Stone Duality

An application in the theory of formal languages

A Master thesis by
Mirte Dekkers
Under supervision of
Prof. dr. Mai Gehrke

IN·DEI·NOMINE·FELICITER

Stone duality

An application in the theory of formal languages

Mirte Dekkers

December 2008

Supervisor: Prof. Dr. Mai Gehrke
Second reader: Dr. Wieb Bosma

Faculty of Science
Radboud University Nijmegen
Student number: 0229032

To my parents

Preface

This thesis is the final work of my study of Mathematics at the Radboud University Nijmegen. After 6 years and three months of hard work I am just a few days away from finishing. The people around me know that I have often doubted the fact whether or not I would ever become a real mathematician. And still, while writing this preface, I can hardly believe that I did it! There has been a lot of struggle and doubt along the way and I would like to use these first pages of my thesis to express my gratitude towards those who have helped me to find my path during the last couple of years.

To start at the beginning, I would like to thank Meron and Daan who made me come to Nijmegen in the first place. Your craziness and warm welcome during my first visit and the first years of studying are something I will not easily forget. You taught me all the things I needed to know to become a real mathematician: LaTeX, fingerslapping, exploring the basements, surviving 'gorefests', gunning, and many more crazy things. Meron, you became a true friend during these last years. Thanks for your loyal friendship and support.

Secondly I want to thank all the staff members of the department of Mathematics. I have always appreciated the open atmosphere in the department and the informal contact between students and staff. A special thanks of course goes to Mai, my thesis supervisor. Thank you so much for your help and support, both mathematically and personally. As you know, until very recently, I did not fully believe that I could do this, and I could certainly not have done it without your help. It was great to work with you and I could not have wished for a better supervisor than you! I also want to thank Wieb for being my second advisor and a special thanks to Klaas who's enthusiasm and encouragement during the last years have been of great support and have contributed to my self-confidence.

And then to my fellow students and friends. Dion, it was an honour to be in the organizing team of the Mathematics Tournament with you. Thanks for always being interested in me, encouraging me to keep on studying and for going through my thesis carefully. I am happy that we will continue to stay colleagues for some more years. Lorijn, thanks for your support. I just got to know you quite recently, but I am really happy I did. I like your down-to-earth opinion, your relaxed way of living and your sense of humor. I think that, at least as far as mathematics concerns, we have a lot in common. So please believe in yourself for, as this thesis shows, you can do more than you think! Tanneke, thanks for being such a warm and loving person. Your positive attitude towards life is amazing and inspiring. I have seen you grow during the last years and from a 'little sister' you became a true and loving friend. A big hug for you. And last, but certainly not least, Renée. What can I say? I don't know how many hours we spend together last years. Singing, laughing, talking, eating, studying (not too often lately), drinking (too often lately), watching TV-series, playing games, making plans, philosophizing about where our lives are going. . . or just doing nothing at all. Your friendship is invaluable to me and the past years would not have been the same without you. Thanks for everything!

Finally, I would like to thank my sister Merel and my parents, Daan and Maria, to whom this thesis is dedicated. This thesis could not have been written if it weren't for you. No words can express my gratitude for your unconditional love and support.

Mirte Dekkers

Nijmegen, December 2008.

Contents

Introduction

The central question in this thesis is what the duality between Boolean algebras and certain topological spaces, known as Stone duality, has to do with the theory of formal languages. In addition, we want to see whether the connection with Stone Duality can be used to generalize the tools available for the so-called regular languages to a wider setting. We will see that certain classes of languages can be characterized by sets of equations and that determining whether or not a given language belongs to such a class, can be done by verifying that its so-called *syntactic Stone frame* satisfies these equations. The syntactic Stone frame is a generalization of the notion of syntactic semigroup in the case of the regular languages. This result is the core of this thesis and will be presented in part IV. We illustrate the general theory with an example of a class of languages called the commutative languages.

However, before we get to the main result there is a long way to go. We will introduce concepts and results from the theory of formal languages, the theory of Boolean algebras and the theory of Stone duality.

Part I is about formal languages. In Chapter 1 we introduce the main terminology and present the mathematical framework, that is, the algebraic structure for working with (classes of) formal languages. Furthermore, we give a motivation why one studies classes of languages by showing the relationship between formal languages and the theory of computability. In Chapter 2 we introduce the class of regular languages and the concept of a finite automaton. Also we point out the relationship between regular languages and finite semigroups. The class of regular languages will play a special role in the theory developed in this thesis. The results obtained through duality will turn out to be a generalization of the theory of finite semigroups available for regular languages. Also, the class of regular languages provides a nice illustration of the relationship between languages and models of computation.

Part II is about Boolean algebras. It is an easy observation that the class of all languages over a certain alphabet is a Boolean algebra. Moreover, subclasses of languages correspond to subalgebras. After we have introduced the concept of a Boolean algebra in Chapter 3, we study the special properties of the ordered set underlying a Boolean algebra in Chapter 4. This point of view will be advantageous when we study representations of Boolean algebras in terms of fields of sets in Chapter 5. The representation theorem for Boolean algebras is presented at the end of part II.

Part III shows how the results in part II can be extended to obtain a full duality between the category of Boolean algebras with Boolean homomorphisms and the category of Stone spaces which continuous maps. In Chapter 6 we work out the details for the objects, that is, Boolean algebras and Stone spaces. In Chapter 7 we extend this to a full duality between categories that also captures maps. In addition, we show how to translate algebraic concepts and additional structure to their topological counterparts. In particular we will see that subalgebras correspond to equivalence relations on the dual space and that an additional binary operation on the algebra gives rise to a ternary relation on the dual space. It is this extended duality that lies at the heart of the results presented in part IV.

In part IV we apply the duality theory developed in the previous chapters to the Boolean algebra of formal languages. In Chapter 8 we introduce the concept of quotienting subalgebra of a Boolean algebra and we motivate why classes of languages representing a certain level of complexity are often of this kind. The main content of Chapter 8 is the proof of a general duality result for quotienting subalgebras. More specifically, we show that quotienting subalgebras correspond to so-called R-congruences on the dual space (R being the relation dual to the additional operations). In Chapter 9 we apply the general theory to the class of regular languages. In particular, we relate the duality approach to the finite semigroup theory for regular languages by proving that the dual space of the quotienting subalgebra generated by a regular language L is (isomorphic to) the syntactic semigroup of L. For non-regular languages this no longer holds and this motivates us to generalize the notions of syntactic congruence and syntactic semigroup to the *syntactic Stone congruence* and *syntactic Stone frame.* These concepts are introduced in Chapter 10 and we will illustrate the theory with the example of the class of commutative languages.

Because of the many different areas of study the theory developed in this thesis involves, the introduction to each of these subjects is minimal and by no means complete. The interested reader should consult the specialized literature for a more extensive introduction. The books consulted for this thesis are [14] and [20] on the theory of formal languages, [7] and [12] on Boolean algebras and [3], [6] and [7] on Stone duality.

What is new with this thesis?

Most of the theory in Part I to III is available in the books mentioned above. However, the duality for subalgebras and Stone equivalences worked out in Section 7.3 is not generally available and can only be found in more specialized research papers. Also, the extended duality for Boolean algebras with additional operations presented in Section 7.4 is only in advanced research papers and mainly in terms of canonical extensions. We tried to present these two results using Stone duality in its traditional form, without having to go through the theory of canonical extensions.

The main results that are new with this thesis are in Part IV. Central is the theorem that says that quotienting subalgebras are dually related to R-congruences of the dual space. While this duality result is typical of duality results, it is also quite novel among duality results because the quotienting subalgebras we consider are closed under taking quotients with denominators from outside the subalgebras. This situation has not been considered in duality theory before, and the shape of the result is surprisingly nice. It is a specialization of the relationship between subalgebras and Stone equivalences developed in Section 7.3.

The fact that Stone duality plays a pertinent role in the theory of regular languages was first observed by Pippenger in [16]. A general framework, however, was not worked out until very recently. This was done by Gehrke, Grigorieff and Pin and they presented their results in a prizewinning paper [9] and a more comprehensive paper [10] that has not been published yet. They did not only observe that classes of languages could be characterized by equations on the dual space, but their theory also involves additional operations on the Boolean algebra and the relation dual to them. In this thesis we use the methods and ideas of these two papers in order to generalize the theory available for regular languages.

Historical background

Soon after the introduction of regular languages and finite automata it was discovered that these concepts are closely related to the theory of finite semigroups and an algebraic counterpart of the definition of recognizability by a finite automata in terms of finite semigroups was established. In particular, every language was associated to its so-called syntactic semigroup [17]. This opened the door to study classes of languages through their syntactic semigroups. The first important result using this connection was Schützenberger's classification of the *star-free* languages, a subclass of regular languages that can be obtained by a restricted use of the star-operation [19]. Schützenberger's theorem says that a regular language is star-free if and only if its syntactic semigroup is *aperiodic*. A more general result that subsumes this result was obtained by Eilenberg [8]. Eilenberg's theorem characterized those classes of languages that are given by *pseudo-varieties* of finite semigroups. A pseudo-variety is an analogue for finite algebras of the notion of variety introduced by Birkhoff [4]. It is a class of finite algebras of a given type closed under homomorphic images, subalgebras and finite products. Reiterman gave an equivalent of Birkhoff's Theorem for finite algebras saying that pseudo-varieties correspond to equational classes [18]. Reiterman's theorem already introduced a topology but the connection with duality theory was not clear. Combining the results of Eilenberg and Reiterman yields an equational theory for classes of regular languages.

The application of duality theory in the theory of formal languages places these former results in a more general setting. It turns out that duality theory provides a natural framework to describe the results established by Eilenberg and Reiterman. In particular, a lot of results established through the years now appear to be special instances of a very general theory. The equational characterization of classes of regular languages is just a special case of duality theory. Furthermore, this new framework enables us to extend the results to classes of languages outside the regular languages. In particular this leads to a generalization of the concepts of syntactic congruence and syntactic semigroup for non-regular languages.

Part I

Languages

Chapter 1

Formal languages

In this chapter we introduce the concept of a formal language. We cover the main terminology and notation and describe the mathematical framework for studying classes of formal languages. Finally, we show the connection between formal languages and the theory of computability.

1.1 Introduction

Although the theory of formal languages has been strongly influenced by developments in areas like linguistics and computer science, it has its origins strictly within mathematics. The quest for the foundations of mathematics at the beginning of the 20th century led to a reinvestigation of the notion of mathematical proof. A formalized interpretation of the concept of proof arose, in which the validity of a mathematical argument was established by its structure rather than by the nature of the argument itself. From this point of view, mathematical proofs were studied as sequences of statements for which the ordering of these statements decided whether or not the argument was valid. This led to a structural study of sequences of symbols.

It is the description of properties of *sequences of symbols* that forms the basis of the theory of formal languages. Since many circumstances, both within and outside mathematics, give rise to such sequences, the applications of formal language theory are widely spread. Besides mathematical proofs one can think of computer programs, sentences in natural languages, sequences of molecules and neural processes in the brain as just a few of the numerous examples. All are instances of a set of sequences of symbols from some fixed set A. This observation has lead to the following definition of a formal language.

Definition 1.1 Let A be a finite set of symbols. A **formal language** L over A is a set of finite sequences of symbols in A.

Most of the time the word 'formal' is not included and we will just speak about 'a language'. The set A is called the **alphabet** of the language L. The alphabet is always finite and its elements are usually denoted by the first letters from the Roman alphabet $(a, b, c, \ldots)$. Note that these letters can represent anything from mathematical statements to molecules. As long as it is clear over which alphabet we are working, we will just state 'L is a language' without specifying the alphabet.

A finite sequence of elements of A is called a **word**. The set of all words over some alphabet A is denoted by A^*. To distinguish in notation between elements of the alphabet and sequences of these elements (words), we use the letters $u, v, w \ldots$ to denote words. The empty sequence, denoted by λ, is also an element of A^*. In this notation L is a language (over A) if and only if $L \subseteq A^*$, or equivalently $L \in \mathcal{P}(A^*)$.

Example 1.2 Let $A = \{a, b\}$ be a set of two symbols. Then $A^* = \{\lambda, a, b, aa, ab, ba, bb, aaa, aab, aba, baa, \ldots\}$. Every subset of this set, finite or infinite, determines a language. For example: $L_1 = \{a, ab, ba, aab\}$ and $L_2 = \{u \in A^* \mid u \text{ has an even number of symbols}\}$ are both languages over A.

We conclude this section by defining an operation, called **product** or **concatenation**, on the set A^* of words. Given two words $u = a_1 \ldots a_n$ and $v = b_1 \ldots b_m$ the product of these words is the word $uv = a_1 \ldots a_n b_1 \ldots b_m$. This operation is associative as you can easily prove by induction on the length of the words.

1.2 The algebra of languages

Mathematically spoken we can make two observations about the concepts introduced so far:

1. A language is a *set* of words.
2. The set A^* with the concatenation operation is a *semigroup*.

These observations may not seem that exciting at first sight, but it turns out that it are these two facts that give rise to the algebraic structure of the class of formal languages.

Let A be a finite alphabet and L_1 and L_2 languages over this alphabet. The first observation tells us that we can apply the usual set theoretic operations, like union, intersection and complement on languages. More specifically, we have:

$$\begin{aligned} L_1 \cup L_2 &= \{u \in A^* \mid u \in L_1 \text{ or } u \in L_2\} \\ L_1 \cap L_2 &= \{u \in A^* \mid u \in L_1 \text{ and } u \in L_2\}. \end{aligned}$$

To define the complement of a language we need to take into account over which alphabet we are working. That is, let L be a language over A, then

$$L^c = \{u \in A^* \mid u \notin L\}.$$

Note that $L_1 \cup L_2, L_1 \cap L_2$ and L^c are again languages over the alphabet A. In other words, the set-theoretic operations $\cap, \cup$ and $()^c$ give rise to two binary and one unary operation on the set $\mathcal{P}(A^*)$ of all languages over A. Adding the two nullary operations (constants) $\emptyset$ and A^*, we obtain the algebraic structure $\langle \mathcal{P}(A^*), \cup, \cap, ()^c, \emptyset, A^* \rangle$. It is an easy observation that this algebraic structure is a *Boolean algebra*. Boolean algebras are central in this work and we will study them in chapter II.

So far we have just used the first observation. But what about the semigroup structure of A^*? In addition to the operations introduced above, we can endow the set $\mathcal{P}(A^*)$ with an additional operation arising from the semigroup structure of A^*. That is, we can 'lift' the product on words to a product on languages. For $K, L \in \mathcal{P}(A^*)$ we define

$$K \cdot L = \{uv \in A^* \mid u \in K \text{ and } v \in L\}.$$

It is easy to show that this operation is associative. To shorten notation we often write KL instead of $K \cdot L$.

Furthermore we define two quotient operations, / and \, on the set $\mathcal{P}(A^*)$. Let $K, L \in \mathcal{P}(A^*)$. We define

$$\begin{aligned} L/K &= \{u \in A^* \mid uv \in L \text{ for all } v \in K\} \\ K\backslash L &= \{u \in A^* \mid vu \in L \text{ for all } v \in K\}. \end{aligned}$$

The relationship between the operations on $\mathcal{P}(A^*)$ defined above, is expressed in the following theorem.

Theorem 1.3 *For all $K, L, M \in \mathcal{P}(A^*)$ we have*

$$K \cdot L \subseteq M \Longleftrightarrow K \subseteq M/L \Longleftrightarrow L \subseteq K \backslash M.$$

Proof. This is easily deduced by using the definitions of the operations $\cdot, /$ and $\backslash$. We leave the details to the reader.

□

Those familiar with the concept of residuation will recognize that the triple $(\cdot, /, \backslash)$ forms a so-called *residuated family* on $\mathcal{P}(A^*)$. We come back to this observation in Chapter 8.

1.3 Languages and computation

We will use this last section to motivate the kind of problems we consider in this thesis. First of all we need to point out that it is *classes* of languages that we study, rather than individual languages. It is the tight relationship between the theory of formal languages and the theory of computability that motivates the study of classes of languages. Questions about complexity of models can be translated into questions about classes of languages and vice versa, so answering these questions gives insight in the theory of computability.

The specification of a formal language requires an unambiguous description of which words belong to the language. One way to specify a formal language is through a machine that *recognizes* the language. A machine recognizes L if it can decide for every element $u \in A^*$ whether or not $u \in L$. Such a machine can be of any kind and several computational models to describe machines that recognize formal languages have been developed since the the beginning of the 20th century. Every one of these models gives rise to a class of machines and therefore to a class of languages recognized by these machines. Two models of computation can thus be compared by comparing the classes of languages recognized by instances of these models. For example, given two computational models M_1 and M_2, we can determine whether or not one model is more complex than the other by comparing the classes of languages that are recognized by the models. That is, for $\mathcal{C}_1$ and $\mathcal{C}_2$ being the classes of languages recognized by these models, the observation $\mathcal{C}_1 \subseteq \mathcal{C}_2$

would tell us that the first model is less complex than the second one. Also, given some mathematical decision problem, we can determine whether this problem can be solved using a particular computational model M, by determining whether or not the language L (corresponding to the problem) belongs to the class $\mathcal{C} \subseteq \mathcal{P}(A^*)$ of languages that can be recognized by an instance of the computational model M. Finally, given a formal language L we would like to determine which languages are related to L in terms of complexity.

Using this relationship between languages and models of computation, an important classification of formal languages in terms of complexity was given by the linguist Noam Chomsky in 1956. He used the observation that computing machines of different types can recognize languages of different complexity to construct a hierarchy of computational models and corresponding language types. The simplest model of computation in this hierarchy is a *finite state automaton.* For higher levels more powerful machines, ranging from pushdown automata to Turing machines, are required. The languages that can be recognized by finite automata are called the *regular languages.* As this class of languages plays a special role in our theory, it is the subject of the next chapter.

Chapter 2

Regular languages and finite automata

In this chapter we give a short introduction in the theory of regular languages and finite automata. In the first section we give a recursive definition of the class of regular languages. From this definition it is easily seen that every regular languages can be specified by a regular expression. The second section introduces finite automata and shows their relationship to regular languages. Finally we make an important observation about the syntactic semigroup of a regular language.

2.1 The regular languages

The regular languages are those languages that can be generated from the one-element languages by applying certain operations a finite number of times. Before we can make this idea more precise we need to define an additional operation on the algebra of languages, known as the *star* operator. For $L \in \mathcal{P}(A^*)$ we introduce

$$L^* = \bigcup_{n \in \mathbb{N}} L^n,$$

where $L^n = L^{n-1}L$ is defined inductively.

We can now give a formal definition of the class of regular languages.

Definition 2.1 Let A be an alphabet. The class of **regular languages** over A is defined recursively by the following three steps

i) $\emptyset, \{\lambda\}$ and $\{a\}$, for all $a \in A$, are regular languages.

ii) Let L_1 and L_2 be regular languages. Then $L_1 \cup L_2, L_1L_2$ and L_1^* are regular languages.

iii) Only the languages that can be obtained in this way are regular languages.

In other words, the class of regular language is the smallest class of languages that contains the finite languages and is closed under the operations $\cup, \cdot$ and $()^*$. It can be shown that the class of regular languages is closed under the operations $\cap$ and $()^c$ as well. Furthermore it contains the languages $\emptyset$ and A^*. Hence it is a subalgebra of the algebra $\langle \mathcal{P}(A^*), \cup, \cap, ()^c, \emptyset, A^* \rangle$. The set of all regular languages over the alphabet A is denoted by $\mathrm{Reg}(A^*)$.

Example 2.2 Let $A := \{a, b\}$. Let $L \subseteq A^*$ be the language that contains all words that start with an a and end with a b. By definition $\{a\}$ and $\{b\}$ are regular languages. Applying the operations $\cup$ and $()^*$ gives rise to $\{a, b\}^*$, the set of all strings over A. Using concatenation twice, we get $\{a\}\{a, b\}^*\{b\}$, which represents the language L. So L is regular.

The string $\{a\}\{a, b\}^*\{b\}$ is an example of a **regular expression**. From the definition of a regular language it follows directly that every regular language can be described by such an expression. To obtain a reduction of parenthesis a priority is assigned to the operations, which appoints the star operation as the most binding one, followed by concatenation and finally union. Furthermore the set brackets are left out. The expression $\{a\}\{a, b\}^*\{b\}$ is then abbreviated by $a(a \cup b)^*b$. Note that the regular expression for a certain language is not unique. That is, a regular language can be described by two different regular expressions.

Example 2.3 Define $A := \{a, b\}$. The regular expressions $(a \cup b)^*a(a \cup b)^*a(a \cup b)^*$ and $b^*ab^*a(a \cup b)^*$ both define regular languages over A consisting of all the words that contain at least two a's.

It can be proved that not all languages are regular. In the next section we see an example of a language that is not regular.

2.2 Finite automata

At the end of the previous chapter we briefly discussed the relationship between formal languages and computation. In particular we pointed out that computational models can be associated with the classes of languages they recognize. In this section we will introduce the notion of a *finite automaton.* A finite automaton is a simple computing machine with a very restricted memory. By machine we do not mean a physical piece of hardware but an abstract machine defined mathematically. An important result in the theory of formal languages is the fact that the class of languages recognized by a finite automaton is equal to the class of regular languages.

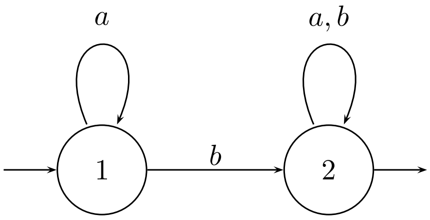
The formal definition of a finite automaton is the following.

Definition 2.4 A **finite automaton** is a quintuple $\mathcal{A} = (Q, A, q, F, \delta)$ where

- Q is a finite set of *states*
- A is a finite *alphabet*
- $q \in Q$ is the *initial state*
- $F \subseteq Q$ is the set of *final states*
- δ is a function from $Q \times A$ to Q, called the *transition function.*

The formal definition of a finite automaton is not a very intuitive one and the best way to get an idea of a finite automaton is by studying a graphical representation of it. We can represent every finite automaton by a directed labeled graph. The vertices of the graph indicate the states of the automaton and the (labeled) edges represent the transitions between states. Furthermore we indicate the initial state with an incoming arrow and the final states with outgoing arrows. Let us start with an easy example.

Example 2.5 Let $\mathcal{A}_1 = (\{1,2\}, \{a,b\}, 1, \{2\}, \delta)$ be an automaton with $\delta = \{(1,a,1),(1,b,2),(2,a,2),(2,b,2)\}$. This automaton is given by the following graph.



A **path** in an automaton is a finite walk along the the edges of the graph that represents the automaton. In terms of transitions a path is a finite sequence of transitions

$$\delta_1 = (q_0, a_0, q_0'), \delta_2 = (q_1, a_2, q_1'), \ldots, \delta_n = (q_n, a_n, q_n')$$

such that

$$q_0' = q_1, q_1' = q_2, \ldots, q_{n-1}' = q_n.$$

We denote a path also by

$$q_0 \xrightarrow{a_0} q_1 \xrightarrow{a_1} q_2 \ldots \xrightarrow{a_{n-1}} q_n$$

We can now define what it means for a word u in A^* to be accepted by a finite automaton.

Definition 2.6 Let $\mathcal{A}$ be a finite automata and $u \in A^*$. We say that u is **accepted** by $\mathcal{A}$ if and only if there is a path

$$q_0 \xrightarrow{a_0} q_1 \xrightarrow{a_1} q_2 \ldots \xrightarrow{a_{n-1}} q_n$$

in $\mathcal{A}$ such that q_0 is the initial state, q_n is one of the final states and $u = a_0 a_1 \ldots a_{n-1}$.

Example 2.7 Observe that

$$1 \xrightarrow{a} 1 \xrightarrow{b} 2 \xrightarrow{a} 2 \xrightarrow{b} 2$$

is a path in $\mathcal{A}_1$. Hence the word $abab$ is accepted by $\mathcal{A}_1$. Also

$$1 \xrightarrow{a} 1 \xrightarrow{a} 1$$

is a path in $\mathcal{A}_1$ but 1 is not a final state, hence aa is not accepted by $\mathcal{A}_1$. It is not hard to see that a word is accepted by $\mathcal{A}_1$ if and only if it contains at least one b.

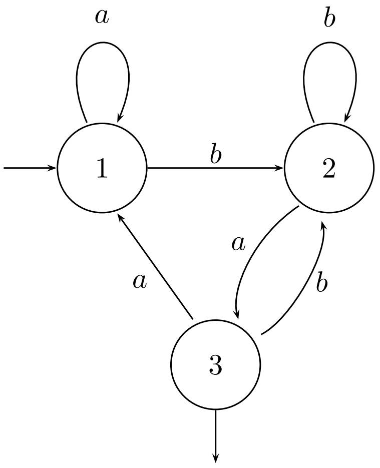
Definition 2.8 Let $\mathcal{A}$ be a finite automaton. The language **recognized** by $\mathcal{A}$ is the set of words accepted by $\mathcal{A}$. We denote this language by $L(\mathcal{A})$.

Example 2.9 The language recognized by $\mathcal{A}_1$ consists of all the words in A^* that contain at least one b. That is,

$$L(\mathcal{A}_1) = a^* b (a \cup b)^*.$$

To get familiar with finite automata we consider another example.

Example 2.10 Let $\mathcal{A}_2$ be the following automaton.



The paths

$$1 \xrightarrow{a} 1 \xrightarrow{b} 2 \xrightarrow{a} 3$$

and

$$1 \xrightarrow{b} 2 \xrightarrow{a} 3 \xrightarrow{b} 2 \xrightarrow{a} 3$$

indicate that the words *aba* and *baba* are in the language. The path

$$1 \xrightarrow{a} 1 \xrightarrow{b} 2 \xrightarrow{a} 3 \xrightarrow{b} 2$$

indicates that the word *abab* is not in the language. Determining for some more words whether or not they are accepted by $\mathcal{A}_2$ should convince you that the language recognized by $\mathcal{A}_2$ consists of all the words that end with ba. That is,

$$L(\mathcal{A}_2) = (a \cup b)^* ba.$$

In our examples we only considered automata with one final state. However, note that the definition of a finite automata allows there to be more final states.

The languages recognized by the automata in the examples 2.5 and 2.10 could both be specified by a regular expression and hence were both regular. A surprising result in the theory of formal languages is that this is the case for every finite automaton. On the other hand, given a regular language we can construct an automaton that recognizes this languages. This correspondence between finite automata and regular languages is a central result in the theory of formal languages and is known as Kleene's theorem.

Theorem 2.11 *(Kleene) A language is regular if and only if it is recognized by a finite automata.*

This theorem says that if a language can be generated in a simple way, it can also be decided in a simple way, and vice versa. The proof of this theorem can be found in any standard work on formal languages and automata. See for example [14] or [20]. Among other things Kleene's theorem gives a tool for proving that not every language is regular.

Example 2.12 The language $L = \{a^i b^i \mid i \in \mathbb{N}\}$ is not regular.

This can be proved by showing that there is no finite automaton that recognizes L. The proof can be found in [20].

2.3 Regular languages and finite semigroups

The theory of regular languages is closely related to the theory of finite semigroups. In this section we will see that every formal language L gives rise to a congruence $\sim_L$ on A^* and hence to a semigroup $A^*/{\sim_L}$.

Let $L \subseteq A^*$ be a language. We define a relation $\sim_L$ on A^* by

$$u \ \sim_L v \quad \text{if and only if, for all } s, t \in A^* : sut \in L \Leftrightarrow svt \in L.$$

It is easily checked that $\sim_L$ is an equivalence relation on L. In addition we can prove that $\sim_L$ 'respects' the product on A^*. That is, $\sim_L$ is a *congruence* on the semigroup $\langle A^*, \cdot\rangle$.

Definition 2.13 Let $\langle S, \cdot\rangle$ be a semigroup and let $\sim$ be an equivalence relation on S. Then $\sim$ is a **congruence** on S if and only if for all $s_1, s_2, t_1, t_2 \in S$

$$s_1 \sim t_1 \text{ and } s_2 \sim t_2 \text{ implies } s_1 \cdot s_2 \sim t_1 \cdot t_2.$$

The following lemma shows that $\sim_L$ is a congruence on A^*.

Lemma 2.14 *Let $L \subseteq A^*$ be a language and $\sim_L$ as defined above. Then $\sim_L$ is a congruence on A^*.*

Proof. Let u_1, u_2, v_1, v_2 such that $u_1 \sim_L v_1$ and $u_2 \sim_L v_2$. We have for all $s, t \in A^*$

$$\begin{aligned} su_1u_2t \in L &\Leftrightarrow sv_1u_2t \in L \\ &\Leftrightarrow sv_1v_2t \in L. \end{aligned}$$

Hence $u_1u_2 \sim_L v_1v_2$.

□

The congruence $\sim_L$ is called the **syntactic congruence** of L.

It is a well-known fact in algebra that any congruence on an algebraic structure defines an algebraic structure on the set of equivalence classes. In particular, if $\langle S, \cdot\rangle$ is a semigroup and $\sim$ a congruence relation, then the operation on $S/\sim$ defined by for all $s, t \in S$

$$s/\sim \cdot\, t/\sim = (s \cdot t)/\sim$$

makes $S/\sim$ into a semigroup. The fact that this operation is well-defined on $S/\sim$ relies exactly on the requirement that $\sim$ is a congruence relation. When it is clear which congruence relation is involved, the equivalence class $s/\sim$ is often denoted by $\overline{s}$.

As the equivalence relation $\sim_L$ is a congruence on the semigroup A^*, it gives rise to a semigroup structure on $A^*/\sim_L$. This semigroup is called the **syntactic semigroup** of L. For regular languages we have the following theorem.

Theorem 2.15 *A language $L \subseteq A^*$ is regular if and only if its syntactic congruence is of finite index. Or equivalently, its syntactic semigroup is finite.*

The proof of this theorem can be found in [14]. It can be derived from the fact that every regular languages can be recognized by finite automaton as is the case by Kleene's theorem.

The observation that regular languages correspond to finite semigroups has given rise a number of characterizations of classes of regular languages by their syntactic semigroups. For example, Schützenberger proved that a regular language is star-free if and only if its syntactic semigroup is *aperiodic* [19]. More generally, Eilenberg gave a characterization of those classes of languages that are given by *pseudo-varieties* of finite semigroups [8]. In part IV we will see that the finite semigroup theory for regular languages is a specialization of the general duality theory for formal languages.

Part II

Boolean Algebras

Chapter 3

Boolean algebras

In this chapter we will introduce the concept of a Boolean algebra. This algebraic structure arises naturally in the study of classes of formal languages and hence plays a central role in this work. We will discuss some elementary results and introduce the concepts of Boolean subalgebra and Boolean homomorphism.

3.1 Boolean algebras

Boolean algebras were introduced by George Boole in the 1850's to study the laws of logic. Since then they have arisen in many different situations and have been studied extensively. Let us first give a formal definition of the concept of a Boolean algebra.

Definition 3.1 A **Boolean algebra** $\langle B, \vee, \wedge, \neg, 0, 1\rangle$ is a set with two binary operations, one unary operation, and two nullary operations such that for all $a, b, c \in B$:

i) (commutativity)

   (a) $a \vee b = b \vee a$
   (b) $a \wedge b = b \wedge a$

ii) (associativity)

   (a) $a \vee (b \vee c) = (a \vee b) \vee c$
   (b) $a \wedge (b \wedge c) = (a \wedge b) \wedge c$

iii) (idempotency)

(a) $a \vee a = a$
(b) $a \wedge a = a$

iv) (absorption)

(a) $a = a \vee (a \wedge b)$
(b) $a = a \wedge (a \vee b)$

v) (distributivity)

(a) $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$
(b) $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$

vi) (top and bottom)

(a) $a \wedge 0 = 0$
(b) $a \vee 1 = 1$

vii) (complementation)

(a) $a \wedge \neg a = 0$
(b) $a \vee \neg a = 1$

Example 3.2 Let X be a set. Then $\langle \mathcal{P}(X), \cup, \cap, ()^c, \emptyset, X \rangle$ is a Boolean algebra. In particular, $\langle \mathcal{P}(A^*), \cup, \cap, ()^c, \emptyset, A^* \rangle$, the class of all languages over some alphabet A, together with the operations union, intersection and complement, is a Boolean algebra.

Definition 3.3 Let X be a set. A **field of sets** over X is a non-empty subset of $\mathcal{P}(X)$ closed under union, intersection and complementation.

Let V be a subset of $\mathcal{P}(X)$ that forms a field of sets. Then note that the non-emptyness of V implies that $\emptyset$ and X are in V.

Example 3.4 We have seen that the set of regular languages is closed under union, intersection and complement. Hence $\mathrm{Reg}(A^*)$ is a field of sets over A^*.

Lemma 3.5 *Let X be a set and $V \subseteq \mathcal{P}(X)$ a field of sets over X. Then $\langle V, \cup, \cap, ()^c, \emptyset, X \rangle$ is a Boolean algebra.*

Proof. It is an easy exercise to check that the algebra $\langle V, \cup, \cap, ()^c, \emptyset, X \rangle$ satisfies conditions *i)*-*vii)* from definition 3.1.

□

Corollary 3.6 $\langle Reg(A^*), \cup, \cap, ()^c, \emptyset, A^* \rangle$ *is a Boolean algebra.*

When, for a certain set B, there is no confusion about which operations are taken to make it a Boolean algebra, we will not specify them every time. For example, if V is a field of sets we will say 'V is a Boolean algebra' instead of '$\langle V, \cup, \cap, ()^c, \emptyset, A^* \rangle$' is a Boolean algebra. Later on in Chapter 5 we will prove that, up to isomorphism, every Boolean algebra is isomorphic to a field of sets. And therefore, up to isomorphism, the only operations involved are the set-theoretic operations.

3.2 The principle of duality

An observation one can make on the definition of a Boolean algebra is that interchanging the roles of $\vee$ and $\wedge$ and of 0 and 1 does not change the set of identities. Consequently, the same is true for all results deduced from these identities. That is, facts about Boolean algebras come in dual pairs. It is always sufficient to prove only one half of each pair, the other half is obtained by interchanging $\vee$ and $\wedge$ and 0 and 1. This observation is known as the **principle of duality**.

One word of warning on the terminology: although the word 'duality' in mathematics always refers to something being turned 'upside down', it may refer to different instances of this concept. The duality we are talking about here considers the algebraic operations of a Boolean algebra. Much more important in this work is another meaning of the term duality, which will be the subject of part III. In this duality, not the structures themselves are turned upside down but the structure preserving maps between these structures are turned around. We will see that this gives rise to a duality of *categories* between the category of Boolean algebras with Boolean homomorphisms and the category of so-called Stone spaces with continuous maps. Another usage of the word duality we encounter in the theory of ordered sets, where the dual order is obtained by 'flipping the order upside down'. Order duality is actually closely related to the algebraic duality, because of the tight connection between Boolean algebras and ordered sets. We will go into detail about this later.

Often the context makes clear what meaning of the word 'duality' we are referring to. If not, we will explicitly use the terms algebraic, topological or order duality.

3.3 Subalgebras and homomorphisms

The concepts of subalgebra and homomorphism are central in the study of algebras. In particular in this work these concepts will turn out to play an important role.

Definition 3.7 A non-empty subset $A \subseteq B$ is a **Boolean subalgebra** of a Boolean algebra $\langle B, \vee, \wedge, \neg, 0, 1\rangle$ if A is closed under $\vee, \wedge$ and $\neg$.

Example 3.8 Any field of subsets over X is a Boolean subalgebra of $\mathcal{P}(X)$. In particular, $\mathrm{Reg}(A^*)$ is a Boolean subalgebra of $\mathcal{P}(A^*)$.

Definition 3.9 Let B and C be Boolean algebras. A map $f : B \to C$ is a **Boolean homomorphism** if f preserves $\vee, \wedge$ and $\neg$. That is, for all $a, b \in B$,

i) $f(a \vee b) = f(a) \vee f(b)$

ii) $f(a \wedge b) = f(a) \wedge f(b)$

iii) $f(\neg a) = \neg f(a)$.

Furthermore, if f is one-to-one it is called a **(Boolean) embedding**. If f is a bijection it is called a **(Boolean) isomorphism**.

Since Boolean algebras are the pertinent algebraic structures in this work we will just talk about subalgebras and homomorphism instead of Boolean subalgebras and Boolean homomorphisms.

The next lemma gives an equivalent characterization of the last condition in the definition above.

Lemma 3.10 *Let $f : B \to C$ be a $\vee$- and $\wedge$-preserving map. Then the following are equivalent:*

i) $f(0) = 0$ *and* $f(1) = 1$

ii) $f(\neg a) = \neg f(a)$ *for all* $a \in B$.

Proof. Suppose (i) holds. Then

$$\begin{aligned} f(a) \wedge f(\neg a) &= f(a \wedge \neg a) = f(0) = 0, \\ f(a) \vee f(\neg a) &= f(a \vee \neg a) = f(1) = 1. \end{aligned}$$

Thus, $f(\neg a) = \neg f(a)$.
Suppose (ii) holds. Then

$$\begin{aligned} f(0) &= f(a \wedge \neg a) = f(a) \wedge f(\neg a) = f(a) \wedge \neg f(a) = 0, \\ f(1) &= f(a \vee \neg a) = f(a) \vee f(\neg a) = f(a) \vee \neg f(a) = 1. \end{aligned}$$

□

The next lemma states that a homomorphism between Boolean algebras B and C gives rise to a subalgebra of C.

Lemma 3.11 *Let B and C be Boolean algebras and $f : B \to C$ a homomorphism. Then $f(B)$ is a subalgebra of C.*

Proof. Suppose $c_1, c_2 \in f(B)$. Then there exist $b_1, b_2 \in B$ such that $c_1 = f(b_1)$ and $c_2 = f(b_2)$. We have

$$c_1 \vee c_2 = f(b_1) \vee f(b_2) = f(b_1 \vee b_2) \in f(B).$$

By a similar argument $c_1 \wedge c_2 \in f(B)$. Furthermore $\neg c_1 = \neg f(b_1) = f(\neg b_1) \in f(B)$. So $f(B)$ is a subalgebra of C.

□

Also, if A a subalgebra of B, then the inclusion of A into B is an (injective) homomorphism of A into B.

Chapter 4

Boolean algebras as ordered sets

We introduced Boolean algebras from a purely algebraic point of view. Boolean algebras can however also be thought of as special kinds of ordered sets. In this chapter we will explore the connection between Boolean algebras and ordered sets. We show how to obtain an ordered set from a Boolean algebra and deduce special properties of the ordered sets arising in this way.

4.1 Ordered sets

Let us first give a formal definition of an ordered set.

Definition 4.1 Let P be a set and $\leq$ a binary relation on P. Then $(P, \leq)$ is called a **(partially) ordered set** or **poset** if for all $x, y, z \in P$:

1. $x \leq x$ ($\leq$ is reflexive)
2. $x \leq y$ and $y \leq x$ implies $x = y$ ($\leq$ is anti-symmetric)
3. $x \leq y$ and $y \leq z$ implies $x \leq z$ ($\leq$ is transitive).

If in addition for all $x, y \in P$, either $x \leq y$ or $y \leq x$, then P is called a **totally ordered set** or **chain**. The relation $\leq$ is called a **partial order**.

Usually we shall say 'P is an ordered set' without specifying the order, when it is clear which order is to be considered.

Example 4.2 $\langle \mathbb{N}, \leq \rangle$ and $\langle \mathbb{N}, | \rangle$ are familiar examples of posets, of which the first one is also a totally ordered set but the second one is not.

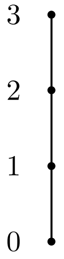
Example 4.3 Given a set X, $\langle \mathcal{P}(X), \subseteq \rangle$ is a poset. Also $\langle V, \subseteq \rangle$ is a poset for every $V \subseteq \mathcal{P}(X)$. More generally, any subset of a poset is again a poset.

For $x, y \in P$ we write $x < y$ to denote $x \leq y$ and $x \neq y$.

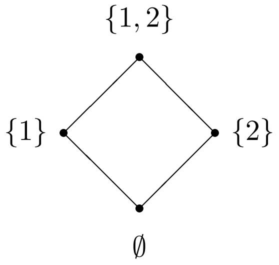
Definition 4.4 Let P be a poset and $x, y \in P$. We say that y **covers** x, if $x < y$ and $x \leq z \leq y$ implies $x = z$ or $y = z$. We denote this by $x \prec y$.

Using this covering relation we can represent a finite ordered set P by a **diagram**. Every element of P is represented by a dot and there is a line between two elements x and y if an only if $x \prec y$. Furthermore, if this is the case, the element x is drawn below y.

Example 4.5 The ordered set $\langle \{0, 1, 2, 3\}, \leq \rangle$ is represented by



Example 4.6 The ordered set $\langle \{\emptyset, \{1\}, \{2\}, \{1, 2\}\}, \subseteq \rangle$ is represented by



Definition 4.7 Let P be an ordered set. The **dual**, P^δ, of P is obtained by 'flipping around' the order. That is,

$$x \leq y \text{ in } P^\delta \Leftrightarrow y \leq x \text{ in } P.$$

The diagram of P^δ is obtained by turning the diagram of P upside down.

A special class of ordered sets are the ones that are bounded.

Definition 4.8 The smallest element of a poset P, if it exists, is denoted by 0. That is, $0 \leq x$ for all $x \in P$. The greatest element of a poset, if it exists, is denoted it by 1. That is, $x \leq 1$ for all $x \in P$. A poset that has both a 0 and a 1 we call **bounded**.

There are several notions of structure preserving maps between ordered sets.

Definition 4.9 Let P and Q be ordered sets. A map $\phi : P \to Q$ is called

i) **order-preserving** if $x \leq y$ in P implies $\phi(x) \leq \phi(y)$ in Q;

ii) **order-reversing** if $x \leq y$ in P implies $\phi(y) \leq \phi(x)$ in Q.

Definition 4.10 Let P and Q be ordered sets. A map $\phi : P \to Q$ is called an **order-embedding** if $x \leq y$ in P if and only if $\phi(x) \leq \phi(y)$ in Q.

Note that from this definition it follows that an order-embedding is injective. If it is, in addition, onto, it is called an **order-isomorphism**.

4.2 Boolean algebras as ordered sets

The next theorem states that there is an easy way to turn a Boolean algebra into a partially ordered set.

Theorem 4.11 *Let $\langle B, \vee, \wedge, \neg, 0, 1\rangle$ be a Boolean algebra and let $\leq$ denote the binary relation on B defined as follows. For all $a, b \in B$*

$$a \leq b \textit{ if and only if } a \vee b = b. \qquad \textit{(ord)}$$

Then $\leq$ is a partial order on B.

Proof. By idempotency of $\vee$, we have $a \vee a = a$ for all $a \in B$, so $\leq$ is reflexive. Furthermore, suppose $a \leq b$ and $b \leq a$. This, together with commutativity of $\vee$, implies $b = a \vee b = b \vee a = a$. Hence, $\leq$ is anti-symmetric. Finally, let $a \leq b$ and $b \leq c$. Then $a \vee b = b$ and $b \vee c = c$. Together with associativity of $\vee$ this implies $a \vee c = a \vee (b \vee c) = (a \vee b) \vee c = b \vee c = c$. Thus $a \leq c$. That is, $\leq$ is transitive.

□

Proposition 4.12 *The ordered set coming from a Boolean algebra is bounded.*

Proof. We have $0 \vee b = b$ for all $b \in B$. Hence, $0 \leq b$ for all $b \in B$. Also $a \vee 1 = 1$ for all $a \in B$. That is, $a \leq 1$ for all $a \in B$.

□

The following lemma gives two alternative characterization of the order on B.

Lemma 4.13 *Let B be a Boolean algebra. Then the following are equivalent:*

i) $a \vee b = b$

ii) $a \wedge b = a$

iii) $a \wedge \neg b = 0$.

Proof. Suppose $a \vee b = b$. Then

$$\begin{aligned} a \wedge b &= a \wedge (a \vee b) \\ &= a \text{ (by absorption).} \end{aligned}$$

Now suppose $a \wedge b = a$. Then

$$\begin{aligned} a \vee b &= (a \wedge b) \vee b \\ &= b \text{ (by absorption).} \end{aligned}$$

This proves $i) \Leftrightarrow ii)$.

Now suppose $a \vee b = b$. Then

$$\begin{aligned} a \wedge \neg b &= (a \wedge \neg b) \vee 0 \\ &= (a \wedge \neg b) \vee (b \wedge \neg b) \\ &= (a \vee b) \wedge \neg b \\ &= b \wedge \neg b \\ &= 0. \end{aligned}$$

Finally, suppose $a \wedge \neg b = 0$. Then

$$\begin{aligned} a \vee b &= (a \vee b) \wedge 1 \\ &= (a \vee b) \wedge (\neg b \vee b) \\ &= (a \wedge \neg b) \vee b \\ &= 0 \vee b \\ &= b. \end{aligned}$$

This establishes $i) \Leftrightarrow iii)$.

□

From the equivalence of (i) and (ii) in the above lemma, we obtain the following corollary relating the notions of algebraic duality and order duality discussed in Section 3.2.

Corollary 4.14 *Let B be a Boolean algebra. The order induced by the dual Boolean algebra is the order dual to the order induced by B.*

4.3 Supremum and infimum

The posets arising from Boolean algebras in the way described above have some special properties, considering the existence of certain upper and lower bounds of subsets of B. First we need some more definitions.

Definition 4.15 Let $\langle P, \leq \rangle$ be a poset and $S \subseteq P$. An element $x \in P$ is an **upper bound** for S if $s \leq x$ for all $s \in S$. Dually $x \in P$ is a **lower bound** for S if $x \leq s$ for all $s \in S$.

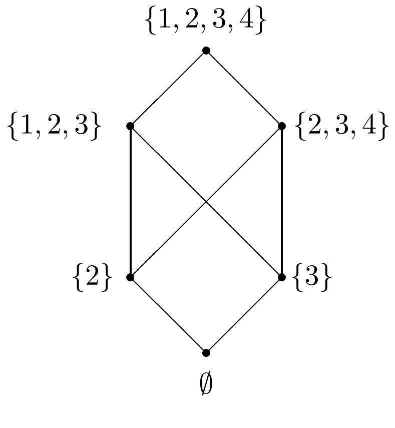
Definition 4.16 Let $\langle P, \leq \rangle$ be a poset and $S \subseteq P$. An element $s \in P$ is a **supremum** of S if s is the smallest among the upper bounds of S. That is, s is an upper bound of S and $s \leq x$ for all upper bounds x of S. Dually the **infimum** of S is defined to be the greatest lower bound of S.

Note that from this definition it follows that if the supremum of a set exists it is unique. If it exists we denote the supremum of a set S by $\sup S$ and the infimum by $\inf S$.

Now let S be a subset of an ordered set P. If the set S is empty, then every element of P is an upper bound of S and consequently the supremum of S is 0, which is the smallest of all upper bounds. Equivalently the infimum of the empty-set is 1. If S just contains one element p, it obviously has a supremum and an infimum, which are both equal to p.

The following example shows that the situation becomes more interesting if S has two elements.

Example 4.17 Consider the poset P:



and subsets $S := \{\{1,2,3\},\{2,3,4\}\}$ and $S' := \{\{2\},\{3\}\}$. The only upper bound of S is $\{1,2,3,4\}$ and hence this is the supremum of S. The upper bounds of S' are $\{1,2,3\},\{2,3,4\}$ and $\{1,2,3,4\}$. However, there is no smallest upper bound, so the supremum of S' does not exist in P.

This example shows that the supremum (or infimum) of a subset of an arbitrary poset may not exist. However, for the posets arising from Boolean algebras we have the following theorem stating that the supremum and infimum of all *finite* subsets exist.

Theorem 4.18 *Let $\langle B, \vee, \wedge, \neg, 0, 1\rangle$ be a Boolean algebra and $\leq$ the order on B as defined in Theorem 4.11. Then every finite subset A of B has both a supremum and an infimum. Moreover for all $a, b \in B$,*

$$\sup\{a, b\} = a \vee b \quad \textit{and} \quad \inf\{a, b\} = a \wedge b.$$

Proof. We first prove the second claim. We have

i) $a \vee (a \vee b) = (a \vee a) \vee b = a \vee b$ implies $a \leq a \vee b$

ii) $b \vee (a \vee b) = b \vee (b \vee a) = (b \vee b) \vee a = b \vee a = a \vee b$ implies $b \leq a \vee b$.

So $a \vee b$ is an upper bound of $\{a, b\}$. Suppose $x \in B$ is also an upper bound of $\{a, b\}$. In other words, $a \vee x = x$ and $b \vee x = x$. This implies $(a \vee b) \vee x = a \vee (b \vee x) = a \vee x = x$, hence $a \vee b \leq x$. So $a \vee b$ is the least upper bound of $\{x, y\}$.
For the first claim note, that $\sup\{a\} = a$ for all $a \in B$ and $\sup\ \emptyset = 0$ (since every element of B is an upper bound of the empty set). Hence by induction on the number of elements, we obtain that every finite subset of A has a supremum.
The result for the infimum is obtained by duality on the order and the algebra.

□

In the rest of this work we will simply use $x \vee y$ and $x \wedge y$ to denote the supremum, respectively infimum, of the set $\{x, y\}$. For an arbitrary subset A of a poset we use $\bigvee A$ to denote the supremum of A, if it exists. Similarly we use $\bigwedge A$ to denote the infimum of A, if it exists. When A is a subset of a Boolean algebra the supremum is often referred to as the **join** of A and the infimum of A is often called the **meet** of A. We use this terminology from now on.

Ordered sets that have the property that all finite subsets have a supremum and infimum are better known as **lattices** (or bounded lattices). Hence every Boolean algebra gives rise to a lattice. Moreover, the lattices arising from Boolean algebras have some additional structure. They are distributive and admit a complement. We will not give formal definitions of these properties, but just note the fact that any lattice having these properties gives rise to a Boolean algebra. That is, there is a one-to-one correspondence between Boolean algebras and distributive lattices admitting a complement.

In this chapter we have discovered that there are two different ways of thinking about Boolean algebras. That is, as an algebra satisfying certain identities or as an ordered set having some special properties. In the following chapters we will use both points of view, depending on which serves our purposes best and gives the most insight. When we say B is a Boolean algebra, we will use that it satisfies the identities we have given for the operations $\vee, \wedge$ and $\neg$ as well as the fact that it is an ordered set satisfying certain properties.

Chapter 5

Representation of Boolean algebras

An important result in the theory of Boolean algebras is the fact that every Boolean algebra can be represented as a field of sets. It is this result that lies at the heart of the topological duality theory developed in part III. In this chapter we will first consider the representation of finite Boolean algebras, which is somewhat simpler than the infinite case. In order to obtain a similar theorem for infinite Boolean algebras we introduce the notion of a prime filter. It is this concept that is the basis of the general representation theorem for Boolean algebras, which is the subject of the last section of this chapter.

5.1 Representations of Boolean algebras: the finite case

In this section we will prove that every finite Boolean algebra is isomorphic to the powerset algebra of some set X. Moreover, we will show that this set X can be obtained as a special subset of our original algebra, namely the set of *atoms* of B.

Definition 5.1 Let B be a Boolean algebra. A non-zero element $x \in B$ is called an **atom** if $0 \leq a \leq x$ implies $a = 0$ or $a = x$, for all $a \in B$. In other words, atoms are the elements that cover 0.

We denote the set of atoms of a Boolean algebra B by $\mathcal{A}(B)$.

Example 5.2 In a field of sets every singleton set is an atom. More precisely, the atoms of a powerset algebra are exactly the singletons.

The following lemma states that the set of atoms of a finite Boolean algebra B is a *join-dense* subset of B. That is, that every element of B can be obtained as the join of some subset of $\mathcal{A}(B)$. More explicitly, every element a of B is the join of the set of the atoms below a.

Lemma 5.3 *Let B be a finite Boolean algebra. Then for each $a \in B$,*

$$a = \bigvee \{x \in \mathcal{A}(B) \mid x \leq a\}.$$

Proof. Let $a \in B$ and $S = \{x \in \mathcal{A}(B) \mid x \leq a\}$. Obviously a is an upper bound for S, so $\bigvee S \leq a$. Let b be an upper bound for S. We have to show $a \leq b$. Suppose $a \not\leq b$. Then, by lemma 4.13 $a \wedge \neg b > 0$. Now since B is finite, we can show (by induction), that every element $c \in B$ with $c > 0$ has an atom below it (just go down until you reach an atom). In particular, we can find an $x \in \mathcal{A}(B)$ such that $x \leq a \wedge \neg b$. Then $x \leq a$, so $x \in S$ and hence $x \leq b$. But $x \leq \neg b$ as well, so we have $x \leq b \wedge \neg b = 0$, a contradiction. Hence $a \leq b$. That is, a is the smallest upper bound for S and thus the join of S.

□

Before moving on to the main theorem of this section we state two lemmas that will be used in the proof of this theorem.

Lemma 5.4 *Let B be a Boolean algebra, $x \in \mathcal{A}(B)$ and $a, b \in B$. Then $x \leq a \vee b$ implies $x \leq a$ or $x \leq b$.*

Proof. Suppose $a = 0$. Then $x \leq a \vee b = b$. Similarly $b = 0$ implies $x \leq a$. Now suppose $a, b > 0$ and $x \not\leq a$, $x \not\leq b$. Since $x \in \mathcal{A}(B)$ the only lower bounds of x are x itself and 0. Hence $x \wedge a = x \wedge b = 0$. So $x = x \wedge (a \vee b) = (x \wedge a) \vee (x \wedge b) = 0$, a contradiction. So $x \leq a$ or $x \leq b$.

□

An easy inductive argument gives rise to the following generalization of lemma 5.4.

Lemma 5.5 *Let B be a Boolean algebra, $x \in \mathcal{A}(B)$ and $S \subseteq B$ finite. Then $x \leq \bigvee S$ implies $x \leq s$ for some $s \in S$.*

We are now ready to prove the main theorem of this section known as the representation theorem for finite Boolean algebras.

Theorem 5.6 *Let B be a finite Boolean algebra. Then B is isomorphic to $\mathcal{P}(\mathcal{A}(B))$. Moreover, the map $\varphi : B \to \mathcal{P}(\mathcal{A}(B))$ defined by*

$$\varphi : a \mapsto \{x \in \mathcal{A}(B) \mid x \leq a\}$$

is a Boolean isomorphism.

Proof. We will first show that φ is a homomorphism. By lemma 3.10, this is established by the following observations:

$$\begin{aligned}
\varphi(a \wedge b) &= \{x \in \mathcal{A}(B) \mid x \leq a \wedge b\} \\
&= \{x \in \mathcal{A}(B) \mid x \leq a \text{ and } x \leq b\} \\
&= \{x \in \mathcal{A}(B) \mid x \leq a\} \cap \{x \in \mathcal{A}(B) \mid x \leq a\} \\
&= \varphi(a) \cap \varphi(b)
\end{aligned}$$

$$\begin{aligned}
\varphi(a \vee b) &= \{x \in \mathcal{A}(B) \mid x \leq a \vee b\} \\
&= \{x \in \mathcal{A}(B) \mid x \leq a \text{ or } x \leq b\} \text{ (by lemma 5.4)} \\
&= \{x \in \mathcal{A}(B) \mid x \leq a\} \cup \{x \in \mathcal{A}(B) \mid x \leq a\} \\
&= \varphi(a) \cup \varphi(b)
\end{aligned}$$

$$\begin{aligned}
\varphi(0) &= \emptyset \\
\varphi(1) &= \mathcal{A}(B).
\end{aligned}$$

Furthermore, by lemma 5.3, we have

$$a = \bigvee\{x \in \mathcal{A}(B) \mid x \leq a\} = \bigvee \varphi(a).$$

Hence $\varphi(a) = \varphi(b)$ implies $a = \bigvee \varphi(a) = \bigvee \varphi(b) = b$. That is, φ is one-to-one.
It remains to show that φ is onto. Note that $\varphi(0) = \emptyset$. Now let $S \subseteq \mathcal{A}(B)$ non-empty. We will prove $\varphi(\bigvee S) = S$. Obviously $S \subseteq \varphi(\bigvee S)$. Now let $x \in \varphi(\bigvee S)$, that is $x \in \mathcal{A}(B)$ and $x \leq \bigvee S$. By lemma 5.5 we obtain $x \leq s$ for some $s \in S$. As s and x are both atoms we have $x = s$. Hence $x \in S$, that is $\varphi(\bigvee S) \subseteq S$.

□

Before moving on to the infinite case we draw some attention to a special property of powerset algebras.

Definition 5.7 Let B be a Boolean algebra. If every subset of B has both a join and a meet, then B is called a **complete** Boolean algebra.

Example 5.8 Every powerset algebra is complete, since $\bigvee S = \bigcup S$ for all subsets S.

Now consider the finite-cofinite algebra of the natural numbers defined by

$$B := \{A \subseteq \mathbb{N} \mid A \text{ is finite or } \mathbb{N} \setminus A \text{ is finite }\}.$$

It is easily checked that this is a field of sets and hence a Boolean algebra. Since the subset of B consisting of the singletons over the even numbers does not have a join, B is not a complete Boolean algebra. Note that completeness is preserved by isomorphisms, and therefore B can not be isomorphic to a powerset algebra.

5.2 Prime filters

The above observation shows that in general a Boolean algebra does not have to be isomorphic to a powerset algebra. So we can not extend the result from the finite case to the infinite case unchanged. However, in the next section we will prove that every Boolean algebra can be *embedded* in a powerset algebra.

Let B be an infinite Boolean algebra. One of the problems one encounters when extending the argument from the finite case, is that B might not have any atoms. We therefore have to come up with an alternative set X underlying the powerset algebra in which we are going to embed B. First we need some more definitions.

Definition 5.9 Let B be a Boolean algebra. A non-empty subset F of B is called a **filter** if

i) F is an up-set. That is, $a \in F$ and $a \leq b$ implies $b \in F$ for all $b \in B$,

ii) F is closed under $\wedge$. That is, $a, b \in F$ implies $a \wedge b \in F$.

The set of filters of a Boolean algebra B is denoted by $\mathcal{F}(B)$.

Example 5.10 For each $a \in B$, the set of elements greater or equal than a is denoted by $\uparrow a$. You can easily check that $\uparrow a$ is a filter of B. The set $\uparrow a$ is called the **principal filter** generated by a.

Definition 5.11 Let B be a Boolean algebra. A filter F of B is called **proper** if $F \neq B$. This is equivalent to the condition that $0 \notin F$.

Definition 5.12 Let B be a Boolean algebra. A proper filter F of B is called **prime** if $a \vee b \in F$ implies $a \in F$ or $b \in F$ for all $a, b \in B$.

Example 5.13 Let $\mathcal{P}(X)$ be a powerset algebra and $x \in X$. Then the subset $\uparrow\{x\} = \{A \in \mathcal{P}(X) \mid x \in A\}$ is a prime filter in $\mathcal{P}(X)$.

The following lemma gives a different characterization of prime filters.

Lemma 5.14 *Let B be a Boolean algebra and F a proper filter in B. Then the following are equivalent:*

i) F is prime,

ii) for all $a \in B$: $\quad a \in F$ if and only if $\neg a \notin F$,

iii) F is maximal with respect to the inclusion order. That is, the only filter properly containing F is B.

Proof. i) $\Rightarrow$ ii). Suppose F is prime. Since F is non-empty we have $a \vee \neg a = 1 \in F$. Hence $a \in F$ or $\neg a \in F$. If both a and $\neg a$ belong to F then $a \wedge \neg a = 0 \in F$, a contradiction.
ii) $\Rightarrow$ iii). Suppose F satisfies condition ii) and F' is a filter properly containing F. That is, there is an $a \in F'$ such that $a \notin F$. But then $\neg a \in F \subset F'$. Hence $a \wedge \neg a = 0 \in F'$. So $F' = B$.
iii) $\Rightarrow$ i) Suppose F is maximal, $a \vee b \in F$ and $a \notin F$. We want to show $b \in F$. Define $F_a = \uparrow\{a \wedge c \mid c \in F\}$. Then F_a is a filter containing F and a. Hence, since F is maximal $F_a = B$. In particular $0 \in F_a$, so $0 = a \wedge c$ for some $c \in F$. Then

$$b \wedge c = (a \wedge c) \vee (b \wedge c) = (a \vee b) \wedge c \in F.$$

Since $b \wedge c \leq b$, we have $b \in F$.

□

A remark about terminology: filters that contain either a or $\neg a$ for all $a \in B$ are called **ultrafilters**. Hence, lemma 5.14 shows that the set of ultrafilters of a Boolean algebra coincides with the set of prime filters.

The set of all prime filters of a Boolean algebra B is denoted by $\mathcal{F}_p(B)$. The concepts of an **ideal** and **prime ideal** are defined dually.

Definition 5.15 Let B be a Boolean algebra. A non-empty subset I of B is called an **ideal** if
i) I is an down-set. That is, $a \in I$ and $b \leq a$ implies $b \in I$ for all $b \in B$,
ii) I is closed under $\vee$. That is, $a, b \in I$ implies $a \vee b \in I$.

Definition 5.16 Let B be a Boolean algebra. A proper ideal I of B is called **prime** if $a \wedge b \in I$ implies $a \in I$ or $b \in I$ for all $a, b \in B$.

The set of all prime ideals is denoted by $\mathcal{I}_p(B)$. It is easily proven that a subset F of B is a prime filter if and only if $B \setminus F$ is a prime ideal. So the prime filters of a Boolean algebra are in one-to-one correspondence to its prime ideals.

5.3 Representation of Boolean algebras: the infinite case

As mentioned before, the set of atoms no longer suffices as the underlying set when representing an infinite Boolean algebra B as a field of sets. In this section we will prove that in the infinite case the role of the atoms can be replaced by the prime filters of B. That is, we show that B can be embedded in $\mathcal{P}(\mathcal{F}_p(B))$, the powerset of the set of all prime filters of B. This implies that B is isomorphic to a field of subsets over $\mathcal{F}_p(B)$.

Before we can give the proof of this result, we have to draw some attention to the existence of prime filters. Theorem 5.14 shows that a prime filter of B is just a maximal element of the set of all proper filters of B. The existence of maximal elements is closely related to set theory and its axioms. In our case, to prove that there are enough prime filters to 'distinguish' between the elements of B we have to use a form of the Axiom of Choice, known as

Zorn's lemma. The form of Zorn's lemma we will use in our proof is the following:

Zorn's Lemma
Let P be a non-empty ordered set in which every non-empty chain has an upper bound. Then P has a maximal element.

The following theorem, which is known as the Prime Filter Theorem for Boolean algebras, shows the existence of prime filters, using this lemma.

Theorem 5.17 *Let B be a Boolean algebra. Given a proper filter F of B, then there exist a prime filter F' such that $F \subseteq F'$.*

Proof. Let F be a proper filter of B. We apply Zorn's lemma to the set

$$P := \{G \in \mathcal{F}(B) \mid F \subseteq G \neq B\}$$

ordered by inclusion. Since P contains F, it is non-empty. Let $\mathcal{C}$ be a non-empty chain in P. We show that $\bigcup \mathcal{C}$, which is an upper bound for $\mathcal{C}$, is an element of P. Clearly $\bigcup \mathcal{C}$ is an up-set containing F. Furthermore, $\bigcup \mathcal{C} \neq B$, because $0 \notin \bigcup \mathcal{C}$. It remains to show that $a, b \in \bigcup \mathcal{C}$ implies $a \wedge b \in \bigcup \mathcal{C}$. If $a, b \in \bigcup \mathcal{C}$, then we can find filters $F_1, F_2 \in \mathcal{C}$ such that $a \in F_1$ and $b \in F_2$. Since $\mathcal{C}$ is a chain we may assume without loss of generality that $F_1 \subseteq F_2$. Then $a, b \in F_2$, which implies $a \wedge b \in F_2 \subseteq \bigcup \mathcal{C}$. Hence Zorn's lemma can be applied and yields the existence of a maximal element of P. It is easy to see that this element is a maximal filter and hence, by lemma 5.14, is a prime filter. Furthermore, it obviously contains F and therefore is the prime filter we require.

□

We can also formulate this theorem slightly differently, using both prime filters and prime ideals. This formulation of the Prime Filter Theorem will turn out to be useful in some proofs later on.

Theorem 5.18 *Let B be a Boolean algebra, F a filter of B and I and ideal of B such that $F \cap I = \emptyset$. Then there exist a prime filter F' and a prime ideal I' such that $F \subseteq F'$, $I \subseteq I'$ and $F' \cap I' = \emptyset$.*

Proof. Define

$$G := \{c \in B \mid \exists a \in F, b \in I : a \wedge \neg b \leq c\}.$$

We prove that G is a proper filter of B. Clearly G is an up-set. Given c_1, c_2 determine $a_1, a_2 \in F$ and $b_1, b_2 \in I$ such that $a_1 \wedge \neg b_1 \leq c_1$ and $a_2 \wedge \neg b_2 \leq c_2$. Then $(a_1 \wedge \neg b_1) \wedge (a_2 \wedge \neg b_2) \leq c_1 \wedge c_2$. Furthermore

$$\begin{aligned} (a_1 \wedge \neg b_1) \wedge (a_2 \wedge \neg b_2) &= (a_1 \wedge a_2) \wedge (\neg b_1 \wedge \neg b_2) \\ &= (a_1 \wedge a_2) \wedge \neg (b_1 \vee b_2). \end{aligned}$$

Since $a_1 \wedge a_2 \in F$ and $b_1 \vee b_2 \in I$ this implies $c_1 \wedge c_2 \in G$. Hence G is a filter of B. Now suppose $0 \in G$. That is, there exist $a \in F$ and $b \in I$ such that $a \wedge \neg b = 0$. By Lemma 4.13 this implies $a \leq b$. Hence $a \in I$ and $b \in F$, but this is in contradiction with $F \cap I = \emptyset$. Hence G is a proper filter of B. By the previous theorem there exists a prime filter F' such that $F \subseteq F'$. Furthermore, observe that $B \setminus F'$ is a prime ideal and, since $\neg b \in F'$ for all $b \in I$, we have $I \subseteq B \setminus F'$.

□

Now we are ready to prove the main theorem of this section. This theorem can be seen as a generalization of Theorem 5.6, which said that every finite Boolean algebra is isomorphic to a powerset algebra. We have observed that, in general, this no longer holds. We can however show that an arbitrary Boolean algebra can be *embedded* in a powerset algebra.

Theorem 5.19 *Let B be a Boolean algebra. Then B is isomorphic to a subalgebra of $\mathcal{P}(\mathcal{F}_p(B))$. Moreover, the map $\varphi : B \to \mathcal{P}(\mathcal{F}_p(B))$ defined by*

$$\varphi : a \mapsto \{F \in \mathcal{F}_p(B) \mid a \in F\}$$

is a Boolean algebra embedding.

Proof. We first show that φ is a homomorphism. Clearly $\varphi(0) = \emptyset$, since no prime filter contains 0 and $\varphi(1) = \mathcal{F}_p(B)$, because each prime filter contains 1. We have to show $\varphi(a \wedge b) = \varphi(a) \cap \varphi(b)$ and $\varphi(a \vee b) = \varphi(a) \cup \varphi(b)$, for all $a, b \in B$. Take $F \in \mathcal{F}_p(B)$. Since F is a filter,

$$a \wedge b \in F \text{ if and only if } a \in F \text{ and } b \in F$$

and since F is prime,

$$a \vee b \in F \text{ if and only if } a \in F \text{ or } b \in F.$$

Thus we have

$$\begin{aligned}\varphi(a \wedge b) &= \{F \in \mathcal{F}_p(B) \mid a \wedge b \in F\} \\ &= \{F \in \mathcal{F}_p(B) \mid a \in F \text{ and } b \in F\} \\ &= \{F \in \mathcal{F}_b(B) \mid a \in F\} \cap \{F \in \mathcal{F}_b(B) \mid b \in F\} \\ &= \varphi(a) \cap \varphi(b).\end{aligned}$$

and

$$\begin{aligned}\varphi(a \vee b) &= \{F \in \mathcal{F}_p(B) \mid a \vee b \in F\} \\ &= \{F \in \mathcal{F}_p(B) \mid a \in F \text{ or } b \in F\} \\ &= \{F \in \mathcal{F}_b(B) \mid a \in F\} \cup \{F \in \mathcal{F}_b(B) \mid b \in F\} \\ &= \varphi(a) \cup \varphi(b).\end{aligned}$$

So φ is a homomorphism.
It remains to be shown that φ is one-to-one. Let a, b be distinct elements of B. Without loss of generality we may assume $a \not\leq b$. By Lemma 4.13, this implies $a \wedge \neg b \neq 0$. Hence $F = \uparrow(a \wedge \neg b)$ is a proper filter that contains a and $\neg b$. By Theorem 5.17, there is a prime filter F' containing F. Obviously $a, \neg b \in F'$, but $b \notin F$, because F' is prime. So $F \in \varphi(a)$ and $F \notin \varphi(b)$, that is $\varphi(a) \neq \varphi(b)$.

□

Corollary 5.20 *Every Boolean algebra is isomorphic to a field of sets.*

It turns out that the embedding φ as defined in Theorem 5.19 has some interesting properties. First of all we show that φ is dense.

Definition 5.21 Let A and B be Boolean algebras and $h : A \to B$ a Boolean embedding. If every element of B can be expressed both as a join of meets and as a meet of joins of elements in $h(A)$, then the embedding h is called **dense**.

Theorem 5.22 *Let B be a Boolean algebra. Then the embedding $\varphi : B \hookrightarrow \mathcal{P}(\mathcal{F}_p(B))$, as defined in Theorem 5.19 is dense.*

Proof. Let $V \in \mathcal{P}(\mathcal{F}_p(B))$. Obviously V is the join of singletons in $\mathcal{P}(\mathcal{F}_p(B))$. So to prove the first claim it suffices to show that every singleton of $\mathcal{P}(\mathcal{F}_p(B))$ can be expressed as a meet of elements in $\varphi(B)$. This is obtained by the claim that

$$\{F\} = \bigwedge\{\varphi(a) \mid a \in F\} \text{ for all } F \in \mathcal{F}_p(B).$$

Obviously $F \in \bigwedge\{\varphi(a) \mid a \in F\}$. Furthermore, suppose $F' \in \bigwedge\{\varphi(a) \mid a \in F\}$. Then $F \subseteq F'$, which implies $F = F'$ (since F and F' are both prime and hence, by lemma 5.14, maximal).
The second claim, that V can be expressed as the meet of joins, is obtained by order duality.

□

A second observation about the embedding φ is that it is a compact embedding.

Definition 5.23 Let A and B be Boolean algebras and $h : A \to B$ a Boolean embedding. Assume also that B is complete. If for all $S, T \subseteq A$ with $\bigwedge h(S) \leq \bigvee h(T)$, there exist finite $S' \subseteq S$ and $T' \subseteq T$ such that $\bigwedge S' \leq \bigvee T'$, then the embedding h is called **compact**.

Observe that $\mathcal{P}(\mathcal{F}_p(B))$ is a complete Boolean algebra as it is a powerset.

Theorem 5.24 *Let B be a Boolean algebra. The embedding $\varphi : B \hookrightarrow \mathcal{P}(\mathcal{F}_p(B))$, as defined in Theorem 5.19, is* ***compact****.*

Proof. Let $S, T \subseteq B$ such that $\bigwedge \varphi(S) \leq \bigvee \varphi(T)$. Define $\neg T := \{\neg t \mid t \in T\}$ and define $A \subseteq B$ as follows

$$A := S \cup (\neg T).$$

Let F be the smallest filter containing A. That is,

$$F = \{b \in B \mid a_1 \wedge \ldots \wedge a_n \leq b \text{ for some } a_1, \ldots, a_n \in A\}.$$

If F is not proper, then $0 \in F$ and $a_1 \wedge \ldots \wedge a_n = 0$ for some finite subset $\{a_1, \ldots a_n\}$ of A. That is, there exist $s_1, \ldots s_k \in S$ and $t_1, \ldots, t_l \in T$ such that

$$s_1 \wedge \ldots \wedge s_k \wedge (\neg t_1) \wedge \ldots \wedge (\neg t_l) = 0.$$

Hence, by de Morgan's law, we obtain

$$s_1 \wedge \ldots \wedge s_k \wedge \neg(t_1 \vee \ldots \vee t_l) = 0.$$

By Lemma 4.13, this implies

$$s_1 \wedge \ldots \wedge s_k \leq t_1 \vee \ldots \vee t_l$$

which proves our assertion.
Now suppose F is proper. Then, by Lemma 5.17, there is a prime filter F' containing F. Since $S \subseteq F'$ we have,

$$F' \in \{F \in \mathcal{F}_p(B) \mid S \subseteq F\} = \bigcap \varphi(S) = \bigwedge \varphi(S).$$

On the other hand $\neg T \subseteq F'$, implies $t \notin F'$ for all $t \in T$ (since F' is proper). Hence

$$F' \notin \{F \in \mathcal{F}_p(B) \mid t \in F \text{ for some } t \in T\} = \bigcup \varphi(S) = \bigvee \varphi(T).$$

This is a contradiction of $\bigwedge \varphi(S) \leq \bigvee \varphi(T)$.

□

In the next chapter we introduce a topology on the $\mathcal{F}_p(B)$, which captures exactly the subsets of $\mathcal{F}_p(B)$ that are in the image of φ. We will see that the compactness of φ corresponds to the compactness of the topology.

Part III

Duality

Chapter 6

A topological representation for Boolean algebras

In this chapter we use the representation results from the previous chapter to get a correspondence between Boolean algebras and certain topological spaces. We define a topology on the set of prime filters of a Boolean algebra and we show that the topological spaces arising in this way are of a special kind, called Stone spaces. These are the first steps into the theory of categorical duality which is pursued in the next chapter.

6.1 The dual space of a Boolean algebra

In the previous chapter we have shown that the map $\varphi : B \to \mathcal{P}(\mathcal{F}_p(B))$ defined by

$$\varphi : a \mapsto \{F \in \mathcal{F}_p(B) \mid a \in F\}$$

is a Boolean algebra embedding of B into the powerset of its prime filters, $\mathcal{P}(\mathcal{F}_p(B))$. The image $\varphi(B)$ is a representation of B as a field of sets and is a subalgebra of the powerset algebra $\mathcal{P}(\mathcal{F}_p(B))$. In general, the powerset algebra $\mathcal{P}(\mathcal{F}_p(B))$ alone does not give enough information to recover the structure of the original algebra B. One can find non-isomorphic Boolean algebras A and B such that $\mathcal{P}(\mathcal{F}_p(A)) \cong \mathcal{P}(\mathcal{F}_p(B))$. We will therefore put additional structure on the set $\mathcal{F}_p(B)$ that enables us to determine exactly which elements of $\mathcal{P}(\mathcal{F}_p(B))$ belong to the image of φ. More specifically, we will equip $\mathcal{F}_p(B)$ with a topology $\mathcal{T}_B$ and we will prove that the clopen sets of this topology are exactly the elements in the image of φ, and hence form a Boolean algebra isomorphic to B.

Let B be a Boolean algebra. The family of subsets of $\mathcal{F}_p(B)$ in the image of φ does in general not form a topology on $\mathcal{F}_p(B)$ since it is not closed under arbitrary union. Note that the image of φ is closed under finite intersection since $\varphi(a) \cap \varphi(b) = \varphi(a \wedge b)$ and furthermore contains $\varphi(0) = \emptyset$ and $\varphi(1) = \mathcal{F}_p(B)$. Hence the set $\varphi(B)$ is the basis for some topology. We will define $\mathcal{T}_B$ to be the topology generated by this basis.

Definition 6.1 Let B be a Boolean algebra and $\mathcal{F}_p(B)$ the set of all prime filters in B. We define the set of open subset of $\mathcal{F}_p(B)$ by

$$\mathcal{T}_B := \{U \subseteq \mathcal{F}_p(B) \mid U \text{ is a union of members of } \varphi(B)\}$$

The topological space $\langle \mathcal{F}_p(B), \mathcal{T}_B \rangle$ is called the **dual space** of B.

Earlier we made the comment that words like 'dual' and 'duality' are often used in mathematics to denote that things are turned upside down. Although it might not be clear from this definition that this is the case here, we will see in the next chapter (when introducing categories and dual maps) that there is actually a notion of 'turning things around' involved at the categorical level.

Usually write $X_B := \mathcal{F}_p(B)$. This makes it easier to think of elements in X_B as points in a space rather than subsets of B. The dual space of a Boolean algebra B is denoted by $\langle X_B, \mathcal{T}_B \rangle$. As long as $\mathcal{T}_B$ is the only topology under consideration we will often write 'X_B is the dual space of B' without specifying the topology.

The next proposition shows the relationship between the topology and the image of φ.

Proposition 6.2 *Let B be a Boolean, X_B its dual space, and $\varphi : B \hookrightarrow \mathcal{P}(X_B)$ defined by*

$$\varphi : a \mapsto \{F \in \mathcal{F}_p(B) \mid a \in F\}.$$

Then the sets in the image of φ are exactly the clopen sets of X_B.

Proof. Let $U \in \varphi(B)$. That is, there is an $a \in B$ such that $U = \varphi(a)$. By definition, $\varphi(a)$ is open. Furthermore $X_B \setminus \varphi(a) = \varphi(\neg a)$ is open. So $\varphi(a)$ is a clopen subset of X_B.

Now suppose U is a clopen subset of X_B. Because U is open, $U = \bigcup \varphi(T)$ for some $T \subseteq B$. But U is also closed and so $X_B \setminus U = \bigcup \varphi(S)$ for some $S \subseteq B$. Define $\neg S := \{\neg s \mid s \in S\}$ and observe that

$$\begin{array}{rrcl} & X_B \setminus U & = & \bigcup \varphi(S) \\ \Leftrightarrow & U & = & X_B \setminus \bigcup \varphi(S) \\ \Leftrightarrow & U & = & \bigcap \varphi(\neg S) \end{array}$$

So we have

$$\bigcap \varphi(\neg S) = \bigcup \varphi(T).$$

Hence, by Theorem 5.24, which shows that φ is a compact embedding, there are finite $S' \subseteq \neg S$ and $T' \subseteq T$ such that

$$\bigwedge S' \leq \bigvee T'.$$

As φ is a homomorphism it preserves order and finite joins. This implies

$$U = \bigcap \varphi(\neg S) \subseteq \bigcap \varphi(S') = \varphi(\bigwedge S') \subseteq \varphi(\bigvee T') = \bigcup \varphi(T') \subseteq \bigcup \varphi(T) = U.$$

Hence $\bigvee \varphi(T') = U$. Because T' is finite this implies $U = \varphi(a)$ with $a = \bigvee T' \in B$. That is, $U \in \varphi(B)$.

□

Clearly, the clopen subsets of any topological space form a Boolean algebra. In particular, the clopen subsets of the dual space of some Boolean algebra B form again a Boolean algebra B'. The following theorem, which is a direct consequence of the previous proposition and theorem 5.19, proves that B and B' are isomorphic.

Theorem 6.3 *Let B be a Boolean algebra and X_B its dual space. Then*

$$B \cong Cl(X_B),$$

where $Cl(X_B)$ denotes the set of clopen subsets of X_B.

Proof. Let $\varphi : B \hookrightarrow \mathcal{P}(X_B)$ be the embedding as defined in Proposition 6.2. This proposition shows that the image of φ are exactly the clopen sets of $\langle X_B, \mathcal{T}_B \rangle$. Since φ is a Boolean algebra embedding we have $B \cong \varphi(B) = Cl(X_B)$.

□

6.2 Stone spaces

We will now take a closer look at the topological spaces arising as the dual space of a Boolean algebra. First of all, the compactness of the embedding φ established in theorem 5.24 in the previous chapter, can be translated into topological compactness of the dual space.

Proposition 6.4 *Let B be a Boolean algebra and X_B its dual space. Then X_B is compact.*

Proof. Let $\mathcal{U}$ be an open cover of X_B. We have to show that $\mathcal{U}$ contains a finite subcover of X_B. Since every open set is a union of elements of $\varphi(B)$ we may assume without loss of generality that $\mathcal{U} \subseteq \varphi(B)$. That is, there is a subset $A \subseteq B$ such that $\mathcal{U} = \{\varphi(a) \mid a \in A\}$. In other words

$$X_B = \bigcup \varphi(A).$$

Observe that $X_B = \varphi(1) = \bigcap \varphi(1)$. Hence

$$\bigcap \varphi(1) = \bigcup \varphi(A).$$

Now we can use the fact that φ is a compact embedding (Theorem 5.24) to obtain a finite subset $A' \subseteq A$ such that

$$X_B = \bigcap \varphi(1) = \bigcup \varphi(A').$$

Hence $\mathcal{U}' = \{\varphi(a) \mid a \in A'\}$ is a finite subcover of X_B.

□

Another common property of the topological spaces arising as dual spaces of Boolean algebras, is that they satisfy some separation property.

Definition 6.5 A topological space $\langle X; \mathcal{T} \rangle$ is said to be **totally disconnected** if, given distinct points $x, y \in X$, there exists a clopen subset V of X such that $x \in V$ and $y \notin V$.

Note that this definition implies that a totally disconnected space is Hausdorff: if $x, y \in X$ and V clopen such that $x \in V$ and $y \notin V$ then $y \in X \setminus V$, which is open, while V and $X \setminus V$ are disjoint.

Proposition 6.6 *Let $\langle X_B, \mathcal{T}_B \rangle$ be the dual space of a Boolean algebra B. Then $\langle X_B, \mathcal{T}_B \rangle$ is totally disconnected.*

Proof. Let F_1 and F_2 be distinct elements of X_B. We have to find a clopen subset of X_B that contains F_1 but not F_2. Since F_1 and F_2 are distinct, there is an $a \in B$ with $a \in F_1$ and $a \notin F_2$. Hence $F_1 \in \varphi(a)$ and $F_2 \notin \varphi(a)$ and, by theorem 6.2, $\varphi(a)$ is clopen.

□

Definition 6.7 A topological space that is compact and totally disconnected is called a **Stone space** (or **Boolean space**).

Corollary 6.8 *The dual space of a Boolean algebra is a Stone space.*

Let us summarize what we have established so far. Starting from a Boolean algebra B we have constructed its dual space $\langle X_B, \mathcal{T}_B \rangle$. Then we made the following two observations:

1. $\langle X_B, \mathcal{T}_B \rangle$ is a Stone space.
2. The clopen subsets of $\langle X_B, \mathcal{T}_B \rangle$ give rise to a Boolean algebra isomorphic to B.

Now, if we start from a Stone space, obviously the clopen subsets give rise to a Boolean algebra. The next theorem shows that the dual space of this Boolean algebra is homeomorphic to the original space.

Theorem 6.9 *Let Y be a Stone space. Define B to be the Boolean algebra of clopen subsets of Y and let X_B be the dual of B. Then Y and X_B are homeomorphic.*

Proof. We show that

$$\psi : x \mapsto \{a \in B \mid x \in a\}$$

is a homeomorphism between Y and X_B.
It is easy to check that $\psi(x)$ is a prime filter in B. So ψ is well-defined. Since Y is a compact and Hausdorff it is sufficient to prove that ψ is continuous and bijective.
Let $x, y \in Y$ be distinct. Since Y is totally disconnected there is a clopen set $a \in B$ containing x but not y. Hence $a \in \psi(x)$ and $a \notin \psi(y)$. That is, $\psi(x) \neq \psi(y)$. Hence ψ is injective.
To establish continuity is suffices to show that $\psi^{-1}(U)$ is open for every U in the basis of X_B. That is, we have to show that $\psi^{-1}(\varphi(a))$ is open for

all $a \in B$, with φ as defined in Proposition 6.2. This is established by the observation that

$$\psi^{-1}(\varphi(a)) = \{x \in Y \mid \psi(x) \in \varphi(a)\} = \{x \in Y \mid a \in \psi(x)\} = a$$

for every $a \in B$ and that, by definition, B is the set of clopen subsets of Y. Finally we have to prove that ψ is surjective. Suppose that there is an $x \in X_B$ such that $x \notin \psi(Y)$. As X_B is totally disconnected, there exists for every $y \in \psi(Y)$ a clopen subset V_y of X_B such that $y \in V_y$ and $x \notin V_y$. As $\psi(Y)$ is closed and hence compact there is an open subcover of $\psi(Y)$. That is, there exist $y_1, y_2, \ldots, y_n$ such that $\psi(Y) \subseteq V = V_{y_1} \cup \ldots V_{y_n}$. As V is a finite union of clopen sets it is again clopen. So there exists an $a \in B$ such that $V = \varphi(a)$. By definition of V this implies $\psi(Y) \subseteq \varphi(a)$ and $x \notin \varphi(a)$. Hence $Y = \psi^{-1}(\varphi(a)) = a$. But this is in contradiction with the fact that $x \notin \varphi(a)$.

□

Theorems 6.3 and 6.9 establish a tight connection between Boolean algebras and Stone spaces. Essentially they argue that, up to isomorphisms and homeomorphisms, there is a one-to-one correspondence between Boolean algebras and Stone spaces. This correspondence is part of what is called a duality between the category of Boolean algebras and the category of Stone spaces. In the next chapter we will introduce the notion of a category and see that the relationship between Boolean algebras and Stone spaces can be extended to a much broader (categorical) setting that captures maps and substructures as well.

Chapter 7

Stone duality

In the previous chapter we have seen that, up to isomorphism, there is a one-to-one correspondence between Boolean algebras and Stone spaces. In this chapter we will reformulate these results in a categorical setting. This enables us to extend the correspondence between the structures to a correspondence between their structure preserving maps. That is, we translate Boolean homomorphisms into continuous maps and vice versa.

7.1 Category theory: an introduction

The mathematical discipline that studies *classes* of structures rather than structures on their own, is known as category theory. In addition to classes of structures, category theorists are also interested in the relationship between the structures in a particular class, that is in *structure preserving maps* or *morphisms.* The central object of study in this discipline is a category.

Definition 7.1 A **category** $\mathcal{C}$ consists of:

1. a collection of **objects**, denoted by $\mathrm{Obj}(\mathcal{C})$
2. a collection of **morphisms**, denoted by $\mathrm{Mor}(\mathcal{C})$.

Each morphism f has two objects A and B, are associated with it: the **domain** and **codomain**. This is denoted by $f : A \to B$. Furthermore, a notion of **composition** of morphisms is defined that is associative. The composition of two morphisms f and g is denoted by $g \circ f$ and is defined if and only if the codomain of f equals the domain of g. Finally, there is an **identity morphism** $\mathrm{id}_A : A \to A$ associated with each object A, such that

$f \circ \mathrm{id}_A = f$ for every $f : A \to B$ and $\mathrm{id}_A \circ g = g$ for every $g : B \to A$.

This somewhat technical definition is best illustrated with some familiar examples.

Example 7.2 The class of ordered sets together with order preserving maps forms a category. The class of Boolean algebras with Boolean homomorphisms is a subcategory of this category. Also the class of topologies with continuous maps is an example of a category and the class of Stone spaces with continuous maps forms a subcategory of this category.

Like other mathematical structures, also for categories there exists a notion of structure preserving map. These are called functors. Functors map objects to objects and morphisms to morphisms. Furthermore they preserve the identity morphisms and respect the composition operation. This is expressed in the following definition.

Definition 7.3 Let $\mathcal{C}_1$ and $\mathcal{C}_2$ be categories. A **(covariant) functor** $F : \mathcal{C}_1 \to \mathcal{C}_2$ is a map which assigns to each $A \in \mathrm{Obj}(\mathcal{C}_1)$ an element $F(A) \in \mathrm{Obj}(\mathcal{C}_2)$ and to each morphism $f : A \to B \in \mathrm{Mor}(\mathcal{C}_1)$ a morphism $F(f) : F(A) \to F(B) \in \mathrm{Mor}(\mathcal{C}_2)$ such that

i) for every object $A \in \mathcal{C}_1, F(\mathrm{id}_A) = \mathrm{id}_{F(A)}$;

ii) if $f \circ g$ is defined in $\mathcal{C}_1$ then $F(f \circ g) = F(f) \circ F(g)$.

In addition to the notion of a covariant functor there is the notion of *contravariant* functor, which reverses the direction of the morphisms.

Definition 7.4 Let $\mathcal{C}_1$ and $\mathcal{C}_2$ be categories. A **contravariant functor** $F : \mathcal{C}_1 \to \mathcal{C}_2$ is a map which assigns to each $A \in \mathrm{Obj}(\mathcal{C}_1)$ an object $F(A) \in \mathrm{Obj}(\mathcal{C}_2)$ and to each morphism $f : A \to B \in \mathrm{Mor}(\mathcal{C}_1)$ a morphism $F(f) : F(B) \to F(A) \in \mathrm{Mor}(\mathcal{C}_2)$ such that

i) for every object $A \in \mathcal{C}_1, F(\mathrm{id}_A) = \mathrm{id}_{F(A)}$;

ii) if $f \circ g$ is defined in $\mathcal{C}_1$ then $F(f \circ g) = F(g) \circ F(f)$.

7.2 Duality between Boolean algebras and Stone spaces

We will now place the results of the previous chapter in a categorical setting. Let $\mathcal{B}$ be the category of Boolean algebras with Boolean homomorphisms and $\mathcal{S}$ the category of Stone spaces with continuous maps. In the previous chapter we have established a correspondence between the objects of $\mathcal{B}$ and the objects of $\mathcal{S}$. In this section we will extend this correspondence to the morphisms of the categories $\mathcal{B}$ and $\mathcal{S}$. More specifically, we define two contravariant functors $F : \mathcal{B} \to \mathcal{S}$ and $G : \mathcal{S} \to \mathcal{B}$ and prove that they establish a close relationship between the categories $\mathcal{B}$ and $\mathcal{S}$. That is, that they give rise to a categorical **duality** between the category of Boolean algebras and the category of Stone spaces. The term 'duality' refers here to the direction of the maps that is 'turned upside down' in the dual category. We will give an overview of the theory here. A complete treatment of this material is available in [3], [6] and [7].

We define $F : \mathrm{Obj}(\mathcal{B}) \to \mathrm{Obj}(\mathcal{S})$ and $G : \mathrm{Obj}(\mathcal{S}) \to \mathrm{Obj}(\mathcal{B})$ as follows:

$$\begin{array}{rcll} F(B) & := & X_B & \text{for all } B \in \mathrm{Obj}(\mathcal{B}) \\ G(X) & := & Cl(X) & \text{for all } X \in \mathrm{Obj}(\mathcal{S}). \end{array}$$

The results from the previous chapter are formulated in the following theorem.

Theorem 7.5 *Let $\mathcal{B}$ be the category of Boolean algebras and Boolean homomorphisms and $\mathcal{S}$ be the category of Stone spaces and continuous maps.*

- *i) For every $B \in Obj(\mathcal{B})$ there is an isomorphism $\varphi_B : B \to GF(B)$.*
- *ii) For every $X \in Obj(\mathcal{S})$ there is a homeomorphism $\psi_X : X \to FG(X)$.*

Proof. The isomorphisms φ_B and ψ_X are the maps defined in the proofs of theorem 6.3 and 6.9 respectively.

□

Note that the isomorphisms φ_B and the homeomorphisms ψ_X are defined in a uniform way for all the objects in the categories involved. We will not make this precise here but just remark that this is a necessary condition for a duality between categories. The technical notation needed is that of a natural transformation between functors, see for example [6, page 4]

The following two propositions show how we can translate Boolean homomorphisms to continuous maps and vice versa.

Proposition 7.6 *Let $h : A \to B$ be a Boolean homomorphism. For each $y \in X_B$, let*

$$(F(h))(y) := h^{-1}(y).$$

Then $F(h) : X_B \to X_A$ is continuous.

Proof. It is easy to check that the set $h^{-1}(y)$ is a prime filter in A whenever y is a prime filter in B. Hence, $F(h) : X_B \to X_A$ is well-defined. To prove that $F(h)$ is continuous it is enough to prove that $F(h)^{-1}(\varphi_A(a))$ is open in X_B for every $a \in A$, since the set $\{\varphi_A(a) \mid a \in A\}$ is a basis for the topology X_A. We have

$$\begin{aligned} y \in F(h)^{-1}(\varphi_A(a)) &\Leftrightarrow (F(h))(y) \in \varphi_A(a) \\ &\Leftrightarrow h^{-1}(y) \in \varphi_A(a) \\ &\Leftrightarrow a \in h^{-1}(y) \\ &\Leftrightarrow h(a) \in y \\ &\Leftrightarrow y \in \varphi_B(h(a)) \end{aligned}$$

and by definition of the topology of X_B, $\varphi_B(h(a))$ is open in X_B.

□

Proposition 7.7 *Let $f : Y \to X$ be a continuous map. Let*

$$(G(f))(U) := f^{-1}(U) \text{ for all } U \in Cl(X).$$

Then $G(f) : Cl(X) \to Cl(Y)$ is a Boolean homomorphism.

Proof. Let $U \in Cl(X)$ then certainly $f^{-1}(U) \in Cl(Y)$, as f is continuous. So $G(f) : Cl(X) \to Cl(Y)$ is well-defined. Clearly $(G(f))(\emptyset) = \emptyset$ and $(G(f))(X) = Y$. Furthermore, taking the inverse image of subsets respects union and intersection. That is,

$$\begin{aligned} (G(f))(U \cup V) &= (G(f))(U) \cup (G(f))(V) \\ (G(f))(U \cap V) &= (G(f))(U) \cap (G(f))(V). \end{aligned}$$

So, by Lemma 3.10, $G(f)$ is a Boolean homomorphism.

□

Theorem 7.8 *The maps $F : \mathcal{B} \to \mathcal{S}$ and $G : \mathcal{S} \to \mathcal{B}$ as defined above are contravariant functors.*

Proof: The results from the previous chapter show that $F(B) \in \mathrm{Obj}(\mathcal{S})$ for all $B \in \mathrm{Obj}(\mathcal{B})$ and $G(X) \in \mathrm{Obj}(\mathcal{B})$ for all $X \in \mathrm{Obj}(\mathcal{S})$. Furthermore propositions 7.6 end 7.7 show that $F(f) \in \mathrm{Mor}(\mathcal{S})$ for all $f \in \mathrm{Mor}(\mathcal{B})$ and $G(h) \in \mathrm{Mor}(\mathcal{B})$ for all $h \in \mathrm{Mor}(\mathcal{S})$. We will show that F preserves the identity morphisms and composition and leave the corresponding proof for G to the reader.
Let B be a Boolean algebra and $y \in X_B$. Then $(F(\mathrm{id}_B))(y) := \mathrm{id}_B^{-1}(y) = y = \mathrm{id}_{F(B)}(y)$. So $F(\mathrm{id}_B) = \mathrm{id}_{F(B)}$.
Now let $g : A \to B$ and $h : B \to C$ be Boolean homomorphisms and $y \in X_C$. Then:

$$\begin{aligned} F(h \circ g)(y) &= (h \circ g)^{-1}(y) \\ &= (g^{-1}(h^{-1}(y)) \\ &= (F(g) \circ F(h))(y) \end{aligned}$$

□

We have already observed that the composition of F and G maps objects to isomorphic copies of these objects. We will now show that in a similar way morphisms are mapped to 'isomorphic' morphisms.

Theorem 7.9 *Let A, B be Boolean algebras and X, Y Stone spaces. Let $h : A \to B$ be a Boolean homomorphism and $f : Y \to X$ a continuous map. The functors $F : \mathcal{B} \to \mathcal{S}$ and $G : \mathcal{S} \to \mathcal{B}$ as defined above make the following diagrams commute.*

$$\begin{tikzcd}
A \arrow[r, "h"] \arrow[d, shift right, "\varphi_A"'] & B \arrow[d, shift right, "\varphi_B"'] \\
GF(A) \arrow[r, "GF(h)"'] \arrow[u, shift right, "\varphi_A^{-1}"'] & GF(B) \arrow[u, shift right, "\varphi_B^{-1}"']
\end{tikzcd}
\qquad
\begin{tikzcd}
Y \arrow[r, "f"] \arrow[d, shift right, "\psi_Y"'] & X \arrow[d, shift right, "\psi_X"'] \\
FG(Y) \arrow[r, "FG(f)"'] \arrow[u, shift right, "\psi_Y^{-1}"'] & FG(X) \arrow[u, shift right, "\psi_X^{-1}"']
\end{tikzcd}$$

Proof. We just show $GF(h) \circ \varphi_A = \varphi_B \circ h$ as an instructive example and leave the rest of the equalities to the reader. Let $a \in A$ and $y \in X_B$. Then

$$\begin{array}{rcl} y \in (GF(h) \circ \varphi_A)(a) & \Leftrightarrow & y \in GF(h)(\varphi_A(a)) \\ & \Leftrightarrow & y \in (F(h))^{-1}(\varphi_A(a)) \\ & \Leftrightarrow & (F(h))(y) \in \varphi_A(a) \\ & \Leftrightarrow & h^{-1}(y) \in \varphi_A(a) \\ & \Leftrightarrow & a \in h^{-1}(y) \\ & \Leftrightarrow & h(a) \in y \\ & \Leftrightarrow & y \in \varphi_B(h(a)) \\ & \Leftrightarrow & y \in (\varphi_B \circ h)(a) \end{array}$$

□

Theorems 7.5 and 7.9 establish a tight correspondence between Boolean algebras and Stone spaces and their structure preserving maps. The functors F and G give rise to what is known as a **categorical duality**. Since the notion of duality is central in our theory we will spend some words on the meaning of this concept.

In general, duality has to do with translating mathematical concepts and theorems to other mathematical concepts and theorems, back and forth. Roughly speaking, there are two versions of duality: dualities that relate concepts and theorems of one and the same mathematical discipline (like order duality) and dualities that relate concepts and theorems of two different branches of mathematics. The duality between Boolean algebras and Stone spaces is of the latter kind, it relates the theory of algebra and the theory of topology.

The duality between Boolean algebras and Stone spaces is a very powerful tool. It translates algebraic objects to topological objects, without losing any structural information about the original object. As Theorem 7.9 shows, the same holds for the translation of maps. But there is more. In principle, it is possible to dualize every fact on Boolean algebras to its topological counterpart and vice versa. This often gives surprising and illuminating results in both the theory of algebra as well as the theory of topology. Because of the twist in the domain and codomain when we translate maps, some problems become easier to solve after translating them to their dual counterpart. The change of the direction of the maps is also the reason why the term 'duality' applies here. As we have seen before duality in mathematics usually refers to certain things being turned upside down.

In the last two sections of this chapter we give examples of the translation of algebraic concepts into their topological dual concepts. In the next section we show how subalgebras relate to quotients of the dual spaces. In the last section of this chapter we will see how an additional operation on a Boolean algebra gives rise to a relational structure on the dual space.

7.3 Subalgebras and quotient spaces

It is part of the standard material on Stone duality that injective Boolean algebra homomorphisms correspond exactly to the surjective continuous maps between Stone spaces. In this section we translate this to a correspondence between subalgebras and certain equivalence relations on Stone spaces. A form of this material can also be found in [1] and [9].

Let X be a topological space and $E \subseteq X^2$ an equivalence relation on X. Furthermore let $\pi : X \to X/E$ be the canonical embedding that maps every element of X to its equivalence class. It is a well-known result in topology that

$$\mathcal{T}_{X/E} := \{V \subseteq X/E \mid \pi^{-1}(V) \text{ open in } X\}$$

defines a topology on the set X/E of equivalence classes that makes π continuous.

Definition 7.10 We call X/E with this topology the **quotient space** of X with respect to E.

If X is a Stone space and E an arbitrary equivalence relation on X, the quotient space X/E is in general not again a Stone space.

Definition 7.11 Let X be a Stone space and $E \subseteq X^2$ an equivalence relation on X. We call E a **Stone equivalence** if and only if X/E is again a Stone space.

In this section we will discover that subalgebras of a Boolean algebra B correspond one-to-one to Stone equivalence relations on X_B.

Observe that every map $f : X \to Y$ between topological spaces X and Y defines a binary relation $\mathrm{Ker}(f)$ on X in the following way

$$\mathrm{Ker}(f) = \{(x, y) \in X^2 \mid f(x) = f(y)\}.$$

It is easy to see that $\mathrm{Ker}(f)$ is an equivalence relation. Hence every map $f : X \to Y$ gives rise to a quotient space $X/\mathrm{Ker}(f)$. In addition, the map f factors through the quotient space. That is,

$$\begin{tikzcd} X \arrow[rr, "f"] \arrow[dr, "\pi"'] & & Y \\ & X/\mathrm{Ker}(f) \arrow[ur, "\tilde{f}"'] & \end{tikzcd}$$

commutes, where $\pi(x) = x/\mathrm{Ker}(f)$ and $\tilde{f}(x/\mathrm{Ker}(f)) = f(x)$ for all $x \in X$. Notice that by the definition of a quotient space, the map $\tilde{f}$ is always continuous. It is clearly also injective.

Definition 7.12 Let $(X, \mathcal{T}_X)$ and $(Y, \mathcal{T}_Y)$ be topological spaces. A map $f : X \to Y$ is called a **topological quotient map** if the map $\tilde{f} : X/\mathrm{Ker}(f) \to Y$ defined by

$$\tilde{f}(x/\mathrm{Ker}(f)) = f(x)$$

is a homeomorphism. That is, if and only if $\tilde{f}$ satisfies the following conditions:

i) $\tilde{f}$ is surjective

ii) $\tilde{f}^{-1}(U)$ is open in $X \iff U$ is open in Y.

Lemma 7.13 *Let A, B be Boolean algebras and $f : A \to B$ a Boolean algebra embedding. Then the dual map, $F(f) : X_B \to X_A$, is surjective.*

Proof. Suppose $x \in X_A$. We have to show that there is an $y \in X_B$ such that $(F(f))(y) = f^{-1}(y) = x$. Because x is a prime filter of $A, \uparrow f(x)$ is a proper filter in B. Hence, by Theorem 5.17, there is a prime filter that contains $\uparrow f(x)$. Define y to be such a prime filter. Then $f^{-1}(y)$ is a prime filter in X_A and $x = f^{-1}(f(x)) \subseteq f^{-1}(y)$. But x is maximal, hence $f^{-1}(y) = x$.

□

Theorem 7.14 *Let A, B be Boolean algebras and $f : A \to B$ a Boolean algebra embedding. Then the dual map, $F(f) : X_B \to X_A$, is a topological quotient map.*

Proof. By Lemma 7.13 $F(f)$ is surjective. Furthermore, by Theorem 7.6, $F(f)$ is continuous. So if $U \subseteq X_A$ is open, then $F(f)^{-1}(U)$ is open.
Now assuming $V := F(f)^{-1}(U)$ is open, we have to prove that U is open. That is, $U = \bigcup_{a \in A'} \varphi_A(a)$ for some $A' \subseteq A$.
Let $y \in V$. We will show that there exists an $a \in A$ such that $y \in \varphi_B(f(a)) \subseteq V$. Define

$$[y] := \{y' \in X_B \mid F(f)(y) = F(f)(y')\}.$$

Observe that

$$\begin{aligned} y' \in [y] &\Leftrightarrow f^{-1}(y') = f^{-1}(y) \\ &\Leftrightarrow f(a) \in y' \text{ for all } a \in f^{-1}(y) \\ &\Leftrightarrow y' \in \varphi_B(f(a)) \text{ for all } a \in f^{-1}(y) \\ &\Leftrightarrow y' \in \bigcap_{a \in f^{-1}(y)} \varphi_B(f(a)). \end{aligned}$$

Hence $[y] = \bigcap_{a \in f^{-1}(y)} \varphi_B(f(a))$. Furthermore $[y] \subseteq V$ because V is saturated. This implies

$$\begin{aligned} X_B &= V \cup (\bigcap_{a \in f^{-1}(y)} \varphi_B(f(a)))^c \\ &= V \cup (\bigcup_{a \in f^{-1}(y)} \varphi_B(f(a))^c). \end{aligned}$$

Note that V is open and $\varphi_B(f(a))^c$ is open for every $a \in A$. Hence, by compactness of X_B, there are $a_1, \ldots, a_n \in f^{-1}(y)$ such that

$$X_B = V \cup (\bigcup_{i=1}^{i=n} \varphi_B(f(a_i))^c).$$

So

$$\bigcap_{i=1}^{i=n} (\varphi_B(f(a_i))^c)^c = \bigcap_{i=1}^{i=n} \varphi_B(f(a_i)) \subseteq V.$$

Define $a := a_1 \wedge \ldots \wedge a_n$. Then $a \in f^{-1}(y)$ so $f(a) \in y$. That is, $y \in \varphi_B(f(a))$. Moreover

$$\bigcap_{i=1}^{i=n} \varphi_B(f(a_i)) = \varphi_B(f(a_1) \wedge \ldots \wedge f(a_n)) = \varphi_B(f(a)) \subseteq V.$$

So for every $y \in V$ there is an $a \in A$ such that $y \in \varphi_B(f(a)) \subseteq V$. This means that $V = \bigcup_{a\in A'} \varphi_B(f(a))$ for some $A' \subseteq A$. Finally, we observe that

$$\begin{aligned}
& x \in (F(f))(\bigcup_{a\in A'} \varphi_B(f(a))) \\
\Leftrightarrow\quad & \exists a \in A', x \in (F(f))(\varphi_B(f(a))) \\
\Leftrightarrow\quad & \exists a \in A', \exists y \in \varphi_B(f(a)), x = f^{-1}(y) \\
\Leftrightarrow\quad & \exists a \in A', \exists y \in X_B \text{ such that } f(a) \in y, x = f^{-1}(y) \\
\Leftrightarrow\quad & \exists a \in A' \text{ with } a \in x \text{ (as } f \text{ is onto)} \\
\Leftrightarrow\quad & x \in \bigcup_{a\in A'} \varphi_A(a).
\end{aligned}$$

Which implies $U = (F(f))(V) = \bigcup_{a\in A'} \varphi_A(a)$ for some $A' \subseteq A$.

□

We will now see how a subalgebra of a Boolean algebra gives rise to a Stone equivalence relation on its dual space.

Definition 7.15 Let B be a Boolean algebra and A a Boolean subalgebra of B. We define a relation θ_A on the dual space of B in the following way

$$\theta_A = \{(x, y) \in X_B^2 \mid \forall a \in A : a \in x \Leftrightarrow a \in y\}.$$

The next lemma shows that θ_A is equal to the kernel of some map. So, in particular, it shows that θ_A is an equivalence relation.

Lemma 7.16 *Let B be a Boolean algebra and A a Boolean subalgebra of B. Furthermore, let $i : A \to B$ the inclusion of A in B. Then*

$$\theta_A = Ker(F(i))$$

where $F(i) : X_B \to X_A$ is the dual of the inclusion map i.

Proof. Observe that for all $x, y \in X_B$

$$\begin{aligned}
(x, y) \in Ker(F(i)) \quad \Leftrightarrow\quad & F(i)(x) = F(i)(y) \\
\Leftrightarrow\quad & i^{-1}(x) = i^{-1}(y) \\
\Leftrightarrow\quad & x \cap A = y \cap A \\
\Leftrightarrow\quad & \forall a \in A : a \in x \Leftrightarrow a \in y \\
\Leftrightarrow\quad & (x, y) \in \theta_B.
\end{aligned}$$

So $\theta_B = Ker(F(i))$.

□

Now we can easily deduce the following proposition.

Proposition 7.17 *Let A, B be Boolean algebras such that A is a subalgebra of B. Then*

$$\tilde{F(i)} : X_B/\theta_A \to X_A.$$

is a homeomorphism. In particular, θ_A is a Stone equivalence for all subalgebras A of B.

Proof. By Lemma 7.16 we have

$$X_B/\theta_A = X_B/\mathrm{Ker}(F(i)),$$

where $F(i)$ is the dual of the inclusion map $i : A \to B$. By Theorem 7.14 the map $F(i)$ is a topological quotient map. That is,

$$\tilde{F(i)} : X_B/\mathrm{Ker}(F(i)) \to X_A$$

is a homeomorphism. Certainly X_A is a Stone space, as it is the dual space of the Boolean algebra A.

□

This proposition gives rise to another way of looking at the dual space of a subalgebra A of B. We can either think of X_A as the set of prime filters of the algebra A or we can consider X_A to be the set of all equivalence classes of X_B under the equivalence relation θ_A. That is, that elements of X_A are sets of prime filters of X_B.

Now suppose A and C are both subalgebras of a Boolean algebra B. This gives rise to two Stone equivalence relations θ_A and θ_C on X_B. The following theorem points out the relationship between subalgebras and their corresponding equivalence relations.

Proposition 7.18 *Let B be Boolean algebra and A and C subalgebras of B. Then*

$$A \text{ is a subalgebra of } C \quad \Longleftrightarrow \quad \theta_C \subseteq \theta_A.$$

For the proof of this proposition we need the following lemma on the existence of certain prime filters.

Lemma 7.19 *Let B be a Boolean algebra, A a proper subalgebra of B and $a \in B \setminus A$. There exist prime filters x, y of B such that $x \cap A = y \cap A$ and $a \in x, a \notin y$.*

Proof. Consider $F = \uparrow a$ and $I = \downarrow(\downarrow a \cap A)$. Then (F, I) is a disjoint filter-ideal pair and applying the Prime Filter Theorem we obtain a prime filter x of B such that $F \subseteq x$ and $x \cap I = \emptyset$. Clearly $a \in x$.
Now consider $F' = \uparrow(\uparrow x \cap A)$ and $I' = \downarrow a$. Also (F', I') is a disjoint filter-ideal pair and applying the Prime Filter Theorem we obtain a prime filter y of B such that $F' \subseteq y$ and $I' \cap y = \emptyset$. As $a \in I'$ this implies $a \notin y$.
Now let $b \in x \cap A$. Then $b \in F'$ and thus $b \in y$. That is, $x \cap A \subseteq y \cap A$. Both $x \cap A$ and $y \cap A$ are prime filters of A hence $x \cap A = y \cap A$.

□

Proof of Theorem 7.18. $\Rightarrow$ Let A be a subalgebra of C and suppose and $(x, y) \in \theta_C$. This implies

$$\begin{aligned} x \cap A &= x \cap (C \cap A) \\ &= (x \cap C) \cap A \\ &= (y \cap C) \cap A \\ &= y \cap (C \cap A) \\ &= y \cap A. \end{aligned}$$

That is, $(x, y) \in \theta_A$.

$\Leftarrow$ Now suppose $\theta_C \subseteq \theta_A$. Assume there exists $a \in A$ such that $a \notin C$. Then, by Lemma 7.19, there exist $x, y \in X_B$ such that $x \cap C = y \cap C$ and $a \in x, a \notin y$. That is $x \cap A \neq y \cap A$, a contradiction.
Hence $A \subseteq C$. As we know that both A and C are subalgebras of B this implies that A is a subalgebra of C.

□

Corollary 7.20 *Let A, C be subalgebras of a Boolean algebra B. Then*

$$\theta_A = \theta_C \Longleftrightarrow A = C.$$

This corollary tells us in particular that the map

$$A \mapsto \theta_A.$$

that sends subalgebras of a Boolean algebra to Stone equivalence relations on its dual space, is injective. In a similar way we can construct a map that sends (Stone) equivalence relations to subalgebras.

Definition 7.21 Let B be a Boolean algebra and $\theta \subseteq X_B^2$ an equivalence relation on the dual space of B. We define

$$A_\theta = \{b \in B \mid \forall (x, y) \in \theta : b \in x \Leftrightarrow b \in y\}.$$

This now gives a correspondence between equivalences on X_B and subalgebras of B as is expressed in the following lemma.

Lemma 7.22 *Let B be a Boolean algebra and θ an equivalence on X_B. Then A_θ is a subalgebra of B.*

Proof. First of all note that $0 \notin x$ for all $x \in X_B$ and $1 \in x$ for all $x \in X_B$. Hence $0, 1 \in A_\theta$. Secondly, note that for all $b \in B$ and $x \in X_B$

$$b \in x \Leftrightarrow \neg b \notin x.$$

Hence $b \in A_\theta$ implies $\neg b \in A_\theta$. Now suppose $a, b \in A_\theta$ and $(x, y) \in \theta$. Then

$$a \in x \Leftrightarrow a \in y \text{ and } b \in x \Leftrightarrow b \in y.$$

Now there are four possibilities:

1. $a, b \in x$. Then $a, b \in y$. Hence $a \vee b \in x$ and $a \vee b \in y$, as x and y are both up-sets. Also $a \wedge b \in x$ and $a \wedge b \in y$ as x and y are prime and thus closed under $\wedge$.
2. $a \in x$ and $b \notin x$. Then $a \in y$ and $b \notin y$. Hence $a \vee b \in x$ and $a \vee b \in y$. Furthermore $a \wedge b \notin x$ and $a \wedge b \notin y$ as x and y do not contain b.
3. $a \notin x$ and $b \in x$. This case is similar to the previous.
4. $a \notin x$ and $b \notin x$. Then $a, b \notin y$ and hence $a \vee b \notin x$ and $a \vee b \notin y$ as x and y are both prime. And obviously $a \wedge b \notin x$ and $a \wedge b \notin y$.

Hence for $a, b \in A_\theta$ and $(x, y) \in \theta$ we have

$$a \vee b \in x \Leftrightarrow a \vee b \in y \text{ and } a \wedge b \in x \Leftrightarrow a \wedge b \in y.$$

That is, $a \vee b, a \wedge b \in A_\theta$. Together with the previous observations this proves that A_θ is a subalgebra of B.

□

Proposition 7.23 *Let B be a Boolean algebra and C a Boolean subalgebra of B. Furthermore let θ_C be the relation as defined in Definition 7.15. Then*

$$A_{\theta_C} = C.$$

Proof. Let $b \in C$ and $(x, y) \in \theta_C$. That is $x \cap C = y \cap C$. So $b \in x \Leftrightarrow b \in y$. Hence $b \in A_{\theta_C}$.
Now suppose $b \notin C$. Then, by lemma 7.19, there exist prime filters x, y of B, such that $x \cap C = y \cap C$ (that is $(x, y) \in \theta_C$) and $b \in x, b \notin y$. This implies $b \notin A_{\theta_c}$.
Thus $A_{\theta_C} = C$.

□

Lemma 7.22 tells us that the map

$$\theta \mapsto A_\theta$$

sends equivalence relations on X_B to subalgebras of B. In addition, Proposition 7.23 shows that it is onto and that it is the inverse of the map

$$A \mapsto \theta_A$$

that relates subalgebras to Stone equivalence relations. These observations lead to the following corollary.

Corollary 7.24 *Let B be a Boolean algebra. There is a one-to-one correspondence between subalgebras of a Boolean algebra B and Stone equivalence relations on its dual space X_B given by*

$$\begin{array}{rcl} \theta & \mapsto & A_\theta = \{a \in B \mid \forall (x, y) \in \theta : (a \in x \Leftrightarrow a \in y)\} \\ A & \mapsto & \theta_A = \{(x, y) \in X_B^2 \mid \forall a \in A : (a \in x \Leftrightarrow a \in y)\}. \end{array}$$

7.4 Extended Stone duality

In this section we consider a Boolean algebra B equipped with an additional binary operation $\cdot$. We assume that the additional operation satisfies for all $a, b, c \in B$

$$\begin{array}{rcl} (a \vee b) \cdot c & = & a \cdot c \vee b \cdot c \\ a \cdot (b \vee c) & = & a \cdot b \vee a \cdot c. \end{array}$$

And for all $a \in B$

$$\begin{aligned} a \cdot 0 &= 0 \\ 0 \cdot a &= 0. \end{aligned}$$

That is, we assume that the operation preserves binary and empty joins in each coordinate. It is easy to see this is equivalent to preserving all finite joins in each coordinate. Note that the product of languages on the Boolean algebra $\mathcal{P}(A^*)$ as defined in Chapter 1 is exactly of this kind. Also in the general case we will refer to the additional operation as the product operation.

In this section we will show how an operation on a Boolean algebra satisfying the properties above defines a ternary relation on the dual space of the Boolean algebra. This observation gives rise to an extension of the duality between Boolean algebras and Stone spaces developed so far. The theory of extended Stone duality for additional operations, including the proofs of all the theorems discussed in this section, can be found in [11]. Here we will just present the minimal amount of information needed for the development of the rest of the theory.

Let $\langle B, \cdot \rangle$ be a Boolean algebra with an additional operation. We can define a ternary relation R on the dual space of B in the following way:

$$R = \{(x, y, z) \in X_B^3 \mid \forall a, b \in B : [a \in x, b \in y] \Rightarrow a \cdot b \in z\}.$$

In order for this relation to make sense as the dual concept of the additional operation, we have to make sure that we are able to recover $\cdot$ from R. Let us first fix some notation.

For $R \subseteq X^3$ a ternary relation and $x, y \in X$, we define

$$\begin{aligned} R[x, y, \_] &= \{z \in X \mid R(x, y, z)\} \\ R[x, \_, \_] &= \{(y, z) \in X^2 \mid R(x, y, z)\}. \end{aligned}$$

The sets $R[\_, y \_], R[x, \_, z]$, etc. are defined in a similar way. Furthermore given subsets U, V of X we define

$$\begin{aligned} R[U, V, \_] &= \{z \in X \mid \exists x \in U, \exists y \in V : R(x, y, z)\} \\ R[U, \_, \_] &= \{(y, z) \in X^2 \mid \exists x \in U : R(x, y, z)\}. \end{aligned}$$

The sets $R[\_, V, \_], R[U, \_, W]$, etc. are defined in a similar way.

The next theorem shows how the operation $\cdot$ can be recovered from the relation R.

Theorem 7.25 *Let $\langle B, \cdot \rangle$ be a Boolean algebra with an additional operation that is finite join preserving and let R be the ternary relation on X_B as defined above. Then for all $a, b \in B$*

$$a \cdot b = \varphi_B^{-1}(R[\varphi_B(a), \varphi_B(b), \_]).$$

That is, given the relation R on X_B, we are able to recover the operation $\cdot$ on B.

Proof. Let $a, b \in B$. Since φ_B is an isomorphism and hence bijective, it is sufficient to prove

$$\varphi_B(a \cdot b) = \varphi_B(\varphi_B^{-1}(R[\varphi_B(a), \varphi_B(b), \_])).$$

That is,

$$\varphi_B(a \cdot b) = R[\varphi_B(a), \varphi_B(b), \_].$$

Let $z \in R[\varphi_B(a), \varphi_B(b), \_]$. Then we can find $x \in \varphi_B(a)$ and $y \in \varphi_B(b)$ such that $R(x, y, z)$. Since $a \in x$ and $b \in y$, this implies $a \cdot b \in z$. That is $z \in \varphi_B(a \cdot b)$. So

$$\varphi_B(a \cdot b) \supseteq R[\varphi_B(a), \varphi_B(b), \_].$$

Now suppose $z \in \varphi_B(a \cdot b)$. We want to find $x \in \varphi_B(a)$ and $y \in \varphi_B(b)$ such that $z \in R(x, y, z)$. In other words, we have to find $x, y \in X_B$, such that $a \in x, b \in y$ and $x \cdot y \subseteq z$. Since $a \in x$ we have to make sure

$$y \subseteq \{c \in B \mid a \cdot c \in z\}.$$

In other words, we are looking for a prime filter y of B that does not intersect the set

$$S = \{c \in B \mid a \cdot c \notin z\}.$$

We show that S is an ideal disjoint from $\uparrow b$ and then apply the Prime Filter Theorem to obtain y.
First of all note that $a \cdot 0 = 0 \notin z$. Hence $0 \in S$, so S is non-empty. Suppose $s \in S$ and $c \in B$ with $c \leq s$. Then $a \cdot c \leq a \cdot s \notin z$, which implies $a \cdot c \notin z$ as z is an up-set. Hence $c \in S$, so S is a down-set. Now suppose $s_1, s_2 \in S$.

That is, $a \cdot s_1, a \cdot s_2 \notin z$. As z is a prime filter, this implies $a \cdot s_1 \vee a \cdot s_2 \notin z$. But $a \cdot s_1 \vee a \cdot s_2 = a \cdot (s_1 \vee s_2)$ as $\cdot$ is join preserving. Hence $s_1 \vee s_2 \in S$. So S is an ideal. Finally $a \cdot b \in z$, implies $b \notin S$. Since S is a down-set this implies $S \cap {\uparrow} b = \emptyset$.
By the Prime Filter Theorem there exists a prime filter y such that ${\uparrow} b \subseteq y$ and $y \cap S = \emptyset$. Obviously $y \in \varphi_B(b)$. Now we are going to construct a prime filter $x \in \varphi_B(a)$, such that $x \cdot y \subseteq z$ in a similar way. Define

$$T = \{c \in B \mid \exists c' \in y : c \cdot c' \notin z\}.$$

We show that T is an ideal disjoint from $\uparrow a$.
First of all note that $0 \cdot b = 0 \notin z$. Hence $0 \in T$, so T is non-empty. Suppose $t \in T$ and $c \in B$ with $c \leq t$. As $t \in T$ there exists $c' \in y$ with $t \cdot c' \notin z$. But $c \cdot c' \leq t \cdot c'$, hence $c \cdot c' \notin z$. That is, $c \in T$. So T is a down-set. Now suppose $t_1, t_2 \in T$. That is, there exists $c_1, c_2 \in y$ such that $t_1 \cdot c_1 \notin z$ and $t_2 \cdot c_2 \notin z$. Hence $t_1 \cdot (c_1 \wedge c_2) \notin z$ and $t_2 \cdot (c_1 \wedge c_2) \notin z$. This implies $t_1 \cdot (c_1 \wedge c_2) \vee t_2 \cdot (c_1 \wedge c_2) \notin z$ as z is a prime filter. But $t_1 \cdot (c_1 \wedge c_2) \vee t_2 \cdot (c_1 \wedge c_2) = (t_1 \vee t_2) \cdot (c_1 \wedge c_2)$ as $\cdot$ is join-preserving. Furthermore $c_1 \wedge c_2 \in y$ since y is a filter, hence $t_1 \vee t_2 \in T$. Finally recall that for all $c \in y$, $a \cdot c \in z$, so $a \notin T$. Also T is a down-set, so $T \cap {\uparrow} a = \emptyset$. So T is an ideal disjoint from ${\uparrow} a$. By the prime filter theorem we can construct an prime filter that contains ${\uparrow} a$ and is disjoint from T. Define x to be such a prime filter. Then obviously $a \in x$ and $x \cdot y \subseteq z$.

□

The ternary relations arising as the duals of an additional operation on a Boolean algebra are characterized in the following proposition.

Proposition 7.26 *Let $\langle B, \cdot \rangle$ be a Boolean algebra with an additional operation that is finite join preserving and let R be the ternary relation on X_B as defined above. Then*

i) for all clopen subsets U and V of X_B the set $R[U, V, \_]$ is clopen;

ii) for each $x \in X_B$, the set $R[\_, \_, x]$ is closed in the product topology on X_B^2.

We will not prove this proposition here but note that the first property is a direct consequence of the proof of Theorem 7.25.

Now the full duality between Boolean algebras with an additional operations and Stone spaces equipped with a ternary relation is expressed in the following theorem. This theorem is proved (in a more general setting) in [11].

Theorem 7.27 *There is a one-to-one correspondence between Boolean algebras with an additional finite join preserving operation and Stone spaces with a ternary relation satisfying the two properties in Proposition 7.26.*

In the same way as we have seen before, we can extend this correspondence to a categorical correspondence involving structure preserving maps as well. Although we do not need this full correspondence in the rest of the theory, we just give it here as another example of a one-to-one relationship between an algebraic concept and its topological counterpart. Remember that in the previous section we had a similar one-to-one relationship between subalgebras of a Boolean algebra and Stone equivalence relations on its dual space. In part IV we encounter a third correspondence that combines these two. That is, we will show that certain subalgebras, the so-called quotienting subalgebras, correspond one-to-one to Stone equivalences that 'respect' the relation R dual to the additional operations.

Part IV

Languages, Boolean Algebras and Duality

Chapter 8

Quotienting subalgebras

The main result in this chapter is a specialization of the duality between subalgebras and Stone equivalence relations for so-called quotienting subalgebras. Before we state this result we will motivate why, from a computational point of view, it makes sense to consider quotienting algebras.

8.1 Quotienting subalgebras and computation

In order to be able to apply any of the duality theory developed in the previous chapters, we have to assume that the classes of languages we work with are Boolean subalgebras of $\mathcal{P}(A^*)$. In this section we will argue that classes of interest to the theory of computation are not only Boolean subalgebras, but often so-called *quotienting subalgebras* of $\mathcal{P}(A^*)$. We will motivate this by considering the class of regular languages and their correspondence to finite automata.

Recall that we defined two quotient operations / and \ on $\mathcal{P}(A^*)$ as follows

$$\begin{aligned} L/K &= \{u \in A^* \mid uv \in L \text{ for all } v \in K\} \\ K\backslash L &= \{u \in A^* \mid vu \in L \text{ for all } v \in K\}. \end{aligned}$$

By considering the singleton languages we can easily specialize these definitions for quotients by words in A^*. That is,

$$\begin{aligned} L/v &= \{u \in A^* \mid uv \in L\} \\ v\backslash L &= \{u \in A^* \mid vu \in L\}. \end{aligned}$$

Let L be a regular language and $\mathcal{A}_L$ an automaton that recognizes L. Then for any word $v \in A^*$, the automata that recognize L/v and $v\backslash L$ can be obtained from the automaton $\mathcal{A}_L$ by simply changing the initial or final state(s). The automaton for the language L/v is obtained from $\mathcal{A}_L$ by moving every final state 'backwards' along a path with label v. The automaton for the language $v \setminus L$ is obtained from $\mathcal{A}_L$ by moving the initial state 'forward' along the path with label v. We will illustrate this by some examples.

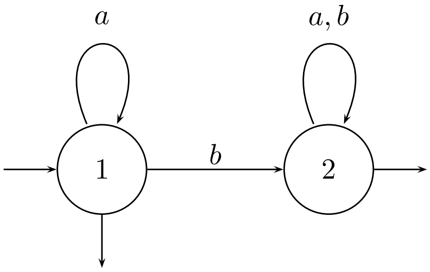
Example 8.1 Recall the automaton $\mathcal{A}_1$ from example 2.5 in Chapter 2. Let $L(\mathcal{A}_1)$ be the language recognized by this automaton. We will show how to obtain automata for the languages $L(\mathcal{A}_1)/ab$ and $ab \setminus L(\mathcal{A}_1)$, respectively. The 'backward' paths

$$2 \overset{b}{\leftarrow} 2 \overset{a}{\leftarrow} 2$$

and

$$2 \overset{b}{\leftarrow} 1 \overset{a}{\leftarrow} 1$$

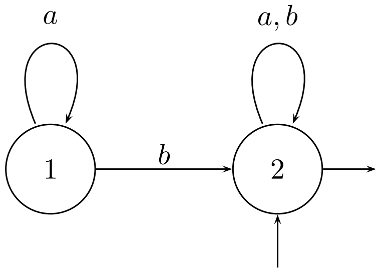
show that the automaton for $L(\mathcal{A}_1)/ab$ is obtained from $\mathcal{A}_1$ by adding the final state 1. Hence the language $L(\mathcal{A}_1)/ab$ is recognized by the following automaton.



Similarly the path

$$1 \overset{a}{\rightarrow} 1 \overset{b}{\rightarrow} 2$$

shows that the automaton for $ab \setminus L(\mathcal{A}_1)$ is obtained from $\mathcal{A}_1$ by changing the initial state from 1 to 2. Hence the language $ab \setminus L(\mathcal{A}_1)$ is recognized by the following automaton.



It is not hard to see that these constructions work for arbitrary finite automata. Note that it may be possible that there are no backwards paths for certain words. Then the set of final states becomes the empty set and the corresponding language the empty language.

The observations above show that the complexity of an automaton, and therefore the complexity of a language, does not change significantly when taking its left or right quotient by a word. That is, the transition structure stays the same, only the final or initial state may have changed. So if $\mathcal{C} \subseteq \mathrm{Reg}(A^*)$ is a class of languages that represents the complexity of a certain model of computation, it will usually be the case that both L/u and $u\backslash L$ are both elements of $\mathcal{C}$ for all $u \in A^*$.

The next proposition shows that, in the case of the regular languages, closure under the quotient operations with respect to singleton languages in the denominator is the same as closure under the quotient operations with respect to arbitrary languages in the denominator.

Proposition 8.2 *Let $\mathcal{C}$ be a Boolean subalgebra of Reg(A^*). Then the following are equivalent*

i) $u\backslash L, L/u \in \mathcal{C}$ *for all* $L \in \mathcal{C}, u \in A^*$

ii) $K\backslash L, L/K \in \mathcal{C}$ *for all* $L \in \mathcal{C}, K \in \mathcal{P}(A^*)$.

The proof of this proposition uses the following lemma.

Lemma 8.3 *Let $L, K \in \mathcal{P}(A^*)$ and $\sim$ a congruence on A^* that saturates L. Then*

$$\overline{K}\backslash L = K\backslash L \text{ and } L/\overline{K} = L/K$$

where $\overline{K}$ is the closure of K with respect to $\sim$.

Proof. Certainly $K \subseteq \overline{K}$ implies $\overline{K} \setminus L \subseteq K \setminus L$. Now let $u \in K \setminus L$. That is, $ku \in L$ for all $k \in K$. Suppose $k' \in \overline{K}$. Then there is a $k \in K$ with $k \sim k'$. As $\sim$ is a congruence this implies $ku \sim k'u$. Hence $k'u \in L$, since $\sim$ saturates L.

□

Proof of Proposition 8.2. Assume $u\backslash L, L/u \in \mathcal{C}$ for all $L \in \mathcal{C}$ and $u \in A^*$. Suppose $L \in \mathcal{C}$ and $K \in \mathcal{P}(A^*)$. As $L \in \mathrm{Reg}(A^*)$, the syntactic congruence

$\sim_L$ on A^* is of finite index. Hence the closure $\overline{K}$ of K with respect to this congruence is a union of finitely many equivalence classes. That is, there exist $u_1, \ldots, u_n$ such that

$$\overline{K} = \bigcup_{i=1}^{i=n} \overline{u_i}$$

Applying Lemma 8.3 we obtain

$$\begin{aligned} K\backslash L &= \overline{K}\backslash L \\ &= (\bigcup_{i=1}^{i=n} \overline{u_i})\backslash L \\ &= \bigcap_{i=1}^{i=n} (\overline{u_i}\backslash L) \\ &= \bigcap_{i=1}^{i=n} (u_i\backslash L). \end{aligned}$$

As $\mathcal{C}$ is a Boolean algebra it is in particular closed under finite intersection. Hence, $K\backslash L \in \mathcal{C}$. In a similar way we can show $L/K \in \mathcal{C}$. The other direction of the equivalence is obvious.

□

The observations above lead to the following definition of a special kind of subalgebra of $\mathcal{P}(A^*)$, that involves a special closure property with respect to the additional operations / and \.

Definition 8.4 Let $\langle \mathcal{P}(A^*), \cdot, /, \backslash \rangle$ be the Boolean algebra of languages with additional operations. We say that $\mathcal{C} \subseteq \mathcal{P}(A^*)$ is a **quotienting subalgebra** of $\mathcal{P}(A^*)$ if

i) $\mathcal{C}$ is a Boolean subalgebra of $\mathcal{P}(A^*)$

ii) for all $L \in \mathcal{C}$ and $K \in \mathcal{P}(A^*)$, $L/K \in \mathcal{C}$ and $K\backslash L \in \mathcal{C}$.

That is, $\mathcal{C}$ is a subalgebra closed under the quotient operations with denominators taken in $\mathcal{P}(A^*)$.

For Boolean subalgebras of $\text{Reg}(A^*)$ we get the following corollary as a direct consequence of Proposition 8.2.

Corollary 8.5 *Let $\mathcal{C}$ be a Boolean subalgebra of Reg(A^*). Then the following are equivalent:*

i) $L/u, u\backslash L \in \mathcal{C}$ for all $L \in \mathcal{C}, u \in A^*$.

ii) $\mathcal{C}$ is a quotienting subalgebra of $\mathcal{P}(A^*)$.

Since we argued before that classes of regular languages corresponding to some level of complexity are closed under taking quotients by words, this lemma tells that us that these classes actually are quotienting algebras. Although we can only give a proper motivation in the regular case, it turns out that also outside the regular languages classes of languages related to computational problems are often closed under taking quotients (rather than that they are closed under product, the class of commutative languages we introduce later on is a nice example of this).

One of the questions of interest posed in Chapter 1 was to determine for a certain language L what languages are related to it in terms of computation. The observations above motivate the following definition.

Definition 8.6 For any $L \in \mathcal{P}(A^*)$ we define $\mathcal{Q}_L$ to be the smallest quotienting subalgebra of $\mathcal{P}(A^*)$ such that $L \in \mathcal{Q}_L$.

At least in the regular case $\mathcal{Q}_L$ contains languages that are of a similar complexity as L since the automata for languages in $\mathcal{Q}_L$ can be obtained by moving around initial and final states and applying the Boolean operations to the automaton recognizing L.

One may wonder what the use is of considering quotienting subalgebras instead of ordinary subalgebras. In order to take full advantage of the fact that we are dealing with quotienting algebras, we will explore what special properties the Stone equivalence relations coming from quotienting algebras have. This is the content of the last section of this chapter. Before we get to this, we will consider the results on duality for additional operations for so-called residuated Boolean algebras.

8.2 Residuated Boolean algebras

So far we have only applied Stone duality in its traditional form. We did not yet use the fact that $\mathcal{P}(A^*)$ is equipped with an additional operation and that we have a duality for this operation as observed in Section 7.4. In fact, $\mathcal{P}(A^*)$ is endowed with *three* additional operations, the product and the left and right quotient operations, which form a so-called *residuated family.*

Definition 8.7 Let B be a Boolean algebra and $(\cdot, /, \backslash)$ be a triple of binary operations on B. We call $(\cdot, /, \backslash)$ a residuated family if for all $a, b, c \in B$

$$a \cdot b \leq c \iff a \leq c/b \iff b \leq a\backslash c.$$

We refer to the operation $\cdot$ as the product and to the operations $/$ and $\backslash$ as the left, respectively, right residual of $\cdot$.

The following definition captures the abstract properties of the algebra $\mathcal{P}(A^*)$ and in particular of the additional operations.

Definition 8.8 Let $\langle B, \cdot, /, \backslash\rangle$ be a Boolean algebra with three additional binary operations. We call B a **residuated Boolean algebra** if the operations $(\cdot, /, \backslash)$ form a residuated family on B.

As a result of Theorem 1.3 the algebra $\langle\mathcal{P}(A^*), \cdot, /, \backslash\rangle$ as defined in Chapter 1 is an example of a residuated Boolean algebra.

It can be shown that the triple $(\cdot, /, \backslash)$ being a residuated family implies that $\cdot$ preserves finite joins in each coordinate. In section 7.4 we considered a Boolean algebra with an additional operation that preserved finite joins in each coordinate. Since the product operation of a residuated Boolean algebra has this properties, the extended duality theory developed in section 7.4 applies. That is, a residuated Boolean algebra $\langle B, \cdot, /, \backslash\rangle$ gives rise to a dual space with a ternary relation on it. Recall that the relation R on X_B dual to the additional operation was defined as follows.

$$R \;\; = \;\; \{(x, y, z) \in X_B^3 \mid \forall a, b \in B : [a \in x, b \in y] \Rightarrow a \cdot b \in z\}.$$

Because of the residuation relationship between the operations $(\cdot, /, \backslash)$, we could also have defined R in terms of either $/$ or $\backslash$, as is shown by the following proposition.

Proposition 8.9 *Let $\langle B, \cdot, /, \backslash\rangle$ be a residuated Boolean algebra and $x, y, z \in X_B$. Then the following are equivalent:*

i) $\forall a, b \in B : a \in x, b \in y \Rightarrow a \cdot b \in z$

ii) $\forall a, b \in B : b \in y, a \notin z \Rightarrow a/b \notin x$

iii) $\forall a, b \in B : b \in x, a \notin z \Rightarrow b\backslash a \notin y$

Proof. We only prove i) $\Leftrightarrow$ ii) since the proof of the other equivalence is similar. Observe that

$$\forall a, b \in B : b \in y, a \notin z \Rightarrow a/b \notin x$$

is equivalent to

$$\forall a, b \in B : a/b \in x, b \in y \Rightarrow a \in z.$$

So let $x, y, z \in X_B$ such that

$$\forall a, b \in B : a \in x, b \in y \Rightarrow a \cdot b \in z.$$

Assume $a/b \in x$ and $b \in y$. We want to derive $a \in z$. We have that $a/b \in x$ and $b \in y$ implies $(a/b) \cdot b \in z$. And by residuation

$$a/b \leq a/b \Longleftrightarrow (a/b) \cdot b \leq a.$$

Hence $a \in z$.
Now let $x, y, z \in X_B$ such that.

$$\forall a, b \in B : a/b \in x, b \in y \Rightarrow a \in z.$$

Suppose $a \in x$ and $b \in y$. We want to derive $a \cdot b \in z$. By residuation

$$a \cdot b \leq a \cdot b \Leftrightarrow a \leq (a \cdot b)/b.$$

Hence $a \leq (a \cdot b)/b \in x$. Since $b \in y$ this implies $a \cdot b \in z$.

□

The definition of R in terms of the residuals makes it possible to define a relation R_A on a subalgebra A of B that is closed under the residuals but not necessarily under the product operation. In particular, Proposition 8.9 makes it possible to define a relation R_A on the dual spaces of quotienting subalgebras A of B. Such algebras are the subject of the next section.

8.3 Quotienting subalgebras and R-congruences

In this section we explore, in a general setting, the interaction between the relation R dual to a residuated family of operations and the Stone equivalences coming from quotienting subalgebras. First we need an abstract definition of the concept of quotienting algebra (similar to Definition 8.4).

Definition 8.10 Let $\langle B, \cdot, /, \backslash \rangle$ be a residuated Boolean algebra. We call a subset A of B a **quotienting** subalgebra if

i) A is a Boolean subalgebra of B

ii) for all $a \in A$ and $b \in B : a/b \in A$ and $b \backslash a \in A$.

That is, A is closed under the operations / and \ with denominators taken in B. Note that a quotienting subalgebra of $\langle B, \cdot, /, \backslash \rangle$ does not have to be closed under the product operation.

In Chapter 7 we have seen that a subalgebra A of a Boolean algebra B gives rise to a Stone equivalence relation θ_A on X_B. Furthermore we have seen that the additional operations on B give rise to a ternary relation R on X_B. It will turn out that the Stone equivalence relations coming from quotienting subalgebras of B in some sense 'respect' the relation R. What this means is made precise by the following definition.

Definition 8.11 Let X be a Stone space and θ a Stone equivalence relation on X. Furthermore let R be a ternary relation on X. We say that θ is an **R-congruence** on X provided for all $x, x', y, y', z \in X$

$$R(x, y, z) \text{ and } (x, x'), (y, y') \in \theta \Rightarrow \exists z' : R(x', y', z') \text{ and } (z, z') \in \theta.$$

Observe that this definition is in fact a generalization of the concept of congruence with respect to an operation as defined in Definition 2.13. That is, Definition 2.13 deals with the case that R is functional.

Definition 8.12 Let R be a ternary relation on a set X. We say that R is **functional** if for all $x, y \in X$ there exists exactly one $z \in X$ such that $R(x, y, z)$.

Every ternary relation that is functional defines a binary operation on X and an R-congruence in this case is the same as a congruence with respect to this operation. Using the notion of R-congruence, we can formulate the main theorem of this chapter.

Theorem 8.13 *Let $\langle B, \cdot, /, \backslash \rangle$ be a residuated Boolean algebra and A a Boolean subalgebra of B. Furthermore let R be the relation dual to the additional operations on B and θ_A the Stone equivalence on X_B induced by A. Then A is a quotienting subalgebra of B if and only if θ_A is an R-congruence.*

For the proof of this theorem we need the following lemma about prime filters.

Lemma 8.14 *Let $\langle B, \cdot, /, \backslash \rangle$ be a residuated Boolean algebra and $x \in X_B$. We have for all $a, b \in B$*

$$a/b \notin x \Longleftrightarrow \exists y, z \in X_B : a \notin z, b \in y \text{ and } R(x, y, z).$$

Proof. $\Rightarrow$) Let $a, b \in B$ and $x \in X_B$ such that $a/b \notin x$. Define

$$I = \{d \in B \mid \exists c \in x \text{ with } c \cdot d \leq a\}.$$

Certainly I is a down-set. Furthermore let $d_1, d_2 \in a$. That is, there exist $c_1, c_2 \in x$ such that $c_1 \cdot d_1 \leq a$ and $c_2 \cdot d_2 \leq a$. Define $c = c_1 \wedge c_2$. Then $c \in x$ and $c \cdot d_1 \leq a$ and $c \cdot d_2 \leq a$, hence $c \cdot (d_1 \vee d_2) = c \cdot d_1 \vee c \cdot d_2 \leq a$. That is $d_1 \vee d_2 \in I$. Hence I is an ideal.
Now suppose $b \in I$. Then there is a $c \in x$ such that $c \cdot b \leq a$. By residuation

$$c \cdot b \leq a \Leftrightarrow c \leq a/b.$$

But then $a/b \in x$, as $c \in x$. This is a contradiction. So $b \notin I$.
Hence defining $F = \uparrow b$ gives rise to a disjoint filter-ideal pair (F, I). We apply the Prime Filter Theorem and obtain a filter y such that $b \in y$ and $y \cap I = \emptyset$.
To obtain z we will again use the Prime Filter Theorem. We define

$$F' = \{e \in B \mid \exists c \in x, c' \in y \text{ with } c \cdot c' \leq e\}.$$

Certainly F' is an up-set. Now suppose $e_1, e_2 \in F'$. Then there are $c_1, c_2 \in x$ and $c_1', c_2' \in y$ such that $c_1 \cdot c_1' \leq e_1$ and $c_2 \cdot c_2' \leq e_2$. Thus $c_1 \wedge c_2 \in x$ and $c_1' \wedge c_2' \in y$ and

$$(c_1 \wedge c_2) \cdot (c_1' \wedge c_2') \leq c_1 \cdot c_1' \wedge c_2 \cdot c_2' \leq e_1 \wedge e_2.$$

Hence $e_1 \wedge e_2 \in F'$. Thus F' is a filter. By the definition of I and the fact that $y \cap I = \emptyset$ we have that $a \notin F'$. Hence defining $I' = {\downarrow} a$ gives rise to a disjoint filter-ideal pair (F', I'). We apply the Prime Filter Theorem and obtain a prime filter z such that $F' \subseteq z$ and $a \notin z$. But $F \subseteq z$ implies $c \cdot c' \in z$ for all $c \in x, c' \in y$. That is, $R(x, y, z)$.
$\Leftarrow$) The other direction follows directly from Proposition 8.9.

□

Apart from being a technical useful Lemma, the above result actually tells us that we can recover the operations / and \ directly from R. This is useful for quotienting subalgebras not closed under the product operation.

Proof of Theorem 8.13. $\Rightarrow$) Suppose that A is a quotienting subalgebra of B. We have to show that

$$\theta_A = \{(x, x') \in X_B^2 \mid x \cap A = x' \cap A\}$$

is an $R-$congruence. That is, given $x, x', y, y', z \in X_B$ such that $(x, x'), (y, y') \in \theta_A$ and $R(x, y, z)$, we want to find $z' \in X_B$ such that $(z, z') \in \theta_A$ and $R(x', y', z')$. We define

$$F = {\uparrow}(x' \cdot y') = \{d \in B \mid \exists b \in x', c \in y' \text{ with } b \cdot c \leq d\}.$$

Then F is a prime filter (as the definition of F is similar to the definition of F' in the proof of Lemma 8.14). Furthermore we define

$$I = {\downarrow}(A \cap z^c).$$

Certainly I is a down-set. Now let $c_1, c_2 \in I$. Then you can find $a_1, a_2 \in A \cap (z)^c$ such that $c_i \leq a_i$. That means $c_1 \vee c_2 \leq a_1 \vee a_2$ and the latter is in $A \cap z^c$ as A is a subalgebra and z^c is a (prime) ideal. Hence I is an ideal.

Now we will prove that F and I are disjoint by showing that $a \in F$ and $a \in A$ implies $a \in z$. Assume $a \in A$ and $a \in F$. The latter means that we can find $b \in x'$ and $c \in y'$ such that $b \cdot c \leq a$. By residuation this implies $c \leq b \backslash a$. Because $c \in y'$, we have $b \backslash a \in y'$. In addition $b \backslash a \in A$ as $a \in A$ and A is a quotienting subalgebra of B. Hence

$$b \backslash a \in y' \cap A = y \cap A.$$

In particular $b \backslash a \in y$.
Furthermore observe that $b \cdot b \backslash a \leq a$ implies $b \leq a/(b \backslash a)$. As $b \in x'$ this implies $a/(b \backslash a) \in x'$. Also $a/(b \backslash a) \in A$. Hence

$$a/(b \backslash a) \in x' \cap A = x \cap A.$$

In particular $a/(b\backslash a) \in x$.
So $a/(b\backslash a) \cdot b\backslash a \in z$. As $a/(b\backslash a) \cdot b\backslash a \leq a$ (by residuation) this implies $a \in z$. Hence (F, I) is a disjoint filter-ideal pair. Now we apply the Prime Filter Theorem and obtain a prime filter z' that contains F. By definition of F this means $R(x', y', z')$. Furthermore, $z' \cap (A \cap (z)^c) = \emptyset$, so $z' \cap A \subseteq z \cap A$. But as $z \cap A$ and $z' \cap A$ are both prime filters of A this implies $z' \cap A = z \cap A$. That is $(z, z') \in \theta_A$.

$\Leftarrow$) Suppose θ_A is an R-congruence. By Theorem 7.23 $A_{\theta_A} = A$. Hence it suffices to show that A_{θ_A} is a quotienting subalgebra of B. Recall that

$$\begin{aligned} A_{\theta_A} &= \{b \in B \mid \forall (x, x') \in \theta_A : b \in x \Leftrightarrow b \in x'\} \\ &= \{b \in B \mid \forall (x, x') \in \theta_A : b \notin x \Leftrightarrow b \notin x'\} \end{aligned}$$

Let $a \in A_{\theta_A}$ and $b \in B$. We will show that $a/b \in A_{\theta_A}$. So let $(x, x') \in \theta_A$. We have

$$\begin{aligned} a/b \notin x &\Leftrightarrow \exists y, z \in X_B \text{ with } a \notin z, b \in y \text{ and } R(x, y, z)) \\ &\Leftrightarrow \exists y, z, z' \in X_B \text{ with } a \notin z, b \in y \text{ and } R(x', y, z') \text{ and } (z, z') \in \theta_A \\ &\Leftrightarrow \exists y, z' \in X_B \text{ with } a \notin z', b \in y \text{ and } R(x', y, z') \\ &\Leftrightarrow a/b \notin x'. \end{aligned}$$

The first and the last equivalence are established by Lemma 8.14. The second and third equivalence are established by taking $y' = y$ and the fact that θ_A is an R-congruence. Also $(z, z') \in \theta_A$ implies $a' \notin z' \Leftrightarrow a' \notin z$ for all $a' \in A$, in particular for our given a.
Hence $a/b \in A_{\theta_A}$. In a similar way you can prove that $b\backslash a \in A_{\theta_A}$ for all $a \in A, b \in B$.

□

A pertinent result in algebra is the fact that a congruence with respect to an operation gives rise to an operation on the quotient algebra. For example if $\langle X, \cdot \rangle$ is a semigroup and θ a congruence with respect to $\cdot$ this defines an operation $\cdot$ on X/θ is the following way

$$x/\theta \cdot y/\theta = (x \cdot y)/\theta.$$

That is, we have

$$x/\theta \cdot y/\theta = z/\theta \Leftrightarrow (x \cdot y, z) \in \theta.$$

or

$$x/\theta \cdot y/\theta = z/\theta \Leftrightarrow \exists z' \in X : x \cdot y = z' \text{ and } (z, z') \in \theta.$$

We have formulated it in this last form to draw a parallel with the observation that an R-congruence θ_A defines a relation R_A on the quotient space X_B/θ_A.

Definition 8.15 Let B be a residuated Boolean algebra and R the relation on X_B dual to the additional operations. We define for all $x, y, z \in X_B$

$$R_A(\overline{x}, \overline{y}, \overline{z}) \Leftrightarrow \exists z' \in X_B : R(x, y, z') \text{ and } (z, z') \in \theta_A.$$

where $\overline{x}, \overline{y}, \overline{z}$ are the equivalence classes of x, y, z with respect to θ_A.

By Proposition 7.17 we have that $X_B/\theta_A \cong X_A$. Hence the relation R defines a relation R_A on X_A for every quotienting subalgebra A of B.

The equivalent definitions of R obtained in Proposition 8.9 also make it possible to define the relation R on subalgebras closed under one of the three additional operations. It can be proved that for quotienting subalgebras these two notions agree. That is, it can be shown that

$$\begin{aligned} R_A(x, y, z) &\Leftrightarrow \forall a, b \in A : b \in y, a \notin z \Rightarrow a/b \notin x \\ &\Leftrightarrow \forall a, b \in A : b \in x, a \notin z \Rightarrow b\backslash a \notin y. \end{aligned}$$

As remarked above, Lemma 8.14 implies that R_A captures / and \ on the dual space and thus one can develop a dual correspondence for the operations / and \ without $\cdot$.

Before we apply the theory developed in this chapter to the residuated Boolean algebra of all languages we consider the special case of the regular languages in the next chapter. As mentioned before this will a formulation of the relationship between regular languages and semigroups in terms of duality.

Chapter 9

The dual space of the regular languages

In this chapter we will apply the duality theory developed in the previous chapter to the Boolean algebra of regular languages. The main result of this chapter is the fact that the relation R dual to the additional operations is functional on the dual space of this algebra. Furthermore we prove that the dual space of the quotienting subalgebra generated by a regular language L is (isomorphic to) the syntactic semigroup of L. This observation relates the duality approach to the semigroup setting and places the previously existing results about semigroups and regular languages in a more general setting. Also it opens the door to generalize the theory and study classes outside the regular languages as we will do in the next chapter.

9.1 The residuated Boolean algebra of regular languages

As $\langle \mathcal{P}(A^*), \cdot, /, \backslash \rangle$ is an example of a residuated Boolean algebra, we can apply the theory developed so far to this setting. But before we discuss the general case we will have a closer look at the Boolean algebra $\mathrm{Reg}(A^*)$ of all regular languages. First we observe that this is also a residuated Boolean algebra, as it is a subalgebra of $\mathcal{P}(A^*)$ closed under the additional operations.

Proposition 9.1 *The class of regular languages is closed under the operations $\cdot, /$ and $\backslash$.*

Proof. Let K, L be regular languages. By definition $K \cdot L$ is regular. Also L/u and $u\backslash L$ are regular for all $u \in A^*$, since automata for L/u and $u\backslash L$ can easily be constructed from an automaton for L as we have seen in Section 8.1. By Proposition 8.2 this implies that the languages L/K and $K\backslash L$ are regular for all $K \in \mathcal{P}(A^*)$. So in particular L/K and $K\backslash L$ are regular for $K \in \text{Reg}(A^*)$.

□

So we can consider $\langle \text{Reg}(A^*), \cdot, /, \backslash \rangle$, which is a residuated Boolean algebra. The theory developed so far gives rise to a relation R on the dual space $X_{\text{Reg}(A^*)}$ of this algebra. In the case of the regular languages the relation R dual to the additional operations turns out to be functional.

9.2 The dual space of the regular languages

To prove that R is functional on $X_{\text{Reg}(A^*)}$ we need the following lemma.

Lemma 9.2 *For $x, y \in X_{Reg(A^*)}$ define*

$$z = \{L \in Reg(A^*) \mid \exists K \in x \text{ and } H \in y \text{ such that } K \cdot H \subseteq L\}.$$

Then z is a prime filter of $Reg(A^*)$.

Proof. It is easy to check that $A^* \in z$ and $\emptyset \notin z$. Furthermore z is an up-set.
Now let $L_1, L_2 \in z$. That is, there are $K_1, K_2 \in x$ and $H_1, H_2 \in y$ such that $K_1H_1 \subseteq L_1$ and $K_2H_2 \subseteq L_2$. Hence $K_1 \cap K_2 \in x$ and $L_1 \cap L_2 \in y$ and

$$(K_1 \cap K_2)(H_1 \cap H_2) \subseteq K_1H_1 \cap K_2H_2 \subseteq L_1 \cap L_2.$$

So $L_1 \cap L_2 \in z$.
Now suppose $L = L_1 \cup L_2 \in z$. We have to show $L_1 \in z$ or $L_2 \in z$. Let $\sim_{L_1}$ and $\sim_{L_2}$ be the syntactic congruences of L_1 and L_2. As L_1 and L_2 are regular languages both $\sim_{L_1}$ and $\sim_{L_2}$ are of finite index. Define

$$\sim := \sim_{L_1} \cap \sim_{L_2}.$$

Then also $\sim$ is of finite index and it is easily checked that $\sim$ saturates L. The fact that $L \in z$ tells us that we can find $K \in x$ and $H \in y$ such that $KH \subseteq L$. As $\sim$ is of finite index, there exists a finite $K' \subseteq K$ such that

$$K \subseteq \overline{K} = \bigcup_{u \in K'} \overline{u}.$$

where $\overline{u}$ denotes the equivalence class of u with respect to $\sim$. Hence $\overline{u} \in x$ for some $u \in K'$. Furthermore $KH \subseteq L$ implies $H \subseteq K \setminus L$. This implies

$$\begin{array}{rcl} H & \subseteq & K \setminus L \\ & = & \overline{K} \setminus L \\ & \subseteq & \overline{u} \setminus L \\ & = & u \setminus L \\ & = & u \setminus (L_1 \cup L_2) \\ & = & u \setminus L_1 \cup u \setminus L_2. \end{array}$$

Hence $u \setminus L_1 \in y$ or $u \setminus L_2 \in y$. Without loss of generality we may assume $u \setminus L_1 \in y$. We have $u \setminus L_1 = \overline{u} \setminus L_1$ and $\overline{u} \cdot (\overline{u} \setminus L_1) \subseteq L_1$. Hence $L_1 \in z$.

□

This lemma tells us that we can endow the space $X_{\mathrm{Reg}(A^*)}$ with an operation $* : X_{\mathrm{Reg}(A^*)} \times X_{\mathrm{Reg}(A^*)} \to X_{\mathrm{Reg}(A^*)}$ defined by

$$x * y := \{L \in \mathrm{Reg}(A^*) \mid \exists K \in x, H \in y \text{ such that } K \cdot H \subseteq L\}.$$

The following theorem points out that the relation R on $X_{\mathrm{Reg}(A^*)}$ dual to the additional operations is functional and gives rise to the operation $*$.

Theorem 9.3 *Let $\langle Reg(A^*), \cdot, /, \backslash \rangle$ be the residuated Boolean algebra of regular languages and R the relation on $X_{Reg(A^*)}$ dual to the additional operations. Then R is functional. In particular for all $x, y, z \in X_{Reg(A^*)}$*

$$R(x, y, z) \iff z = x * y.$$

Proof. Let $x, y \in X_{\mathrm{Reg}(A^*)}$. Obviously $L \in x$ and $M \in y$ implies $L \cdot M \in x * y$. Hence $R(x, y, x * y)$. Now suppose $z \in X_{\mathrm{Reg}(A^*)}$ and $R(x, y, z)$. By definition of R and the fact that z is a prime filters (and hence, in particular, an up-set), this implies $x * y \subseteq z$. By Lemma 9.2 we have that also $x * y$ is a prime filter. Thus $x * y = z$.

□

From the definition of the operation $*$ it is easily deduced that this operation is associative. Hence it turns $X_{\mathrm{Reg}(A^*)}$ into a semigroup. If we combine this result with Theorem 8.13 which says that quotienting subalgebras give rise to R-congruences, we get the following corollaries.

Corollary 9.4 *Let $\mathcal{C}$ be a quotienting subalgebra of Reg(A^*). Then $\theta_{\mathcal{C}}$ is a congruence on the semigroup $\langle X_{Reg(A^*)}, *\rangle$.*

Corollary 9.5 *Let $\mathcal{C}$ be a quotienting subalgebra of Reg(A^*). Then $X_{\mathcal{C}}$ is a semigroup, with the operation inherited from $X_{Reg(A^*)}$.*

9.3 Dual space versus syntactic semigroup

We will now see that the application of duality theory to the residuated Boolean algebra of regular languages gives rise to (an isomorphic copy) of the syntactic semigroup of a regular language. Recall that for every language $L \in \mathcal{P}(A^*)$ we defined the quotienting subalgebra generated by L as follows.

Definition 9.6 For any $L \in \mathcal{P}(A^*)$ we define $\mathcal{Q}_L$ to be the smallest quotienting subalgebra of $\mathcal{P}(A^*)$ such that $L \in \mathcal{Q}_L$.

The last corollary of the previous section shows that the dual space $X_{\mathcal{C}}$ of any quotienting subalgebra of Reg(A^*) is a semigroup. In particular, every regular language L gives rise to a semigroup $\langle X_{\mathcal{Q}_L}, *\rangle$, that is the dual space of the quotienting algebra $\mathcal{Q}_L$ generated by L. At the end of this section we will show that this semigroup is in fact isomorphic to the syntactic semigroup $A^*/{\sim_L}$. First we prove that the syntactic congruence determines exactly the languages that belong to the quotienting algebra $\mathcal{Q}_L$.

Definition 9.7 Let $L \in \mathcal{P}(A^*)$ be a language and $\sim$ a congruence on A^*. We say that $\sim$ **recognizes** L if $\sim$ saturates L. That is, if L is a union of equivalence classes with respect to $\sim$.

It is not hard to prove that for every formal language L, the syntactic congruence $\sim_L$ recognizes L.

Lemma 9.8 *Let $L \in \mathcal{P}(A^*)$ be a languages and $\sim_L$ be the syntactic congruence of L. Then $\sim_L$ recognizes L.*

Proof. Suppose $u \sim_L v$. By definition this implies for all $s, t \in A^*$

$$sut \in L \Leftrightarrow svt \in L.$$

In particular, $s = \lambda$ and $t = \lambda$ implies

$$u \in L \Leftrightarrow v \in L.$$

That is, $\sim_L$ saturates L.

□

In addition we can prove that the languages recognized by $\sim_L$ are exactly those in $\mathcal{Q}_L$.

Proposition 9.9 *Let L be a regular language. We have for all $L' \in Reg(A^*)$*

$$L' \in \mathcal{Q}_L \Longleftrightarrow L' \text{ can be recognized by } \sim_L .$$

The proof of this proposition uses the following lemma.

Lemma 9.10 *Let $L \subseteq A^*$ and $\sim$ a congruence on A^* that saturates L, that is, for which L is a union of equivalence classes. Then $\sim$ saturates $K \setminus L$ and L/K for all $K \in \mathcal{P}(A^*)$.*

Proof. Let $v, v' \in A^*$ with $v \sim v'$. As $\sim$ is a congruence, this implies

$$uv \sim uv' \quad \forall u \in A^*.$$

In particular

$$kv \sim kv' \quad \forall k \in K.$$

Now let $L \in \mathcal{P}(A^*)$ be a language that is saturated by $\sim$ we have

$$\begin{aligned} v \in K \setminus L \quad &\Leftrightarrow \quad k \cdot v \in L \quad \forall k \in K \\ &\Leftrightarrow \quad kv' \in L \quad \forall k \in K \\ &\Leftrightarrow \quad v' \in K \setminus L. \end{aligned}$$

So $K \setminus L$ is saturated by $\sim$. In the same way we can prove that $\sim$ saturates L/K.

□

Proof of Proposition 9.9. First observe that a language is recognized by $A^*/{\sim_L}$ if and only if it is saturated by $\sim_L$. By Lemma 9.8 L is saturated by $\sim_L$. Now suppose L_1 and L_2 are saturated by $\sim_L$. Then obviously $(L_1)^c, L_1 \cap L_2$ and $L_1 \cup L_2$ are saturated by $\sim_L$. Furthermore by Lemma 9.10 L_1/K and $K \backslash L_1$ are saturated for all $K \in \mathcal{P}(A^*)$. Hence every language in $\mathcal{Q}_L$ is saturated by $\sim_L$. Note that this does not rely on the fact that L is regular and this is true for all $L \in \mathcal{P}(A^*)$
What is left to show is that every language that is saturated by $\sim_L$ is in $\mathcal{Q}_L$. Let $u \in A^*$ and $\overline{u}$ the equivalence class of u with respect to $\sim_L$. We want to show that $\overline{u}$ is in $\mathcal{Q}_L$. Since L is saturated by $\sim_L$ we have

$$\overline{u} \subseteq L \text{ or } \overline{u} \subseteq L^c.$$

Both L and L^c are in $\mathcal{Q}_L$, so without loss of generality we may assume $\overline{u} \in L$. If $\overline{u} = L$, then $\overline{u} \in \mathcal{Q}_L$ and we are done. So suppose there is a $v \in A^*$ such that $v \in L$ and $v \notin \overline{u}$. Then not $u \sim_L v$. That is, there exist $s, t \in A^*$ such that either

$$sut \in L \text{ and } svt \notin L$$

or

$$sut \notin L \text{ and } svt \in L.$$

Without loss of generality we may assume that the first is the case. This implies

$$u \in s\backslash L/u \text{ and } v \notin s\backslash L/u.$$

Now define

$$L_1 = L \cap s\backslash L/u.$$

Now observe that $L_1 \in \mathcal{Q}_L$ and $\overline{u} \subseteq L_1$. Furthermore L_1 is strictly smaller than L as $v \in L$ and $v \notin L_1$. Now using the fact that L is regular, we not that $A^*/{\sim_L}$ is finite. Thus we can repeat this procedure finitely many times to obtain a language L_n such that $\overline{u} = L_n$ and $L_n \in \mathcal{Q}_L$. Hence $\overline{u} \in \mathcal{Q}_L$ for every $u \in A^*$. Certainly this implies that any union of equivalence classes is also in $\mathcal{Q}_L$ as $\mathcal{Q}_L$ is closed under union. Hence all the languages that are recognized by $A^*/{\sim_L}$ are in $\mathcal{Q}_L$.

□

In particular this proposition 9.9 implies that $\mathcal{Q}_L$ is finite for every regular language L, as $\sim_L$ is of finite index. Hence $X_{\mathcal{Q}_L}$ is a finite semigroup

Lemma 9.11 *The elements of $X_{\mathcal{Q}_L}$ are all of the form*

$$x_{\overline{u}} = \{K \in \mathcal{Q}_L \mid \overline{u} \subseteq K\}$$

where $\overline{u}$ denotes the equivalence class of u with respect to $\sim_L$. Furthermore the semigroup operation on $X_{\mathcal{Q}_L}$ inherited from $\langle X_{Reg(A^*)}, *\rangle$ is defined by

$$x_{\overline{u}} * x_{\overline{v}} = x_{\overline{uv}}.$$

Proof. First of all observe that Proposition 9.9 implies that

$$\mathcal{Q}_L \cong \mathcal{P}(A^*/{\sim_L})$$

and that $\mathcal{Q}_L$ is finite. This shows that every prime filter of $\mathcal{Q}_L$ is principal. That is, that it is of the form $x_{\overline{u}}$ for some $u \in A^*$.

For the second assertion, denote by x_u the principal filter of $\mathrm{Reg}(A^*)$ generated by u. That is,

$$x_u = \{K \in \mathrm{Reg}(A^*) \mid u \in K\}.$$

It is not hard to see that $x_u * x_v = x_{uv}$. By Theorem 7.17 we have that $X_{\mathrm{Reg}(A^*)}/\theta_{\mathcal{Q}_L} \cong X_{\mathcal{Q}_L}$ and an isomorphism is given by

$$f(\overline{x}) = x \cap \mathcal{Q}_L \text{ for all } x \in X_{\mathrm{Reg}(A^*)}$$

where $\overline{x}$ denotes the equivalence class of x with respect to $\theta_{\mathcal{Q}_L}$. This implies $f(\overline{x_u}) = x_u \cap \mathcal{Q}_L = x_{\overline{u}}$ for all $u \in A^*$. Hence we have

$$\begin{aligned} x_{\overline{u}} * x_{\overline{v}} &= f(\overline{x_u}) * f(\overline{x_v}) \\ &= f(\overline{x_u} * \overline{x_v}) \\ &= f(\overline{x_u * x_v}) \\ &= f(\overline{x_{uv}}) \\ &= x_{\overline{uv}}. \end{aligned}$$

This establishes the second assertion.

□

We are now ready to prove the main result of this section.

Proposition 9.12 *Let L be a regular language. Then*

$$\langle A^*/{\sim_L}, \cdot\rangle \textit{ is isomorphic to } \langle X_{\mathcal{Q}_L}, *\rangle$$

Proof. Define a map $g : A^*/{\sim_L} \to X_{\mathcal{Q}_L}$ by

$$g : \overline{u} \mapsto x_{\overline{u}}$$

where $\overline{u}$ denotes the equivalence class of u with respect to $\sim_L$ for every $u \in A^*$. As a result of the previous lemma this map is onto. Also $x_{\overline{u}} = x_{\overline{v}}$ implies $\overline{u} = \overline{v}$, thus g is injective. Furthermore for all $u, v \in A^*$ we have

$$\begin{aligned} g(\overline{u}) * g(\overline{v}) &= x_{\overline{u}} * x_{\overline{v}} \\ &= x_{\overline{uv}} \\ &= g(\overline{uv}) \\ &= g(\overline{u} \cdot \overline{v}). \end{aligned}$$

Hence g defines a semigroup isomorphism between the semigroups $A^*/{\sim_L}$ and $X_{\mathcal{Q}_L}$.

□

The results presented in this chapter show that in the case of the regular languages the application of duality theory is closely related to the known theory about finite semigroups. These results can also be found in [9] and [10]. The duality approach has as a big advantage above the semigroup approach, that it can be generalized for studying classes outside the regular languages. In the next chapter we will show how this can be established.

9.4 An example outside the regular languages

We will end this chapter by showing that the relation R dual to the additional operations, in general, does not have to be functional. We will prove this by giving an example of language L for which the relation R is not functional on the dual space of $\mathcal{Q}_L$, the quotienting subalgebra generated by L. In particular, this shows that the algebra $\mathcal{Q}_L$ is not inside the regular languages, and thus that L is not a regular language.

Let $A = \{a, b\}$ and define for $u \in A^*$

$$\begin{aligned} |u|_a &:= \text{ number of } a\text{'s occuring in } u \\ |u|_b &:= \text{ number of } b\text{'s occuring in } u. \end{aligned}$$

Let $L \in \mathcal{P}(A^*)$ be the language defined by

$$L := \{u \in A^* \mid |u|_a - |u|_b \geq 0\}.$$

We denote by $\mathcal{Q}_L$ the quotienting subalgebra generated by L.

Proposition 9.13 *$\mathcal{Q}_L$ is isomorphic to the quotienting subalgebra of $\mathcal{P}(\mathbb{Z})$ generated by $\uparrow 0$.*

Proof. Consider the semigroup $(\mathbb{Z}, +)$ and the semigroup morphism $h : A^* \to \mathbb{Z}$ defined by

$$h : u \mapsto |u|_a - |u|_b.$$

Then $L = h^{-1}(\uparrow 0)$. As h is onto, we have that $h^{-1} : \mathcal{P}(\mathbb{Z}) \to \mathcal{P}(A^*)$ is a Boolean algebra embedding. Furthermore, by Lemma 9.10, h^{-1} preserves the operations $/$ and $\backslash$, hence the image of the quotienting subalgebra of $\mathcal{P}(\mathbb{Z})$ generated by $\uparrow 0$ under h^{-1} is exactly $\mathcal{Q}_L$ and furthermore, h^{-1} defines an isomorphism between the two.

□

Now let Q be the quotienting subalgebra of $\mathcal{P}(\mathbb{Z})$ generated by $\uparrow 0$. We have

$$(\uparrow 0)/T = \{k \in \mathbb{Z} \mid k + t \geq 0 \text{ for all } t \in T\}.$$

This implies

$$(\uparrow 0)/T = \begin{cases} \emptyset & \text{if } T \text{ has no minimum} \\ \uparrow(-\min(T)) & \text{otherwise.} \end{cases}$$

Hence $\uparrow k \in Q$ for all $k \in \mathbb{Z}$. Observe that $(\uparrow 0)/T = T \setminus (\uparrow 0)$, since $+$ is commutative. Closing under the Boolean algebra operation gives rise to the following characterization of Q.

$$\begin{aligned} Q \;\; = \;\; & \{S \subseteq \mathbb{Z} \mid S \text{ is finite}\} \\ & \cup \{S \subseteq \mathbb{Z} \mid \exists k \in \mathbb{Z}, \exists \text{ finite } S' \subseteq \mathbb{Z} : S = \uparrow k \cup S'\} \\ & \cup \{S \subseteq \mathbb{Z} \mid \exists k \in \mathbb{Z}, \exists \text{ finite } S' \subseteq \mathbb{Z} : S = \downarrow k \cup S'\} \\ & \cup \{S \subseteq \mathbb{Z} \mid \exists k, k' \in \mathbb{Z}, \exists \text{ finite } S' \subseteq \mathbb{Z} : S = \uparrow k \cup \downarrow k' \cup S'\} \end{aligned}$$

We will now consider the dual space X_Q of Q, consisting of all the prime filters of Q.

Proposition 9.14 *The prime filters of Q are exactly*

i) $x_k = \uparrow\{k\} = \{S \in Q \mid k \in S\}$ *for all* $k \in \mathbb{Z}$

ii) $x_\infty = \{S \in Q \mid \exists k : \uparrow k \subseteq S\}$

iii) $x_{-\infty} = \{S \in Q \mid \exists k : \downarrow k \subseteq S\}$.

Proof. It is easy to check that all three definitions above yield prime filters of Q. Now let x be a prime filter of Q. If x contains a finite subset of $\mathbb{Z}$, then obviously $x = \uparrow\{k\}$ for some $k \in \mathbb{Z}$. Now suppose that x does not contain any finite subset of $\mathbb{Z}$. As $\uparrow k \cup \downarrow k = \mathbb{Z}$ for all $k \in \mathbb{Z}$ we have that $\uparrow k \in x$ or $\downarrow k \in x$ for all $k \in \mathbb{Z}$. Suppose there exist $k, k' \in \mathbb{Z}$ such that $\uparrow k \in x$ and $\downarrow k' \in x$, then $\uparrow k \cap \downarrow k' \in x$. This is a contradiction as $\uparrow k \cap \downarrow k'$ is finite. Hence

$$x = \{S \in Q \mid \exists k : \uparrow k \subseteq S\} = x_\infty$$

or

$$x = \{S \in Q \mid \exists k : \downarrow k \subseteq S\} = x_{-\infty}.$$

□

Now we can show that R is not functional on X_Q. That is, we will show $R(x_{-\infty}, x_\infty, x)$ for all $x \in X_Q$. Recall that by definition of R

$$R(x_{-\infty}, x_\infty, x) \Longleftrightarrow \forall S, T \in \mathcal{P}(\mathbb{Z}) : [S \in x_{-\infty}, T \in x_\infty] \Rightarrow S + T \in x.$$

Observe that $S \in x_{-\infty}$ implies ${\downarrow}k \subseteq S$ for some $k \in \mathbb{Z}$. Similarly, $T \in x_\infty$ implies ${\uparrow}m$ for some $m \in \mathbb{Z}$. As ${\downarrow}k + {\uparrow}m = \mathbb{Z}$ for all $k, m \in \mathbb{Z}$, we have $S + T = \mathbb{Z}$ for all $S \in x_{-\infty}, T \in x_\infty$. Hence

$$R(x_{-\infty}, x_\infty, x) \Longleftrightarrow \mathbb{Z} \in x.$$

This is the case for all $x \in X_Q$.

In particular, this example shows that in general the dual space $\langle X_{\mathcal{Q}_L}, R\rangle$ of an quotienting subalgebra of a language does not have to be a semigroup. The dual space $\langle X_{\mathcal{Q}_L}, R\rangle$ of $\mathcal{Q}_L$ can be seen as a generalization of the notion of syntactic semigroup outside the regular languages. We will come back to this observation in the next chapter.

Chapter 10

Syntactic Stone congruences and frames

In this final chapter we will apply the duality results presented in the previous chapters to the theory of formal languages. We introduce the notions of syntactic Stone congruence and syntactic Stone frame that generalize the notions of syntactic congruence and syntactic semigroup in the regular case. In the last section we consider the example of the class of commutative languages and illustrate that the syntactic Stone frame of an arbitrary language tells whether or not it is commutative. More specifically, we show that a language is commutative if and only if it syntactic Stone frame is commutative.

10.1 Syntactic congruences and semigroups for non-regular languages

Before we introduce the notion of syntactic Stone congruence and syntactic Stone frame, we will shortly motivate why the syntactic congruence and semigroup in general do not suffice to describe arbitrary classes of languages.

In the previous chapter we have seen that for regular languages the syntactic semigroup of a language L recognizes precisely all the languages in $\mathcal{Q}_L$, the quotienting subalgebra of $\mathcal{P}(A^*)$ generated by L. For non-regular languages this can no longer hold as can be shown by an easy argument on the cardinalities of both sets. The set of languages in $\mathcal{Q}_L$ is the underlying set of a finitely generated algebra and hence countable. The set of all languages recognized by $A^*/{\sim_L}$ is of the same cardinality as $\mathcal{P}(A^*/{\sim_L})$ and hence not countable ($A^*/{\sim_L}$ is of infinite index, since L is not regular). So in the case of the non-regular languages the syntactic semigroup of a language recognizes too many languages.

There is another reason why congruences on A^* are not suitable to study classes of languages in general. Given a class of languages $\mathcal{C}$, there is a good chance that it is not possible to find a congruence $\sim$ on A^* such that $\sim$ recognizes exactly the languages in $\mathcal{C}$. For example, if a congruence on A^* recognizes all the regular languages then it can only contain the diagonal. But not every languages is regular. So this congruence recognizes to many languages.

10.2 Syntactic Stone congruences and frames

In the previous chapters we have developed a duality theory for Boolean algebras. That is, we have seen that Boolean algebras are in one-to-one correspondence to Stone spaces and that concepts in the theory of Boolean algebras have topological counterparts. In particular we observed that subalgebras of a Boolean algebra are in one-to-one correspondence to Stone equivalence relations on the dual space and that quotienting subalgebras are in one-to-one correspondence to R-congruences on the dual space. We will now see how we can use this relationship to obtain results in the theory of formal languages.

In this section we will introduce the notions of *syntactic Stone congruence* and *syntactic Stone frame*. In the regular case, the syntactic Stone frame is actually a semigroup and it is isomorphic to the syntactic semigroup introduced in Chapter 1. However, in the non-regular case these notions do in general not agree, as we will show.

We can use the one-to-one correspondence between quotienting subalgebras and Stone congruences established in Chapter 8 to define the notion of syntactic Stone congruence.

Definition 10.1 Let $\mathcal{C}$ be a class of languages such that $\mathcal{C}$ is a quotienting subalgebra of $\mathcal{P}(A^*)$. We call the relation $\theta_{\mathcal{C}}$ on $X_{\mathcal{P}(A^*)}$ as defined in Definition 7.15 the **syntactic Stone congruence** of $\mathcal{C}$.

We can now define what it means for language to be recognized by a Stone congruence.

Definition 10.2 Let θ be a Stone congruence on $X_{\mathcal{P}(A^*)}$ and $L \in \mathcal{P}(A^*)$ a language. We say that θ **recognizes** L if and only if $L \in A_\theta$. Where A_θ is the subalgebra of $\mathcal{P}(A^*)$ as defined in Definition 7.21.

The following proposition shows that for every quotienting subalgebra of $\mathcal{P}(A^*)$ there is a Stone congruence that recognizes exactly the languages in $\mathcal{C}$. Note that for syntactic congruences on A^* this was not the case.

Proposition 10.3 *Let $\mathcal{C}$ be a class of languages that is a quotienting subalgebra of $\mathcal{P}(A^*)$. Then the syntactic Stone congruence of $\mathcal{C}$ recognizes exactly all the languages in $\mathcal{C}$.*

Proof. Let $L \in \mathcal{C}$. Then L is recognized by $\theta_{\mathcal{C}}$ if and only if $L \in A_{\theta_{\mathcal{C}}}$. By Proposition 7.23 we have

$$A_{\theta_{\mathcal{C}}} = \mathcal{C}.$$

□

Using the syntactic Stone congruence of a class of languages, you can construct its syntactic Stone frame, as is shown in the following definition.

Definition 10.4 Let $\mathcal{C}$ be a quotienting subalgebra of $\mathcal{P}(A^*)$. The relational space $\langle X/\theta_{\mathcal{C}}, R_{\mathcal{C}} \rangle$, where $R_{\mathcal{C}}$ is the relation on $X/\theta_{\mathcal{C}}$ as defined in Definition 8.15, is called the **syntactic Stone frame of $\mathcal{C}$**.

In this terminology, Proposition 9.12 in the previous chapter says that for regular languages the syntactic Stone frame of $\mathcal{Q}_L$ is isomorphic to the syntactic semigroup of L. This motivates the following definition.

Definition 10.5 Let $L \in \mathcal{P}(A^*)$. The syntactic Stone frame of L is $\langle X_{\mathcal{Q}_L}, R_{\mathcal{Q}_L} \rangle$. That is, the syntactic Stone frame of $\mathcal{Q}_L$.

As mentioned before the syntactic Stone frame does not agree with the syntactic semigroup in the non-regular case. In the last section of this chapter we will consider the class of *commutative languages* and see how we can characterize these languages by their syntactic Stone frames.

Let us recall what kind of problems in formal language theory we are interested in from a computational point of view.

- Given $L \in \mathcal{P}(A^*)$ and $\mathcal{C} \subseteq \mathcal{P}(A^*)$. Does L belong to $\mathcal{C}$?
- Given $\mathcal{C}_1, \mathcal{C}_2 \subseteq \mathcal{P}(A^*)$. Is $\mathcal{C}_1$ a subset of $\mathcal{C}_2$?
- Given $L \in \mathcal{P}(A^*)$. What languages are related to L in terms of computation?

In Chapter 8 we have argued that quotienting subalgebras are particularly interesting from a computational point of view. Here we will assume our classes $\mathcal{C}$ are quotienting subalgebras $\mathcal{P}(A^*)$. Note that the first question can then easily be translated into a question of the second type, by considering the quotienting Boolean subalgebra of $\mathcal{P}(A^*)$ generated by L.

Applying the relationship between subalgebras and Stone equivalences on the dual space obtained in Theorem 7.18 to the second question we get the following result. It translates the second question above about containment of Boolean algebras into a question about the corresponding Stone equivalence relations.

Proposition 10.6 *Let $\mathcal{C}_1$ and $\mathcal{C}_2$ be Boolean subalgebras of $\mathcal{P}(A^*)$. Then*

$$\mathcal{C}_1 \subseteq \mathcal{C}_2 \iff \theta_{\mathcal{C}_2} \subseteq \theta_{\mathcal{C}_1}.$$

where $\theta_{\mathcal{C}_1}$ and $\theta_{\mathcal{C}_2}$ are the equivalence relations on $X_{\mathcal{P}(A^*)}$ as defined in Definition 7.15.

So far we have not used the fact that the classes $\mathcal{C}_1$ and $\mathcal{C}_2$ are quotienting subalgebras of $\mathcal{P}(A^*)$. In this case the relations $\theta_{\mathcal{C}_1}$ and $\theta_{\mathcal{C}_2}$ are not just Stone equivalence relations but also R-congruences. The benefit of this is that we may be able to find a fairly small subset of $E \subset \theta_{\mathcal{C}_2}$ that generates $\theta_{\mathcal{C}_2}$ as an R-congruence. It is then sufficient to check that $E \subseteq \theta_{\mathcal{C}_1}$, as also $\mathcal{C}_1$ is an R-congruence. We close with an illustration of this principle.

10.3 An example: the class of commutative languages

The relationship between regular languages and finite semigroups has given rise to numerous characterizations of classes of languages in terms of their syntactic semigroups. For example, in the introduction we have mentioned the characterization of the star-free languages by those languages whose syntactic semigroup is *aperiodic*. We would like the syntactic Stone frame to play a similar role in the setting of arbitrary languages to the role played by the syntactic semigroup in the regular case.

We will illustrate that this is at least the case for a relatively simple class of languages, called the *commutative* languages. From the definition of commutativity it is an easy observation that a language is commutative if and only if its syntactic semigroup is commutative (in fact this is where the name comes from). Although a characterization of commutative languages can be given purely in terms of semigroups, this class provides an instructive example, for which the answers are known and simple. This gives us a chance to test the definition of a syntactic Stone frame. We will generalize the notion of commutativity to frames and show that a language is commutative if and only if its syntactic frame is commutative.

Definition 10.7 A language $L \in \mathcal{P}(A^*)$ is called **commutative** if and only if for all $t, u, v, w \in A^*$

$$tuvw \in L \Longleftrightarrow tvuw \in L.$$

We denote the class of all commutative languages over the alphabet A by $\mathcal{C}(A^*)$ or simply by $\mathcal{C}$ when we are working over a fixed alphabet.

We can give a slightly different characterization of commutativity of languages by introducing the notion of a *permutation* of a word.

Definition 10.8 Let $u \in A^*$. A **permutation** of u is a word u' obtained by permutating the symbols of u. So let $u = a_1a_2 \dots a_n$, then $u' = a_{\alpha(1)}a_{\alpha(2)} \dots a_{\alpha(n)}$ is a permutation of u, for any permutation α of $\{1, 2, \dots, n\}$.

Now let A be a fixed alphabet. The concept of permutation defines an equivalence relation $\sim_{\mathcal{C}}$ on A^* in the following way:

$$\forall u, v \in A^* : u \sim_{\mathcal{C}} v \Longleftrightarrow u \text{ is a permutation of } v.$$

Note that $u_1 \sim_{\mathcal{C}} v_1$ and $u_2 \sim_{\mathcal{C}} v_2$ implies

$$u_1u_2 \sim_{\mathcal{C}} v_1v_2.$$

Hence $\sim_{\mathcal{C}}$ is a congruence of the semigroup $(A^*, \cdot)$.

For every $u \in A^*$ we define $\overline{u} \subseteq A^*$ to be the equivalence class of u under $\sim_{\mathcal{C}}$. That is,

$$\overline{u} = \{v \in A^* \mid v \text{ is a permutation of } u\}.$$

By the definition of a commutative language any permutation of a word is again in the language. This observation proves the following lemma that contains an equivalent characterization of the commutative languages.

Lemma 10.9 *A language L is commutative if and only if*

$$L = \bigcup_{u \in L} \overline{u}.$$

In other words, L is commutative if and only if L is saturated by $\sim_{\mathcal{C}}$.

Using this lemma we prove that the class of commutative languages is closed under complement, union, intersection and under the quotient operators $/$ and $\backslash$ with arbitrary denominator.

Proposition 10.10 *The class of commutative languages is a quotienting subalgebra of $\mathcal{P}(A^*)$.*

Proof. Let $L_1, L_2 \in \mathcal{C}$. Then L_1 and L_2 are saturated by $\sim_{\mathcal{C}}$. This implies $L_1 \cup L_2, L_1 \cap L_2$ and $(L_1)^c$ are saturated by $\sim_{\mathcal{C}}$ and hence commutative. Furthermore, by lemma 9.10 L_1/K and $K\backslash L_1$ are saturated by $\sim_{\mathcal{C}}$ for all $K \in \mathcal{P}(A^*)$. Hence $L_1/K, K\backslash L_1 \in \mathcal{C}$ for all $K \in \mathcal{P}(A^*)$.

□

Observe that from Lemma 10.9 it is easily deduced that a language is commutative if and only if its syntactic semigroup $A^*/\sim_L$ is a commutative semigroup. In fact, this is exactly where the definition of a commutative language originally comes from. However, as we have argued above, for non-regular languages L the syntactic semigroup of L is not as useful an invariant as the syntactic Stone frame of L. For this reason we want to be able to recognize that L is commutative by an 'equational' property of its syntactic Stone frame. The following equational property of Stone frames allows us to do just that.

Definition 10.11 Let $\langle X_{\mathcal{C}}, R_{\mathcal{C}} \rangle$ be the syntactic Stone frame of a quotienting subalgebra $\mathcal{C}$ of $\mathcal{P}(A^*)$. We call $\langle X_{\mathcal{C}}, R_{\mathcal{C}} \rangle$ **commutative** if

$$R_{\mathcal{C}}[x, y, \_] = R_{\mathcal{C}}[y, x, \_] \text{ for all } x, y \in X_{\mathcal{C}}.$$

We will now prove that the syntactic Stone frame of a language tells us whether or not the language is commutative in a similar way as the syntactic semigroup does in the regular case.

Theorem 10.12 *Let $L \in \mathcal{P}(A^*)$ be a language. We have*

$$L \in \mathcal{C} \Longleftrightarrow \langle X_{\mathcal{Q}_L}, R_{\mathcal{Q}_L} \rangle$$

is commutative.

For the proof of this theorem we need the following lemmas.

Lemma 10.13 *Let $L, K \in \mathcal{C}$. Then*

$$L/K = K \backslash L.$$

Proof. We have for all $u \in A^*$

$$\begin{aligned} u \in L/K &\Leftrightarrow uv \in L \text{ for all } v \in K \\ &\Leftrightarrow vu \in L \text{ for all } v \in K \\ &\Leftrightarrow u \in K \backslash L. \end{aligned}$$

Hence $L/K = K \backslash L$.

□

Lemma 10.14 *The relation R on $X_{\mathcal{P}(A^*)}$ dual to the additional operations, is functional on the set of principal prime filters of $\mathcal{P}(A^*)$. That is,*

$$R(x_u, x_v, z) \Longleftrightarrow z = x_{uv}.$$

where

$$x_u = \{L \in \mathcal{P}(A^*) \mid u \in L\}$$

for all $u \in A^*$.

Proof. Suppose $u, v \in A^*$ and $z \in \mathcal{P}(A^*)$ such that $R(x_u, x_v, z)$. By definition this means for all $L \in x_u$ and $K \in x_v$ we have $LK \in z$. In particular $\{u\} \cdot \{v\} = \{uv\} \in z$. As z is an up-set, this implies $x_{uv} \subseteq z$. The fact that x_{uv} and z are both prime filters establishes $z = x_{uv}$.

□

Proof of Theorem 10.12. $\Rightarrow$) Suppose $L \in \mathcal{C}$. This implies $\mathcal{Q}_L \subseteq \mathcal{C}$, as $\mathcal{C}$ is a quotienting algebra by Proposition 10.10. By the observation at the end of section 8.3 we have for all $x, y, z \in X_{\mathcal{Q}_L}$

$$\begin{aligned} R_{\mathcal{Q}_L}(x,y,z) &\Leftrightarrow \forall L, K \in A : K \in y, L \notin z \Rightarrow L/K \notin x \\ &\Leftrightarrow \forall L, K \in A : K \in y, L \notin z \Rightarrow K\backslash L \notin x \\ &\Leftrightarrow R_{\mathcal{Q}_L}(y,x,z). \end{aligned}$$

That is, $\langle X_{\mathcal{Q}_L}, R_{\mathcal{Q}_L}\rangle$ is commutative.

$\Leftarrow$) Now suppose $\langle X_{\mathcal{Q}_L}, R_{\mathcal{Q}_L}\rangle$ is commutative. Then by lemma 10.14 we have

$$(x_{uv}, x_{vu}) \in \theta_{\mathcal{Q}_L}$$

By Theorem 8.13 $\theta_{\mathcal{Q}_L}$ is an R-congruence. This implies

$$(x_{suvt}, x_{svut}) \in \theta_{\mathcal{Q}_L}$$

for all $s, t \in A^*$. But that means for all $L' \in \mathcal{Q}_L$

$$suvt \in L' \Longleftrightarrow svut \in L'.$$

So in particular

$$suvt \in L \Longleftrightarrow svut \in L.$$

That is, L is commutative.

□

10.4 Further research

The last theorem of the previous section shows that the information whether or not a language is commutative is given by its syntactic Stone frame. This is similar to the characterization of classes of languages by their syntactic semigroup in the regular case. Although the class of commutative languages is just one quite trivial example, it does show that it is possible to extend certain results beyond the setting of regular languages. In order to explore this theory fully more classes of languages would have to be studied. The next thing is to find an interpretation for the operations used to define the 'implicit' equations, introduced by Reiterman, to define pseudo-varieties of finite semigroups.

Bibliography

[1] M.E. Adams. The frattini sublattice of a distributive lattice. *Algebra Universalis*, 3:216–228, 1973.

[2] J. Almeida. *Finite Semigroups and Universal Algebra*. World Scientific, 1994.

[3] R. Balbes and P. Dwinger. *Distributive Lattices*. University of Missouri Press, 1974.

[4] G. Birkhoff. On the structure of abstract algebras. *Proc. Cambridge Phil. Soc.*, 31:433–454, 1935.

[5] S. Burris and H.P. Sankappanavar. *A Course in Universal Algebra*. Free online edition, 1981.

[6] D.M. Clark and B.A. Davey. *Natural Dualities for the Working Algebraist*. Cambridge University Press, 1998.

[7] B.A. Davey and H.A. Priestley. *Lattices and Order*. Cambridge University Press, 2002.

[8] S. Eilenberg. *Automata, languages, and machines. Vol. B.* Academic Press [Harcourt Brace Jovanovich Publishers], 1976.

[9] M. Gehrke, S. Grigorieff, and J.-E. Pin. Duality and equational theory of regular languages. *Lecture Notes in Computer Science*, 5125:246–257, 2008.

[10] M. Gehrke, S. Grigorieff, and J.-E. Pin. Monoids, recognizable languages and duality. Unpublished, February 2008.

[11] R. Goldblatt. Varieties of complex algebras. *Annals of Pure and Applied Logic*, 44:173–242, 1989.

[12] P. Halmos. *Lectures on Boolean Algebras.* Springer-Verlag, 1974.

[13] S.C. Kleene. Representation of events in nerve nets and finite automata. *Automata Studies*, 8:3–42, 1956.

[14] J. Martin. *Introduction to Languages and the Theory of Computation.* McGraw-Hill Higher Education, 2003.

[15] J.-E. Pin. Finite semigroups and recogizable languages: an introduction. Unpublished, May 2002.

[16] N. Pippenger. Regular languages and stone duality. *Theory of Computing Systems*, 30:121–134, 1997.

[17] M.O. Rabin and D. Scott. Finite automata and their decision problems. *IBM J. Res. and Develop.*, 3:114–125, 1959.

[18] J. Reiterman. The birkhoff theorem for finite algebras. *Algebra Universalis*, 14:1–10, 1982.

[19] M.P. Schützenberger. On finite monoids having only trivial subgroups. *Information and Control*, 8:190–194, 1965.

[20] T.A. Sudkamp. *Languages and Machines.* Addison-Wesley Publishing Company Inc., 1991.