Logical reflections

Profinite monoids, propositional quantifiers, and temporal operators

Sam van Gool

Manuscript for the *Habilitation à Diriger des Recherches* (HDR)
Université Paris Cité
December 6, 2024

Defended at Université Paris Cité on December 6, 2024.
Reviewers and jury members:

- **Valérie Berthé**

  Directrice de Recherche, CNRS, Université Paris Cité examinatrice

- **Jean Goubault-Larrecq**

  Professeur des Universités, Université Paris-Saclay rapporteur

- **Victoria Gould**

  Professeure, University of York rapportrice

- **Dexter Kozen**

  Professeur émérite, Cornell University rapporteur

- **Paul-André Melliès**

  Directeur de Recherche, CNRS, INRIA, Université Paris Cité examinateur

- **Carlos Simpson**

  Directeur de Recherche, CNRS, Université Côte d’Azur examinateur

- **Christine Tasson**

  Professeure des Universités, Institut Supérieur de l’Aéronautique et de l’Espace examinatrice

Contents

Introduction

In this document, I will survey some of the research in algebra, topology, logic, and the foundations of computer science that I have contributed to since completing my PhD in 2014, and I will also suggest directions for further research in this field. The work I report on here falls into three research themes: profinite monoids and their relationship to automata and regular languages (Chapter 1); uniform interpolation and its relationship to model-complete theories (Chapter 2); axiomatization and unifiability for temporal logics (Chapter 3). In each chapter, I will place the research theme in context, discuss my own results that fall under it, and, at the end of the chapter, I will provide an outlook on possible future research questions and directions that these results suggest. In this brief introduction to the document as a whole, I will only briefly comment on what I consider to be the most important common points between these three themes, which benefit greatly from cross-pollination.

I am generally interested in decidability and computational problems in logic for which the solution requires understanding their underlying mathematical invariants, typically algebraic and topological in nature. Specific instances in this document include: the solution of separation and covering problems for regular languages via pointlike sets in monoids (Section 1.2); the computation of uniform interpolants via a topological open mapping theorem (Section 2.1); and the solution of a unifiability problem in temporal logic via graph homomorphisms (Section 3.2). Many of these results contain applications of dualities between algebra and topology, which reflect the intimate bond between syntax and semantics in logic. An introduction to duality theory and a number of such applications are in the book [60], which we started writing in 2014, and which appeared earlier this year. In order to keep the current document self-contained, I recall the necessary notions from duality theory as I need them, and the overlap with [60] is small: This text is about the evolution of my research interests *beyond* the material in [60]. Furthermore, in the interest of maximizing accessibility, I avoid relying on category-theoretic terminology and techniques when possible, preferring to give elementary proofs that can be followed without too many preliminary definitions.

While the results discussed in this document take place in a variety of subfields, a thematic coherence can be found in the recurring appearance of certain mathematical objects and techniques. I highlight two particular instances of this. First, I often use concepts from *infinite model theory* and *universal algebra* for studying finitary phenomena, e.g., the use of saturated models for pseudofinite words in Section 1.1, adjunctions between compact congruences in Section 2.3, and model companions for temporal logic in Section 3.1. Second, *projective limits of finite structures* appear in various guises in each chapter, e.g., as profinite monoids in Section 1.1, as Esakia spaces in Section 2.1, and as generalized Cantor spaces in Section 3.2. For each of the problems solved in those sections, these profinite objects give a canonical, continuous, topological structure that allows for a mathematical, infinitary analysis of the particular, discrete, finite structures at hand. This kind of interplay between the finite and infinite, and between the discrete and continuous, never ceases to fascinate me, and I hope this document can convey some of that fascination to the reader.

Acknowledgments

First of all, I am immensely grateful to all the co-authors with whom I have had the joy to collaborate on the research contained in this document,[1] and beyond. In addition, I have been lucky to exchange scientific ideas with, and learn from, a number of colleagues who are not my co-authors (yet); several of them have also provided crucial support and advice to me at various stages of my post-doctoral career, especially since my arrival as a *maître de conférences* in France in 2019 and during the preparation of this HDR manuscript. The people who provide such mentorship to their peers are not always visible on a bibliography or curriculum vitae, but their words of support and advice have often been as important to my development as new scientific insights. I am very fortunate that all of these colleagues are not only highly skilled, imaginative, and original thinkers, but also, on the human level, incredibly generous, humorous, and loyal friends. If you ever meet any one of them, try asking them what they have been thinking about recently; you will probably hear a good story. Samson Abramsky, Quentin Aristote, Guillaume Baudart, Mikołaj Bojańczyk, Olivier Carton, Vincent Cheval, Thomas Colcombet, Pierre-Évariste Dagand, Laure Daviaud, Mirna Džamonja, Thomas Ehrhard, Hugo Férée, Marie Fortin, Wesley Fussner, Mai Gehrke, Silvio Ghilardi, Iris van der Giessen, Adrien Guatto, Rosalie Iemhoff, Peter Jipsen, Achim Jung, Ganna Kudryavtseva, Denis Kuperberg, Jérémie Marquès, Vincenzo Marra, Johannes Marti, Paul-André Melliès, George Metcalfe, Vincent Moreau, Rémi Morvan, Filippo Nuccio, Michele Pagani, Alessandra Palmigiano, Charles Paperman, Daniela Petrişan, Jean-Éric Pin, Hilary Priestley, Luca Reggio, Colin Riba, Sylvain Salvati, Simon Santschi, Alexis Saurin, Sylvain Schmitz, Ian Shillito, Mahsa Shirmohammadi, Benjamin Steinberg, Michelle Sweering, Ralf Treinen, Constantine Tsinakis, Yde Venema, Noam Zeilberger: Thank you.

Specifically regarding this HDR manuscript, there is one colleague whom I would like to thank a second time: Olivier Carton, thank you for all your encouragement during the writing of this manuscript, and also for your help with navigating the administrative steps required to formally defend it.

If this text fulfills its main purpose, it will officially allow me to direct research. In addition to those already mentioned above, I would like to thank all the Master, PhD students and postdocs who have so far allowed an un-habilitated researcher like myself to advise them on their research.

I would also like to thank both the institute for research in the foundations of computer science, *IRIF*, and the computer science teaching and research department, *UFR d'informatique*, of Université Paris Cité, and all members of these institutions, for providing me with such a welcoming and inspiring working environment. My special thanks go to the administrative staff of both institutions, who have provided crucial support for many tasks since my arrival, helping me to free up time to do research. Omur Avci, Thomas Beraud, Juliette Calvi, Natalia Hacquart, Marie-José Iarifina, Houy Kuoy, Maximilien Lesellier, Eva Ryckelynck, Jemuel Samtchar, Marie-Laure Susairaj: Thank you.

[1] The main text contains substantial original parts, and also draws from articles published by myself with various collaborators. I provide references to the relevant articles at the start of each section, and then freely cite from them without repeating the reference at every citation.

I gratefully acknowledge the financial support received from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program, grant agreements #655941 and #670624, and from the Agence Nationale de Recherche project ANR-23-CE48-0012.

I thank the reviewers of this manuscript, Jean Goubault-Larrecq, Victoria Gould, and Dexter Kozen, for accepting to perform this task, in the middle of summer, no less, and for producing such thoughtful and detailed reports in the fall. I would also like to thank Valérie Berthé, Paul-André Melliès, Carlos Simpson, and Christine Tasson, for accepting to be members of the jury of this manuscript's defense, just before the start of winter.

Finally, I thank my family and friends for their love and support. Even if this document may contain some things that you do not understand, you all contributed greatly to its existence.

Paris, December 2024

1 Monoids: Profiniteness, models and pointlikes

Whereas groups are gems, all of them precious, the garden of semigroups is filled with weeds. One needs to yank out these weeds to find the interesting semigroups.

– *J. Rhodes & B. Steinberg* [148, p. 2]

The algebraic structure of a *monoid*, that is, a set equipped with an associative binary operation and a neutral element for it, is central to the study of finite words. Indeed, the set of finite words over a fixed alphabet Σ, denoted Σ^*, carries the structure of a *free* monoid over Σ, namely, the binary operation of concatenation, with neutral element the empty word, ϵ. A foundational fact in the study of automata is that a set L of finite words over a finite alphabet Σ is recognizable by a finite automaton if, and only if, there exists a congruence relation θ on Σ^* such that the quotient Σ^*/θ is finite and L is a union of θ-classes. A set L that satisfies these conditions is called *regular*, referring to the fact that such sets are also exactly the ones definable using regular expressions.[1]

Our focus in this chapter will be on the interplay between monoids, regular sets, and their definability using *logic*. Here, a logical formula is used to define a *language*, namely, the set of those finite words which satisfy the formula. To give an example, the formula

$$\exists x. \, ((\forall y. \, x \leq y) \wedge a(x)) \tag{1.1}$$

is interpreted in a finite word as: "there exists a position, x, such that all other positions, y, come after it, and the letter at position x is a". Thus, the formula in Eq. (1.1) *defines* the language $a\Sigma^*$, that is, the set of finite, non-empty words whose first letter is a.

In this way, logics can be used as a measure of the complexity of a set of finite words: If a set L is definable in a 'simple' logic, then it is understood to have 'low' complexity. More precisely, we will be interested in sets definable in *monadic second-order* logic over finite linear orders, and its *first-order* fragment, to be defined in more detail in Section 1.1 below. Monadic second-order definable sets turn out to be precisely the regular sets of finite words, and thus give yet another characterization of regular sets, in addition to those via automata, monoids, and regular expressions already mentioned. Among the monadic second-order definable sets, the first-order definable sets can be characterized as precisely those which can be recognized by a finite monoid containing no non-trivial subgroups; again, more precise definitions and statements will be given in Section 1.1.[2]

[1] The facts stated in this paragraph are sometimes collectively referred to as the *Kleene-Schützenberger theorem*, as they are rooted in [106, 157]. See [137, Ch. 1] for a modern account, and [135] for details about the history.

[2] The history of the results in this paragraph is complex, and explained in depth in [138, 165, 170]. In short, the monadic second-order result goes back to [27, 49, 171], while the first-order result is essentially due to [156], taking into account also the works [126, 127].

Historically, these results were the starting point for a study of *decidability* results at the interface of logic and regular languages. In this context, the *membership problem* for a class $\mathcal{C}$ of regular languages is the computational problem that asks, given as input a regular language L, to determine whether or not L belongs to $\mathcal{C}$. For instance, when $\mathcal{C}$ is the class of first-order definable languages, the above characterization via monoids leads to a simple algorithm to decide the membership problem for $\mathcal{C}$: Given L, first compute a *minimal* finite monoid recognizing L, and then check whether or not it contains non-trivial subgroups. This minimal monoid is called the *syntactic* monoid for L, and it is classical [144] that this monoid can indeed be computed from a description of L, be it via an automaton, regular expression, or monadic second-order formula.[3]

A generalization of the membership problem is the *separation problem* for a class $\mathcal{C}$, which asks, given two disjoint regular sets L_1 and L_2, to decide whether or not there exists a set $K \in \mathcal{C}$ such that $L_1 \subseteq K$ and $L_2 \cap K = \emptyset$. In algebraic terms, this problem is closely related to the question of determining whether or not a given subset of a finite monoid is *pointlike* with respect to the class $\mathcal{C}$, see further Section 1.2. On the logic side, separation problems are closely related to *interpolation*, in the sense of Chapter 2. I will come back to this at the end of Section 3.3.

In this chapter, I will explain a number of my contributions to the theory of finite and profinite monoids, most of them developed between 2016 and 2018, during my postdoc with B. Steinberg at City College of City University of New York:

1. A model-theoretic point of view on pro-aperiodic monoids, with an application to deciding equality of ω-terms (Section 1.1);
2. Decidability of separation for logics defined via groups, using pointlike sets (Section 1.2).

Before coming to these results, I will briefly recall some basics on the theory of profinite monoids and logic on words, which I need in the rest of the chapter.

1.1 Proaperiodic monoids and saturated models

The correspondence between first-order definable sets and aperiodic monoids was a starting point for a study of the general correspondence between classes of finite monoids and classes of regular languages, now referred to as Eilenberg pseudovariety theory [48]. Here, a *pseudovariety* is a class of finite monoids closed under homomorphic images, submonoids, and finite products. *Profinite monoids* naturally emerge as the canonical free objects for pseudovarieties, and provide the appropriate language for expressing equational properties of classes of finite monoids. A *topological monoid* is a monoid equipped with a topology such that the multiplication on M is continuous when regarded as a function from $M \times M$ to M. A topological monoid is *profinite* if, and only if, its underlying topology is *Boolean*[4], by which we mean: compact, Hausdorff, and zero-dimensional, i.e., having a basis consisting of sets that are both closed and open (clopen).[5] Note that any finite monoid is profinite when

[3] For more about syntactic monoids, see [138], and see [60, Sec. 8.1] for a textbook account of the connection between syntactic monoids and duality theory, on which the work in Section 1.1 below builds.

[4] Boolean spaces are also known as *Stone spaces* in the literature.

[5] The definition of profinite monoid we give here is the one that is easiest to state, but it does not generalize to other algebraic structures. The generally correct definition says that an algebra is profinite when it is an inverse limit of finite discrete algebras in the category of topological algebras. In general, one then needs to carefully distinguish between profinite algebras and topological algebras whose underlying topology is Boolean, since the first class may be strictly

equipped with the discrete topology, and, since this is the only Hausdorff topology on a finite set, we will always tacitly assume that the topology on a finite monoid is discrete.

Definition 1.1. A *free profinite monoid* over Σ is an embedding of Σ into a topological monoid, Σ^{pro}, such that, for every profinite monoid M and function $h : \Sigma \to M$, there exists a unique continuous monoid homomorphism $\widehat{f} : \Sigma^{\mathsf{pro}} \to M$ that extends f, as in the following diagram:

$$\begin{tikzcd}
\Sigma^{\mathsf{pro}} \arrow[dr, dashed, "\widehat{f}"] & \\
\Sigma \arrow[u, hook] \arrow[r, "f"'] & M
\end{tikzcd}$$

Other traditional notations for Σ^{pro} are $\overline{\Omega}_\Sigma\mathbf{M}$ and $\widehat{\Sigma^*}$. For general abstract reasons, a free profinite monoid is unique up to an isomorphism of topological monoids, see, e.g., [60, Exercise 8.2.9]. From the above abstract definition, it is not clear a priori that the free profinite monoid Σ^{pro} exists. There are a number of ways to construct Σ^{pro}; notably, as a completion of Σ^* under an appropriate metric or uniform structure, see, e.g., [137, Ch. 17]. For what we will do below, however, the most convenient construction of Σ^{pro} is as $\mathsf{spec}\,\mathsf{Reg}_\Sigma$, i.e., the Stone dual space of the Boolean algebra of regular languages over the alphabet Σ, equipped with a continuous multiplication. Here, recall that the *Stone dual space* $\mathsf{spec}\,B$ of a Boolean algebra B is the set of ultrafilters, equipped with the Boolean topology that is generated by declaring each set $\widehat{a} \overset{\text{def}}{=} \{x \in \mathsf{spec}\,B \mid a \in x\}$, for $a \in B$, to be open.[6] In terms of duality theory for Boolean algebras with operators, one can define this multiplication as dual to a 'residuation structure' on the regular languages, as was proved in [62].[7] I will briefly recall one concrete way to do this, which we explained in more detail in [60, Sec 8.2].

First note that the free monoid Σ^* embeds into $\mathsf{spec}\,\mathsf{Reg}_\Sigma$ via the map that sends a finite word w to the ultrafilter of regular sets that contain the word w. The image of this embedding $\Sigma^* \hookrightarrow \mathsf{spec}\,\mathsf{Reg}_\Sigma$ is a dense subspace. It follows that, for any homomorphism $h : \Sigma^* \to M$, with M a finite monoid, there exists a unique continuous function $\overline{h} : \mathsf{spec}\,\mathsf{Reg}_\Sigma \to M$. Now, one can prove that, for any ultrafilters u and v of Reg_Σ, there is a unique ultrafilter $u \cdot v$ of Reg_Σ such that, for every $h : \Sigma^* \twoheadrightarrow M$, $\overline{h}(u \cdot v) = \overline{h}(u) \cdot \overline{h}(v)$ [60, Cor. 8.37]. This defines a continuous monoid multiplication on $\mathsf{spec}\,\mathsf{Reg}_\Sigma$, and we have the following; a detailed proof is given in [60, Lem. 8.38].

Theorem 1.2. *The topological monoid* $\mathsf{spec}\,\mathsf{Reg}_\Sigma$ *is the free profinite monoid over* Σ*:*

$$\Sigma^{\mathsf{pro}} \cong \mathsf{spec}\,\mathsf{Reg}_\Sigma\ .$$

Our aim in this section is to use the point of view of Theorem 1.2 to study free pro-*aperiodic* monoids via model theory.

contained in the second. In the case of monoids, however, the two classes coincide. For more about the general case, see, e.g., [10].

[6] In this chapter, we only use Stone duality for Boolean algebras. We will say more about Stone duality, for more general lattice-based structures than Boolean algebras, in Section 2.1.

[7] The importance of profinite monoids in automata theory and finite monoid theory was first highlighted by Almeida, starting in the late 1980s; see his influential book [3], and the monograph [148], for more background. In more recent years, a number of authors have made explicit use of duality theory to redevelop and expand the foundations of the profinite approach to studying varieties of languages; most closely related to our work here are [62] and [148, Chapter 8], also see [25].

Logical definability and recognizability. We first recall how monadic second-order logic can be used to capture the notion of *regularity*. We will then focus on the subclass of regular sets definable in first-order logic, which turn out to be exactly the aperiodic-recognizable ones, see Theorem 1.5 below.

Let $\bar{x}$ and $\bar{P}$ be two sequences of symbols, whose elements will be called first- and second-order variables, respectively. An *atomic formula* is any expression which is either of the form $P(x)$, where x is a symbol in $\bar{x}$ and P is a symbol in $\bar{P}$, or of the form $x < y$, where both x and y are symbols in $\bar{x}$. A *first-order formula* is defined recursively to be either an atomic formula, or an expression of the form $\phi \vee \psi$, $\neg\phi$, or $\exists x.\phi$, where ϕ and ψ are first-order formulas and x is a first-order variable. A *monadic second-order formula* is defined in the same way, further allowing monadic second-order quantification, $\exists P.\phi$, where P is a second-order variable. A variable is *free* in a formula ϕ if it does not occur under the scope of a quantifier. The *set of first-order formulas* with free variables among $\bar{x}, \bar{P}$ is denoted $\mathsf{FO}(\bar{x}, \bar{P})$ and the analogous *set of monadic second-order formulas* is denoted $\mathsf{MSO}(\bar{x}, \bar{P})$. A *first-order sentence* is a first-order formula in which no first-order variable x occurs freely; we denote by $\mathsf{FO}(\bar{P})$ the set of first-order sentences with free second-order variables among $\bar{P}$.[8]

We recall how monadic second-order formulas are naturally interpreted in finite words. This requires first defining some notation for the relevant alphabets and maps between them. For $\bar{P}$ a finite sequence of second-order variables, we define $\Sigma_{\bar{P}} \overset{\text{def}}{=} \mathbf{2}^{\bar{P}}$, that is, a letter $a \in \Sigma_{\bar{P}}$ is a $\bar{P}$-indexed string of bits. Note that, then, a word w in $(\Sigma_{\bar{P}})^*$ contains two distinct levels of sequences: a bit-string of length $|\bar{P}|$ gives a single letter, and a sequence of such letters gives a word. For instance, when $\bar{P} = (P, Q, R)$, we have the word 011 100 which has length 2, and whose letter at the second position is $(P \mapsto 1, Q \mapsto 0, R \mapsto 0)$.

If $\bar{Q}$ is a subsequence of $\bar{P}$, then for any $a \in \Sigma_{\bar{P}}$, we write $a|_{\bar{Q}}$ for the restriction of the string a to the domain $\bar{Q}$. Thus, we get a function $(-)|_{\bar{Q}} : \Sigma_{\bar{P}} \to \Sigma_{\bar{Q}}$, which extends uniquely to a letter-to-letter homomorphism $(\Sigma_{\bar{P}})^* \to (\Sigma_{\bar{Q}})^*$, and which we also denote by $(-)|_{\bar{Q}}$. Now, when $\bar{x}$ is a set of first-order variables and $w \in (\Sigma_{\bar{P}})^*$, by a *valuation* of $\bar{x}$ in w we mean a function that associates with every x in $\bar{x}$ a position $v(x)$ in w. Write $W(\bar{x}, \bar{P})$ for the set of words-with-valuations, that is, pairs (w, v) where $w \in (\Sigma_{\bar{P}})^*$ and v is a valuation of $\bar{x}$ in w. With these notations in place, we define a *semantics* function

$$[\![-]\!] : \mathsf{MSO}(\bar{x}, \bar{P}) \to \mathcal{P}(W(\bar{x}, \bar{P})) ,$$

as follows. For the atomic cases, $(w, v) \in [\![P(x)]\!]$ if, and only if, the index P bit of w at position $v(x)$ is 1, and $(w, v) \in [\![x < y]\!]$ if, and only if, the position $v(x)$ is to the left of the position $v(y)$. The cases $\vee$, $\neg$, and $\exists$ are standard, by induction: $[\![\phi \vee \psi]\!] \overset{\text{def}}{=} [\![\phi]\!] \cup [\![\psi]\!]$, $[\![\neg\phi]\!] \overset{\text{def}}{=} [\![\phi]\!]^{\mathrm{c}}$, and $(w, v) \in [\![\exists x.\phi]\!]$ if, and only if, there exists a position p in w such that $(w, v_{x \mapsto p}) \in [\![\phi]\!]$, where $v_{x \mapsto p}$ denotes the modification of v that sends the variable x to p. For the case of a second-order quantifier, we extend the alphabet: $(w, v) \in [\![\exists Q.\phi]\!]$ if, and only if, there exists a word $\tilde{w} \in (\Sigma_{\bar{P} \cup \{Q\}})^*$ such that $(\tilde{w}, v) \in [\![\phi]\!]$ and $\tilde{w}|_{\bar{P}} = w$.

A subset L of $W(\bar{x}, \bar{P})$ is called *monadic second-order definable* if it lies in the image of the function $[\![-]\!]$. Note that, in particular, $W(\emptyset, \bar{P})$ is just the set of finite words over the alphabet $\Sigma_{\bar{P}}$, so that we can speak of monadic second-order definable sets of finite words over $\Sigma_{\bar{P}}$. In order for this definition to be meaningful when Σ is an arbitrary finite alphabet, we will always tacitly assume that Σ comes with an arbitrary fixed injection $\Sigma \hookrightarrow \Sigma_{\bar{P}}$, for some sufficiently large finite sequence of variables $\bar{P}$.

[8] We use some common macro formulas: We write $x = y$ for $\neg((x < y) \vee (y < x))$, $x \leq y$ for $(x < y) \vee (x = y)$, $\phi \wedge \psi$ for $\neg(\neg\phi \vee \neg\psi)$, $\phi \to \psi$ for $\neg\phi \vee \psi$, $\phi \leftrightarrow \psi$ for $(\phi \wedge \psi) \vee (\neg\phi \wedge \neg\psi)$, and $\forall x.\phi$ for $\neg\exists x.\neg\phi$.

This allows us to regard, for any $a \in \Sigma$, the expression $a(x)$ as a formula which is true exactly when the letter at position x is a; more formally, $a(x)$ is a macro for $\bigwedge_{P : a(P)=1} P(x) \wedge \bigwedge_{Q : a(Q)=0} \neg Q(x)$. We then also use the notations $\mathsf{MSO}(\Sigma)$ and $\mathsf{FO}(\Sigma)$ for the set of monadic second-order and first-order sentences over the alphabet Σ. We also note explicitly that the empty word is allowed as a model. The set of non-empty words is first-order definable, for example by the sentence $\exists x.(P(x) \vee \neg P(x))$.

Theorem 1.3 ([27]). *A set of finite words is monadic second-order definable if, and only if, it is regular.*

Example 1.4. The semantics of the formula $\exists x. ((\forall y.\, x \leq y) \wedge \neg P(x) \wedge Q(x))$ is the set of non-empty words in $(\Sigma_{\{P,Q\}})^*$ such that the letter at the first position is the bit-string 01. Let us now define the formulas

$$F(x) \overset{\text{def}}{=} \forall t.\, x \leq t, \quad L(y) \overset{\text{def}}{=} \forall t.\, t \leq y, \text{ and}$$

$$S(x,y) \overset{\text{def}}{=} (x < y) \wedge \forall t.\, \neg(x < t) \vee \neg(t < y).$$

Note that $(w, v) \in [\![F(x)]\!]$ if and only if $v(x)$ is the first position in w, $(w, v) \in [\![L(y)]\!]$ if and only if $v(y)$ is the last position in w, and $(w, v) \in [\![S(x, y)]\!]$ if and only if $v(y)$ is the successor position of $v(x)$. Now consider the sentence

$$E \overset{\text{def}}{=} \exists X.\, [\exists x.\, (X(x) \wedge F(x))] \wedge [\exists y.\, (X(y) \wedge L(y))] \wedge [\forall x.\, \forall y.\, S(x,y) \rightarrow (X(x) \leftrightarrow \neg X(y))]\,.$$

The sentence E says: ‘there exists a subset, X, which contains the first and the last position, and which contains a position x if and only if it does not contain the successor position of x’. Thus, E defines the set of words of odd length. One may similarly define, for example, the set of words w in $(\Sigma_{\{P,Q\}})^*$ such that the bit for P is 1 at an odd number of positions in w.

In light of Theorem 1.3, it is now natural to ask which regular sets can be defined by a *first-order* formula. The answer comes from the following foundational theorem of Schützenberger [156], combined with the logical point of view [126, 127].

Theorem 1.5. *A set of finite words is first-order definable if, and only if, it can be recognized by a finite aperiodic monoid.*

Here and in what follows, when Σ is a finite alphabet and $L \subseteq \Sigma^*$, we say that a monoid M *recognizes* L if there is a homomorphism $h : \Sigma^* \to M$ such that $h^{-1}(h[L]) = L$. A *subgroup* G of a monoid M is, by definition, a subset closed under the multiplication of M, but not necessarily containing the neutral element, so that G with the restricted multiplication is a group. A monoid is called *aperiodic* if all of its subgroups are trivial.

Example 1.6. To illustrate the above definitions, we consider the set E of words of odd length in the alphabet $\Sigma \overset{\text{def}}{=} \{a\}$. We first show how to recognize it, non-optimally, with a submonoid of the monoid $\mathsf{End}(\mathbf{3})$ of functions from the set $\mathbf{3} = \{0,1,2\}$ to itself, under function composition. Let a be the following element of $\mathsf{End}(\mathbf{3})$ (see Figure 1.1):

$$a \overset{\text{def}}{=} (0 \mapsto 1, 1 \mapsto 0, 2 \mapsto 1).$$

The submonoid M of $\mathsf{End}(\mathbf{3})$ that is generated by a contains the identity function, e, and one further function, $a^2 = (0 \mapsto 0, 1 \mapsto 1, 2 \mapsto 0)$. Write h for the homomorphism from $\{a\}^*$ to M which sends, for

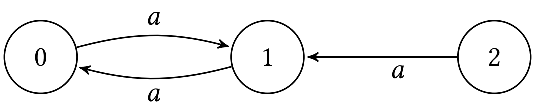


Figure 1.1: A deterministic automaton on three states and one letter.

$n \geq 0$, a word a^n to the n-fold composition of a, that is, h sends the empty word to e, $h(a^n) = a$ if n is odd and $h(a^n) = a^2$ if n is even and non-zero. Thus, $E = h^{-1}(\{a\})$, and the homomorphism h recognizes E. We now explain why this particular recognizing monoid M is non-optimal: While M has three elements, the two-element subgroup $\{a, a^2\}$ could also be used to recognize this same set E, modifying h to h' which sends the empty word to a^2, and noting that h' is still a homomorphism. Using a little more theory of finite monoids, which we omit here, see e.g. [137, Ch. 1], one may establish that the *syntactic monoid* of E is isomorphic to the two-element group, and thus, by Theorem 1.5, E cannot be first-order definable: any finite monoid recognizing E must contain an even subgroup, and is therefore not aperiodic.

A logical view on the free proaperiodic monoid. In Proposition 1.2, we saw that elements of Σ^{pro}, which are sometimes called *profinite words*, can be realized as ultrafilters of regular sets. In the rest of this section, I report on our work in [86, 87], where we gave a model-theoretic perspective on profinite monoids. Since model theory is most readily applicable to first-order logic, in view of Theorem 1.5 we restrict our attention in this section to aperiodic monoids.[9] A *proaperiodic* monoid is a profinite monoid which is aperiodic. The *free proaperiodic monoid* generated by a finite set Σ is a proaperiodic monoid Σ^{ap} containing Σ such that any function $f : \Sigma \to M$, with M a finite aperiodic monoid, extends uniquely to a continuous homomorphism $\overline{f} : \Sigma^{\mathsf{ap}} \to M$. The free proaperiodic monoid is unique up to topological isomorphism, and the same extension property still holds if in the previous sentence M is replaced by an arbitrary proaperiodic monoid. Let us now write AP_Σ for the Boolean algebra of regular languages which are recognizable by a finite aperiodic monoid. An aperiodic version of Theorem 1.2, which can be proved in the same way, is the following.

Theorem 1.7. *The topological monoid* $\mathsf{spec}\ \mathsf{AP}_\Sigma$ *is the free proaperiodic monoid over* Σ.

To give a logical perspective on Theorem 1.7, recall the semantics function, which associates in particular with any first-order sentence $\phi \in \mathsf{FO}(\Sigma)$ a set of words $[\![\phi]\!]$ in the alphabet Σ. Theorem 1.5 tells us that the image of this function is exactly AP_Σ. Write $\phi \sim_{\mathsf{FO}} \psi$ if, and only if, $[\![\phi]\!] = [\![\psi]\!]$. The first isomorphism theorem of Boolean algebras yields

$$\mathsf{FO}(\Sigma)/{\sim_{\mathsf{FO}}} \quad \cong \quad \mathsf{AP}_\Sigma\,. \tag{1.2}$$

The Boolean algebra on the left is a special case of the well-known construction in logic of the *Lindenbaum-Tarski algebra* with respect to the following first-order theory:

$$T_\Sigma^{\mathrm{fin}} \overset{\mathrm{def}}{=} \{\phi \in \mathsf{FO}(\Sigma) \mid [\![\phi]\!] = \Sigma^*\}\,,$$

[9] After publication of our work in [84, 86], some of the results we give here were extended to monadic second-order logic [111, 112], and also to larger classes of possibly non-aperiodic profinite monoids in [7]; further see Section 1.3.

that is, $T^{\mathrm{fin}}_{\Sigma}$ is the set of first-order sentences that hold in every finite word. We will axiomatize and further describe $T^{\mathrm{fin}}_{\Sigma}$ below.

The dual space of the Lindenbaum-Tarski algebra, $\mathsf{spec}(\mathsf{FO}(\Sigma)/{\sim_{\mathsf{FO}}})$, is also well-known in logic: points of this space are complete theories extending $T^{\mathrm{fin}}_{\Sigma}$, also known as 0-*types*. A more concrete view on these points can be given by considering *models of* $T^{\mathrm{fin}}_{\Sigma}$, which, following usual terminology in model theory, will be called *pseudofinite words*. In our setting, a *model* is a relational structure $W = (|W|, <^W, (W_a)_{a\in\Sigma})$, where $|W|$ is any set, $<^W$ is a binary relation on $|W|$, and each W_a is a unary relation, that is, a subset of $|W|$. Note that the above definition of semantics for first order formulas in fact never used that the base order of a word was finite, so that any first-order sentence has a well-defined truth value in any given model. We say that a model W is a *pseudofinite word* over Σ if all sentences in $T^{\mathrm{fin}}_{\Sigma}$ are true in W. Two models W and W' are called *elementarily equivalent*, notation $W \equiv W'$, if exactly the same first-order sentences are true in W and W'. We will abbreviate 'elementary equivalence class' to 'class'.

With this terminology in place, one may now use the completeness theorem of first-order logic to prove that points of $\mathsf{spec}(\mathsf{FO}(\Sigma)/{\sim_{\mathsf{FO}}})$ exactly correspond to classes of pseudofinite words. Applying the Stone duality functor spec to both sides of the Boolean algebra isomorphism Eq. (1.2), we obtain a homeomorphism between the space of classes of pseudofinite words and the free proaperiodic monoid Σ^{ap}, see [84, Thm. 2.4].

Note that, for now, this is a homeomorphism, but it is not clear yet how to interpret the monoid multiplication on the space of classes of pseudofinite words. To do so, we need to understand more concretely what a pseudofinite word is. As a first approximation, a pseudofinite word is at least a Σ-*labeled pseudofinite linear order*, i.e., a model $(W, <^W, (W_a)_{a\in\Sigma})$ in which $<^W$ is a discrete linear order with endpoints, and $(W_a)_{a\in\Sigma}$ is a partition[10] of the set W. We will call a Σ-labeled pseudofinite linear order a Σ-*word*. In a Σ-word, we will write $W(i)$ for the unique letter $a \in \Sigma$ such that $i \in W_a$. The following is an example of a Σ-word that is *not* a pseudofinite word.

Example 1.8. Let W be the word over the alphabet $\{a, b\}$ with underlying order $\mathbb{N} + \mathbb{N}^{\mathrm{op}}$, where $W(i) = a$ for all $i \in \mathbb{N}$ and $W(i) = b$ for all $i \in \mathbb{N}^{\mathrm{op}}$; visually, W is the word

$$aaaa\ldots \quad \ldots bbbb.$$

The sentence

$$\exists x.a(x) \to \exists x.\big[a(x) \land \forall y.(y > x \to \neg a(y))\big]$$

expressing 'if there exists an a-position, then there exists a last such' is true in every finite word, and therefore lies in $T^{\mathrm{fin}}_{\Sigma}$, but it fails to hold in W. Thus, W is not pseudofinite.

Extending the idea of Example 1.8, we show in [84, Thm. 4.1] that the theory $T^{\mathrm{fin}}_{\Sigma}$ is not finitely axiomatizable. The reason for this is that $T^{\mathrm{fin}}_{\Sigma}$ contains a *first-order induction principle*, namely, for every first-order formula $\phi(x)$ in one free variable, the first-order sentence

$$\mathrm{Last}_{\phi}: \qquad \exists x.\phi(x) \to \exists x.\big[\phi(x) \land \forall y.(y > x \to \neg\phi(y))\big]$$

[10] Here, and in what follows, we call a collection $\mathcal{K}$ of subsets of a set W a *partition* if $\bigcup \mathcal{K} = W$ and any pair of distinct sets in $\mathcal{K}$ are pairwise disjoint. Note that, for us, a partition may have the empty set as one of its elements.

is in T_Σ^{fin}. This sentence expresses the fact that, in a finite word, every first-order property that happens at least once, must happen a last time. The theory T_Σ^{fin} is axiomatized by adding to the theory of Σ-labeled pseudofinite linear orders the set of sentences Last_ϕ, where ϕ ranges over all first-order formulas with one free variable [84, Prop. 4.2].

Multiplication and substitution of pseudofinite words. We established above that Σ^{ap}, the free proaperiodic monoid, is homeomorphic to the space of classes of pseudofinite words, we identify the two from now on. The first main advantage of viewing Σ^{ap} in this way is that the monoid structure on this topological space becomes more tangible than when working with the abstract definition of Σ^{ap}, as I will illustrate now.

Suppose that V is a word over a finite alphabet Δ, and that for each $b \in \Delta$, U_b is a word over a finite alphabet Σ. We obtain a new word $V[b/U_b]$ over Σ by substituting the word U_b for each occurrence of the letter b in V; see Figure 1.2.

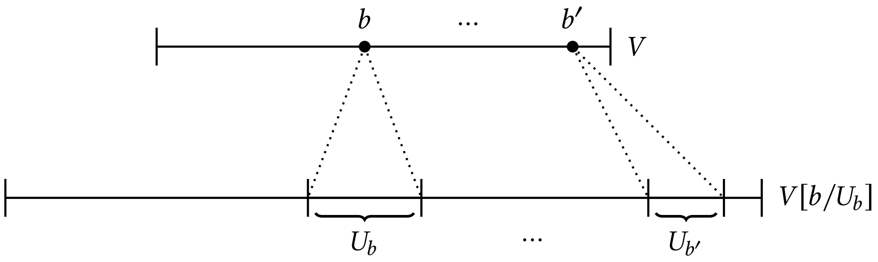


Figure 1.2: Substituting $(U_b)_{b\in B}$ into V

Formally, the *substitution* of the Σ-words $(U_b)_{b\in\Delta}$ into the Δ-word V is the Σ-word $W = V[b/U_b]$ defined as follows.

- The underlying order of W is the *lexicographic order* on the disjoint union $|W| := \bigsqcup_{i\in|V|} |U_{V(i)}|$, i.e.,
$$(i,j) <^W (i',j') \overset{\text{def}}{\iff} i <^V i', \text{ or } i = i' \text{ and } j <^{U_{V(i)}} j'.$$
- The letter at position (i,j) in W is the letter at position j in $U_{V(i)}$.

There are two important special cases of substitution. If U_0 and U_1 are Σ-words, then the *concatenation* $U_0 \cdot U_1$ of U_0 and U_1 is defined as the substitution of $(U_b)_{b\in\{0,1\}}$ into the $\{0,1\}$-word 01. If U is a Σ-word and λ is a discrete linear order with endpoints, then the *λ-power* U^λ of U is defined as the substitution of $U_b = U$ into the unique $\{b\}$-word with underlying order λ. We then prove the following crucial proposition [84, Prop. 3.7]:

Proposition 1.9. *Let $(U_b)_{b\in\Delta}$ be a Δ-indexed collection of pseudofinite Σ-words. Then the function $f : \Delta^{\mathsf{ap}} \to \Sigma^{\mathsf{ap}}$ which sends an element $[V]_\equiv$ of Δ^{ap} to $[V[b/U_b]]_\equiv$ is a well-defined continuous homomorphism. Moreover, any continuous homomorphism from Δ^{ap} to Σ^{ap} arises in this manner.*

Before commenting on the proof of Proposition 1.9, let us draw two important consequences. First, it follows from the fact that concatenation is a special case of substitution that the multiplication on Σ^{ap} is precisely concatenation of pseudofinite words, which is well-defined on classes. Thus, we obtain the following result.

Theorem 1.10. *The topological monoid of classes of pseudofinite words, under concatenation, is the free proaperiodic monoid* Σ^{ap} *over* Σ.

As a further consequence of Proposition 1.9, we obtain a concrete description of the ω*-power* operation on Σ^{ap}. To explain what this is, recall first that, in any finite monoid M, any element $x \in M$ has an idempotent power, that is, there exists $n \geq 1$ such that $x^{2n} = x^n$. Moreover, the set $\{x^m \mid m \geq n\}$ will always form a cyclic subgroup of M, and thus, if M is aperiodic, then $x^n = x^{n+1}$. By general topological principles, these arguments transfer to any *pro*finite monoid, that is, for any element x in a profinite monoid M, there exists a unique idempotent element in the closure of $\{x^n \mid n \geq 1\}$. This unique idempotent is called the ω*-power* of x and denoted x^ω. The ω-power operation can be used to define classes of monoids via profinite identities; for instance, aperiodic finite monoids are exactly those satisfying the profinite identity $x^\omega = x^{\omega+1}$. In fact, a profinite version of Birkhoff's theorem, due to [147], says that *any* pseudovariety can be defined by profinite identities; see, e.g., [3, Thm. 3.5.1], for a proof.

Applying Proposition 1.9 in the specific case of a λ-power, with λ any infinite pseudofinite linear order, allows us to calculate the element x^ω. For example, if x is the finite word ab, then the element $(ab)^\omega$ of Σ^{ap} is the class of the pseudofinite word depicted visually as

$$ababab \quad \ldots \quad ababab\ .$$

One may use these concrete incarnations of product as concatenation and ω-power as substitution into an infinite order to reason combinatorially about equations in Σ^{ap}. For example, an equation such as $(xy)^\omega x = x(yx)^\omega$ is now easily seen to be valid for any $x, y \in \Sigma^{\mathsf{ap}}$: If X and Y, respectively, are pseudofinite words in the classes x and y, then one verifies that the pseudofinite words that realize either side of the equation are isomorphic. We exploit this idea further below to obtain a proof of decidability of the word problem for aperiodic ω-terms.

We now comment on the proof of Proposition 1.9, proved in detail in [84, Prop. 3.7]. An important ingredient in that proof is the standard logic technique of Ehrenfeucht-Fraïssé games with a bounded number of rounds. Without going into the details of this proof here, we note that these games essentially describe the elementary equivalence relation $\equiv$ as the intersection of a decreasing sequence of finite-index equivalence relations $\equiv_k$, with $k \in \mathbb{N}$, where two models W and W' are equivalent in $\equiv_k$ if, and only if, the same first-order sentences of quantifier rank $\leq k$ are true in W and W'. Here, the *quantifier rank* of a formula ϕ is the maximum nesting depth of quantifiers in ϕ. Ehrenfeucht-Fraïssé games give an inductive way of characterizing the relation $\equiv_{k+1}$ in terms of the relation $\equiv_k$. This allows for a concrete approach to various general properties. For instance, one may show by induction on k that, for any pseudofinite words U, U', V, if $U \equiv_k U'$, then $UV \equiv_k U'V$. The proof of Proposition 1.9 generalizes this to arbitrary substitutions. As another application of the use of the sequence of relations $\equiv_k$, our strategy to prove that T_Σ^{fin} is not finitely axiomatizable [84, Thm. 4.1] is to construct a sequence of non-pseudofinite models W_k such that, for every k, W_k is $\equiv_k$-equivalent to some finite word.

In more algebraic terms, this sequence of equivalence relations $\equiv_k$ gives a chain of finite aperiodic monoids Σ_k^{ap}, consisting of the $\equiv_k$-classes of pseudofinite words. The free proaperiodic monoid Σ^{ap} is then realized as the limit, in the category of topological monoids, of the inverse chain of finite discrete monoids

$$\Sigma^{\mathsf{ap}} \twoheadrightarrow \quad \cdots \quad \twoheadrightarrow \Sigma^{\mathsf{ap}}_{k+1} \twoheadrightarrow \Sigma^{\mathsf{ap}}_{k} \twoheadrightarrow \quad \cdots \quad \twoheadrightarrow \Sigma^{\mathsf{ap}}_{0}. \tag{1.3}$$

In terms of finite semigroups, the finite monoids Σ^{ap}_k also admit a natural description: for every k, Σ^{ap}_k is the relatively free monoid on Σ of the k-fold semidirect product of the pseudovariety of semilattices with itself; see [84, Rem. 3.6] for more details.

Saturated pseudofinite words. The notion of *saturated model* is fundamental in model theory. Our idea here is to employ it in the specific case of pseudofinite models. I only give the relevant definitions in that case, referring to [84, App. A] for detailed proofs that our terminology indeed corresponds to the standard one in model theory.

For any pseudofinite Σ-word U and position i in U, we call the *type*[11] of i in U the triple $t^U(i) \overset{\text{def}}{=} ([P]_\equiv, U(i), [S]_\equiv)$, where P is the prefix of U until i, and S is the suffix of U after i, both non-inclusive. More formally, P is the pseudofinite word based on the set $\{j \in |U| \mid j <^U i\}$, equipped with the relations that are the restrictions of the ones in U, and similarly for S – one obtains from the fact that U is pseudofinite that P and S are pseudofinite, too. Thus, $i \mapsto t^U(i)$ defines a function from U to $\Sigma^{\mathsf{ap}} \times \Sigma \times \Sigma^{\mathsf{ap}}$. We call the image of this function the set of *types realized* by U. The set of types *consistent with* U is the set of types that are realized by some pseudofinite word V such that $U \equiv V$. One may prove that the set of types consistent with U is in fact the topological *closure* of the set of types realized by U [84, Lem. 5.3]. We call the pseudofinite Σ-word U *weakly saturated* or 1-*saturated* if the set of types realized by U is closed, or equivalently, if every type consistent with U is realized by U. Moreover, U is ω-*saturated* if and only if every closed interval in U is weakly saturated, and *countably saturated* if it is ω-saturated and the underlying set of U is countable.

Importantly, it follows from known results in model theory that any class of pseudofinite words contains an ω-saturated word [84, Prop. A.5], which may in general need to be uncountable. We will see below that we can sometimes give a concrete construction of ω-saturated pseudofinite words. First, the following example will help clarify some of the above definitions.

Example 1.11. We consider the notion of saturation in the simple case of a one-letter alphabet, $\{a\}$. All non-finite pseudofinite words are in the same elementary equivalence class. It follows that $\{a\}^{\mathsf{ap}}$ is topologically isomorphic to the topological monoid $\mathbb{N} \cup \{\omega\}$, i.e., the one-point compactification of $\mathbb{N}$ with the usual addition, where ω is an absorbing element. We denote the unique infinite element of $\{a\}^{\mathsf{ap}}$ by a^ω. Types of pseudofinite $\{a\}$-words are of one of the following four forms:

- (a^n, a, a^m) for $n, m \in \mathbb{N}$;
- (a^n, a, a^ω) for $n \in \mathbb{N}$;
- (a^ω, a, a^m) for $m \in \mathbb{N}$;
- (a^ω, a, a^ω).

Now consider the words $W_1 \overset{\text{def}}{=} \mathbb{N} + \mathbb{N}^{\text{op}}$, $W_2 \overset{\text{def}}{=} \mathbb{N} + \mathbb{Z} + \mathbb{N}^{\text{op}}$, and $W_3 \overset{\text{def}}{=} \mathbb{N} + \mathbb{Q} \times \mathbb{Z} + \mathbb{N}^{\text{op}}$, where $+$ is the order sum, $\mathbb{Q} \times \mathbb{Z} = \sum_{q \in \mathbb{Q}} \mathbb{Z}$ carries the lexicographic order, and the predicate W_a holds in all

[11] A more precise name for such a triple would be a '1-parameter' type, but we suppress the '1-parameter' since we have no need for types with respect to more than one parameter.

positions. The word W_1 is not weakly saturated, because the elementarily equivalent word W_2 realizes the type (a^ω, a, a^ω), which is not realized in W_1. The word W_2 is weakly saturated, because it realizes all the types. However, W_2 is not ω-saturated, because the prefix to the left of i, where i is any position in the summand $\mathbb{Z}$, is isomorphic to W_1, and not weakly saturated. Notice that any closed interval in the word in W_3 is either finite or isomorphic to W_3, using the well-known fact that any open interval in the order $\mathbb{Q}$ is isomorphic to $\mathbb{Q}$ (cf. e.g., [97, p. 100]). Since finite words and W_3 are weakly saturated, the word W_3 is in fact ω-saturated, and thus countably saturated.

Our main technical result about ω-saturation is that it is stable under substitution:

Theorem 1.12. *If V is an ω-saturated Δ-word and $(U_b)_{b\in\Delta}$ is a Δ-indexed collection of Σ-words, each of which is ω-saturated, then $V[b/U_b]$ is ω-saturated.*

Applications to aperiodic ω-terms. The remainder of our work in [84, Sec. 5–8] contains a number of applications of the above theory, of which I only highlight a few here. We begin with the decidability of the word problem for aperiodic ω-terms.[12] Let Σ be a finite alphabet. An *ω-term* over Σ is a term built up from finite words by recursively applying concatenation and ω-power. If M is a profinite monoid containing the alphabet Σ, then any ω-term t has a natural interpretation t_M in M. The word problem for aperiodic ω-terms asks, given two ω-terms s and t, to decide whether or not $s_{\Sigma^{\mathsf{ap}}} = t_{\Sigma^{\mathsf{ap}}}$.

In order to solve this problem, following [100], we will now define, for any ω-term t, a particular Σ-word U_t in the class $t_{\Sigma^{\mathsf{ap}}}$. Let ρ denote the linear order $\mathbb{N} + \mathbb{Q} \times \mathbb{Z} + \mathbb{N}^{\mathrm{op}}$, which is countably saturated (Example 1.11). We recursively define:

- If t is a term representing a finite word, let U_t be that finite word.
- If $t = t_1 \cdot t_2$, let U_t be the Σ-word $U_{t_1} \cdot U_{t_2}$.
- If $t = s^\omega$, let U_t be the Σ-word $(U_s)^\rho$.

A simple induction, using Theorem 1.12, now shows that U_t is countably saturated for every term t. Since countably saturated models are unique up to isomorphism, it follows that $s_{\Sigma^{\mathsf{ap}}} = t_{\Sigma^{\mathsf{ap}}}$ if, and only if, U_s and U_t are isomorphic. Thus, in order to decide the word problem for aperiodic ω-terms, one can now proceed as in [100] and use a decidability procedure for isomorphism of regular words (cf. [24] or [114]).[13]

A further result on ω-terms, originally due to [4] and reproved and generalized with our methods in [84], is the following. By a *factor* of an element x in a monoid M, we mean an element $y \in M$ for which there exist α, β in M such that $x = \alpha y \beta$. In the monoid literature, y is also said to be *$\mathcal{J}$-above* x, denoted $y \geq_{\mathcal{J}} x$. Our use of saturated models allows us to give a fine combinatorial analysis of the set of factors of a given element x of Σ^{ap}. In particular, this allows us to deduce the fact that, for any ω-term t, the set of factors of the corresponding element $t_{\Sigma^{\mathsf{ap}}}$ is a well-quasi-order in the factor ordering

[12] The original proof of decidability was due to McCammond [124], and relied on the word problem for free Burnside semigroups of sufficiently large exponent [125].

[13] The correctness proof for this algorithm that was given in [100] relies on a non-trivial part of the original work by McCammond: the proof of [100, Proposition 5.2] goes through the non-trivial direction of McCammond's normal forms [124] (see also [5]). Our identification of the word U_t associated with an ω-term t as a countably saturated models allows us to avoid this.

$\geq_{\mathcal{J}}$, i.e., it does not contain infinite antichains nor infinite descending chains. This was originally proved as [4, Thm. 7.3], see [84, Cor. 8.9] for our generalization.

1.2 Covering, separation and pointlike sets

As explained in the introduction to this chapter, the theory of finite monoids is intimately related with questions of decidability for regular sets of finite words. There, we formulated two decidability problems for a class $\mathcal{C}$ of regular languages, namely, *membership* and *separation*. We now formulate a third algorithmic problem on classes of languages, called *covering*, which has both of the other problems as special cases, as we will see shortly. In the case where $\mathcal{C}$ is the class of languages recognizable by a pseudovariety $\mathbf{V}$, which will always be the case for us here, the problem is computationally equivalent to an older problem from finite semigroup theory, namely, that of computing the $\mathbf{V}$*-pointlike sets*. We now define these notions and explain why they are two sides of the same coin.

Refinements and the covering problem. In what follows, $\mathbf{V}$ is an arbitrary pseudovariety of semigroups[14], and a subset $L \subseteq \Sigma^+$ is $\mathbf{V}$*-recognizable* if it can be recognized by some semigroup in $\mathbf{V}$. Below, we will use without further mention the fundamental fact that complements and finite unions of $\mathbf{V}$-recognizable languages are again $\mathbf{V}$-recognizable.

Definition 1.13. If $\vec{L} = (L_1, \dots, L_n)$ is a finite sequence of regular languages over an alphabet Σ, we say that a sequence $\vec{K} = (K_1, \dots, K_n)$ is a $\mathbf{V}$*-refinement* of $\vec{L}$ if $\vec{K}$ is a partition of Σ^+, each K_i is $\mathbf{V}$-recognizable, and $K_i \subseteq L_i$ for each $1 \leq i \leq n$.

The *covering problem*[15] for the pseudovariety $\mathbf{V}$ is the following computational problem: Given as input a finite sequence $\vec{L}$ of regular languages, output a $\mathbf{V}$-refinement $\vec{K}$ of $\vec{L}$, or output 'impossible' if none such exists.

Proposition 1.14. *For any pseudovariety* $\mathbf{V}$*, if the covering problem is decidable, then the separation problem is decidable, and if the separation problem is decidable, then the membership problem is decidable.*

Proof. To see that the separation problem reduces to the covering problem, let L_1, L_2 be regular languages, and consider the instance of the covering problem for the input sequence $(L_1^{\mathrm{c}}, L_2^{\mathrm{c}})$. We claim that a $\mathbf{V}$-refinement (K_1, K_2) exists for $(L_1^{\mathrm{c}}, L_2^{\mathrm{c}})$ if, and only if, there exists a $\mathbf{V}$-separator K for L_1, L_2, i.e., a $\mathbf{V}$-recognizable set K such that $L_1 \subseteq K$ and $L_2 \cap K = \emptyset$. Indeed, if a $\mathbf{V}$-separator K for L_1, L_2 exists, then (K^{c}, K) is a $\mathbf{V}$-refinement of $(L_1^{\mathrm{c}}, L_2^{\mathrm{c}})$. Conversely, if (K_1, K_2) is a $\mathbf{V}$-refinement of $(L_1^{\mathrm{c}}, L_2^{\mathrm{c}})$, then $L_1 \subseteq K_1^{\mathrm{c}}$ and $L_2 \cap K_1^{\mathrm{c}} \subseteq K_2^{\mathrm{c}} \cap K_1^{\mathrm{c}} = \emptyset$, using that (K_1, K_2) covers Σ^+. The set $K \overset{\text{def}}{=} K_1^{\mathrm{c}}$ is thus a $\mathbf{V}$-separator for L_1 and L_2. In turn, the membership problem reduces to the separation problem, since a language L belongs to $\mathbf{V}$ if, and only if, L is $\mathbf{V}$-separable from its complement L^{c}. □

[14] For the statements and results in this section, it is more convenient to work with semigroups than with monoids. The definitions of 'pseudovariety', 'recognizable', and '(free) profinite' are the same as for monoids, replacing 'monoid' with 'semigroup' everywhere. The free semigroup on Σ is denoted Σ^+, and can be realized as the set of finite *non-empty* words over Σ.

[15] The covering problem as stated here first appeared in print under the name 'cover-computability' in [6]. As we will see below, it is equivalent to computability of $\mathbf{V}$-pointlike sets, which originate with Rhodes, with the first decidability result, in the case where $\mathbf{V}$ is the pseudovariety of finite aperiodic monoids, due to Henckell [93]. The covering problem has recently been revisited and generalized to the context of classes of languages not necessarily closed under complement, with applications to the quantifier alternation hierarchy, see e.g. [96, 142]. Further see [162] for a comprehensive historical survey, including a careful translation of existing results between the semigroup- and language-theoretic approaches.

Pointlike sets. We now give an equivalent, and historically earlier, formulation of the $\mathbf{V}$-covering problem in terms of pointlike sets. For this, recall first that, whenever S is a semigroup, the power set $\mathcal{P}S$ also carries a semigroup structure, defined by $UV \overset{\text{def}}{=} \{uv \mid u \in U, v \in V\}$. A *relational morphism* from a semigroup S to T is a function $\phi \colon S \to \mathcal{P}T$ such that, for any $s \in S$, $\phi(s) \neq \emptyset$, and, for any $s, s' \in S$, $\phi(s)\phi(s') \subseteq \phi(ss')$. Note that a function $\phi \colon S \to \mathcal{P}T$ may be equivalently described by $R_\phi \overset{\text{def}}{=} \{(s,t) \in S \times T \mid t \in \phi(s)\}$, and that ϕ is a relational morphism if, and only if, R_ϕ is a subsemigroup of the product $S \times T$ such that, for every $s \in S$, there exists $t \in T$ such that $(s,t) \in R_\phi$.

Definition 1.15. A subset X of a finite semigroup S is $\mathbf{V}$-*pointlike* if, for every finite semigroup $V \in \mathbf{V}$ and every relational morphism ϕ from S to V, the set $\bigcap_{x \in X} \phi(x)$ is non-empty. The *pointlike problem* for the pseudovariety $\mathbf{V}$ is the computational problem that asks to decide, given as input a finite semigroup S and a subset X of S, whether or not X is $\mathbf{V}$-pointlike, and, in the negative case, to output a finite semigroup V in $\mathbf{V}$ and a relational morphism $\phi \colon S \to \mathcal{P}V$ such that $\bigcap_{x \in X} \phi(x) = \emptyset$.

Example 1.16. To illustrate the general definition in a case that will concern us below, consider the pseudovariety $\mathbf{A}$ of aperiodic semigroups. We show that, if G is a subgroup of any finite semigroup S, then G is $\mathbf{A}$-pointlike. Indeed, let $\phi \colon S \to \mathcal{P}A$ be a relational morphism to an aperiodic semigroup, and consider $R_\phi \subseteq S \times A$. The projection π_1 of the semigroup R_ϕ onto the first coordinate is surjective. We can therefore (see, e.g., [148, Prop. 4.1.44]) pick a subgroup H of R_ϕ such that still $\pi_1[H] = G$. The projection $\pi_2[H]$ of this subgroup H onto the second coordinate is a subgroup of A, and thus it must be trivial since A is aperiodic. For the unique element e in $\pi_2[H]$, it follows that $e \in \bigcap_{g \in G} \phi(g)$.

Remark 1.17. To make a link with the work described in Section 1.1, I recall a remark from J. Rhodes (see [162, p. 28]) which shows that pointlike sets are also natural from a profinite perspective. Just as we defined the free proaperiodic monoid in Section 1.1, one may define, for any pseudovariety $\mathbf{V}$, the relatively free pro-$\mathbf{V}$ semigroup over an alphabet Σ, which we will denote $\Sigma^{\mathbf{V}}$. By general principles, there is a continuous homomorphism $\pi_{\mathbf{V}}$ from the free profinite semigroup over Σ, $\Sigma^{+\mathsf{pro}}$, to $\Sigma^{\mathbf{V}}$. Now, if S is any finite semigroup, and $w \in S^{+\mathsf{pro}}$ is a non-empty profinite word over S, let us denote by $[w]_S$ its image under the unique continuous homomorphism from $S^{+\mathsf{pro}}$ to S that extends the identity function $S \to S$. Using this notation, we obtain a 'canonical' relational morphism $\chi_{\mathbf{V}}$ from S to $S^{\mathbf{V}}$, defined, for $s \in S$, by:

$$\chi_{\mathbf{V}}(s) \overset{\text{def}}{=} \{\pi_{\mathbf{V}}(w) \mid w \in S^{+\mathsf{pro}}, [w]_S = s\}.$$

This relational morphism is canonical in the sense that a subset X of S is $\mathbf{V}$-pointlike if, and only if, $\bigcap_{x \in X} \chi_{\mathbf{V}}(x)$ is non-empty; a proof is given in, e.g. [136, Thm. 3.3]. While this gives a natural conceptual view on pointlikes using profinite semigroups, it can not be immediately used if one's goal is to compute pointlikes, since the semigroup $S^{\mathbf{V}}$ is typically infinite and not easy to understand. Nonetheless, it has been successfully combined with algorithmic arguments to compute pointlikes for certain pseudovarieties, see e.g., [9, 161] and the survey [162]. However, it is an open problem to give a 'profinite' proof of decidability of the pointlike problem for $\mathbf{A}$.

The fact that the covering and pointlike problems are equivalent is due to [6, Sec. 3]. The problems are moreover equivalent to a third reformulation, given in [142], and also called covering problem there. While this three-way equivalence seems to be well-known, I was not able to locate a direct explicit proof in the literature, so I give one here in Proposition 1.18. The moral of this proposition is that pointlike sets form *obstructions* to coverability. In other words, a pointlike set provides a 'witness' that allows a negative answer to the **V**-covering problem. We will also deduce from this, in Corollary 1.19, that the covering and pointlike problems are algorithmically reducible to each other.[16]

Proposition 1.18. *Let S be a finite semigroup, $f : \Sigma^+ \twoheadrightarrow S$ a surjective homomorphism, $F_1, \dots, F_n$ a finite sequence of subsets of S, and, for each $1 \le i \le n$, write $L_i \overset{\text{def}}{=} f^{-1}(F_i)$. The following are equivalent:*

1. *For any $\vec{x} = (x_1, \dots, x_n) \in F_1 \times \cdots \times F_n$, the set $\{x_1, \dots, x_n\}$ is not* **V***-pointlike;*

2. *There exists a* **V***-refinement of the sequence $(L_1^{\mathrm{c}}, \dots, L_n^{\mathrm{c}})$;*

3. *There exists a finite set $\mathcal{K}$ of* **V***-recognizable languages such that $\bigcup \mathcal{K} = \Sigma^+$ and, for every $K \in \mathcal{K}$, there exists $1 \le i \le n$ such that $K \cap L_i = \emptyset$.*[17]

Proof. (1) ⇒ (2). Write $F \overset{\text{def}}{=} F_1 \times \cdots \times F_n$. Using the assumption, for each $\vec{x} \in F$, choose a finite semigroup $V_{\vec{x}}$ in **V** and a relational morphism $\phi_{\vec{x}} : S \to V_{\vec{x}}$ such that $\bigcap_{i=1}^n \phi_{\vec{x}}(x_i) = \emptyset$. Consider the finite product $V \overset{\text{def}}{=} \prod_{\vec{x} \in F} V_{\vec{x}}$, which is again in **V**, and define the function $\phi : S \to \mathcal{P}(V)$ by

$$\phi(t) \overset{\text{def}}{=} \{(v_{\vec{x}})_{\vec{x} \in F} \mid v_{\vec{x}} \in \phi_{\vec{x}}(t) \text{ for every } \vec{x} \in F\}.$$

One readily checks that ϕ is again a relational morphism from S to V. In particular, for each $a \in \Sigma$, we can pick some $g(a) \in \phi(f(a))$. Let $g : \Sigma^+ \to V$ be the unique extension of this assignment to a homomorphism. We note that, for every $w \in \Sigma^+$, we have $g(w) \in \phi(f(w))$, by induction on the length of w: The base case is by construction, and for the inductive step, if $w = w'a$, then

$$g(w'a) = g(w')g(a) \in \phi(f(w'))\phi(f(a)) \subseteq \phi(f(w')f(a)) = \phi(f(w'a)) \,,$$

where we have used that ϕ is a relational morphism and f is a homomorphism.

We now show that, for every $v \in V$, there exists $1 \le i \le n$ such that $g^{-1}(v) \cap L_i = \emptyset$. Towards a contradiction, suppose that no such i exists, and pick, for each $1 \le i \le n$, a word $w_i \in L_i$ such that $g(w_i) = v$. Then the sequence $\vec{x} \overset{\text{def}}{=} (f(w_1), \dots, f(w_n))$ is in F, and, for each $1 \le i \le n$, we have $v = g(w_i) \in \phi(f(w_i))$. By definition of ϕ, we obtain $v_{\vec{x}} \in \phi_{\vec{x}}(f(w_i))$ for every $1 \le i \le n$, which contradicts the choice of $\phi_{\vec{x}}$. Thus, choose a function $i : V \to \{1, \dots, n\}$ such that $g^{-1}(v) \cap L_{i(v)} = \emptyset$ for every $v \in V$. We then obtain a **V**-refinement of $(L_1^{\mathrm{c}}, \dots, L_n^{\mathrm{c}})$ by putting, for each $1 \le i \le n$, $K_i \overset{\text{def}}{=} \bigcup \{g^{-1}(v) \mid i(v) = i\}$.

(2) ⇒ (3). Immediate from the definition of **V**-refinement, putting $\mathcal{K} \overset{\text{def}}{=} \{K_1, \dots, K_n\}$.

(3) ⇒ (1). Pick a finite set $\mathcal{K}$ of **V**-recognizable languages as in (3). Using that **V** is closed under finite products, we can pick a semigroup $V \in$ **V** and a homomorphism $g : \Sigma^+ \to V$ such that g recognizes each of the languages in $\mathcal{K}$. Define a function ϕ from S to $\mathcal{P}(V)$, for each $s \in S$, by

$$\phi(s) \overset{\text{def}}{=} g[f^{-1}(s)] = \{g(w) \mid w \in \Sigma^+ \text{ and } f(w) = s\} \,.$$

[16] The equivalence of (1) and (2) is already contained in [6, Sec. 3], but later literature does not explicitly link it to the formulation (3). The closest I could find was [96], which shows the equivalence between a more general covering problem considered in [142] and a newly introduced notion of 'cone-like set', which generalizes pointlike sets to ordered monoids. Note that the proof I give here in fact does not require that **V** is a pseudovariety, but only that it is a class of finite monoids closed under finite products.

[17] For readers familiar with the formulation of the covering problem in [142, Sec. 3.2], note that this property precisely says, in the terminology used there, that the pair $(A^+, \vec{L})$ is $\mathcal{V}$-coverable, for $\mathcal{V}$ the class of **V**-recognizable languages over Σ.

Note, using that f is surjective, that ϕ is a relational morphism. Now let $\vec{x} = (x_1, \dots, x_n) \in F_1 \times \cdots \times F_n$ be arbitrary. We claim that $\bigcap_{i=1}^n \phi(x_i) = \emptyset$. Indeed, towards a contradiction, suppose that there would exist $v \in \bigcap_{i=1}^n \phi(x_i)$. Then, for each $1 \le i \le n$, pick w_i such that $f(w_i) = x_i$ and $g(w_i) = v$. Since $\mathcal{K}$ covers Σ^+, pick $K \in \mathcal{K}$ such that $w_1 \in K$. By assumption, pick $1 \le i \le n$ such that $K \cap L_i = \emptyset$. Since g recognizes K, and $g(w_i) = v = g(w_1)$, we obtain $w_i \in K$. But also, since $f(w_i) = x_i \in F_i$, we have $w_i \in L_i$, contradicting that $K \cap L_i = \emptyset$. □

Corollary 1.19. *For any pseudovariety* $\mathbf{V}$*, the* $\mathbf{V}$*-covering problem is decidable if, and only if, the* $\mathbf{V}$*-pointlike problem is decidable.*

Proof. To see that the covering problem reduces to the pointlike problem, suppose given an algorithm for the pointlike problem. We describe an algorithm for the covering problem based on this. Let $\vec{L}$ be a finite sequence of regular languages. It is classical to construct a finite semigroup S and a surjective homomorphism $f \colon \Sigma^+ \twoheadrightarrow S$ that recognizes each language in the sequence. One may then put $F_i \stackrel{\text{def}}{=} f[L_i]$ for each i, to be in the situation of Proposition 1.18. For any sequence $\vec{x} = (x_1, \dots, x_n) \in F_1 \times \cdots \times F_n$, use the assumed algorithm for $\mathbf{V}$-pointlikes to check whether the set $\{s_i \mid 1 \le i \le n\}$ is $\mathbf{V}$-pointlike; if this is ever the case, return 'impossible', which is correct by the implication (2) ⇒ (1) in Proposition 1.18. Otherwise, the assumed algorithm for $\mathbf{V}$-pointlikes gives, for every $\vec{x} \in F_1 \times \cdots \times F_n$, a finite semigroup $V_{\vec{x}}$ in $\mathbf{V}$ and a relational morphism $\phi_{\vec{x}} \colon S \to V_{\vec{x}}$ such that $\bigcap_{i=1}^n \phi_{\vec{x}}(x_i) = \emptyset$. The proof of the implication (1) ⇒ (2) in Proposition 1.18 shows how to construct from these data a $\mathbf{V}$-refinement of the sequence $(L_1^c, \dots, L_n^c)$, where, for computing the function i at the end of that proof, we call an algorithm for checking disjointness of regular languages.

Conversely, suppose given an algorithm for the $\mathbf{V}$-refinement problem. Let S be a finite semigroup and let $x_1, \dots, x_n$ be n distinct elements of S. Consider the homomorphism $f \colon S^+ \to S$ that extends the identity function $S \to S$, and, for each $1 \le i \le n$, let $L_i \stackrel{\text{def}}{=} f^{-1}(\{x_i\})$. Proposition 1.18 implies that $\{x_1, \dots, x_n\}$ is $\mathbf{V}$-pointlike if, and only if, the sequence $\vec{L}' \stackrel{\text{def}}{=} (L_1^c, \dots, L_n^c)$ does not have a $\mathbf{V}$-refinement, and the proofs of (2) ⇒ (3) ⇒ (1) in Proposition 1.18 show how to construct, out of a $\mathbf{V}$-refinement for $\vec{L}'$, a relational morphism witnessing that $\{x_1, \dots, x_n\}$ is not $\mathbf{V}$-pointlike. □

We note that one may adapt Corollary 1.19 to prove that the $\mathbf{V}$-separation problem is decidable if, and only if, one can decide the $\mathbf{V}$-pointlike problem for pairs, i.e., subsets of size 2. It is unknown whether there exists a pseudovariety $\mathbf{V}$ such that the $\mathbf{V}$-pointlike problem is undecidable, but the $\mathbf{V}$-pointlike problem is decidable for pairs.

Krohn-Rhodes and aperiodic pointlikes via merge decomposition. We will now look at the pointlike problem in the specific pseudovariety of aperiodic semigroups. Henckell [93] showed that the pointlike problem is decidable for this pseudovariety. In light of Theorem 1.5 and Corollary 1.19, Henckell's theorem immediately implies that the covering problem for first-order definable languages is decidable.

The original motivation for Henckell's theorem, and more generally for much of the development of the theory pointlike sets, was the problem of computing *complexity* of finite semigroups. To explain what this problem is about, recall first that the *Krohn-Rhodes theorem* [110] shows that any finite semigroup can be decomposed into 'prime factors', which are aperiodic semigroups or groups. The decomposition in this theorem is by means of *wreath product*, which can be thought of as the semigroup analogue of a sequential composition of automata or transducers. The *complexity* of a finite semigroup S is defined to be the least number of groups that is needed in a wreath product decomposition of S. The problem that asks, given a finite semigroup, to compute its complexity, has been open for almost sixty years.[18] The major difficulty in this problem is to compute, for a finite semigroup S, a mathematical object that shows that a low-complexity decomposition of S is *not* possible; such a result is called a *lower bound*. The computation of aperiodic-pointlike sets due to Henckell [93] can be used to provide such a lower bound for level 1 in the semigroup complexity hierarchy [119, Sec. 4].

In [85], we gave a short proof of both Henckell's theorem and the two-sided Krohn-Rhodes theorem, using a new construction that we called the *merge decomposition*, and was inspired by the language-theoretic work in [141].[19] I will now give the key definitions and statements of our work in [85].

When performing an inductive argument on finite words over an alphabet Σ in order to prove a property of a homomorphism f from Σ^+ to a finite semigroup, one often encounters the following situation: The alphabet Σ decomposes into two disjoint non-empty subalphabets Σ_1 and Σ_2, in such a way that the restrictions of f to the free semigroups over Σ_i are already covered by induction. For the inductive step, any word w in Σ^+ can then be uniquely decomposed into maximal blocks of letters belonging only to Σ_1 or only to Σ_2. For instance, suppose $\Sigma_1 = \{a, b\}$ and $\Sigma_2 = \{x, y\}$, and consider the word $w = xxabbxbyxxbay$. The unique decomposition of this word into the subalphabets is the sequence $(xx, abb, x, b, yxx, ba, y)$. The idea of the *merge decomposition* is that the value of f on w can be reconstructed from the values of f on each individual block, together with some information on how blocks from Σ_1 and Σ_2 should be composed.

More formally, let us fix a finite alphabet Σ and two disjoint non-empty subalphabets Σ_1, Σ_2 such that $\Sigma = \Sigma_1 \uplus \Sigma_2$, and let T_1, T_2 be finite semigroups with $f_i : \Sigma_i^+ \to T_i$ a homomorphism for $i = 1, 2$. Further, let $g : (T_1 \times T_2)^+ \to T_0$ be a homomorphism to a finite semigroup T_0. We would like to use these data to assign, to every word in Σ^+, a value in a *merge semigroup*, T_M, that we shall define below.

First, for any pair of words $w_1 \in \Sigma_1^+$, $w_2 \in \Sigma_2^+$, let us write $p(w_1 w_2) \overset{\text{def}}{=} (f_1(w_1), f_2(w_2))$, a pair in $T_1 \times T_2$. Since Σ_1 and Σ_2 are disjoint, whenever $w \in (\Sigma_1^+ \Sigma_2^+)^+$, this allows us to uniquely define a sequence $\bar{p}(w) \in (T_1 \times T_2)^+$, by applying p to each $\Sigma_1^+ \Sigma_2^+$-block of w. We obtain a homomorphism $f_0 : (\Sigma_1^+ \Sigma_2^+)^+ \to T_0$, defined by $f_0(w) \overset{\text{def}}{=} g(\bar{p}(w))$ for every w. In order to extend the definition of f_0 to give a function defined on all of Σ^+, we need to take into account the possibility that a word $w \in \Sigma$ may start with a letter from Σ_2 and may end with a letter from Σ_1. We first extend each of the semigroups T_0, T_1, T_2 to a monoid, by adding a (new) identity element, I_i, thus obtaining monoids T_k^I for $k = 0, 1, 2$. We also still denote by f_k the unique extension of f_k to a monoid homomorphism $(\Sigma_k)^* \to T_k^I$, i.e., $f_k(\epsilon) \overset{\text{def}}{=} I_k$, for $k = 0, 1, 2$. Now, when $w \in \Sigma^+$, we can uniquely write $w = v_2 u v_1$, with $v_2 \in \Sigma_2^*$, $u \in (\Sigma_1^+ \Sigma_2^+)^*$, and $v_1 \in \Sigma_1^*$, and define $\phi(w) := (f_2(v_2), f_0(u), f_1(v_1))$.

While this fulfills our desire of extending the definition of f_0 to the entire domain Σ^+, the function $\phi : \Sigma^+ \to T_2^I \times T_0^I \times T_1^I$ is not a homomorphism, as it is not even clear how the multiplication on $T_2^I \times T_0^I \times T_1^I$ should be defined. The *merge decomposition*, that we introduce now, will refine the function ϕ to a homomorphism f, whose codomain semigroup moreover lies in a well-controlled pseudovariety,

[18] In June 2024, Margolis, Rhodes, and Schilling published a preprint proving decidability of this problem [118].

[19] The paper [141] also proved separation for first-order logic on infinite words indexed by the ordinal ω. In 2022, together with Thomas Colcombet and with Rémi Morvan, who at the time was a student in the Parisian Research Master in Computer Science (MPRI), we extended this result to obtain decidability of the covering problem for first-order logic over any *countable ordinals* [37].

parametric in the semigroups T_0, T_1, T_2 that we started from.

Our construction of the codomain semigroup T_M of this homomorphism f will be an instance of the *triple product* [48, Sec V.9], of which we recall the definition first. Let $(S, +)$ be a (not necessarily commutative) semigroup equipped with two actions on it, a left action of a semigroup $(S_L, \cdot)$ and a right action of a semigroup $(S_R, \cdot)$, and suppose that the two actions *commute*, i.e., $(s_L \cdot s) \cdot s_R = s_L \cdot (s \cdot s_R)$ for any $s_L \in S_L, s \in S, s_R \in S_R$. The *triple product* (S_R, S, S_L) is the semigroup of triples (s_R, s, s_L), with multiplication defined by

$$(s_R, s, s_L) \cdot (s'_R, s', s'_L) := (s_R s'_R, s s'_R + s_L s', s_L s'_L) .$$

Note that the multiplication can be viewed as a matrix multiplication, if we represent an element (s_R, s, s_L) by the lower triangular matrix $\left(\begin{smallmatrix} s_R & 0 \\ s & s_L \end{smallmatrix}\right)$.

We now apply the triple product construction in order to define the codomain semigroup, T_M, in the situation described above. For the semigroup S, we take $S \overset{\text{def}}{=} (T_0^I)^{T_1^I \times T_2^I}$, equipped with the pointwise product of T_0^I, which we denote by $+$. For the semigroup S_L, we take the submonoid of endofunctions $\mathsf{End}(T_1^I)$ consisting of the right multiplication functions $t \mapsto tx$, for each $x \in T_1$, and the constant functions $c_x : t \mapsto x$, for each $x \in T_1^I$, and we define S_R dually to consist of the subsemigroup of endofunctions $\mathsf{End}(T_2^I)$ consisting of the left multiplications and constant functions. Now, define commuting actions of S_L on the left of S and S_R on the right of S by setting, for any $x \in S_L$, $y \in S_R$, and $(t_1, t_2) \in T_1^I \times T_2^I$:

$$xsy(t_1, t_2) \overset{\text{def}}{=} s(x(t_1), y(t_2)) .$$

Finally, let $T_M \overset{\text{def}}{=} (S_R, S, S_L)$ be the triple product; we call T_M the *merge semigroup* associated to f_1, f_2, g. The homomorphism $f : \Sigma^+ \to T_M$ is defined, on letters $a \in \Sigma$, by a case distinction: If $a \in \Sigma_1$ then $f(a) \overset{\text{def}}{=} (c_{I_2}, s_a, f_1(a))$, where

$$s_a(t_1, t_2) \overset{\text{def}}{=} \begin{cases} I_0 & \text{if } t_2 = I_2, \\ g(t_1 f_1(a_1), t_2)) & \text{otherwise.} \end{cases}$$

If $a \in \Sigma_2$, then $f(a) \overset{\text{def}}{=} (f_2(a), 0, c_{I_1})$, where 0 denotes the identity of S.

Crucially, we can show that the homomorphism f *refines* the function ϕ, in the sense that $f(w) = f(w')$ implies $\phi(w) = \phi(w')$. Indeed, the function $\alpha : T_M \to T_2^I \times T_0^I \times T_1^I$ which sends (y, s, x) to $(y(I_2), s(I_1, I_2), x(I_1))$ is such that, for any word $w \in \Sigma^+$, $\alpha(f(w)) = \phi(w)$ [85, Prop. 2.2]. This implies in particular that the complexity of T_M is bounded above by the sum of the complexity of T_0 and the maximum complexity of T_1 and T_2, where we use a two-sided variant of the semigroup complexity discussed above.

We applied the merge decomposition to prove the two-sided Krohn-Rhodes theorem and Henckell's theorem [85, Sec. 3 and 4]. Here, I will only briefly comment on how we use the merge decomposition to prove the latter result. Let S be a finite semigroup, and denote by $\mathcal{P}_{\mathbf{A}}(S)$ the subset of $\mathcal{P}(S)$ consisting of the aperiodic-pointlike sets. The crucial idea of Henckell is that $\mathcal{P}_{\mathbf{A}}(S)$ is itself a subsemigroup of the power semigroup $\mathcal{P}(S)$, and that this semigroup structure can be used to compute it. Indeed, for a general pseudovariety $\mathbf{V}$, denoting by $\mathcal{P}_{\mathbf{V}}(S)$ the collection of $\mathbf{V}$-pointlike subsets of S, one may prove the following four properties:

1. (Singletons) For any $s \in S$, the singleton $\{s\}$ is in $\mathcal{P}_{\mathbf{V}}(S)$.
2. (Multiplication) For any $U, V \in \mathcal{P}_{\mathbf{V}}(S)$, the product set UV is also in $\mathcal{P}_{\mathbf{V}}(S)$.
3. (Downward closed) For any $U \in \mathcal{P}_{\mathbf{V}}(S)$, if $U' \subseteq U$, then $U' \in \mathcal{P}_{\mathbf{V}}(S)$.
4. (Submonad of $\mathcal{P}$) For any $\mathcal{X} \subseteq \mathcal{P}_{\mathbf{V}}(S)$, if $\mathcal{X}$ is a **V**-pointlike set of the semigroup $\mathcal{P}_{\mathbf{V}}(S)$, then $\bigcup \mathcal{X}$ is a **V**-pointlike subset of S.

The last property in the list above is not immediately useful for *computing* pointlike sets, since, to use it, one would already need to know what the pointlike subsets of $\mathcal{P}_{\mathbf{V}}(S)$ are. However, in many concrete cases, such as that of the pseudovariety **A** of finite aperiodic semigroups, one may show that certain kinds of subsets of a semigroup are always pointlike, thus providing a base case for a recursive definition. Indeed, as we saw in Example 1.16, in any finite semigroup S, a subgroup G is always **A**-pointlike. It is therefore natural to define, for any finite semigroup S, and subset $U \subseteq \mathcal{P}(S)$, the set $\mathcal{H}(U) \subseteq \mathcal{P}(S)$ to be the smallest downward closed subsemigroup of $\mathcal{P}(S)$ which contains U, and is such that, for any subgroup $\mathcal{G}$ of $\mathcal{H}(U)$, the union $\bigcup \mathcal{G}$ also lies in $\mathcal{H}(U)$. The set $\mathcal{H}(S)$ can clearly be computed by a simple saturation procedure, by adding unions of subgroups, and closing under multiplication and subset inclusion. Writing $\tilde{S} \overset{\text{def}}{=} \{\{s\} \mid s \in S\}$, the above arguments show that $\mathcal{H}(\tilde{S}) \subseteq \mathcal{P}_{\mathbf{A}}(S)$. The difficult part of Henckell's theorem is to show the other inclusion, and this is where, in [85], we use our merge decomposition. More precisely, we use it to prove the following by induction (see [85, Thm. 4.3]):

Proposition 1.20. *Let Σ be a finite alphabet, S a finite semigroup, and $f : \Sigma^+ \to \mathcal{P}(S) \setminus \{\emptyset\}$ a homomorphism. There exist a finite aperiodic semigroup T and a homomorphism $g : \Sigma^+ \to T$ such that, for every $t \in T$, the set $\bigcup f[g^{-1}(t)]$ is in $\mathcal{H}(\mathrm{im}(f))$.*

Given Proposition 1.20, we now deduce that any **A**-pointlike subset of S lies in $\mathcal{H}(\tilde{S})$. Indeed, consider the homomorphism $f : S^+ \to \mathcal{P}(S)$ which sends s to $\{s\}$, so that $\mathrm{im}(f) = \tilde{S}$, and pick T, g as in Proposition 1.20. Let the relational morphism $\phi : S \to \mathcal{P}(T)$ be defined by

$$\phi(s) \overset{\text{def}}{=} \{t \in T \mid \text{there exists } w \in S^+ \text{ such that } g(w) = t \text{ and } s \in f(w)\}\,.$$

Then, if X is any **A**-pointlike subset of S, we can pick $t \in \bigcap_{x \in X} \phi(x)$. This implies, by definition of ϕ, that X is a subset of $\bigcup f[g^{-1}(t)]$. By the choice of g and the fact that $\mathcal{H}(\mathrm{im}(f))$ is downward closed, we conclude that $X \in \mathcal{H}(\mathrm{im}(f))$, as required.

Recall from the discussion under Eq. (1.3) that every finite aperiodic monoid divides a k-fold semidirect product of semilattices. Our proof of Proposition 1.20 is constructive, in the sense that it gives a computable bound on the number k such that T divides a k-fold semidirect product of semilattices. In logical terms, this means that, given a finite sequence of regular languages, if it admits an aperiodic refinement, then we can compute an upper bound on the quantifier alternation depth that is needed to define the refining languages. A class of major open problems in finite semigroup theory, known as the decidability of the *dot-depth* and the related *Straubing-Thérien* hierarchy, ask to determine *lower* bounds on such k. More precisely, the current state-of-the-art result is that it is decidable, for a given first-order definable language L, whether or not it can be defined with at most 3 quantifier alternations [140]. That work uses a generalization of covering problems and pointlike sets, as well as profinite equations.

Pointlikes for excluded subgroups and modular quantifiers. For $\mathbf{H}$ a pseudovariety of finite *groups*, write $\overline{\mathbf{H}}$ for the pseudovariety of finite semigroups for which all subgroups are in $\mathbf{H}$. This generalizes the pseudovariety of aperiodic semigroups, since $\mathbf{A} = \overline{\mathbf{1}}$, where $\mathbf{1}$ denotes the pseudovariety containing only the trivial group. Another natural class of examples are the pseudovarieties of finite p-groups, for any prime p, which we denoted $\mathbf{G}_p$. Here, recall that a finite group is a p-group if the order of any element is a power of p, or equivalently, the cardinality of the group is a power of p. Then, the pseudovariety of semigroups $\overline{\mathbf{G}_p}$ has a natural logical interpretation, using *modular quantifiers*, which can 'modulo-count the number of satisfying assignments'. More formally, let q be a positive integer. For every $0 \leq r < q$, we add to the syntax of first-order logic a new quantifier $\exists^{(q,r)}$. The definition of the semantics is then extended with a clause stating that, for any $w \in \Sigma^+$, any formula ϕ, and any valuation v of the free variables in ϕ, $(w, v) \in [\![\exists^{(q,r)} x.\phi]\!]$ if, and only if, the number of positions p in w such that $(w, v_{x \mapsto p}) \in [\![\phi]\!]$ is congruent to r modulo q [164, Ch. VII]. Straubing [164, Thm. VII.2.1] proved, using the Krohn-Rhodes theorem, that a language L is recognizable by a semigroup in $\overline{\mathbf{G}_p}$ if, and only if, L is definable in first-order logic with modular quantifiers of modulus p. Moreover, it was proved in [95, Cor. 2.4] that the pointlike problem for $\overline{\mathbf{G}_p}$ is decidable. In light of Corollary 1.19, covering and separation are thus decidable for first-order logic with quantifiers modulo p. In [84], we extended this result to arbitrary pseudovarieties of finite groups, showing:

Theorem 1.21 ([84, Thm. 3.2])**.** *For any pseudovariety of finite groups* $\mathbf{H}$ *with decidable membership problem, the pointlike problem for* $\overline{\mathbf{H}}$ *is decidable.*

This theorem in particular yields, as an immediate corollary, the decidability of the covering problem for regular languages with respect to first-order logic enriched with *all* modular quantifiers. Indeed, again by Straubing's theorem [164, Thm. VII.2.1], languages definable in first-order logic with arbitrary modular quantifiers are exactly those recognizable by a semigroup in $\overline{\mathbf{G}_{\mathrm{sol}}}$, where $\mathbf{G}_{\mathrm{sol}}$ is the variety of *solvable* groups. Here, recall that a group is *solvable* if the so-called derived sequence $G^{(0)} \overset{\text{def}}{=} G, G^{(n+1)} \overset{\text{def}}{=} [G^{(n)}, G^{(n)}] = \{ghg^{-1}h^{-1}, g, h \in G^{(n)}\}$ terminates in the trivial group. Theorem 1.21 shows that $\overline{\mathbf{G}_{\mathrm{sol}}}$ has decidable pointlike problem, from which the claimed decidability of the covering problem then follows, again using Corollary 1.19.

The algorithm for computing pointlikes that we develop in the proof of Theorem 1.21 adapts Henckell's algorithm described above, modifying the step that uses the monadic property of $\mathcal{P}_{\overline{\mathbf{H}}}$ to take into account the specific pseudovariety $\mathbf{H}$; see [84, Def. 3.1].

1.3 Outlook on profinite monoids and pointlike sets

The results described in this chapter naturally suggest a number of further directions and questions, which I will briefly discuss here. A first natural direction to explore is how far the results in Section 1.1 might be pushed beyond the first-order setting. The beginnings of an extension to monadic second-order logic were developed in Linkhorn's PhD work [111, 112]. This work in particular views monadic second-order logic as an instance of first-order logic interpreted in the power set algebra, and relates this to Shelah's composition theorem [158]. It is proved in [112, Thm. 4.12] that the free profinite monoid over Σ can be seen as a type space of the pseudofinite monadic second-order theory of finite words. It is natural to ask whether, similar to our work in Section 1.1, this logical point of view on the free profinite monoid can have an impact on the theory of (pro)finite monoids. In particular, what is the correct interpretation of saturation in this setting? Can this be used to solve profinite word problems for pseudovarieties outside the aperiodic setting? The paper [7] contains a related but different 'concrete' approach to profinite monoids and pseudofinite words. From a model-theoretic point of view, the pseudofinite words that [7] associate to elements of the free proaperiodic monoid can be seen as the *prime* models, as we show in the unpublished draft [88]. It is interesting that the results in [7] apply more generally than in just the aperiodic setting. Can a model-theoretic analysis shed further light on the results of [7], which go beyond the aperiodic case, but do not use logic explicitly yet?

A second direction to explore is suggested by work of Marquès [121], which relates our results in Section 1.1 to categorical logic, in particular, hyperdoctrines and polyadic spaces, which play the role of, respectively, Boolean algebras and Boolean spaces in a first-order setting. In a recent joint paper with Marquès, who is currently a postdoc with me at IRIF [79], we develop a Stone duality theory for this setting, and use it to prove general completeness and interpolation theorems in a categorical way. We also identify the notion of saturation in the categorical logic setting [79, Rem. 5.8]. In future work, I would like to investigate how this categorical view can help to 'lift' our results from profinite monoids to profinite structures relevant to recognition of more general structures than words. Here, I have in mind in particular the general notion of profinite monad introduced in [25], and the specific notion of recognition for λ-terms that we developed in collaboration with Melliès and Moreau in [77], to be investigated further in Moreau's forthcoming PhD thesis, co-supervised by Melliès and me.

In the theory of pointlike sets (Section 1.2), in addition to the questions already pointed to in the main text, I belive there is also room for a more category-theoretic investigation, see for example [94]. Moreover, in our work on pointlike sets with Steinberg we were not able to fully exploit the profinite perspective, and instead had to give intricate, combinatorial arguments involving concrete finite semigroups. As mentioned in Remark 1.17, it remains a fascinating open question whether there are 'profinite' proofs of Henckell's theorem on aperiodic pointlike sets, and its generalizations. It would also be useful here to develop a profinite approach for other structures than monoids, in order to see, for example, if our results for countable ordinals in [37] could extend to a logic beyond first-order.

2 Uniform interpolation: Topology, proof theory, and compact congruences

The results presented in this paper have had a rather long gestation period. (. . .) That [the uniform interpolation] Theorem 1 is true is quite a surprise to me.

–*A. M. Pitts* [139, p. 5]

Interpolation is the problem that asks, given a logical entailment

$$A \vdash C\,,$$

to find B that only uses symbols that appear in both A and C, such that

$$A \vdash B \text{ and } B \vdash C\,.$$

This concept was introduced by Craig [39], who showed how one can always find an interpolant when A and C are formulas in first-order logic. Craig's interpolation theorem became a cornerstone result in model theory, see [54] for a survey.

The focus of this chapter is on a number of recent results around an interpolation property of *intuitionistic* logic that was first established by Pitts [139]. In order to motivate Pitts' theorem (stated as Theorem 2.1 below), we first show how to prove a strong version of Craig's theorem in a specific setting, namely, the propositional one. Let $A(\bar{x}, \bar{p})$ and $C(\bar{x}, \bar{q})$ be formulas of Boolean propositional logic $\mathbf{B}$ and suppose that $A \vdash_{\mathbf{B}} C$, that is, the formula $\neg A \vee C$ is a Boolean tautology. Now define

$$B(\bar{x}) \overset{\text{def}}{=} \bigvee \left\{ A(\bar{x}, \bar{b}) \mid \bar{b} \in \{\bot, \top\}^{\bar{p}} \right\}\ , \tag{2.1}$$

that is, B is the disjunction of all possible variants of A obtained by substituting some vector of truth values for the propositional variables $\bar{p}$. One readily observes that B is an interpolant for the entailment $A \vdash_{\mathbf{B}} C$: The fact that $A \vdash_{\mathbf{B}} B$ follows from the definition of B, and the fact that $B \vdash_{\mathbf{B}} C$ uses the assumption that $A \vdash_{\mathbf{B}} C$.

Notice that the interpolant B defined here does not depend on the precise shape of the consequent formula C, but only on the antecedent formula A and the subset $\bar{p}$ of variables 'to be eliminated' from A. In other words, B works *uniformly* as an interpolant, namely, for any consequence C' of A that does not contain the variables $\bar{p}$. Thus, B is called a *right uniform interpolant* for A with respect to $\bar{p}$. Symmetrically, the expression $\bigwedge\{C(\bar{x}, \bar{b}) \mid \bar{b} \in \{\bot, \top\}^{\bar{q}}\}$ defines a *left uniform interpolant* for C with respect to $\bar{q}$.

The purpose of this chapter is to study uniform interpolation in the context of *intuitionistic* logic. Intuitionistic propositional logic $\mathbf{I}$ is a proper subsystem of $\mathbf{B}$ that finds its origins in the 1920s in foundational work by Brouwer, subsequently formalized by Kolmogorov and Heyting; see [12] and the references therein. A relationship between **I** and the formal study of computation was suggested early on by Curry [40], and realized by Howard [98], who reinterpreted formulas of **I** as type expressions in a λ-calculus with type constructors for functions, products, sums, unit, and empty types, and showed that formal proofs of **I**-formulas correspond to programs written in this calculus. For our purposes in this chapter, it is most convenient to first define the logic **I** algebraically, through the notion of Heyting algebra, which is the appropriate intuitionistic generalization of the classical concept of Boolean algebra. In addition to the algebraic and computational views on **I** mentioned in this introduction, there exist at least two further important points of view, namely a proof-theoretic and a semantic one. We will touch on both of these later in this chapter.

The algebraic approach to logic considers formulas as elements of an abstract algebraic structure, which is typically a *bounded lattice*, that is, a tuple $(L, \vee, \wedge, \bot, \top)$ such that $\vee$ ('join') and $\wedge$ ('meet') are commutative idempotent monoid operations with neutral elements $\bot$ and $\top$, respectively, satisfying the *absorption* laws $a \vee (a \wedge b) = a = a \wedge (a \vee b)$ for all $a, b \in L$. A bounded lattice has a natural partial order defined by $a \leq b$ if, and only if, $a \wedge b = a$, or equivalently $a \vee b = b$. A *Heyting algebra* is a bounded lattice $(H, \vee, \wedge, \bot, \top)$ in which, for any $a \in A$, the 'meet with a' operation $c \mapsto a \wedge c$ has an upper adjoint, i.e., for any $b \in H$, the set $\{c \in H \mid a \wedge c \leq b\}$ has a maximum, which is denoted $a \Rightarrow b$. We also write $\neg a \overset{\text{def}}{=} a \Rightarrow \bot$. Note that Heyting algebras are always *distributive*, i.e., $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$ for every $a, b, c \in H$. A *Boolean algebra* a Heyting algebra in which $a \vee \neg a = \top$ for every $a \in A$.

By general principles of universal algebra (see, e.g., [29, §II.11]), the *free* Heyting algebra on any set of generators X, $\mathbb{H}(X)$, exists, and can be based on the set of equivalence classes of **I**-formulas with variables in X, with the various operations given by the syntax. Figure 2.1 shows a part of the diagram of the free Heyting algebra on a single generator p. We often identify elements of $\mathbb{H}(X)$ with terms. Free finitely generated Heyting algebras have a rich structure, which was deeply studied throughout the twentieth century; see, e.g. [31, Ch. 7] and the references therein for an overview.

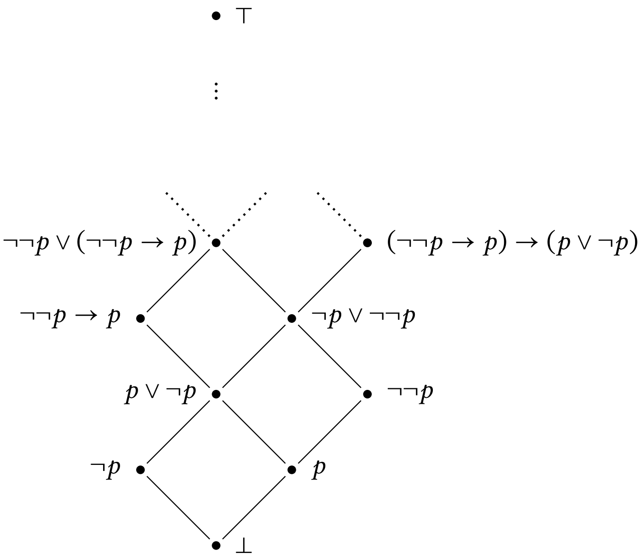


Figure 2.1: The 'Rieger-Nishimura lattice': a free Heyting algebra on one generator, p.

An *intuitionistic entailment* is defined to be a pair of propositional formulas A, C such that, in the free Heyting algebra generated by the variables appearing in A and C, we have $A \leq C$; in this case, we write $A \vdash_{\mathbf{I}} C$, and the collection $\mathbf{I}$ of intuitionistic tautologies consists of those C for which $\top \vdash_{\mathbf{I}} C$. Similarly, a classical or *Boolean entailment* $A \vdash_{\mathbf{B}} C$ means that $A \leq C$ in the free Boolean algebra.

One important difference between intuitionistic and Boolean entailment is that Boolean algebras are *locally finite*, that is, the free Boolean algebra on a finite set of generators is finite, while the free Heyting algebra even on a single generator is already infinite. The fact that free Heyting algebras are infinite means that the proof strategy for interpolation given in Eq. (2.1) above can no longer work for intuitionistic logic: If one tried to imitate that definition by substituting for the variables to be eliminated, instead of just the values $\top$ and $\bot$, all possible intuitionistic equivalence classes of formulas, then one would need to take a disjunction of an infinite set, which is no longer a formula. It is all the more surprising that the uniform interpolation property still holds for $\mathbf{I}$:

Theorem 2.1 (Pitts, [139])**.** *Any formula of intuitionistic propositional logic has both left and right uniform interpolants with respect to any propositional variables.*

To make this statement more formal, let F be any formula, and p a propositional variable. Pitts' theorem says that there exist formulas $\mathsf{E}_p(F)$ and $\mathsf{A}_p(F)$, both *p-free*, i.e., not containing p, such that

$$F \vdash_{\mathbf{I}} \mathsf{E}_p(F), \quad \mathsf{A}_p(F) \vdash_{\mathbf{I}} F,$$

and such that, for any p-free formula A,

$$\text{if } A \vdash_{\mathbf{I}} F \text{ then } A \vdash_{\mathbf{I}} \mathsf{A}_p(F),$$

and, for any p-free formula C,

$$\text{if } F \vdash_{\mathbf{I}} C \text{ then } \mathsf{E}_p(F) \vdash_{\mathbf{I}} C.$$

In other words, $\mathsf{E}_p(F)$ is a *right uniform interpolant* of F with respect to the variable p, and $\mathsf{A}_p(F)$ is a *left uniform interpolant* of F with respect to the variable p. Note that the rules imposed on the operations E_p and A_p in this definition are exactly the rules for second-order *propositional quantifiers*. From this viewpoint, the uniform interpolation theorem states that these propositional quantifiers can be encoded in the propositional logic itself, and this was one of Pitts' original motivations for Theorem 2.1. We will return to propositional quantifiers in Section 2.3.

Recall that a pair of monotone functions $f : P \leftrightarrows Q : g$ between partially ordered sets P and Q is called an *adjunction* if, for any $x \in P$, $y \in Q$, we have $f(x) \leq y$ if, and only if $x \leq g(y)$; in this case f is called the *lower* or *left* adjoint of g, and g is called the *upper* or *right* adjoint of f. In algebraic terms, Pitts' theorem has the following consequence:

Corollary 2.2. *Every homomorphism between finitely presented Heyting algebras has both a lower and an upper adjoint.*

To get an idea why Corollary 2.2 follows from Theorem 2.1, let us prove it in the special case of finitely generated free algebras. With a slight abuse of notation, we may view E_p as a function from the free Heyting algebra $\mathbb{H}(\bar{x}, p)$ to the free Heyting algebra $\mathbb{H}(\bar{x})$.[1] The defining properties of the

[1] Indeed, left and right uniform interpolants are unique up to equivalence: if both E and E' are right uniform interpolants for F with respect to p, then E' is a p-free formula for which $F \vdash E'$ holds, so that $E \vdash E'$ must hold, and symmetrically, $E' \vdash E$. The same argument applies for left uniform interpolants.

right uniform interpolant then imply that, for any $C \in \mathbb{H}(\bar{x})$, $\mathsf{E}_p(F) \leq C$ if, and only if, $F \leq i_p(C)$, where i_p denotes the *natural inclusion*, that is, the homomorphism $\mathbb{H}(\bar{x}) \hookrightarrow \mathbb{H}(\bar{x}, p)$ which is defined by sending each variable x in $\bar{x}$ to itself. In other words, E_p *is* the lower adjoint of i_p. Similarly, A_p is the upper adjoint of i_p. With a little more algebraic work, one can extend this argument to apply to any homomorphism between finitely presented Heyting algebras, see [139, p. 16], and our more general analysis of the situation in Section 2.3 below.

In the rest of this chapter, I will survey my contributions with various collaborators to the theory of uniform interpolation, both in intuitionistic logic **I**, and beyond that:

1. An open mapping theorem for Esakia spaces, leading in particular to a topological semantic proof of Pitts' theorem (Section 2.1);
2. Formalized computation of uniform interpolants, leading to usable and verified implementations of Pitts' algorithm for computing interpolants, with modal extensions (Section 2.2);
3. A universal-algebraic study of uniform interpolation, via lattices of compact congruences, with an application to model-completeness (Section 2.3).

2.1 An open mapping theorem for Esakia spaces

In this section, we develop a *topological* approach to proving Pitts' theorem (Theorem 2.1).[2] The idea of using topology for studying intuitionistic logic goes back to Stone [163]. To motivate this idea, observe first that the collection $\mathcal{O}(X)$ of open sets of any topological space X is a Heyting algebra, which is even *complete*: For any collection of open sets $\mathcal{U}$, the union of the collection, $\bigcup \mathcal{U}$, is an open set, and therefore gives the supremum of $\mathcal{U}$ in the lattice $\mathcal{O}(X)$. The infimum of $\mathcal{U}$ is calculated by taking the interior of the intersection of the collection $\mathcal{U}$. The Heyting implication in $\mathcal{O}(X)$ is given, for U and V open sets, by letting $U \Rightarrow V$ be the interior of the set of points that are either in V, or not in U. In particular, $\neg U$ is the interior of the complement of U.

A Heyting algebra is called a *spatial frame*[3] if it is isomorphic to $\mathcal{O}(X)$, for some topological space X. The fact that spatial frames are complete makes them not immediately usable for studying Heyting algebras in general: When restricting to the class of spatial frames, one loses the finitary nature inherent in Heyting algebras. In particular, *free* Heyting algebras on two or more generators are not complete [20, Thm. 4.2]. To obtain *all* Heyting algebras from a topological construction, one may use a theorem due to Stone [163], which shows that any bounded distributive lattice (and thus in particular any Heyting algebra) can be represented as the collection of open *and compact* subsets of a certain topological space. Homomorphisms of Heyting algebras then correspond to certain strongly continuous functions between the spaces in the other direction. We will here use a more modern point of view on this duality due to Priestley [143] and Esakia [52, 53], who consider subsets that are

[2] This section is based on joint work from 2016 with Luca Reggio, who was a PhD student at the time, and draws from our joint publication [83].

[3] The partial orders underlying *complete Heyting algebras* are also known as *frames* in the literature, and as such are central to the study of point-free topology and topos theory. The logic underlying frames is commonly referred to as 'geometric' logic, as opposed to the 'intuitionistic' logic considered here. While frames are exactly the same objects as complete Heyting algebras, the correspondence breaks down at the level of morphisms: A frame is an algebraic structure in the infinitary algebraic language $(\bigvee, \wedge, \top)$, while a Heyting algebra is considered in the finitary language $(\vee, \wedge, \Rightarrow, \bot, \top)$. A Heyting algebra homomorphism may fail to be a frame homomorphism, and vice versa, see, e.g., [60, Ex. 4.6.5].

clopen (i.e., closed and open) up-sets (i.e., upward closed) with respect to a partial order that is added as additional structure to the topological space. One way to view the results of Stone, Priestley, and Esakia is that they provide a *canonical* choice for an embedding of a Heyting algebra into a complete Heyting algebra of the form $\mathcal{O}$. Indeed, for H a Heyting algebra, Stone's representing space X is such that $\mathcal{O}(X)$ is the *ideal completion* of H. I will recall the basic facts of this duality theory now; I am intentionally brief here, and refer to, e.g., [60, Ch. 3, 4] for a more thorough introduction.

We will work with *partially ordered topological spaces*, that is, tuples $(X, \tau, \leq)$ such that τ is a topology on X and $\leq$ is a partial order on X. The natural mappings between such spaces are the continuous monotone functions, although we will also consider an important subclass of these below. The richness of the theory comes from the interaction between this partial order and the topology.[4] An ordered space X is *totally order-disconnected* provided that, for any $x, y \in X$ such that $x \not\leq y$, there exists a clopen, upward closed subset U of X such that $x \in U$ and $y \notin U$. Now, X is called an *Esakia space* if (i) X is compact, (ii) X is totally order-disconnected, and (iii) the downward order-closure ${\downarrow}U$ of any open subset U of X is open. A space satisfying (i) and (ii), but not necessarily (iii), is called a *Priestley space*. When X is a Priestley space, the clopen sets in particular form a basis for the topology on X, so the underlying topological space (X, τ) is a *Boolean space*, i.e., a compact zero-dimensional Hausdorff space. The partial order on a Priestley space is truly additional structure: A Boolean space admits many distinct partial orders that make it into a Priestley space. For a simple example, any finite set with the discrete topology is a Boolean space, and any partial order on it makes it a Priestley space, which, in this case, is also automatically an Esakia space. More truly topological examples of Boolean and Priestley spaces are given below, also see Example 3.4 in Chapter 3.

Esakia [53] proved that the category of Heyting algebras is dually equivalent to a category of Esakia spaces and Priestley [143] showed that the category of bounded distributive lattices is dually equivalent to a category of Priestley spaces. To introduce some notation, for any bounded distributive lattice A, we write $\mathsf{spec}\, A$ for the up to order-homeomorphism unique Priestley space such that A is isomorphic to the lattice of clopen up-sets of $\mathsf{spec}\, A$. The lattice A and the space $\mathsf{spec}\, A$ are called each other's *dual*. Moreover, the morphism part of Priestley duality says that there is a natural bijection between lattice homomorphisms $A \to B$ and continuous monotone functions $\mathsf{spec}\, B \to \mathsf{spec}\, A$. A distributive lattice A is a Heyting algebra if, and only if, its dual $\mathsf{spec}\, A$ is an Esakia space. In this case, identifying A with the clopen up-sets of $\mathsf{spec}\, A$, we have, for any $a, b \in A$, that the Heyting implication $a \Rightarrow b$ is given by

$$a \Rightarrow b = \{x \in \mathsf{spec}\, A \mid \text{ for all } y \geq x, \text{ if } y \in a \text{ then } y \in b\}\,.$$

A lattice homomorphism $h \colon A \to B$ preserves the Heyting implication $\Rightarrow$ if, and only if, the dual function $f \colon \mathsf{spec}\, B \to \mathsf{spec}\, A$ has the property that, for every subset $S \subseteq \mathsf{spec}\, A$,

$${\uparrow} f^{-1}(S) = f^{-1}({\uparrow} S)\,. \tag{2.2}$$

[4] Ordered topological spaces where first studied systematically by Nachbin [132]. The theory becomes especially nice when the topology is compact and the the partial order is closed as a subset of the square; such structures are called *compact ordered spaces*, see, e.g. [60, Sec. 2.3] for an introduction. A complementary point of view on compact ordered spaces is provided by *stably compact spaces*, in which the topology is refined to only contain open up-sets, and the partial order may be forgotten, in exchange for working with a non-Hausdorff topology. We do not pursue this point of view any further here, but refer to, e.g., [44, 89] for much more information on non-Hausdorff spaces and their applications in domain theory and ring theory.

A function $f : X \to Y$ between posets is called *bounded* if it satisfies Eq. (2.2) for any subset of the domain X. Note that f is bounded if, and only if, f is monotone and for any points $x \in X$ and $y \in Y$, if $f(x) \leq y$, then there exists $x' \geq x$ such that $f(x') = y$. The semantically inclined reader may recognize in this definition of boundedness the 'back-and-forth' morphisms of intuitionistic Kripke frames. Indeed, this is no coincidence: from the point of view of duality, the canonical semantics for **I** *is* the embedding of a free Heyting algebra into the upward closed sets of its dual Esakia space, and the natural morphisms in this setting are the morphisms dual to Heyting algebra homomorphisms.[5]

We write $\mathbb{E}(\bar{x})$ for $\mathsf{spec}\,\mathbb{H}(\bar{x})$, the Esakia space dual to the free Heyting algebra $\mathbb{H}(\bar{x})$ over set of generators $\bar{x}$, also known as the *canonical model* for intuitionistic propositional logic. Concretely, one may take as the points of $\mathbb{E}(\bar{x})$ the prime theories of intuitionistic propositional logic **I** in variables $\bar{x}$, where a *theory* is a set of formulas closed under entailment and finite conjunction, and a theory T is *prime* if $\bot \notin T$ and $A \vee B \in T$ implies $A \in T$ or $B \in T$. The partial order is given by inclusion of prime theories. The topology is generated by the collection of subsets $\{\widehat{A}, \widehat{A}^{\mathrm{c}} : A \in \mathbb{H}(\bar{x})\}$, where, for any $A \in \mathbb{H}(\bar{x})$, $\widehat{A}$ denotes the set of prime theories containing A, and $\widehat{A}^{\mathrm{c}}$ denotes the complement of $\widehat{A}$. We call an Esakia space X *finitely copresented* if X is order-homeomorphic to a clopen up-set of $\mathbb{E}(\bar{x})$, for some finite $\bar{x}$. The name comes from the fact that it is equivalent to saying that the Heyting algebra dual to X is finitely presented; see also Section 2.3 below. The main contribution of [83] is the following open mapping theorem.

Theorem 2.3 ([83, Thm. 2]). *Every continuous bounded map between finitely copresented Esakia spaces is open.*

Given this result, one obtains Pitts' theorem (Theorem 2.1) as follows. When F is an intuitionistic formula in variables $\bar{x}, p$, it defines a clopen up-set $\widehat{F}$ of the Esakia space $\mathbb{E}(\bar{x}, p)$. The inclusion homomorphism $i_p : \mathbb{H}(\bar{x}) \hookrightarrow \mathbb{H}(\bar{x}, p)$ introduced above has a dual continuous bounded morphism $\pi_p : \mathbb{E}(\bar{x}, p) \to \mathbb{E}(\bar{x})$. Concretely, this morphism sends an intuitionistic prime theory in variables $\bar{x}, p$ to its intersection with the set of formulas that use only variables $\bar{x}$. By Theorem 2.3, the morphism π_p is open, and one obtains that the direct image of $\widehat{F}$ under π_p is a clopen up-set. Therefore, there exists a formula $\mathsf{E}_p(F) \in \mathbb{H}(\bar{x})$ such that

$$\widehat{\mathsf{E}_p(F)} = \pi_p[\widehat{F}].$$

One can then verify that $\mathsf{E}_p(F)$ indeed satisfies the required properties for the right uniform interpolant of F with respect to p. The left uniform interpolant $\mathsf{A}_p(F)$ can be obtained in a similar way, by considering the 'universal image' of $\widehat{F}$, that is, one finds a formula $\mathsf{A}_p(F)$ in $\mathbb{H}(\bar{x})$ such that

$$\widehat{\mathsf{A}_p(F)} = \{x \in \mathbb{E}(\bar{x}) \mid \text{ for all } y \in \mathbb{E}(\bar{x}, p), \text{ if } \pi_p(y) \geq x, \text{ then } y \in \widehat{F}\},$$

and one shows that this formula $\mathsf{A}_p(F)$ indeed satisfies the properties for the left uniform interpolant of F with respect to p. This completes the proof that Theorem 2.1 follows from Theorem 2.3.

[5] The point of view on duality for lattice-based structures that I point to in this paragraph uses the *canonical extension*, i.e., the embedding of a lattice into the lattice of up-sets of its dual space. This idea originates with [105], for Boolean algebras, [64], for distributive lattices and their expansions, and [63] for lattice expansions in general. I previously studied canonical extensions for stably compact spaces and proximity lattices [78], and also used them for obtaining dualities for non-distributive lattices [59] and algebras for many-valued logic [58]. I use them as a tool for proving completeness in Section 3.1.

I will end this section by sketching some of the ingredients of our proof of Theorem 2.3 itself, which we gave in [83]. One first shows, with a simple topological argument, that it suffices to consider morphisms between the Esakia spaces $\mathbb{E}(\bar{x})$ that are dual to free finitely generated Heyting algebras. These spaces are metrizable by general topological principles, and our proof uses an explicit definition of a metric for the topology, that I will recall now. By Esakia duality, we identify $\mathbb{H}(\bar{x})$ with the clopen up-sets of $\mathbb{E}(\bar{x})$. That is, any clopen up-set of $\mathbb{E}(\bar{x})$ can be *described* by a Heyting algebra term with variables in $\bar{x}$. The *depth* of an clopen up-set K of $\mathbb{E}(\bar{x})$ is the minimum nesting depth of the operation $\Rightarrow$ that is required to describe K; we denote it by $|K|$. For any two points $x, y \in \mathbb{E}(\bar{x})$, we say that a clopen up-set K *separates* x from y if exactly one of the points x and y is in the set K. We now define

$$c(x,y) \overset{\text{def}}{=} \min\{|K| \ : \ K \text{ separates } x \text{ from } y\}, \qquad d(x,y) \overset{\text{def}}{=} 2^{-c(x,y)} \, .$$

The function d defines an ultrametric on $\mathbb{E}(\bar{x})$, and the topology on $\mathbb{E}(\bar{x})$ is generated by the clopen balls in this ultrametric [83, Lem. 9]. This is a familiar type of definition in the theory of profinite monoids, here applied in the slightly different case of Priestley space, which are in fact the same thing as profinite partial orders [160]. Indeed, the ultrametric d provides a *profinite approximation* of the ordered space $\mathbb{E}(\bar{x})$ by a sequence of finite partially ordered sets, in the following sense. For any $k \geq 0$ and $x, y \in \mathbb{E}(\bar{x})$, we write $x \leq_k y$ if $x \in K$ implies $y \in K$ for every $K \in \mathbb{H}(\bar{x})$ with $|K| \leq k$. The relation $\leq_k$ is a pre-order on $\mathbb{E}(\bar{x})$, and the quotient partial order, which we denote P_k, is finite. From the definitions, it follows that two points x and y are identified in P_k if, and only if, $d(x,y) < 2^{-k}$; thus, the finite poset P_k essentially consists of the clopen balls of radius 2^{-k}. With these definitions, proving that a continuous bounded map is open then reduces to a combinatorial argument that entirely takes place in this sequence P_k; I refer to our article [83, Sec. 5], for more details.

Note that the stratification of an Esakia space $\mathbb{E}(\bar{p})$ that we use here is similar in spirit to the sequence building up the free pro-aperiodic monoid in Eq. (1.3) used in Section 1.1 of Chapter 1, even if that stratification was related to nestings of quantifiers in first-order logic, rather than nestings of implications in intuitionistic propositional logic. I leave investigation of a more formal link between, or common framework for, these two methods to future work, also see my remarks in Section 2.4.

Our approach in [83] towards proving Pitts' theorem follows the spirit of an earlier proof [73], but avoids the heavier machinery of sheaves and games used there. In essence, we replace the categorical sheaf machinery by the use of topology, and we use an ultrametric on the space $\mathbb{E}(\bar{x})$ instead of model-theoretic games, in order to give us an appropriate induction parameter. Still, the combinatorial argument that we need to complete the proof of our open mapping theorem [83, Lemma 10] was directly inspired by the one given in [73], and, as far as we can tell, this combinatorial complication cannot be avoided in semantic proofs. As a consequence, while the proof outlined here gives a topological proof of *existence* of Pitts' uniform interpolants, it does not provide a feasible way of constructing them, nor of obtaining a reasonable bound on their complexity. In Section 2.2, I will discuss a more tractable way of computing uniform interpolants, going back to Pitts' original proof method.

2.2 Verified computation of uniform interpolants

In this section, I will discuss a verified implementation of the computation of uniform interpolants in intuitionistic propositional logic **I**, and in certain modal and intuitionistic modal logics.[6]

Existing proof methods for uniform interpolation can be divided, roughly, into two strands: one is syntactic and relies on the existence of a well-behaved sequent calculus for the logic, as in Pitts' original proof [139], see also [101], the other is semantic and uses either topology, as in Section 2.1 above, or Kripke models, in order to establish definability of bisimulation quantifiers [73]. An advantage of the syntactic method over the semantic one is that, at least in theory, it provides better bounds on the complexity of computing uniform interpolants. In practice, however, it is not feasible to compute uniform interpolants by hand, as the calculations quickly become complex even on small examples. The algorithms for computing uniform interpolants are often intricate, and it is a non-trivial task to implement them correctly.

It occurred to me around 2019 that this situation might make the syntactic method of proving uniform interpolation an excellent candidate for *verified* computation. Having recently started to work at a computer science laboratory, IRIF, and being lucky enough to share an office with an experienced user of the Coq proof assistant/Rocq Prover[7] [168], we set to work to implement a verified version of Pitts' algorithm for computing uniform interpolants in **I**, which we published in [56]. We subsequently extended these methods to provide a verified computation of uniform interpolants for the modal logics **K** and **GL**, and for the intuitionistic modal logic **iSL**. This in particular gave the new mathematical result that the logic **iSL** has uniform interpolation, resolving an open question of [75]. A different, semantic proof of this result has since appeared in the preprint [175]. I will here focus on describing the results for **I** and its modal extension **iSL**, referring to our publication [55] for more information about the cases of **K** and **GL**. I will first give an overview of the proof, and I will comment a bit more on their formalization in Coq/Rocq at the end of the section.

The proof we follow in this line of work is based on the original method of Pitts [139], and relies on a proof calculus for **I** known as LJT or G4iP in the literature [47, 99, 176]. The main features of the calculus G4iP are that it allows for a terminating proof search without loop checking, and that it does not have a contraction rule. This calculus has itself often been at the basis of the implementation of proof search for proof assistants, notably Coq/Rocq's 'firstorder' tactic [38]. The most intricate part of Pitts' proof, and consequently also of our formalization, is the proof of correctness of the definition of propositional quantifiers, which is done by induction on the structure of a G4iP-proof. We adapt this to a sequent calculus G4iSLt developed in [159].

The *intuitionistic modal language* contains, in addition to the language of intuitionistic propositional logic **I**, an additional unary operator $\Box$. We will from now on denote formulas by lowercase Greek letters $\phi, \psi, \ldots$ and we write $\mathsf{Var}(\phi)$ to denote the set of all propositional variables occurring in the formula ϕ. The *normal axiom* (k) is the formula $\Box(p \Rightarrow q) \Rightarrow \Box p \Rightarrow \Box q$, and the *strong Löb axiom* (sl)[8] is the formula $(\Box p \Rightarrow p) \Rightarrow p$. Also recall the rules *modus ponens*: From ϕ and $\phi \Rightarrow \psi$ infer

[6] The work reported in this section started in 2020 in collaboration with my then-new office-mate at IRIF, Hugo Férée, published in [56]. This led to a subsequent paper of ours, in collaboration with two post-docs, van der Giessen and Shillito, which received the Best Paper Award in IJCAR 2024 [55].

[7] It has been decided a few years ago by the Coq/Rocq development community that Coq will be renamed to 'The Rocq Prover'. At the time of writing (July 2024), according to the software's homepage https://coq.inria.fr, the rename was in preparation, with a new visual identity and website planned for the end of 2024. I refer to it as 'Coq/Rocq' in this text.

[8] This axiom was introduced in the context of the study of provability logic in extensions of Heyting arithmetic, where the

ψ, *necessitation*: From ϕ infer $\Box\phi$, and *substitution*: From ϕ infer $\sigma\phi$, for any uniform substitution σ. Now, intuitionistic modal logic **iSL** is defined as the smallest set of formulas containing all intuitionistic tautologies, axioms k and sl, and closed under the rules modus ponens, necessitation, and substitution.

An intuitionistic *sequent* is a pair of a finite multiset of formulas Γ and a formula ϕ, which we denote by $\Gamma \vdash \phi$. Given two multisets Γ and Δ, we write Γ, Δ for the multiset addition of Γ and Δ, and, when ϕ is a formula, we write Γ, ϕ as notation for $\Gamma, \{\phi\}$. We also write $\mathsf{Var}(\Gamma) \overset{\text{def}}{=} \bigcup_{\gamma \in \Gamma} \mathsf{Var}(\gamma)$. For p a propositional variable, we write $\Gamma_p \overset{\text{def}}{=} \Gamma \setminus \{p\}$ for any multiset Γ. We also use the following notation $\Box^{-1}$ on formulas:

$$\Box^{-1}\psi \overset{\text{def}}{=} \begin{cases} \phi & \text{if } \psi = \Box\phi \text{ for some formula } \phi, \\ \psi & \text{otherwise.} \end{cases}$$

This notation is naturally overloaded to also apply to (multi)sets of formulas: $\Box^{-1}\Gamma \overset{\text{def}}{=} \{\Box^{-1}\phi \mid \phi \in \Gamma\}$.

We work with the sequent calculus G4iSLt [159], which was specifically designed with the aim of proving uniform interpolation for **iSL**. The calculus is an extension of the calculus G4iP for **I** [46]. We show the calculi G4iP and G4iSLt in Figure 2.2, using the $\Box^{-1}$ operator to rephrase its definition slightly compared to [159]. In that figure we use the notation, common in proof theory, that an expression of the form

$$\frac{\Gamma_1 \vdash \phi_1 \qquad \dots \qquad \Gamma_n \vdash \phi_n}{\Delta \vdash \psi}\ (\mathrm{R})$$

denotes a *derivation rule* named 'R', which states: 'given the sequents $\Gamma_1 \vdash \phi_1, \dots, \Gamma_n \vdash \phi_n$, one may derive the sequent $\Delta \vdash \phi$'. A *sequent calculus* is a set of derivation rules. For a sequent calculus S, we denote by $\vdash_{\mathsf{S}}$ the set of sequents that are *derivable* using the rules in S. More formally, $\vdash_{\mathsf{S}}$ is defined to be the smallest set of sequents such that, for any rule (R) in S, if all the sequents above the horizontal line are in $\vdash_{\mathsf{S}}$, then the sequent below the horizontal line is also in $\vdash_{\mathsf{S}}$. We then write $\Gamma \vdash_{\mathsf{S}} \phi$ to mean that the sequent $\Gamma \vdash \phi$ is an element of the set $\vdash_{\mathsf{S}}$. An equivalent way of stating this definition is: $\Gamma \vdash_{\mathsf{S}} \phi$ if, and only if, there exists a finite, upwards growing tree whose nodes are sequents, with $\Gamma \vdash \phi$ at the root, and such that each node and its children above it form an instance of a rule in S; in particular, leaves should be instances of rules in S that have no formulas above the horizontal line.

The sequent calculi of Fig. 2.2 are *sound and complete* for the logics **iSL** and **I**, meaning that a sequent of the form $\emptyset \vdash \phi$ is derivable in G4iP if, and only if, it is an intuitionistic tautology in **I**, and it is derivable in G4iSLt if, and only if, it is in **iSL**. Crucially, these calculi also represent a *terminating strategy* for a proof of a tautology, as we explain now.

The *weight* $w(\phi)$ of a formula ϕ is defined by adding up weights for symbols occurring in the formula: the symbols $\bot, \Box, \Rightarrow$ and variables count for 1, $\wedge$ for 2 and $\vee$ for 3. This naturally defines a well-founded strict preorder on the set of formulas: $\phi \prec_f \psi$ iff $w(\phi) < w(\psi)$. In [46], the preorder on sequents used to prove the termination of G4iP comes from the *Dershowitz-Manna* ordering on (finite) multisets induced by this preorder on the elements, where a multiset B is considered greater than a multiset A if B can be obtained from A by replacing elements from A by greater elements, and/or adding new elements. We then define $\Gamma \vdash \phi \prec \Delta \vdash \psi$ if the multiset Γ, ϕ is smaller than the multiset Δ, ψ in this ordering. The important point, which ensures termination for G4iP, is that the

intended interpretation of '$\Box p$' is 'p is provable'; see [174, Ch. 4] and [74, Sec. 1.3.2].

$$\frac{}{\bot, \Gamma \vdash \chi}\ (\bot L) \qquad \frac{}{\Gamma, p \vdash p}\ (\mathrm{IdP}) \qquad \frac{\Gamma, \varphi, \psi \vdash \chi}{\Gamma, \varphi \wedge \psi \vdash \chi}\ (\wedge L) \qquad \frac{\Gamma \vdash \varphi \qquad \Gamma \vdash \psi}{\Gamma \vdash \varphi \wedge \psi}\ (\wedge \mathrm{R})$$

$$\frac{\Gamma, \varphi \vdash \chi \qquad \Gamma, \psi \vdash \chi}{\Gamma, \varphi \vee \psi \vdash \chi}\ (\vee \mathrm{L}) \qquad \frac{\Gamma \vdash \varphi_i}{\Gamma \vdash \varphi_1 \vee \varphi_2}\ (\vee R_i)(i\in\{1,2\}) \ \frac{\Gamma, \varphi \vdash \psi}{\Gamma \vdash \varphi \Rightarrow \psi}\ (\Rightarrow \mathrm{R})$$

$$\frac{\Gamma, \varphi \Rightarrow (\psi \Rightarrow \chi) \vdash \delta}{\Gamma, (\varphi \wedge \psi) \Rightarrow \chi \vdash \delta}\ (\wedge{\Rightarrow}\mathrm{L}) \qquad \frac{\Gamma, \varphi \Rightarrow \chi, \psi \Rightarrow \chi \vdash \delta}{\Gamma, (\varphi \vee \psi) \Rightarrow \chi \vdash \delta}\ (\vee{\Rightarrow}\mathrm{L})$$

$$\frac{\Gamma, p, \varphi \vdash \chi}{\Gamma, p, p \Rightarrow \varphi \vdash \chi}\ (p{\Rightarrow}\mathrm{L}) \qquad \frac{\Gamma, \psi \Rightarrow \chi \vdash \varphi \Rightarrow \psi \qquad \Gamma, \chi \vdash \delta}{\Gamma, (\varphi \Rightarrow \psi) \Rightarrow \chi \vdash \delta}\ ({\Rightarrow}{\Rightarrow}\mathrm{L})$$

$$\frac{\Box^{-1}\Gamma, \Box\phi \vdash \phi}{\Gamma \vdash \Box\phi}\ (\Box\mathrm{R}) \qquad \frac{\Box^{-1}\Gamma, \Box\phi, \psi \vdash \phi \qquad \Gamma, \psi \vdash \chi}{\Gamma, \Box\phi \Rightarrow \psi \vdash \chi}\ (\Box \Rightarrow\mathrm{L})$$

Figure 2.2: The sequent calculus G4iSLt. The sequent calculus G4iP is the restriction of G4iSLt obtained by omitting the two rules involving □.

Dershowitz-Manna ordering of a well-founded ordering is again well-founded. However, the $\Box_R$-rule of G4iSLt is not always compatible with this ordering. Indeed, for example, with $\Gamma = \emptyset$ and $\phi = \bot$, note that $\{\Box\bot, \bot\} \not\prec \{\Box\bot\}$. The reason is that this rule both replaces a boxed formula on the right hand side with its unboxed version, which is a strict subformula, but also moves the boxed formula to the left-hand side.

We fix this issue by modifying the weight of a sequent so that the right-hand side of the sequent in the multiset counts double, accounting for the fact that a formula on the right-hand side of a sequent might be duplicated using a $\Box_R$ rule: We write $\Gamma \vdash \phi \prec' \Delta \vdash \psi$ whenever Γ, ϕ, ϕ is smaller than Δ, ψ, ψ for the multiset ordering induced by $\prec_f$. The ordering $\prec'$ is again well-founded, and any hypothesis of a G4iSLt rule has strictly smaller weight than its conclusion.

We use the ordering $\prec'$ to recursively construct uniform interpolants. Adapting Pitts' original proof scheme for **I**, we now define the left and right uniform interpolants for **iSL** as follows. Let Γ be a multiset of formulas and ϕ a formula. Formulas $\mathsf{E}_p^{\mathbf{iSL}}(\Gamma)$ and $\mathsf{A}_p^{\mathbf{iSL}}(\Gamma \vdash \phi)$ are defined by mutual induction on the $\prec'$ ordering, respectively as a conjunction of a multiset of formulas $\mathcal{E}_{\mathsf{p}}(\Gamma)$ and as a disjunction of a multiset of formulas $\mathcal{A}_{\mathsf{p}}(\Gamma \vdash \phi)$, given in Fig. 2.3 below. We then show that the right uniform interpolant of ϕ with respect to the variable p is the formula $\mathsf{E}_p^{\mathbf{iSL}}(\{\phi\})$, and that the left uniform interpolant is the formula $\mathsf{A}_p^{\mathbf{iSL}}(\emptyset \vdash \phi)$.

An intuition behind the table of Fig. 2.3, in analogy with the classical case shown in Eq. (2.1) at the start of this chapter, is that $\mathsf{E}_p(\Gamma)$ should provide a basis for the set of those p-free formulas ψ so that $\Gamma \vdash \psi$ is derivable. The formula $\mathsf{A}_p(\Gamma \vdash \phi)$ should provide a basis for the set of those p-free formulas ψ such that $\Gamma, \psi \vdash \phi$ is derivable. Each line in the table in Fig. 2.3 then corresponds to an upward application of a deduction rule of the sequent calculus to the sequent in the middle column which might be used to construct such a derivation. The complexity of the procedure comes from the fact that the right column then recursively calls the procedure on $\prec'$-smaller sequents.

Our adaptation to **iSL** of Pitts' construction for **I** adds formulas to the sets $\mathcal{E}_p$ and $\mathcal{A}_p$ of Pitts' construction only in the cases where some formula in Δ, θ contains a boxed subformula. As a consequence, $\mathsf{A}_p^{\mathbf{iSL}}(\Gamma \vdash \phi) = \mathsf{A}_p^{\mathbf{I}}(\Gamma \vdash \phi)$ and $\mathsf{E}_p^{\mathbf{iSL}}(\Gamma) = \mathsf{E}_p^{\mathbf{I}}(\Gamma)$ whenever Γ and ϕ do not contain the □ modality.

	Γ matches	$\mathcal{E}_\mathsf{p}(\Gamma)$ contains
$(\mathsf{E}^{\mathbf{I}}_{p}0)$	$\Gamma', \bot$	$\bot$
$(\mathsf{E}^{\mathbf{I}}_{p}1)$	Γ', q	$\mathsf{E}_p(\Gamma') \wedge q$
$(\mathsf{E}^{\mathbf{I}}_{p}2)$	$\Gamma', \psi_1 \wedge \psi_2$	$\mathsf{E}_p(\Gamma', \psi_1, \psi_2)$
$(\mathsf{E}^{\mathbf{I}}_{p}3)$	$\Gamma', \psi_1 \vee \psi_2$	$\mathsf{E}_p(\Gamma', \psi_1) \vee \mathsf{E}_p(\Gamma', \psi_2)$
$(\mathsf{E}^{\mathbf{I}}_{p}4)$	$\Gamma', (q \Rightarrow \psi)$	$q \Rightarrow \mathsf{E}_p(\Gamma', \psi)$
$(\mathsf{E}^{\mathbf{I}}_{p}5)$	$\Gamma', p, (p \Rightarrow \psi)$	$\mathsf{E}_p(\Gamma', p, \psi)$
$(\mathsf{E}^{\mathbf{I}}_{p}6)$	$\Gamma', (\delta_1 \wedge \delta_2) \Rightarrow \delta_3)$	$\mathsf{E}_p(\Gamma', (\delta_1 \Rightarrow (\delta_2 \Rightarrow \delta_3)))$
$(\mathsf{E}^{\mathbf{I}}_{p}7)$	$\Gamma', (\delta_1 \vee \delta_2) \Rightarrow \delta_3)$	$\mathsf{E}_p(\Gamma', (\delta_1 \Rightarrow \delta_3), (\delta_2 \Rightarrow \delta_3)))$
$(\mathsf{E}^{\mathbf{I}}_{p}8)$	$\Gamma', (\delta_1 \Rightarrow \delta_2) \Rightarrow \delta_3)$	$[\mathsf{E}_p(\Gamma', (\delta_2 \Rightarrow \delta_3)) \Rightarrow \mathsf{A}_p(\Gamma', (\delta_2 \Rightarrow \delta_3) \vdash \delta_1 \Rightarrow \delta_2)] \Rightarrow \mathsf{E}_p(\Gamma', \delta_3)$
$(\mathsf{E}^{\mathbf{iSL}}_{p}9)$	$\Gamma', \Box\delta$	$\Box\mathsf{E}_p(\Box^{-1}\Gamma', \delta)$
$(\mathsf{E}^{\mathbf{iSL}}_{p}10)$	$\Gamma', (\Box\delta_1 \Rightarrow \delta_2)$	$\Box[\mathsf{E}_p(\Box^{-1}\Gamma', \delta_2, \Box\delta_1) \Rightarrow \mathsf{A}_p(\Box^{-1}\Gamma', \delta_2, \Box\delta_1 \vdash \delta_1)] \Rightarrow \mathsf{E}_p(\Gamma', \delta_2)$

	s matches	$\mathcal{A}_\mathsf{p}(s)$ contains
$(\mathsf{A}^{\mathbf{I}}_{p}1)$	$\Gamma, q \vdash \phi$	$\mathsf{A}_p(\Gamma \vdash \phi)$
$(\mathsf{A}^{\mathbf{I}}_{p}2)$	$\Gamma, \psi_1 \wedge \psi_2 \vdash \phi$	$\mathsf{A}_p(\Gamma, \psi_1, \psi_2 \vdash \phi)$
$(\mathsf{A}^{\mathbf{I}}_{p}3)$	$\Gamma, \psi_1 \vee \psi_2 \vdash \phi$	$[\mathsf{E}_p(\Gamma, \psi_1) \Rightarrow \mathsf{A}_p(\Gamma, \psi_1 \vdash \phi)] \wedge [\mathsf{E}_p(\Gamma, \psi_2) \Rightarrow \mathsf{A}_p(\Gamma, \psi_2 \vdash \phi)]$
$(\mathsf{A}^{\mathbf{I}}_{p}4)$	$\Gamma, (q \Rightarrow \psi) \vdash \phi$	$q \wedge \mathsf{A}_p(\Gamma, \psi \vdash \phi)$
$(\mathsf{A}^{\mathbf{I}}_{p}5)$	$\Gamma, p, (p \Rightarrow \psi) \vdash \phi$	$\mathsf{A}_p(\Gamma, \psi \vdash \phi)$
$(\mathsf{A}^{\mathbf{I}}_{p}6)$	$\Gamma, (\delta_1 \wedge \delta_2) \Rightarrow \delta_3) \vdash \phi$	$\mathsf{A}_p(\Gamma, (\delta_1 \Rightarrow (\delta_2 \Rightarrow \delta_3)) \vdash \phi)$
$(\mathsf{A}^{\mathbf{I}}_{p}7)$	$\Gamma, (\delta_1 \vee \delta_2) \Rightarrow \delta_3) \vdash \phi$	$\mathsf{A}_p(\Gamma, (\delta_1 \Rightarrow \delta_3), (\delta_2 \Rightarrow \delta_3)) \vdash \phi)$
$(\mathsf{A}^{\mathbf{I}}_{p}8)$	$\Gamma, (\delta_1 \Rightarrow \delta_2) \Rightarrow \delta_3) \vdash \phi$	$[\mathsf{E}_p(\Gamma, (\delta_2 \Rightarrow \delta_3)) \Rightarrow \mathsf{A}_p(\Gamma, (\delta_2 \Rightarrow \delta_3) \vdash \delta_1 \Rightarrow \delta_2)] \wedge\ \mathsf{A}_p(\Gamma, \delta_3 \vdash \phi)$
$(\mathsf{A}^{\mathbf{I}}_{p}9)$	$\Gamma \vdash q$	q
$(\mathsf{A}^{\mathbf{I}}_{p}10)$	$\Gamma, p \vdash p$	$\top$
$(\mathsf{A}^{\mathbf{I}}_{p}11)$	$\Gamma \vdash \phi_1 \wedge \phi_2$	$\mathsf{A}_p(\Gamma \vdash \phi_1) \wedge \mathsf{A}_p(\Gamma \vdash \phi_2)$
$(\mathsf{A}^{\mathbf{I}}_{p}12)$	$\Gamma \vdash \phi_1 \vee \phi_2$	$\mathsf{A}_p(\Gamma \vdash \phi_1) \vee \mathsf{A}_p(\Gamma \vdash \phi_2)$
$(\mathsf{A}^{\mathbf{I}}_{p}13)$	$\Gamma \vdash \phi_1 \Rightarrow \phi_2$	$\mathsf{E}_p(\Gamma, \phi_1) \Rightarrow \mathsf{A}_p(\Gamma, \phi_1 \vdash \phi_2)$
$(\mathsf{A}^{\mathbf{iSL}}_{p}14)$	$\Gamma \vdash \Box\delta$	$\Box(\mathsf{E}_p(\Box^{-1}\Gamma, \Box\delta) \Rightarrow \mathsf{A}_p(\Box^{-1}\Gamma, \Box\delta \vdash \delta)).$
$(\mathsf{A}^{\mathbf{iSL}}_{p}15)$	$\Gamma, \Box\delta_1 \Rightarrow \delta_2 \vdash \phi$	$\Box[\mathsf{E}_p(\Box^{-1}\Gamma, \delta_2, \Box\delta_1) \Rightarrow \mathsf{A}_p(\Box^{-1}\Gamma, \delta_2, \Box\delta_1 \vdash \delta_1)] \wedge \mathsf{A}_p(\Gamma, \delta_2 \vdash \phi)$

Figure 2.3: Tables for computing the uniform interpolants for **I** and **iSL** syntactically. The top part of each table, i.e., $(\mathsf{E}^{\mathbf{I}}_{p}0)$-$(\mathsf{E}^{\mathbf{I}}_{p}8)$ and $(\mathsf{A}^{\mathbf{I}}_{p}1)$-$(\mathsf{A}^{\mathbf{I}}_{p}13)$, define $\mathcal{E}_\mathsf{p}(\Gamma)$ and $\mathcal{A}_\mathsf{p}(\Gamma \vdash \phi)$ in the case of **I**, as in [139]. The complete table provides definitions for $\mathcal{E}_\mathsf{p}(\Gamma)$ and $\mathcal{A}_\mathsf{p}(\Gamma \vdash \phi)$ for **iSL**. In all clauses, q denotes any propositional variable distinct from p.

The main claim is now that this algorithm for computing uniform interpolants outlined above is correct, by which we mean that, for any formula ϕ, the formulas $\mathsf{E}^{\mathbf{iSL}}_{p}(\{\phi\})$ and $\mathsf{A}^{\mathbf{iSL}}_{p}(\varnothing \vdash \phi)$ are indeed uniform interpolants on the right and left for ϕ. In order to prove this correctness, we rely on the *admissibility* of the weakening and contraction rules for G4iSLt, by which we mean that these rules, if added to G4iSLt, do not add any new derivable sequents. Combining this with an induction on the length of a possible derivation of a sequent and the weight-based sequent ordering $\prec'$, and a large case distinction on the last rule in this derivation, we show correctness of our computation for **iSL**. Note that this in particular establishes Theorem 2.1, by omitting any rules that mention $\Box$.

I end this section with a few remarks on our mechanization of the computation of uniform interpolants, and the formal proof of correctness, referring to our more detailed comments in [55, 56] for more information. As mentioned above, the mechanization of our proof was carried out in Coq/Rocq, which is a piece of software that takes as input an encoding of mathematical statements in a formal language of *dependent type theory*. The formal statements given by the user as input are *checked* by the software, in much the same way that a programming language compiler checks that a program written in the language is correctly typed. An important, and at first rather surprising, fact is that *any* statement that one would write in mathematics can, in principle, be translated into a formal statement in the type theory of Coq/Rocq. A second important property of Coq/Rocq is that it is *constructive*, by which I here mean that the formal statements encoded in it, once checked by Coq/Rocq, can also be interpreted as the types of *programs* that the computer can execute.[9] This second property is relevant to our work here because it allows us to *extract* out of our Coq/Rocq formalization a usable program, which lets end-users compute uniform interpolants, in a way that does not require any installation or knowledge of the Coq/Rocq development in the background. Such an extracted program for our work here is available for online experimentation at https://hferee.github.io/UIML/demo.html. The Coq/Rocq development itself is available at https://github.com/hferee/UIML.

2.3 Compact congruences and model-completeness

In this section, I take a broader view on uniform interpolation, by generalizing from the intuitionistic and modal logics discussed in the previous sections to arbitrary classes of algebraic structures. This leads us to establish an intimate connection between uniform interpolation, preservation properties of *compact congruences*, and quantifier elimination, in the form of *model-completeness*.[10]

Throughout the rest of this section, we fix an algebraic type in the sense of universal algebra (see, e.g., [29, §II.1]), and by the word 'algebra' we mean any structure interpreting that type. Concretely, this means that we fix a finite set σ of *operation symbols*, and, for each $f \in \sigma$, an *arity* $n(f) \in \mathbb{N}$. An *algebra* is, by definition, a set A equipped with a function $f^A : A^{n(f)} \to A$, for each $f \in \sigma$. We further assume for convenience that σ contains at least one operation symbol of arity 0.

The key notion in the algebraic study of uniform interpolation is that of a *compact congruence*. Recall[11] that a *congruence* on an algebra A is an equivalence relation on A that is invariant under all operations of the algebraic type. When θ is a congruence on A, the quotient set A/θ admits a unique algebra structure so that $\nu_\theta : A \to A/\theta$ is a homomorphism. The *universal property* of the quotient algebra A/θ says that any homomorphism $h : A \to B$ such that $h(a) = h(a')$ for every $(a, a') \in \theta$ factors uniquely as $A \xrightarrow{\nu_\theta} A/\theta \xrightarrow{\bar{h}} B$. The set of congruences on A, $\mathrm{Con}(A)$, when ordered by inclusion, is a complete lattice, in which the infimum is given by intersection, and the supremum of a set of congruences is the congruence generated by the union. A congruence is called *compact*[12] if it is finitely generated.

Congruences can be used to define a general notion of equational consequence, as follows. Denote by $T(X)$ the *term algebra* over a set of variables X, in our fixed algebraic type σ. If A is any algebra of type σ and $V : X \to A$ is a function sending each variable in X to an element of A, then the unique

[9] There is a rich and fascinating theory behind this idea, which extends the Curry-Howard correspondence mentioned in the introduction to this chapter (p. 22). A readable introduction to the topic is [133, Ch. 5], including, in Sec. 5.7, some history and references for the practical applications of this theory to proof assistant software, including Coq/Rocq.

[10] The work reported in this section was started during my post-doc in Bern in 2014, and published in [82].

[11] For a textbook treatment of congruences in universal algebra, see, e.g., [29, §II.5–7].

[12] This usage of the word 'compact' originates with the theory of directedly complete partial orders, see, e.g., [60, Def. 7.10], for more context.

extension of V to a homomorphism $\bar{V} : T(X) \to A$ yields, for every term $t \in T(X)$, an element $\bar{V}(t)$ of A, that we call the result of *evaluating* the term t in A under V; when $x_1, \dots, x_n$ are the variables occurring in t, we may also denote this evaluation result by $t(V(x_1), \dots, V(x_n))$.

By an *equation* over X, we mean a pair of terms, which we denote by $s \approx t$ to signify its intended interpretation. When A is an algebra, we say that $s \approx t$ *holds* in A, written $A \vDash s \approx t$, if, for every valuation of the variables of s and t in A, the results of evaluating s and t are the same. Let $\mathcal{K}$ be a class of algebras. We define the *equational consequence* relation in the class $\mathcal{K}$. Let $\mathcal{E} \subseteq T(X)^2$ be a set of equations and let $s \approx t$ be an equation. We define:

$$\begin{aligned}\mathcal{E} \vDash_{\mathcal{K}} s \approx t \overset{\text{def}}{\iff} & \text{ for every algebra } A \in \mathcal{K} \text{ and every homorphism } f : T(X) \to A, \\ & \text{if } f(u) = f(v) \text{ for all}(u, v) \in \mathcal{E}, \text{ then } f(s) = f(t).\end{aligned}$$

For a set of equations $\mathcal{F} \subseteq T(X)^2$, we also write $\mathcal{E} \vDash_{\mathcal{K}} \mathcal{F}$ if $\mathcal{E} \vDash_{\mathcal{K}} s \approx t$ holds for *every* equation $s \approx t$ in $\mathcal{F}$. In other words, $\mathcal{E} \vDash_{\mathcal{K}} \mathcal{F}$ is shorthand for the assertion that the infinitary formula

$$\forall \bar{x}. \left(\bigwedge_{(u,v) \in \mathcal{E}} u = v \right) \to \left(\bigwedge_{(s,t) \in \mathcal{F}} s = t \right)$$

is verified in every algebra A of the class $\mathcal{K}$. When $\mathcal{E}$ is empty, we simply write $\vDash_{\mathcal{K}} s \approx t$ and $\vDash_{\mathcal{K}} \mathcal{F}$.

Now fix an arbitrary class of algebras $\mathcal{K}$ and write $\vDash$ instead of $\vDash_{\mathcal{K}}$. We formally define the algebraic counterparts of the notions of interpolant and uniform interpolant that we considered above.

Let X_1 and X_2 be sets of variables, let $\mathcal{A} \subseteq T(X_1)^2$ and $\mathcal{C} \subseteq T(X_2)^2$ be finite sets of equations, and suppose that $\mathcal{A} \vDash \mathcal{C}$. An *interpolant* of this consequence relation is defined to be a finite set of equations $\mathcal{B} \subseteq T(X_1 \cap X_2)^2$ such that $\mathcal{A} \vDash \mathcal{B}$ and $\mathcal{B} \vDash \mathcal{C}$.

Now let X be a set of variables, Y a subset of X, and $\mathcal{A} \subseteq T(X)^2$ a finite set of equations. A *right uniform restriction* of $\mathcal{A}$ with respect to the set of variables Y is a finite set of equations $\mathcal{R} \subseteq T(Y)^2$ such that $\mathcal{A} \vDash \mathcal{R}$, and, for any equation $\alpha \in T(Y)^2$,

$$\text{if } \mathcal{A} \vDash \alpha, \text{ then } \mathcal{R} \vDash \alpha \,. \tag{2.3}$$

We say that a right uniform restriction is a *right uniform interpolant* if we can further allow the equation α to contain any other variables not in X, that is, if Eq. (2.3) holds for any equation $\alpha \in T(Y \cup Z)^2$, where Z is any set of variables disjoint from X. We say that a class of algebras *has* interpolants if an interpolant exists for any finite $\mathcal{A}, \mathcal{C}$ such that $\mathcal{A} \vDash \mathcal{C}$. We say that a class *has* uniform restrictions if a uniform restriction exists for any $\mathcal{A}$, and any subset of the variables appearing in $\mathcal{A}$, and analogously for uniform interpolants.

The reader will hopefully recognize the similarity between the above definitions in terms of the equational consequence relation and the 'logical' definitions given at the start of this chapter. The subtle distinction between 'right uniform restriction' and 'right uniform interpolant' is introduced here to be able to speak algebraically about model-completeness in the absence of interpolation, as we will see below. We did not encounter this distinction in the preceding sections, because the logics that we considered so far all have interpolants, and in that case, uniform restriction and uniform interpolation coincide [82, Prop. 3.5].

An algebra A in $\mathcal{K}$ is called *finitely presented* if it is isomorphic to a quotient of a free algebra by a compact congruence. A class $\mathcal{K}$ is called *coherent* if any finitely generated subalgebra of a finitely presented algebra is itself finitely presented.[13] The following theorem contains the first main result we proved in [82, Thm. 3.2], incorporating also a later improvement of [108, Thm 2.3].[14]

Theorem 2.4. *A variety has right uniform restrictions if, and only if, it is coherent. A variety has right uniform interpolants if, and only if, it is coherent and has interpolants.*

For the proof of Theorem 2.4, we developed a general theory for compact congruences. Note first that, when $h \colon A \to B$ is a homomorphism between finitely presented algebras, we always have an adjunction $h^* \colon \operatorname{Con} A \leftrightarrows \operatorname{Con} B \colon h^{-1}$ between the congruence lattices: If ψ is a congruence on A, then $h^*(\psi)$ is defined as the congruence on B generated by the pairs $(h(a), h(b))$, as (a, b) ranges over all pairs in ψ, and if θ is a congruence on B, then $h^{-1}(\theta)$ is defined as the kernel of the composite map $A \to B \to B/\theta$. This pushforward h^* always restricts to a map between the compact congruences, $h^* \colon K\operatorname{Con} A \to K\operatorname{Con} B$, but the inverse image h^{-1} need not preserve compactness of congruences; for a concrete example of such a failure in the case of the variety of groups, see [82, Exa. 3.6]. We then establish that a variety has right uniform restrictions if, and only if, for any finite generating sets X and Y, the natural inclusion $i \colon F(Y) \hookrightarrow F(X \cup Y)$ has the property that i^{-1} sends compact congruences to compact congruences. We show that, in this case, it moreover follows that h^{-1} preserves compactness for *any* homomorphism between finitely presented algebras [82, Prop. 3.8], and is right adjoint to the map $h^* \colon K\operatorname{Con} A \to K\operatorname{Con} B$. From these results, one may deduce Theorem 2.4 by some general algebra, see [82, Sec. 3] and [108, Thm. 2.3] for more details.

Left and right uniform interpolation do not behave symmetrically from an algebraic point of view. Let Y and Z be finite sets of variables and $C \subseteq T(Y \cup Z)^2$ a finite set of equations. A *left uniform interpolant* of C with respect to Z is a finite set of equations $\mathcal{L} \subseteq T(Y)^2$ such that $\mathcal{L} \models C$ and, for any set of equations $\mathcal{E}$ not using any variables from Z, if $\mathcal{E} \models C$ then $\mathcal{E} \models \mathcal{L}$. We show in [82, Prop. 4.3] that a variety $\mathcal{V}$ has left uniform interpolants if, and only if, $\mathcal{V}$ has interpolants, and for any finite generating sets X and Y, the inclusion $i \colon F(Y) \hookrightarrow F(X \cup Y)$ has the property that i^* has a left adjoint. This property however does *not* lift to arbitrary homomorphisms of finitely presented algebras in general. The condition that h^* has a left adjoint for any homomorphism h between finitely presented algebras is examined in detail in [82, Sec. 4], where we prove the following. When X is a finite set of variables and $\mathcal{A}, \mathcal{B}$ are finite sets of equations, we call a finite set of equations $\mathcal{S}$ the *subtraction* of $\mathcal{A}$ from $\mathcal{B}$ if, for any finite set of equations $\mathcal{E}$, we have that $\mathcal{E}, \mathcal{A} \models \mathcal{B}$ if, and only if, $\mathcal{E} \models \mathcal{S}$. We say that a class of algebras *has subtractions* if subtractions exist for any finite sets of equations. The condition that a variety $\mathcal{V}$ has subtractions of equations says, in algebraic terms, that, for any finitely presented $\mathcal{V}$-algebra A, the join operation $\vee$ on the semilattice $K\operatorname{Con} A$ has a lower adjoint [82, Prop. 4.7].

Theorem 2.5 ([82, Thm. 4.10])**.** *For any variety $\mathcal{V}$, the following are equivalent:*

1. *$\mathcal{V}$ has left uniform interpolants and subtractions;*
2. *$\mathcal{V}$ has interpolants, and, for any homomorphism $h \colon A \to B$ between finitely presented $\mathcal{V}$-algebras, $h^* \colon K\operatorname{Con} A \to K\operatorname{Con} B$ has a left adjoint.*

[13] Coherence is a classical notion in the theory of modules, and was introduced into model theory by Wheeler [178]. The same notion has been studied in-depth for theories of actions of a monoid on a set, see, e.g., [41, 90].

[14] I only state the theorem in the case of *varieties*, i.e., classes of algebras defined by equations. The follow-up work [128] proves more general results, which apply to any *universal class* of algebras.

Model-completeness and model companions. Model-theoretic algebra, originating with the ground-breaking work of A. Robinson [150, 151], casts the basic problem of *solving equations* in a logical form, and uses this setting to solve algebraic problems via model theory. A central notion is that of an *existentially closed structure* for a first-order theory T, which we explain now. Call a quantifier-free first-order formula[15] ϕ, with parameters in a model M of T, *solvable* if there is an extension M' of M which is a model of T and has a *solution* for the formula ϕ, i.e., an assignment of the free variables such that ϕ is verified. The model M is *existentially closed* if any solvable quantifier-free formula already has a solution in M itself. The original motivating example for this definition is given by taking T to be the theory of fields. The field of real numbers is not existentially closed: The formula $x^2 + 1 = 0$ is solvable, but does not have a solution in $\mathbb{R}$. The field of complex numbers is the existentially closed extension of this model $\mathbb{R}$.

Although the definition of existential closedness is formally clear, a major drawback is that the property of being existentially closed is in general not definable by a set of first-order sentences, and thus falls out of the scope of usual model-theoretic methods. However, in fortunate cases, the class of existentially closed models of a first-order theory T are exactly the models of another first-order theory, T^*. In this case, the theory T^* is called the *model companion*[16] of T.

For later use, I now recall a more abstract characterization of the model companion of a first-order theory T, which is usually taken as the definition in model theory. Let $T \subseteq T^*$. We say that T^* is a *cotheory* of T if any universal sentence ϕ that follows from T^* already follows from T; equivalently, any model of T embeds into some model of T^*. A theory T^* is called *model complete* if for any sentence ϕ there is an existential sentence ψ such that ϕ is equivalent to ψ in the theory T^*; equivalently, any injective homomorphism between models of T^* is an elementary embedding. Now, T^* is a *model companion* of T if it is a model-complete cotheory of T. If, moreover, the theory T has *amalgamation*, then T^* is called a *model completion* of T. Here, a theory T has *amalgamation* if, for any model A of T and extensions B and C of A that are also models of T, there exists a model D which extends both B and C, in such a way that A is preserved. For example, the theory of Heyting algebras has amalgamation; this statement is essentially equivalent to the fact that intuitionistic propositional logic has (non-uniform) interpolants [117]. A model completion T^* always has the quantifier elimination property. For more details on these definitions and their history, see, e.g., [178, Sec. 1] and [32, Sec. 3.5].

Ghilardi and Zawadowski [72, 73] realized that the uniform interpolants of intuitionistic logic can be used to establish the existence of model companions. In particular, they use Pitts' Theorem 2.1 to prove that the theory of Heyting algebras has a model completion. In their own words ([73, p. 4]),

> "This is a rather interesting kind of connection: it says that the existence of a classical theory (the model completion) is equivalent to the existence of a suitable intuitionistic theory."

The main idea for establishing this connection is that the propositional quantifiers that are guaranteed to exist by Theorem 2.1 allow one to express, inside the algebraic type of Heyting algebras, the solvability of a quantifier-free formula. Based on Wheeler's previous work [178], Ghilardi and Zawadowski also connect this to properties of the category of finitely presented Heyting algebras,

[15] In some contexts, including the ones in this section, quantifier-free formulas reduce to conjunctions of equations; the notion is then also called *algebraically closed*. The definitions we give here apply when the first-order theory T is axiomatized by universal sentences; the more general definitions can be found in, e.g., [32, Sec. 3.5].

[16] Note that, despite the one-letter difference, 'model companion' is an entirely different concept from 'modal companion'. For the latter, see, e.g., [31, Sec. 9.6].

and this work was an important precursor to our algebraic characterizations in [82] described above. Incorporating these ideas in our setting allows us to prove the following:

Theorem 2.6 ([82, Thm. 5.2])**.** *Let $\mathcal{V}$ be a variety of algebras with the amalgamation property, such that $\mathcal{V}$ has left and right uniform interpolants and subtractions. Then $\mathcal{V}$ has a model completion.*

This theorem was subsequently extended in [128] to give a syntactic characterization of the existence of model completions in any universal class. We will return to model companions for algebras associated to temporal logics in Section 3.1. The methods that we use there are not directly related to uniform interpolation, but use automata theory to compute propositional quantifiers in that setting.

2.4 Outlook on uniform interpolation

In this chapter, I described some work on uniform interpolation and the computation of propositional quantifiers in intuitionistic and modal logics, and made a connection between this theory and universal algebra and model theory, through the notions of compact congruence and model companion. I will now outline some questions and directions for further research.

The approach from [83] that I described in Section 2.1 was recently generalized by Marquès to the class of *intuitionistic compact ordered spaces* in [120, Sec. 1.4]. These spaces generalize Esakia spaces by relaxing the requirement that the space be totally-order-disconnected to only requiring that the partial order be a closed subset of the square, while still requiring that ${\downarrow}U$ is open for any open subset U. The motivation for generalizing in this way comes from the desire to allow truth values of basic propositions to be taken in a *continuous* domain, such as $[0, 1]$, rather than the discrete $\{0, 1\}$.[17] Marquès develops a duality between these spaces and a continuous, $[0, 1]$-valued version of Heyting algebras, and constructs co-free intuitionistic compact ordered spaces, which correspond to the type spaces of an intuitionistic continuous propositional logic. He uses this to establish an open mapping theorem for the spaces, following our methods in [83], and, hence, a uniform interpolation theorem for continuous intuitionistic logic. This exciting development clearly begs for further investigation (also see [120, Question 1.4.30]): Can the uniform interpolants for this logic also be understood in a more syntactic way? And, recalling that Pitts' original theorem was motivated by a question in topos theory [139, p. 36], does this more general uniform interpolation theorem also have a geometric interpretation?

The stratification technique that we employ in our proof in Section 2.1 is not limited to Esakia spaces, and, as already pointed out in that section, it bears a formal similarity to the construction of certain profinite monoids in terms of explicit approximating chains of finite monoids, as in Eq. (1.3) of Chapter 1. What is a common framework that can capture and exploit this similarity? A line of recent work that goes in this direction proposes *game comonads* to give a general framework for dealing with 'resource bounds' (such as the quantifier depth of a formula, or the number of rounds in a game), and uses these to study composition theorems like the ones mentioned in Section 1.3, see, e.g., [102]. Abramsky and Reggio [1] introduce an axiomatic categorical framework of *arboreal categories* for understanding game comonads and homomorphism preservation theorems, with further applications

[17] This idea has appeared in many forms in the logic literature, and has been variously referred to as 'fuzzy', 'many-valued' or 'continuous' logic. It originates with Łukaciewicz [116], Chang and Keisler [33, 34, 35], and is intensely studied in algebraic logic [36, 131]; it has also recently been receiving renewed interest in the model theory community [92]. In my PhD, I developed dualities and sheaf representations for algebraic structures associated to these logics [58, 61, 78].

to logic given in [146]. Can these techniques also provide a framework for studying (uniform) interpolation theorems? Specifically, the proof-theoretic methods of Section 2.2 are amenable to proving interpolation results for logics other than **I**, and there exists a fine analysis of the relationship between sequent calculi and interpolation properties [101]. On the semantic side, there also exist applications of duality for proving uniform interpolation of certain modal logics such as **K** and **GL** [73, 173], but a full understanding of when, and why, uniform interpolation holds and fails for a logic, remains to be developed. I suggest that it would be worth investigating if some of the algebraic and categorical techniques mentioned here could serve this purpose.

Interpolation has been studied in proof theory for various substructural logics, including linear logic, using a technique called 'Maehara's method' [167, p. 32]. In 2023, I presented the results of [56] at a meeting of the French research community of formal structures for computation and proofs ('*GT Scalp*'), where I posed the question what might be the computational, or 'proof-relevant', meaning of interpolation, and in particular *uniform* interpolation, in light of the sequent-based proofs discussed in Section 2.2 above. The question was taken up enthusiastically by an IRIF colleague of mine, Alexis Saurin, who points out in [154] that a proof-relevant interpolation theorem is essentially the same thing as a method for introducing syntactically controlled cuts in a sequent calculus derivation. Extending this viewpoint to *uniform* interpolation theorems remains an open problem, that I hope to address in future work in collaboration with Férée and Saurin.

In light of the work discussed in Section 2.2, I will now briefly comment more generally on the activity of *formalizing* (also known as *mechanizing* or *machine-checking*) mathematics, in which I have participated over the past few years in various collaborations.[18] While formalization is not my main research activity, it has been an enriching experience for me to take part in it. Formalization is a way of practicing 'applied logic'. As such, it is an experience that I would recommend to any logically inclined mathematician or computer scientist to try out. The activity of formalizing often leads to new questions and issues, ranging from the efficiency of an implementation to the organization and distribution of software and software development.

Proof assistants come in many forms and flavors. I have so far mostly experimented with Coq/Rocq and Lean. The latter has the distinguishing feature of having an associated mathematical library, which is being developed in an ongoing large-scale collaborative effort [123]. Thanks to my previous experience with Coq/Rocq, I have recently been able to contribute some formalizations of results related to Stone duality to this library. The vision behind such formalization work is a bit different from the one underlying the results that I described in Section 2.2: Rather than verifying a particular algorithm that leads to a usable piece of software, or machine-checking one particular mathematical result, one contributes incrementally to the construction of a shared resource of mechanized mathematical facts. Such a resource may then be used by others (humans, or, so some believe, machines [57]) in order to formalize their own theorems.

Building on the work reported on in Section 2.3, let me point out a connection between Theorem 2.4 and my recent work with Marquès on hyperdoctrines and polyadic spaces [79]. We have a covariant functor $K\,\mathrm{Con}$ from the category of finitely presented algebras to the category of join-semilattices, which sends a homomorphism h to h^*. From the proof of Theorem 2.4, one obtains that the variety $\mathcal{V}$ has right uniform restrictions if, and only if, for every homomorphism h between finitely presented

[18] A list of my formalization activities is at https://www.samvangool.net/formal.html.

algebras, the pullback h^{-1} sends compact congruences to compact congruences. In other words, a variety $\mathcal{V}$ is coherent if, and only if, the functor $K\,\mathrm{Con}$ sends any homomorphism to a join-semilattice morphism which has an upper adjoint. Moreover, the variety $\mathcal{V}$ has interpolants if, and only if, for any injective homomorphism h, these adjoints satisfy the Beck-Chevalley condition. Adopting language of [79], a coherent variety with interpolants gives rise to a 'semilattice-valued' hyperdoctrine based on the category of finitely presented algebras in $\mathcal{V}$ with injective homomorphisms. As a small technical point, in order to clarify the link with categorical logic, it is in fact probably more opportune to consider the *meet-semilattice* $K\,\mathrm{Con}(A)^{\mathrm{op}}$ rather than the join-semilattice $K\,\mathrm{Con}(A)$. The property of $\mathcal{V}$ having subtractions, then, means that $K\,\mathrm{Con}(A)^{\mathrm{op}}$ carries a Heyting implication. The hope is that this line of thought would lead to a fibered version of the perspective taken in [73, Ch. 2], in which the regular subobjects in the opposite of the category of finitely presented algebras play a crucial role. I leave to future work a more in-depth exploration of this connection between uniform interpolants and hyperdoctrines.

In Section 2.3, I also described how propositional quantifiers and uniform interpolation relate to the existence of model companions, for instance, the existence of the model completion of the first-order theory of Heyting algebras [72]. As mentioned there, this result means that there is a first-order axiomatization of the class of existentially closed Heyting algebras. However, there is no concrete and direct understanding of what it means for a Heyting algebra to be existentially closed. This is because the naive way of extracting a concrete axiomatization out of the existence proof of [72] would go through Pitts' construction of propositional quantifiers, which does not admit an immediate intuitive interpretation (recall Fig. 2.3). A line of work due to Darnière and Junker analyzes the property of existential closedness in *subvarieties* of Heyting algebras, and makes interesting connections to the model theory of rings and p-adic geometry, see [43] and [42, Ch. 2]. It would be worthwhile to see if we can combine that work with our recently increased understanding of uniform interpolation to better understand, and axiomatize, what it means for a Heyting algebra to be existentially closed.

3 Temporal logic: Model companions and unification

We will then want to *isolate* general properties of the programs (*defined* already with the aid of the machine model) in order to *organize* our *deductions* more clearly (the so-called "axiomatic" method). At one level of abstraction this has already been illustrated above. To really carry out the proposal for the "real life" situation is a big, big "program".

– *J. de Bakker & D. Scott* [15, p. 30]

Alas, conditioned states are impermanent, subject to arising and decaying.

– *The Sūtra on Impermanence* [45]

Temporal logics allow for reasoning about situations that change over time. One of the simplest temporal operators is denoted X, pronounced 'next', and added as a unary operation to the propositional language considered in Chapter 2, now usually with the rules of Boolean, rather than intuitionistic, logic. Formulas of this enriched language can be evaluated as subsets of a *transition system*, by which we mean a set S of *states* equipped with a binary *one-step transition* relation R. The semantics of $\mathrm{X}p$ is defined to be the set of states from which one can access, via R, a state in which p holds.[1]

One may enrich the temporal language further with operators derived from this basic operation X via fixpoint definitions. For example, the operator 'future', denoted F, can be defined by declaring, for any property p, that $\mathrm{F}p$ be the least fixed point of the function which sends a property x to $p \vee \mathrm{X}x$. The intended semantics is that the property $\mathrm{F}p$ should hold at a state s if, and only if, there exists a finite R-path to a state t where p holds; that is, the temporal operator F should correspond to the reflexive transitive closure of the one-step transition relation R. However, in general, it can be challenging to prove that the natural fixpoint axioms are indeed complete with respect to these intended semantics (see Section 3.1 below). The extension of propositional logic by X and *any* fixed point formula expressible in terms of Boolean operations and X is referred to as the propositional μ*-calculus* [15, 109], where μ is the common notation for least fixed point recursion. For instance, the above definition of F is expressed in the μ-calculus as $\mathrm{F}p \overset{\text{def}}{=} \mu x.(p \vee \mathrm{X}x)$.

We will here be interested in a number of *equational* theories that are fragments of the μ-calculus. Formally, we consider extensions of the basic algebraic type of Boolean algebras by one or more operations and a number of equations for them. We then consider questions about the *decidability* and *completeness* of these theories. More specifically, in this chapter I will describe my contributions to:

[1] Note that we do not a priori assume X to be deterministic, i.e., for us, a state in a transition system may have multiple 'next' states, or none at all. In this non-deterministic setting, the literature also uses, rather than X, the symbols '$\Diamond$' or '$\langle a \rangle$', where a ranges over an alphabet of 'basic actions'. I use X here so as not to introduce too many different notations, and because, below, we will often restrict to a deterministic setting, but, when we do, I will say so explicitly.

1. Model-complete extensions for equational theories of *linear temporal logic* and *fair computational tree logic* (Section 3.1);

2. Decidability of the *unifiability problem* for the deterministic logic of next, via an algorithm on *de Bruijn graphs* (Section 3.2).

3.1 Model-complete extensions of temporal theories

The decidability of monadic second-order logic on natural numbers with successor, originally due to Büchi [28], makes use of a conversion between logic and automata. The key idea, which we also encountered in Section 1.1, is to view interpretations of unary predicates over natural numbers as infinite words over a suitable alphabet. One can then associate an automaton with a formula, and vice versa, in such a way that the automaton accepts exactly those infinite words that, viewed as interpretations of second-order variables, satisfy the formula. Converting a formula ϕ into an automaton A_ϕ and then the automaton A_ϕ again into a formula, one does not get back the same formula ϕ, but a formula ϕ' which is equivalent to ϕ in the intended model of the natural numbers. One may view ϕ' as a 'normalization' of ϕ. Morally, ϕ' is an existential formula; although this is not formally true in Büchi's construction, the 'existential nature' of ϕ' is rather evident. In order to make ϕ' into an actual existential formula, an enlargement of the language is needed. This enlargement is naturally obtained by adding temporal operators to the algebraic type of the Boolean algebra $\mathcal{P}(\omega)$.

The work I report on in this section[2] makes the observations of the previous paragraph precise. This allows us to fit monadic second-order logic and temporal logic into the framework of modern model-theoretic algebra, using the concepts of model-completeness and model companions already introduced in Section 2.3.

Algebras for linear temporal logic. We recall the basic theory of algebraic structures for linear temporal logic, comparable to the one given in, e.g., [76, §8]. These structures play the same role for linear temporal logic as Heyting algebras played for intuitionistic logic in Chapter 2. The foundational concept for us here is that of a deterministic modal algebra.

Definition 3.1. A *modal algebra* is a structure $(A, \vee, \neg, \bot, \mathrm{X})$ such that $(A, \vee, \neg, \bot)$ is a Boolean algebra, and $\mathrm{X} : A \to A$ is a function that preserves finite joins, i.e., $\mathrm{X}(a \vee b) = \mathrm{X}a \vee \mathrm{X}b$ for every $a, b \in A$, and $\mathrm{X}\bot = \bot$. A modal algebra is *deterministic* if, moreover, $\mathrm{X}(\neg a) = \neg \mathrm{X}a$ for every $a \in A$. In other words, a deterministic modal algebra is a Boolean algebra equipped with an endomorphism.

If R is a binary relation on a set S, then the Boolean algebra $\mathcal{P}(S)$ admits a modal algebra structure $\mathrm{X}_R : \mathcal{P}(S) \to \mathcal{P}(S)$, defined, for $U \in \mathcal{P}(S)$, by

$$\mathrm{X}_R(U) \overset{\text{def}}{=} R^{-1}[U] = \{s \in S \mid \text{ there exists } t \in U \text{ such that } sRt\} .$$

The *Jónsson-Tarski representation theorem* [105] implies that every modal algebra A embeds into a modal algebra of the form $\mathbb{P}(S, R) \overset{\text{def}}{=} (\mathcal{P}(S), \cup, (-)^{\mathrm{c}}, \emptyset, \mathrm{X}_R)$, for some transition system (S, R). Indeed, given a modal algebra A, take S_A to be the set of ultrafilters of A, and take R_A to be the binary relation

[2] The work described in this section was first published as [70, 71], joint with S. Ghilardi.

on S defined, for any $x, y \in S_A$, by $xR_{\mathrm{X}}y$ if, and only if, for any $a \in y$, $\mathrm{X}a \in x$. Then A admits an injective homomorphism of modal algebras into $\mathbb{P}(S_A, R_A)$, called the *canonical extension* of A. For a detailed proof, see, e.g., [21, Sec. 5.3]. The Jónsson-Tarski theorem may be seen as a consequence of extended Stone duality for Boolean algebra with operators, also see [60, Sec. 4.4]. One may further prove that a modal algebra of the form $\mathbb{P}(S, R)$ is deterministic, if, and only if, the relation R is the graph of a total function, i.e., for every $s \in S$ there exists a unique $t \in S$ such that sRt. This is an instance of a more general theory of *correspondence* between modal axioms and first-order properties on frames, see, e.g., [21, Ch. 3].

The Jónsson-Tarski theorem can be used to prove a *completeness theorem for modal logic*. Below, we will extend this proof method in a number of directions, so we briefly recall how this works. A *modal formula* is, by definition, a term in the algebraic type of modal algebras, and the completeness theorem states that, if a modal formula ϕ is *consistent* (meaning that the sentence $\forall \bar{p}.\phi(\bar{p}) = \bot$ is not derivable in the theory of modal algebras), then there exist a transition system (S, R) and a valuation $V : \bar{p} \to \mathcal{P}(S)$ so that the evaluation of ϕ in the modal algebra $\mathbb{P}(S, R)$ is non-empty. To prove this theorem, if $\phi(\bar{p})$ is a consistent modal formula, note that $\phi \neq \bot$ in the free modal algebra over $\bar{p}$. The Jónsson-Tarski theorem gives an embedding e of this free modal algebra into a modal algebra of the form $\mathbb{P}(S, R)$. Taking $V(p) \overset{\text{def}}{=} e(p)$ for each propositional variable p, it follows that ϕ evaluates to $e(\phi)$, which is non-empty because e is injective and $\phi \neq \bot$.

In Definition 3.2, we extend the definition of deterministic modal algebra to include a 'Future' and 'Initial' operator. The idea of this definition is to capture in an equational way the theory of the familiar *linear* transition system based on ω with the successor function. The corresponding deterministic modal algebra $\mathcal{P}(\omega)$ carries as its 'Next' operation $\mathrm{X}U \overset{\text{def}}{=} \{t \in \omega \mid t + 1 \in U\}$. To make this into a linear temporal algebra in the sense of Definition 3.2, one defines, for any $U \in \mathcal{P}(\omega)$, $\mathrm{F}U \overset{\text{def}}{=} {\downarrow}U$, where $\downarrow$ is downward closure with respect to the order on ω, and $\mathrm{I} \overset{\text{def}}{=} \{0\}$.

Definition 3.2. A *linear temporal algebra*[3] is a deterministic modal algebra equipped with a further unary operation $\mathrm{F} : A \to A$ and an element $\mathrm{I} \in A$ such that

1. for any $a \in A$, $\mathrm{F}a = a \vee \mathrm{XF}a$, and, for any $b \in A$, if $a \vee \mathrm{X}b \leq b$, then $\mathrm{F}a \leq b$;
2. $\mathrm{XI} = \bot$ and, for any $a \in A$, if $a \neq \bot$ then $\mathrm{I} \leq \mathrm{F}a$.

We note that both properties in Definition 3.2 can be written as *universal* sentences in first-order equational logic over the algebraic type of Boolean algebras with two additional unary operations X, F and one additional constant I. We denote by LT the universal first-order theory of linear temporal algebras. Since $\mathcal{P}(\omega)$, as defined above, is a linear temporal algebra, the full first-order theory $\mathsf{Th}(\mathcal{P}(\omega))$ of sentences true in this algebra is an extension of the universal theory LT. Our main theorem about linear temporal algebras is the following.

Theorem 3.3. *The theory* $\mathsf{Th}(\mathcal{P}(\omega))$ *is the model companion of* LT.

We may view the first-order theory of $\mathcal{P}(\omega)$ as the monadic second-order theory of ω. Thus, in a slogan, Theorem 3.3 says that 'monadic second-order logic on ω is the model companion of linear temporal logic'.

[3] Note that we omit the common 'Until' operator of linear temporal logic here, so strictly speaking we only work with the 'Next-Future-Initial' fragment of LTL. We do introduce an 'Until'-type operator for the case of tree logic below.

In order to establish Theorem 3.3, we need to prove two parts: (1) Any universal sentence in $\mathsf{Th}(\mathcal{P}(\omega))$ is already in LT (Corollary 3.8); (2) The theory $\mathsf{Th}(\mathcal{P}(\omega))$ is model-complete (Theorem 3.9). Part (1) boils down to proving a completeness theorem for linear temporal algebras with respect to the transition system ω, as we will see shortly. For part (2), we will use the correspondence between monadic second-order logic on ω and automata on infinite words.

Completeness of axiomatization of linear temporal algebras. A point that distinguishes linear temporal algebras from the deterministic modal algebras defined above is that the class of linear temporal algebras is not *canonical*, meaning that there exists a linear temporal algebra A whose canonical extension is no longer a linear temporal algebra, see Example 3.4. This makes the proof of completeness more involved than for modal algebras.

Example 3.4. Consider the linear temporal algebra $\mathcal{P}(\omega)$ defined above, and let A be the collection of finite subsets of ω and their complements (called *cofinite* subsets). One verifies that A is a subalgebra of $\mathcal{P}(\omega)$, and thus a linear temporal algebra in its own right. The ultrafilters of A are in bijection with the ordinal $\omega + 1$: Each $n \in \omega$ gives an ultrafilter $\{u \in A \mid n \in u\}$, and there is one additional ultrafilter $\infty \overset{\text{def}}{=} \{u \in A \mid u \text{ is cofinite}\}$. Thus, the canonical extension of A, as a Boolean algebra, is $\mathcal{P}(\omega + 1)$, with the embedding sending any $a \in A$ to the set of ultrafilters that contain a. The successor function R_A^{X} dual to the operator X sends ∞ to itself, and the relation dual to the operator F is the usual order on $\omega + 1$. We claim that the second part of the fixed point axiom (1) in Definition 3.2 no longer holds in the modal algebra $\mathcal{P}(\omega + 1)$. Indeed, take $a \overset{\text{def}}{=} \{\infty\}$ and $b \overset{\text{def}}{=} \{\infty\}$. The operation X on the canonical extension $\mathcal{P}(\omega + 1)$ satisfies $\mathrm{X}\{\infty\} = (R_A^{\mathrm{X}})^{-1}(\infty) = \{\infty\}$, and we get $a \vee \mathrm{X}b = b$. However, $\mathrm{F}\{\infty\} = \top$, and thus $Fb \not\leq b$.

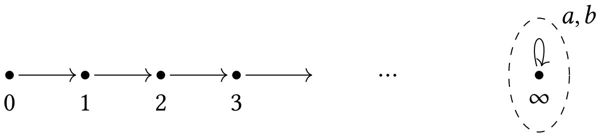


Figure 3.1: The set of ultrafilters of the algebra A in Example 3.4, with the relation R_A^{X} drawn as arrows and the sets a, b as the dashed ellipse.

In topological terms, the problem in Example 3.4 arises because $\{\infty\}$ is a closed, non-open set of the dual space $\omega + 1$ of A, and the axioms of the linear temporal algebra A are only assumed to hold for clopen subsets of the dual space. By involving topology, as we will do now, one may recover a full representation of linear temporal algebras.

Definition 3.5. A *linear temporal space* is a tuple $(S, s, \leq, x_0)$, where S is a Boolean topological space, $\leq$ is a preorder on S, $s : S \to S$ is a continuous function, and $x_0 \in S$ is a point such that, for any $x, y \in S$ and clopen $K \subseteq S$,

1. $\{x_0\}$ is clopen,
2. $\uparrow x$ is closed,
3. $\downarrow K$ is clopen,
4. $x \leq s(x)$,
5. if $x < y$ then $f(x) \leq y$,
6. $x_0 \leq x$,
7. $s(x) \neq x_0$, and
8. if $s^{-1}(K) \subseteq K$ then $\downarrow K \subseteq K$,

The *dual algebra* of $(S, s, \leq, x_0)$ is the tuple $(A, \mathrm{X}, \mathrm{F}, \mathrm{I})$, where A is the Boolean algebra of clopen subsets of S, and for any $K \in A$, $\mathrm{X}K := s^{-1}(K)$, $\mathrm{F}K := {\downarrow}K$, and $\mathrm{I} := \{x_0\}$.

Proposition 3.6 ([71, Thm. 3.5])**.** *For every linear temporal algebra, there exists a linear temporal space whose dual algebra is isomorphic to A.*

In the proof of Proposition 3.6, if A is a linear temporal algebra, we consider the ultrafilter space S_A, with s the function dual to X, $\leq$ the preorder dual to F, and x_0 the unique ultrafilter containing the atom I. For an appropriate notion of isomorphism, this is in fact the up to isomorphism *unique* linear temporal space with dual algebra A. One could extend this object correspondence to a dual equivalence of categories of linear temporal algebras and linear temporal spaces, with appropriate morphisms, but we do not need this in what follows. Proposition 3.6 is an essential building block towards proving the completeness of our axioms with respect the intended semantics based on ω:

Theorem 3.7 ([71, Thm. 3.3])**.** *If a term t evaluates to $\top$ under any valuation in the linear temporal algebra $\mathcal{P}(\omega)$, then t evaluates to $\top$ under any valuation in any linear temporal algebra.*

I give a rough sketch of the proof of Theorem 3.7, showing the similarity with the proof of completeness for modal algebras, referring to [71, Sec. 3] for all the details. Reasoning by contraposition, assume that $t(\bar{p}) \neq \top$ under some valuation V of the variables $\bar{p}$ in some linear temporal algebra A. We use Proposition 3.6 to choose a linear temporal space $(S, s, \leq, x_0)$ whose dual algebra is isomorphic to A. The assumption on the valuation t gives, for each variable p_i, a clopen subset K_i of S, and hence a clopen set $[\![t(\bar{K})]\!]$ in S with a point x outside it. From these data, applying standard filtration techniques from modal logic and a fine analysis of linear temporal spaces, we eventually obtain a valuation V' of the variables $\bar{p}$ in $\mathcal{P}(\omega)$ under which t does not evaluate to $\top$.

Corollary 3.8. *If a universal first-order sentence ϕ holds in $\mathcal{P}(\omega)$, then it holds in any linear temporal algebra.*

In order to deduce Corollary 3.8 from Theorem 3.7, it suffices to rewrite any universal first-order sentence ϕ, in the first-order theory of linear temporal algebras LT, as $\forall \bar{p}.t_\phi(\bar{p}) = \top$, for some term t_ϕ. We show how to do this in [71, Lem. 3.2], using basic facts about Boolean algebra, together with the observation that, for any term t, a negated equality $t \neq \top$ is in fact equivalent in LT to an equality, namely, $(\mathrm{I} \Rightarrow \mathrm{F}\neg t) = \top$. (Recall that, in a Boolean algebra, $a \Rightarrow b$ is a macro for $\neg a \vee b$.)

Model-completeness of the theory of the LTL-algebra $\mathcal{P}(\omega)$. I will now outline the proof of:

Theorem 3.9. *The first-order theory of the linear temporal algebra $\mathcal{P}(\omega)$ is model-complete.*

Let ϕ be a first-order formula in the algebraic type of linear temporal algebras. We need to show that ϕ is equivalent to an existential formula ϕ' in $\mathsf{Th}(\mathcal{P}(\omega))$. Our proof proceeds according to the following scheme.

1. Syntactically transform ϕ into a formula Φ of monadic second-order logic on ω-words.
2. Associate to the formula Φ a Büchi automaton $\mathcal{A}_\Phi$.
3. Associate to the Büchi automaton $\mathcal{A}_\Phi$ a linear temporal algebra term t and a formula ϕ' that encode its behavior.

Throughout the proof, we use a correspondence between *words on* ω and *valuations in* $\mathcal{P}(\omega)$. Let $\bar{p} = (p_1, \dots, p_n)$ be the propositional variables free in ϕ, and fix $\bar{P} = (P_1, \dots, P_n)$ a sequence of monadic second-order variables of the same length. In the same way as in Section 1.1 (p. 4), a *word on* ω *in alphabet* $2^{\bar{P}}$, i.e., a function $w : \omega \to 2^{\bar{P}}$, corresponds to a unique valuation $V_w : \bar{p} \to \mathcal{P}(\omega)$. Indeed, we define $V_w(p_i) \overset{\text{def}}{=} \{t \in \omega \mid \text{ the bit of } w \text{ at position } t \text{ at index } P_i \text{ is } 1\}$.

For step 1 in the above outline, we use the *standard translation* from modal logic, see, e.g., [21, Def. 2.45]. When $t(\bar{p})$ is a term in the algebraic type of linear temporal algebras, we define a monadic second-order formula $\dot{t}(\bar{P})$, in such a way that a word $w : \omega \to 2^{\bar{P}}$ is in the semantics of the formula $\dot{t}(\bar{P})$ if, and only if, w is in the set $t(V_w(p_1), \dots, V_w(p_n))$, that is, the result of evaluating t in the linear temporal algebra $\mathcal{P}(\omega)$ under the assignment V. We extend the translation to assign, to any first-order formula $\phi(\bar{p})$ in the algebraic type of linear temporal algebras, a monadic second-order formula Φ, in such a way that, for any $w \in (\Sigma_{\bar{P}})^\omega$, we have

$$\mathcal{P}(\omega), V_w \vDash \phi \text{ if, and only if, } w \in [\![\Phi]\!]. \tag{3.1}$$

For step 2, we make use of the following result of Büchi [28] for ω*-words*, analogous to the result already cited as Theorem 1.3 in Chapter 1.

Proposition 3.10. *For any monadic second-order formula* $\Phi(\bar{P})$*, there exists a finite non-deterministic automaton* $\mathcal{A}_\Phi$ *on the finite alphabet* $\Sigma_{\bar{P}} = 2^{\bar{P}}$ *such that*

$$[\![\Phi]\!] = \{w \in (\Sigma_{\bar{P}})^\omega \mid \mathcal{A}_\Phi \textit{ has a run on } w \textit{ that visits a final state infinitely often.}\} \tag{3.2}$$

A run of an automaton A on an ω-word w is called *Büchi-accepting* if it visits a final state infinitely often. Detailed proofs of Proposition 3.10 can be found in standard references on the connection between monadic second-order logic and automata, e.g., [169, Thm. 5.9] or [91, Thm. 12.15].

For step 3, we show how to encode the behavior of a Büchi automaton as a linear temporal algebra term. Let $\mathcal{A}$ be any finite non-deterministic automaton on the finite alphabet $\Sigma_{\bar{P}}$. Let us write $\{q_0, \dots, q_m\}$ for the set of states of $\mathcal{A}$, where q_0 is the initial state. Write F for the set of final states of $\mathcal{A}$, and, for any $0 \le i \le m$ and $a \in \Sigma_{\bar{P}}$, let $\delta(q_i, a)$ denote the set of states accessible from q_i when reading the letter a. We now define the following linear temporal algebra terms with variables in $\{p_1, \dots, p_n\} \cup \{q_0, \dots, q_m\}$:[4]

$$\circ a \overset{\text{def}}{=} \textstyle\bigwedge_{i\,:\,a_i=1} p_i \wedge \bigwedge_{i\,:\,a_i=0} \neg p_i \text{ (for } a \in \Sigma_{\bar{P}}\text{)}, \qquad \mathsf{Part} \overset{\text{def}}{=} \textstyle\bigvee_{i=0}^{m} (q_i \wedge \bigwedge_{j \neq i} \neg q_j),$$

$$\mathsf{Init} \overset{\text{def}}{=} \mathrm{I} \Rightarrow q_0, \qquad \mathsf{Accept} \overset{\text{def}}{=} \textstyle\bigvee_{q_i \in F} \mathrm{F} q_i,$$

$$\mathsf{Trans} \overset{\text{def}}{=} \textstyle\bigwedge_{i=0}^{m} \left[q_i \Rightarrow \left(\bigvee_{q_j \in \delta(q_i, a)} (\circ a \wedge \mathrm{X} q_j)\right)\right], \qquad t_{\mathcal{A}} \overset{\text{def}}{=} \mathsf{Part} \wedge \mathsf{Init} \wedge \mathsf{Trans} \wedge \mathsf{Accept}.$$

These terms describe the behavior of the automaton $\mathcal{A}$, in the sense that, for any word $w \in (\Sigma_{\bar{P}})^\omega$,

$$\text{there exists a Büchi-accepting run of } \mathcal{A} \text{ on } w \text{ if, and only if, } \mathcal{P}(\omega), V_w \vDash \exists \bar{q}.\, t_{\mathcal{A}} = \top. \tag{3.3}$$

Indeed, any valuation $U : \{q_0, \dots, q_m\} \to \mathcal{P}(\omega)$ can be seen as a potential run on the word w, where $U(q_i)$ denotes the set of times in ω where the run is in state q_i. Then, under the valuation U, the

[4] Note that the symbol 'q_i' is used both for a state of $\mathcal{A}$ and for a variable in the linear temporal algebra terms.

term Part evaluates to $\top$ in the linear temporal algebra $\mathcal{P}(\omega)$ if the run is in exactly one state at each time, Init evaluates to $\top$ if w is in the initial state at time 0, Trans evaluates to $\top$ if the run follows the transitions of $\mathcal{A}$, and Accept evaluates to $\top$ if the run visits a final state infinitely often.

Combining Eq. (3.1), Eq. (3.2), and Eq. (3.3), we conclude that the formula $\phi' \overset{\text{def}}{=} \exists \bar{q}.\ t_{\mathcal{A}_\Phi} = \top$ is an existential formula equivalent to ϕ in $\mathsf{Th}(\mathcal{P}(\omega))$, as required for Theorem 3.9.

Extensions to trees. In [70], we extend the above results to temporal logics on trees. The proof follows largely a similar scheme as the proof outlined above, although some of the technical details are much more intricate, especially in the part where we prove the completeness of the axiomatization. I will here only briefly give the main statement and highlight the main difficulty compared to what we already discussed above.

The characteristic difference between temporal logic on trees vs. temporal logic on words is an additional layer of *non-determinism*, namely in the transition systems used for evaluating formulas. We would like to use, as above, tree automata to express arbitrary formulas of monadic second-order logic on trees in an existential way. The automata that we employ were introduced by [103] in the context of the propositional μ-calculus. We briefly recall the basic definitions that we need.

By a *tree* we mean a transition system (S, R, s_0), where R is a binary relation on S and $s_0 \in S$ is such that, for every node $s \in S$, there is a unique finite R-path from s_0 to s. When $\bar{p}$ is a finite set of variables, we call a function $w \colon S \to 2^{\bar{p}}$ a *$\bar{p}$-coloring* of S. As in the case of ω-words above, $\bar{p}$-colorings of a tree correspond bijectively to valuations $\bar{p} \to \mathcal{P}(S)$. A *branch* in the tree is an infinite R-path.

A *tree automaton*[5] consists of a finite set of states Q, an initial state q_0, a function $\Omega \colon Q \to \omega$, and a function δ which associates, to any state $q \in Q$ and letter $a \in 2^{\bar{p}}$, a set $\delta(q, a) \subseteq \mathcal{P}(Q)$. Let (S, R, s_0) be a tree with $\bar{p}$-coloring $w \colon S \to 2^{\bar{p}}$. A *run* of a tree automaton on this $\bar{p}$-colored tree is a function $r \colon S \to Q$ with the property that $r(s_0) = q_0$, for any $s \in S$, the set $\{r(s') \mid s' \in R[s]\}$ is in $\delta(r(s), c(s))$, and for any infinite branch $(s_t)_{t=0}^{\infty}$ in S,

$$\pi((s_t)_{t=0}^{\infty}) \overset{\text{def}}{=} \min\{\Omega(q) \mid r(s_t) = q \text{ for infinitely many } t \in \omega\} \text{ is even.} \tag{3.4}$$

The last condition is referred to as a *parity acceptance condition* and goes back to Mostowski [130], who gave these conditions as a normal form for the tree automata originally introduced by Rabin [145].

The crucial observation for our work here is that the parity acceptance condition can be expressed in terms of a temporal logic operator on trees, that we call AF, following common acronyms from the literature on computational tree logic with path quantifiers [50], where AF stands for: "on All branches, at some Future point." Indeed, note that a run $r \colon S \to Q$ satisfies condition Eq. (3.4) precisely when, for every infinite branch $(s_t)_{t=0}^{\infty}$ in S and every state q:

$$\begin{aligned}&\text{If } \Omega(q) \text{ is odd and } r(s_t) = q \text{ infinitely often, then there exists } q' \text{ with } \Omega(q') \text{ even}, \Omega(q') < \Omega(q),\\ &\text{and, for every } t, \text{ there is } t' > t \text{ such that } r(s_{t'}) = q'\ .\end{aligned}$$

We introduce a binary temporal operator $\mathrm{AF}(a, b)$, whose intended semantics is that $\mathrm{AF}(a, b)$ holds in

[5] The more usual definition of acceptance by a tree automaton (also called μ-automaton) uses an acceptance game. The definition we give here is technically only equivalent to the usual definition in the case of trees that are so called *ω-expansions*. Since every tree is bisimilar to an ω-expansion, and we will only need to consider trees up to bisimilarity here, this definition suffices, see [2, Lem. 2.4 & Thm. 2.5].

a node s of a tree S if, and only if, for every infinite branch starting from s, if the property $\neg b$ holds infinitely often on the branch, then there exists a point on the branch where a holds. This operator AF allows us to express the parity condition Eq. (3.4) by saying that a number of terms of the form $\mathrm{AF}(\bigvee U, \neg q)$ evaluate to $\top$ under a valuation corresponding to the run r, analogous to the condition Accept in the case of ω-words discussed above. The model-completeness part of our proof then goes through relatively easily [70, Sec. 4].

However, to achieve a result analogous to Theorem 3.3 for trees, we now need to axiomatize the new temporal binary operator AF. We start from a modal algebra (A, X), and we also assume $\mathrm{X}\top = \top$, with intended meaning that 'every node has at least one successor'. We first introduce an auxiliary binary operator EU ('Exists Until'), with intended semantics: $\mathrm{EU}(a, b)$ holds in a node s if, and only if, there is a finite path from s to a node t where a holds, such that b holds along the entire path (but possibly not at t). In the μ-calculus, such an operator $\mathrm{EU}(a, b)$ can be defined as

$$\mathrm{EU}(a, b) \stackrel{\text{def}}{=} \mu x.\, a \vee (b \wedge \mathrm{X}x)\ ,$$

a variation on our earlier fixed point definition of F, now with an additional parameter b. Given the binary operator EU, we define a binary operator EG as the greatest fixed point

$$\mathrm{EG}(a, b) \stackrel{\text{def}}{=} \nu y.\, a \wedge \mathrm{X}(\mathrm{EU}(b \wedge y, a))\ .$$

The intended semantics of $\mathrm{EG}(a, b)$ is that it holds in a node s if, and only if, there exists an infinite path along which a always holds, and b holds infinitely often. Finally, the operator AF is just defined as the De Morgan dual of EG, i.e., $\mathrm{AF}(a, b) \stackrel{\text{def}}{=} \neg\mathrm{EG}(\neg a, \neg b)$, which then indeed has semantics mentioned above, assuming the semantics of EG.

We may now formalize the above ideas to define a quasi-equational theory that we call *fair* tree logic, reflecting the common terminology that a 'fairness' side condition states that a certain property is sastified infinitely often. We recall the definition (the corresponding theory is denoted CTL^f_I in [70]).

Definition 3.11. A *fair tree logic algebra* is a tuple $(A, \mathrm{X}, \mathrm{EU}, \mathrm{EG}, \mathrm{I})$, where (A, X) is a modal algebra with $\mathrm{X}\top = \top$, EU and EG are binary operations on A satisfying, for any $a, b, c \in A$:

$$\begin{aligned}
&a \vee (b \wedge \mathrm{XEU}(a, b)) \leq \mathrm{EU}(a, b)\ ,\\
&\text{if } a \vee (b \wedge \mathrm{X}c) \leq c,\ \text{ then } \mathrm{EU}(a, b) \leq c\ ,\\
&\mathrm{EG}(a, b) \leq a \wedge \mathrm{XEU}(b \wedge \mathrm{EG}(a, b), a)\ ,\\
&\text{if } c \leq a \wedge \mathrm{XEU}(b \wedge c, a),\ \text{ then } c \leq \mathrm{EG}(a, b)\ .
\end{aligned}$$

and I is an element of A such that $\mathrm{I} \neq \bot$, $\mathrm{XEU}(\mathrm{I}, \top) = \bot$, and, for every a, if $a \neq \bot$ then $\mathrm{I} \leq \mathrm{EU}(a, \top)$.

Theorem 3.12 ([70, Thm. 4.9])**.** *The theory of fair tree logic algebras has a model companion.*

As before, the proof of Theorem 3.12 separates into two parts: First, the completeness of fair tree logic with respect to its intended semantics; Second, a proof of model-completeness via automata. I have explained above the essential ingredient for the second part, namely how the operator AF allows to express the acceptance condition of automata with an existential formula. For the first part, we prove the following theorem, analogous to Theorem 3.7, but more difficult to establish:

Theorem 3.13. *If a term t evaluates to $\top$ under any valuation in $\mathcal{P}(S)$, with S a tree, then t evaluates to $\top$ in any fair tree logic algebra.*

The proof of this theorem is in [70, Sec. 3], and combines the idea of Jónsson-Tarski representation used in the linear case with an ad hoc, rather combinatorial, tableau construction. We also establish in [70, Thm. 4.15] a variant of Theorem 3.12 for *binary* trees is also established. In that case, the model companion is precisely the first-order theory of $\mathcal{P}(2^*)$, where 2^* is the full binary tree. Analogous to the slogan mentioned after Theorem 3.3, this result may be summarized as, 'monadic second-order logic on binary trees is the model companion of fair binary tree logic'.

3.2 De Bruijn graphs and unification for deterministic next

In this section, we will study *unification* for temporal logics.[6] We first state the general problem for an arbitrary class of algebras, using the notations introduced in Section 2.3. Let $\mathcal{K}$ be a class of algebras. A *unification instance* is a pair of terms (s, t) in the algebraic type of $\mathcal{K}$, also sometimes denoted $s \approx t$. Let (s, t) be a unification instance, and denote by $\bar{x} = (x_1, \dots, x_n)$ the variables occurring in s or t. A *unifier* of the instance (s, t) is a sequence of terms $\bar{u} = (u_1, \dots, u_n)$ such that $\vDash_{\mathcal{K}} s[x_i \mapsto u_i] \approx t[x_i \mapsto u_i]$.

Definition 3.14. The *unifiability problem* for a class of algebras $\mathcal{K}$ is the following computational problem: Given as input a unification instance (s, t), output a unifier of (s, t), or output 'impossible' if none such exists.

Since unifiers can be enumerated, the problem is clearly semi-decidable, and the first computationally interesting question is how to determine when a unification instance (s, t) does *not* admit a unifier. Logic provides a rich class of unifiability problems. When $\mathcal{K}$ is the class of Boolean algebras, any unifiable problem always has, up to equivalence, an effectively computable *most general* unifier, see, e.g., [122, Sec. 3]. This classical result admits generalizations to intuitionistic propositional logic and *transitive* modal logics and, in many cases, leads to a decision procedure for the unifiability problem [66, 68]. A major open question in the area is whether or not unifiability is decidable for the class of modal algebras. The existing methods, typically based on the idea of most general unifiers, are known to fail here [104]. Moreover, the problem appears to be right on the edge of decidability, as unifiability is known to be undecidable for a slight extension of modal algebras, when enriched with a so-called universal modality [179]. For a survey of unification problems in modal logic, with connections to description logic, we refer to [13], and we also refer to the introductions of [16, 17] for more information about recent progress in unification for modal logic.

In this section, we outline an approach to unifiability problems for varieties of modal algebras, and we use it to show decidability of the unifiability problem for the variety of *deterministic* modal algebras with an arbitrary number of constants. A combination of *duality* and *step-by-step constructions* gives us a useful perspective on unifiability problems for modal logics beyond the transitive. Both ingredients were already present in the ground-breaking series of papers on unification in transitive modal and intuitionistic logics [66, 68], and the duality perspective on unification has been made more explicit since then, for example in [18, 30, 67]. The step-by-step approach is an algebraic way of viewing

[6] This section is based on joint work with J. Marti that first appeared as an extended abstract in the workshop UNIF 2023 [81], and on ongoing joint work-in-progress with J. Marti and M. Sweering.

normal forms in modal logic, systematically studied in [65] and giving rise to finite model results in [129]. On the semantic side of the duality, this is similar to the *terminal sequence* for a coalgebra functor, see e.g. [19, 180], which has also been applied to prove finite model properties [134].

Unification and free algebras. For the rest of this section, let $\mathbf{V}$ denote an arbitrary variety of modal algebras.[7] We show how the unifiability problem can be phrased algebraically. First, we recall how to express unifiers as certain homomorphisms between finitely presented algebras [69, Sec. 3]. Recall that the free $\mathbf{V}$-algebra on n variables, $F_{\mathbf{V}}(x_1, \dots, x_n)$, can be realized as the quotient of the term algebra $T(x_1, \dots, x_n)$ under the equivalence relation that identifies two terms u and u' if, and only if, $\vDash_{\mathbf{V}} u \approx u'$; we write $[u]_{\mathbf{V}}$ for the class of the term u in $F_{\mathbf{V}}(x_1, \dots, x_n)$. By a slight abuse of notation, we also sometimes just denote this class by u. Now, given a unification instance (s, t) in variables $\bar{x} = (x_1, \dots, x_n)$, let $F_{\mathbf{V}}(\bar{x}; s \approx t)$ denote the quotient of the free algebra $F_{\mathbf{V}}(\bar{x})$ by the congruence generated by the pair $([s]_{\mathbf{V}}, [t]_{\mathbf{V}})$. We call two sequences of terms $(u_1, \dots, u_n)$ and $(u'_1, \dots, u'_n)$ *equivalent* if $\vDash_{\mathbf{V}} u_i \approx u'_i$ for every $1 \leq i \leq n$.

Lemma 3.15. *Let (s, t) be a unification instance in variables $\bar{x} = (x_1, \dots, x_n)$. The set of equivalence classes of unifiers for (s, t) in variables $\bar{y} = (y_1, \dots, y_m)$ is in bijection with the set of homomorphisms from $F_{\mathbf{V}}(\bar{x}; s \approx t)$ to $F_{\mathbf{V}}(\bar{y})$.*

Proof. By the universal property of the quotient, the set of homomorphisms $\bar{h} : F_{\mathbf{V}}(\bar{x}; s \approx t) \to F_{\mathbf{V}}(\bar{y})$ is in bijection with the set of homomorphisms $h : F_{\mathbf{V}}(\bar{x}) \to F_{\mathbf{V}}(\bar{y})$ such that $h([s]_{\mathbf{V}}) = h([t]_{\mathbf{V}})$. The latter set is in bijection with the set of equivalence classes of unifiers of (s, t), since two homomorphisms h and h' with domain $F_{\mathbf{V}}(\bar{x})$ are equal if, and only if, they agree on the set of generators $\bar{x}$. □

Second, we will show that we may restrict our attention to particular unification instances and potential unifiers. Note first that an instance (s, t) is unifiable if, and only if, the instance $(s \Leftrightarrow t, \top)$ is unifiable.[8] Thus, it suffices to consider instances of the form $(\phi, \top)$, for ϕ any term. With a slight abuse of notation, given a term $\phi(x_1, \dots, x_n)$, we call a sequence of terms $(u_1, \dots, u_n)$ a *unifier* for ϕ if $\vDash_{\mathbf{V}} \phi[x_i \mapsto u_i] \approx \top$. A unifier is called *ground* if it does not use any variables. Note that a term ϕ is unifiable if, and only if, ϕ is ground-unifiable. To prove the non-trivial direction, if $\bar{u}$ is a unifier for ϕ, then the sequence of terms $\bar{v} = (v_1, \dots, v_n)$, where v_i is obtained by substituting $\top$ (or, in general, any constant symbol) for each variable appearing in u_i, is again a unifier for ϕ; this follows immediately from the definition of $\vDash_{\mathbf{V}}$.

In light of these observations, we conclude that, for our variety of modal algebras $\mathbf{V}$, the computational problem of unifiability is equivalent to the following more algebraic problem.

Definition 3.16. The *algebraic unifiability problem* for $\mathbf{V}$ is the following computational problem: Given as input a term $\phi(\bar{x})$, output a homomorphism $F_{\mathbf{V}}(\bar{x}; \phi \approx \top) \to F_{\mathbf{V}}(\emptyset)$, or output 'impossible' if none such exists.

The interest of this reformulation of unifiability as an algebraic problem is that it allows us to dualize it into a *coalgebraic* problem, as we will do below in the specific case where $\mathbf{V}$ is the variety of deterministic modal algebras. More precisely, we will consider the unifiability problem *with constants*,

[7] Some results discussed here hold more generally, but this level of generality suffices for our intended application.

[8] The notation $s \Leftrightarrow t$ is shorthand for the term $(s \Rightarrow t) \wedge (t \Rightarrow s)$.

also sometimes called *parameters* in the literature. Algebraically, this means that we fix a natural number k, we enrich the algebraic type of $\mathbf{V}$ with k new constant symbols, $p_1, \dots, p_k$, and we define $\mathbf{V}^{(k)}$ to be the variety of algebras $(A, a_1, \dots, a_k)$, where A is an algebra in $\mathbf{V}$ and, for each $1 \leq i \leq k$, a_i is an element of A that interprets the constant symbol p_i. Then, the *unifiability problem with k constants* for $\mathbf{V}$ is, by definition, the unifiability problem for $\mathbf{V}^{(k)}$.

Let $\mathbf{X}$ denote the variety of deterministic modal algebras (Definition 3.1). Our main theorem is the following.

Theorem 3.17. *For any k, the unifiability problem with k constants for* $\mathbf{X}$ *is decidable.*

In order to solve the algebraic version of the unifiability problem given in Definition 3.16, our construction proceeds in two steps:

1. We show that the algebraic unifiability problem for $\mathbf{X}$ admits an exponential time reduction to a graph-theoretic problem on homomorphisms between *de Bruijn graphs*;
2. We show that the de Bruijn graph homomorphism problem is decidable in exponential time.

It will follow that our algorithm has a time complexity that is doubly exponential in the size of the input. Before proceeding with the general description of our algorithm, we give a few examples of unification instances in $\mathbf{X}$ with constants.

Example 3.18. Consider the system of temporal equations in variables x, y, with constant symbol p:

$$\begin{cases} x \approx \neg \mathrm{X}p \wedge \mathrm{XX}(x \vee y) \\ y \approx x \Rightarrow p \end{cases} \tag{3.5}$$

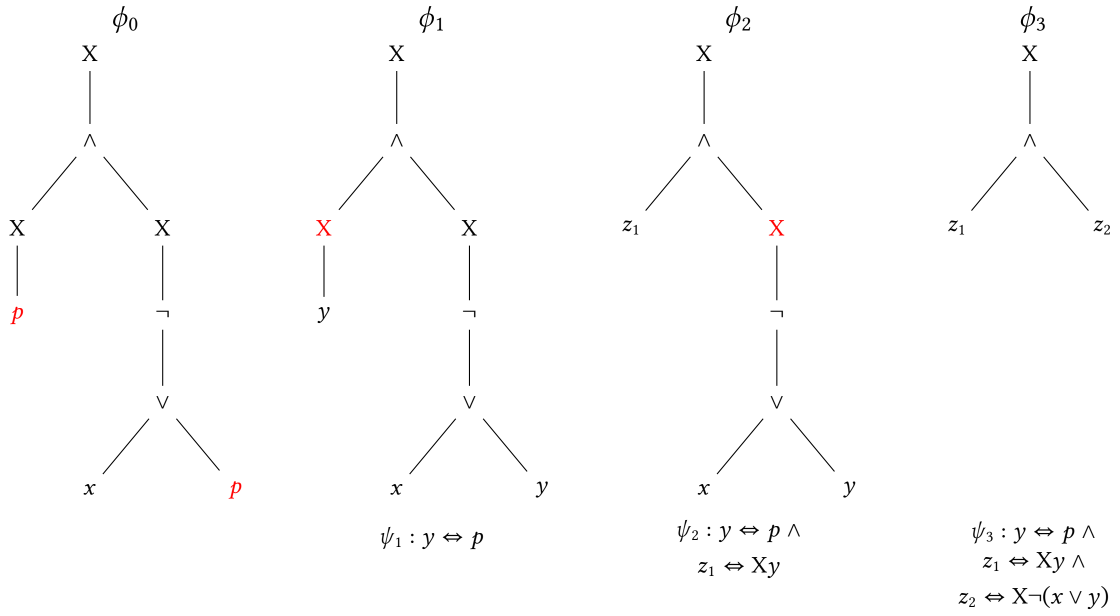
A substitution that unifies the two equations in Eq. (3.5) is the same thing as a substitution that makes the formula

$$[x \Leftrightarrow (\neg \mathrm{X}p \wedge \mathrm{XX}(x \vee y))] \wedge [y \Leftrightarrow (x \Rightarrow p)] \tag{3.6}$$

equivalent to $\top$. In this case, there turns out to be a unique such substition. One may deduce this syntactically by first noting that a unifier of $y \Leftrightarrow (x \Rightarrow p)$ must make $x \vee y$ equivalent to $x \vee (x \Rightarrow p)$, which is equivalent to $\top$. Thus, such a unifier must make $\neg \mathrm{X}p \wedge \mathrm{XX}(x \vee y)$ equivalent to $\neg \mathrm{X}p \wedge \mathrm{XX}\top$, which is in turn equivalent to $\neg \mathrm{X}p$. This means that the only possible substitution σ must send the variable x to $\neg \mathrm{X}p$, and y to $\neg \mathrm{X}p \Rightarrow p$. One then verifies that this is indeed a unifier. We give another interpretation of this example, via graph homomorphisms, in Example 3.27 below. For a negative example, the equation $x \approx \neg \mathrm{X}x$ does *not* have a unifier. We will also explain a proof of this in Example 3.27 below.

Flattening. We now show one further syntactic simplification that we can make on our unification instances. We call a term ϕ in the algebraic type of $\mathbf{X}^{(k)}$ *affine* if it is equivalent to a Boolean combination of variables, constants, and terms of the form $\mathrm{X}x_i$, for x_i a variable. A term is affine if, and only if, every variable occurs under the scope of at most one occurrence of X, and no constant occurs under the scope of X. For example, the term $(\neg p \vee \mathrm{X}x) \wedge \mathrm{X}\top$ is affine, while the terms $\mathrm{X}(p \vee x)$ and $\mathrm{X}(x \wedge \mathrm{X}\neg y)$ are not affine. We now set out to show that, for any $\mathbf{X}^{(k)}$-term ϕ, we can construct an affine term ϕ' such that the set of unifiers of ϕ is in bijection with the set of unifiers of ϕ'. We do this by a syntactic procedure called *flattening*. The term ϕ' will in general use more variables than ϕ, although no more than the number of constants plus the number of occurrences of X in ϕ. Let ϕ be an arbitrary $\mathbf{X}^{(k)}$-term. We first show the idea of the flattening transformation in an example.

Example 3.19. Consider the non-affine term $\phi_0 = \mathrm{X}(\mathrm{X}p \wedge \mathrm{X}\neg(x \vee p))$. We show how to successively transform it into an affine term, while preserving unifiers. (The process is illustrated as a transformation of syntax trees in Figure 3.2.) We first introduce one new variable y and define $\phi_1 \overset{\text{def}}{=} \mathrm{X}(\mathrm{X}y \wedge \mathrm{X}\neg(x \vee y))$ and $\psi_1 \overset{\text{def}}{=} p \Leftrightarrow y$. This ensures that no constants occur under the scope of X, and unifiers of $\phi_1 \wedge \psi_1$ are essentially the same thing as unifiers of ϕ_0, since they must send y to p. Now, the next 'problem' that prevents ϕ_1 from being affine is the subformula Xy, which is under the scope of another X. We introduce a variable z_1 and set $\phi_2 \overset{\text{def}}{=} \mathrm{X}(z_1 \wedge \mathrm{X}\neg(x \vee y))$ and $\psi_2 \overset{\text{def}}{=} \psi_1 \wedge (z_1 \Leftrightarrow \mathrm{X}y)$. Any unifier of ψ_2 must send z_1 to a formula that is equivalent to Xy, and thus, if it also unifies ϕ_2, then it unifies ϕ_1. Conversely, any unifier of $\phi_1 \wedge \psi_1$ extends uniquely to a unifier of $\phi_2 \wedge \psi_2$, by sending z_1 to the term obtained by applying the given unifier to Xy. The last problematic subformula in ϕ_2 is $\mathrm{X}\neg(x \vee y)$, so we define $\phi_3 \overset{\text{def}}{=} \mathrm{X}(z_1 \wedge z_2)$ and $\psi_3 = \psi_2 \wedge (z_2 \Leftrightarrow \mathrm{X}\neg(x \vee y))$. By the same argument as before, unifiers of $\phi_3 \wedge \psi_3$ are in bijection with unifiers of $\phi_2 \wedge \psi_2$. We note that $\phi_3 \wedge \psi_3$ is an affine term.



ϕ_0
X
∧
X
X
p
¬
∨
x
p
ϕ_1
X
∧
X
X
y
¬
∨
x
y
$\psi_1 : y \Leftrightarrow p$
ϕ_2
X
∧
z_1
X
¬
∨
x
y
$\psi_2 : y \Leftrightarrow p \wedge$
$z_1 \Leftrightarrow \mathrm{X}y$
ϕ_3
X
∧
z_1
z_2
$\psi_3 : y \Leftrightarrow p \wedge$
$z_1 \Leftrightarrow \mathrm{X}y \wedge$
$z_2 \Leftrightarrow \mathrm{X}\neg(x \vee y)$

Figure 3.2: Flattening a formula.

To formalize the idea of Example 3.19, let $\phi_0(\bar{x})$ be an arbitrary $\mathbf{X}^{(k)}$-term. We construct a sequence of terms $(\phi_r)_{r=0}^R$ and a sequence of affine terms $(\psi_r)_{r=0}^R$ in such a way that the unifiers of the term $\phi_r \wedge \psi_r$ are in bijection with the unifiers of $\phi_{r+1} \wedge \psi_{r+1}$, and moreover ϕ_R will be affine. Let $\phi_0 \overset{\text{def}}{=} \phi$ and $\psi_0 = \top$. First, we introduce a new variable y_j for each constant symbol p_j, and we define $\phi_1 \overset{\text{def}}{=} \phi[p_j \mapsto y_j]$, $\psi_1 \overset{\text{def}}{=} \bigwedge_{j=1}^n (y_j \Leftrightarrow p_j)$. Note that any unifier of ψ_1 must send the variable y_j to p_j, and therefore, if it is also a unifier of ϕ_1, then its restriction to $\bar{x}$ is a unifier of ϕ. Conversely, any unifier of ϕ extends to a unifier of $\phi_1 \wedge \psi_1$ by sending y_j to p_j. Further note that in $\phi_1 \wedge \psi_1$, no constant occurs under the scope of X, and we will maintain this property throughout the rest of the construction. Now, at stage $r \geq 1$, if ϕ_r is affine, we are done. Otherwise, there must be a variable in ϕ_r that is under the scope of more than one occurrence of X. Let θ be the maximal subterm of ϕ_r containing this variable occurrence and no occurrence of X. It follows that Xθ is also a subterm of ϕ_r. We define ϕ_{r+1} to be term obtained by replacing this subterm Xθ by a new variable z_r, and we let $\psi_{r+1} \overset{\text{def}}{=} \psi_r \wedge (z_r \Leftrightarrow \mathrm{X}\theta)$. Note that X$\psi$ is affine by construction, so that ψ_{r+1} remains affine. A unifier of $\phi_{r+1} \wedge \psi_{r+1}$ must send z_r to a term that is equivalent to Xθ. Therefore, its restriction to all the variables except z_r is a unifier of $\phi_r \wedge \psi_r$. Conversely, a unifier of $\phi_r \wedge \psi_r$ extends uniquely to a unifier of $\phi_{r+1} \wedge \psi_{r+1}$ by sending z_r to the result of applying the unifier to Xψ. Since ϕ_{r+1} is 'one step closer' to being affine than ϕ_r, this procedure eventually terminates.

We conclude that *we may assume that all unification instances are affine*, at the expense of introducing a number of fresh variables, which is linear in the size of the input formula.

The free deterministic modal algebra and de Bruijn graphs. Fix a finite sequence of constant symbols $p_1, \dots, p_k$, $k \geq 1$. For any $n \geq 0$, we will write $C_n \overset{\text{def}}{=} \{x_1, \dots, x_n, p_1, \dots, p_k\}$, where $x_1, \dots, x_n$ are n distinct variables. A natural first step in studying the unifiability problem for $\mathbf{X}^{(k)}$ is to describe the free algebra $F_{\mathbf{X}^{(k)}}(\bar{x})$, for $\bar{x} = (x_1, \dots, x_n)$ a finite set of variables. We now show that this algebra has a simple form: It is a free Boolean algebra on a countable set of generators.

Proposition 3.20. *The set of terms*

$$\mathcal{G}_n \overset{\text{def}}{=} \{\mathrm{X}^t c \mid c \in C_n, t \geq 0\}$$

freely generates $F_{\mathbf{X}^{(k)}}(x_1, \dots, x_n)$ as a Boolean algebra.

Proof. Since X is a Boolean endomorphism, in any term, all occurrences of the operation X can be pushed down to the level of variables. Thus, the given set is generating. To see that it is *freely* generating, it suffices (see, e.g., [107, Prop. 9.4]) to prove that for any finite disjoint subsets F, G of the set of generators $\mathcal{G}_n$, we have $\bigwedge_{f \in F} f \wedge \bigwedge_{g \in G} \neg g \neq \bot$ in the algebra $F_{\mathbf{X}^{(k)}}(x_1, \dots, x_n)$. For this, it suffices to construct some deterministic modal algebra and a valuation of the variables and constants in it such that this term does not evaluate to $\bot$. To this end, we use the deterministic modal algebra $\mathcal{P}(\omega)$ introduced in Section 3.1, and, for any $c \in C_n$, define $V(c)$ to be the set of those $t \geq 0$ such that $\mathrm{X}^t c \in F$. Observe that, under this valuation V, the evaluation of the term $\bigwedge_{f \in F} f \wedge \bigwedge_{g \in G} \neg g$ will contain the point 0, and is thus non-empty. □

For the algebraic unifiability problem, we are interested in the existence of certain homomorphisms $F_{\mathbf{X}^{(k)}}(\bar{x}) \to F_{\mathbf{X}^{(k)}}(\emptyset)$. To study this question, we now use extended Stone duality. The free Boolean algebra generated by a set $\mathcal{G}$ is isomorphic to the Boolean algebra of clopen sets of generalized Cantor space $2^{\mathcal{G}}$ [60, Cor 4.12]. We apply this fact to the Boolean algebra $F_{\mathbf{X}^{(k)}}(\bar{x}) = F_{\mathbf{BA}}(\mathcal{G}_n)$, with $\mathcal{G}_n$ as in Proposition 3.20, and choose the Boolean algebra isomorphism $\widehat{(\cdot)}$ from $F_{\mathbf{X}^{(k)}}(\bar{x})$ to the algebra of clopen subsets of $2^{\mathcal{G}_n}$ which is defined by sending $g \in \mathcal{G}_n$ to

$$\widehat{g} \overset{\text{def}}{=} \{w \in 2^{\mathcal{G}_n} \mid w_g = 1\} .$$

In order to facilitate notation below, we will write, for $t \geq 0$ and $c \in C_n$, $w_{t,c}$ for the value of $w \in 2^{\mathcal{G}_n}$ at the term $\mathrm{X}^t c$. We may thus think of w as an ω-indexed word over the alphabet 2^{C_n}. With this notation, $\widehat{\mathrm{X}^t c}$ is the set of w such that $w_{t,c} = 1$.

Transporting the Boolean algebra endomorphism X on $F_{\mathrm{X}^{(k)}}(\bar{x})$ through the Boolean algebra isomorphism $\widehat{(\cdot)}$ gives an endomorphism s^{-1} on the Boolean algebra of clopen subsets of $2^{\mathcal{G}_n}$, where $s\colon\ 2^{\mathcal{G}_n} \to 2^{\mathcal{G}_n}$ is the continuous function that sends $w \in 2^{\mathcal{G}_n}$ to $s(w) \in 2^{\mathcal{G}_n}$ defined by

$$s(w)_{t,c} \overset{\text{def}}{=} w_{t+1,c}, \qquad \text{for each } c \in \mathcal{C}_n\ . \tag{3.7}$$

Let $\mathcal{X}^{(k)}$ denote the category of deterministic modal algebras with k constants, with morphisms the Boolean algebra homomorphisms preserving both X and the k constants. Let $\mathcal{S}$ denote the category of tuples $(S, s, P_1, \ldots, P_k)$, where S is a Boolean space, $s\colon\ S \to S$ is a continuous function, and, for each $1 \leq j \leq k$, P_j is a clopen subset of S. A morphism $\alpha\colon\ (S, s, P_1, \ldots, P_k) \to (S', s', P'_1, \ldots, P'_k)$ is a continuous *invariant* function $\alpha\colon\ S \to S'$, that is, it satisfies $\alpha \circ s = s' \circ \alpha$, and for each $1 \leq j \leq k$, $\alpha^{-1}(P'_j) = P_j$. Then $\mathcal{X}^{(k)}$ is dually equivalent to $\mathcal{S}$, so we have a bijection

$$\mathrm{Hom}_{\mathcal{S}}(2^{\mathcal{G}_0}, 2^{\mathcal{G}_n}) \quad \overset{\cong}{\to} \quad \mathrm{Hom}_{\mathcal{X}^{(k)}}(F_{\mathrm{X}^{(k)}}(\bar{x}), F_{\mathrm{X}^{(k)}}(\varnothing))\ , \tag{3.8}$$

which sends a continuous invariant function $\alpha\colon\ 2^{\mathcal{G}_0} \to 2^{\mathcal{G}_n}$ to the unique homomorphism $h_\alpha\colon\ F_{\mathrm{X}^{(k)}}(\bar{x}) \to F_{\mathrm{X}^{(k)}}(\varnothing)$ that satisfies $\widehat{h_\alpha(\phi)} = \alpha^{-1}(\widehat{\phi})$ for every term $\phi(\bar{x})$.

Now observe that, if $\phi \approx \top$ is a unification instance, then under the bijection in Eq. (3.8), the functions α such that $h_\alpha(\phi) = \top$ are exactly those for which $\alpha^{-1}(\widehat{\phi}) = 2^{\mathcal{G}_0}$, that is, $\mathrm{im}(\alpha) \subseteq \widehat{\phi}$. To sum up, we have proved the following:

Proposition 3.21. *For any term $\phi(\bar{x})$ in the algebraic type of $\mathrm{X}^{(k)}$, there is a bijective correspondence between the set of ground unifiers for ϕ and the set of continuous invariant functions $\alpha\colon\ 2^{\mathcal{G}_0} \to 2^{\mathcal{G}_n}$ such that* $\mathrm{im}(\alpha) \subseteq \widehat{\phi}$.

Remark 3.22. In the work that we discuss here, we only address the decidability question of *existence* of unifiers, so we only need the corollary of Proposition 3.21 that one of the sets is non-empty if, and only if, the other one is. In future work, we hope to exploit Proposition 3.21 to perform a finer analysis of the structure of unifiers.

To make Proposition 3.21 more directly useful, we now first interpret the invariance conditions in the specific case of a continuous function $\alpha\colon\ 2^{\mathcal{G}_0} \to 2^{\mathcal{G}_n}$. In light of Eq. (3.7), the condition $\alpha \circ s = s' \circ \alpha$ is equivalent to:

$$\text{for any } w \in 2^{\mathcal{G}_0}, t \geq 0 \text{ and } c \in \mathcal{C}_n, \alpha(s(w))_{t,c} = \alpha(w)_{t+1,c}. \tag{3.9}$$

Also, since the j^{th} clopen set P_j in the space $2^{\mathcal{G}_n}$ is by definition $\widehat{p_j}$, the condition $\alpha^{-1}(P'_j) = P_j$ becomes:

$$\text{for any } w \in 2^{\mathcal{G}_0}, \alpha(w)_{0,p_j} = w_{0,p_j}\ , \text{ for every } 1 \leq j \leq k. \tag{3.10}$$

Continuous functions satisfying Eq. (3.9) and Eq. (3.10) are ar special case of the *sliding block codes* studied in symbolic dynamics, see, e.g., [8, Prop. 5.4.1]. It is well-known that such functions can be defined by specifying only a finite amount of information, which is important to be able to use Proposition 3.21 to devise an algorithm for unifiability. I will now give a simple direct proof of this fact for our specific setting here. In the following proposition, when $F \subseteq \mathcal{G}_0$ and $u \in 2^{\mathcal{G}_0}$, we write $u{\restriction_F}$ for the restriction of u to a word in 2^F. For any $d \geq 0$, let $\mathcal{G}_0^d \overset{\text{def}}{=} \{\mathrm{X}^t p \mid 0 \leq t < d, p \in \mathcal{C}_0\}$.

Proposition 3.23. *Let $F \subseteq \mathcal{G}_0$ be a finite subset containing C_0, and let $f : 2^F \to 2^{C_n}$ be a function such that $f(u)_{0,p} = u_{0,p}$ for all $u \in 2^F$ and $p \in C_0$. Then the function $\bar{f} : 2^{\mathcal{G}_0} \to 2^{\mathcal{G}_n}$, defined, for $w \in 2^{\mathcal{G}_0}$, by*

$$\bar{f}(w)_{t,c} \overset{\text{def}}{=} f(s^t(w){\restriction_F})_{0,c} \quad (t \geq 0, c \in C_n)$$

is a continuous invariant function. Moreover, any continuous invariant function $\alpha : 2^{\mathcal{G}_0} \to 2^{\mathcal{G}_n}$ is equal to $\bar{f}$ for some $d \geq 0$ and $f : 2^{\mathcal{G}_0^d} \to 2^{C_n}$.

Proof. To see that the function $\bar{f}$ is continuous, it suffices to check that each coordinate projection of $\bar{f}$ is continuous. Indeed, for any $t \geq 0$ and $c \in C_n$, the function $\pi_{t,c} \circ \bar{f}$ is equal to the composition of continuous functions $\pi_{0,c} \circ f \circ (-){\restriction_F} \circ s^t$. To see that $\bar{f}$ is invariant, we check conditions Eq. (3.9) and Eq. (3.10). For any $w \in 2^{\mathcal{G}_0}$, $t \geq 0$ and $c \in C_n$, we have

$$\bar{f}(s(w))_{t,c} = f(s^{t+1}(w){\restriction_F})_{0,c} = \bar{f}(w)_{t+1,c} \ ,$$

and, when $c = p_j$, we have, by the assumption on f, that

$$\bar{f}(w)_{0,p_j} = f(w{\restriction_F})_{0,p_j} = w_{0,p_j} \ .$$

For the moreover statement, let $\alpha : 2^{\mathcal{G}_0} \to 2^{\mathcal{G}_n}$ be a continuous invariant function. The function $\tilde{\alpha} : 2^{\mathcal{G}_0} \to 2^{C_n}$ defined by $\tilde{\alpha}(w) \overset{\text{def}}{=} \alpha(w){\restriction_{C_n}}$ is still continuous. Since 2^{C_n} is finite, for any $u \in 2^{C_n}$, the set $\tilde{\alpha}^{-1}(u)$ is clopen, so it is a Boolean combination of a finite number of sets of the form $\widehat{g}$, where $g \in \mathcal{G}_n$. We may therefore pick a finite subset F of $\mathcal{G}_n$ such that, for every $u \in 2^{C_n}$, the set $\tilde{\alpha}^{-1}(u)$ is a Boolean combination of sets of the form $\widehat{g}$, with $g \in F$. It follows that, for any $w, w' \in 2^{\mathcal{G}_0}$, if $w{\restriction_F} = w'{\restriction_F}$, then $\tilde{\alpha}(w) = \tilde{\alpha}(w')$. Pick $d \geq 0$ sufficiently large so that $F \subseteq \mathcal{G}_0^d$. The choice of F and d then allows us to choose a function $f : 2^{\mathcal{G}_0^d} \to 2^{C_n}$ such that $f(w{\restriction_{\mathcal{G}_0^d}}) = \tilde{\alpha}(w)$ for all $w \in 2^{\mathcal{G}_0}$. The fact that α verifies Eq. (3.10) immediately implies that $f(u)_{0,p} = u_{0,p}$ for all $u \in 2^{\mathcal{G}_0^d}$ and $p \in C_0$. To see that $\bar{f} = \alpha$, we calculate, for any $t \geq 0$ and $c \in C_n$,

$$\bar{f}(w)_{t,c} = f(s^t(w){\restriction_{\mathcal{G}_0^d}})_{0,c} = \tilde{\alpha}(s^t(w))_{0,c} = \alpha(s^t(w))_{0,c} = \alpha(w)_{t,c}$$

where the last equality holds by repeated application of Eq. (3.9). □

In order to complete the first step of our unification algorithm, which reduces the algebraic unification problem to a finite graph problem, we need to understand which functions $f : 2^F \to 2^{C_n}$ are such that $\mathrm{im}(\bar{f}) \subseteq \widehat{\phi}$. As explained above, we may assume that ϕ is affine. Let us write $V_n \overset{\text{def}}{=} 2^{\{x_1,\dots,x_n\}}$ and $\Sigma \overset{\text{def}}{=} 2^{\{p_1,\dots,p_k\}}$. We write ϕ in disjunctive normal form: For any $u \in V_n$, write $\bullet u \overset{\text{def}}{=} \bigwedge_{i\,:\, u(x_i)=1} x_i \wedge \bigwedge_{i\,:\, u(x_i)=0} \neg x_i$, and similarly for any $a \in \Sigma$, write $\circ a \overset{\text{def}}{=} \bigwedge_{i\,:\, a(x_i)=1} p_i \wedge \bigwedge_{i\,:\, u(a_i)=0} \neg p_i$. Then ϕ is equivalent to

$$\bigvee \{\mathrm{X}(\bullet v) \wedge (\circ a) \wedge (\bullet u) \mid (u, a, v) \in E_\phi\}$$

where $E_\phi \overset{\text{def}}{=} \{(v, a, u) \in V_n \times \Sigma \times V_n \mid \ \vDash_{\mathbf{X}^{(k)}} [\mathrm{X}(\bullet v) \wedge (\circ a) \wedge (\bullet u)] \Rightarrow \phi \approx \top\}$. Note that it may be checked whether or not $(v, a, u) \in E_\phi$, for example by checking whether ϕ is true in a linear transition system on two states, where $\bullet u$ and $\circ a$ hold in the root, and $\bullet v$ holds in the successor of the root; for a single triple, this can be done in polynomial time, and there are exponentially many triples to consider, so the entire set E_ϕ can be computed within exponential time.

It follows that, for a given $f : 2^{\mathcal{G}_0^d} \to 2^{\mathcal{C}_n}$ as in Proposition 3.23, we have $\mathrm{im}(\bar{f}) \subseteq \widehat{\phi}$ if, and only if, for every $w \in 2^{\mathcal{G}_0}$, there exists $(v, a, u) \in E_\phi$ such that $\bar{f}(w)$ is an element of the clopen set that represents the term $\mathrm{X}(\bullet v) \wedge (\circ a) \wedge (\bullet u)$. The condition that $\bar{f}(w)$ is in the clopen set $\widehat{\mathrm{X}(\bullet v) \wedge (\circ a) \wedge (\bullet u)}$ can be expressed more directly in terms of f: Unraveling the definitions, we see that it is verified if, and only if, for any variable x and constant p,

$$f(s(w)\restriction_{\mathcal{G}_0^d})_{0,x} = v_x, \quad w_{0,p} = f(w)_{0,p} = a_p, \quad f(w\restriction_{\mathcal{G}_0^d})_{0,x} = u_x \,. \tag{3.11}$$

We now show that Eq. (3.11) is equivalent to a graph homomorphism condition. Here, by *graph* we will always mean a finite directed graph with Σ-labeling on the edges, i.e., a pair $G = (V_G, E_G)$ with V_G a finite set and $E_G \subseteq V_G \times \Sigma \times V_G$. We often write $x \xrightarrow{a} y$ to mean $(x, a, y) \in E_G$, and we extend this notation to paths, writing, for $w \in \Sigma^+$, $x \xrightarrow{w} y$ if there exists a w-labeled path in G starting in x and ending in y. A *homomorphism* from a graph (V_G, E_G) to a graph (V_H, E_H) is a function $f : V_G \to V_H$ such that, for any $x, y \in V_G$ and $a \in \Sigma$, if $x \xrightarrow{a} y$, then $f(x) \xrightarrow{a} f(y)$. We recall the definition of the sequence of *de Bruijn graphs* [26, 152].

Definition 3.24. The *de Bruijn graph* of dimension $d \geq 1$ over alphabet Σ is the graph $B_d(\Sigma) = (\Sigma^d, S_d)$ with set of nodes Σ^d and edge relation defined as

$$S_d \overset{\text{def}}{=} \{(b\alpha, a, \alpha a) \mid a, b \in \Sigma, \alpha \in \Sigma^{d-1}\} \subseteq \Sigma^d \times \Sigma \times \Sigma^d \,.$$

In Fig. 3.3, we show the de Bruijn graph B_3 of dimension 3 for $\Sigma = \{0, 1\}$.

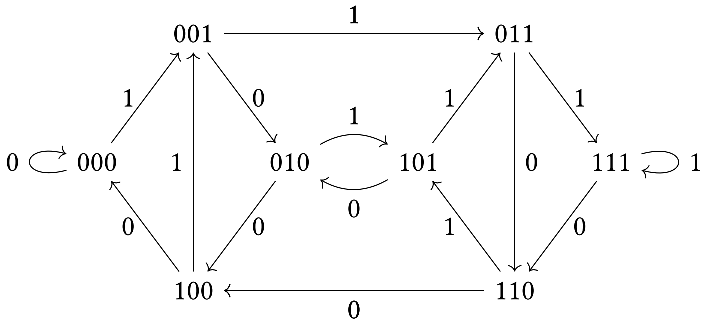


Figure 3.3: The de Bruijn graph $B_3(\{0, 1\})$.

The De Bruijn graph can be viewed as a deterministic automaton that ‘remembers’ the last d letters of an input word: Indeed, from any start node, there is a unique path labeled by $w \in \Sigma^*$, and if w is of length at least d, then the node reached by this unique path is the length d suffix of w. In what follows, when a finite alphabet Σ is clear from the context, we write B_d for $B_d(\Sigma)$.

To explain the connection with our unification problem for $\mathbf{X}^{(k)}$, we note that an element w of $2^{\mathcal{G}_0^d}$ may be viewed as a node $\widetilde{w}$ of Σ^d, where $\Sigma = 2^{\{p_1, \ldots, p_k\}}$, as follows. For $1 \leq t \leq d$, the letter in $\widetilde{w}$ at position t *counting from the right*[9] is the bit-string in Σ whose j^{th} bit has value w_{t,p_j}, for every $1 \leq j \leq k$. We claim that, then, the edges of the de Bruijn graph B_d are exactly the triples of the form $(\widetilde{s(w)\restriction_{\mathcal{G}_0^d}}, a, \widetilde{w\restriction_{\mathcal{G}_0^d}})$, where w is any element of $2^{\mathcal{G}_0}$. We give an example illustrating this fact

[9] We here start counting positions from the right, following the convention in the de Bruijn graph literature that the ‘most recent’ bits of information are added onto the right of the word.

when $k = 1$ (we write p for the only constant), so that $\Sigma = \{0, 1\}$. Let w be the element of $2^{\mathcal{G}_0}$ that starts $(p \mapsto 1, \mathrm{X}p \mapsto 1, \mathrm{X}^2 p \mapsto 0, \mathrm{X}^3 p \mapsto 1)$, and then has only zeros. The element $s(w)$ starts with $(p \mapsto 1, \mathrm{X}p \mapsto 0, \mathrm{X}^2 p \mapsto 1)$, and then has only zeros. Then $\widetilde{w\restriction_{\mathcal{G}_0^3}}$ is the node 011, and $\widetilde{s(w)\restriction_{\mathcal{G}_0^3}}$ is the node 101, and we indeed have an edge $(101, 1, 011)$ in $B_3(\{0, 1\})$.

For any function $f : 2^{\mathcal{G}_0^d} \to 2^{C_n}$, we now also write $\tilde{f}$ for the function $\Sigma^d \to V_n$ which is defined by sending, for any $w \in 2^{\mathcal{G}_0^d}$, the node $\tilde{w} \in \Sigma^d$ to $f(w)\restriction_{\{x_1,\dots,x_n\}}$. Then, using the observation of the previous paragraph, Eq. (3.11) holds for f if, and only if, $\tilde{f}$ is a homomorphism from the de Bruijn graph B_d to the graph (V_n, E_ϕ). We conclude:

Proposition 3.25. *An affine $\mathrm{X}^{(k)}$-term ϕ is ground-unifiable if, and only if, for some $d \geq 1$, there exists a homomorphism from B_d to (V_n, E_ϕ).*

Thus, the unifiability problem for $\mathrm{X}^{(k)}$ has an exponential-time reduction to the following problem.

Definition 3.26. Let Σ a finite alphabet. The *de Bruijn graph mapping problem* over Σ is the following computational problem: Given as input a finite directed graph G with edge-labeling from Σ, output a number $d \geq 1$ and a homomorphism from $B_d(\Sigma)$ to G, or output ‘impossible’ if none such exists.

Example 3.27. We revisit the examples of Example 3.18. To make the computations in the first example a bit simpler, we already use the first observation from Example 3.18, that $\mathrm{XX}(x \vee y)$ will necessarily be unified with $\top$ by any solution, so that the unification instance, after flattening, becomes

$$\phi \overset{\text{def}}{=} [x \Leftrightarrow (\neg \mathrm{X} z)] \wedge [y \Leftrightarrow (x \Rightarrow p)] \wedge [z \Leftrightarrow p]\,.$$

One may compute that the set E_ϕ contains 16 triples, and that 4 nodes that appear as start nodes of edges. If a node does not appear as a start node of an edge, then we can clearly discard it from the graph, as it can never be reached by a homomorphism from a de Bruijn graph. Making this further reduction, we obtain a graph on just 4 nodes, with 8 edges, depicted on the left in Fig. 3.4. We write, for instance, $x\neg y\neg z$ for the node $(x \mapsto 1, y \mapsto 0, z \mapsto 0)$ in V_3.

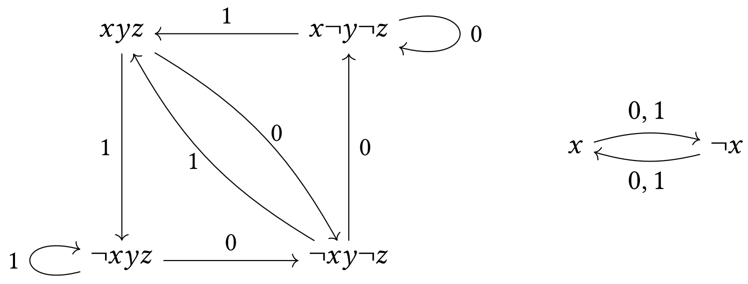


Figure 3.4: The graphs associated to the formulas ϕ and ψ of Example 3.27.

One may now check that the graph on the left in Fig. 3.4 admits a unique homomorphism from the de Bruijn graph B_2, namely, the function $\tilde{f}$ defined by $(00 \mapsto x\neg y\neg z, 01 \mapsto xyz, 10 \mapsto \neg xy\neg z, 11 \mapsto \neg xyz)$. Retracing steps through Proposition 3.23 and Proposition 3.21, one may compute from this function $\tilde{f}$ a unifier for ϕ, by sending each variable $c \in \{x, y, z\}$ to a formula σ_c such that $\widehat{\sigma_c} = \tilde{f}^{-1}(\hat{c})$. To compute such formulas σ_c, one may use the fact that each node of B_d can be described by a conjunction of literals; for example, 01 is the unique node in $\widetilde{p \wedge \neg \mathrm{X}p}$. In the example, this gives us the formulas $\sigma_x = (\neg p \wedge \neg \mathrm{X}p) \vee (p \wedge \neg \mathrm{X}p)$, $\sigma_y = (p \wedge \neg \mathrm{X}p) \vee (\neg p \wedge \mathrm{X}p) \vee (p \wedge \mathrm{X}p)$, and $\sigma_z = (p \wedge \neg \mathrm{X}p) \vee (p \wedge \mathrm{X}p)$. With some simple manipulations in deterministic modal algebra, one sees that this unifier is equivalent to $x \mapsto \neg \mathrm{X}p$, $y \mapsto p \vee \mathrm{X}p$ and $z \mapsto p$, as expected.

On the other hand, we can now also show that $\psi \overset{\text{def}}{=} (x \Leftrightarrow \neg \mathrm{X}x)$ does *not* have a unifier. Indeed, the graph with edges E_ψ is depicted on the right in Fig. 3.4. This graph clearly does not admit a homomorphism from any de Bruijn graph, for example because it does not have any loops.

Deciding the de Bruijn graph mapping problem. At the end of Example 3.27, we saw a simple reason for answering 'impossible' in the de Bruijn graph mapping problem. In general, however, it is not a priori clear when a graph does *not* admit a homomorphism from any de Bruijn graph B_d.

We call a graph G a *target* (*of a de Bruijn graph*) if there exists, for some $d \geq 1$, a homomorphism $B_d \to G$. Our goal is, then, to give a decidable characterization of the graphs that are targets. In [23, Prop. 3.4], such a characterization was given in case the input graph G is *deterministic* i.e., for every $x \in V_G$ and $a \in \Sigma$, there exists a unique $y \in V_G$ such that $x \xrightarrow{a} y$.[10] It is proved there that, for any $d \geq 1$, a deterministic graph G admits a surjective homomorphism from B_d if, and only if, G is strongly connected and d*-synchronizing*. Here, a graph is *strongly connected* if there is a path between any two nodes, and d*-synchronizing* if, for every $w \in \Sigma^d$, there exists a node y_w such that, for every $x \in V_G$, there exists a path $x \xrightarrow{w} y_w$ in G. It is also shown in [22, Sec. 8.2] that it can be checked whether or not there exists d such that a graph is d-synchronizing; in fact, checking whether this is the case for $d \leq |V_G|$ suffices, and the procedure outlined in [22, Sec 8.2] takes linear time in the size of the input graph.

Note, however, that the homomorphic image of a deterministic graph, such as B_d, may fail to be deterministic. Moreover, one may show that essentially any Σ-labeled graph, not necessarily deterministic, can occur as the graph associated with a unification problem for the logic $\mathbf{X}^{(k)}$, with $|\Sigma| \leq 2^k$. We thus need to generalize the results of [22, 23] to the non-deterministic setting. We first establish that the criteria from the deterministic case are no longer sufficient.

Example 3.28 (The 'hamburger' graph)**.** Consider the graph in Fig. 3.5. This graph is strongly connected and 2-synchronizing. However, it is *not* an image of B_2. One may see this as follows: In B_2, we have $00 \xrightarrow{1} 01 \xrightarrow{1} 11$; a homomorphism must map this to $x \xrightarrow{1} z \xrightarrow{1} y$. Similarly, $11 \xrightarrow{0} 10 \xrightarrow{0} 00$ must be sent to $y \xrightarrow{0} z \xrightarrow{0} x$. But now the edge $10 \xrightarrow{1} 01$ is not preserved: $z \overset{1}{\not\to} z$. In fact, as will follow from Theorem 3.29, the hamburger graph does not admit a homomorphism from B_d for any d.

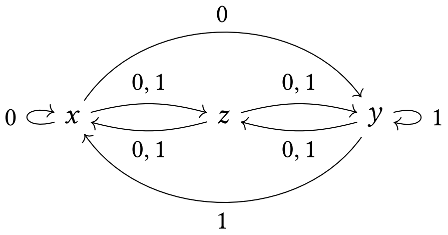


Figure 3.5: A graph that is strongly connected, 2-synchronizing, but not a de Bruijn graph target.

We will now identify the property that all targets of de Bruijn graphs have, and that the hamburger

[10] Deterministic graphs are simply called 'automata' in [23], but we avoid this terminology here in order not to clash with more general notions of automaton used earlier in this chapter.

graph of Example 3.28 lacks. We will call this property *power-connectedness*, as it is a strong form of connectedness for the power graph.

First, the strong form of connectedness is defined as follows. Let $H = (V_H, E_H)$ be a graph. A node $u \in V_H$ is a *predecessor* of a set $S \subseteq V_H$ if, for every $a \in \Sigma$, there exists $s \in S$ such that $u \xrightarrow{a} s$. A set $C \subseteq V_H$ is *closed* if it contains all of its predecessors. Note that the closed subsets of a graph form a family that is closed under intersections. Thus, we can define the *closure* of a set $S \subseteq V_H$ to be the smallest closed set containing S. Note that the closure of a set can be computed in at most $|V_H|$ steps, by a simple saturation algorithm, where each step takes polynomial time.

Now, for a graph $G = (V_G, E_G)$, the *power graph* of G is the graph $\mathbb{P}(G) = (\mathcal{P}(V_G), E_{\mathbb{P}(G)})$, where, for any $S, T \subseteq V_G$,

$$S \xrightarrow{a}_{\mathbb{P}(G)} T \overset{\text{def}}{\iff} \text{for every } x \in S, \text{ there exists } y \in T \text{ such that } x \xrightarrow{a}_G y\ .$$

We now say that G is *power-connected* if, in the power graph $\mathbb{P}(G)$, the node V_G is in the closure of the set of singleton nodes $\{\{u\} \mid u \in V_G\}$. For example, the hamburger graph of Example 3.28 fails to be power-connected: One may compute that the closure of the set of singleton nodes is $\{\{x\}, \{y\}, \{z\}, \{x, y\}\}$, which does not contain $V_G = \{x, y, z\}$. Computing the power graph of a graph takes exponential time, so that checking whether a graph is power-connected takes exponential time as well.

A second property that is necessary to be a target of a de Bruijn graph is to contain ‘a lot of cycles, reachable from anywhere’. Indeed, observe that, in the de Bruijn graph B_d itself, for any word $w \in \Sigma^+$, if one reads the word w^d starting from anywhere in the graph, then one will end up on a cycle labeled by w. This property is transferred to the homomorphic image. The following definition formalizes this idea.

Let $G = (V_G, E_G)$ be a graph. For $w \in \Sigma^+$, we call a node u a *w-cycle basepoint* if there exists a path $u \xrightarrow{w} u$. We say that a w-cycle basepoint u is *reachable* if there exists $\ell \geq 1$ such that, for every node $v \in V_G$, there is a path $v \xrightarrow{w^\ell} u$. Finally, we say that G is *cycle-connected* if, for every $w \in \Sigma^+$, there exists a reachable w-cycle basepoint in G.

Neither of the properties of cycle-connectedness and power-connectedness implies the other: One direction is given by the hamburger graph of Example 3.28, and Fig. 3.6 gives an example of a graph (the ‘cone of fries’) that is power-connected but clearly not cycle-connected, as it does not contain any 0-cycle.

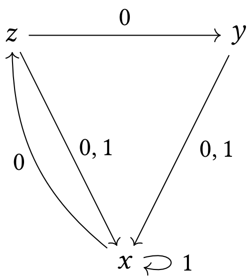


Figure 3.6: A power-connected graph that is not cycle-connected.

Our main result, which characterizes those graphs that admit a homomorphism from a de Bruijn graph, is the following.

Theorem 3.29. *A graph G has a cycle- and power-connected subgraph if, and only if, there exist $d \geq 1$ and a homomorphism $B_d \to G$.*

We further claim that it can be checked in exponential time whether or not a given input graph has a cycle- and power-connected subgraph. Thus, Theorem 3.29 in particular implies that the de Bruijn graph mapping problem is decidable in exponential time. Combining this with the exponential-time reduction that led up to Definition 3.26, we obtain the claimed doubly-exponential algorithm for the unifiability problem for $\mathbf{X}$ with a fixed number of constants.

Having formulated the conditions of cycle- and power-connectedness, it is reasonably straightforward to show that any surjective graph homomorphism must preserve them, so that any homomorphic image of a de Bruijn graph must be cycle- and power-connected. Thus, any target of a de Bruijn graph contains a cycle- and power-connected subgraph, namely, the image of the homomorphism. The proof of the left-to-right direction of Theorem 3.29 is the most combinatorially involved part of this work, and we only give a brief impression of its main ingredients here.

Given a cycle- and power-connected graph G, we need to define a homomorphism $h : B_d \to G$, for some d. Observe that, if we are able to define such h, then, for any letter $a \in \Sigma$, the node of the form a^d *must* be sent to a node with a-loop under the homomorphism. An a-loop exists in G, by an application of cycle-connectedness. By similarreasoning, for any word $w \in \Sigma^+$, the node of B_d that is reached from anywhere by reading w^d must be sent by h to a w-cycle basepoint in G. However, there is no clear canonical choice for such a basepoint, and a problem may arise if one chooses these basepoints too arbitrarily: For instance, consider the nodes $u_1 = (01)^{d/2}$ and $u_2 = (10)^{d/2}$ of B_d (where we assume for convenience that d is even). The basepoint chosen in G as the image of u_1 must have a 0-edge to the basepoint chosen as the image of u_2. A consistent such choice can indeed be made, using again the cycle-connectedness property and choosing an ordering on the cyclic conjugacy class of a word. The final, and arguably most difficult problem, in this part of the proof is where to send nodes of B_d that are *not* powers of small words. We cannot assign them completely arbitrary, since such nodes will have paths to nodes that are powers of small words, so they should not be mapped too far away from the chosen cycle basepoints. To achieve this, we use the power-connectedness assumption, from which we can deduce that, from any node in the graph, there are consistent choices of paths which will allow us to 'synchronize', in a similar way to the deterministic case. To make the details of this synchronization process work, we use a technique from the theory of string compression, called *minimizers* [149, 155], combined with techniques from combinatorics on finite words, notably the *critical factorization theorem* [115, Ch. 8]. Further details of the proof of Theorem 3.29 will be provided in the forthcoming preprint [80], also see [166].

3.3 Outlook on temporal logic

The work on temporal logic, model-completeness, and decidability of unification problems that I discussed in this chapter suggests several directions for further research, which I will outline now.

The general fair tree logic described in Section 3.1 only considers trees up to bisimilarity, and its model companion thus only encodes bisimulation-invariant monadic second-order logic, which explains the close connection to the μ-calculus, in light of the equi-expressivity of the two logics [103]. In current ongoing work with L. Carai and S. Ghilardi, we aim to extend our result so as to view *full* monadic second-order logic on trees as a model companion. This will involve breaking the bisimulation-invariance by extending our fair tree logic with *counting* temporal operators. We will also have to reprove the completeness of the axiomatization of this version of fair tree logic with counting, that is, the analogue of Theorem 3.13.

While the Kozen-Park axiomatization of the full μ-calculus is known to be complete for the intended transition system semantics [109, 177], no general method is known for proving, when $\mathcal{F}$ is a definable fragment of the μ-calculus, the completeness of the Kozen-Park axioms *restricted to the fragment* $\mathcal{F}$. Such a completeness result is really what we needed in Theorem 3.13: If t is a term of fair tree logic that is valid on any tree, then the cited completeness results only imply that there is a syntactic derivation of the equality $t = \top$ *in the full μ-calculus*. However, such a derivation might, a priori, always require passing through other fixed point formulas, which are not available in fair tree logic signature itself. Our Theorem 3.13 establishes that this cannot happen in the case of fair tree logic, but a general method for establishing such a 'conservativity' result for sufficiently well-behaved fragments $\mathcal{F}$ is an important open problem, which might benefit from a more general vantage point than the *ad hoc* construction we developed in [70, Sec. 3] for proving completeness in the case of fair tree logic. A line of work by Y. Venema and various co-authors suggests a similar direction, and uses algebra and coalgebra for proving such generic completeness results for certain fragments of the μ-calculus, see e.g. [51, 153, 172].

In Section 3.2, we only looked at the most basic computational problem regarding unification, namely, whether or not a unifier exists for a given instance. The unification literature also commonly considers questions of unification *type* [14, Def. 3.4], and the possibility of finding *projective* unifiers [69]. These questions are open for the varieties $\mathbf{X}^{(k)}$, and one way to investigate them would be to reduce them to graph problems, as well.

A notable, and somewhat notorious, open problem in the realm of unifiability is the question whether the unifiability problem for *non-deterministic* modal logic **K**, with any number of parameters (including 0), is decidable. Our work in Section 3.2 emerged out of an attempt to study this problem coalgebraically. In [81], we explained how to perform the first part of the reduction done for *deterministic* modal algebras in Section 3.2 in the more general setting of **K**. This gives a reformulation of the unifiability problem for **K** in terms of a computational problem about neighborhood frames or hypergraphs. However, decidability of this problem is so far beyond our reach.

The reductions of unification problems to graph-like problems given in Section 3.2 rely, in the background, on the fact that both **X** and **K** are varieties of *algebras for a functor*, which allows us to make use of algebra-coalgebra duality. Unification is also of interest for other logics, such as **S4** [66] and intuitionistic logic **I** [68], whose associated varieties are not immediately algebras for a functor in the usual sense. A further equational theory of interest is that of Boolean algebra in the signature that only has the XOR operator and an endomorphism X for it. This unification problem has been previously studied with different methods in the context of cryptographic protocols [113], and would be interesting to revisit with our methods. Following [67], which was an inspiration for our work in Section 3.2, I would like to investigate if any of these algebraic/coalgebraic methods may still be applicable to unification problems in these logics.

Let me end by mentioning a curious and unexpected connection [11, p. 3, footnote 1] which touches on all three themes discussed in this document, and also deserves further investigation. This fact concerns a connection between the *separation* problem for first-order logic (Chapter 1), and interpolation

(Chapter 2) for linear temporal logic (Chapter 3). It is known that linear temporal logic does not have interpolants, in general. The *interpolant existence problem* for linear temporal logic asks, given two linear temporal formulas ϕ and ψ such that $\phi \Rightarrow \psi$ is valid, whether or not there exists an interpolant θ for ϕ and ψ. The answer turns out to be positive if, and only if, certain regular languages associated to ϕ and ψ admit a first-order separator. The reason for this is, in short, that linear temporal logic with added propositional quantifiers can define any regular language. Thus, the decidability of separation for first-order definable languages (which we discussed in Section 1.2) implies the decidability of this interpolant existence problem for linear temporal logic. The interpolant existence problem is of course not limited to linear temporal logic, and it is thus tempting to investigate whether interpolation existence problems for other modal and temporal logics could also be solved by automata- or monoid-theoretic techniques. I leave this to future work.

Bibliography

[1] S. Abramsky and L. Reggio. “Arboreal categories and equi-resource homomorphism preservation theorems”. *Annals of Pure and Applied Logic* 175.6 (2024), p. 103423.

[2] G. D’Agostino and M. Hollenberg. “Logical questions concerning the μ-calculus: interpolation, Lyndon and Łoś-Tarski”. *J. Symbolic Logic* 65.1 (2000), pp. 310–332.

[3] J. Almeida. *Finite semigroups and universal algebra.* Vol. 3. Series in Algebra. Translated from the 1992 Portuguese original and revised by the author. World Scientific Publishing Co. Inc., 1994. xviii+511.

[4] J. Almeida, J. C. Costa, and M. Zeitoun. “Iterated periodicity over finite aperiodic semigroups”. *European J. Combin.* 37 (2014), pp. 115–149.

[5] J. Almeida, J. C. Costa, and M. Zeitoun. “McCammond’s normal forms for free aperiodic semigroups revisited”. *LMS J. Comput. Math.* 18.1 (2015), pp. 130–147.

[6] J. Almeida. “Some algorithmic problems for pseudovarieties”. *Publ. Math. Debrecen* 54.1 (1999), pp. 531–552.

[7] J. Almeida, A. Costa, J. C. Costa, and M. Zeitoun. “The linear nature of pseudowords”. *Publicacions matematiques* 63.2 (2019), pp. 361–422.

[8] J. Almeida, A. Costa, R. Kyriakoglou, and D. Perrin. *Profinite semigroups and symbolic dynamics.* Springer, 2020.

[9] J. Almeida, J. C. Costa, and M. Zeitoun. “Pointlike sets with respect to R and J”. *Journal of Pure and Applied Algebra* 212.3 (2008), pp. 486–499.

[10] J. Almeida, H. Goulet-Ouellet, and O. Klíma. “What makes a Stone topological algebra Profinite”. *Algebra universalis* 84.1 (2023).

[11] A. Artale, J. C. Jung, A. Mazzullo, A. Ozaki, and F. Wolter. “Living without Beth and Craig: Explicit definitions and interpolants in description logics with nominals”. In: *Proceedings of the 33rd International Workshop on Description Logics (DL 2020).* 2020.

[12] M. v. Atten. “The Development of Intuitionistic Logic”. In: *The Stanford Encyclopedia of Philosophy.* Ed. by E. N. Zalta and U. Nodelman. Fall 2023. Metaphysics Research Lab, Stanford University, 2023. URL: https://plato.stanford.edu/entries/intuitionistic-logic-development/.

[13] F. Baader and S. Ghilardi. “Unification in modal and description logics”. *Logic Journal of IGPL* 19.6 (2011), pp. 705–730.

[14] F. Baader, W. Snyder, P. Narendran, M. Schmidt-Schauss, and K. Schulz. “Unification Theory”. In: *Handbook of Automated Reasoning.* Ed. by A. Robinson and A. Voronkov. Handbook of Automated Reasoning. Amsterdam: North-Holland, 2001, pp. 445–533.

[15] J. d. Bakker and D. Scott. “A theory of programs : an outline of joint work : IBM seminar Vienna, August 1969”. In: *J.W. de Bakker, 25 jaar semantiek.* C, Jan. 1989, pp. 1–30.

[16] P. Balbiani, C. Gencer, M. Mojtahedi, M. Rostamigiv, and T. Tinchev. “A gentle introduction to unification in modal logics”. In: *13èmes Journées d’Intelligence Artificielle Fondamentale (JIAF 2019).* 2019.

[17] P. Balbiani, C. Gencer, M. Rostamigiv, and T. Tinchev. “Remarks about the unification types of some locally tabular normal modal logics”. *Logic Journal of the IGPL* 31.1 (2022), pp. 115–139.

[18] P. Balbiani and Q. Gougeon. “Projective unification through duality”. In: *Advances in Modal Logic, AiML 2022, Rennes, France, August 22-25, 2022.* Ed. by D. Fernández-Duque, A. Palmigiano, and S. Pinchinat. College Publications, 2022, pp. 119–134.

[19] M. Barr. “Terminal coalgebras in well-founded set theory”. *Theoretical Computer Science* 114.2 (1993), pp. 299–315.

[20] F. Bellissima. "Finitely generated free Heyting algebras". *Journal of Symbolic Logic* 51.1 (1986), pp. 152–165.

[21] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Vol. 53. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 2001.

[22] C. Bleak, P. Cameron, Y. Maissel, A. Navas, and F. Olukoya. *The Further Chameleon Groups of Richard Thompson and Graham Higman: Automorphisms via Dynamics for the Higman–Thompson Groups $G_{n,r}$*. Preprint. 2016. URL: https://arxiv.org/abs/1605.09302.

[23] C. Bleak, P. J. Cameron, and F. Olukoya. "Automorphisms of shift spaces and the Higman–Thompson groups: the one-sided case". *Discrete Analysis* (2021).

[24] S. L. Bloom and Z. Ésik. "The equational theory of regular words". *Information and Computation* 197 (2005), pp. 55–89.

[25] M. Bojańczyk. *Recognisable languages over monads*. Preprint. 2015. URL: http://arxiv.org/abs/1502.04898.

[26] N. G. de Bruijn. "A combinatorial problem". *Proceedings of the Koninklijke Nederlandse Akademie van Wetenschappen* 49.7 (1946), pp. 758–764.

[27] J. R. Büchi. "Weak second-order arithmetic and finite automata". *Zeitschr. f. math. Logik und Grundlagen d. Math.* 6 (1960), pp. 66–92.

[28] J. R. Büchi. "On a decision method in restricted second-order arithmetic". In: *Proceedings 1960 International Congress for Logic, Methodology and Philosophy of Science*. Ed. by E. Nagel, P. Suppes, and A. Tarski. Stanford University Press, 1962, pp. 1–11.

[29] S. Burris and H. P. Sankappanavar. *A Course in Universal Algebra*. Update of the original 1981 Springer edition. Springer, 2012.

[30] L. Cabrer and G. Metcalfe. "Admissibility via natural dualities". *Journal of Pure and Applied Algebra* 219.9 (2015), pp. 4229–4253.

[31] A. Chagrov and M. Zakharyaschev. *Modal Logic*. Vol. 35. Oxford Logic Guides. Clarendon Press, Oxford, 1997.

[32] C. C. Chang and H. J. Keisler. *Model theory, 3rd edition*. North Holland, 1990.

[33] C. C. Chang. "A new proof of the completeness of the Lukasiewicz axioms". *Transactions of the American Mathematical Society* 93.1 (1959), pp. 74–80.

[34] C. C. Chang. "Algebraic analysis of many valued logics". *Transactions of the American Mathematical society* 88.2 (1958), pp. 467–490.

[35] C. C. Chang and H. J. Keisler. *Continuous Model Theory*. Princeton University Press, 1966.

[36] R. L. Cignoli, I. M. d'Ottaviano, and D. Mundici. *Algebraic foundations of many-valued reasoning*. Vol. 7. Springer Science & Business Media, 2000.

[37] T. Colcombet, S. v. Gool, and R. Morvan. "First-order separation over countable ordinals". In: *Foundations of software science and computation structures (FoSSaCS)*. Ed. by P. Bouyer and L. Schröder. 2022.

[38] P. Corbineau. "First-Order Reasoning in the Calculus of Inductive Constructions". In: *TYPES Conference*. Ed. by S. Berardi, M. Coppo, and F. Damiani. Vol. 3085. Lecture Notes in Computer Science. Springer, 2003, pp. 162–177.

[39] W. Craig. "Linear reasoning. A new form of the Herbrand-Gentzen theorem". *Journal of Symbolic Logic* 22.3 (1957), pp. 250–268.

[40] H. B. Curry. "Functionality in Combinatory Logic". *Proc. Natl. Acad. Sci. U. S. A.* 20 (Nov. 1934), pp. 584–590.

[41] Y. Dandan and V. Gould. "Coherency for monoids and purity for their acts". *Advances in Mathematics* 429 (2023), p. 109182.

[42] L. Darnière. *De la triangulation p-adique à la théorie des modèles des algèbres de Heyting, et vice-versa*. Habilitation à diriger des recherches. 2019. URL: https://math.univ-angers.fr/~darniere/pub/hdr/darniere-hdr.pdf.

[43] L. Darnière and M. Junker. "Model completion of varieties of co-Heyting algebras". *Houston J. Math.* 44.1 (2018), pp. 49–82.

[44] M. Dickmann, N. Schwartz, and M. Tressl. *Spectral Spaces.* New Mathematical Monographs. Cambridge University Press, 2019.

[45] C. DiSimone (translator). "The Sūtra on Impermanence (2) (Anityatāsūtra, mi rtag pa nyid kyi mdo, Toh 310)". In: *84000: Translating the Words of the Buddha.* 84000, 2024. URL: https://read.84000.co/translation/toh310.html.

[46] R. Dyckhoff. "Contraction-free sequent calculi for intuitionistic logic". *Journal of Symbolic Logic* 57.3 (1992), pp. 795–807.

[47] R. Dyckhoff and S. Negri. "Admissibility of Structural Rules for Contraction-Free Systems of Intuitionistic Logic". *The Journal of Symbolic Logic* 65.4 (2000).

[48] S. Eilenberg. *Automata, languages, and machines. Vol. B.* With two chapters ("Depth decomposition theorem" and "Complexity of semigroups and morphisms") by Bret Tilson, Pure and Applied Mathematics, Vol. 59. New York: Academic Press, 1976, pp. xiii+387.

[49] C. C. Elgot. "Decision Problems of Finite Automata Design and Related Arithmetics". *Trans. Amer. Math. Soc.* 98.1 (1961), pp. 21–51.

[50] E. A. Emerson and J. Y. Halpern. ""Sometimes" and "not never" revisited: on branching versus linear time (preliminary report)". In: *ACM-SIGACT Symposium on Principles of Programming Languages.* 1983.

[51] S. Enqvist, F. Seifan, and Y. Venema. "Completeness for μ-calculi: a coalgebraic approach". *Annals of Pure and Applied Logic* 170.5 (2019), pp. 578–641.

[52] L. Esakia. *Heyting algebras: Duality theory.* Vol. 50. Trends in Logic. 2019 translation of the original. Springer, 1985.

[53] L. Esakia. "Topological Kripke models". *Soviet Math. Dokl.* 15 (1974), pp. 147–151.

[54] S. Feferman. "Harmonious logic: Craig's interpolation theorem and its descendants". *Synthese* 164.3 (2008), pp. 341–357.

[55] H. Férée, I. v. d. Giessen, S. v. Gool, and I. Shillito. "Mechanised uniform interpolation for K, GL, and iSL". In: *Automated Reasoning: 12th International Joint Conference, IJCAR 2024, Nancy, France, July 3–6, 2024, Proceedings, Part I.* Ed. by C. Benzmüller, M. J. Heule, and R. A. Schmidt. 2024.

[56] H. Férée and S. v. Gool. "Formalizing and computing propositional quantifiers". *Certified proofs and programs (CPP)* (2023).

[57] M. Fraser, A. Granville, M. H. Harris, C. McLarty, E. Riehl, and A. Venkatesh. "Will machines change mathematics?" *Bulletin of the American Mathematical Society* 61.2 (2024). Introduction to a special issue on mathematics and artificial intelligence.

[58] W. Fussner, M. Gehrke, S. v. Gool, and V. Marra. "Priestley duality for MV algebras and beyond". *Forum Mathematicum* 33 (4 2021).

[59] M. Gehrke and S. v. Gool. "Distributive envelopes and topological duality for lattices via canonical extensions". *Order* 31 (3 2013), pp. 435–461.

[60] M. Gehrke and S. v. Gool. *Topological Duality for Distributive Lattices: Theory and Applications.* Cambridge University Press, 2024, pp. 352+xvi.

[61] M. Gehrke, S. v. Gool, and V. Marra. "Sheaf representations of MV-algebras and lattice-ordered abelian groups via duality". *Journal of Algebra* 417 (2014), pp. 290–332.

[62] M. Gehrke, S. Grigorieff, and J. Pin. "Duality and Equational Theory of Regular Languages". In: *Automata, Languages and Programming, 35th International Colloquium, ICALP 2008, Reykjavik, Iceland, July 7-11, 2008, Proceedings, Part II.* Ed. by L. Aceto, I. Damgård, L. A. Goldberg, M. M. Halldórsson, A. Ingólfsdóttir, and I. Walukiewicz. Vol. 5126. Lecture Notes in Computer Science. Springer, 2008, pp. 246–257.

[63] M. Gehrke and J. Harding. "Bounded Lattice Expansions". *Journal of Algebra* 238.1 (2001), pp. 345–371.

[64] M. Gehrke and B. Jónsson. "Bounded distributive lattices with operators". *Mathematica Japonica* 40.2 (1994), pp. 207–215.

[65] S. Ghilardi. "An algebraic theory of normal forms". *Annals of Pure and Applied Logic* 71.3 (1995), pp. 189–245.

[66] S. Ghilardi. "Best solving modal equations". *Annals of Pure and Applied Logic* 102.3 (2000), pp. 183–198.

[67] S. Ghilardi. *Handling Substitutions via Duality.* Slides. 2018. URL: https://easychair.org/smart-slide/slide/Bn7h.

[68] S. Ghilardi. "Unification in intuitionistic logic". *The Journal of Symbolic Logic* 64.2 (1999), pp. 859–880.

[69] S. Ghilardi. "Unification through projectivity". *Journal of Logic and Computation* 7.6 (1997), pp. 733–752.

[70] S. Ghilardi and S. J. v. Gool. "Monadic second order logic as the model companion of temporal logic". In: *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science, LICS '16, New York, NY, USA, July 5-8, 2016.* Ed. by M. Grohe, E. Koskinen, and N. Shankar. ACM, 2016, pp. 417–426.

[71] S. Ghilardi and S. v. Gool. "A model-theoretic characterization of monadic second-order logic on infinite words". *Journal of Symbolic Logic* 82 (1 2017), pp. 62–76.

[72] S. Ghilardi and M. Zawadowski. "Model completions and r-Heyting categories". *Annals of Pure and Applied Logic* 88.1 (1997), pp. 27–46.

[73] S. Ghilardi and M. Zawadowski. *Sheaves, Games, and Model Completions.* Springer, 2002.

[74] I. van der Giessen. *Uniform Interpolation and Admissible Rules.* Vol. 138. Quaestiones Infinitae. PhD thesis, Utrecht University. 2022.

[75] I. van der Giessen and R. Iemhoff. "Proof theory for intuitionistic strong Löb logic". English. *Accepted for publication in Special Volume of the Workshop Proofs! held in Paris in 2017* (2020). Preprint arXiv:2011.10383v2.

[76] R. Goldblatt. *Logics of time and computation.* 2nd. CSLI Lecture Notes 7. Stanford University, 1992.

[77] S. van Gool, P.-A. Melliès, and V. Moreau. "Profinite lambda-terms and parametricity". *Electronic Notes in Theoretical Informatics and Computer Science* Volume 3 - Proceedings of MFPS XXXIX (2023).

[78] S. v. Gool. "Duality and canonical extensions for stably compact spaces". *Topology and its Applications* 159 (1 2012), pp. 341–359.

[79] S. v. Gool and J. Marquès. "On duality and model theory for polyadic spaces". *Annals of Pure and Applied Logic* 175 (2024).

[80] S. v. Gool, J. Marti, and M. Sweering. *A decidable characterization of images of de Bruijn graphs.* Unpublished draft. 2024.

[81] S. v. Gool and J. Marti. "Modal unification step by step". *37th international workshop on Unification (UNIF)* (2023).

[82] S. v. Gool, G. Metcalfe, and C. Tsinakis. "Uniform Interpolation and Compact Congruences". *Annals of Pure and Applied Logic* 168 (2017), pp. 1927–1948.

[83] S. v. Gool and L. Reggio. "An open mapping theorem for finitely copresented Esakia spaces". *Topology and its Applications* 240 (2018), pp. 69–77.

[84] S. v. Gool and B. Steinberg. "Pointlike sets for varieties determined by groups". *Adv. Math.* 348 (2019), pp. 18–50.

[85] S. v. Gool and B. Steinberg. "Merge decompositions, two-sided Krohn-Rhodes, and aperiodic pointlikes". *Canadian Mathematical Bulletin* 62 (1 2019), pp. 199–208.

[86] S. v. Gool and B. Steinberg. "Pro-aperiodic monoids via saturated models". *Symposium on theoretical aspects of computer science (STACS)* 66 (2017), 39:1–39:14.

[87] S. v. Gool and B. Steinberg. "Pro-aperiodic monoids via saturated models". *Israel Journal of Mathematics* 234 (2019), pp. 451–498.

[88] S. v. Gool and B. Steinberg. *Proaperiodic monoids via prime models.* Draft. 2019. URL: https://www.samvangool.net/assets/pdf/GS2019primemodels-note.pdf.

[89] J. Goubault-Larrecq. *Non-Hausdorff Topology and Domain Theory.* Cambridge University Press, 2013.

[90] V. Gould. "Coherent monoids". *J. Australian Math. Soc.* 53 (1992), pp. 166–182.

[91] E. Grädel, W. Thomas, and T. Wilke, eds. *Automata Logics, and Infinite Games: A Guide to Current Research.* Vol. 2500. Lecture Notes in Computer Science. Springer, 2002.

[92] B. Hart. "An introduction to continuous model theory". In: *Model Theory of Operator Algebras.* Ed. by I. Goldbring. Berlin, Boston: De Gruyter, 2023, pp. 83–132.

[93] K. Henckell. "Pointlike sets: the finest aperiodic cover of a finite semigroup". *J. Pure Appl. Algebra* 55 (1988), pp. 85–126.

[94] K. Henckell and S. Herman. "A General Theory of Pointlike Sets". Preprint. 2021. URL: https://arxiv.org/abs/2108.12824.

[95] K. Henckell, J. Rhodes, and B. Steinberg. "Aperiodic pointlikes and beyond". *International Journal of Algebra and Computation* 20.02 (2010), pp. 287–305.

[96] V. Henriksson and M. Kufleitner. "Conelikes and Ranker Comparisons". In: *LATIN 2022: Theoretical Informatics: 15th Latin American Symposium, Guanajuato, Mexico, November 7–11, 2022, Proceedings.* Guanajuato, Mexico: Springer-Verlag, 2022, pp. 359–375.

[97] W. Hodges. *Model theory.* Vol. 42. Encyclopedia of mathematics and its applications. Cambridge University Press, 1993.

[98] W. A. Howard. "The formulae-as-types notion of construction". In: *To H.B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism.* Ed. by J. P. Seldin and J. R. Hindley. (Original paper manuscript from 1969). 1980, pp. 479–490.

[99] J. Hudelmaier. *A Prolog program for intuitionistic logic.* Tech. rep. SNS-Bericht 88-28. University of Tübingen, 1988.

[100] M. Huschenbett and M. Kufleitner. "Ehrenfeucht-Fraisse Games on Omega-Terms". In: *STACS.* 2014, pp. 374–385.

[101] R. Iemhoff. "Uniform interpolation and sequent calculi in modal logic". *Arch. Math. Logic* 58.1-2 (2019), pp. 155–181.

[102] T. Jakl, D. Marsden, and N. Shah. *A categorical account of composition methods in logic (extended version).* Preprint. 2024. URL: https://arxiv.org/pdf/2405.06664.

[103] D. Janin and I. Walukiewicz. "On the expressive completeness of the propositional mu-calculus with respect to monadic second order logic". In: *CONCUR '96: Concurrency Theory.* Vol. 1119. Lecture Notes in Computer Science. Springer, 1996, pp. 263–277.

[104] E. Jeřábek. "Blending margins: the modal logic K has nullary unification type". *Journal of Logic and Computation* 25.5 (2013), pp. 1231–1240.

[105] B. Jónsson and A. Tarski. "Boolean algebras with operators. I". *American Journal of Mathematics* 73.4 (1951), pp. 891–939.

[106] S. C. Kleene. "Representation of events in nerve nets and finite automata". In: *Automata studies.* Annals of mathematics studies 34. Princeton, N. J.: Princeton University Press, 1956, pp. 3–41.

[107] S. Koppelberg, J. Monk, and R. Bonnet. *Handbook of Boolean Algebras.* Vol. 1. North-Holland, 1989.

[108] T. Kowalski and G. Metcalfe. "Uniform interpolation and coherence". *Annals of Pure and Applied Logic* 170.7 (2019), pp. 825–841.

[109] D. Kozen. "Results on the propositional μ-calculus". *Theor. Comput. Sci.* 27 (1983), pp. 333–354.

[110] K. Krohn and J. Rhodes. "Algebraic theory of machines. I. Prime decomposition theorem for finite semigroups and machines". *Trans. Amer. Math. Soc.* 116 (1965), pp. 450–464.

[111] D. Linkhorn. "Monadic Second Order Logic and Linear Orders". PhD thesis. The University of Manchester, 2021. URL: https://research.manchester.ac.uk/files/208484417/FULL_TEXT.PDF.

[112] D. Linkhorn. *The pseudofinite monadic second order theory of words.* 2022. URL: https://arxiv.org/abs/2202.07774.

[113] Z. Liu and C. Lynch. "Efficient General Unification for XOR with Homomorphism". In: *Automated Deduction – CADE-23.* Ed. by N. Bjørner and V. Sofronie-Stokkermans. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 407–421.

[114] M. Lohrey and C. Mathissen. "Isomorphism of regular trees and words". *Inform. and Comput.* 224 (2013), pp. 71–105.

[115] M. Lothaire. *Combinatorics on Words.* Cambridge, United Kingdom: Cambridge University Press, 1997.

[116] J. Łukasiewicz. "O logice trojwartosciowej". *Ruch Filozoficny* 5 (1920), pp. 170–171.

[117] L. Maksimova. "Craig's theorem in superintuitionistic logics and amalgamable varieties of pseudoboolean algebras". *Algebra and Logic* 16.6 (1977), pp. 427–455.

[118] S. Margolis, J. Rhodes, and A. Schilling. "Decidability of Krohn–Rhodes complexity for all finite semigroups and automata". Preprint. 2024. URL: https://arxiv.org/pdf/2406.18477.

[119] S. Margolis, J. Rhodes, and A. Schilling. “Decidability of Krohn-Rhodes complexity $c = 1$ of finite semigroups and automata”. Preprint. 2023. URL: https://arxiv.org/abs/2110.10373.

[120] J. Marquès. “Categorical logic from the perspective of duality and compact ordered spaces”. PhD thesis. Nice: Université Côte d’Azur, 2023. URL: https://jeremie-marques.name/thesis.pdf.

[121] J. Marquès. “Polyadic Spaces and Profinite Monoids”. In: *Relational and Algebraic Methods in Computer Science*. Ed. by U. Fahrenberg, M. Gehrke, L. Santocanale, and M. Winter. Cham: Springer International Publishing, 2021, pp. 292–308.

[122] U. Martin and T. Nipkow. “Boolean Unification — The Story So Far”. *Journal of Symbolic Computation* 7 (1989). Reprinted in C. Kirchner, *Unification*, Academic Press (1990), 437–455, pp. 275–293.

[123] Mathlib Community. “The Lean mathematical library”. In: *Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs*. ACM, 2020.

[124] J. P. McCammond. “Normal forms for free aperiodic semigroups”. *Int. J. Algebra Comput.* 11.5 (2001), pp. 581–625.

[125] J. P. McCammond. “The solution to the word problem for the relatively free semigroups satisfying $T^a = T^{a+b}$ with $a \geq 6$”. *Internat. J. Algebra Comput.* 1.1 (1991), pp. 1–32.

[126] R. McNaughton. *Symbolic Logic and Automata*. Tech. rep. Wright-Paterson Air Force Base, 1960.

[127] R. McNaughton and S. Papert. *Counter-Free Automata*. Cambridge, Mass.: The MIT Press, 1971.

[128] G. Metcalfe and L. Reggio. “Model Completions for Universal Classes of Algebras: Necessary and Sufficient Conditions”. *The Journal of Symbolic Logic* 88.1 (2023), pp. 381–417.

[129] L. S. Moss. “Finite models constructed from canonical formulas”. *Journal of Philosophical Logic* 36 (2007), pp. 605–640.

[130] A. W. Mostowski. “Regular expressions for infinite trees and a standard form of automata”. In: *Computation Theory*. Ed. by A. Skowron. Berlin, Heidelberg: Springer Berlin Heidelberg, 1985, pp. 157–168.

[131] D. Mundici. *Advanced Łukasiewicz calculus and MV-algebras*. Vol. 35. Springer Science & Business Media, 2011.

[132] L. Nachbin. *Topology and order*. van Nostrand, 1964, p. 122.

[133] R. Nederpelt and H. Geuvers. *Type Theory and Formal Proof: An Introduction*. Cambridge University Press, 2014.

[134] D. Pattinson. “Coalgebraic modal logic: soundness, completeness and decidability of local consequence”. *Theoretical Computer Science* 309.1 (2003), pp. 177–193.

[135] D. Perrin. “Les débuts de la théorie des automates”. *Séminaire de Philosophie et Mathématiques* 1 (1993), pp. 1–17.

[136] J.-É. Pin and P. Weil. “Profinite semigroups, Mal’cev products, and identities”. *Journal of Algebra* 182.3 (1996), pp. 604–626.

[137] J.-É. Pin, ed. *Handbook of automata theory*. Vol. 1 and 2. EMS Press, 2021.

[138] J.-É. Pin. “Syntactic semigroups”. In: *Handbook of language theory*. Ed. by G. Rozenberg and S. A. Vol. 1. Springer Verlag, 1997, pp. 679–746.

[139] A. M. Pitts. “On an interpretation of second-order quantification in first-order intuitionistic propositional logic”. *J. Symbolic Logic* 57.1 (1992), pp. 33–52.

[140] T. Place and M. Zeitoun. “Dot-depth three, return of the J-class”. In: *Proceedings of the 39th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS’24*. 2024, pp. 1–15.

[141] T. Place and M. Zeitoun. “Separating regular languages with first-order logic”. In: *Proceedings of the Joint Meeting of the Twenty-Third EACSL Annual Conference on Computer Science Logic (CSL) and the Twenty-Ninth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*. Vienna, Austria: Association for Computing Machinery, 2014.

[142] T. Place and M. Zeitoun. “The Covering Problem”. *Log. Methods Comput. Sci.* 14.3 (2018).

[143] H. A. Priestley. “Representation of distributive lattices by means of ordered Stone spaces”. *Bull. London Math. Soc.* 2 (1970), pp. 186–190.

[144] M. Rabin and D. Scott. "Finite automata and their decision problems". *IBM J. Res. and Develop.* 3 (1959), pp. 114–125.

[145] M. O. Rabin. "Decidability of second-order theories and automata on infinite trees." *Transactions of the american Mathematical Society* 141 (1969), pp. 1–35.

[146] L. Reggio and C. Riba. *Finitely accessible arboreal adjunctions and Hintikka formulae.* Preprint. 2023. URL: https://arxiv.org/abs/2304.12709.

[147] J. Reiterman. "The Birkhoff theorem for finite algebras". *Algebra Universalis* 14 (1982), pp. 1–10.

[148] J. Rhodes and B. Steinberg. *The q-theory of Finite Semigroups.* Springer, 2009.

[149] M. Roberts, W. Hayes, B. R. Hunt, S. M. Mount, and J. A. Yorke. "Reducing storage requirements for biological sequence comparison". *Bioinformatics* 20.18 (July 2004), pp. 3363–3369.

[150] A. Robinson. *Introduction to model theory and to the metamathematics of algebra.* Studies in logic and the foundations of mathematics. North-Holland, 1963.

[151] A. Robinson. *On the Metamathematics of Algebra.* North-Holland, 1951.

[152] C. F.-M. Sainte-Marie. "Question 48". *L'intermédiaire des Mathématiciens* 1 (1894), pp. 107–110.

[153] L. Santocanale and Y. Venema. "Completeness for flat modal fixpoint logics". *Annals of Pure and Applied Logic* 162.1 (2010), pp. 55–82.

[154] A. Saurin. *Interpolation as Cut-introduction.* Draft. 2024. URL: https://www.irif.fr/_media/users/saurin/pub/interpolation_as_cut_introduction.pdf.

[155] S. Schleimer, D. S. Wilkerson, and A. Aiken. "Winnowing: local algorithms for document fingerprinting". In: *Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data.* SIGMOD '03. San Diego, California: Association for Computing Machinery, 2003, pp. 76–85.

[156] M. P. Schützenberger. "On Finite Monoids Having Only Trivial Subgroups". *Information and Control* 8.2 (1965), pp. 190–194.

[157] M. P. Schützenberger. "Une théorie algébrique du codage". *Séminaire Dubreil. Algèbre et théorie des nombres* 9 (1956), pp. 1–24.

[158] S. Shelah. "The monadic theory of order". *Ann. of Math.* 102 (1975), pp. 379–419.

[159] I. Shillito, I. van der Giessen, R. Goré, and R. Iemhoff. "A New Calculus for Intuitionistic Strong Löb Logic: Strong Termination and Cut-Elimination, Formalised". In: *Automated Reasoning with Analytic Tableaux and Related Methods.* Ed. by R. Ramanayake and J. Urban. Cham: Springer Nature Switzerland, 2023, pp. 73–93.

[160] T. P. Speed. "Profinite posets". *Bulletin of the Australian Mathematical Society* 6.2 (1972), pp. 177–183.

[161] B. Steinberg. "On Pointlike Sets and Joins of Pseudovarieties". *International Journal of Algebra and Computation* 08.02 (1998), pp. 203–231.

[162] B. Steinberg. "Pointlike Sets and Separation: A Personal Perspective". In: *Developments in Language Theory - 25th International Conference, DLT 2021, Porto, Portugal, August 16-20, 2021, Proceedings.* Ed. by N. Moreira and R. Reis. Vol. 12811. Lecture Notes in Computer Science. Springer, 2021, pp. 27–40.

[163] M. H. Stone. "Topological representations of distributive lattices and Brouwerian logics". *Čas. Mat. Fys.* 67 (1938), pp. 1–25.

[164] H. Straubing. *Finite Automata, Formal Logic, and Circuit Complexity.* Birkhauser, 1994.

[165] H. Straubing. "First-order logic and aperiodic languages: a revisionist history". *ACM SIGLOG News* 5.3 (2018), pp. 4–20.

[166] M. Sweering. *Deciding homomorphic images of De Bruijn graphs.* Answer to a MathOverflow question of S. v. Gool and J. Marti. 2023. URL: https://mathoverflow.net/q/452566.

[167] G. Takeuti. *Proof Theory (Second edition).* North-Holland, 1987.

[168] The Coq Development Team. *The Coq Reference Manual – Release 8.19.0.* 2024.

[169] W. Thomas. "Languages, Automata, and Logic". In: *Handbook of Formal Languages.* Springer, 1996, pp. 389–455.

[170] W. Thomas. "Ehrenfeucht, Vaught, and the Decidability of the Weak Monadic Theory of Successor". *ACM SIGLOG News* 5.1 (2018), pp. 13–18.

[171] B. A. Trakhtenbrot. “Synthesis of logical nets whose operators are given by one-place predicate calculus (in Russian)”. *Doklady Akademia Nauk SSR* 118.4 (1958).

[172] Y. Venema. *Lectures on the modal μ-calculus*. Lecture notes. URL: https://staff.science.uva.nl/y.venema/teaching/ml/notes/20231215-mu.pdf.

[173] A. Visser. “Uniform interpolation and layered bisimulation”. In: *Gödel ’96 proceedings*. Ed. by P. Hájek. Vol. 6. Lecture Notes in Logic. Springer-Verlag, 1996, pp. 139–164.

[174] A. Visser. “Aspects of Diagonalization & Provability”. PhD thesis. Utrecht University, 1981.

[175] A. Visser and T. Litak. *Lewis and Brouwer meet Strong Löb*. Preprint arXiv:2404.11969. 2024.

[176] N. N. Vorobev. “A new algorithm for derivability in the constructive propositional calculus”. *American Mathematical Society Translations*. 2nd ser. 94 (1970). Translated from the 1952 Russian original, pp. 37–71.

[177] I. Walukiewicz. “Completeness of Kozen’s axiomatisation of the propositional μ-calculus”. *Information and Computation* 157.1-2 (2000), pp. 142–182.

[178] W. H. Wheeler. “Model-companions and definability in existentially complete structures”. *Israel Journal of Mathematics* 25 (1976), pp. 305–330.

[179] F. Wolter and M. Zakharyaschev. “Undecidability of the Unification and Admissibility Problems for Modal and Description Logics”. *ACM Trans. Comput. Logic* 9.4 (2008).

[180] J. Worrell. “Terminal Sequences for Accessible Endofunctors”. *Electronic Notes in Theoretical Computer Science* 19 (1999), pp. 24–38.