcal{Q}'$ and $\mathbb{Q}(\mathbf{F}_{\mathcal{Q}}(\omega)) \subseteq \mathcal{Q}'$. Suppose for a contradiction that $\mathbb{Q}(\mathbf{F}_{\mathcal{Q}}(\omega)) \subset \mathcal{Q}'$. Since $\mathcal{Q}'$ is structurally complete, $\mathbb{V}(\mathcal{Q}) = \mathbb{V}(\mathbf{F}_{\mathcal{Q}}(\omega)) = \mathbb{V}(\mathbb{Q}(\mathbf{F}_{\mathcal{Q}}(\omega))) \subset \mathbb{V}(\mathcal{Q}')$ by Corollaries 2.16 and 2.17. But $\mathcal{Q}' \subseteq \mathcal{Q}$, so $\mathbb{V}(\mathcal{Q}') \subseteq \mathbb{V}(\mathcal{Q})$, a contradiction. □

We now present another characterization of structural completeness. We have already seen in Theorem 3.17 that whenever each algebra of a given class $\mathcal{K}$ embeds into the free algebra $\mathbf{F}_{\mathcal{K}}(\omega)$, then $\mathcal{K}$ is structurally complete. The converse is not true in general.

Example 3.22. *Consider the four element algebra $\mathbf{P} := \langle \{a, b, c, d\}, * \rangle$ where the unary operation $*$ and the free algebras $\mathbf{F}_{\mathbf{P}}(n)$ are described by Figure 3.2.*

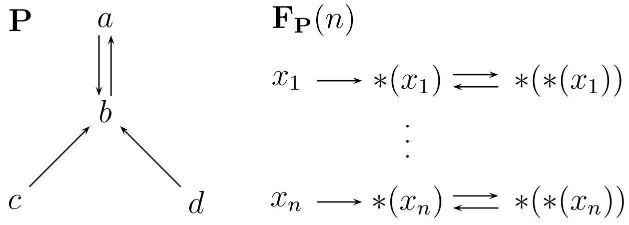


Figure 3.2: The algebra $\mathbf{P}$ and its free algebras $\mathbf{F_P}(n)$.

We calculate that $\textsc{MinGenSet}(\{\mathbf{P}\}) = \textsc{MinGenSet}(\{\mathbf{F_P}(2)\}) = \{\mathbf{F_P}(1)\}$, *where* $\mathbf{F_P}(2)$ *is the minimal generating free algebra for* $\mathbf{F_P}(\omega)$*. Hence* $\mathbb{Q}(\mathbf{P}) = \mathbb{Q}(\mathbf{F_P}(1)) = \mathbb{Q}(\mathbf{F_P}(\omega))$ *and* $\mathbf{P}$ *is structurally complete by Theorem 3.16. But* $\mathbf{P}$ *can not be embedded into* $\mathbf{F_P}(n)$ *for any* $n \in \mathbb{N}$ *since there is no element* $b' \in \mathbf{F_P}(n)$ *that is the* $*$*-image of three pairwise different other elements.*

It turns out that we have to check the embeddings for the minimal generating sets to have a nice characterization of structural completeness[4]:

Theorem 3.23. *Let* $\mathcal{K}$ *be a finite set of finite* $\mathcal{L}$*-algebras. Then the following are equivalent:*

(1) $\mathcal{K}$ *is structurally complete.*

(2) $\textsc{MinGenSet}(\mathcal{K}) \subseteq \mathbb{IS}(\mathbf{F}_\mathcal{K}(n))$ *where* $n := \max\{|C| : \mathbf{C} \in \mathcal{K}\}$.

Proof. (1) ⇒ (2) If $\mathcal{K}$ is structurally complete, then, by Theorem 3.16 and Corollary 3.12, $\mathbb{Q}(\mathcal{K}) = \mathbb{Q}(\mathbf{F}_\mathcal{K}(\omega)) = \mathbb{Q}(\mathbf{F}_\mathcal{K}(n))$ where $n := \max\{|C| : \mathbf{C} \in \mathcal{K}\}$. So $\textsc{MinGenSet}(\mathcal{K}) \subseteq \mathbb{Q}(\mathbf{F}_\mathcal{K}(n))$. But each $\mathbf{A} \in \textsc{MinGenSet}(\mathcal{K})$ is $\mathbb{Q}(\mathbf{F}_\mathcal{K}(n))$-subdirectly irreducible, so by Lemma 2.9, $\mathbf{A}$ embeds into $\mathbf{F}_\mathcal{K}(n)$. I.e., $\textsc{MinGenSet}(\mathcal{K}) \subseteq \mathbb{IS}(\mathbf{F}_\mathcal{K}(n))$.

(2) ⇒ (1) Suppose that each $\mathbf{A} \in \textsc{MinGenSet}(\mathcal{K})$ embeds into $\mathbf{F}_\mathcal{K}(n)$. Then $\mathbb{Q}(\mathbf{F}_\mathcal{K}(n)) \subseteq \mathbb{Q}(\mathcal{K}) = \mathbb{Q}(\textsc{MinGenSet}(\mathcal{K})) \subseteq \mathbb{Q}(\mathbf{F}_\mathcal{K}(n))$. So $\mathcal{K}$ is structurally complete by Theorem 3.16. □

[4] Note that Rybakov has a similar result in the context of logics possessing an analogue of the deduction theorem (see [99, Theorem 5.1.4]).

Note that we can reduce the number of generators of the free algebra in which we embed the minimal generating set of $\mathcal{K}$, using Corollaries 2.16 and 3.13:

Corollary 3.24. *Let $\mathcal{K}$ be a finite set of finite $\mathcal{L}$-algebras. Then the following are equivalent:*

(1) *$\mathcal{K}$ is structurally complete.*

(2) $\textsc{MinGenSet}(\mathcal{K}) \subseteq \mathbb{IS}(\mathbf{F}_{\mathcal{K}}(n))$ *where n is the smallest natural number such that* $\textsc{MinGenSet}(\mathcal{K}) \subseteq \mathbb{H}(\mathbf{F}_{\mathcal{K}}(n))$.

3.5 Almost Structural Completeness

For certain classes, admissibility and validity coincide for quasiequations with unifiable premises. More precisely, we call a class $\mathcal{K}$ of $\mathcal{L}$-algebras *almost structurally complete* if it satisfies the condition:

$$\Sigma \Rightarrow \varphi \approx \psi \text{ is } \mathcal{K}\text{-admissible} \quad \text{iff} \quad \Sigma \models_{\mathcal{K}} \varphi \approx \psi \ \text{ or } \ \Sigma \text{ is not } \mathcal{K}\text{-unifiable.}$$

Theorem 3.25. *Let $\mathcal{K}$ be a class of $\mathcal{L}$-algebras and $\mathbf{B} \in \mathbb{S}(\mathbf{F}_{\mathcal{K}}(\omega))$. Then the following are equivalent:*

(1) *$\mathcal{K}$ is almost structurally complete.*

(2) $\mathbb{Q}(\{\mathbf{A} \times \mathbf{B} : \mathbf{A} \in \mathcal{K}\}) = \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$.

(3) $\{\mathbf{A} \times \mathbf{B} : \mathbf{A} \in \mathcal{K}\} \subseteq \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$.

Proof. (1) $\Rightarrow$ (2) Suppose that $\mathcal{K}$ is almost structurally complete. To establish $\mathbb{Q}(\{\mathbf{A}\times\mathbf{B} : \mathbf{A} \in \mathcal{K}\}) = \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$, it suffices to show that a quasiequation $\Sigma \Rightarrow \varphi \approx \psi$ is valid in all algebras $\mathbf{A} \times \mathbf{B}$ for $\mathbf{A} \in \mathcal{K}$ iff it is valid in $\mathbf{F}_{\mathcal{K}}(\omega)$. Suppose first that $\Sigma \models_{\mathbf{F}_{\mathcal{K}}(\omega)} \varphi \approx \psi$. Then by Theorem 3.9, either Σ is not $\mathcal{K}$-unifiable or $\Sigma \Rightarrow \varphi \approx \psi$ is $\mathcal{K}$-valid. In the first case, by Theorem 3.7, Σ is not $\mathbf{B}$-satisfiable, so $\Sigma \Rightarrow \varphi \approx \psi$ is valid in $\mathbf{A} \times \mathbf{B}$ for all $\mathbf{A} \in \mathcal{K}$. In the second case, $\Sigma \Rightarrow \varphi \approx \psi$ is valid in $\mathbf{A} \times \mathbf{B} \in \mathbb{Q}(\mathcal{K})$ for all $\mathbf{A} \in \mathcal{K}$.

Conversely, if $\Sigma \Rightarrow \varphi \approx \psi$ is valid in $\mathbf{A} \times \mathbf{B}$ for each $\mathbf{A} \in \mathcal{K}$, then either Σ is not $\mathbf{B}$-satisfiable or $\Sigma \Rightarrow \varphi \approx \psi$ is valid in each $\mathbf{A}$ in $\mathcal{K}$. In the first case, by Theorem 3.7, Σ is not $\mathcal{K}$-unifiable, so $\Sigma \Rightarrow \varphi \approx \psi$ is valid in $\mathbf{F}_{\mathcal{K}}(\omega)$. In the second case, $\Sigma \Rightarrow \varphi \approx \psi$ is valid in $\mathbb{Q}(\mathcal{K})$ and hence valid in $\mathbf{F}_{\mathcal{K}}(\omega)$.

(2) $\Rightarrow$ (1) Suppose that $\mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega)) = \mathbb{Q}(\{\mathbf{A} \times \mathbf{B} : \mathbf{A} \in \mathcal{K}\})$. Then whenever $\Sigma \Rightarrow \varphi \approx \psi$ is $\mathcal{K}$-admissible, it is $\mathbf{F}_{\mathcal{K}}(\omega)$-valid and hence also valid in $\mathbf{A} \times \mathbf{B}$ for all $\mathbf{A} \in \mathcal{K}$. Moreover, if Σ is $\mathcal{K}$-unifiable, then, by Theorem 3.7, it is $\mathbf{B}$-satisfiable. I.e., there exists a homomorphism $h \colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{B}$ with $\Sigma \subseteq \ker h$. For any $\mathbf{A} \in \mathcal{K}$ and homomorphism $k \colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{A}$ with $\Sigma \subseteq \ker k$, define $e_{\mathbf{A}} \colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{A} \times \mathbf{B}$ by $e_{\mathbf{A}}(u) := (k(u), h(u))$. Then, since $\Sigma \Rightarrow \varphi \approx \psi$ is valid in $\mathbf{A} \times \mathbf{B}$ for all $\mathbf{A} \in \mathcal{K}$, $\Sigma \subseteq \ker e$, so $e(\varphi) = e(\psi)$ and $k(\varphi) = k(\psi)$. I.e., $\Sigma \models_{\mathbf{A}} \varphi \approx \psi$. So we have shown that $\Sigma \models_{\mathcal{K}} \varphi \approx \psi$.

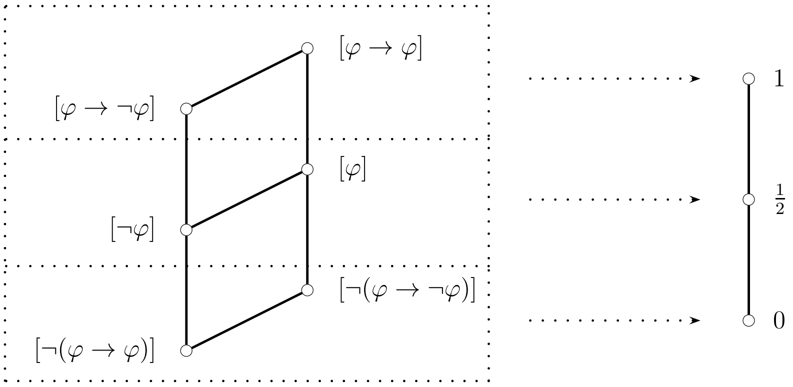
(2) $\Rightarrow$ (3) Immediate.

(3) $\Rightarrow$ (2) Suppose that $\{\mathbf{A} \times \mathbf{B} : \mathbf{A} \in \mathcal{K}\} \subseteq \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$. Then also, since $\mathbf{A} \in \mathbb{H}(\mathbf{A} \times \mathbf{B})$ for each $\mathbf{A} \in \mathcal{K}$, we obtain $\mathcal{K} \subseteq \mathbb{V}(\{\mathbf{A} \times \mathbf{B} : \mathbf{A} \in \mathcal{K}\})$. Hence by Theorem 3.11, $\mathbb{Q}(\{\mathbf{A} \times \mathbf{B} : \mathbf{A} \in \mathcal{K}\}) = \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$. □

Example 3.26. *Consider the Wajsberg algebras with two and three elements* $\mathbf{L_2} := \langle \{0, 1\}, \to, \neg \rangle$ *and* $\mathbf{L_3} := \langle \{0, \frac{1}{2}, 1\}, \to, \neg \rangle$ *where*

$$x \to y \ := \ \min(1, 1 - x + y) \quad \textit{and} \quad \neg x \ := \ 1 - x.$$

$\mathbf{L_2}$ *embeds into* $\mathbf{F}_{\mathbf{L_3}}(\omega)$ *via* $0 \mapsto [\neg(x \to x)]$, $1 \mapsto [x \to x]$ *and hence is (isomorphic to) a subalgebra of* $\mathbf{F}_{\mathbf{L_3}}(\omega)$. *The algebra* $\mathbf{L_3} \times \mathbf{L_2}$ *embeds into* $\mathbf{F}_{\mathbf{L_3}}(\omega)$, *as illustrated in the diagram below by the terms associated to elements, and has* $\mathbf{L_3}$ *as a homomorphic image, as indicated by the arrows. Hence by Corollary 3.13 and Theorem 3.25,* $\mathbf{L_3}$ *is almost structurally complete. However, it is not structurally complete since, e.g.,* $x \approx \neg x \Rightarrow x \approx y$ *is* $\mathbf{L_3}$*-admissible, but not* $\mathbf{L_3}$*-valid. On the other hand, its implicational reduct* $\mathbf{L}_{\mathbf{3}}^{\to} := \langle \{0, \frac{1}{2}, 1\}, \to \rangle$ *is structurally complete, since it embeds into* $\mathbf{F}_{\mathbf{L}_{\mathbf{3}}^{\to}}(2)$ *(see Theorem 3.17).*



$[\varphi \to \varphi]$
$[\varphi \to \neg\varphi]$
$[\varphi]$
$[\neg\varphi]$
$[\neg(\varphi \to \neg\varphi)]$
$[\neg(\varphi \to \varphi)]$
1
$\frac{1}{2}$
0

$\varphi := (x \to \neg x) \to \neg x$

We now are able to prove a characterization for almost structural completeness similar to Theorem 3.23:

Theorem 3.27. *Let $\mathcal{K}$ be a finite set of finite $\mathcal{L}$-algebras, $\mathbf{B} \in \mathbb{S}(\mathbf{F}_{\mathcal{K}}(\omega))$ and $n := \max\{|C| : \mathbf{C} \in \mathcal{K}\}$. Then the following are equivalent:*

(1) *$\mathcal{K}$ is almost structurally complete.*

(2) $\textsc{MinGenSet}(\{\mathbf{A} \times \mathbf{B} : \mathbf{A} \in \mathcal{K}\}) \subseteq \mathbb{IS}(\mathbf{F}_{\mathcal{K}}(n))$.

Proof. (1) ⇒ (2) If $\mathcal{K}$ is almost structurally complete, then by Theorem 3.25 and Corollary 3.12, $\mathbb{Q}(\{\mathbf{A} \times \mathbf{B} : \mathbf{A} \in \mathcal{K}\}) = \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega)) = \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(n))$ where $n := \max\{|C| : \mathbf{C} \in \mathcal{K}\}$. In particular, $\textsc{MinGenSet}(\{\mathbf{A} \times \mathbf{B} : \mathbf{A} \in \mathcal{K}\}) \subseteq \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(n))$. But each $\mathbf{C} \in \textsc{MinGenSet}(\{\mathbf{A} \times \mathbf{B} : \mathbf{A} \in \mathcal{K}\})$ is $\mathbb{Q}(\mathbf{F}_{\mathcal{K}}(n))$-subdirectly irreducible, so by Lemma 2.9, $\mathbf{C}$ embeds into $\mathbf{F}_{\mathcal{K}}(n)$. I.e., $\textsc{MinGenSet}(\{\mathbf{A} \times \mathbf{B} : \mathbf{A} \in \mathcal{K}\}) \subseteq \mathbb{IS}(\mathbf{F}_{\mathcal{K}}(n))$.

(2) ⇒ (1) If $\textsc{MinGenSet}(\{\mathbf{A} \times \mathbf{B} : \mathbf{A} \in \mathcal{K}\}) \subseteq \mathbb{IS}(\mathbf{F}_{\mathcal{K}}(n))$, then $\{\mathbf{A} \times \mathbf{B} : \mathbf{A} \in \mathcal{K}\} \subseteq \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(n)) = \mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$. So by Theorem 3.25, $\mathcal{K}$ is almost structurally complete. □

3.6 Clone Equivalences

This section makes a useful observation regarding clone equivalent algebras: There is no need to calculate free algebras, minimal generating sets or the property of structural completeness twice, if the operations of two algebras on the same universe are inter-definable. However, checking whether two finite algebras are clone equivalent is EXPTIME-complete (see [11]).

Recall from Section 2.1 that clones of operations are defined on a fixed universe and hence two clone equivalent algebras $\mathbf{A}$ and $\mathbf{B}$ are isomorphic in the language $\mathcal{L} = \operatorname{Clo}\mathbf{A} = \operatorname{Clo}\mathbf{B}$. The next theorem states that the free algebras and the minimal generating sets of the quasivarieties generated by the free algebras on countably infinitely many generators are clone equivalent for clone equivalent algebras. Hence if we calculated the minimal generating free algebra $\mathbf{F_A}(n)$ for $\mathbb{Q}(\mathbf{F_A}(\omega))$, we only need to translate the operations from the language of $\mathbf{A}$ into the language of $\mathbf{B}$ to get the minimal generating free algebra $\mathbf{F_B}(n)$ for $\mathbb{Q}(\mathbf{F_B}(\omega))$.

Theorem 3.28. *Let $\mathbf{A}$ and $\mathbf{B}$ be two clone equivalent finite algebras.*

(a) $\mathbf{F_A}(n) \approx_{clo} \mathbf{F_B}(n)$ *for all natural numbers $n \geq m$, where m is the maximal arity of the operations on $\mathbf{A}$ and $\mathbf{B}$.*

(b) *Any member of a minimal generating set for $\mathbb{Q}(\mathbf{F_A}(\omega))$ has exactly one clone equivalent member in a minimal generating set for $\mathbb{Q}(\mathbf{F_B}(\omega))$.*

(c) $\mathbf{A}$ *is structurally complete iff* $\mathbf{B}$ *is structurally complete.*

(d) $\mathbf{A}$ *is almost structurally complete iff* $\mathbf{B}$ *is almost structurally complete.*

Proof. (a) $\operatorname{Clo_n}\mathbf{A} = \operatorname{Clo_n}\mathbf{B}$ for any n greater than the maximal arity of the operaions on $\mathbf{A}$ and $\mathbf{B}$ by the assumption, so $\mathbf{F_A}(n) \approx_{clo} \mathbf{F_B}(n)$ follows directly from $\mathbf{F_A}(k) \cong \mathbf{Clo_k}\,\mathbf{A}$ for any $k \in \mathbb{N}$ (see [10, Exercise 4.34.3]), where $\mathbf{Clo_n}\,\mathbf{A}$ is the algebra with universe $\operatorname{Clo_n}\mathbf{A}$ and the natural induced operations.

(b) follows from (a) since $\mathbf{F_A}(n) \cong \mathbf{F_B}(n)$ in $\mathcal{L} = \operatorname{Clo_n}\mathbf{A}$.

(c), (d) then follow directly from (a),(b) using Theorems 3.23 and 3.27. □

3.7 Finite-Valued Logics

For algebraizable logics, admissible rules may be translated into admissible quasiequations and vice versa (see [21]). The characterizations of admissibility we have seen in the preceding sections can be adapted to finite-valued logics. Unlike the algebraic case we have to treat here the designated values of the logic, i.e., the truth values considered true. Here we describe a method that given a finite-valued logic L, provides another (hopefully small) finite-valued logic L$'$ such that validity in L$'$ corresponds to admissibility in L. The more general case, where we search for a smallest finite set of logics such that validity in all members of the set corresponds to admissibility in a logic (or logics), will not be considered here.

Recall that a *finite-valued logic* $\mathrm{L} := (\mathbf{A}, D)$ for a language $\mathcal{L}$ consists of a finite $\mathcal{L}$-algebra $\mathbf{A}$ and a set of *designated values* $D \subseteq A$. Given $\Gamma \cup \{\varphi\} \subseteq \mathrm{Tm}_{\mathcal{L}}$, we let $\Gamma \vdash_{\mathrm{L}} \varphi$ denote that for all homomorphisms $h\colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{A}$, whenever $h[\Gamma] \subseteq D$, also $h(\varphi) \in D$. A term φ is L-*valid* if $\vdash_{\mathrm{L}} \varphi$.

Consider now a finite-valued logic $\mathrm{L} := (\mathbf{A}, D)$ for a language $\mathcal{L}$ and a finite set of terms $\Gamma \subseteq \mathrm{Tm}_{\mathcal{L}}$. We say that Γ is L-*unifiable* if there exists a homomorphism $\sigma\colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{Tm}_{\mathcal{L}}$ such that $\vdash_{\mathrm{L}} \sigma(\psi)$ for all $\psi \in \Gamma$ and call σ in this case an L-*unifier* of Γ. A *rule* is a pair $\langle \Gamma, \varphi \rangle$, $\Gamma \cup \{\varphi\} \subseteq \mathrm{Tm}_{\mathcal{L}}$ finite, where the elements of Γ are called the *premises* and φ the *conclusion* of the rule. The pair $\langle \{\sigma(\varphi_1), \dots, \sigma(\varphi_n)\}, \sigma(\varphi) \rangle$ is called an *instance* of the rule $\langle \{\varphi_1, \dots, \varphi_n\}, \varphi \rangle$, where σ is a substitution on $\mathbf{Tm}_{\mathcal{L}}$. A rule $\langle \{\varphi_1, \dots, \varphi_n\}, \varphi \rangle$ named $\circledast$ is usually written as

$$\varphi_1, \dots, \varphi_n \;/\; \varphi \qquad \text{or} \qquad \frac{\varphi_1, \dots, \varphi_n}{\varphi} \circledast.$$

A rule $\Gamma \;/\; \varphi$ is said to be L-*admissible* if every L-unifier of Γ is an L-unifier of φ. Note that if L is an algebraizable logic (see [21]) with equivalent quasivariety $\mathcal{Q}$ and translations E and Δ, then the rule $\Gamma \;/\; \varphi$ is L-admissible if and only if the quasiequation $E[\Gamma] \Rightarrow E(\varphi)$ is $\mathcal{Q}$-admissible. Now if we define the finite-valued logic $\mathrm{L}^* := (\mathbf{F_A}(|A|), D^*)$ where $D^* := \{[\varphi] \in \mathbf{F_A}(|A|) : \vdash_{\mathrm{L}}$

$\varphi\}$, then we obtain the following analogue of Theorem 3.9.

Theorem 3.29. *Let* L $:= (\mathbf{A}, D)$ *be a finite-valued logic for a language* $\mathcal{L}$*. Then* $\Gamma \;/\; \varphi$ *is* L*-admissible iff* $\Gamma \vdash_{\mathrm{L}^*} \varphi$*.*

Proof. ($\Rightarrow$) Suppose that $\Gamma \;/\; \varphi$ is L-admissible and let $h \colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{F}_{\mathbf{A}}(|A|)$ be a homomorphism such that $h[\Gamma] \subseteq D^*$. We define a map σ that sends each variable x to a member of the equivalence class $h(x)$. By the universal mapping property of $\mathbf{Tm}_{\mathcal{L}}$, this extends to a homomorphism $\sigma \colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{Tm}_{\mathcal{L}}$. But since $\nu(\sigma(x)) = h(x)$ for each variable x (ν is the natural homomorphism for the congruence $\Psi_{\mathbf{A}}(|A|)$), we obtain $\nu \circ \sigma = h$. So for each $\psi \in \Gamma$, we have $\nu(\sigma(\psi)) \in D^*$ and therefore $\vdash_{\mathrm{L}} \sigma(\psi)$. Hence by assumption, $\vdash_{\mathrm{L}} \sigma(\varphi)$, and $h(\varphi) = \nu(\sigma(\varphi)) \in D^*$ as required.

($\Leftarrow$) Suppose that $\Gamma \vdash_{\mathrm{L}^*} \varphi$ and let $\sigma \colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{Tm}_{\mathcal{L}}$ be a unifier of Γ, i.e., $\vdash_{\mathrm{L}} \sigma(\psi)$ for all $\psi \in \Gamma$ and hence $\nu(\sigma(\psi)) \in D^*$. By assumption, $\nu(\sigma(\varphi)) \in D^*$. Hence $\vdash_{\mathrm{L}} \sigma(\varphi)$ as required. □

The next result may then be understood as an analogue of Theorem 3.11.

Theorem 3.30. *Let* L $:= (\mathbf{A}, D_A)$ *and* L$' := (\mathbf{B}, D_B)$ *be finite-valued logics for a language* $\mathcal{L}$ *such that* $\mathbf{B}$ *is a subalgebra of* $\mathbf{F}_{\mathbf{A}}(|A|)$*,* $D_B = D_A^* \cap B$ *and there exists a surjective homomorphism* $h \colon \mathbf{B} \to \mathbf{A}$ *satisfying* $h[D_B] \subseteq D_A$*. Then* $\Gamma \;/\; \varphi$ *is* L*-admissible iff* $\Gamma \vdash_{\mathrm{L}'} \varphi$*.*

Proof. If $\Gamma \;/\; \varphi$ is L-admissible, then by Lemma 3.29, $\Gamma \vdash_{\mathrm{L}^*} \varphi$. Since $\mathbf{B} \leq \mathbf{F}_{\mathbf{A}}(|A|)$ and $D_B = D_A^* \cap B$, also $\Gamma \vdash_{\mathrm{L}'} \varphi$. Conversely, suppose that $\Gamma \vdash_{\mathrm{L}'} \varphi$ and that σ is an L-unifier of Γ. Notice that if $\vdash_{\mathrm{L}} \psi$, then $\vdash_{\mathrm{L}^*} \psi$ and $\vdash_{\mathrm{L}'} \psi$. So σ is also an L*-unifier and L$'$-unifier of Γ. But $\sigma(\Gamma) \vdash_{\mathrm{L}'} \sigma(\varphi)$ and therefore $\vdash_{\mathrm{L}'} \sigma(\varphi)$. Now consider any homomorphism $e \colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{A}$. Since h is a surjective homomorphism from $\mathbf{B}$ to $\mathbf{A}$, there exists a homomorphism $k \colon \mathbf{A} \to \mathbf{B}$ such that $h \circ k$ is the identity map on $\mathbf{A}$. But $\vdash_{\mathrm{L}'} \sigma(\varphi)$ and hence $k \circ e \circ \sigma(\varphi) \in D_B$. Therefore $e \circ \sigma(\varphi) = h \circ k \circ e \circ \sigma(\varphi) \in h[D_B] \subseteq D_A$. So $\vdash_{\mathrm{L}} \sigma(\varphi)$. □

Example 3.31. *The three-valued Łukasiewicz logic* Ł$_3$ *and Jaśkowski logic* J$_3$ *may both be presented using the three element Wajsberg algebra* $\mathbf{L_3}$ *(Example 3.26) but with* 1 *as designated value for* Ł$_3$ *and* $\frac{1}{2}$ *and* 1 *as designated values for* J$_3$*. That is,* Ł$_3 := (\mathbf{L_3}, \{1\})$ *and* J$_3 := (\mathbf{L_3}, \{\frac{1}{2}, 1\})$*. In this case, there is a smallest subalgebra of* $\mathbf{F_{L_3}}(\omega)$ *isomorphic to* $\mathbf{L_3} \times \mathbf{L_2}$ *with a surjective homomorphism that maps* $\mathbf{L_3} \times \mathbf{L_2}$ *onto* $\mathbf{L_3}$ *and sends the inherited designated values* $(1, 1)$ *to* 1 *and* $(\frac{1}{2}, 1)$ *to* $\frac{1}{2}$*. We therefore obtain a logic* $(\mathbf{L_3} \times \mathbf{L_2}, \{(1, 1)\})$ *corresponding to admissibility in* Ł$_3$*, and another logic* $(\mathbf{L_3} \times \mathbf{L_2}, \{(\frac{1}{2}, 1), (1, 1)\})$ *corresponding to admissibility in* J$_3$*.*

3.8 Automatically Generated Proof Systems

Here we show how proof systems for admissibility and validity can be generated using the system MUltlog [101]. We first give a brief overview of the most important definitions; please refer to [110] for a detailed introduction.

Let L $:= (\mathbf{Alg}, D)$ be an n-valued logic for a language $\mathcal{L}$. A *sequent* Γ of L is an n-tuple $\Gamma_{a_1} \mid \ldots \mid \Gamma_{a_n}$ of finite sequences Γ_{a_i} of $\mathcal{L}$-terms, where $Alg := \{a_1, \ldots, a_n\}$. The Γ_{a_i} are called the components of Γ. Let $h\colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{Alg}$ be a homomorphism (also called an *interpretation*). h *satisfies* a sequent Γ if there is an $a \in A$ such that $h(\varphi) = a$ for some $\mathcal{L}$-term $\varphi \in \Gamma_a$. In this case, h is called a *model of* Γ, written $h \models \Gamma$. Γ is called *satisfiable* if there is an interpretation h such that $h \models \Gamma$ and *valid* if for every interpretation h, $h \models \Gamma$. The *sequent calculus* $\mathcal{SC}_{\text{L}}$ for the logic L is given by the following rules:

- an *axiom* for every $\mathcal{L}$-term φ:

$$\frac{}{\varphi \mid \ldots \mid \varphi}\ ax_\varphi$$

- *weakening rules* for every truth value a_k:

$$\frac{\Gamma_1 \mid \ldots \mid \Gamma_n}{\Gamma_1 \mid \ldots \mid \Gamma_k, \varphi \mid \ldots \mid \Gamma_n}\ weak_{a_k}$$

- *exchange rules* for every truth value a_k:

$$\frac{\Gamma_1 \mid \ldots \mid \Gamma_k, \varphi, \psi, \Delta_k \mid \ldots \mid \Gamma_n}{\Gamma_1 \mid \ldots \mid \Gamma_k, \psi, \varphi, \Delta_k \mid \ldots \mid \Gamma_n}\, exch_{a_k}$$

- *contraction rules* for every truth value a_k:

$$\frac{\Gamma_1 \mid \ldots \mid \Gamma_k, \varphi, \varphi \mid \ldots \mid \Gamma_n}{\Gamma_1 \mid \ldots \mid \Gamma_k, \varphi \mid \ldots \mid \Gamma_n}\, cont_{a_k}$$

- *cut rules* for every two truth values $a_k \neq a_l$:

$$\frac{\Gamma_1 \mid \ldots \mid \Gamma_k, \varphi \mid \ldots \mid \Gamma_n \quad \Delta_1 \mid \ldots \mid \Delta_l, \varphi \mid \ldots \mid \Delta_n}{\Gamma_1, \Delta_1 \mid \ldots \mid \Gamma_n, \Delta_n}\, cut_{a_k a_l}$$

- an *introduction rule*[5] $*_{a_k}$ for every connective $*$ and truth value a_k.

A finite tree P of sequents is called a *proof in the sequent calculus* $\mathcal{SC}_{\mathrm{L}}$ if every leaf is an axiom of $\mathcal{SC}$, and all other sequents are obtained from their children by applying one of the rules of $\mathcal{SC}$. The sequent at the root of P is called its *end-sequent*. A sequent Γ is called *provable* in $\mathcal{SC}$, written $\vdash_{\mathcal{SC}} \Gamma$, if it is the end-sequent of some proof in $\mathcal{SC}$. Soundness, completeness and cut-elimination for $\mathcal{SC}$ are proved in [110].

Note that the choice of designated values for the logic L does not affect the structure of the rules of the sequent calculus $\mathcal{SC}$. Also, if we want to check whether an $\mathcal{L}$-equation or $\mathcal{L}$-quasiequation is valid in L, the choice of designated values does not change anything.

Let us now consider the three element algebra $\mathbf{G}_9 := \langle \{0,1,2\}, * \rangle$ with the binary operation $*$ where $x * y := 2$ when $x = 2$ and $y \in \{1,2\}$, $x * y := 0$ otherwise (see also Appendix A). We input this information to the tool MUltlog (see Figure 3.3) which then outputs, amongst many other things, the introduction rules for the operation $*$ (see Figure 3.4). Intuitively, the

[5] We leave out a proper explanation of the construction of these logical rules here, but will present the concrete introduction rules in the upcoming examples.

```
logic "G9".
truth_values { 0 , 1 , 2 }.
designated_truth_values { 2 }.
operator( ast/2, table [
      0, 1, 2,
  0,  0, 0, 0,
  1,  0, 0, 0,
  2,  0, 2, 2
]
).
```

Figure 3.3: Input file *G9.lgc* for the system MUltlog.

rule $*_0$ of Figure 3.4 expresses the fact that $\varphi * \psi$ takes value 0 under some interpretation $h\colon \mathbf{Tm}_{\mathcal{L}} \to \mathbf{G}_9$ whenever $h(\varphi) = 0$ or $h(\psi) = 0$ or $h(\varphi) = 1$. I.e., the stroke "|" denotes "or" between different values of the underlying logic while the comma "," denotes "or" between different formulas of the sequents. Together with the structural rules explained above they build the proof system $\mathcal{SC}_{\mathbf{G_9}}$ to check validity in $\mathbf{G_9}$.

$$\frac{\Gamma_1, \varphi, \psi \mid \Gamma_2, \varphi \mid \Gamma_3}{\Gamma_1, \varphi * \psi \mid \Gamma_2 \mid \Gamma_3}\ *_0$$

$$\frac{\Gamma_1 \mid \Gamma_2 \mid \Gamma_3}{\Gamma_1 \mid \Gamma_2, \varphi * \psi \mid \Gamma_3}\ *_1$$

$$\frac{\Gamma_1 \mid \Gamma_2, \psi \mid \Gamma_3, \psi \quad \Gamma_1 \mid \Gamma_2 \mid \Gamma_3, \varphi}{\Gamma_1 \mid \Gamma_2 \mid \Gamma_3, \varphi * \psi}\ *_2$$

Figure 3.4: The introduction rules for the operation $*$ of $\mathbf{G_9}$.

Using the tool MUltseq (see [47]), the companion of MUltlog, we can check whether the following quasiequation holds in $\mathbf{G_9}$ (we write x^2 to denote $(x*x)$ for convenience):

$$x^2 \approx y * x^2,\ \ x * y \approx y * x^2,\ \ y * x \approx y^2 \ \ \Rightarrow\ \ y * x \approx y * x^2. \tag{3.1}$$

The output of MUltseq tells us that proving (3.1) is equivalent to proving the following sequents[6], which is not possible and hence (3.1) does not hold in $\mathbf{G_9}$. MUltseq even provides a counterexample: $x = 1$, $y = 2$.

$$\begin{array}{l}
y * x,\ y^2 \mid x^2,\ x * y,\ y * x^2 \mid x^2,\ x * y,\ y * x,\ y^2,\ y * x^2 \\
y * x,\ y^2 \mid x^2,\ x * y,\ y * x,\ y^2,\ y * x^2 \mid x^2,\ x * y,\ y * x^2 \\
x^2,\ x * y,\ y * x^2 \mid y * x,\ y^2 \mid x^2,\ x * y,\ y * x,\ y^2,\ y * x^2 \\
x^2,\ x * y,\ y * x,\ y^2,\ y * x^2 \mid y * x,\ y^2 \mid x^2,\ x * y,\ y * x^2 \\
x^2,\ x * y,\ y * x^2 \mid x^2,\ x * y,\ y * x,\ y^2,\ y * x^2 \mid y * x,\ y^2 \\
x^2,\ x * y,\ y * x,\ y^2,\ y * x^2 \mid x^2,\ x * y,\ y * x^2 \mid y * x,\ y^2
\end{array}$$

Using TAFA (see Chapter 5) we calculate the minimal generating free algebra for $\mathbb{Q}(\mathbf{F}_{\mathbf{G_9}}(\omega))$ which has two generators and seven elements. Calculating $\textsc{MinGenSet}(\mathbf{F}_{\mathbf{G_9}}(2))$ returns $\mathbf{AdmG_9} := \langle \{a, b, c, d\}, *\rangle$ with

$*$	a	b	c	d
a	c	d	c	d
b	b	b	d	d
c	c	d	c	d
d	d	d	d	d

MUltlog then calculates the introduction rules for the operation $*$ of the algebra $\mathbf{AdmG_9}$ (see Figure 3.5). Running MUltseq with the input for $\mathbf{AdmG_9}$ (see Figure 3.6) confirms that the quasiequation (3.1) is provable in $\mathcal{SC}_{\mathbf{AdmG_9}}$ and hence is $\mathbf{G_9}$-admissible. It is also possible to output proof trees of specific sequents. In this case (here with one out of twelve sequents to check for the proof of (3.1)) MUltseq then also outputs a skeleton of the proof as follows[7] ($(\varphi)^{a_i}$ means that the term φ stands in the i-th position of the sequent):

Proof skeleton of $[(x * x)^a, (x * x)^c, (x * x)^d, (x * y)^a, (x * y)^c, (x * y)^d, (y *$

[6] It is not hard to see that checking the validity of an equation $\varphi \approx \psi$ for, e.g., a three-valued algebra, is equivalent to checking the validity of the three sequents $\varphi \mid \psi \mid \psi$, $\psi \mid \varphi \mid \psi$ and $\psi \mid \psi \mid \varphi$. This idea is then extended combinatorially to quasiequations.

[7] The right upper side of the proof tree (which is obviously equal to the left part) is abbreviated here because of the space.

$$\frac{\Gamma_1 \mid \Gamma_2 \mid \Gamma_3 \mid \Gamma_4}{\Gamma_1, \varphi * \psi \mid \Gamma_2 \mid \Gamma_3 \mid \Gamma_4}\ *_a$$

$$\frac{\Gamma_1, \psi \mid \Gamma_2, \psi \mid \Gamma_3 \mid \Gamma_4 \quad \Gamma_1 \mid \Gamma_2, \varphi \mid \Gamma_3 \mid \Gamma_4}{\Gamma_1 \mid \Gamma_2, \varphi * \psi \mid \Gamma_3 \mid \Gamma_4}\ *_b$$

$$\frac{\Gamma_1, \psi \mid \Gamma_2 \mid \Gamma_3, \psi \mid \Gamma_4 \quad \Gamma_1, \varphi \mid \Gamma_2 \mid \Gamma_3, \varphi \mid \Gamma_4}{\Gamma_1 \mid \Gamma_2 \mid \Gamma_3, \varphi * \psi \mid \Gamma_4}\ *_c$$

$$\frac{\Gamma_1 \mid \Gamma_2, \varphi, \psi \mid \Gamma_3 \mid \Gamma_4, \varphi, \psi \quad \Gamma_1, \varphi \mid \Gamma_2 \mid \Gamma_3, \varphi, \psi \mid \Gamma_4, \varphi, \psi}{\Gamma_1 \mid \Gamma_2 \mid \Gamma_3 \mid \Gamma_4, \varphi * \psi}\ *_d$$

Figure 3.5: The introduction rules for the operation $*$ of **AdmG₉**.

$x)^a, (y{*}x)^b, (y{*}x)^c, (y{*}y)^a, (y{*}y)^b, (y{*}y)^c, (y{*}(x{*}x))^a, (y{*}(x{*}x))^c, (y{*}(x{*}x))^d]$:

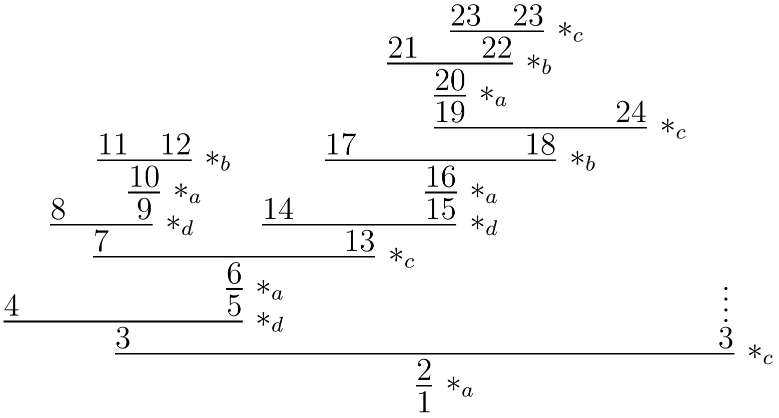


Table of sequents[8]:

1. $[(x*x)^a, (x*x)^c, (x*x)^d, (x*y)^a, (x*y)^c, (x*y)^d, (y*x)^a, (y*x)^b, (y*x)^c, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

2. $[(x*x)^c, (x*x)^d, (x*y)^a, (x*y)^c, (x*y)^d, (y*x)^a, (y*x)^b, (y*x)^c, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

3. $[x^a, x^c, (x*x)^d, (x*y)^a, (x*y)^c, (x*y)^d, (y*x)^a, (y*x)^b, (y*x)^c, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

[8] Note that we can apply axiom rules to the leaves of the proof tree, e.g., ax_x to no. 4.

```
% admgnine - specification of a sequent calculus for AdmG9

option(tex_rulenames(on)).
truth_values([a,b,c,d]).
designated_truth_values([d]).

% Definition of the operation ast.
tex_op((A*B), ["(", A, bslash, "ast ", B, ")"]).
rule((A*B)^a, [[]], asta).
rule((A*B)^b, [[B^a,B^b],[A^b]], astb).
rule((A*B)^c, [[B^a,B^c],[A^a,A^c]], astc).
rule((A*B)^d, [[A^b,A^d,B^b,B^d],[A^a,A^c,A^d,B^c,B^d]], astd).
tex_rn(asta, ["{", bslash, "ast_a}"]).
tex_rn(astb, ["{", bslash, "ast_b}"]).
tex_rn(astc, ["{", bslash, "ast_c}"]).
tex_rn(astd, ["{", bslash, "ast_d}"]).

% Test the derivability of a sequent
ts(s1, [(a*a)^a,(a*a)^c,(a*a)^d,(a*b)^a,(a*b)^c,(a*b)^d,
(b*a)^a,(b*a)^b,(b*a)^c,(b*b)^a,(b*b)^b,(b*b)^c,(b*(a*a))^a,
(b*(a*a))^c,(b*(a*a))^d]).

% Test the validity of a quasiequation
tqe(qe1, [a*a=b*(a*a),a*b=b*(a*a),b*a=b*b], b*a=b*(a*a)).

tex_opname(a, ["x"]).
tex_opname(b, ["y"]).
```

Figure 3.6: Input file *admgnine.lgc* for the system MUltseq.

4. $[x^a, x^b, x^c, x^d, (x*y)^a, (x*y)^c, (x*y)^d, (y*x)^a, (y*x)^b, (y*x)^c, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

5. $[x^a, x^c, x^d, (x*y)^a, (x*y)^c, (x*y)^d, (y*x)^a, (y*x)^b, (y*x)^c, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

6. $[x^a, x^c, x^d, (x*y)^c, (x*y)^d, (y*x)^a, (y*x)^b, (y*x)^c, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

7. $[x^a, x^c, x^d, y^a, y^c, (x*y)^d, (y*x)^a, (y*x)^b, (y*x)^c, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

8. $[x^a, x^b, x^c, x^d, y^a, y^b, y^c, y^d, (y*x)^a, (y*x)^b, (y*x)^c, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

9. $[x^a, x^c, x^d, y^a, y^c, y^d, (y*x)^a, (y*x)^b, (y*x)^c, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

10. $[x^a, x^c, x^d, y^a, y^c, y^d, (y*x)^b, (y*x)^c, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

11. $[x^a, x^b, x^c, x^d, y^a, y^c, y^d, (y*x)^c, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

12. $[x^a, x^c, x^d, y^a, y^b, y^c, y^d, (y*x)^c, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

13. $[x^a, x^c, x^d, (x*y)^d, (y*x)^a, (y*x)^b, (y*x)^c, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

14. $[x^a, x^b, x^c, x^d, y^b, y^d, (y*x)^a, (y*x)^b, (y*x)^c, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

15. $[x^a, x^c, x^d, y^c, y^d, (y*x)^a, (y*x)^b, (y*x)^c, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

16. $[x^a, x^c, x^d, y^c, y^d, (y*x)^b, (y*x)^c, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

17. $[x^a, x^b, x^c, x^d, y^c, y^d, (y*x)^c, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

18. $[x^a, x^c, x^d, y^b, y^c, y^d, (y*x)^c, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

19. $[x^a, x^c, x^d, y^b, y^c, y^d, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

20. $[x^a, x^c, x^d, y^b, y^c, y^d, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

21. $[x^a, x^c, x^d, y^a, y^b, y^c, y^d, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

22. $[x^a, x^c, x^d, y^b, y^c, y^d, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

23. $[x^a, x^c, x^d, y^a, y^b, y^c, y^d, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

24. $[x^a, x^c, x^d, y^a, y^b, y^c, y^d, (y*y)^a, (y*y)^b, (y*y)^c, (y*(x*x))^a, (y*(x*x))^c, (y*(x*x))^d]$

Chapter 4

Case Studies

In this chapter, we make use of the methods and theorems of the previous chapter to investigate admissibility for some well-known (classes of) algebras, obtaining new structural completeness, almost structural completeness and axiomatization results. We start with the proof that every two element algebra is structurally complete (*Section 4.1*). *Section 4.2* investigates admissibility for three element algebras with one binary operation. *Section 4.3* starts an investigation into admissibility of standard, bounded and pseudocomplemented finite lattices. In *Section 4.4* we present bases for admissible quasiequations for De Morgan and Kleene algebras and lattices. *Section 4.5* studies reducts of Sugihara monoids and *Section 4.6* finally summarizes the obtained results[1].

4.1 Two Element Algebras

In this section we prove that that admissibility in a two element algebra **A** coincides with validity in this algebra **A**, i.e., that every two element algebra is structurally complete. We remark that another proof of this fact was given by Rautenberg in [89, Corollary 1] by proving that each two element algebra generates a minimal quasivariety (compare Theorem 3.16).

[1] Note that most of the calculations in this chapter were done with TAFA (see Chapter 5).

c_0^1	
0	0
1	0

c_1^1	
0	1
1	1

id	
0	0
1	1

$\neg$	
0	1
1	0

c_0^2	0	1
0	0	0
1	0	0

$\wedge$	0	1
0	0	0
1	0	1

$\nrightarrow$	0	1
0	0	0
1	1	0

id_x	0	1
0	0	0
1	1	1

$\nleftarrow$	0	1
0	0	1
1	0	0

id_y	0	1
0	0	1
1	0	1

$\nleftrightarrow$	0	1
0	0	1
1	1	0

$\vee$	0	1
0	0	1
1	1	1

$\downarrow$	0	1
0	1	0
1	0	0

$\leftrightarrow$	0	1
0	1	0
1	0	1

$\neg_y$	0	1
0	1	0
1	1	0

$\leftarrow$	0	1
0	1	0
1	1	1

$\neg_x$	0	1
0	1	1
1	0	0

$\rightarrow$	0	1
0	1	1
1	0	1

$\uparrow$	0	1
0	1	1
1	1	0

c_1^2	0	1
0	1	1
1	1	1

Figure 4.1: Possible unary and binary operations on $\{0, 1\}$.

Theorem 4.1. *Any two element algebra* $\mathbf{A}$ *is structurally complete.*

Proof. Without loss of generality we assume that $\mathbf{A} := \langle\{0, 1\}, \mathcal{F}\rangle$. By Corollary 3.18 it suffices to find an embedding $h\colon \mathbf{A} \to \mathbf{F}_{\mathbf{A}}(\{x_0, x_1\})$. First notice that there are only four unary and sixteen binary operations on the elements 0 and 1 (see Figure 4.1). Also note that some of the binary operations are not proper binary operations in the sense that they do not depend on both variables as, e.g., $\mathrm{id}_x(x, y) := x$ does not depend on y. We proceed by a case distinction on $\mathcal{F}$, always trying to find a suitable embedding $h\colon \mathbf{A} \to \mathbf{F}_{\mathbf{A}}(\{x_0, x_1\})$. For convenience we write $\mathbf{F}$ for $\mathbf{F}_{\mathbf{A}}(\{x_0, x_1\})$ and we say that 0 or 1 is definable, if some c_0^i or c_1^i is definable for $i \in \mathbb{N}$, respectively.

Case 1: $\mathcal{F} = \emptyset$. Define $h(0) := [x_0]$, $h(1) := [x_1]$. Obviously this map is injective and it is a homomorphism since there are no operations to be preserved.

Case 2: 0 *is definable by* $\mathcal{F}$, 1 *is not.* Assume without loss of generality that c_0^i defines 0. Define $h(0) := [c_0^i]$, $h(1) := [x_0]$. h is injective so it remains to show that $h(\circledcirc^{\mathbf{A}}(a_1, \dots, a_n)) = \circledcirc^{\mathbf{F}}(h(a_1), \dots, h(a_n))$ for every $\circledcirc \in \mathcal{F}$ and $a_i \in \{0, 1\}$. It suffices to show that $h(\circledcirc^{\mathbf{A}}(0, \dots, 0, 1, \dots, 1))$ $= \circledcirc^{\mathbf{F}}(h(0), \dots, h(0), h(1), \dots, h(1))$. Note that $\circledcirc^{\mathbf{A}}(0, \dots, 0, x, \dots, x)$ cannot be $\neg x$ or 1, since otherwise 1 would be definable. So there are only two cases:

(i) If $\circledcirc^{\mathbf{A}}(0, \dots, 0, x, \dots, x) = 0$, then $h(\circledcirc^{\mathbf{A}}(0, \dots, 0, 1, \dots, 1)) = h(0) = [c_0^i]$ and $\circledcirc^{\mathbf{F}}(h(0), \dots, h(0), h(1), \dots, h(1)) = [c_0^i]$.

(ii) If $\circledcirc^{\mathbf{A}}(0, \dots, 0, x, \dots, x) = x$, then $h(\circledcirc^{\mathbf{A}}(0, \dots, 0, 1, \dots, 1)) = h(1) = [x_0]$ and $\circledcirc^{\mathbf{F}}(h(0), \dots, h(0), h(1), \dots, h(1)) = [x_0]$.

Case 3: 1 *is definable by* $\mathcal{F}$, 0 *is not.* Very similar to the preceding case.

Case 4: Both 0 *and* 1 *are definable by* $\mathcal{F}$. Assume without loss of generality that c_0^i defines 0 and c_1^j defines 1. The map defined by $h(0) := [c_0^i]$ and $h(1) := [c_1^j]$ is certainly injective and preserves every operation $\circledcirc$ of $\mathcal{F}$.

Case 5: $\mathcal{F} \neq \emptyset$ *only contains unary and binary operations, but* 0 *and* 1 *are not definable by* $\mathcal{F}$. Since $\mathcal{F}$ only contains unary or binary operations, all the possible operations on $\mathbf{A}$ are listed in Figure 4.1. But the following operations cannot be in $\mathcal{F}$ since they define 0 or 1: c_0^1, c_1^1, c_0^2, c_1^2, $\leftarrow$, $\rightarrow$, $\leftrightarrow$, $\not\leftarrow$, $\not\rightarrow$, $\not\leftrightarrow$, $\downarrow$ and $\uparrow$, as e.g., $(x \uparrow x) \uparrow x = 1$. We also do not need to consider binary operations depending only on one variable (c_0^2, c_1^2, id_x, id_y, $\neg_x$, $\neg_y$) since they are preserved by any homomorphism preserving the unary operations. Since id is compatible with every operation, we only have to consider cases where $\mathcal{F}$ contains $\neg$, $\wedge$ or $\vee$. Note that $\neg$ cannot occur together with $\wedge$ or $\vee$, since then 0 and 1 would be definable (e.g., $\neg x \wedge x = 0$).

(i) $\mathcal{F} = \{\neg\}$: Define $h(0) := [x_0]$, $h(1) := [\neg x_0]$. This map is injective and $h(\neg^{\mathbf{A}} x) = \neg^{\mathbf{F}} h(x)$ since $h(\neg^{\mathbf{A}} 0) = h(1) = [\neg x_0] = \neg^{\mathbf{F}}[x_0] = \neg^{\mathbf{F}} h(0)$ and $h(\neg^{\mathbf{A}} 1) = h(0) = [x_0] = \neg^{\mathbf{F}}[\neg x_0] = \neg^{\mathbf{F}} h(1)$.

(ii) $\mathcal{F} = \{\wedge\}$: Let $h(0) := [x_0 \wedge x_1]$, $h(1) := [x_0]$. This map is injective. Also

- $h(0 \wedge^{\mathbf{A}} 0) = h(0) = [x_0 \wedge x_1] = [x_0 \wedge x_1] \wedge^{\mathbf{F}} [x_0 \wedge x_1] = h(0) \wedge^{\mathbf{F}} h(0)$.
- $h(0 \wedge^{\mathbf{A}} 1) = h(0) = [x_0 \wedge x_1] = [x_0 \wedge x_1] \wedge^{\mathbf{F}} [x_0] = h(0) \wedge^{\mathbf{F}} h(1)$.
- $h(1 \wedge^{\mathbf{A}} 0) = h(0) = [x_0 \wedge x_1] = [x_0] \wedge^{\mathbf{F}} [x_0 \wedge x_1] = h(1) \wedge^{\mathbf{F}} h(0)$.
- $h(1 \wedge^{\mathbf{A}} 1) = h(1) = [x_0] = [x_0] \wedge^{\mathbf{F}} [x_0] = h(1) \wedge^{\mathbf{F}} h(1)$.

(iii) $\mathcal{F} = \{\vee\}$: Dual to the previous case with $h(0) := [x_0]$ and $h(1) := [x_0 \vee x_1]$.

(iv) $\mathcal{F} = \{\wedge, \vee\}$: The map defined by $h(0) := [x_0 \wedge x_1]$ and $h(1) := [x_0 \vee x_1]$ is injective and (the preservation of $\vee$ is shown dually)

- $h(0 \wedge^{\mathbf{A}} 0) = h(0) = [x_0 \wedge x_1] = [x_0 \wedge x_1] \wedge^{\mathbf{F}} [x_0 \wedge x_1] = h(0) \wedge^{\mathbf{F}} h(0)$.
- $h(0 \wedge^{\mathbf{A}} 1) = h(0) = [x_0 \wedge x_1] = [x_0 \wedge x_1] \wedge^{\mathbf{F}} [x_0 \vee x_1] = h(0) \wedge^{\mathbf{F}} h(1)$.
- $h(1 \wedge^{\mathbf{A}} 0) = h(0) = [x_0 \wedge x_1] = [x_0 \vee x_1] \wedge^{\mathbf{F}} [x_0 \wedge x_1] = h(1) \wedge^{\mathbf{F}} h(0)$.
- $h(1 \wedge^{\mathbf{A}} 1) = h(1) = [x_0 \vee x_1] = [x_0 \vee x_1] \wedge^{\mathbf{F}} [x_0 \vee x_1] = h(1) \wedge^{\mathbf{F}} h(1)$.

Case 6: $\mathcal{F} \neq \emptyset$ and $\mathcal{F}$ contains operations with arity greater than two, but 0 *and* 1 *are not definable by $\mathcal{F}$.* Let $\mathcal{G}$ be the set of all unary and binary operations obtained by using at most two different parameters of operations in $\mathcal{F}$. A ternary operation $\circledcirc \in \mathcal{F}$, for example, produces $\{\circledcirc_{xxx}, \circledcirc_{xxy}, \circledcirc_{xyx}, \circledcirc_{yxx}\} \subseteq \mathcal{G}$, where, e.g., $\circledcirc_{xxy}(x, y) := \circledcirc(x, x, y)$. By assumption $\mathcal{G}$ fits into *(i)*–*(iv)* of the previous case. Define the appropriate embedding h from $\langle A, \mathcal{G}\rangle$ into $\langle F, \mathcal{G}\rangle$. Indeed, this also embeds $\mathbf{A}$ into $\mathbf{F}$. Since A has only two elements, it suffices to prove that for an arbitrary n-ary operation symbol $\circledcirc \in \mathcal{F}$

$$h(\circledcirc^{\mathbf{A}}(x_0, \ldots, x_0, x_1, \ldots, x_1)) \;=\; \circledcirc^{\mathbf{F}}(h(x_0), \ldots, h(x_0), h(x_1), \ldots, h(x_1)).$$

But by the definition of $\mathcal{G}$ there is a binary $g \in \mathcal{G}$ such that $g^{\langle A,\mathcal{G}\rangle}(x_0, x_1) = \circledcirc^{\mathbf{A}}(x_0, \ldots, x_0, x_1, \ldots, x_1)$, so $h(\circledcirc^{\mathbf{A}}(x_0, \ldots, x_0, x_1, \ldots, x_1)) = h(g^{\langle A,\mathcal{G}\rangle}(x_0, x_1))$. With the fact that h embeds $\langle A, \mathcal{G}\rangle$ into $\langle F, \mathcal{G}\rangle$ we get $h(g^{\langle A,\mathcal{G}\rangle}(x_0, x_1)) = g^{\langle F,\mathcal{G}\rangle}(h(x_0), h(x_1)) = \circledcirc^{\mathbf{F}}(h(x_0), \ldots, h(x_0), h(x_1), \ldots, h(x_1))$ as required. □

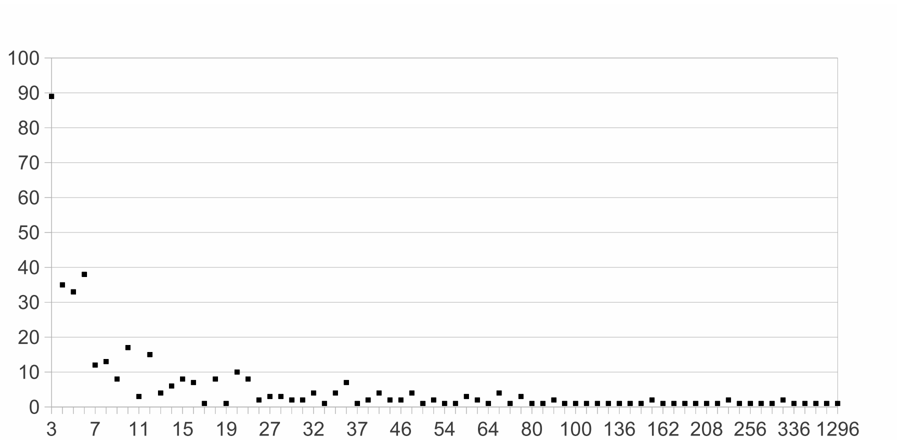


Figure 4.2: Cardinality of the minimal generating free algebras for $\mathbb{Q}(\mathbf{F}_{\mathbf{G}}(\omega))$ (x-axis) and the number of corresponding clone equivalence classes (y-axis).

4.2 Three Element Groupoids

An algebra $\mathbf{G}$ having exactly one binary operation $\star$ is called a *groupoid*. The goal of the present section is to investigate the minimal generating sets of the quasivarieties $\mathbb{Q}(\mathbf{F}_{\mathbf{G}}(3))$ for all three element groupoids $\mathbf{G} := \langle\{0, 1, 2\}, \star\rangle$ (see also [16]). Using Theorems 3.23 and 3.27 we also check which groupoids are (almost) structurally complete. Furthermore we calculate the size of the smallest subalgebra of the free algebra $\mathbf{F}_{\mathbf{G}}(3)$ suitable for checking unifiability in the quasivariety generated by the groupoid $\mathbf{G}$ (see Theorem 3.7).

There are 3330 different groupoids up to isomorphism (out of $3^9 = 19683$ in total) which build 411 classes of clone equivalent algebras. By Theorem 3.28 it suffices to calculate the mentioned properties just once for each clone equivalence class. The full list of the results obtained can be found in Appendix A.

Figure 4.2 gives a rough idea of the distribution of the cardinalities of the minimal generating free algebras of all clone equivalence classes. The number of generators is not always the same to produce a free algebra of a given cardinality and there are even sixteen cases where three generators are needed.

The main goal was to calculate the smallest set of algebras to check admissibility for all groupoids $\mathbf{G}$, namely the results of $\textsc{MinGenSet}(\{\mathbf{F}_{\mathbf{G}}(3)\})$ (see

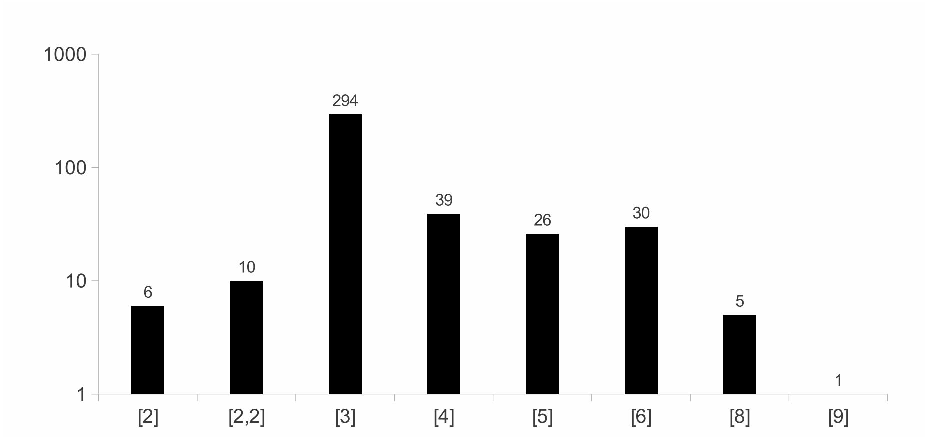


Figure 4.3: Cardinalities of MinGenSet($\{\mathbf{F_G}(3)\}$) (x-axis) and the number of corresponding clone equivalence classes (logarithmic scaled y-axis).

Section 3.1). For free algebras with less than 25 elements we performed MinGenSet directly, for the larger cases we used AdmAlgs (see Section 3.3). The algebras of the minimal generating sets all have fewer than ten elements. Figure 4.3 lists the multisets of cardinalities of the minimal generating sets and for how many clone equivalence class they occur.

Performing the completeness checks to representatives of the groupoid clone equivalence classes confirmed that 107 of the investigated algebras are not structurally complete, of which 31 are almost structurally complete. The remaining 304 groupoids are structurally complete.

Finally, the checks for unifiability showed that for most groupoids unification is trivial: 344 of the groupoids have a one element algebra as subalgebra of the free algebra $\mathbf{F_G}(\omega)$. For the remaining free algebras the smallest subalgebras had two (fifty-seven cases), three (eight cases) or four elements (two cases).

4.3 Lattices

In this section we begin an investigation into admissibility in finite lattices. For small lattices up to five elements we easily confirm structural completeness with TAFA[2], i.e., validity and admissibility coincide for the quasivarieties generated by these lattices. For some lattices, structural completeness also follows from well-known theorems:

Example 4.2. *A* modular lattice $\mathbf{L}$ *may be characterized as a lattice satisfying the equation* $(x \wedge y) \vee (y \wedge z) \approx y \wedge ((x \wedge y) \vee z)$. *Famously, a lattice* $\mathbf{L}$ *is non-modular if and only if the lattice* $\mathbf{L_5}$ *(often called* $\mathbf{N_5}$*) displayed in Table 4.1 embeds into* $\mathbf{L}$ *(see [25, Theorem I.3.5]). But since* $\mathbf{L_5}$ *is non-modular, also* $\mathbf{F}_{\mathbf{L_5}}(\omega)$ *(which must satisfy the same equations) is non-modular. So* $\mathbf{L_5}$ *embeds into* $\mathbf{F}_{\mathbf{L_5}}(\omega)$*, hence* $\mathbf{L_5}$ *is structurally complete. Similarly, it is well-known that a lattice* $\mathbf{L}$ *is distributive if and only if neither* $\mathbf{L_5}$ *nor* $\mathbf{L_4}$ *(often called* $\mathbf{M_5}$*), also displayed in Table 4.1, embeds into* $\mathbf{L}$ *(see [25, Theorem I.3.6]). Since* $\mathbf{L_4}$ *is non-distributive and modular, also* $\mathbf{F}_{\mathbf{L_4}}(\omega)$ *is non-distributive and modular. So* $\mathbf{L_4}$ *embeds into* $\mathbf{F}_{\mathbf{L_4}}(\omega)$*, and* $\mathbf{L_4}$ *is structurally complete.*

Note that bounded lattices (see Section 2.3), obtained from lattices by just adding the constants $\bot$ and $\top$ to the language $\mathcal{L} := \{\wedge, \vee\}$, are not structurally complete in general:

Theorem 4.3. *The smallest bounded lattice which is not structurally complete has five elements.*

Proof. TAFA provides embeddings from the bounded lattices with up to four elements into the corresponding free algebras, so these lattices are structurally complete by Corollary 3.18. Let $\mathbf{L}^b$ be the five element bounded lattice with the universe of $\mathbf{L_4}$ (see Table 4.1), i.e., $\mathbf{L}^b := \langle L_4, \wedge, \vee, \bot, \top \rangle$. TAFA confirms that $\textsc{MinGenSet}(\mathbf{L}^b) = \{\mathbf{L}^b\}$ and that there is no embedding from $\mathbf{L}^b$ into $\mathbf{F}_{\mathbf{L}^b}(3)$, the minimal generating free algebra for $\mathbb{Q}(\mathbf{F}_{\mathbf{L}^b}(\omega))$. Hence $\mathbf{L}^b$ is not structurally complete by Corollary 3.24. An example of a quasiequation that is admissible but not valid in $\mathbf{L}^b$ is

$$x \vee y \approx \top,\ x \wedge z \approx \bot,\ y \wedge z \approx \bot \quad \Rightarrow \quad z \approx \bot. \qquad \square$$

[2] A list of all (non-trivial) lattices up to size seven can be found on `http://math.chapman.edu/~jipsen/posets/lattices77.html`.

We were unable to check structural completeness for all six element lattices since the free algebras for the lattices $\mathbf{L_9}$ and $\mathbf{L_{10}}$ are too big for TAFA (e.g., the algebra $\mathbf{F}_{\mathbf{L_9}}(4)$ has 56694 elements). For all other lattices with up to six elements (see Table 4.1) TAFA confirms structural completeness. To our knowledge it is still an open question whether all finite lattices are structurally complete. However, the variety of all lattices is not structurally complete, since every free lattice satisfies the semi-distributivity laws (see Example 3.10), but there are lattices which are not semi-distributive, e.g., $\mathbf{L_4}$ (see also [106, 80]).

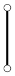
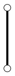
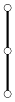
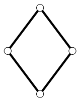
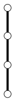
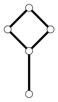
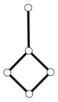
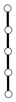
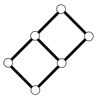
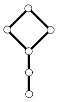
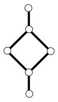
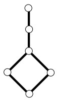
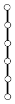
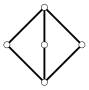
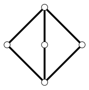
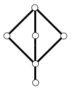
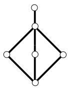
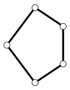
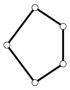
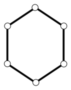
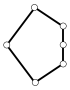
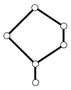
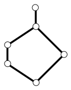
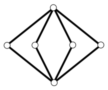
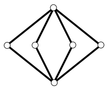
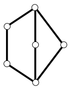
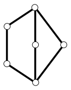
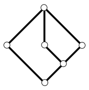
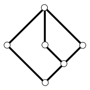
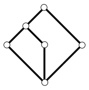
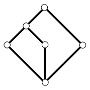
We now consider a special class of distributive lattices extended not only with $\top$ and $\bot$, but also a unary operation *. A *pseudocomplemented distributive lattice* (*PCL* for short) is an algebra $\mathbf{L} := \langle L, \wedge, \vee, ^*, \bot, \top \rangle$ such that $\langle L, \wedge, \vee, \bot, \top \rangle$ is a distributive bounded lattice and the unary operation * is *pseudocomplementation*, i.e.,

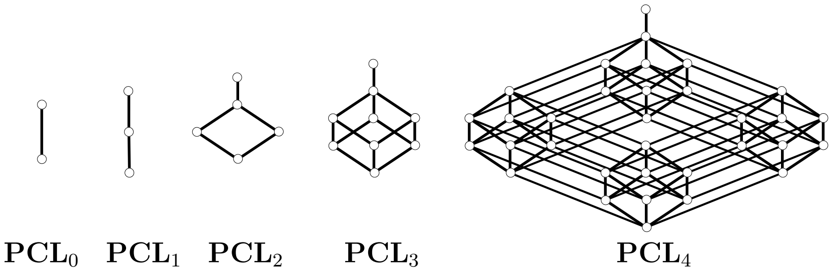
$$x \wedge y \; = \; \bot \quad \text{iff} \quad y \; \leq \; x^*.$$

It is known that the class of PCLs is a variety (see [90]) and that the subdirectly irreducible pseudocomplemented distributive lattices are exactly (up to isomorphism) Boolean algebras extended with an extra top element corresponding to the constant $\top$ where the negation is adapted such that both $\neg\bot = \top$ and $\neg\top = \bot$ hold ([67, Theorem 2]).

We have considered here the first five subdirectly irreducible PCLs, depicted in Figure 4.4. Note that $\mathbf{PCL}_0$, the smallest non-trivial PCL, is just the two element Boolean algebra. The cardinalities of the minimal generating free algebras and the minimal generating sets (column "M") are listed in Table 4.2. The algebra $\mathbf{PCL}_1$ generates the variety of *Stone algebras* (see, e.g., [52]), which is structurally complete. $\mathbf{PCL}_2$ is also structurally complete, but not $\mathbf{PCL}_3$ or $\mathbf{PCL}_4$.

Table 4.1: Lattices with up to six elements.

MinGenSet($\mathbf{L}$)	Lattices		
	$\mathbf{L}_t$		
	$\mathbf{L}_0$ 	$\mathbf{L}_1$ 	$\mathbf{L}_2$ 
	$\mathbf{L}_3$ 	$\mathbf{L}_6$ 	$\mathbf{L}_7$ 
	$\mathbf{L}_8$ 	$\mathbf{L}_{17}$ 	$\mathbf{L}_{18}$ 
	$\mathbf{L}_{21}$ 	$\mathbf{L}_{22}$ 	$\mathbf{L}_{23}$ 
	$\mathbf{L}_4$ 	$\mathbf{L}_{12}$ 	$\mathbf{L}_{19}$ 
	$\mathbf{L}_5$ 	$\mathbf{L}_{14}$ 	$\mathbf{L}_{15}$ 
	$\mathbf{L}_{16}$ 	$\mathbf{L}_{20}$ 	
	$\mathbf{L}_9$ 		
	$\mathbf{L}_{10}$ 		
	$\mathbf{L}_{11}$ 		
	$\mathbf{L}_{13}$ 		



PCL$_0$ **PCL**$_1$ **PCL**$_2$ **PCL**$_3$ **PCL**$_4$

Figure 4.4: The five first (non-trivial) subdirectly irreducible PCLs.

Table 4.2: Admissibility for PCLs.

Lattice	Cardinality	Free algebra	M
$\mathbf{PCL}_0$	$2^0 + 1 = 2$	$\lvert\mathbf{F}_{\mathbf{PCL}_0}(0)\rvert = 2$	2
$\mathbf{PCL}_1$	$2^1 + 1 = 3$	$\lvert\mathbf{F}_{\mathbf{PCL}_1}(1)\rvert = 6$	3
$\mathbf{PCL}_2$	$2^2 + 1 = 5$	$\lvert\mathbf{F}_{\mathbf{PCL}_2}(1)\rvert = 7$	5
$\mathbf{PCL}_3$	$2^3 + 1 = 9$	$\lvert\mathbf{F}_{\mathbf{PCL}_3}(2)\rvert = 625$	19
$\mathbf{PCL}_4$	$2^4 + 1 = 17$	$\lvert\mathbf{F}_{\mathbf{PCL}_4}(2)\rvert = 626$	167[3]

4.4 De Morgan and Kleene Algebras

This section provides bases for the admissible quasiequations of the classes of Kleene lattices KL, Kleene algebras KA, De Morgan lattices DML and De Morgan algebras DMA, mainly making use of Theorems 3.17 and 3.21.

Recall from Example 3.8 that *De Morgan algebras* are defined as algebras $\langle A, \wedge, \vee, \neg, \bot, \top\rangle$ such that $\langle A, \wedge, \vee, \bot, \top\rangle$ is a bounded distributive lattice satisfying the De Morgan laws and $\neg$ is an involutive negation. The class DMA of De Morgan algebras forms a variety containing just two proper non-trivial subvarieties: the class KA of *Kleene algebras* satisfying $x \wedge \neg x \leq y \vee \neg y$ and the class BA of *Boolean algebras* satisfying $x \leq y \vee \neg y$ (see [65]), where $x \leq y$ stands for $x \wedge y \approx y$. The classes DML, KL and BL of *De Morgan*, *Kleene* and *Boolean lattices* are defined analogously by omitting the

[3] We have found a subalgebra of $\mathbf{F}_{\mathbf{PCL}_4}(2)$ with 167 elements which is a prehomomorphic image of $\mathbf{PCL}_4$, but we were not able to confirm that this is the smallest subalgebra with this property. Also, for the procedure MinGenSet this algebra is too big.

constants $\bot$ and $\top$ from the language. We define $\mathcal{L}_{\mathrm{DMA}} := \{\wedge, \vee, \neg, \bot, \top\}$, $\mathcal{L}_{\mathrm{DML}} := \mathcal{L}_{\mathrm{DMA}} \setminus \{\bot, \top\}$ and write $\mathbf{A}^{\ell}$ to denote the $\mathcal{L}_{\mathrm{DML}}$-reduct of a De Morgan algebra $\mathbf{A}$. Moreover, we define the following finite members of KA for $1 \leq m \in \mathbb{N}$, with operations $x \wedge y := \min\{x, y\}$, $x \vee y := \max\{x, y\}$, $\neg x := -x$, $\bot := -m$ and $\top := m$:

$$\begin{aligned} \mathbf{C_{2m}} &:= \langle \{-m, -m+1, \ldots, -1, 1, \ldots, m-1, m\}, \wedge, \vee, \neg, \bot, \top \rangle \\ \mathbf{C_{2m+1}} &:= \langle \{-m, -m+1, \ldots, -1, 0, 1, \ldots, m-1, m\}, \wedge, \vee, \neg, \bot, \top \rangle. \end{aligned}$$

The "fuzzy algebra" $\langle [0,1], \min, \max, 1-x, 0, 1 \rangle$ and also each $\mathbf{C_n}$ for any *odd* $n \geq 3$, generates KA as a quasivariety. In particular, $\mathsf{KA} = \mathbb{Q}(\mathbf{C_3})$ (see, e.g. [65, 88]). Now consider the quasiequation

$$x \approx \neg x \quad \Rightarrow \quad x \approx y. \tag{4.1}$$

(4.1) is not $\mathbf{C_3}$-valid: just consider the homomorphism $h\colon \mathbf{Tm}_{\mathcal{L}_{\mathrm{DMA}}}(x, y) \to \mathbf{C_3}$ defined by $h(x) := 0$ and $h(y) := 1$. But there is no term φ such that $\varphi \approx \neg\varphi$ holds in all Kleene algebras (or indeed, in all Boolean algebras). So the quasiequation (4.1) is admissible and by Theorem 3.16, KA is not structurally complete. However, the proper subquasivariety of KA generated by $\mathbf{C_n}$ for any *even* $n \geq 4$ is structurally complete. In particular, using Corollary 3.18 we can show that $\mathbf{C_4}$ is structurally complete with the map $g^{\mathbf{C_4}}\colon \mathrm{C}_4 \to \mathrm{Tm}_{\mathcal{L}_{\mathrm{DMA}}}$ defined by

$$\begin{aligned} 2 &\mapsto \top \\ 1 &\mapsto x \vee \neg x \\ -1 &\mapsto x \wedge \neg x \\ -2 &\mapsto \bot. \end{aligned}$$

Lemma 4.4. *$\mathbb{Q}(\mathbf{C_4})$ is axiomatized relative to KA by the quasiequation*

$$\neg x \leq x,\ x \wedge \neg y \leq \neg x \vee y \quad \Rightarrow \quad \neg y \leq y. \tag{4.2}$$

Proof. Very similar to the proof of [88], Proposition 4.7, which states that

$\mathbb{Q}(\mathbf{C}_\mathbf{4}^\ell)$ is axiomatized relative to KL by the quasiequation (4.2). □

Theorem 4.5. $\{(4.2)\}$ *is a basis for the admissible quasiequations of* KA.

Proof. $\mathbb{Q}(\mathbf{C_4})$ is structurally complete and axiomatized relative to KA by $\{(4.2)\}$ by Lemma 4.4. Moreover, $\mathbf{C_3}$ is a homomorphic image of $\mathbf{C_4}$, so $\mathbb{V}(\mathbf{C_4}) = \mathbb{V}(\mathbf{C_3}) = \mathsf{KA}$. Hence, since (4.2) holds in $\mathbf{C_4}$, it is admissible in KA, and the result follows by Theorem 3.21. □

Note that the quasiequation (4.1) does not provide a basis for the admissible quasiequations of KA. In fact, it axiomatizes the quasivariety $\mathbb{Q}(\mathbf{C_3} \times \mathbf{C_2})$ relative to KA (see [88], Proposition 4.5). With the same reasoning we also obtain a basis for the admissible quasiequations of KL:

Lemma 4.6 ([88, Proposition 4.7])**.** $\mathbb{Q}(\mathbf{C}_\mathbf{4}^\ell)$ *is axiomatized relative to* KL *by the quasiequation* (4.2).

Theorem 4.7. $\{(4.2)\}$ *is a basis for the admissible quasiequations of* KL.

Proof. Using Corollary 3.18 with the map $g^{\mathbf{C}_\mathbf{4}^\ell}\colon \mathrm{C}_4^\ell \to \mathrm{Tm}_{\mathcal{L}_{\mathrm{DML}}}$ defined by

$$\begin{array}{rcl} 2 & \mapsto & (x \vee \neg x) \vee y \\ 1 & \mapsto & x \vee \neg x \\ -1 & \mapsto & x \wedge \neg x \\ -2 & \mapsto & (x \wedge \neg x) \wedge y, \end{array}$$

$\mathbb{Q}(\mathbf{C}_\mathbf{4}^\ell)$ is structurally complete. By Lemma 4.6, $\mathbb{Q}(\mathbf{C}_\mathbf{4}^\ell)$ is axiomatized relative to KL by $\{(4.2)\}$. Moreover, $\mathbf{C}_\mathbf{3}^\ell$ is a homomorphic image of $\mathbf{C}_\mathbf{4}^\ell$, so $\mathbb{V}(\mathbf{C}_\mathbf{4}^\ell) = \mathbb{V}(\mathbf{C}_\mathbf{3}^\ell) = \mathsf{KL}$. Hence, since (4.2) holds in $\mathbf{C}_\mathbf{4}^\ell$, it is admissible in KL, and the result follows by Theorem 3.21. □

We now turn our attention to the classes of De Morgan algebras DMA and De Morgan lattices DML (see Example 3.8 or Figure 4.5), which are generated as quasivarieties by the algebras $\mathbf{D_4}$ and $\mathbf{D}_\mathbf{4}^\ell$, respectively (see [65]). De Morgan lattices were first studied by Moisil [79] and Kalman [65], and subsequently, with or without the constants $\bot$ and $\top$, by many other researchers.

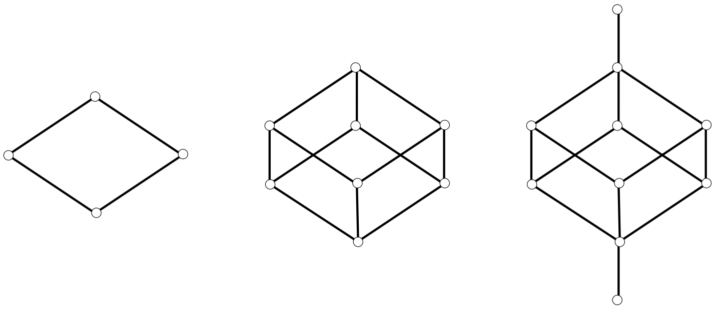


Figure 4.5: The De Morgan algebras $\mathbf{D_4}$, $\mathbf{D_{42}}$ and $\bar{\mathbf{D}}_{\mathbf{42}}$.

In particular, the quasivariety lattice of De Morgan lattices has been fully characterized by Pynko in [88] (see Figure 4.6), while the more complicated (infinite) quasivariety lattice of De Morgan algebras has been investigated by Gaitán and Perea in [40].

As before, we use an axiomatization lemma and Theorem 3.21 to find a basis for the admissible quasiequations of DML:

Lemma 4.8 ([88, Proposition 4.2])**.** $\mathbb{Q}(\mathbf{D^\ell_{42}})$ *is axiomatized relative to* DML *by the quasiequation* (4.1)*.*

Theorem 4.9. $\{(4.1)\}$ *is a basis for the admissible quasiequations of* DML*.*

Proof. By Theorem 3.16 a quasivariety $\mathcal{Q}$ is structurally complete if every proper subquasivariety of $\mathcal{Q}$ generates a proper subvariety of $\mathbb{V}(\mathcal{Q})$. The only non-trivial varieties of De Morgan lattices are $\mathsf{BL} = \mathbb{Q}(\mathbf{C^\ell_2})$, $\mathsf{KL} = \mathbb{Q}(\mathbf{C^\ell_3})$ and $\mathsf{DML} = \mathbb{Q}(\mathbf{D^\ell_4})$. Hence by inspection of the subquasivariety lattice, the only non-trivial structurally complete subquasivarieties of DML are $\mathsf{BL} = \mathbb{Q}(\mathbf{C^\ell_2})$, $\mathbb{Q}(\mathbf{C^\ell_4})$ and $\mathbb{Q}(\mathbf{D^\ell_{42}})$ where $\mathbf{D_{42}}$ is defined as the direct product $\mathbf{D_4} \times \mathbf{C_2}$ (see Figure 4.5)[4]. By Lemma 4.8, $\mathbb{Q}(\mathbf{D^\ell_{42}})$ is axiomatized relative to DML by $\{(4.1)\}$. Moreover, $\mathbf{D^\ell_4}$ is a homomorphic image of $\mathbf{D^\ell_{42}}$ using the first projection homomorphism, so $\mathbb{V}(\mathbf{D^\ell_{42}}) = \mathbb{V}(\mathbf{D^\ell_4}) = \mathsf{DML}$. Hence, since (4.1)

[4] We also easily find an embedding of $\mathbf{D^\ell_{42}}$ into $\mathbf{F}_{\mathbf{D^\ell_{42}}}(2)$ using TAFA. Then $\mathbb{Q}(\mathbf{D^\ell_{42}})$ is structurally complete using Corollary 3.18.

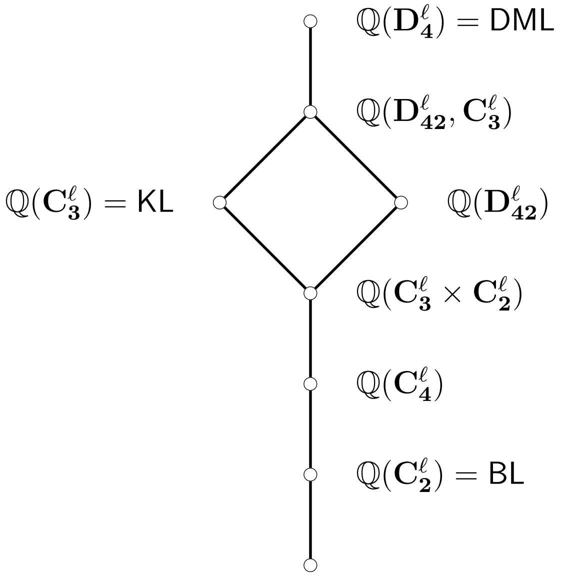

$\mathbb{Q}(\mathbf{D}^\ell_\mathbf{4}) = \mathsf{DML}$
$\mathbb{Q}(\mathbf{D}^\ell_\mathbf{42}, \mathbf{C}^\ell_\mathbf{3})$
$\mathbb{Q}(\mathbf{C}^\ell_\mathbf{3}) = \mathsf{KL}$
$\mathbb{Q}(\mathbf{D}^\ell_\mathbf{42})$
$\mathbb{Q}(\mathbf{C}^\ell_\mathbf{3} \times \mathbf{C}^\ell_\mathbf{2})$
$\mathbb{Q}(\mathbf{C}^\ell_\mathbf{4})$
$\mathbb{Q}(\mathbf{C}^\ell_\mathbf{2}) = \mathsf{BL}$

Figure 4.6: Subquasivarieties of DML.

holds in $\mathbf{D}^\ell_\mathbf{42}$, it is admissible in DML, and the result follows by Theorem 3.21. □

The case of De Morgan algebras is more complicated since the lattice of quasivarieties is infinite (see [40, Figure 7]). Unlike the case of DML, the quasiequation (4.1) does not provide a basis for the admissible quasiequations of DMA. It follows from results of Pynko [88] that $\{(4.1)\}$ axiomatizes the quasivariety $\mathbb{Q}(\mathbf{D_{42}})$ relative to DMA. However, the quasiequation

$$(x \land \neg x) \lor y \approx \top \quad \Rightarrow \quad y \approx \top$$

is admissible in DMA but does not hold in the De Morgan algebra $\mathbf{D_{42}}$. So $\{(4.1)\}$ cannot suffice as a basis for the admissible quasiequations of DMA.

Let us consider instead the De Morgan algebra $\bar{\mathbf{D}}_\mathbf{42}$ obtained from $\mathbf{D_{42}}$ by adding an extra top element $\top$ and bottom element $\bot$ (see Figure 4.5). Note that $\mathbf{D_4}$ is a homomorphic image of $\bar{\mathbf{D}}_\mathbf{42}$ under the composition of $f\colon \bar{\mathbf{D}}_\mathbf{42} \to \mathbf{D_{42}}$, $f(\top) := (\top, 1)$, $f(\bot) := (\bot, 0)$, $f((x, y)) := (x, y)$ for all $(x, y) \notin \{\bot, \top\}$ and the projection $p^2_1\colon \mathbf{D_{42}} \to \mathbf{D_4}$. Hence $\mathbb{V}(\mathbb{Q}(\bar{\mathbf{D}}_\mathbf{42})) =$

DMA. TAFA provides an embedding of $\mathbf{D}_{\mathbf{42}}^{-}$ into the free algebra $\mathbf{F}_{\mathbf{D}_{\mathbf{42}}^{-}}(\{x,y\})$ defined by

$$\begin{array}{rcl@{\qquad}rcl}
(\bot,\bot) & \mapsto & (\varphi\wedge\psi)\vee(\neg\varphi\wedge\neg\psi) & (\top,\top) & \mapsto & (\varphi\vee\psi)\wedge(\neg\varphi\vee\neg\psi)\\
(a,\bot) & \mapsto & (\neg\varphi\wedge\neg\psi)\vee\varphi & (a,\top) & \mapsto & (\varphi\vee\psi)\wedge\neg\varphi\\
(b,\bot) & \mapsto & (\varphi\wedge\psi)\vee\neg\psi & (b,\top) & \mapsto & (\neg\varphi\vee\neg\psi)\wedge\psi\\
(\top,\bot) & \mapsto & \varphi\vee\neg\psi & (\bot,\top) & \mapsto & \neg\varphi\wedge\psi,
\end{array}$$

where $\varphi := x\wedge\neg x$ and $\psi := y\vee\neg y$. Hence $\mathbf{D}_{\mathbf{42}}^{-}$ is structurally complete by Corollary 3.18, so the admissible quasiequations of DMA consist of those quasiequations that hold in $\mathbb{Q}(\mathbf{D}_{\mathbf{42}}^{-})$.

We now present an axiomatization of the admissible quasiequations of De Morgan algebras using also clauses and not only quasiequations. Observe that the following clause holds in $\mathbf{D}_{\mathbf{42}}^{-}$ and hence also in $\mathbf{F}_{\mathsf{DMA}}$ using $\mathbb{V}(\mathbf{D}_{\mathbf{42}}^{-}) = \mathbb{V}(\mathsf{DMA})$, Theorem 2.14 and Corollary 2.16:

$$x\vee y\approx\top \quad\Rightarrow\quad x\approx\top,\ \ y\approx\top. \tag{4.3}$$

We define $\mathsf{DMA}^* := \{\mathbf{A}\in\mathsf{DMA} : \mathbf{A}$ satisfies (4.1) and (4.3)$\}$. We will show that a quasiequation is admissible in DMA if and only if it is valid in DMA^*. The main idea of the proof is to reduce the question of the admissibility of a quasiequation in DMA to the question of the admissibility of quasiequations in DML. The following lemma will be useful in this respect. For a set of $\mathcal{L}_{\mathrm{DMA}}$-equations, let $c(\Sigma)$ be the number of occurrences of connectives $\wedge$, $\vee$ and $\neg$.

Lemma 4.10. *For any $\varphi\in\mathrm{Tm}_{\mathcal{L}_{\mathrm{DMA}}}$, one of the following holds:*

(i) $\models_{\mathsf{DMA}}\varphi\approx\bot$.

(ii) $\models_{\mathsf{DMA}}\varphi\approx\top$.

(iii) $\models_{\mathsf{DMA}}\varphi\approx\psi$ *for some* $\psi\in\mathrm{Tm}_{\mathcal{L}_{\mathrm{DML}}}$ *with* $c(\psi)\leq c(\varphi)$.

Proof. For an arbitrary $\varphi\in\mathrm{Tm}_{\mathcal{L}_{\mathrm{DMA}}}$, we proceed by induction on the length of the term: In the *base case* φ is atomic, i.e., $\varphi=\bot$, $\varphi=\top$ or $\varphi=x$ for some variable x as required. For the *inductive step* suppose the assumption holds for $|\varphi| < n$. Then there are three cases:

(i) $\varphi = \varphi_1 \wedge \varphi_2$. Without loss of generality we have the following cases:

- $\vDash_{\mathsf{DMA}} \varphi_1 \approx \bot \Rightarrow \vDash_{\mathsf{DMA}} \varphi \approx \bot \wedge \varphi_2 = \bot$.
- $\vDash_{\mathsf{DMA}} \varphi_1 \approx \top \Rightarrow \vDash_{\mathsf{DMA}} \varphi \approx \top \wedge \varphi_2 = \varphi_2$.
- $\vDash_{\mathsf{DMA}} \varphi_1 \approx \psi_1$ and $\vDash_{\mathsf{DMA}} \varphi_2 \approx \psi_2$ for some $\psi_1, \psi_2 \in \mathrm{Tm}_{\mathcal{L}_{\mathrm{DML}}}$ with $c(\psi_1) \leq c(\varphi_1)$ and $c(\psi_2) \leq c(\varphi_2) \Rightarrow \vDash_{\mathsf{DMA}} \varphi \approx \psi_1 \wedge \psi_2 = \psi$ with $\psi \in \mathrm{Tm}_{\mathcal{L}_{\mathrm{DML}}}$ and $c(\psi) \leq c(\varphi)$.

(ii) $\varphi = \varphi_1 \vee \varphi_2$. Dual to the previous case.

(iii) $\varphi = \neg\varphi_1$. There are three cases:

- $\vDash_{\mathsf{DMA}} \varphi_1 \approx \bot \Rightarrow \vDash_{\mathsf{DMA}} \varphi \approx \top$.
- $\vDash_{\mathsf{DMA}} \varphi_1 \approx \top \Rightarrow \vDash_{\mathsf{DMA}} \varphi \approx \bot$.
- $\vDash_{\mathsf{DMA}} \varphi_1 \approx \psi_1$ for some $\psi_1 \in \mathrm{Tm}_{\mathcal{L}_{\mathrm{DML}}}$ with $c(\psi_1) \leq c(\varphi_1) \Rightarrow \vDash_{\mathsf{DMA}} \varphi \approx \neg\psi_1 = \psi$ with $\psi \in \mathrm{Tm}_{\mathcal{L}_{\mathrm{DML}}}$ and $c(\psi) \leq c(\varphi)$. □

Let us say that an $\mathcal{L}_{\mathrm{DMA}}$-equation $\varphi \approx \psi$ is in *normal form* if φ and ψ are either $\bot$, $\top$ or members of $\mathrm{Tm}_{\mathcal{L}_{\mathrm{DML}}}$.

Theorem 4.11. *Let $\Sigma \Rightarrow \varphi \approx \psi$ be an $\mathcal{L}_{\mathrm{DMA}}$-quasiequation. Then*

$$\Sigma \Rightarrow \varphi \approx \psi \text{ is admissible in } \mathsf{DMA} \quad \text{iff} \quad \Sigma \models_{\mathsf{DMA}^*} \varphi \approx \psi.$$

Proof. Suppose first that $\Sigma \models_{\mathsf{DMA}^*} \varphi \approx \psi$. Both the quasiequation (4.1) and the clause (4.3) hold in $\mathbf{F}_{\mathsf{DMA}}$, so $\mathbf{F}_{\mathsf{DMA}} \in \mathsf{DMA}^*$. Hence $\Sigma \models_{\mathbf{F}_{\mathsf{DMA}}} \varphi \approx \psi$ and by Theorem 3.9, $\Sigma \Rightarrow \varphi \approx \psi$ is admissible in DMA.

For the other direction, it suffices, using Lemma 4.10 and Theorem 3.9, to prove that for any finite set $\Sigma \cup \{\varphi \approx \psi\}$ of $\mathcal{L}_{\mathrm{DMA}}$-equations in normal form:

$$\Sigma \models_{\mathbf{F}_{\mathsf{DMA}}} \varphi \approx \psi \quad \text{implies} \quad \Sigma \models_{\mathsf{DMA}^*} \varphi \approx \psi. \qquad (\star)$$

We prove ($\star$) by induction on the lexicographically ordered pair $\langle c(\Sigma), s(\Sigma)\rangle$, where $s(\Sigma)$ be the number of equations in Σ containing $\bot$ or $\top$. The idea is to successively eliminate occurrences of $\bot$ and $\top$ in Σ by reducing $\langle c(\Sigma), s(\Sigma)\rangle$.

Base case. Suppose that there are no occurrences of $\bot$ and $\top$ in Σ, i.e., $s(\Sigma) = 0$. If $\varphi = \psi$ or $\{\varphi, \psi\} \subseteq \{\bot, \top\}$, then we are done. Moreover, if $\varphi \in \mathrm{Tm}_{\mathcal{L}_{\mathrm{DML}}}$ and $\psi \in \{\bot, \top\}$, then $\Sigma \not\models_{\mathbf{F}_{\mathsf{DMA}}} \varphi \approx \psi$: just consider a homomorphism from $\mathbf{Tm}_{\mathcal{L}_{\mathrm{DMA}}}$ to $\mathbf{D_4}$ that maps all the variables to a. Finally, consider $\varphi, \psi \in \mathrm{Tm}_{\mathcal{L}_{\mathrm{DML}}}$. Suppose that $\Sigma \models_{\mathbf{F}_{\mathsf{DMA}}} \varphi \approx \psi$. By Theorem 3.9, $\Sigma \Rightarrow \varphi \approx \psi$ is admissible in DMA. But for any $\varphi', \psi' \in \mathrm{Tm}_{\mathcal{L}_{\mathrm{DML}}}$, we have $\models_{\mathsf{DMA}} \varphi' \approx \psi'$ iff $\models_{\mathbf{D_4}} \varphi' \approx \psi'$ iff $\models_{\mathbf{D_4^\ell}} \varphi' \approx \psi'$ iff $\models_{\mathsf{DML}} \varphi' \approx \psi'$. So $\Sigma \Rightarrow \varphi \approx \psi$ is admissible in DML. Hence by Theorem 4.9, $\Sigma \Rightarrow \varphi \approx \psi$ holds in $\mathbb{Q}(\mathbf{D^\ell_{42}})$. But every De Morgan algebra in DMA^* is also (ignoring $\bot$ and $\top$ in the language) a De Morgan lattice in $\mathbb{Q}(\mathbf{D^\ell_{42}})$, so $\Sigma \models_{\mathsf{DMA}^*} \varphi \approx \psi$.

Inductive step. Given the set Σ, suppose that ($\star$) holds for all Δ and $\langle c(\Delta), s(\Delta)\rangle < \langle c(\Sigma), s(\Sigma)\rangle$. We use $A \sqcup B$ to denote the *disjoint union* of two sets A and B, i.e., $A \cap B = \emptyset$. Consider the following cases:

- $\Sigma = \Delta \sqcup \{\bot \approx \top\}$. Then ($\star$) clearly holds since $\Sigma \models_{\mathsf{DMA}^*} \varphi \approx \psi$.

- $\Sigma = \Delta \sqcup \{\chi \approx \chi\}$. Then $\Sigma \models_{\mathbf{F}_{\mathsf{DMA}}} \varphi \approx \psi$ implies $\Delta \models_{\mathbf{F}_{\mathsf{DMA}}} \varphi \approx \psi$ and, by the induction hypothesis, $\Delta \models_{\mathsf{DMA}^*} \varphi \approx \psi$. So $\Delta \sqcup \{\chi \approx \chi\} \models_{\mathsf{DMA}^*} \varphi \approx \psi$ as required.

- $\Sigma = \Delta \sqcup \{\chi_1 \vee \chi_2 \approx \bot\}$. Suppose that $\Delta \sqcup \{\chi_1 \vee \chi_2 \approx \bot\} \models_{\mathbf{F}_{\mathsf{DMA}}} \varphi \approx \psi$. Then also $\Delta \cup \{\chi_1 \approx \bot, \chi_2 \approx \bot\} \models_{\mathbf{F}_{\mathsf{DMA}}} \varphi \approx \psi$. So by the induction hypothesis, $\Delta \cup \{\chi_1 \approx \bot, \chi_2 \approx \bot\} \models_{\mathsf{DMA}^*} \varphi \approx \psi$. But then since $\{\chi_1 \vee \chi_2 \approx \bot\} \models_{\mathsf{DMA}^*} \chi_i \approx \bot$ for $i = 1, 2$, we obtain $\Delta \sqcup \{\chi_1 \vee \chi_2 \approx \bot\} \models_{\mathsf{DMA}^*} \varphi \approx \psi$ as required.

- $\Sigma = \Delta \sqcup \{\chi_1 \vee \chi_2 \approx \top\}$. Suppose that $\Delta \sqcup \{\chi_1 \vee \chi_2 \approx \top\} \models_{\mathbf{F}_{\mathsf{DMA}}} \varphi \approx \psi$. Then $\Delta \cup \{\chi_i \approx \top\} \models_{\mathbf{F}_{\mathsf{DMA}}} \varphi \approx \psi$ for $i = 1, 2$. So by the induction hypothesis, $\Delta \cup \{\chi_i \approx \top\} \models_{\mathsf{DMA}^*} \varphi \approx \psi$ for $i = 1, 2$. But now, since (4.3) holds in every algebra in DMA^*, we have $\Delta \sqcup \{\chi_1 \vee \chi_2 \approx \top\} \models_{\mathsf{DMA}^*} \varphi \approx \psi$ as required.

- $\Sigma = \Delta \sqcup \{\neg\chi \approx \top\}$. Suppose that $\Delta \sqcup \{\neg\chi \approx \top\} \models_{\mathbf{F}_{\mathsf{DMA}}} \varphi \approx \psi$. Then $\Delta \cup \{\chi \approx \bot\} \models_{\mathbf{F}_{\mathsf{DMA}}} \varphi \approx \psi$, so by the induction hypothesis, $\Delta \cup \{\chi \approx \bot\} \models_{\mathsf{DMA}^*} \varphi \approx \psi$. But then also $\Delta \sqcup \{\neg\chi \approx \top\} \models_{\mathsf{DMA}^*} \varphi \approx \psi$ as required.

- $\Sigma = \Delta \sqcup \{x \approx \top\}$. Suppose that $\Delta \sqcup \{x \approx \top\} \models_{\mathbf{F}_{\mathsf{DMA}}} \varphi \approx \psi$. Let Δ' and $\varphi' \approx \psi'$ be the result of substituting every occurrence of $\top$ for x in Δ and $\varphi \approx \psi$, respectively. Then $\Delta' \models_{\mathbf{F}_{\mathsf{DMA}}} \varphi' \approx \psi'$. Notice that $c(\Delta') = c(\Sigma)$ and $s(\Delta') < s(\Sigma)$. By Lemma 4.10, we can find equations Δ^* and $\varphi^* \approx \psi^*$ in normal form such that

  (a) $\Delta^* \models_{\mathbf{F}_{\mathsf{DMA}}} \varphi^* \approx \psi^*$.

  (b) $c(\Delta^*) \leq c(\Delta')$ and $s(\Delta^*) = s(\Delta')$.

  (c) $\Delta^* \models_{\mathsf{DMA}^*} \varphi^* \approx \psi^*$ implies $\Delta \cup \{x \approx \top\} \models_{\mathsf{DMA}^*} \varphi \approx \psi$.

  By the induction hypothesis, using 1. and 2., $\Delta^* \models_{\mathsf{DMA}^*} \varphi^* \approx \psi^*$. But then also by 3., $\Delta \cup \{x \approx \top\} \models_{\mathsf{DMA}^*} \varphi \approx \psi$ as required.

- The cases $\Sigma = \Delta \sqcup \{\chi_1 \wedge \chi_2 \approx \bot\}$, $\Sigma = \Delta \sqcup \{\chi_1 \wedge \chi_2 \approx \top\}$, $\Sigma = \Delta \sqcup \{\neg\chi \approx \bot\}$ and $\Sigma = \Delta \sqcup \{x \approx \bot\}$ are treated symmetrically to the preceding cases. □

We close this section by remarking that Cabrer and Metcalfe (see [26, Theorem 30]) have recently used natural dualities to show that the following quasiequations (4.4) and (4.5) provide a basis for the admissible quasiequations of DMA:

$$x \leq \neg x, \quad \neg(x \vee y) \leq x \vee y, \quad \neg y \vee z \approx \top \quad \Rightarrow \quad z \approx \top \tag{4.4}$$

$$x \leq \neg x, \quad y \leq \neg y, \quad x \wedge y \approx \bot \quad \Rightarrow \quad x \vee y \leq \neg(x \vee y) \tag{4.5}$$

4.5 Reducts of Sugihara Monoids

In this section we consider (reducts of) *Sugihara monoids*, members of the variety generated by the algebras $\{\mathbf{Z^e_{2m}} : m \geq 1\}$, where

$$\mathbf{Z^e_{2m}} \;:=\; \langle\{-m, -m+1, \ldots, -1, 1, \ldots, m-1, m\}, \wedge, \vee, \to, \neg, e\rangle$$

with $\wedge$ and $\vee$ as min and max, respectively, $\neg x := -x$, $x \to y := \neg x \vee y$ if $x \leq y$ and $\neg x \wedge y$ otherwise, and $e := 1$ (see [35]). We also define the algebras

$$\begin{aligned} \mathbf{Z^e} \;&:=\; \langle\mathbb{Z}, \wedge, \vee, \to, \neg, e\rangle \quad \text{and} \\ \mathbf{Z^e_{2m+1}} \;&:=\; \langle\{-m, -m+1, \ldots, -1, 0, 1, \ldots, m-1, m\}, \wedge, \vee, \to, \neg, e\rangle, \end{aligned}$$

with the same definitions of the operations except that the constant e has value 0. For any $\mathcal{L}$-reduct of a Sugihara monoid $\mathbf{A}$ we write $\mathbf{A}^{\mathcal{L}}$, except that we delete the set $\{\wedge, \vee, \to, \neg\}$ from the superscript for convenience. The variety of *Sugihara algebras* $\mathbb{V}(\mathbf{Z})$ builds the algebraic semantics (see [21]) for the relevant logic *R-mingle* RM, i.e., for a set of terms $\Gamma \cup \{\varphi\}$, $\Gamma \vdash_{\mathrm{RM}} \varphi$ if and only if $\{\psi \approx \psi \to \psi : \psi \in \Gamma\} \models_{\mathbf{Z}} \varphi \approx \varphi \to \varphi$. The logic RM as well as the variety of Sugihara algebras have been studied intensively (see, e.g., [22, 35, 73, 20, 81]). Note that in particular the algebra $\mathbf{Z_3}^{\to\neg}$ generates the variety of multiplicative Sugihara algebras (see [73] for details):

Theorem 4.12 ([103], see also [73, Theorem 5.1])**.** *Let* $\mathsf{SA_m}$ *be the algebraic semantics of the* $\{\to, \neg\}$*-fragment of the logic* RM*, denoted* $\mathrm{RM_m}$*. Then* $\mathbb{V}(\mathsf{SA_m}) = \mathbb{V}(\mathbf{Z_3}^{\to\neg})$*.*

Moreover, $\mathbb{Q}(\mathbf{Z_3}^{\to\neg})$ provides algebraic semantics[5] for the logic $\mathrm{RM_m}$ extended by the modus-ponens-like “Avron-rule”

$$\varphi,\ (\varphi \to (\psi \to \psi)) \to (\varphi \to \psi) \ \ / \ \ \psi \tag{A}$$

Theorem 4.13 ([73, Lemma 5.4])**.** $\mathbb{Q}(\mathbf{Z_3}^{\to\neg})$ *builds the algebraic semantics of* $\mathrm{RM_m} + (\mathrm{A})$*.*

[5] Note that the multiplication $\cdot$ used in [73] can be defined by $x \cdot y := \neg(x \to \neg y)$.

$\to^{\mathbf{Z_3^e}}$	-1	0	1
-1	1	1	1
0	-1	0	1
1	-1	-1	1

$\to^{\mathbf{Z_4^e}}$	-2	-1	1	2
-2	2	2	2	2
-1	-2	1	1	2
1	-2	-1	1	2
2	-2	-2	-2	2

Figure 4.7: The tables for → of the algebras $\mathbf{Z_3^e}$ and $\mathbf{Z_4^e}$.

We study here some reducts of the Sugihara monoids $\mathbf{Z_3^e}$ and $\mathbf{Z_4^e}$ with universes $\{-1, 0, 1\}$ and $\{-2, -1, 1, 2\}$, respectively. The tables of the corresponding implications $\to$ are shown in Figure 4.7.

We list in Table 4.3 the results obtained when applying AdmAlgs to $\mathbf{Z_3^e}$ and $\mathbf{Z_4^e}$, respectively, while changing the underlying language. The algebras $\mathbf{Z_3}$ and $\mathbf{Z_3}^{\to\neg}$ are the only three element algebras of the list which are not structurally complete, since there are quasiequations which are admissible but not valid in the corresponding algebras. E.g., considering the truth table for the equation

$$y \to (x \to x) \;\approx\; (x \wedge \neg x) \wedge (y \wedge \neg y) \tag{4.6}$$

confirms that (4.6) is only satisfiable with $x = y = 0$ or $x = y = -1$. But it is not hard to see that there cannot be any $\{\wedge, \vee, \to, \neg\}$-term which always takes value 0 or -1, respectively. Hence (4.6) is not $\mathbf{Z_3}$-unifiable. So the quasiequation (4.7) is $\mathbf{Z_3}$-admissible, but not $\mathbf{Z_3}$-valid and $\mathbf{Z_3}$ is not structurally complete[6].

$$y \to (x \to x) \;\approx\; (x \wedge \neg x) \wedge (y \wedge \neg y) \quad\Rightarrow\quad x \;\approx\; z. \tag{4.7}$$

Note moreover that although $\mathbf{Z_3}$ and $\mathbf{Z_3}^{\to\neg}$ are not clone equivalent by Theorem 3.28 and the size of their free algebras, their minimal generating algebras are isomorphic when we define $\wedge$ and $\vee$ component-wise for $\mathbf{Z_3}^{\to\neg}$. In

[6] The argument also holds for $\mathbf{Z_3}^{\to\neg}$, since $(x \wedge \neg x) \wedge (y \wedge \neg y) = \neg(((x \to x) \to y) \to y)$.

Table 4.3: Admissibility for reducts of Sugihara monoids.

$\mathbf{A}$	$\|A\|$	Language	n	$\mathbf{F}(n)$	M	SC
$\mathbf{Z_3^e}$	3	$\wedge, \vee, \to, \neg, e$	1	9	3	sc
$\mathbf{Z_3}$	3	$\wedge, \vee, \to, \neg$	2	1296	6	asc
$\mathbf{Z_3^{\to\neg}}$	3	$\to, \neg$	2	264	6	asc
$\mathbf{Z_3^{\to}}$	3	$\to$	2	60	3	sc
$\mathbf{Z_3^{\to\neg e}}$	3	$\to, \neg, e$	1	5	3	sc
$\mathbf{Z_3^{\to e}}$	3	$\to, e$	1	5	3	sc
$\mathbf{Z_4^e}$	4	$\wedge, \vee, \to, \neg, e$	1	64	8	asc
$\mathbf{Z_4}$	4	$\wedge, \vee, \to, \neg$	2	20736	?	?
$\mathbf{Z_4^{\to\neg}}$	4	$\to, \neg$	2	264	6	no
$\mathbf{Z_4^{\to}}$	4	$\to$	2	60	3	no
$\mathbf{Z_4^{\to\neg e}}$	4	$\to, \neg, e$	1	18	6	no
$\mathbf{Z_4^{\to e}}$	4	$\to, e$	2	453	4	no

fact they are isomorphic to the product $\mathbf{Z_3} \times \mathbf{Z_2}$. On the other hand, $\mathbf{Z_3^{\to\neg e}}$ and $\mathbf{Z_3^{\to e}}$ are clone equivalent, since $\neg x = x \to e$ if $e = 0$.

It is remarkable that although the algebra $\mathbf{Z_4^{\to e}}$ is not structurally complete and not $\mathbf{Z_4^{\to e}}$-irreducible, nevertheless the algorithm AdmAlgs produces a four element algebra that is not isomorphic to $\mathbf{Z_4^{\to e}}$. For $\mathbf{Z_4^{\to}}$ we even obtain a three element algebra.

Even though the free algebra of $\mathbf{Z_4}$ is too big[7] to run AdmAlgs($\{\mathbf{Z_4}\}$) within TAFA, it is clear that $\mathbf{F_{Z_4}}(2)$ is the minimal generating free algebra for $\mathbb{Q}(\mathbf{F_{Z_4}}(\omega))$: Since $\mathbf{F_{Z_4}}(1)$ has four elements and is not isomorphic to $\mathbf{Z_4}$, it cannot be a generating algebra for $\mathbb{Q}(\mathbf{F_{Z_4}}(\omega))$ by Corollary 3.13. So we define a map $h\colon \{x, y\} \to Z_4$ by $h(x) := 1$, $h(y) := 2$. By the universal mapping property of $\mathbf{F_{Z_4}}(\omega)$ for $\mathbb{Q}(\mathbf{Z_4})$ this extends to a homomorphism $\overline{h}\colon \mathbf{F_{Z_4}}(\omega) \to \mathbf{Z_4}$ with $h(\neg x) := -1$ and $h(\neg y) := -2$. Hence $\overline{h}$ is surjective and by Corollary 3.13, $\mathbf{F_{Z_4}}(2)$ is the minimal generating free algebra for $\mathbb{Q}(\mathbf{F_{Z_4}}(\omega))$ as required.

[7] The size of $\mathbf{F_{Z_4}}(2)$ was calculated by the tool *UACalc* of Ralph Freese [38].

4.6 Summary

We remark that all the quasivarieties $\mathbb{Q}(\mathcal{K})$ studied so far had only one generating algebra, i.e., $|\mathcal{K}| = 1$. There are certainly interesting examples with more generating algebras (see, e.g., Example 4.14 below). Nevertheless every finitely generated quasivariety is also generated by one finite algebra, i.e., $\mathbf{F}_{\mathcal{K}}(\omega) = \mathbf{F}_{\mathbf{A}_1 \times \cdots \times \mathbf{A}_n}(\omega)$ for a finite set of finite $\mathcal{L}$-algebras $\mathcal{K} := \{\mathbf{A}_1, \ldots, \mathbf{A}_n\}$ by Corollary 2.16 and $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n \in \mathbb{P}(\mathcal{K})$ and $\mathbf{A}_i \in \mathbb{S}(\{\mathbf{A}_1 \times \cdots \times \mathbf{A}_n\})$ using the i-th projection homomorphism for $i \in \{1, \ldots, n\}$.

Example 4.14. *Consider the two chains* $\mathbf{C_2} := \langle\{\bot, \top\}, \wedge, \vee, \neg, c\rangle$ *and* $\mathbf{C_3} := \langle\{\bot, e, \top\}, \wedge, \vee, \neg, c\rangle$ *where* $\neg$ *swaps* $\bot$ *and* $\top$ *and leaves e fixed and* $c^{\mathbf{C_2}} := \top$, $c^{\mathbf{C_3}} := e$. *Individually, these algebras are structurally complete. However, applying* AdmAlgs *to* $\mathcal{K} := \{\mathbf{C_2}, \mathbf{C_3}\}$*, we find that* $\mathcal{K}$ *is not structurally complete: both* $\mathbf{C_2}$ *and* $\mathbf{C_3}$ *are homomorphic images of the sixteen element free algebra* $\mathbf{F}_{\mathcal{K}}(1)$*, and the minimal generating set for* $\mathbb{Q}(\mathbf{F}_{\mathcal{K}}(\omega))$ *consists of a single four element algebra.*

Table 4.4 summarizes the results (without the lattices of Table 4.1), ordered by the cardinalities of the algebras (first priority) and their free algebras (second priority).

Table 4.4: Algebras for checking admissibility. The column "n" lists the number of generators needed to generate $\mathbb{Q}(\mathbf{F}_{\mathbf{A}}(\omega))$, "$\mathbf{F}_{\mathbf{A}}(n)$" the cardinality of the minimal generating free algebra, "M" the cardinalities of the minimal generating set for $\mathbb{Q}(\mathbf{F}_{\mathbf{A}}(n))$ and "SC" whether $\mathbf{A}$ is structurally complete ("sc"), almost structurally complete ("asc") or none of the two ("no").

$\mathbf{A}$	$\vert A\vert$	Language	Quasivariety $\mathbb{Q}(\mathbf{A})$	n	$\mathbf{F}(n)$	M	SC
$\mathbf{B_2}$	2	$\wedge, \vee, \neg, \bot, \top$	$\mathbb{Q}(\mathbf{B_2})$ (Exs 2.4,3.19)	0	2	2	sc
$\mathbf{Z_3^{\to\neg e}}$	3	$\to, \neg, e$	$\mathbb{Q}(\mathbf{Z_3^{\to\neg e}})$ (Sec. 4.5)	1	5	3	sc
$\mathbf{Z_3^{\to e}}$	3	$\to, e$	$\mathbb{Q}(\mathbf{Z_3^{\to e}})$ (Sec. 4.5)	1	5	3	sc
$\mathbf{C_3}$	3	$\wedge, \vee, \neg, \bot, \top$	Kleene algebras (Sec. 4.4)	1	6	4	no
$\mathbf{PCL}_1$	3	$\wedge, \vee, ^*, \bot, \top$	Stone algebras (Sec. 4.3)	1	6	3	sc
$\mathbf{G_9}$	3	$*$	$\mathbb{Q}(\mathbf{G_9})$ (Sec. 3.8)	2	7	4	no
$\mathbf{S}$	3	$\supset, \neg$	Algebras for P^1 (Sec. 5.3)	1	9	9	no
$\mathbf{Z_3^e}$	3	$\wedge, \vee, \to, \neg, e$	$\mathbb{Q}(\mathbf{Z_3^e})$ (Sec. 4.5)	1	9	3	sc
$\mathbf{G_{106}}$	3	$\circ$	$\mathbb{Q}(\mathbf{G_{106}})$ (Ex. 3.14)	2	10	2,2	no
$\mathbf{L_3}$	3	$\to, \neg$	Algebras for Ł$_3$ (Ex. 3.26)	1	12	6	asc
$\mathbf{L_3^{\to}}$	3	$\to$	Algebras for Ł$_3^{\to}$ (Ex. 3.26)	2	40	3	sc
$\mathbf{Z_3^{\to}}$	3	$\to$	Algebras for $\mathrm{RM}^{\to}$ (Sec. 4.5)	2	60	3	sc
$\mathbf{C_3^{\ell}}$	3	$\wedge, \vee, \neg$	Kleene lattices (Sec. 4.4)	2	82	4	no
$\mathbf{Z_3^{\to\neg}}$	3	$\to, \neg$	Algebras for $\mathrm{RM}^{\to\neg}$ (Sec. 4.5)	2	264	6	asc
$\mathbf{Z_3}$	3	$\wedge, \vee, \to, \neg$	$\mathbb{Q}(\mathbf{Z_3})$ (Sec. 4.5)	2	1296	6	asc
$\mathbf{P}$	4	$*$	$\mathbb{Q}(\mathbf{P})$ (Ex. 3.22)	2	6	3	sc
$\mathbf{Z_4^{\to\neg e}}$	4	$\to, \neg, e$	Algebras for $\mathrm{RM}^{\to\neg\mathrm{e}}$ (Sec. 4.5)	1	18	6	no
$\mathbf{Z_4^{\to}}$	4	$\to$	$\mathbb{Q}(\mathbf{Z_4^{\to}})$ (Sec. 4.5)	2	60	3	no
$\mathbf{Z_4^e}$	4	$\wedge, \vee, \to, \neg, e$	$\mathbb{Q}(\mathbf{Z_4^e})$ (Sec. 4.5)	1	64	8	asc
$\mathbf{D_4^{\ell}}$	4	$\wedge, \vee, \neg$	De Morgan lattices (Sec. 4.4)	2	166	8	asc
$\mathbf{D_4}$	4	$\wedge, \vee, \neg, \bot, \top$	De Morgan algebras (Sec. 4.4)	2	168	10	no
$\mathbf{Z_4^{\to\neg}}$	4	$\to, \neg$	$\mathbb{Q}(\mathbf{Z_4^{\to\neg}})$ (Sec. 4.5)	2	264	6	no
$\mathbf{Z_4^{\to e}}$	4	$\to, e$	Algebras for $\mathrm{RM}^{\to\mathrm{e}}$ (Sec. 4.5)	2	453	4	no
$\mathbf{Z_4}$	4	$\wedge, \vee, \to, \neg$	$\mathbb{Q}(\mathbf{Z_4})$ (Sec. 4.5)	2	20736	?	?
$\mathbf{PCL}_2$	5	$\wedge, \vee, ^*, \bot, \top$	$\mathbb{Q}(\mathbf{PCL}_2)$ (Sec. 4.3)	1	7	5	sc
$\mathbf{PCL}_3$	9	$\wedge, \vee, ^*, \bot, \top$	$\mathbb{Q}(\mathbf{PCL}_3)$ (Sec. 4.3)	2	625	19	no
$\mathbf{PCL}_4$	17	$\wedge, \vee, ^*, \bot, \top$	$\mathbb{Q}(\mathbf{PCL}_4)$ (Sec. 4.3)	2	626	?	no

Chapter 5

TAFA - A Toolbox for Finite Algebras

This chapter presents TAFA (standing for "**T**ool for **A**dmissibility in **F**inite **A**lgebras"), an implementation of the algebraic tools and algorithms from Chapter 3. Nearly all the calculations made in this thesis, in particular, those in Chapter 4, were made using TAFA[1]. We implemented TAFA using Delphi XE2, a development environment for Object Pascal. It is currently compiled for Windows, but can easily be used on Mac and Linux using an emulator such as Wine[2]. Many ideas concerning the data structures and basic operations are taken from the source code of the Algebra Workbench (see [104, 91]). An executable file of TAFA is available from `https://sites.google.com/site/admissibility/`.

Sections 5.1 and *5.2* provide an overview of the features offered by TAFA. *Section 5.3* then gives an insight into the look-and-feel of the tool by guiding the reader through an example session related to the paraconsistent Sette algebra, which was introduced in [87].

[1] Note, however, that to calculate the size of a free algebra with more than 1500 elements (without having the corresponding operation tables) we used UACalc [38].

[2] Wine can be downloaded from `http://www.winehq.org/`.

5.1 Basic Operations

In order to use TAFA the user should first either define the algebras of interest in TAFA or load some predefined (see *File > Predefined algebras*[3]) or previously stored algebras (from a file). Defining a new algebra (see *File > New algebra*) includes giving it a name, labeling the elements and defining the operations. The user can easily rename, sort, delete or edit algebras, their elements and operations or add some comment by either double clicking the corresponding field of the grid in the main window or using the menu *Edit*. The main window of TAFA contains a list showing for each algebra its name, cardinality, the names and arities of its operations and any comments.

TAFA can save the selected or chosen[4] algebras as a binary file (*.fab, fast, illegible), as a text file (*.fai, slower, legible) or, if the algebra is a partially ordered set with an operation "meet", to a *.osf file which can be read by the Algebra Workbench to visualize the corresponding Hasse diagram. TAFA loads algebras from fab- or fai-files and is able to copy or remove algebras in the main window (menu *File*). The algebras are stored as the data type TAlgebra within TAFA, which is connected to lists of the type TAlgebraUniverse and TOperationList providing further procedures and objects. Once the algebras of interest are defined in the main window, the basic operations of universal algebra described below can be performed.

The menu item *Tools > Morphisms* opens a dialogue window where the user can choose a domain $\mathbf{A}_1$ and a codomain $\mathbf{A}_2$ (of the same language) from the list of defined algebras. It is possible to choose whether to calculate all homomorphisms between $\mathbf{A}_1$ and $\mathbf{A}_2$ or only those that are surjective, injective or bijective. When the button "Calculate" is pressed, TAFA lists the homomorphisms satisfying the chosen criteria. Double-clicking on an entry of the list shows the mappings from elements of $\mathbf{A}_1$ to elements of $\mathbf{A}_2$. Using the *Tools* menu of this dialogue window it is also possible to add the

[3] Navigation through the menus is denoted here by *Menu > Menu item*.

[4] We say that an entry of a list, e.g., an algebra in the main window, is *selected* if it is highlighted, and *chosen* if the appropriate check box is checked.

homomorphic image as a new algebra to the main window or to save the mapping information to a text file.

The menu item *Tools > Subalgebras* opens a dialogue window which lists all the subalgebras of the active algebra. The subalgebras are stored as entities of TAlgebraUniverse within this dialogue window to save time (there is no need to build up the operation tables), but it is possible to add the checked subalgebras as new algebras to the main window using the menu *Tools* of the dialogue window. The *Options* menu of the dialogue window offers the possibility to (heuristically) first list the smaller and then the bigger algebras by first calculating the subalgebras generated by zero or one element, storing their sizes and then trying to combine the given generators in such a way that the subalgebras generated are potentially small.

Tools > Generating subalgebra opens a dialogue window where the user can choose some elements $a_1, \ldots, a_k$ of the active algebra $\mathbf{A}$. TAFA then calculates the unique subalgebra of $\mathbf{A}$ generated by the elements $a_1, \ldots, a_k$ and adds it as a new algebra to the main window.

Having defined algebras $\mathbf{A}_1, \ldots, \mathbf{A}_n$ of the same language in the main window of TAFA, the user can calculate the direct product $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ using *Tools > Direct product*. Specifying some $k \in \mathbb{N}$ with *Tools > Direct power*, the direct power $\mathbf{A}^k$ of the selected $\mathcal{L}$-algebra $\mathbf{A}$ is calculated.

Tools > Congruences opens a dialogue window which lists the congruences $\mathrm{Con}(\mathbf{A})$ of the selected $\mathcal{L}$-algebra $\mathbf{A}$ in the main window. Selecting a congruence in the list shows the congruence classes on the right side. The dialogue window menu *Tools* lets the user store the congruence lattice $\mathbf{Con}(\mathbf{A})$ as a new algebra (with the lattice operations $\wedge$ and $\vee$ as language) to the main window. It is also possible to quotient the active structure with the selected congruence or to save the congruence information to a text file.

If the set $\mathcal{K} := \{\mathbf{A}_1, \ldots, \mathbf{A}_n\}$ of $\mathcal{L}$-algebras is chosen in TAFA, the menu item *Tools > Free algebra* lets the user specify a natural number $n \in \mathbb{N}$ and TAFA calculates the free algebra $\mathbf{F}_{\mathcal{K}}(n)$. There is also the possibility to search for the smallest generating free algebra for $\mathcal{K}$.

5.2 Advanced Features

Tools > Minimal Generating Set (MinGenSet) calculates $\textsc{MinGenSet}(\mathcal{K})$ for the chosen set of $\mathcal{L}$-algebras $\mathcal{K}$ in TAFA (see Algorithm 3.1).

Let $\mathbf{A}$ be an $\mathcal{L}$-algebra and $\mathbf{F_A}(n)$ the minimal generating free algebra for $\mathbb{Q}(\mathbf{F_A}(\omega))$. We call an $\mathcal{L}$-algebra $\mathbf{B}$ an *admissibility algebra*, if $\mathbf{B} \in \mathbb{S}(\mathbf{F_A}(n))$ and $\mathbf{A} \in \mathbb{H}(\mathbf{B})$ (see Corollary 3.13).

Given a set $\mathcal{K}$ of $\mathcal{L}$-algebras chosen in TAFA, the user selects the appropriate free algebra or lets the program find the smallest generating free algebra for $\mathcal{K}$ with *Tools > Admissibility algebra*. The menu *Options* of the dialogue window for calculating admissibility algebras then lets the user choose whether to search for admissibility algebras from smaller to larger or with the usual algorithm of searching for subalgebras (which is independent of their cardinalities). Although the latter is much quicker for small algebras, there are some cases where the heuristic method performs faster. Once the admissibility algebra is stored as a new algebra in the main window, the user can calculate $\textsc{MinGenSet}(\mathcal{K})$ as needed.

The menu *Check* enables the user to check whether the selected $\mathcal{L}$-algebra $\mathbf{A}$ is subdirectly irreducible, $\mathbb{Q}(\mathbf{A})$-subdirectly irreducible (see Corollary 2.11), structurally complete (see Theorem 3.23) or almost structurally complete (see Theorem 3.27).

5.3 Example Session

In this section we guide the reader through an example TAFA session, trying to find the minimal generating set for the quasivariety $\mathbb{Q}(\mathbf{F_S}(\omega))$, where $\mathbf{S}$ is the *Sette algebra* generating the algebraic semantics $\mathbb{Q}(\mathbf{S})$ for the paraconsistent *Sette logic* P^1 (see [102, 87]).

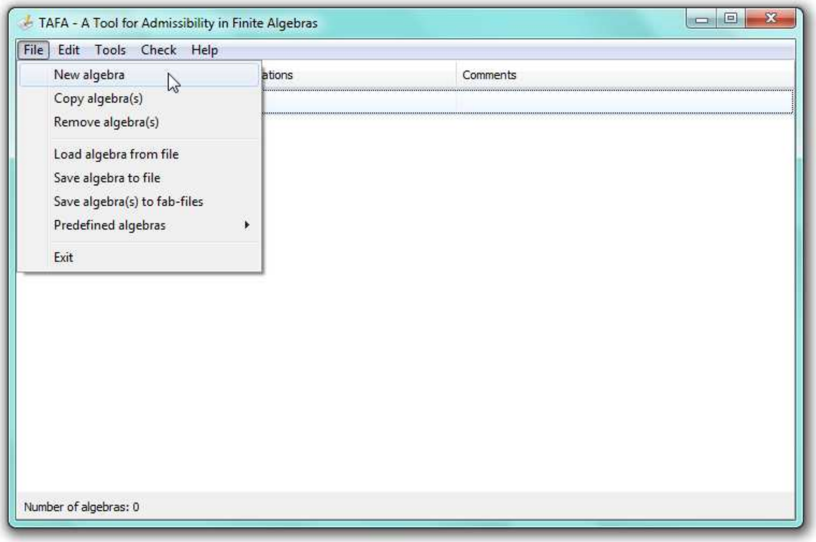
The first step is of course to define the algebra $\mathbf{S}$ within TAFA. $\mathbf{S}$ has three elements $\{0, 0.5, 1\}$, a binary operation $\supset$ and a unary operation $\neg$

defined as follows:

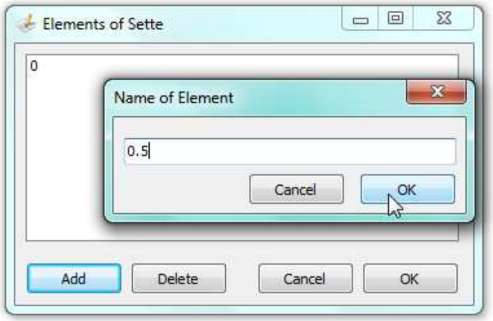
$\supset$	0	0.5	1
0	1	1	1
0.5	0	1	1
1	0	1	1

$\neg$	
0	1
0.5	1
1	0

We first open TAFA, select the menu item *File > New algebra* (see figure below), enter the name "Sette" into the opened text field and hit "OK". The algebra is now defined, but has no elements and operations yet.



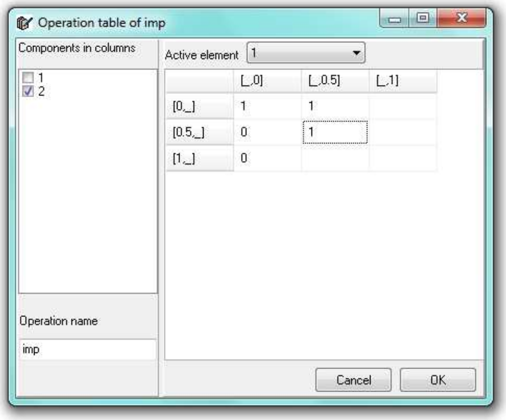
To define the universe select *Edit > Edit elements* or double click onto the "0" in the column called "Card". A dialogue window called "Elements of Sette" opens. Define the elements 0, 0.5 and 1 with the appropriate buttons, then hit "OK" (see figure below). The universe is now defined.



0

0.5

To define the operations select *Edit* > *Edit operations* or double click onto the operations cell of the grid. A dialogue window called "Operations of Sette" opens. Click "Add", then name the first operation (e.g., "imp") and fix the arity (here two). After confirming with "OK" we see a row displayed in red in the grid of this dialogue window, which means that the operation is defined but there are still undefined values. Now we either click the button "Edit" or double click on the line of the operation "imp" to define the table of values for ⊂. In the opened window called "Operation table of imp" we can either enter the values by typing them on the keyboard or by selecting them in the drop down menu called "Active element" and then double clicking on the desired coordinate of the table (see figure below). When the table is completely defined we confirm with "OK" and go through the same procedure to define the operation ¬.



Components in columns

□ 1
☑ 2

Active element 1

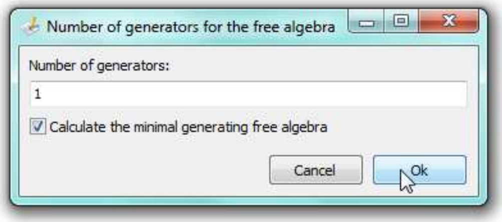
	[_,0]	[_,0.5]	[_,1]
[0,_]	1	1	
[0.5,_]	0	1	
[1,_]	0		

Operation name

imp

By either selecting *Edit* > *Edit comment* or double clicking on the comment cell in the grid we can also add a comment if we like. Now the algebra $\mathbf{S}$ is completely defined and ready to use. With the menu item *File > Save algebra to file* we can save the algebra into a file for later use.

In order to find the minimal generating set for $\mathbb{Q}(\mathbf{F_S}(\omega))$, we first need to calculate the minimal generating free algebra for this quasivariety. The menu item *Tools > Free algebra* opens a dialogue window called "Number of generators for the free algebra", where we check the box "Calculate the minimal generating free algebra" and confirm with "OK":

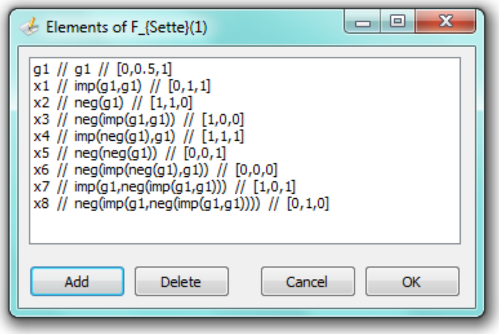


Number of generators for the free algebra

Number of generators: 1

☑ Calculate the minimal generating free algebra

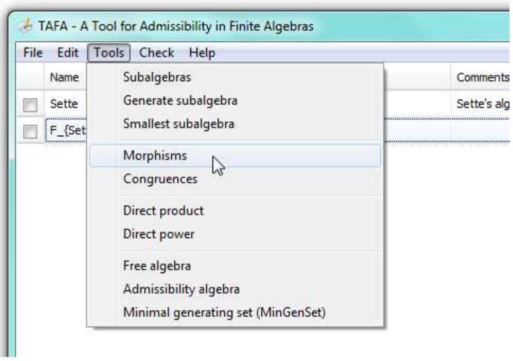
It turns out that the minimal generating free algebra for $\mathbb{Q}(\mathbf{F_S}(\omega))$ is $\mathbf{F_S}(1)$ which has nine elements. By double clicking on the "Card" cell we get a list of representatives of the equivalence classes of the free algebra showing how the elements of the free algebra were generated (see figure below).



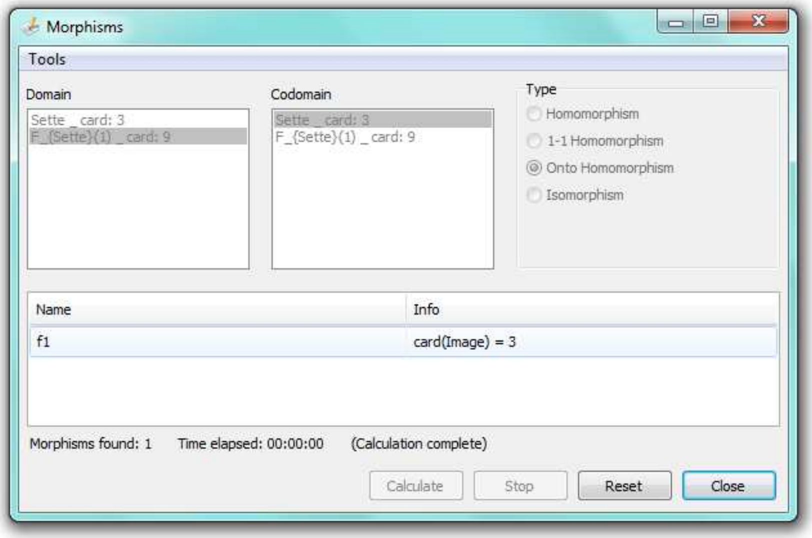
Elements of F_{Sette}(1)

```
g1 // g1 // [0,0.5,1]
x1 // imp(g1,g1) // [0,1,1]
x2 // neg(g1) // [1,1,0]
x3 // neg(imp(g1,g1)) // [1,0,0]
x4 // imp(neg(g1),g1) // [1,1,1]
x5 // neg(neg(g1)) // [0,0,1]
x6 // neg(imp(neg(g1),g1)) // [0,0,0]
x7 // imp(g1,neg(imp(g1,g1))) // [1,0,1]
x8 // neg(imp(g1,neg(imp(g1,g1)))) // [0,1,0]
```

Suppose that we would like to know the definition of a homomorphism from the free algebra $\mathbf{F_S}(1)$ onto $\mathbf{S}$ (note that there must be at least one such homomorphism since $\mathbf{F_S}(1)$ is the minimal generating free algebra for $\mathbb{Q}(\mathbf{F_S}(\omega))$). Calculating morphisms is done through the menu item *Tools > Morphisms*:



The morphisms window opens, where we choose the domain, codomain and type of homomorphism we search for. Clicking "Calculate" shows us that there is only one surjective homomorphism from the free algebra onto **S**.



Domain: Sette _ card: 3 / F_{Sette}(1) _ card: 9

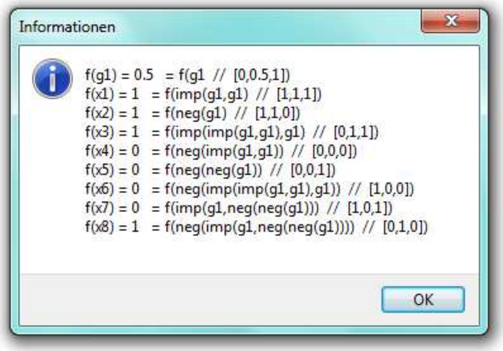
Codomain: Sette _ card: 3 / F_{Sette}(1) _ card: 9

Type:
○ Homomorphism
○ 1-1 Homomorphism
● Onto Homomorphism
○ Isomorphism

Name	Info
f1	card(Image) = 3

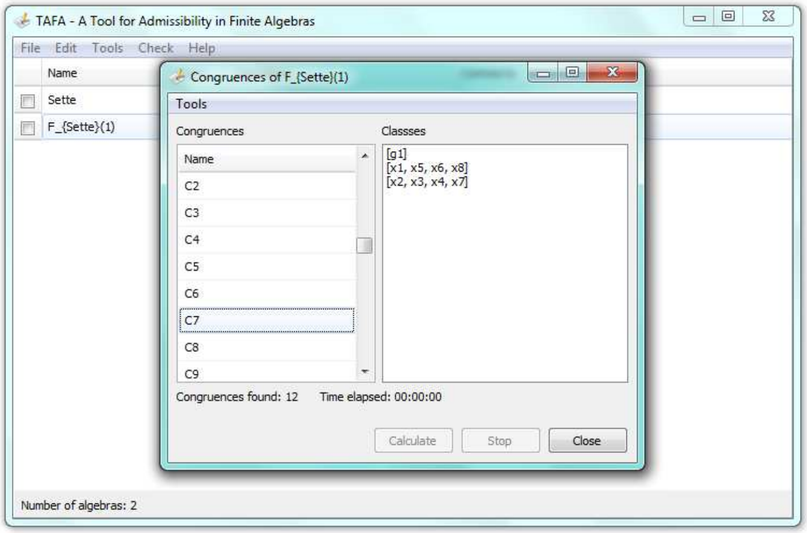
Morphisms found: 1   Time elapsed: 00:00:00   (Calculation complete)

Double clicking the corresponding row presents the mapping:



```
f(g1) = 0.5  = f(g1 // [0,0.5,1])
f(x1) = 1  = f(imp(g1,g1) // [1,1,1])
f(x2) = 1  = f(neg(g1) // [1,1,0])
f(x3) = 1  = f(imp(imp(g1,g1),g1) // [0,1,1])
f(x4) = 0  = f(neg(imp(g1,g1)) // [0,0,0])
f(x5) = 0  = f(neg(neg(g1)) // [0,0,1])
f(x6) = 0  = f(neg(imp(imp(g1,g1),g1)) // [1,0,0])
f(x7) = 0  = f(imp(g1,neg(neg(g1))) // [1,0,1])
f(x8) = 1  = f(neg(imp(g1,neg(neg(g1)))) // [0,1,0])
```

The menu item *Tools* > *Congruences* opens a dialogue window called "Congruences of F_{Sette}(1)". Calculating the congruences by clicking "Calculate" shows that there are only twelve congruences in $\mathrm{Con}(\mathbf{F_S}(1))$ (see figure below) and hence we could directly apply the algorithm MinGenSet to $\{\mathbf{F_S}(1)\}$ to get (in a reasonable amount of time) the minimal generating set for the quasivariety $\mathbb{Q}(\mathbf{F_S}(\omega))$. But for the sake of the example, let us suppose that we want to apply the algorithm AdmAlgs.


TAFA - A Tool for Admissibility in Finite Algebras
File
Edit
Tools
Check
Help
Congruences of F_{Sette}(1)
Tools
Congruences
Classes
Calculate
Stop
Close

Name
□ Sette
□ F_{Sette}(1)

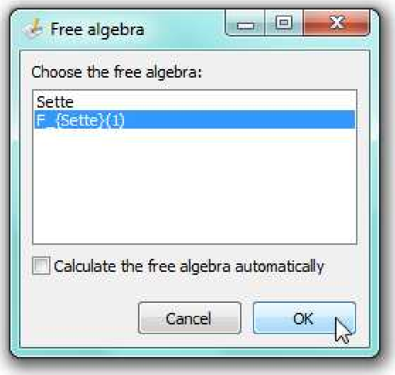
Name
C2
C3
C4
C5
C6
C7
C8
C9

[g1]
[x1, x5, x6, x8]
[x2, x3, x4, x7]

Congruences found: 12   Time elapsed: 00:00:00

Number of algebras: 2

For this we choose the menu item *Tools* > *Admissibility algebra* (note that we need to choose the algebra "Sette" first in the main grid) and then select the free algebra called "F_{Sette}(1)" in the list that pops up:



Choose the free algebra:

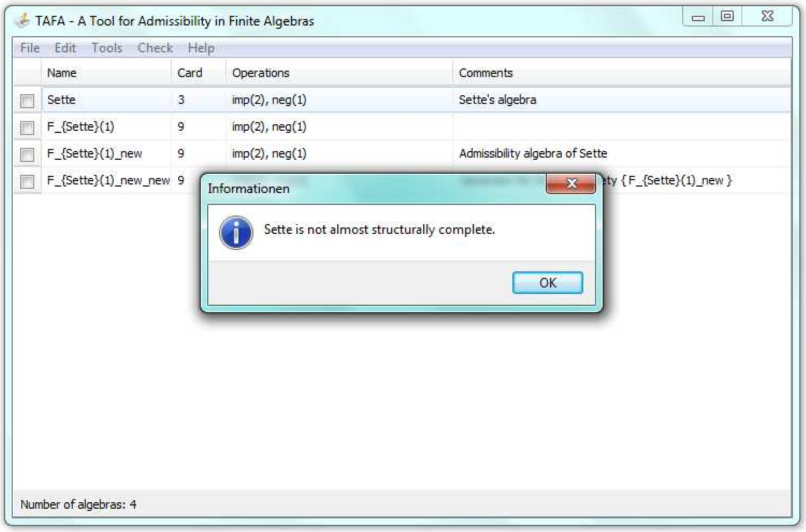
Sette
F_{Sette}(1)

□ Calculate the free algebra automatically

Clicking “OK” opens a dialogue window called “Admissibility algebras for Sette”. We only deselect “Chain of subalgebras (found subalgebras as new starting point)” of the menu *Options* if we want to find all the subalgebras of the minimal generating free algebra (here $\mathbf{F_S}(1)$) which are prehomomorphic images of the generating algebra (here $\mathbf{S}$). For small algebra it is much faster to have the option “From smaller to bigger algebras (heuristic)” deselected. So we start the calculation by clicking the button “Calculate” and then see that the new algebra in the main grid has nine elements like the free algebra and hence must be the free algebra itself. To finish our search we have finally to apply the algorithm MinGenSet to this algebra with the menu item *Tools > Minimal generating set (MinGenSet)* since it could be that there are smaller algebras generating the same quasivariety which are not prehomomorphic images of $\mathbf{S}$. This is not the case here for $\mathbf{F_S}(1)$, hence

$$\textsc{MinGenSet}(\mathbf{F_S}(\omega)) \quad = \quad \{\ \mathbf{F_S}(1)\ \}.$$

Note that we can check that $\mathbf{S}$ is not almost structurally complete (see figure below) with the use of *Check > Almost structural completeness*.



	Name	Card	Operations	Comments
□	Sette	3	imp(2), neg(1)	Sette's algebra
□	F_{Sette}(1)	9	imp(2), neg(1)	
□	F_{Sette}(1)_new	9	imp(2), neg(1)	Admissibility algebra of Sette
□	F_{Sette}(1)_new_new	9		ty {F_{Sette}(1)_new }

Sette is not almost structurally complete.

Number of algebras: 4

Chapter 6

Concluding Remarks

This chapter summarizes the results obtained in this thesis and explains how they fit into the existing theory of admissible rules in universal algebra and finite-valued logics (*Section 6.1*). We conclude the thesis by sketching some ideas for future research into questions related to this work (*Section 6.2*).

6.1 Contribution of the Thesis

Our primary goal in this thesis was to investigate admissibility in finitely generated quasivarieties and finite-valued logics. There has been a substantial amount of research into admissibility for intermediate and modal logics (see, e.g., [99, 43, 44, 56, 60]), but a general theory of admissibility for finite-valued logics was, before the work reported here, lacking. A central aim of the thesis was to establish general algorithms to check whether a given quasiequation is admissible in a finitely generated quasivariety $\mathcal{Q}$. This is the case if and only if it is valid in the free algebra $\mathbf{F}_{\mathcal{Q}}(n)$ where n is the maximum of the cardinalities of the generating algebras (see Theorem 3.9), but free algebras are often quite big even for a small number of generators.

A first step towards addressing this issue was the introduction of *minimal generating sets* for any finitely generated quasivariety $\mathcal{Q}$ (see Algorithm 3.1), i.e., smallest (with respect to the standard multiset ordering) sets of algebras $\mathcal{K}$ such that $\mathcal{Q} = \mathbb{Q}(\mathcal{K})$. Minimal generating sets are unique up to isomorphism (see Theorem 3.3) and provide a useful general tool for investigating finitely generated quasivarieties in universal algebra. The algorithm MinGenSet provides here a possibility of answering the problem of checking admissibility in finitely generated quasivarieties $\mathbb{Q}(\mathcal{K})$, since MinGenSet($\{\mathbf{F}_{\mathbb{Q}(\mathcal{K})}(n)\}$) (where n is the maximum of the cardinalities of the algebras in $\mathcal{K}$) returns the minimal generating set for $\mathbb{Q}(\mathbf{F}_{\mathbb{Q}(\mathcal{K})}(\omega))$.

However, finding a minimal generating set is not generally feasible for larger algebras. A further important ingredient of our approach is therefore Theorem 3.11 since it describes how to replace the generating free algebra with a smaller algebra while making sure that the new (sub)algebra still generates the same quasivariety $\mathbb{Q}(\mathbf{F}_{\mathcal{Q}}(\omega))$. Results from Birkhoff (see Lemma 2.12) and Rybakov (see Theorem 2.18) allow this theorem to be applied to finitely generated quasivarieties. Finally, the procedure AdmAlgs (see Algorithm 3.2) joins the two ideas of finding the minimal generating set and reducing the size of the generating algebras by searching for suitable subalgebras, providing a general algorithm for checking admissibility (see Theorem 3.15). Table 4.4 lists the remarkable reductions of the cardinalities from the appropriate free algebras to the results of the algorithm AdmAlg. These results contribute to the study of some well known classes of algebras, including the varieties of De Morgan and Kleene algebras.

Theorem 3.7 connects unifiability of a set of equations with satisfiability of this set in a subalgebra of a finite free algebra. Hence unifiability is decidable and can be checked in the (usually small) smallest subalgebra of the free algebra. Theorem 3.23 characterizes structural completeness using the algorithm MinGenSet. This implementable (see Chapter 5) characterization provides a nice alternative to known proof techniques for establishing structural completeness in finitely generated quasivarieties or finite-valued logics such as Theorem 3.17 or “Prucnal’s trick” (see [85]). Theorem 3.25 provides a characterization for almost structural completeness similar to Theorem 3.16 for structural completeness. This has been used to describe almost structural completeness in terms of the algorithm MinGenSet in Theorem 3.27.

Theorem 3.28 can save a lot of calculation time since it ensures that free algebras and minimal generating sets of clone equivalent algebras are isomorphic (up to translations inside their clone of operations) and hence we only need to run our algorithms once if the operations of two algebras are inter-definable. Theorem 3.30 transfers Theorem 3.11 into the logical setting, i.e., for a given logic L, it characterizes the admissibility of a rule $\Gamma \ / \ \varphi$ by the validity in another logic L'.

These theoretical results and obtained algorithms for admissibility in finitely generated quasivarieties have allowed us to investigate when admissibility and validity diverge in some basic cases: Theorem 4.1 provides a new proof for the fact that every two element algebra is structurally complete (compare [89, Corollary 1]). Section 4.2 comprehensively investigates the minimal generating sets for $\mathbb{Q}(\mathbf{F_G}(\omega))$ for all three element groupoids $\mathbf{G}$. We have also used the obtained tools to investigate axiomatization problems: Theorems 4.7, 4.5 and 4.9 provide bases for the admissible quasiequations of the quasivarieties of Kleene algebras, Kleene lattices and De Morgan lattices, respectively. Moreover, Theorem 4.11 presents a "basis" for the admissible quasiequations of the variety of De Morgan algebras which does not consist only of quasiequations, but also includes the proper clause (4.3) (in contrast to more recent work [26], where a proper basis is found using natural dualities).

6.2 Outlook

For any finitely generated quasivariety $\mathcal{Q}$, we can find a minimal set $\mathcal{K}$ (with respect to the standard multiset ordering) of algebras to check admissibility in $\mathcal{Q}$ (by checking validity in $\mathcal{K}$). Nevertheless, the algorithm AdmAlgs is not feasible for arbitrary input size because of the complexity of the tasks involved. The bottlenecks are in particular: generating the free algebra, calculating the congruence lattice (or, equivalently, checking homomorphisms) and calculating subalgebras. Checking, e.g., whether $\mathbf{A} \in \mathbb{H}(\mathbf{B})$ or $\mathbf{A} \in \mathbb{S}(\mathbf{B})$ is NP-hard for finite algebras $\mathbf{A}$ and $\mathbf{B}$ (see, e.g., [13, 53]), but running through *all* subalgebras or congruences is EXPTIME-hard in general[1]. The following ideas might be used to obtain faster algorithms:

- Only construct small subalgebras of the free algebra (heuristically) to check if they generate the whole quasivariety, i.e., if they are pre-homomorphic images of the initial algebras.

- Do not calculate the whole lattice of congruences in MinGenSet. Intuitively, we are only interested in the bottom region of the congruence lattice $\mathrm{Con}(\mathbf{A})$ if we want to check whether $\mathbf{A}$ is $\mathbb{Q}(\mathbf{A})$-subdirectly irreducible (see Lemma 3.4 and Corollary 2.11(b)).

- Improve the algorithm for generating subalgebras used for the heuristic procedure where we first check smaller, then bigger subalgebras of the free algebra in AdmAlgs. Construct a directed graph to store (based on the operation tables of the operations of the algebra) which elements are "reachable" by which elements. E.g., in the Kleene lattice $\mathbf{C}_{\mathbf{3}}^{\mathbf{L}}$ (see Section 4.4), 1 is reachable by -1, since $\neg - 1 = 1$, but 0 is not.

- Search for convenient subalgebras of the free algebra top-down rather than bottom-up by systematically excluding elements. This would be particularly helpful if our conjecture is true, that the "admissibility algebras" are always on the top of the lattice of subuniverses. I.e., given a finitely generated quasivariety $\mathcal{Q}$ and its minimal generating free algebra $\mathbf{F}_{\mathcal{Q}}(n)$, then $\mathbb{Q}(\mathbf{B}) = \mathbb{Q}(\mathbf{F}_{\mathcal{Q}}(n))$ for some algebra $\mathbf{B}$ implies $\mathbb{Q}(\mathbf{B}') = \mathbb{Q}(\mathbf{F}_{\mathcal{Q}}(n))$ for all algebras $\mathbf{B}'$ in the upset of $\mathbf{B}$ inside the lattice of subuniverses (compare Figure 3.1 as an example).

- Improve the algorithms by restricting attention to certain classes of algebras. E.g., if we consider congruence-distributive algebras, we

[1] See [14] for investigations on the size of free algebras.

could use a polynomial time algorithm to find a subdirect decomposition (see [33]) instead of a $\mathcal{Q}$-subdirect decomposition, since every $\mathcal{Q}$-subdirectly irreducible algebra is subdirectly irreducible for a congruence-distributive quasivariety[2] $\mathcal{Q}$ (see [37, Theorem 2.3]).

- Try to prohibit redundant calculations as with Theorem 3.28 by considering the type sets (containing the types *unary, affine, Boolean, lattice or semilattice*) of tame congruence theory (see [55]) for the given finite algebra (see, e.g., [68] for complexity studies in universal algebra with respect to tame congruence theory).

The usability of TAFA could also be improved. E.g., rather than calculating the free algebra within TAFA, we could implement an interface to the tool UACalc, since this tool already implemented many optimization tricks like “thinning the coordinates”. Also, it could be convenient to have import (export) possibilities from (to) other formats suchas LaTeX, UACalc, AWB or Sage. Moreover, it could be helpful to save the calculated parts of the free algebra if the calculation is aborted, e.g., to generate the fully defined subalgebras of this part of the free algebra.

Finally, there remain numerous open problems and directions in the theoretical framework of admissible rules that might be tackled using the ideas and tools developed in this thesis. In particular:

- The present work only considers propositional logics. Could the results obtained in this thesis be transferred to predicate logics? Note, however, that admissibility is far from being understood even in the case of classical predicate logic.

- How could we extend the work to locally finite quasivarieties, i.e., where finitely generated algebras are finite, or even infinite algebras? The problem of non-finitely generated quasivarieties is certainly that either

[2] A quasivariety $\mathcal{Q}$ is called *congruence-distributive*, if for every algebra $\mathbf{A} \in \mathcal{Q}$ the lattice $\mathrm{Con}_{\mathcal{Q}}(\mathbf{A})$ is distributive. By [37, Proposition 2.1] this is the case if and only if for each $n \in \mathbb{N}$, $\mathrm{Con}_{\mathcal{Q}}(\mathbf{F}_{\mathcal{Q}}(n))$ is distributive.

only infinitely many algebras generate the quasivariety or some of the generating algebras have infinite cardinalities. So using TAFA to investigate these algebras or hoping for the presented algorithms to terminate will not work. Moreover, we would have to consider ultraproducts in this case, since $\mathbb{Q}(\mathcal{K}) = \mathbb{ISPP}_{\mathbb{U}}(\mathcal{K})$ in general. Nevertheless, it still makes sense to concentrate on $\mathbb{Q}(\mathcal{K})$-subdirectly irreducible algebras (to find minimal generating sets) since they generate the quasivariety (see Theorem 2.8). Also methods for checking, e.g., structural completeness, like finding an embedding into the free algebra (see Theorem 3.17) extend to infinite algebras.

- Finding admissible rules with some algorithm could be helpful in finding bases of admissible rules automatically for a given quasivariety which is not structurally complete. One of the motivations for investigating admissibility is to obtain quasiequations that can be used to tune up proof systems (shortening derivations, constraining proof search, ...) for these algebras. A further step could then also be to find potentially useful quasiequations for a given finite algebra. Note however, that there are finite algebras which do not have a finite basis of admissible rules (see [73, Corollary 5.12]).

- We only treated admissible *quasiequations* here except for the clause

$$x \vee y \approx \top \quad \Rightarrow \quad x \approx \top, \quad y \approx \top,$$

  which is admissible in De Morgan algebras (see Section 4.4), i.e., whenever $\sigma(x) \vee \sigma(y) \approx \top$ is valid in all De Morgan algebras for any substitution σ, then either $\sigma(x) \approx \top$ or $\sigma(y) \approx \top$ is valid in all De Morgan algebras. We would therefore like to investigate how to adapt our algorithms to treat such "multiple-conclusion rules" (see also [26]).

- A logic is said to be *hereditarily structurally complete* if all of its extensions are structurally complete. Algebraically, this corresponds to the fact that every proper subquasivariety is a variety. We would like to investigate whether there is a characterization as for structural and almost structural completeness in terms of minimal generating sets for this property (see Theorems 3.23, 3.27).

Appendix A

List of Three Element Groupoids

Table A.1 lists all three element pairwise not clone equivalent groupoids. The listed numbers are the same for clone equivalent groupoids (see Theorem 3.28)[1]. For the groupoid $\mathbf{G} := \langle\{0,1,2\},\star\rangle$ the operation $\star$ is coded in *flat form* as

$$(\star(0,0),\star(0,1),\star(0,2),\star(1,0),\star(1,1),\star(1,2),\star(2,0),\star(2,1),\star(2,2)),$$

i.e., the operation table "line-by-line". The groupoids are sorted and numbered by the alpha-numerical order of the flat form representation of their operation tables. For each clone equivalence class the first groupoid corresponding to this order is listed. The columns of the table are labelled as follows:

- **CE**: The number of the clone equivalence class $\operatorname{Clo}\mathbf{G}$.
- **G**: The number of the first groupoid **G** in this clone equivalence class.
- **Operation**: The operation table of $\star^{\mathbf{G}}$ in flat form.
- **n**: The number of generators needed to generate $\mathbb{Q}(\mathbf{F}_{\mathbf{G}}(\omega))$.

[1] A list of all non-isomorphic groupoids with the corresponding numbers of the clone equivalence classes can be downloaded from the webpage of S.N.Burris, `www.math.uwaterloo.ca/~snburris/htdocs/MYWORKS/PAPERS/Groupoid_Tables.pdf`

- **F(n)**: The cardinality of the free algebra $\mathbf{F_G}(n)$.
- **SS**: The cardinality of the smallest subalgebra of $\mathbf{F_G}(n)$.
- **MGS**: The cardinalities of the minimal generating set for $\mathbb{Q}(\mathbf{F_G}(n))$.
- **SC**: **G** is structurally complete.
- **ASC**: **G** is almost structurally complete.

Table A.1: Three element groupoids.

CE	**G**	**Operation**	**n**	**F(n)**	**SS**	**MGS**	**SC**	**ASC**
1	1	(0,0,0,0,0,0,0,0,0)	2	3	1	2	yes	yes
2	2	(0,0,0,0,0,0,0,0,1)	1	3	1	3	yes	yes
3	3	(0,0,0,0,0,0,0,0,2)	2	5	1	2,2	yes	yes
4	4	(0,0,0,0,0,0,0,1,0)	2	5	1	3	yes	yes
5	5	(0,0,0,0,0,0,0,1,1)	1	3	1	3	yes	yes
6	6	(0,0,0,0,0,0,0,1,2)	2	7	1	3	yes	yes
7	8	(0,0,0,0,0,0,0,2,1)	1	4	1	3	yes	yes
8	9	(0,0,0,0,0,0,0,2,2)	2	7	1	4	no	no
9	10	(0,0,0,0,0,0,1,0,0)	1	3	1	3	yes	yes
10	11	(0,0,0,0,0,0,1,0,1)	1	3	1	3	yes	yes
11	12	(0,0,0,0,0,0,1,0,2)	2	11	1	3	yes	yes
12	13	(0,0,0,0,0,0,1,1,0)	1	3	1	3	yes	yes
13	14	(0,0,0,0,0,0,1,1,1)	1	3	1	3	yes	yes
14	15	(0,0,0,0,0,0,1,1,2)	2	7	1	3	yes	yes
15	16	(0,0,0,0,0,0,1,2,0)	1	4	1	3	yes	yes
16	18	(0,0,0,0,0,0,1,2,2)	2	13	1	3	yes	yes
17	19	(0,0,0,0,0,0,2,0,0)	2	16	1	4	no	no

Table A.1; continued on next page

continued from previous page

CE	G	Operation	n	F(n)	SS	MGS	SC	ASC
18	21	(0,0,0,0,0,0,2,0,2)	2	6	1	4	no	no
19	22	(0,0,0,0,0,0,2,1,0)	2	24	1	3	yes	yes
20	24	(0,0,0,0,0,0,2,1,2)	2	8	1	3	yes	yes
21	25	(0,0,0,0,0,0,2,2,0)	2	8	1	3	no	no
22	26	(0,0,0,0,0,0,2,2,1)	1	4	1	3	yes	yes
23	27	(0,0,0,0,0,0,2,2,2)	2	4	1	2,2	yes	yes
24	30	(0,0,0,0,0,1,0,1,0)	2	6	1	3	yes	yes
25	31	(0,0,0,0,0,1,0,1,1)	1	3	1	3	yes	yes
26	32	(0,0,0,0,0,1,0,1,2)	2	8	1	3	yes	yes
27	33	(0,0,0,0,0,1,0,2,0)	2	5	1	3	yes	yes
28	34	(0,0,0,0,0,1,0,2,1)	1	4	1	3	yes	yes
29	35	(0,0,0,0,0,1,0,2,2)	2	11	1	3	yes	yes
30	36	(0,0,0,0,0,1,1,0,0)	1	3	1	3	yes	yes
31	37	(0,0,0,0,0,1,1,0,1)	1	3	1	3	yes	yes
32	38	(0,0,0,0,0,1,1,0,2)	2	15	1	3	yes	yes
33	39	(0,0,0,0,0,1,1,1,0)	1	3	1	3	yes	yes
34	40	(0,0,0,0,0,1,1,1,1)	1	3	1	3	yes	yes
35	41	(0,0,0,0,0,1,1,1,2)	2	11	1	3	yes	yes
36	42	(0,0,0,0,0,1,1,2,0)	1	4	1	3	yes	yes
37	43	(0,0,0,0,0,1,1,2,1)	1	4	1	3	yes	yes
38	44	(0,0,0,0,0,1,1,2,2)	2	20	1	3	yes	yes
39	45	(0,0,0,0,0,1,2,0,0)	2	24	1	4	no	no
40	46	(0,0,0,0,0,1,2,0,1)	1	4	1	3	yes	yes
41	47	(0,0,0,0,0,1,2,0,2)	2	10	1	4	no	no
42	48	(0,0,0,0,0,1,2,1,0)	2	36	1	3	yes	yes

Table A.1; continued on next page

continued from previous page

CE	G	Operation	n	F(n)	SS	MGS	SC	ASC
43	49	(0,0,0,0,0,1,2,1,1)	1	4	1	3	yes	yes
44	50	(0,0,0,0,0,1,2,1,2)	2	14	1	3	yes	yes
45	51	(0,0,0,0,0,1,2,2,0)	2	12	1	4	no	no
46	52	(0,0,0,0,0,1,2,2,1)	1	4	1	3	yes	yes
47	53	(0,0,0,0,0,1,2,2,2)	2	6	1	3	yes	yes
48	59	(0,0,0,0,0,2,0,2,1)	1	4	1	3	yes	yes
49	60	(0,0,0,0,0,2,0,2,2)	2	8	1	4	no	no
50	61	(0,0,0,0,0,2,1,0,0)	1	4	1	3	yes	yes
51	63	(0,0,0,0,0,2,1,0,2)	2	29	1	3	yes	yes
52	65	(0,0,0,0,0,2,1,1,1)	1	4	1	3	yes	yes
53	66	(0,0,0,0,0,2,1,1,2)	2	19	1	3	yes	yes
54	67	(0,0,0,0,0,2,1,2,0)	1	4	1	3	yes	yes
55	69	(0,0,0,0,0,2,1,2,2)	2	29	1	3	yes	yes
56	70	(0,0,0,0,0,2,2,0,0)	2	26	1	5	no	no
57	72	(0,0,0,0,0,2,2,0,2)	2	10	1	3	yes	yes
58	73	(0,0,0,0,0,2,2,1,0)	2	50	1	3	yes	yes
59	75	(0,0,0,0,0,2,2,1,2)	2	18	1	3	yes	yes
60	78	(0,0,0,0,0,2,2,2,2)	2	6	1	3	yes	yes
61	79	(0,0,0,0,1,0,0,0,1)	1	3	1	3	yes	yes
62	80	(0,0,0,0,1,0,0,0,2)	2	3	1	2	yes	yes
63	81	(0,0,0,0,1,0,0,1,1)	1	3	1	3	yes	yes
64	82	(0,0,0,0,1,0,0,1,2)	2	5	1	3	yes	yes
65	83	(0,0,0,0,1,0,0,2,1)	1	3	1	3	yes	yes
66	85	(0,0,0,0,1,0,1,0,0)	1	3	1	3	yes	yes
67	87	(0,0,0,0,1,0,1,0,2)	2	14	1	3	yes	yes

Table A.1; continued on next page

continued from previous page

CE	G	Operation	n	F(n)	SS	MGS	SC	ASC
68	88	(0,0,0,0,1,0,1,1,0)	1	3	1	3	yes	yes
69	89	(0,0,0,0,1,0,1,1,1)	1	3	1	3	yes	yes
70	90	(0,0,0,0,1,0,1,1,2)	2	10	1	3	yes	yes
71	91	(0,0,0,0,1,0,1,2,0)	1	3	1	3	yes	yes
72	93	(0,0,0,0,1,0,1,2,2)	2	34	1	3	yes	yes
73	94	(0,0,0,0,1,0,2,0,0)	2	18	1	3	yes	yes
74	96	(0,0,0,0,1,0,2,0,2)	2	6	1	4	no	no
75	97	(0,0,0,0,1,0,2,1,0)	2	30	1	3	yes	yes
76	99	(0,0,0,0,1,0,2,1,2)	2	10	1	3	yes	yes
77	100	(0,0,0,0,1,0,2,2,0)	2	10	1	2,2	yes	yes
78	101	(0,0,0,0,1,0,2,2,1)	1	3	1	3	yes	yes
79	102	(0,0,0,0,1,0,2,2,2)	2	4	1	2,2	yes	yes
80	104	(0,0,0,0,1,1,0,1,1)	2	5	1	2,2	yes	yes
81	105	(0,0,0,0,1,1,0,1,2)	3	7	1	2	yes	yes
82	106	(0,0,0,0,1,1,0,2,1)	2	10	1	2,2	no	no
83	107	(0,0,0,0,1,1,0,2,2)	3	12	1	2,2	no	no
84	111	(0,0,0,0,1,1,1,1,0)	1	3	1	3	yes	yes
85	112	(0,0,0,0,1,1,1,1,1)	2	7	1	4	no	no
86	113	(0,0,0,0,1,1,1,1,2)	2	5	1	3	yes	yes
87	115	(0,0,0,0,1,1,1,2,1)	2	26	1	3	yes	yes
88	116	(0,0,0,0,1,1,1,2,2)	2	8	1	3	yes	yes
89	117	(0,0,0,0,1,1,2,0,0)	2	34	1	3	yes	yes
90	119	(0,0,0,0,1,1,2,0,2)	2	10	1	4	no	no
91	120	(0,0,0,0,1,1,2,1,0)	2	44	1	3	yes	yes
92	121	(0,0,0,0,1,1,2,1,1)	2	9	1	3	yes	yes

Table A.1; continued on next page

continued from previous page

CE	G	Operation	n	F(n)	SS	MGS	SC	ASC
93	122	(0,0,0,0,1,1,2,1,2)	3	54	1	3	yes	yes
94	123	(0,0,0,0,1,1,2,2,0)	2	18	1	3	yes	yes
95	124	(0,0,0,0,1,1,2,2,1)	2	14	1	3	yes	yes
96	125	(0,0,0,0,1,1,2,2,2)	3	18	1	3	yes	yes
97	129	(0,0,0,0,1,2,0,2,1)	2	8	1	2,2	no	no
98	130	(0,0,0,0,1,2,1,0,0)	1	3	1	3	yes	yes
99	132	(0,0,0,0,1,2,1,0,2)	2	38	1	3	yes	yes
100	134	(0,0,0,0,1,2,1,1,1)	2	16	1	4	no	no
101	135	(0,0,0,0,1,2,1,1,2)	2	10	1	3	yes	yes
102	136	(0,0,0,0,1,2,1,2,0)	1	3	1	3	yes	yes
103	137	(0,0,0,0,1,2,1,2,1)	2	24	1	3	yes	yes
104	138	(0,0,0,0,1,2,1,2,2)	2	7	1	3	yes	yes
105	139	(0,0,0,0,1,2,2,0,0)	2	28	1	3	yes	yes
106	141	(0,0,0,0,1,2,2,0,2)	2	10	1	3	yes	yes
107	142	(0,0,0,0,1,2,2,1,0)	2	52	1	3	yes	yes
108	143	(0,0,0,0,1,2,2,1,1)	2	34	1	3	yes	yes
109	144	(0,0,0,0,1,2,2,1,2)	3	183	1	3	yes	yes
110	147	(0,0,0,0,1,2,2,2,2)	3	15	1	3	yes	yes
111	148	(0,0,0,0,2,0,0,0,1)	1	3	1	3	yes	yes
112	149	(0,0,0,0,2,0,0,1,1)	1	7	1	3	yes	yes
113	151	(0,0,0,0,2,0,1,0,0)	1	7	1	3	yes	yes
114	153	(0,0,0,0,2,0,1,0,2)	1	3	1	3	yes	yes
115	155	(0,0,0,0,2,0,1,1,1)	1	7	1	3	yes	yes
116	157	(0,0,0,0,2,0,1,2,0)	1	7	1	3	yes	yes
117	160	(0,0,0,0,2,0,2,0,0)	1	4	1	4	no	no

Table A.1; continued on next page

continued from previous page

CE	G	Operation	n	F(n)	SS	MGS	SC	ASC
118	161	(0,0,0,0,2,0,2,0,1)	1	9	1	3	yes	yes
119	162	(0,0,0,0,2,0,2,0,2)	1	3	1	3	yes	yes
120	163	(0,0,0,0,2,0,2,1,0)	1	6	1	3	yes	yes
121	165	(0,0,0,0,2,0,2,1,2)	1	3	1	3	yes	yes
122	166	(0,0,0,0,2,0,2,2,0)	1	4	1	4	no	no
123	168	(0,0,0,0,2,0,2,2,2)	1	3	1	3	yes	yes
124	169	(0,0,0,0,2,1,0,1,1)	2	24	2	4	no	no
125	170	(0,0,0,0,2,1,0,2,1)	2	8	2	4	no	no
126	171	(0,0,0,0,2,1,1,0,0)	1	9	1	3	yes	yes
127	175	(0,0,0,0,2,1,1,1,1)	2	56	2	4	no	no
128	176	(0,0,0,0,2,1,1,1,2)	2	68	1	3	yes	yes
129	178	(0,0,0,0,2,1,1,2,1)	2	68	2	6	no	yes
130	179	(0,0,0,0,2,1,1,2,2)	2	70	1	3	yes	yes
131	180	(0,0,0,0,2,1,2,0,0)	1	4	1	4	no	no
132	182	(0,0,0,0,2,1,2,0,2)	1	3	1	3	yes	yes
133	183	(0,0,0,0,2,1,2,1,0)	1	6	1	3	yes	yes
134	184	(0,0,0,0,2,1,2,1,1)	2	272	2	6	no	yes
135	185	(0,0,0,0,2,1,2,1,2)	2	24	1	3	yes	yes
136	186	(0,0,0,0,2,1,2,2,0)	1	4	1	4	no	no
137	188	(0,0,0,0,2,1,2,2,2)	2	12	1	4	no	no
138	194	(0,0,0,0,2,2,1,1,1)	2	16	2	4	no	no
139	195	(0,0,0,0,2,2,1,1,2)	2	102	1	3	yes	yes
140	198	(0,0,0,0,2,2,1,2,2)	2	13	1	3	yes	yes
141	199	(0,0,0,0,2,2,2,0,0)	1	5	1	5	no	no
142	201	(0,0,0,0,2,2,2,0,2)	1	3	1	3	yes	yes

Table A.1; continued on next page

continued from previous page

CE	**G**	**Operation**	**n**	**F(n)**	**SS**	**MGS**	**SC**	**ASC**
143	203	(0,0,0,0,2,2,2,1,1)	2	36	2	6	no	yes
144	204	(0,0,0,0,2,2,2,1,2)	2	32	1	3	yes	yes
145	207	(0,0,0,1,0,0,1,0,0)	1	3	1	3	yes	yes
146	209	(0,0,0,1,0,0,1,0,2)	2	60	1	3	yes	yes
147	213	(0,0,0,1,0,0,1,2,0)	1	6	1	3	yes	yes
148	215	(0,0,0,1,0,0,1,2,2)	2	136	1	3	yes	yes
149	216	(0,0,0,1,0,0,2,0,0)	2	7	1	3	yes	yes
150	218	(0,0,0,1,0,0,2,0,2)	2	24	1	4	no	no
151	219	(0,0,0,1,0,0,2,1,0)	2	40	1	3	yes	yes
152	221	(0,0,0,1,0,0,2,1,2)	2	48	1	3	yes	yes
153	222	(0,0,0,1,0,0,2,2,0)	2	14	1	3	yes	yes
154	223	(0,0,0,1,0,0,2,2,1)	1	3	1	3	yes	yes
155	224	(0,0,0,1,0,0,2,2,2)	2	16	1	3	yes	yes
156	235	(0,0,0,1,0,1,2,0,2)	2	16	1	4	no	no
157	239	(0,0,0,1,0,1,2,2,0)	2	6	1	2	yes	yes
158	241	(0,0,0,1,0,1,2,2,2)	2	8	1	2,2	yes	yes
159	244	(0,0,0,1,0,2,1,0,2)	2	160	1	3	yes	yes
160	250	(0,0,0,1,0,2,1,2,2)	2	198	1	3	yes	yes
161	252	(0,0,0,1,0,2,2,0,2)	2	18	1	4	no	no
162	253	(0,0,0,1,0,2,2,1,0)	2	18	1	3	yes	yes
163	255	(0,0,0,1,0,2,2,1,2)	2	72	1	3	yes	yes
164	257	(0,0,0,1,1,0,1,0,0)	1	3	1	3	yes	yes
165	258	(0,0,0,1,1,0,1,0,1)	1	3	1	3	yes	yes
166	259	(0,0,0,1,1,0,1,0,2)	2	18	1	3	yes	yes
167	260	(0,0,0,1,1,0,1,1,0)	1	3	1	3	yes	yes

Table A.1; continued on next page

continued from previous page

CE	G	Operation	n	F(n)	SS	MGS	SC	ASC
168	261	(0,0,0,1,1,0,1,1,1)	1	3	1	3	yes	yes
169	262	(0,0,0,1,1,0,1,1,2)	2	10	1	3	yes	yes
170	263	(0,0,0,1,1,0,1,2,0)	1	3	1	3	yes	yes
171	265	(0,0,0,1,1,0,1,2,2)	2	30	1	3	yes	yes
172	266	(0,0,0,1,1,0,2,0,1)	1	3	1	3	yes	yes
173	267	(0,0,0,1,1,0,2,0,2)	2	4	1	4	no	no
174	268	(0,0,0,1,1,0,2,1,1)	1	3	1	3	yes	yes
175	269	(0,0,0,1,1,0,2,1,2)	2	10	1	3	yes	yes
176	270	(0,0,0,1,1,0,2,2,1)	1	3	1	3	yes	yes
177	271	(0,0,0,1,1,0,2,2,2)	2	4	1	3	yes	yes
178	272	(0,0,0,1,1,1,1,0,0)	1	3	1	3	yes	yes
179	273	(0,0,0,1,1,1,1,0,2)	2	6	1	3	yes	yes
180	274	(0,0,0,1,1,1,1,2,0)	1	3	1	3	yes	yes
181	275	(0,0,0,1,1,1,2,2,2)	3	3	1	2	yes	yes
182	278	(0,0,0,1,1,2,1,0,2)	2	44	1	3	yes	yes
183	280	(0,0,0,1,1,2,1,1,1)	2	20	1	3	yes	yes
184	281	(0,0,0,1,1,2,1,1,2)	2	6	1	3	yes	yes
185	282	(0,0,0,1,1,2,1,2,0)	1	3	1	3	yes	yes
186	283	(0,0,0,1,1,2,1,2,1)	2	16	1	3	yes	yes
187	284	(0,0,0,1,1,2,1,2,2)	2	6	1	3	yes	yes
188	286	(0,0,0,1,1,2,2,1,1)	2	24	1	3	yes	yes
189	287	(0,0,0,1,1,2,2,1,2)	3	36	1	3	yes	yes
190	298	(0,0,0,1,2,0,2,0,1)	1	3	1	3	yes	yes
191	305	(0,0,0,1,2,1,1,1,1)	2	128	2	8	no	no
192	306	(0,0,0,1,2,1,1,1,2)	2	32	1	4	no	no

Table A.1; continued on next page

continued from previous page

CE	G	Operation	n	F(n)	SS	MGS	SC	ASC
193	308	(0,0,0,1,2,1,1,2,1)	2	20	2	8	no	no
194	309	(0,0,0,1,2,1,1,2,2)	2	32	1	4	no	no
195	311	(0,0,0,1,2,1,2,2,1)	2	16	2	8	no	no
196	316	(0,0,0,1,2,2,1,1,1)	2	12	2	8	no	no
197	317	(0,0,0,1,2,2,1,1,2)	2	48	1	4	no	no
198	320	(0,0,0,1,2,2,1,2,2)	2	6	1	4	no	no
199	321	(0,0,0,1,2,2,2,1,1)	2	8	2	8	no	no
200	322	(0,0,0,2,0,0,1,0,0)	1	3	1	3	yes	yes
201	341	(0,0,0,2,0,2,1,1,0)	1	3	1	3	yes	yes
202	347	(0,0,0,2,1,0,1,0,2)	2	15	1	3	yes	yes
203	349	(0,0,0,2,1,0,1,1,2)	2	153	1	3	yes	yes
204	353	(0,0,0,2,1,1,1,1,1)	2	10	1	4	no	no
205	354	(0,0,0,2,1,1,1,1,2)	2	10	1	4	no	no
206	356	(0,0,0,2,1,1,1,2,2)	2	4	1	4	no	no
207	359	(0,0,0,2,1,2,1,1,2)	2	8	1	4	no	no
208	366	(0,0,0,2,2,2,1,1,1)	2	4	2	2	no	yes
209	376	(0,0,1,0,0,0,1,0,0)	1	3	1	3	yes	yes
210	377	(0,0,1,0,0,0,1,0,1)	1	3	1	3	yes	yes
211	378	(0,0,1,0,0,0,1,0,2)	2	15	1	3	yes	yes
212	379	(0,0,1,0,0,0,1,1,0)	1	3	1	3	yes	yes
213	380	(0,0,1,0,0,0,1,1,1)	1	3	1	3	yes	yes
214	381	(0,0,1,0,0,0,1,1,2)	2	14	1	3	yes	yes
215	382	(0,0,1,0,0,0,1,2,0)	1	4	1	3	yes	yes
216	384	(0,0,1,0,0,0,1,2,2)	2	46	1	3	yes	yes
217	385	(0,0,1,0,0,0,2,0,0)	1	4	1	3	yes	yes

Table A.1; continued on next page

continued from previous page

CE	G	Operation	n	F(n)	SS	MGS	SC	ASC
218	387	(0,0,1,0,0,0,2,0,2)	2	37	1	3	yes	yes
219	388	(0,0,1,0,0,0,2,1,0)	1	4	1	3	yes	yes
220	390	(0,0,1,0,0,0,2,1,2)	2	58	1	3	yes	yes
221	391	(0,0,1,0,0,0,2,2,0)	1	4	1	3	yes	yes
222	405	(0,0,1,0,0,1,1,1,0)	1	3	1	3	yes	yes
223	406	(0,0,1,0,0,1,1,1,1)	1	3	1	3	yes	yes
224	407	(0,0,1,0,0,1,1,1,2)	2	8	1	3	yes	yes
225	410	(0,0,1,0,0,1,1,2,2)	2	27	1	3	yes	yes
226	417	(0,0,1,0,0,1,2,2,0)	1	4	1	3	yes	yes
227	434	(0,0,1,0,0,2,1,2,0)	1	4	1	3	yes	yes
228	436	(0,0,1,0,0,2,1,2,2)	2	33	1	3	yes	yes
229	437	(0,0,1,0,0,2,2,0,0)	1	4	1	3	yes	yes
230	439	(0,0,1,0,0,2,2,0,2)	2	83	1	3	yes	yes
231	454	(0,0,1,0,1,0,1,0,0)	1	3	1	3	yes	yes
232	455	(0,0,1,0,1,0,1,0,1)	1	3	1	3	yes	yes
233	456	(0,0,1,0,1,0,1,0,2)	2	15	1	3	yes	yes
234	457	(0,0,1,0,1,0,1,1,0)	1	3	1	3	yes	yes
235	458	(0,0,1,0,1,0,1,1,1)	1	3	1	3	yes	yes
236	459	(0,0,1,0,1,0,1,1,2)	2	17	1	3	yes	yes
237	460	(0,0,1,0,1,0,1,2,0)	1	3	1	3	yes	yes
238	462	(0,0,1,0,1,0,1,2,2)	2	46	1	3	yes	yes
239	463	(0,0,1,0,1,0,2,0,0)	1	3	1	3	yes	yes
240	465	(0,0,1,0,1,0,2,0,2)	2	58	1	3	yes	yes
241	469	(0,0,1,0,1,0,2,2,0)	1	3	1	3	yes	yes
242	483	(0,0,1,0,1,1,1,1,0)	1	3	1	3	yes	yes

Table A.1; continued on next page

continued from previous page

CE	G	Operation	n	F(n)	SS	MGS	SC	ASC
243	484	(0,0,1,0,1,1,1,1,1)	2	8	1	4	no	no
244	485	(0,0,1,0,1,1,1,1,2)	2	6	1	3	yes	yes
245	487	(0,0,1,0,1,1,1,2,1)	2	36	1	3	yes	yes
246	488	(0,0,1,0,1,1,1,2,2)	2	12	1	3	yes	yes
247	493	(0,0,1,0,1,1,2,1,1)	2	12	1	3	yes	yes
248	494	(0,0,1,0,1,1,2,1,2)	2	10	1	3	yes	yes
249	495	(0,0,1,0,1,1,2,2,0)	1	3	1	3	yes	yes
250	496	(0,0,1,0,1,1,2,2,1)	2	20	1	3	yes	yes
251	512	(0,0,1,0,1,2,1,2,0)	1	3	1	3	yes	yes
252	513	(0,0,1,0,1,2,1,2,1)	2	28	1	3	yes	yes
253	514	(0,0,1,0,1,2,1,2,2)	2	5	1	3	yes	yes
254	515	(0,0,1,0,1,2,2,0,0)	1	3	1	3	yes	yes
255	517	(0,0,1,0,1,2,2,0,2)	2	83	1	3	yes	yes
256	519	(0,0,1,0,1,2,2,1,1)	2	58	1	3	yes	yes
257	520	(0,0,1,0,1,2,2,1,2)	2	20	1	3	yes	yes
258	522	(0,0,1,0,1,2,2,2,1)	2	40	1	3	yes	yes
259	532	(0,0,1,0,2,0,1,0,0)	1	9	1	3	yes	yes
260	534	(0,0,1,0,2,0,1,0,2)	1	3	1	3	yes	yes
261	538	(0,0,1,0,2,0,1,2,0)	1	9	1	3	yes	yes
262	562	(0,0,1,0,2,1,1,1,1)	2	82	2	4	no	no
263	563	(0,0,1,0,2,1,1,1,2)	2	324	1	3	yes	yes
264	565	(0,0,1,0,2,1,1,2,1)	2	324	2	6	no	yes
265	566	(0,0,1,0,2,1,1,2,2)	2	486	1	3	yes	yes
266	571	(0,0,1,0,2,1,2,1,1)	2	1296	2	6	no	yes
267	600	(0,0,1,1,0,0,0,0,0)	1	4	1	4	no	no

Table A.1; continued on next page

continued from previous page

CE	G	Operation	n	F(n)	SS	MGS	SC	ASC
268	602	(0,0,1,1,0,0,0,0,2)	2	100	1	3	yes	yes
269	603	(0,0,1,1,0,0,0,1,0)	1	5	1	5	no	no
270	604	(0,0,1,1,0,0,0,1,1)	1	5	1	5	no	no
271	606	(0,0,1,1,0,0,0,2,0)	1	4	1	4	no	no
272	608	(0,0,1,1,0,0,0,2,2)	2	208	1	3	yes	yes
273	609	(0,0,1,1,0,0,1,0,0)	1	5	1	5	no	no
274	612	(0,0,1,1,0,0,1,1,0)	1	5	1	5	no	no
275	613	(0,0,1,1,0,0,1,1,1)	1	5	1	5	no	no
276	615	(0,0,1,1,0,0,1,2,0)	1	6	1	3	yes	yes
277	618	(0,0,1,1,0,0,2,0,0)	1	6	1	3	yes	yes
278	620	(0,0,1,1,0,0,2,0,2)	2	256	1	3	yes	yes
279	624	(0,0,1,1,0,0,2,2,0)	1	3	1	3	yes	yes
280	629	(0,0,1,1,0,1,0,1,0)	1	5	1	5	no	no
281	630	(0,0,1,1,0,1,0,1,1)	1	5	1	5	no	no
282	632	(0,0,1,1,0,1,0,2,0)	1	4	1	4	no	no
283	638	(0,0,1,1,0,1,1,1,0)	1	5	1	5	no	no
284	639	(0,0,1,1,0,1,1,1,1)	1	5	1	5	no	no
285	652	(0,0,1,1,0,2,0,0,0)	1	6	1	3	yes	yes
286	654	(0,0,1,1,0,2,0,0,2)	2	336	1	3	yes	yes
287	658	(0,0,1,1,0,2,0,2,0)	1	6	1	3	yes	yes
288	677	(0,0,1,1,1,0,0,0,0)	1	3	1	3	yes	yes
289	678	(0,0,1,1,1,0,0,0,1)	1	3	1	3	yes	yes
290	679	(0,0,1,1,1,0,0,0,2)	2	10	1	3	yes	yes
291	680	(0,0,1,1,1,0,0,1,0)	1	3	1	3	yes	yes
292	681	(0,0,1,1,1,0,0,1,2)	2	10	1	3	yes	yes

Table A.1; continued on next page

continued from previous page

CE	**G**	**Operation**	**n**	**F(n)**	**SS**	**MGS**	**SC**	**ASC**
293	682	(0,0,1,1,1,0,0,2,0)	1	3	1	3	yes	yes
294	684	(0,0,1,1,1,0,0,2,2)	2	60	1	3	yes	yes
295	687	(0,0,1,1,1,0,1,2,0)	1	3	1	3	yes	yes
296	690	(0,0,1,1,1,0,2,2,0)	1	3	1	3	yes	yes
297	691	(0,0,1,1,1,2,0,0,0)	1	3	1	3	yes	yes
298	693	(0,0,1,1,1,2,0,0,2)	2	112	1	3	yes	yes
299	695	(0,0,1,1,1,2,0,1,1)	2	80	1	3	yes	yes
300	696	(0,0,1,1,1,2,0,1,2)	2	24	1	3	yes	yes
301	697	(0,0,1,1,1,2,0,2,0)	1	3	1	3	yes	yes
302	698	(0,0,1,1,1,2,0,2,1)	2	72	1	3	yes	yes
303	704	(0,0,1,1,1,2,1,1,1)	2	48	1	3	yes	yes
304	705	(0,0,1,1,1,2,1,1,2)	2	14	1	3	yes	yes
305	707	(0,0,1,1,1,2,1,2,1)	2	48	1	3	yes	yes
306	710	(0,0,1,1,1,2,2,0,2)	2	162	1	3	yes	yes
307	712	(0,0,1,1,1,2,2,1,1)	2	72	1	3	yes	yes
308	755	(0,0,1,1,2,1,1,1,1)	2	896	2	6	no	yes
309	756	(0,0,1,1,2,1,1,1,2)	2	224	1	3	yes	yes
310	758	(0,0,1,1,2,1,1,2,1)	2	224	2	6	no	yes
311	780	(0,0,1,1,2,2,1,1,1)	2	144	2	6	no	yes
312	792	(0,0,1,2,0,0,0,0,0)	1	7	1	3	yes	yes
313	870	(0,0,1,2,1,0,0,0,2)	2	729	1	3	yes	yes
314	885	(0,0,1,2,1,0,1,2,2)	2	27	1	3	yes	yes
315	898	(0,0,1,2,1,1,2,0,2)	2	9	1	3	yes	yes
316	984	(0,0,1,2,2,2,1,1,1)	2	36	2	6	no	yes
317	1012	(0,0,2,0,0,0,2,0,0)	2	18	1	5	no	no

Table A.1; continued on next page

continued from previous page

CE	G	Operation	n	F(n)	SS	MGS	SC	ASC
318	1014	(0,0,2,0,0,0,2,1,0)	2	34	1	3	yes	yes
319	1038	(0,0,2,0,0,1,2,1,0)	2	38	1	3	yes	yes
320	1040	(0,0,2,0,0,1,2,1,2)	2	13	1	3	yes	yes
321	1065	(0,0,2,0,0,2,2,2,0)	2	6	1	3	no	no
322	1066	(0,0,2,0,0,2,2,2,1)	1	4	1	3	yes	yes
323	1084	(0,0,2,0,1,0,2,0,0)	2	20	1	3	yes	yes
324	1086	(0,0,2,0,1,0,2,1,0)	2	36	1	3	yes	yes
325	1107	(0,0,2,0,1,1,2,1,0)	2	40	1	3	yes	yes
326	1108	(0,0,2,0,1,1,2,1,2)	3	15	1	3	yes	yes
327	1132	(0,0,2,0,1,2,2,2,0)	2	8	1	2,2	yes	yes
328	1133	(0,0,2,0,1,2,2,2,1)	2	13	1	3	yes	yes
329	1151	(0,0,2,0,2,0,2,0,0)	1	5	1	5	no	no
330	1153	(0,0,2,0,2,0,2,1,0)	1	6	1	3	yes	yes
331	1176	(0,0,2,0,2,1,2,1,0)	1	6	1	3	yes	yes
332	1200	(0,0,2,0,2,2,2,2,0)	1	5	1	5	no	no
333	1202	(0,0,2,1,0,0,0,0,0)	2	164	1	3	yes	yes
334	1205	(0,0,2,1,0,0,0,1,0)	2	240	1	3	yes	yes
335	1219	(0,0,2,1,0,0,2,0,0)	2	160	1	3	yes	yes
336	1221	(0,0,2,1,0,0,2,1,0)	2	216	1	3	yes	yes
337	1225	(0,0,2,1,0,1,0,0,0)	2	68	1	3	yes	yes
338	1227	(0,0,2,1,0,1,0,0,2)	2	20	1	3	yes	yes
339	1231	(0,0,2,1,0,1,0,2,0)	2	96	1	3	yes	yes
340	1233	(0,0,2,1,0,1,0,2,2)	2	32	1	3	yes	yes
341	1242	(0,0,2,1,0,1,2,0,0)	2	64	1	3	yes	yes
342	1249	(0,0,2,1,0,2,0,0,1)	1	6	1	3	yes	yes

Table A.1; continued on next page

continued from previous page

CE	G	Operation	n	F(n)	SS	MGS	SC	ASC
343	1268	(0,0,2,1,0,2,2,2,0)	2	40	1	3	yes	yes
344	1269	(0,0,2,1,0,2,2,2,1)	1	6	1	3	yes	yes
345	1271	(0,0,2,1,1,2,0,0,1)	1	3	1	3	yes	yes
346	1277	(0,0,2,1,1,2,1,0,0)	1	3	1	3	yes	yes
347	1281	(0,0,2,1,1,2,2,2,0)	2	12	1	3	yes	yes
348	1321	(0,0,2,1,2,1,2,0,0)	1	6	1	3	yes	yes
349	1433	(0,0,2,2,1,0,2,0,0)	2	68	1	3	yes	yes
350	1437	(0,0,2,2,1,0,2,2,0)	2	20	1	3	yes	yes
351	1481	(0,0,2,2,2,0,2,2,0)	1	5	1	5	no	no
352	1700	(0,1,1,1,0,0,1,0,0)	1	3	1	3	yes	yes
353	1708	(0,1,1,1,0,0,2,0,0)	1	3	1	3	yes	yes
354	1791	(0,1,1,1,2,1,1,1,1)	2	264	2	6	no	yes
355	1793	(0,1,1,1,2,1,1,2,1)	2	28	2	6	no	yes
356	1799	(0,1,1,1,2,1,2,2,1)	2	52	2	6	no	yes
357	1818	(0,1,1,1,2,2,2,1,1)	2	20	2	6	no	yes
358	1829	(0,1,1,2,0,0,1,0,0)	1	5	1	3	yes	yes
359	1837	(0,1,1,2,0,0,2,0,0)	1	4	1	3	yes	yes
360	1962	(0,1,1,2,2,2,1,1,1)	2	12	2	6	no	yes
361	2088	(0,1,2,1,0,0,2,0,0)	2	6	1	3	yes	yes
362	2090	(0,1,2,1,0,0,2,1,0)	2	36	1	3	yes	yes
363	2102	(0,1,2,1,0,1,2,1,0)	2	9	1	3	yes	yes
364	2104	(0,1,2,1,0,1,2,2,0)	2	12	1	3	yes	yes
365	2116	(0,1,2,1,0,2,2,2,1)	1	3	1	3	yes	yes
366	2124	(0,1,2,1,2,0,2,0,1)	1	3	1	3	yes	yes
367	2135	(0,1,2,1,2,1,2,2,1)	2	20	2	6	no	yes

Table A.1; continued on next page

continued from previous page

CE	G	Operation	n	F(n)	SS	MGS	SC	ASC
368	2144	(0,1,2,2,0,0,1,0,0)	1	3	1	3	yes	yes
369	2159	(0,1,2,2,0,2,1,1,0)	1	3	1	3	yes	yes
370	2171	(0,1,2,2,2,2,1,1,1)	2	12	2	6	no	yes
371	2346	(0,2,1,2,1,0,1,0,2)	2	3	1	3	yes	yes
372	2353	(1,0,0,0,0,0,0,0,0)	1	5	2	5	no	no
373	2354	(1,0,0,0,0,0,0,0,1)	1	9	2	9	no	no
374	2357	(1,0,0,0,0,0,0,2,0)	1	12	2	6	no	yes
375	2369	(1,0,0,0,0,0,2,2,0)	1	12	2	6	no	yes
376	2393	(1,0,0,0,0,2,0,2,0)	1	12	2	6	no	yes
377	2407	(1,0,0,0,2,0,0,0,0)	1	27	3	3	yes	yes
378	2428	(1,0,0,0,2,1,0,1,1)	1	12	2	6	no	yes
379	2430	(1,0,0,0,2,1,0,2,1)	1	6	2	6	no	yes
380	2436	(1,0,0,0,2,1,1,2,1)	1	6	2	6	no	yes
381	2460	(1,0,0,1,0,0,0,0,0)	1	5	2	5	no	no
382	2461	(1,0,0,1,0,0,0,0,1)	1	5	2	5	no	no
383	2462	(1,0,0,1,0,0,0,1,0)	1	5	2	5	no	no
384	2463	(1,0,0,1,0,0,0,1,1)	1	5	2	5	no	no
385	2464	(1,0,0,1,0,0,0,2,0)	1	6	2	6	no	yes
386	2466	(1,0,0,1,0,0,1,0,0)	1	3	2	3	yes	yes
387	2467	(1,0,0,1,0,0,1,0,1)	1	5	2	5	no	no
388	2472	(1,0,0,1,0,0,2,0,0)	1	3	2	3	yes	yes
389	2476	(1,0,0,1,0,0,2,2,0)	1	6	2	6	no	yes
390	2478	(1,0,0,1,0,1,0,0,0)	1	5	2	5	no	no
391	2479	(1,0,0,1,0,1,0,0,1)	1	5	2	5	no	no
392	2480	(1,0,0,1,0,1,0,1,0)	1	5	2	5	no	no

Table A.1; continued on next page

continued from previous page

CE	**G**	**Operation**	**n**	**F(n)**	**SS**	**MGS**	**SC**	**ASC**
393	2483	(1,0,0,1,0,1,1,0,0)	1	5	2	5	no	no
394	2486	(1,0,0,1,0,1,2,2,0)	1	6	2	6	no	yes
395	2487	(1,0,0,1,0,2,0,0,0)	1	6	2	6	no	yes
396	2493	(1,0,0,1,0,2,1,0,0)	1	6	2	6	no	yes
397	2529	(1,0,0,1,2,1,2,2,0)	1	3	3	3	yes	yes
398	2539	(1,0,0,1,2,2,1,1,1)	1	6	2	6	no	yes
399	2545	(1,0,0,1,2,2,2,1,1)	1	6	2	6	no	yes
400	2552	(1,0,0,2,0,0,1,0,0)	1	6	4	4	no	no
401	2558	(1,0,0,2,0,0,2,0,0)	1	5	4	4	no	no
402	2636	(1,0,0,2,2,2,1,1,1)	1	6	2	6	no	yes
403	2654	(1,0,1,0,0,0,1,2,1)	1	6	2	3	yes	yes
404	2686	(1,0,1,0,0,2,1,2,1)	1	6	2	3	yes	yes
405	2698	(1,0,1,0,2,0,1,0,1)	1	10	3	3	yes	yes
406	2702	(1,0,1,0,2,0,1,2,1)	1	12	3	3	yes	yes
407	2739	(1,0,1,1,0,0,0,0,1)	1	5	2	5	no	no
408	2799	(1,0,1,2,0,0,1,0,1)	1	15	3	3	yes	yes
409	2803	(1,0,1,2,0,0,1,2,1)	1	15	3	3	yes	yes
410	2934	(1,0,2,0,2,1,2,1,0)	1	3	3	3	yes	yes
411	3242	(1,1,1,2,2,2,0,0,0)	1	3	3	3	yes	yes

List of Figures

List of Tables

List of Algorithms

Bibliography

[1] F. Baader and S. Ghilardi. Unification in modal and description logics. *Log. J. IGPL*, 19(6):705–730, 2011.

[2] M. Baaz, C.G. Fermüller, and G. Salzer. Automated deduction for many-valued logics. In *Handbook of Automated Reasoning*, volume II, chapter 20, pages 1355–1402. Elsevier, 2001.

[3] M. Baaz, C.G. Fermüller, and R. Zach. Dual systems of sequents and tableaux for many-valued logics. *Bulletin of the EATCS*, 51:192–197, 1993.

[4] M. Baaz and A. Leitsch. *Methods of cut-elimination*, volume 34 of *Trends in Logic—Studia Logica Library*. Springer, Dordrecht, 2011.

[5] S. Babenyshev and V. Rybakov. Unification in linear temporal logic LTL. *Ann. Pure Appl. Logic*, 162(12):991–1000, 2011.

[6] S. Babenyshev, V. Rybakov, R.A. Schmidt, and D. Tishkovsky. A tableau method for checking rule admissibility in S4. In *Proc. M4M 2009*, volume 262 of *ENTCS*, pages 17–32, 2010.

[7] B. Beckert, R. Hähnle, P. Oel, and M. Sulzmann. The tableau-based theorem prover $_3T^AP$, version 4.0. In *CADE '96*, volume 1104 of *LNCS*, pages 303–307. Springer, 1996.

[8] N.D. Belnap. A useful four-valued logic. In *Modern uses of multiple-valued logic (Fifth Internat. Sympos., Indiana Univ., Bloomington, Ind., 1975)*, pages 5–37. Episteme, Vol. 2. Reidel, Dordrecht, 1977.

[9] C. Bergman. Structural completeness in algebra and logic. In *Algebraic Logic*, volume 54 of *Colloq. Math. Soc. János Bolyai*, pages 59–73. North-Holland, Amsterdam, 1991.

[10] C. Bergman. *Universal Algebra: Fundamentals and Selected Topics.* Chapman and Hall Pure and Applied Mathematics Series. CRC Press, 2012.

[11] C. Bergman, D. Juedes, and G. Slutzki. Computational complexity of term-equivalence. *Internat. J. Algebra Comput.*, 9(1):113–128, 1999.

[12] C. Bergman and R. McKenzie. Minimal varieties and quasivarieties. *J. Austral. Math. Soc. Ser. A*, 48(1):133–147, 1990.

[13] C. Bergman and G. Slutzki. Complexity of some problems concerning varieties and quasi-varieties of algebras. *SIAM J. Comput.*, 30(2):359–382, 2000.

[14] J. Berman. Finite algebras with large free spectra. *Algebra Universalis*, 26(2):149–165, 1989.

[15] J. Berman. Upper bounds on the sizes of finitely generated algebras. *Demonstratio Math.*, 44(3):447–471, 2011.

[16] J. Berman and S. Burris. A computer study of 3-element groupoids. In *Logic and algebra (Pontignano, 1994)*, volume 180 of *Lecture Notes in Pure and Appl. Math.*, pages 379–429. Dekker, New York, 1996.

[17] G. Birkhoff. On the structure of abstract algebras. *Proc. Camb. Philos. Soc.*, 31:433–454, 1935.

[18] G. Birkhoff. *Lattice Theory.* Amer. Math. Soc., New York, 1940.

[19] G. Birkhoff. Subdirect unions in universal algebra. *Bull. Amer. Math. Soc.*, 50:764–768, 1944.

[20] W.J. Blok and W. Dziobiak. On the lattice of quasivarieties of Sugihara algebras. *Studia Logica*, 45(3):275–280, 1986.

[21] W.J. Blok and D. Pigozzi. *Algebraizable Logics*, volume 77 of *Mem. Amer. Math. Soc.* Amer. Math. Soc., 1989.

[22] W.J. Blok and J.G. Raftery. Fragments of R-mingle. *Studia Logica*, 78(1-2):59–106, 2004.

[23] D.A. Bochvar. On a three-valued logical calculus and its application to the analysis of the paradoxes of the classical extended functional calculus. *Hist. Philos. Logic*, 2:87–112, 1981. Translated from the Russian by Merrie Bergmann.

[24] D.P. Bovet and P. Crescenzi. *Introduction to the theory of complexity.* Prentice Hall International Series in Computer Science. Prentice Hall International, New York, 1994.

[25] S. Burris and H.P. Sankappanavar. *A Course in Universal Algebra*, volume 78 of *Graduate Texts in Mathematics*. Springer, New York, 1981.

[26] L.M. Cabrer and G. Metcalfe. Admissibility via natural dualities. Manuscript.

[27] X. Caicedo. The subdirect decomposition theorem for classes of structures closed under direct limits. *J. Austral. Math. Soc. Ser. A*, 30(2):171–179, 1980/81.

[28] W.A. Carnielli. Systematization of finite many-valued logics through the method of tableaux. *J. Symbolic Logic*, 52(2):473–493, 1987.

[29] W.A. Carnielli. On sequents and tableaux for many-valued logics. *J. Non-Classical Logic*, 8(1):59–76, 1991.

[30] P. Cintula and G. Metcalfe. Structural completeness in fuzzy logics. *Notre Dame J. Form. Log.*, 50(2):153–183, 2009.

[31] P. Cintula and G. Metcalfe. Admissible rules in the implication-negation fragment of intuitionistic logic. *Ann. Pure Appl. Logic*, 162(10):162–171, 2010.

[32] A.I. Citkin. On structurally complete superintuitionistic logics. *Soviet Mathematics Doklady*, 19:816–819, 1978.

[33] J. Demel. Fast algorithms for finding a subdirect decomposition and interesting congruences of finite algebras. *Kybernetika*, 18:121–130, 1982.

[34] N. Dershowitz and Z. Manna. Proving termination with multiset orderings. *Commun. ACM*, 22:465–476, 1979.

[35] J.M. Dunn. Algebraic completeness for R-mingle and its extensions. *J. Symbolic Logic*, 35:1–13, 1970.

[36] W. Dzik and A. Wronski. Structural completeness of Gödel's and Dummett's propositional calculi. *Studia Logica*, 32:69–73, 1973.

[37] W. Dziobiak. Finitely generated congruence distributive quasivarieties of algebras. *Fund. Math.*, 133(1):47–57, 1989.

[38] R. Freese, E. Kiss, and M. Valeriote. Universal Algebra Calculator, 2011. Available at: `www.uacalc.org`.

[39] H.M. Friedman. One hundred and two problems in mathematical logic. *J. Symbolic Logic*, 40(2):113–129, 1975.

[40] H. Gaitán and M.H. Perea. A non-finitely based quasi-variety of De Morgan algebras. *Studia Logica*, 78(1-2):237–248, 2004.

[41] G. Gentzen. Untersuchungen über das Logische Schliessen. *Math. Zeitschrift*, 39:176–210,405–431, 1935.

[42] S. Ghilardi. Unification through projectivity. *J. Logic Comput.*, 7(6):733–752, 1997.

[43] S. Ghilardi. Unification in intuitionistic logic. *J. Symbolic Logic*, 64(2):859–880, 1999.

[44] S. Ghilardi. Best solving modal equations. *Ann. Pure Appl. Logic*, 102(3):184–198, 2000.

[45] S. Ghilardi. A resolution/tableaux algorithm for projective approximations in IPC. *Log. J. IGPL*, 10(3):227–241, 2002.

[46] S. Ghilardi. Unification, finite duality and projectivity in varieties of Heyting algebras. *Annals of Pure and Applied Logic*, 127(1–3):99–115, 2004.

[47] A.J. Gil and G. Salzer. MUltseq. `http://www.logic.at/multseq`.

[48] K. Gödel. Zum intuitionistischen Aussagenkalkül. *Anzeiger Akademie der Wissenschaften Wien*, 69:65–66, 1932.

[49] V.A. Gorbunov. *Algebraic Theory of Quasivarieties.* Springer, 1998.

[50] V.A. Gorbunov and A.V. Kravchenko. Universal Horn classes and antivarieties of algebraic systems. *Algebra Log.*, 39(1):3–22, 120, 2000.

[51] G. Grätzer and H. Lakser. A note on the implicational class generated by a class of structures. *Canad. Math. Bull.*, 16:603–605, 1973.

[52] G. Grätzer and E.T. Schmidt. On a problem of M. H. Stone. *Acta Math. Acad. Sci. Hungar.*, 8:455–460, 1957.

[53] M. Grohe. The structure of tractable constraint satisfaction problems. In *Mathematical foundations of computer science 2006*, volume 4162 of *Lecture Notes in Comput. Sci.*, pages 58–72. Springer, Berlin, 2006.

[54] R. Hähnle. *Automated Deduction in Multiple-Valued Logics.* Oxford Univ. Press, 1993.

[55] D. Hobby and R.N. McKenzie. *The Structure of Finite Algebras*, volume 76 of *Contemporary Mathematics*. Amer. Math. Soc., 1988.

[56] R. Iemhoff. On the admissible rules of intuitionistic propositional logic. *J. Symbolic Logic*, 66(1):281–294, 2001.

[57] R. Iemhoff. Intermediate logics and Visser's rules. *Notre Dame J. Form. Log.*, 46(1):65–81, 2005.

[58] R. Iemhoff and G. Metcalfe. Hypersequent systems for the admissible rules of modal and intermediate logics. In *Proc. LFCS 2009*, volume 5407 of *LNCS*, pages 230–245. Springer, 2009.

[59] R. Iemhoff and G. Metcalfe. Proof theory for admissible rules. *Ann. Pure Appl. Logic*, 159(1–2):171–186, 2009.

[60] E. Jeřábek. Admissible rules of modal logics. *J. Logic Comput.*, 15:411–431, 2005.

[61] E. Jeřábek. Admissible rules of Łukasiewicz logic. *J. Logic Comput.*, 20(2):425–447, 2010.

[62] E. Jeřábek. Bases of admissible rules of Łukasiewicz logic. *J. Logic Comput.*, 20(6):1149–1163, 2010.

[63] I. Johansson. Der Minimalkalkül, ein reduzierter intuitionistischer Formalismus. *Compositio Math.*, 4:119–136, 1937.

[64] B. Jónsson. Sublattices of a free lattice. *Canad. J. Math.*, 13:256–264, 1961.

[65] J.A. Kalman. Lattices with involution. *Trans. Amer. Math. Soc.*, 87:485–491, 1958.

[66] S.C. Kleene. *Introduction to metamathematics*. D. Van Nostrand Co., Inc., New York, N. Y., 1952.

[67] H. Lakser. The structure of pseudocomplemented distributive lattices. I. Subdirect decomposition. *Trans. Amer. Math. Soc.*, 156:335–342, 1971.

[68] B. Larose and P. Tesson. Universal algebra and hardness results for constraint satisfaction problems. *Theoret. Comput. Sci.*, 410(18):1629–1647, 2009.

[69] P. Lorenzen. *Einführung in die operative Logik und Mathematik*, volume 78 of *Grundlehren der mathematischen Wissenschaften*. Springer, 1955.

[70] J. Łukasiewicz. O logice trójwartościowej. *Ruch Filozoficzny*, 5:170–171, 1920.

[71] J. Łukasiewicz. Philosophische Bemerkungen zu mehrwertigen Systemen des Aussagenkalküls. *Comptes rendus des séances, Soc. des sci. et des lett. de Varsovie*, 23:51–77, 1930.

[72] J. Łukasiewicz. Untersuchungen über den Aussagenkalkül. *Comptes rendus des séances, Soc. des sci. et des lett. de Varsovie*, 23:30–50, 1930.

[73] G. Metcalfe. An Avron rule for fragments of R-mingle. *J. Logic Comput.*, to appear.

[74] G. Metcalfe. Proof calculi for Casari's comparative logics. *J. Logic Comput.*, 16(4):405–422, 2006.

[75] G. Metcalfe and C. Röthlisberger. Admissibility in De Morgan algebras. *Soft Comput.*, 16(11):1875–1882, 2012.

[76] G. Metcalfe and C. Röthlisberger. Unifiability and admissibility in finite algebras. In *Proc. CiE 2012*, volume 7318 of *LNCS*, pages 485–495. Springer, 2012.

[77] G. Metcalfe and C. Röthlisberger. Admissibility in finitely generated quasivarieties. *Logical Methods in Computer Science*, 9 (19 pages), 2013.

[78] G.L. Miller. Graph isomorphism, general remarks. *J. Comput. System Sci.*, 18(2):128–142, 1979.

[79] G. Moisil. Recherches sur l'algébre de la logique. *Ann. Sci. Univ. Jassy*, 22(3):1–117, 1935.

[80] J.B. Nation. Finite sublattices of a free lattice. *Trans. Amer. Math. Soc.*, 269(1):311–337, 1982.

[81] J.S. Olson and J.G. Raftery. Positive Sugihara monoids. *Algebra Universalis*, 57:75–99, 2007.

[82] J.S. Olson, J.G. Raftery, and C.J. Van Alten. Structural completeness in substructural logics. *Log. J. IGPL*, 16(5):453–495, 2008.

[83] W.A. Pogorzelski. Structural completeness of the propositional calculus. *Bulletin de L'Académie Polonaise des Sciences*, 19(5), 1971.

[84] E.L. Post. Introduction to a General Theory of Elementary Propositions. *Amer. J. Math.*, 43(3):163–185, 1921.

[85] T. Prucnal. On the structural completeness of some pure implicational propositional calculi. *Studia Logica*, 32(1):45–50, 1973.

[86] T. Prucnal and A. Wronski. An algebraic characterization of the notion of structural completeness. *Bull. of the Sect. of Logic*, 3:30–33, 1974.

[87] A.P. Pynko. Algebraic study of Sette's maximal paraconsistent logic. *Studia Logica*, 54(1):89–128, 1995.

[88] A.P. Pynko. Implicational classes of De Morgan lattices. *Discrete Math.*, 205(1-3):171–181, 1999.

[89] W. Rautenberg. 2-element matrices. *Studia Logica*, 40(4):315–353 (1982), 1981.

[90] P. Ribenboim. Characterization of the sup-complement in a distributive lattice with last element. *Summa Brasil. Math.*, 2(4):43–49, 1949.

[91] C. Röthlisberger. Handbuch zur AWB. Master's thesis, University of Bern, 2005.
`www.algebraworkbench.net/download/Diplomarbeit.pdf`.

[92] C. Röthlisberger. Checking admissibility in finite algebras. In R.K Rendsvig and S. Katenko, editors, *Proceedings of the ESSLLI 2012 Student Session, Opole, Poland, August 6-17, 2012*, pages 133–141. CEUR Workshop Proceedings, 2012.

[93] C. Röthlisberger. TAFA – a tool for admissibility in finite algebras. In *Proc. TABLEAUX 2013*, volume 8123 of *LNAI*, pages 250–256. Springer, 2013. `https://sites.google.com/site/admissibility`.

[94] P. Rozière. *Regles Admissibles en calcul propositionnel intuitionniste.* PhD thesis, Université Paris VII, 1992.

[95] P. Rozière. Admissible and derivable rules in intuitionistic logic. *Mathematical Structures in Computer Science*, 2(3):129–136, 1993.

[96] V.V. Rybakov. A criterion for admissibility of rules in the modal system S4 and the intuitionistic logic. *Algebra and Logic*, 23:369–384, 1984.

[97] V.V. Rybakov. Bases of admissible rules of the logics S4 and Int. *Algebra i Logika*, 24(1):87–107, 123, 1985.

[98] V.V. Rybakov. Rules of inference with parameters for intuitionistic logic. *J. Symbolic Logic*, 57(3):912–923, 1992.

[99] V.V. Rybakov. *Admissibility of Logical Inference Rules*, volume 136 of *Studies in Logic and the Foundations of Mathematics*. Elsevier, Amsterdam, 1997.

[100] V.V. Rybakov. Construction of an explicit basis for rules admissible in modal system S4. *MLQ Math. Log. Q.*, 47(4):441–446, 2001.

[101] G. Salzer. MUltlog. `http://www.logic.at/multlog`.

[102] A.M. Sette. On the propositional calculus P^1. *Math. Japon.*, 18:173–180, 1973.

[103] B. Sobociński. Axiomatization of a partial system of three-valued calculus of propositions. *J. Computing Systems*, 1:23–55, 1952.

[104] M. Sprenger. AWB. `http://www.algebraworkbench.net`.

[105] A. Tarski. A remark on functionally free algebras. *Ann. of Math. (2)*, 47:163–165, 1946.

[106] P. Whitman. Free lattices. *Ann. of Math.*, 42:325–329, 1941.

[107] P. Wojtylak. A new proof of structural completeness of Łukasiewicz's logics. *Bulletin of the Section of Logic*, 5(4):145–152, 1976.

[108] P. Wojtylak. On structural completeness of the infinite-valued Łukasiewicz's propositional calculus. *Bulletin of the Section of Logic*, 5(4):153–157, 1976.

[109] P. Wojtylak. On structural completeness of many-valued logics. *Studia Logica*, 37(2):139–147, 1978.

[110] R. Zach. Proof theory of finite-valued logics. Master's thesis, Technische Universität Wien, 1993.

Index

Erklärung

gemäss Art. 28 Abs. 2 RSL 05

Name/Vorname	Röthlisberger Christoph
Matrikelnummer	99-114-498
Studiengang	Mathematik, Dissertation
Titel der Arbeit	Admissibility in Finitely Generated Quasivarieties
Leiter der Arbeit	Prof. Dr. George Metcalfe

Ich erkläre hiermit, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen benutzt habe. Alle Stellen, die wörtlich oder sinngemäss aus Quellen entnommen wurden, habe ich als solche gekennzeichnet. Mir ist bekannt, dass andernfalls der Senat gemäss Artikel 36 Absatz 1 Buchstabe o des Gesetzes vom 5. September 1996 über die Universität zum Entzug des auf Grund dieser Arbeit verliehenen Titels berechtigt ist.

Ort/Datum	**Unterschrift**
Bern, 19. November 2013	

Lebenslauf

Christoph Röthlisberger, geboren am 21. Juli 1979 in Sumiswald BE

07.10–heute	Peter Lehmann AG, Bärau *20%-Anstellung IT*
07.10–12.13	Mathematisches Institut, Universität Bern *Doktorstudium Mathematik*
07.10–09.13	Mathematisches Institut, Universität Bern *10%-Anstellung Assistent (Mathematik)*
11.05–07.10	Peter Lehmann AG, Bärau *100%-Anstellung IT*
10.02–02.03	IAM, Universität Bern *25%-Anstellung Hilfsassistent (Datenbanken)*
10.01–02.02	IAM, Universität Bern *50%-Anstellung Hilfsassistent (Neuronale Netze)*
10.99–11.05	Mathematisches Institut, Universität Bern *Diplomstudium Mathematik* *Informatik (60ETCS), Physik (21ECTS)*
08.95–06.99	Gymnasium, Burgdorf *Mathematisch-naturwissenschaftliche Maturität*
08.90–07.95	Sekundarschule, Langnau i.E.
08.86–07.90	Primarschule, Emmenmatt