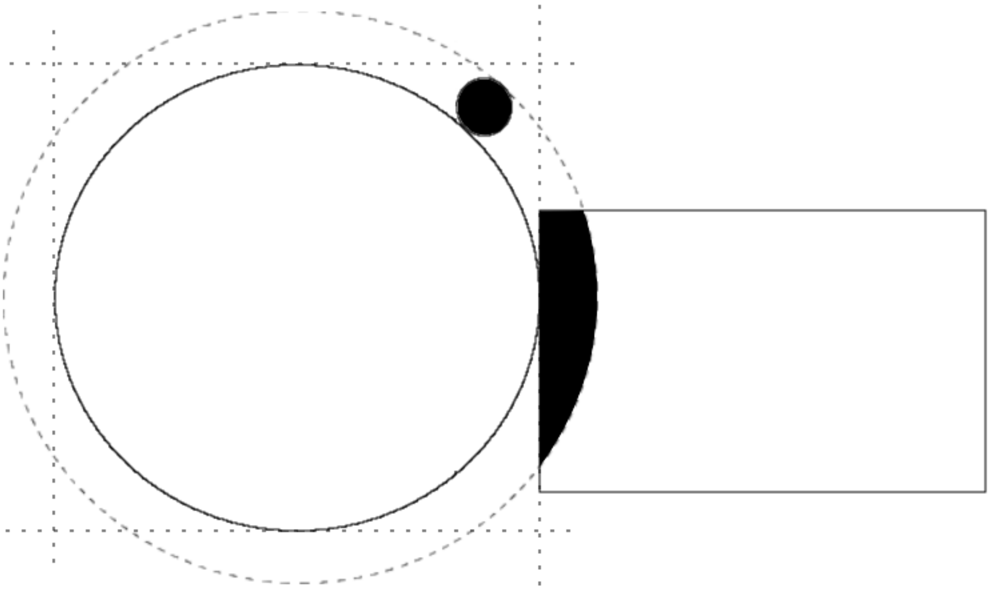
A Logical Perspective on Mathematical Morphology

A MsC Thesis in Artificial Intelligence

Brammert Ottens



supervisors
Marco Aiello
Rein van den Boomgaard

FACULTEIT DER NATUURWETENSCHAPPEN, WISKUNDE EN INFORMATICA

February 2, 2007

abstract

In this thesis a link between Mathematical Morphology and modal Logic is investigated. From the dilation a new modal language is distilled for which two separate axiomatisations are given. Both in an extended modal logic. This is due to the fact that the notion of singletions is needed in the axiomatizations. The applications of this new language in the field of qualitative spatial reasoning are explored. Furthermore, a reasoning method based on resolution is given and a experimental implementation is provided.

Acknowledgements

First of all I would like to thank Marco Aiello. Without his help and supervision this thesis would never have come into being. I would also like to thank Yde Venema, Agata Ciabattoni and Rosalie Iemhoff for the fruitful discussions and the critical but very helpful comments on my early writings. Also, I would like to thank Rein van de Boomgaard for helping me find a supervisor abroad.

My thanks also goes to Scharham Dustard for giving me an office space at the VitaLab in Vienna where I have spend six months of my life. I would like to thank Carlos Areces for providing me with the source code of HyLoRes and I would also Like to thank Gerben de Vries for taking the trouble of reading my Thesis.

Finally my thanks and admiration go to my girlfriend, Mirjam de Vries for the support she gave me during all the ups and downs that come with writing a thesis.

Contents

Chapter 1

Introduction

The field of computer vision has the objective of making computers "see", where by seeing we mean the interpretation of visual data. Visual data can consist of a camera feed coming from a camera, but also a collection of figures, a scan of a text or photo or perhaps even a radar image. One of the things that all these forms of data have in common is that they have a spatial component. The objects that are "visible" in the data have spatial relations to other objects occurring in the data. This spatial data is very important, also in our everyday life. We use it to navigate through a room, to recognize objects and so on.

Apart from having a scientific relevance, computer vision is also applicable in a wide range of non-scientific fields. For instance, in the medical world many new imaging techniques are being developed to help doctors analyze this MRI-data. Techniques like MRI-scanners provide a huge amount of visual data that either needs to be pre-processed or analyzed in order for the doctors to be able to do their jobs. In providing good surveillance of public places like airports a large array of cameras is used. Most of the time, the amount of data that is produced by these cameras is too large to be analyzed by a single person. Computers must help in automatically analyzing the images captured by these security cameras.

Mathematical Morphology is one of the techniques that is used, for instance in computer vision, to solve the above mentioned problems. Above that, it is still an active field of research. Mathematical Morphology is a geometric language of shape. It views images as collections of regions and makes use of geometrical relations between shapes to analyze and modify images. However, the Mathematical Morphologyprocess works mindlessly. One is not able to use the techniques of Mathematical Morphology to reason about the information in the pictures. One can only use it's techniques to distill information from images.

If one wants to reason about information, one normally looks at logic because, traditionally, logic is the field that studies reasoning. Aristotle started formalizing human reasoning using syllogisms like "All humans are mortal, Socrates is human thus Socrates is mortal". From these early beginnings logic has come very far and these days extends from mathematics to linguistics and beyond. One of the subjects logic has found its way into is space. What does logic have to do with space, one might ask. One of the fields that tries to give an answer to this question is the field of Qualitative Spatial Reasoning (QSR). QSR is a general way to look at space. Objects, or regions, have certain properties and satisfy relations between each other. For example, an object can be a circle, a square or some more exotic shape. A region can touch another region or it can be separated from this region. Note that these relations are qualitative. The actual distance or size of an object does not matter. Often basic spatial relations between shapes or regions are the focus of QSR. It tries to formalize these relations and characterizations of regions in space. Since an image is a two dimensional space QSR can also be applied to them.

In this thesis a logic is combined with Mathematical Morphology to create a new language to reason about space. As mentioned above, Mathematical Morphology takes a geometrical view on images, in logic reasoning mechanisms are studied. Combining the two, a language that can talk about space can be created. Contrary to traditional spatial reasoning, the object of study is not some basic relation between shapes or regions, but a morphological operation on these shapes and regions with which several spatial relations can be defined.

The focus of the present thesis is the definition of a language that takes the geometric power of mathematical morphology and enables reasoning about space, especially about mereotopology, shape and basic geometry. In [3, 4, 15, 24] several links between Mathematical Morphology and logic are hinted. First, in [3, 24] a link between Mathematical Morphology and linear logic is introduced. Links between Mathematical Morphology and modal logic [14] are presented in [3, 4, 15, 1]. In this thesis the focus lies on the link between Mathematical Morphology and Modal Logic presented in [3, 4]. The link is explored in two stages. First, it is made explicit. Second, the consequences of this link are explored. How does it help Mathematical Morphology and can it help us in reasoning about space.

Using logic to reason about space and spatial information is not new. In reasoning about space several approaches can be taken. First of all, one can try to capture the underlying structure of space. In this case the underlying structure refers to concepts like the boundary of a region or the part-whole relationships In [38] an axiomatization of topology is given. However, two shapes are considered equal if they can be deformed into each other without cutting through them. For example, a donut and a wheel are considered equal because they both have one hole. An eight on the other hand, has 2 holes and thus belongs to another type of objects. In [39] an axiomatization of geometry is given.

Another approach to reasoning about space is the approach taken in the Qualitative Spatial Reasoning community. They try to define qualitative relations between regions by using certain basic concepts. The best known example is the RCC-8 calculus [30]. This calculus is based on the notion of connectedness, which defines when two regions are connected. Using connectedness a wide array of qualitative relations can be defined. For a good overview of spatial reasoning the reader is referred to [19].

The goal of QSR is not just to define new languages that one can use to model space. One also wants to be able to reason with these languages. One such reasoning method is called resolution. It was introduced by Robinson [32] and aims at decomposing sentences. This way it implicitly tries to build a model for them. If this process fails no model can be build. Resolution is also the reasoning method that is used in this thesis, mainly due to the fact that resolution is, to the authors knowledge, the most matured reasoning mechanism for Hybrid logic (the logic used in this thesis).

Mathematical morphology (MM) deals with space as well, although in a different manner than QSR does. It was developed in the 60s by Matheron and Serra as a method for the estimation of ore deposits [28, 33] and underlies modern image processing, where it has a wide variety of applications. Compared with classical signal processing approaches it is more efficient in image pre-processing, enhancing object structure and segmenting objects from the background. The idea behind MM is that one can find objects with different properties by probing an image with so called 'structuring elements'. Although Serra and Matheron developed their theory for binary images, Morphological operators exist also for both grey scale and color images. Links between space and Mathematical Morphology have been made as well. In [16] the relation "between" is analyzed in terms of Mathematical Morphology. In [34] the concept of convexity is defined using concepts from Mathematical Morphology.

The remainder of this thesis is structured as follows. In Chapter 2, both logic and Mathematical

Morphology are briefly introduced. In Chapter 3, the link between Mathematical Morphology and modal logic is formalized and proven in the form of a new language, the morpho language. In Chapter 4, a resolution calculus is proposed. It facilitates reasoning with the morpho language. The calculus is implemented in an existing theorem prover. Chapter 5 discusses the results of experimenting with the theorem prover introduced in Chapter 4 and finally in Chapter 6, the expressive power of the new language is explored, both with respect to Mathematical Morphology and with respect to Spatial Reasoning. Chapter 7 holds the conclusions.

Parts of this thesis have been published in [1].

Chapter 2

Morphological and logical Preliminaries

Mathematical Morphology is an image processing tool that is used to extract geometric information from pictures. It's was conceived in 1965 and has since developed into a mature field of it's own.

Logic is a field that studies both formal languages as well as human reasoning. It's origins lie in ancient Greece, and it has since developed into a field with a wide array of applications. From natural language understanding to process verification. Several links between Mathematical Morphology and logic exist, allowing us to create a formal language that can be used to both study Mathematical Morphology and model spatial reasoning.

In Section 1, we introduce Mathematical Morphology by giving a brief overview of it's history and looking at both the practical side and the underlying algebraic theory. In Section 2, we introduce the basic logical concepts that are needed in the following treatment. First we give a brief introduction to logic. Second, we introduce Modal Logic. Finally, in Section 3, we introduce the concepts of frame definability and completeness.

2.1 Morphological preliminaries

Mathematical Morphology [22, 33, 35] was born in 1965 from the work of J. Serra and G. Materhorn (for a comprehensive overview on the birth of MM see [29]). They were working on methods for the estimation of ore deposits and found the operations that today from the basis of Mathematical Morphology. From then on, Mathematical Morphology has evolved into a field of it's own, with applications mainly in Image Processing. The idea behind MM is that one can find objects with different properties by probing an image with so called 'structuring elements'. The probing is done by two operations, the dilation and the erosion. Although Serra and Materhorn developed their theory for binary images, morphological operators exist for both gray scale and color images as well. For reasons of simplicity, we focus on binary images.

First, we explain in more detail what Mathematical Morphology is and how it is used in computer vision. Second, we explain the algebraic theory behind Mathematical Morphology.

2.1.1 Dilation and Erosion

Mathematical Morphology consists of a set of operations on images. All these operations are constructed by combining two basic operations, the dilation and erosion. Both are used to 'probe' the image using a structuring element. The dilation, defined in Definition 2.1.1, can be used to see whether an object fits a specific region outside a shape. The erosion, defined in Definition 2.1.2, can be used to see whether a shape (the structuring element) fits in another shape.

Definition 2.1.1 Dilation: Given an image A and a structuring element B, a dilation $\oplus$ is defined as follows

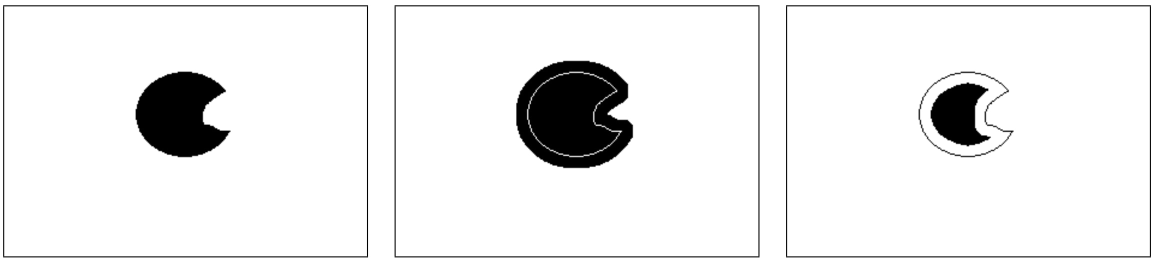
$$A \oplus B = \{x \in \mathbb{R}^2 | \check{B}_x \cap A \neq \emptyset\} = \{a + b | a \in A, b \in B\}$$

where $Y_x = \{x + y | y \in Y\}$ and $\check{B} = \{-b | b \in B\}$.

Definition 2.1.2 Erosion: Given an image A and a structuring element B, an erosion $\ominus$ is defined as follows

$$A \ominus B = \{x \in \mathbb{R}^2 | B_x \in A\}$$

One of the main ideas in Mathematical Morphology is to view images as sets. In the binary case for example, an image is a subset of the $\mathbb{R}^2$. This enables us not only to use the dilation and erosion, but also the concepts of union ($A \cup B$), intersection ($A \cap B$) and complement ($\overline{A}$).



(a) Original figure (b) Dilation (c) Erosion

Figure 2.1: dilation and erosion (with a circular structuring element)

Referring to Figure 2.1, the dilation of an image is equivalent to stamping the structuring element in each pixel in the image. The erosion is equivalent to finding all the points such that the structuring element is contained in the shape when placed on the point.

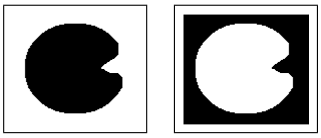
The dilation and erosion have some nice properties. First of all, erosion and dilation are each others dual. This means that the following equation holds

$$A \oplus B = \overline{\overline{A} \ominus \check{B}} \tag{2.1}$$

Informally, dilating an image with a structuring element is equivalent to eroding the complement with the mirrored structuring element (see Figure 2.2). The dilation and erosion are not just two operators, they are linked. Another piece of evidence of this link is the following

$$A \oplus B \subseteq Y \Leftrightarrow A \subseteq Y \ominus B \tag{2.2}$$

This equation tells us that if a region A is contained in a region Y after dilation with B, the original region A is contained in the erosion of Y with B. However, there are some differences in the properties that dilation and erosion posses. For example, it is the case that the dilation distributes over the union, while the erosion distributes over the intersection. This is illustrated by the following formula's



(a) Dilation (b) Erosion

Figure 2.2: duality of erosion and dilation

$$(A \cup B) \oplus C = (A \oplus C) \cup (B \oplus C) \tag{2.3}$$

$$(A \cap B) \ominus C = (A \ominus C) \cap (B \ominus C) \tag{2.4}$$

Furthermore, the dilation operator is an associative and commutative operator. Associativity means that $(A \oplus B) \oplus C = A \oplus (B \oplus C)$, commutativity means that $A \oplus B = B \oplus A$. Interestingly, the erosion is neither associative or commutative. Instead the following relation holds,

$$(A \ominus B) \ominus C = A \ominus (B \oplus C) \tag{2.5}$$

The binary dilation and erosion posses some nice mathematical properties, but what can one do with them? Although the binary dilation and erosion are rather simple operators, many interesting applications can be build using these operators as basic building blocks.

2.1.2 Applications

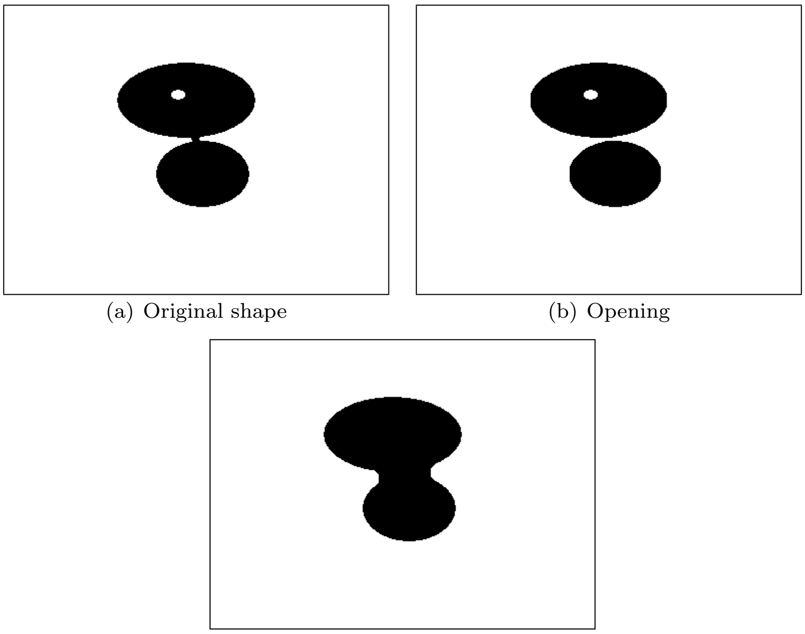
As already mentioned, morphological operations are combinations of dilations and erosions.The most basic of these sequences are the opening and closing. The opening consists of an erosion followed by a dilation $A \circ B = (A \ominus B) \oplus B$, whereas the closing consists of a dilation followed by an erosion, $A \bullet B = (A \oplus B) \ominus B$.

Figure 2.3 shows how the opening and closing operations can be used to remove noise from an image. For example, if two regions in an image are connected through noise, one can use the opening to disconnect these two regions. The closing can be used to close holes in an image. The size of the holes that can be closed depends on the size of the structuring element. One can also combine both the opening and the closing in a *salt-and-pepper* filter

$$(A \bullet B) \circ B \tag{2.6}$$

This filter can remove both holes (salt) and black noise (pepper) from an image. A more advanced application is the so called *reconstructive* opening. The idea behind reconstruction is that one starts with a marker, and by successively applying the dilation find all the regions connected with the marker. In the case of the reconstructive opening the marker is found by opening the image. In Figure 2.4 an example is given.

Another application is called *skeletonisation*. This method is used to find the skeleton (a line model) of the image. For example, in optical character recognition (OCR), one does not need all the information in an image. One does not need to know the thickness of a letter, or the font. One only needs to look at the skeleton of the shape. This skeleton can be found using skeletonisation. The idea behind skeletonisation is the concept of a maximal disk. Given a point in the interior of a binary shape, there exists a largest disk with the point as its center that still lies within the shape. Such a disk is maximal if there is no other disk lying on some other point properly containing this disk (see fig. 2.5). Do note, however, that skeletonisation cannot be easily and uniquely defined on a discrete grid.



(a) Original shape (b) Opening

(c) Closing

Figure 2.3: Dilation and erosion

Finally we briefly introduce the operator that started everything, the hit-or-miss transform. The hit-or-miss transform is a filter that passes shapes with certain properties, but blocks shapes with other properties. For example, using this filter it is possible to get all the points in an image that lie on a left edge. A nice example is seen in Figure 2.6. For a more thorough introduction see chapter 6.

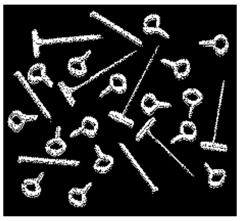
2.1.3 Algebraic theory

Mathematical Morphology is not only about binary images, this binary interpretation is part of a much more generic theory which is called the algebraic basis of Mathematical Morphology [26]. The algebraic theory brings to the surface the link between Mathematical Morphology and modal logic.

In introducing the algebraic theory of Mathematical Morphology we follow [26]. In the following it is assumed that the concepts of a partial order, group and complete lattice are known. In appendix A.0.6 more information can be found.

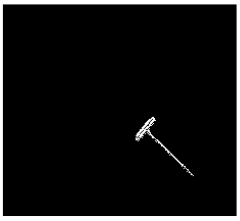
The first thing we can observe is that there is some kind of order in the structure containing the subsets of $\mathbb{R}^2$. On can say that a set A is a subset of another set B, and this subset relation defines an order on the set of subsets of $\mathbb{R}^2$. It is what one calls a partial order. For example, look at the set $A = \{a, b, c\}$. This is a set containing three elements. Both $\{a\}$ and $\{a, b\}$ are subsets of A and $\{a\} \subseteq \{a, b\}$. We can thus say that $\{a\}$ is smaller than $\{a, b\}$. Furthermore, taking the set-operators union and intersection, where the former is the supremum and the latter is the infimum, $(\mathcal{P}(\mathbb{R}^2), \subseteq)$ is a complete lattice.

It turns out that the dilation and erosion operators are part of a larger theory on complete lattices. In the following, consider a complete lattice $\mathcal{L}$ with the order relation $\leq$, supremum and infimum, least element O and greatest element I. Elements of $\mathcal{L}$ will be denoted by X, Y, Z. The set $\mathcal{O}$ is the set of transformations on $\mathcal{L}$. A transformation $\beta \in \mathcal{O}$ is a function $\beta : \mathcal{L} \to \mathcal{L}$. An element from $\mathcal{O}$ will be called an operator.



(a) Original shape

(b) Opening



(c) Closing

Figure 2.4: reconstructive opening

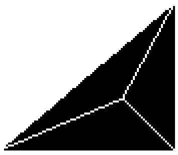


Figure 2.5: skeletonisation, the white lines represent the skeleton of the triangle

Using the order, the supremum and the infimum we define the following three classes of operators:

Definition 2.1.3 Algebraic dilation and erosion. [26]: Let $\beta \in \mathcal{O}$,

a. β is *increasing* if for every $X, Y \in \mathcal{L}$, $X \leq Y$ implies that $\beta(X) \leq \beta(Y)$

b. β is a *dilation* if for every $K \subseteq \mathcal{L}$, $\beta(\bigvee K) = \bigvee_{X \in K} \beta(X)$

c. β is an *erosion* if for every $K \subseteq \mathcal{L}$, $\beta(\bigwedge K) = \bigwedge_{X \in K} \beta(X)$

Note that the binary dilation (2.1.1) and erosion (2.1.2) are a dilation and erosion according to this definition as well. For example, given two regions A and B and a structuring element C. First dilating A and B with C and than taking the union gives the same result as first taking the union of A and B and then dilating with C. Both the dilation and erosion are increasing operators.

(a) Original shape

(b) right side of the triangle

1	1	0
1	1	0
1	1	0

(c) mask

Figure 2.6: hit-or-miss

We have already seen that the dilation and erosion are related. Outside the fact that they are each others dual, they satisfy equation 2.2. The term adjunction, defined below, is a generalization of this concept.

Definition 2.1.4 Let $\delta, \varepsilon \in \mathcal{O}$. Then we say that (ε, δ) is an adjunction if for every $X, Y \in \mathcal{L}$, we have that

$$\delta(X) \leq Y \Leftrightarrow X \leq \varepsilon(Y)$$

ε is called the upper adjoint and δ the lower adjoint.

Looking closely at Equation 2.2, one can see that the binary dilation and erosion together form an adjunction in which $\oplus$ is the lower adjoint and $\ominus$ is the upper adjoint.
The following proposition tells us that there is a very strong link between the notions of algebraic dilation and algebraic erosion and the notion of an adjunct.

Proposition 2.1.5 *Let $\delta, \varepsilon \in \mathcal{O}$. If (ε, δ) is an adjunction, then δ is a dilation and ε and erosion*

The proof can be found in [26]. Another aspect of the binary dilation and erosion as defined in (2.1.1) and (2.1.2) is that they are translation invariant. By this we mean that first translating an image and then applying the dilation/erosion yields the same result as first applying the dilation/erosion and then translating the result. This property, the translation invariance, can also be generalized to the general framework presented here.
In the case of binary MM, a translation maps subsets of $\mathbb{R}^2$ to subsets of $\mathbb{R}^2$. It does that in such a way that it preserves the subset-relation. Hence, it is an automorphism. We write $\mathrm{Aut}(\mathcal{L})$ for the set of automorphism of $\mathcal{L}$. Given an automorphism $\tau \in \mathrm{Aut}(\mathcal{L})$ and an operator $o \in \mathcal{O}$ we say that o is τ-invariant if $o\tau = \tau o$. Given a subset T of $\mathrm{Aut}(\mathcal{L})$, we say that o is T-invariant if o is τ-invariant for every $\tau \in T$.

It turns out that $\mathrm{Aut}(\mathcal{L})$, together with composition is a group. By definition every isomorphism has a reverse. And in the case of automorphism, this reverse is an automorphism itself. Furthermore, given any $\mathcal{Q} \subseteq \mathcal{O}$, the automorphisms of $\mathcal{L}$ that commute with every element of $\mathcal{Q}$ form a subgroup of $\mathrm{Aut}(\mathcal{L})$. Thus, every $\mathbf{T}$-invariant set of operators for some $\mathbf{T} \subseteq \mathrm{Aut}(\mathcal{L})$ is a group.

Given a group of automorphisms T, an adjunction (ε, δ) is called a T-adjunction if δ is a T-dilation and ε is a T-erosion. In the case of Euclidean space ($\mathcal{E}$), translation invariant dilations and erosions can be constructed from translations. First, we fix the origin o. Second, every point x defines a unique translation τ_x defined by $\tau_x(o) = x$. For $X \subset \mathcal{E}$, $\tau_x(X) = X_x$. For every $A, B \in \mathcal{E}$ definitions 2.1.1 and 2.1.2 give us that

$$X \oplus Y = \bigcup_{y \in Y} \tau_y(X) \text{ and } X \ominus Y = \bigcap_{y \in Y} \tau_y^{-1}(X)$$

Using the notation from the complete lattice theory, the dilation $\delta_A : X \to X \oplus A$ and the erosion $\varepsilon_A : X \to X \ominus A$ have the following decomposition in terms of translations:

$$\delta_A = \bigvee_{a \in A} \tau_a \text{ and } \varepsilon_A = \bigwedge_{a \in A} \tau_a^{-1}$$

We can generalize this result to an arbitrary complete lattice $\mathcal{L}$. However, it does not work for arbitrary operators. Hence, we must define the set of operators for which this result is possible. To do this, first look at the properties of Euclidean spaces. The important concepts above are the translations and the singletons that define these translations. Considering the singletons, in Euclidean spaces we know that all the subsets can be build from singletons (the points in space) using the union operation. This property can be generalized to a general complete lattice in the following manner.

Definition 2.1.6 Sup-generating subset Given a complete lattice $(\mathcal{L}, \leq)$, a sup-generating subset l is a subset of $\mathcal{L}$ s.t. every element of $\mathcal{L}$ can be written as a supremum of elements of l.

In the case of Euclidean spaces, l is the set of singletons (i.e. $l = \mathbb{R}^2$) and the supremum is the union. Looking at the translations on the Euclidean space, we see that the group of translations is commutative and transitive on the set of singletons. Furthermore, a translation of a singleton set always gives us another singleton set. These properties combined give us that we can define a unique translation τ_x by $\tau_x(o) = x$.

Generalizing to arbitrary complete lattices, this gives us the following *basic assumption*, considering a complete lattice $\mathcal{L}$ and a commutative group T of automorphisms of $\mathcal{L}$,

Basic Assumption 1 $\mathcal{L}$ has a sup generating subset l s.t.

- T leaves l invariant, in other words for every $\tau \in T$ and $x \in l$, $\tau(x) \in l$
- T is transitive on l, in other words for every $x, y \in l$, there exists $\tau \in T$ such that $\tau(x) = y$.

From the basic assumption we can conclude that for every $x, y \in l$, there is a unique $\tau \in T$ s.t. $\tau(x) = y$. This follows from the following facts. Suppose that $\tau_1(x) = \tau_2(x) = y$. Then $\tau_1^{-1}\tau_2(x) = x$. Because of the basic assumption we know that for any $z \in l$, there is some $\tau_3 \in T$ s.t. $\tau_3(x) = z$. So $\tau_1^{-1}\tau_2(z) = (\tau_1^{-1}\tau_2)\tau_3(x) = \tau_3(\tau_1^{-1}\tau_2)(x) = \tau_3(x) = z$. We now know that for all $z \in l$, $\tau_1^{-1}\tau_2(z) = z$. Thus $\forall z \in l$ we have that $\tau_2(z) = \tau_1(z)$, hence $\tau_1 = \tau_2$.

Using the fact that for every $x, u \in l$ there is a unique $\tau \in T$ such that $\tau(x) = y$ we can define several operators on l.

Definition 2.1.7 Fix some $o \in l$. Next, for every $x \in l$ define τ_x as the unique element of T s.t. $\tau_x(o) = x$. Using this bijection between l and T we can define the binary addition on l by

$$x + y = \tau_x \tau_y(o) = \tau_x(y) = \tau_y(x)$$

Furthermore, we define $-y$ by τ_y^{-1}. Thus

$$x - y = x + (-y) = \tau_x \tau_y^{-1}(o) = \tau_y^{-1}(x)$$

For $X \in \mathcal{L}$ we define $\tau_y(X) = X_y = \{\tau_y(x) | x \in X\}$

The structure $(l, +, -, o)$ constitutes a group. If we apply this to the Euclidean space, we can take the set of singletons as l and T naturally becomes the set of all translations on the set $\mathcal{E}$ of points.

Using the group created by the operators from Definition 2.1.7 we can define the following binary operators

Definition 2.1.8 Given a complete lattice $\mathcal{L}$, a sup-generating subset l and $X, Y \subseteq \mathcal{L}$

- $X \dot{\oplus} Y = \bigvee_{y \in \bar{l}(Y)} X_y$
- $X \dot{\ominus} Y = \bigwedge_{u \in \bar{l}(Y)} X_{-y}$

with $\bar{l} : \mathcal{L} \to l$ s.t. $\bar{l}(X) = \{x \in l | x \leq X\}$

Proposition 3.5 in [26] tells us the following:

Proposition 2.1.9 *For $X, Y \in \mathcal{L}$ and $x, y \in l$ we have,*

- $X \dot{\oplus} Y = Y \dot{\oplus} X = \bigvee \{x + y | x \in \bar{l}(X), y \in \bar{l}(Y)\}$
- $X \dot{\ominus} Y = \bigvee \{z \in l | Y_z \leq X\}$

We now define the operators δ_A and ε_A by

$$\delta_A = \bigvee_{a \in l(A)} \tau_a \text{ and } \varepsilon_A = \bigwedge_{a \in l(A)} \tau_a \tag{2.7}$$

moreover, we have that

$$\delta_A(X) = X \dot{\oplus} A \text{ and } \varepsilon_A(X) = X \dot{\ominus} A \tag{2.8}$$

To end this section, let us look at the following theorem (Theorem 3.6 in [26]).

Theorem 2.1.10 *For any $A \in \mathcal{L}$, $(\varepsilon_A, \delta_A)$ is a T-adjunction on $\mathcal{L}$. Moreover, any T- adjunction has this form.*

Using Theorem 2.1.10 and 2.1.5 we state that for each subset A of a complete lattice $\mathcal{L}$, we can find a dilation δ and an erosion ε such that $\delta(X) = X \dot{\oplus} A$ and $\varepsilon(X) = X \dot{\ominus} A$. This tells us that every translation invariant operator can be written as a union of translations. The fact that operators can be written in this form lies at the heart of the link between Mathematical Morphology and logic.

2.2 Logical preliminaries

In the following we give a brief introduction to logic. We introduce both propositional and modal logic and the notation that will be used in the rest of this treatment. The most well know logic is first order logic. This logic is not treated in the present section but for more information we refer to [42, 14].

Historically, logic has been the study of reasoning. It was designed to formalize the reasoning processes of the human mind. To formalize these processes, one needs a formal language. Such a formal language is exactly what the link between Mathematical Morphology and logic consists of. But before we reach that language, we start with a more intuitive language to explain the basic concepts needed in the rest of this treatment.

2.2.1 Propositional logic

Propositional logic is a simple language that is meant to formalize the reasoning with propositions. Propositions can be statements about the world, for instance,

- today was a rainy day
- Bart is a boy
- I have an umbrella

These propositions can be either true or false. However, reasoning with propositions is more than just stating the truth of a proposition. One can also combine several propositions into more complicated statements like

- today was a rainy day **and** I have an umbrella
- **If** I have an umbrella, **then** I do not get wet
- Bart is a boy **or** Bart is a girl

In formalizing propositions, a proposition is usually denoted with p, q or r to generalize from specific propositions. We say that PROP is the set of all propositions. For creating more complicated statements involving several propositions, we have the following notation

- and: $\wedge$
- or: $\vee$
- if,then: $\rightarrow$
- not: $\neg$
- if and only if: $\leftrightarrow$

Furthermore, we have the symbols $\bot$ and $\top$ that denote false and true respectively. These symbols are from here on called connectives and we can use them to combine propositions.

In fact, we can do the same with less symbols. For example, the $\wedge$ can also be written using only $\vee$ and $\neg$. Then $p \wedge q$ is equal to $\neg(\neg p \vee \neg q)$. We therefore arrive at the following

Definition 2.2.1 Sentence: We call a formula φ a *sentence* if it adheres to the following

$\varphi := p|\bot|\neg\varphi|\varphi \vee \psi$ with $p \in$ PROP, with ψ a sentence.

As for the other connectives, they can be written as follows. $\varphi \wedge \psi = \neg(\neg\varphi \wedge \neg\psi)$, $\varphi \rightarrow \psi = \neg\varphi \vee \psi$, $\varphi \leftrightarrow \psi = (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$ and $\top = \neg\bot$.

Some examples of sentences are

- $p \wedge q$
- $(p \vee q) \rightarrow r$
- $(p \rightarrow q) \leftrightarrow (q \rightarrow p)$

We have defined the language of propositional logic, but still need to define its semantics. We need to define what the connectives mean. In the propositional language, one can discern two parts. First, we have the propositions and second the connectives that are used to combine propositions into more complex propositions. In defining the semantics again a division is made between defining the semantics of the propositions and the semantics of the connectives.

First consider the propositions. A proposition can be either **true** or **false**. Which propositions are **true** and which are **false** is captured in the notion of a model.

Definition 2.2.2 Model: In propositional logic a *model* $\mathcal{M}$ consists of a *valuation* $\mathcal{V}_{\mathcal{M}}$. A valuation is a function $\mathcal{V} : \text{PROP} \mapsto \{\textbf{true}, \textbf{false}\}$ that assigns to every proposition p a truth value. To say that a formula (sentence) φ is true in a model $\mathcal{M}$ we use the following notation

$$\mathcal{M} \models \varphi$$

If a formula is true in every model, it is called a *tautology.* Furthermore, $\mathcal{V}(\bot) = \textbf{False}$ and $\mathcal{M} \not\models \varphi$ means that φ is not true in $\mathcal{M}$.

Using this notion of a model we define the propositional semantics.

Definition 2.2.3 Propositional semantics: Given a model $\mathcal{M}$

- $\mathcal{M} \models p$ with $p \in \text{PROP}$ iff[1] $\mathcal{V}(p) = \textbf{true}$
- $\mathcal{M} \not\models \bot$
- $\mathcal{M} \models \neg\varphi$ iff $\mathcal{M} \not\models \varphi$
- $\mathcal{M} \models \varphi \vee \psi$ iff $\mathcal{M} \models \varphi$ or $\mathcal{M} \models \psi$

To end this section, we provide some examples of propositional sentences. For example, the sentence $p \vee \neg p$ is a simple example of a tautology. It tells us that either p is true or not true, which obviously always is the case. A more elaborate tautology is for example $(p \rightarrow (q \rightarrow p))$. It says that p implies that q implies p. At first sight it is not very clear that this formula is true regardless of the valuation used. However, we can rewrite the $\rightarrow$ to a $\vee$, we get the formula $\neg p \vee (\neg q \vee p)$, which we again can rewrite to $(p \wedge q) \rightarrow p$. This formula has hidden inside it the formula $p \rightarrow p$, which again is another way of writing $p \vee \neg p$ and this is also a tautology.

Next, suppose that we have the following proposition

- When it rains the road is wet

This proposition contains the propositions "it rains" and "the road is wet". Denote the former with p and the later with q. The propositional logic formula that captures the above proposition is

$$p \rightarrow q$$

Then, given the fact that p is true, i.e. it rains, one can deduce that the road must be wet. This is a very simple example of how propositional logic can be used to model the real world using simple propositions.

[1] We use iff to stand for if and only if.

2.2.2 Modal logic

In propositional logic one can talk about propositions, but that is where it ends. A model consist of a set of propositional variables and their truth assignments. There is no further structure in the model. Modal logic is a family of logics that enables more structure in a model. This structure is present in the form of relations between the elements of a modal model. Hence, one can say that modal logic is the logic of relations.

We first look at the basic modal language. The language of the basic modal logic is an extension of the propositional language defined in Section 2.2.1.

Definition 2.2.4 Basic language: A formula φ is a sentence if

$\varphi := p|\bot|\neg\varphi|\varphi \vee \varphi|\Diamond\varphi|$ with $p \in$ PROP

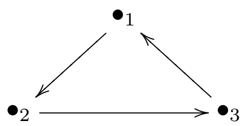
The symbols $\wedge, \rightarrow$ and $\leftrightarrow$ have the same definitions. Furthermore, we define the symbol $\Box = \neg\Diamond\neg$. The difference between this definition and Definition 2.2.1 are the symbols $\Diamond$ and $\Box$. The $\Diamond$ and $\Box$ are called modalities and are used to probe the structure of a model, which is captured by a frame. The way in which truth for propositional letters is defined is very different from the way used in propositional logic. Where in the latter a propositional letter is either **true** or **false**, in modal logic a propositional letter can be true in several worlds. Thus, it's truth value is denoted by a set of worlds in which it is true, rather then just **true** or **false**. This results in the following definition of a model.

Definition 2.2.5 Frames and models: A *frame* for the basic modal language $\mathcal{F}$ is a pair (W, R). W is a non-empty set of worlds, R is a binary relation s.t. $R \subseteq \mathcal{P}(W \times W)$. A *model* $\mathcal{M}$ is a pair $(\mathcal{F}, \mathcal{V})$ in which $\mathcal{F}$ is a frame and $\mathcal{V}$: PROP $\mapsto \mathcal{P}(W)$ a function. If $\mathcal{M} = (\mathcal{F}, \mathcal{V})$ for some valuation $\mathcal{V}$ we say that $\mathcal{M}$ is based on $\mathcal{F}$. A model is thus a frame enriched with a valuation.

The notation $\mathcal{M}, w \Vdash \varphi$ denotes the fact that the formula φ is true in the model $\mathcal{M}$ on world w. $\mathcal{M}, w \nVdash \varphi$ says that φ is false on the world w in $\mathcal{M}$. Where w is an element of W.

Note that the notation introduced above is different from the notation introduced in Definition 2.2.2. This is because we want to distinguish between truth in propositional logic and modal logic.

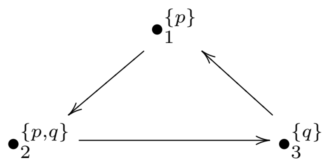
Example 2.2.6 A simple frame is



W $= \{1, 2, 3\}$, $R = \{(1, 2), (2, 3), (3, 1)\}$

If we for example take the valuation $\mathcal{V}(p) = \{1, 2\}$ and $\mathcal{V}(q) = \{2, 3\}$ we get the following model

Example 2.2.7



Just as in Definition 2.2.3 we use the notion of a model to define the semantics of the connectives and modalities.

Definition 2.2.8 Semantics: Given a model $\mathcal{M} = (W, R, \mathcal{V})$ and a $w \in W$ we say that

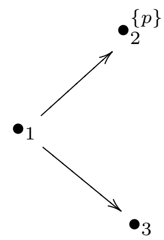
- $\mathcal{M}, w \Vdash p$ with $p \in$ PROP iff $w \in \mathcal{V}(p)$
- $\mathcal{M}, w \Vdash \neg\varphi$ iff $\mathcal{M}, w \nVdash \varphi$
- $\mathcal{M}, w \Vdash \varphi \vee \psi$ iff $\mathcal{M}, w \Vdash \varphi$ or $\mathcal{M}, w \Vdash \psi$
- $\mathcal{M}, w \Vdash \Diamond\varphi$ iff there exists a $v \in W$ s.t. $(w, v) \in R$ and $\mathcal{M}, v \Vdash \varphi$

The meaning of the normal connectives is straightforward. The meaning of the modalities however deserves some more explanation. The definition states that a formula $\Diamond\varphi$ is true in a model $\mathcal{M}$ on a world w if there is a world v s.t. $(w, v) \in R$ and φ is true on v. So $\Diamond\varphi$ is true on a world w if there is a world v that has a relation with w, and v makes φ true. In the same way, $\Box$ will get the following meaning.

$$\mathcal{M}, w \Vdash \Box\varphi \text{ iff for all } v \in W \text{ s.t. } (w, v) \in R \text{ implies } \mathcal{M}, v \Vdash \varphi$$

In other words. $\Box\varphi$ is true in a world w if and only if φ is true in every world v reachable from w.

Example 2.2.9 The formula $\Diamond p$ is true in the following model on world 1



with $W = \{1, 2, 3\}$, $R = \{(1, 2), (1, 3)\}$ and $\mathcal{V}(p) = \{2\}$.

However, the formula $\Box p$ is not true on world 1 in the model from Example 2.2.9 because there is a world accessible from 1 where p is not true. Note that $\Box p$ is true on the worlds 2 and 3 because they do not have any successors.

Definition 2.2.10 Satisfiability and validity: A formula φ is *satisfiable* in a model $\mathcal{M}$ if there is a world $w \in W$ s.t. $\mathcal{M}, w \Vdash \varphi$. A formula φ is *valid* on a model $\mathcal{M}$ if for every world $w \in W$ it is the case that $\mathcal{M}, w \Vdash \varphi$. We say that φ is valid on a frame $\mathcal{F}$, denoted by $\mathcal{F} \Vdash \varphi$ if it is valid on every model $\mathcal{M}$ based on $\mathcal{F}$. A formula is valid with respect to a family of frames if it is valid on every frame in the family. The same goes for families of models.

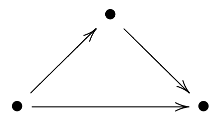
Using the notion of validity on a frame, we can create restrictions on the types of frames we want to consider. For example, the formula $p \to \Diamond p$ is only valid on a frame, if the frame is reflexive. This means that, given a frame $\mathcal{F}$, for all $w \in W$ we have that $(w, w) \in R$. One can see that this is the case through the following reasoning. Suppose that a frame $\mathcal{F}$ is not reflexive and $p \to \Diamond p$ is valid. This means that there is a world $w \in W$ such that $(w, w) \notin R$. Now take the valuation $\mathcal{V}$ such that $\mathcal{V}(p) = \{w\}$. This means that w is the only world that satisfies p. Hence, for the model $\mathcal{M} = (\mathcal{F}, \mathcal{V})$

we have that $\mathcal{M}, w \not\models p \to \Diamond p$ because p is true in w, but since w is the only world that satisfies p, and w cannot reach itself, $\Diamond p$ is false.

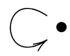
There is a wide array of formulas that restrict the family of frames. For example, the formula $\Diamond\Diamond p \to \Diamond p$ defines transitive frames. A frame $\mathcal{F}$ is transitive if for all $w, v, z \in W$ such that $(w, v) \in R$ and $(v, z) \in R$ we also have that $(w, z) \in R$.

Some examples of formulas together with the first order properties they define.

Example 2.2.11 The property of transitivity can be defined by the formula $\Diamond\Diamond p \to \Diamond p$. It corresponds to the first order property $\forall x, y, z(R(x, y) \wedge R(y, z) \to R(x, z))$. In terms of frames, this corresponds to the following structure



Example 2.2.12 The property of reflexivity can be defined by the formula $p \to \Diamond p$. It corresponds to the First Order property $\forall x R(x, x)$. In terms of frames, this corresponds to



Example 2.2.13 The property of symmetry can be defined by the formula $p \to \Box\Diamond p$, corresponding to the First Order property $\forall x, y(R(x, y) \to R(y, x)$. In terms of frames, this corresponds to

We gave the definition of the basic modal language. This basic modal language can only talk about one relation, a binary relation. But in general one can think of other relations that concern more than just two elements. For example, consider the set of real numbers $\mathcal{R}$ and the addition $+$. This addition can be seen as a ternary relation Plus. $1 + 2 = 3$ would be equivalent to $(3, 1, 2) \in$ Plus.

Also, one wants to be able to talk about several different relations between the elements of the model. Hence, a more general definition of a modal language is needed which can contain several modalities of different *arity*[2]. The first concept we define is the notion of a similarity type. A similarity type can be see as the signature of a logic. It contains the number of modalities, with their respective arities, that occur in the logic.

Definition 2.2.14 Modal similarity type: A *modal similarity type* is a pair $\tau = (O, \rho)$ where O is a non-empty set and ρ is a function $O \to \mathbb{N}$. O consists of modalities, denoted by $\triangle_1, \triangle_2, ...$ The function ρ assigns to every element of O a finite arity. For example, the arity of $\Box$ is 1.

The notion of a similarity type can now be used to generalize the definition of a modal language, together with it's semantics.

Definition 2.2.15 General modal language: Given a modal similarity type τ, we define a modal language such that

$\varphi := |p|\bot|\neg\varphi|\varphi_1 \vee \varphi_2|\triangle(\varphi_1, ..., \varphi_{\rho(\triangle)})$ with $p \in$ PROP, $\triangle \in \tau$ and ρ denoting the arity.

As for the frames and models, we have the following definitions

Definition 2.2.16 Frames and models: A frame $\mathcal{F}$ is a pair $(W, R_\triangle)_{\triangle \in \tau}$ where W is a set of worlds and $R_\triangle$ is a relation on W s.t. $R_\triangle \subseteq \mathcal{P}(W_1 \times ... \times W_{\rho(\triangle)})$. A model $\mathcal{M}$ is a pair $(\mathcal{F}, \mathcal{V})$ where $\mathcal{F}$ is a frame and $\mathcal{V}$ is a valuation.

[2] Arity is the number of arguments a modality takes.

The semantics for the connectives and $\bot$ does not change, and the dual of $\triangle$ is defined as $\nabla\varphi := \neg\triangle\neg\varphi$. Only the semantics for the modalities remains to be given.

Definition 2.2.17 General modal semantics: Given a model $\mathcal{M} = (W, R, \mathcal{V})$ and a world $w \in W$

$$\mathcal{M}, w \Vdash \triangle(\varphi_1, ..., \varphi_n) \text{ iff } \exists v_1, ..., v_n \in W \text{ with } (w, v_1, ..., v_n) \in R_\triangle \text{ such that } \forall i, \mathcal{M}, v_i \Vdash \varphi_i$$

Using this definition we are able to define logics for more complex structures, like for example groups that contain several relations between the elements of the model.

2.3 Frame definability and Completeness

We can define families of frames by modal formulas. In this section we formalise this notion. Furthermore, we look at the set of formulas that is true in a set of models and at the mechanisms that allows one to automatically find all these formulas.

2.3.1 Frame definability

We have already seen that the formula $\Diamond\Diamond p \to \Diamond p$ defines the class of transitive frames and that $p \to \Diamond p$ defines the class of reflexive frames. Both these properties are first order properties. That means that we can express them in First Order Logic.

Example 2.3.1 For example, the formulas defining reflexivity, transitivity and symmetry all have First Order counterparts which are expressed in Table 2.1.

Modal Logic	First Order Logic
$\Diamond\Diamond p \to \Diamond p$	$\forall x, y, z(R(x, y) \wedge R(y, z) \to R(x, z))$
$\Diamond p \to p$	$\forall x R(x, x)$
$p \to \Box\Diamond p$	$\forall x, y R(x, y) \to R(y, x)$

Table 2.1: some modal formulas and their First Order counterparts.

To make the notion of frame definability more formal, we introduce the following definition.

Definition 2.3.2 Frame definability: Given a similarity type τ, φ a formula of this type and **C** a class of τ-frames, we say that φ defines **C** if for all frames $\mathcal{F}$, $\mathcal{F}$ is in **C** if and only if $\mathcal{F} \Vdash \varphi$. Given a set of τ-formulas Γ, we say that Γ defines **C** if a frame $\mathcal{F}$ is in **C** if and only if $\mathcal{F} \Vdash \Gamma$.

A class of frames is modaly definable if there exists a formula or set of formulas that define this class.

2.3.2 Completeness

We have already seen that we can define a class of frames **S** by using formulas. One can ask oneself the question what the set of formulas Λ_S is that is valid on this class of frames. Perhaps more interestingly, is there a mechanism to find this Λ?

The answer to this question comes in the form of the notion of a normal modal logic. A normal modal logic is a set of formulas that satisfies the following properties.

Definition 2.3.3 Normal modal logic: A *normal modal logic* is a set of formulas Λ that contains the following axioms:

(K)	$\Box(p \to q) \to (\Box p \to \Box q)$
(DUAL)	$\Diamond p \leftrightarrow \neg\Box\neg p$

and is closed under the rules of modens ponens, universal generalization and uniform substitution.

Modens Ponens (MP):	from φ and $\varphi \to \psi$ we can derive ψ
Universal generalization (UG):	from φ we can derive $\Box\varphi$
Uniform Substitution (S):	from φ we can derive sub(φ) for any substitution sub

Introducing some useful notation, we say that a formula φ is deducible from a set of formulas Γ, $\vdash_\Gamma \varphi$, if there exists a finite sequence of formulas, ending in φ such that each formula of the sequence either is contained in Γ, is an axiom or can be obtained from previous formulas in the sequence by applying the rules (MP), (UG) and (S).

The logic **K** is defined as the smallest normal modal logic. That is, start with all the propositional tautologies, (K) and (DUAL) and close this set under the rules defined in def. 2.3.3. **K** is the logic of all the frames. But in order to prove that we need the following definitions.

Definition 2.3.4 [14] Soundness: Let S be a family of frames. A normal modal logic Λ is *sound* with respect to S if $\Lambda \subseteq \Lambda_S$. (Equivalently: Λ is sound with respect to S if for all formulas φ, and all structures $\mathcal{S} \in S$, $\vdash_\Lambda \varphi$ implies $\mathcal{S} \Vdash \varphi$. If Λ is sound with respect to S we say that S is a class of frames for Λ.

In other words, a normal modal logic Λ is *sound* with respect to some set of frames S if every formula that is derivable from Λ is true on S.

Definition 2.3.5 [14] Completeness: Let S be a family of frames. A logic Λ is *strongly complete* with respect to S if for any set of formulas $\Gamma \cup \{\varphi\}$, if $\Gamma \Vdash_S \varphi$ then $\Gamma \vdash_\Lambda \varphi$. That is, if Γ semantically entails φ on S, then φ is Λ-deducible from Γ.

A logic Λ is *weakly complete* with respect to S if for a formula φ, if $S \Vdash \varphi$ then $\vdash_\Lambda \varphi$. Λ is strongly complete(weakly complete) with respect to a single structure $\mathcal{S}$ if Λ is strongly complete(weakly complete) with respect to $\{\mathcal{S}\}$.

Informally, completeness means that given some normal modal logic Λ and the set of frames S it defines, every formula that is valid on S is derivable from Λ.

What we want to show is that, given the set of all frames **F**, **K** is both sound and complete with respect to **F**. Theorem 4.23 in [14] tells us that **K** is strongly complete with respect to the class of all frames. As for the soundness. Suppose that there is a frame $\mathcal{F}$ such that (K) is not valid. This would mean that we could find a world w such that $\mathcal{F}, w \Vdash \Box(p \to q)$ and $\mathcal{F}, w \nVdash \Box p \to \Box q$. From $\mathcal{F}, w \nVdash \Box p \to \Box q$ we can derive that $\mathcal{F}, w \Vdash \Box p$ and $\mathcal{F}, w \Vdash \neg\Box q$. The last formula tells us that there is a world v such that $(w, v) \in R$ and $\mathcal{F}, v \Vdash \neg q$. It should also be the case that $\mathcal{F}, v \Vdash p$, which means that $\mathcal{F}, v \nVdash p \to q$ and thus $\mathcal{F}, w \nVdash \Box(p \to q)$. Hence our assumption cannot be the case and $\Box(p \to q) \to (\Box p \to \Box q)$ must be valid. The second axiom reflects the fact that $\Box$ and $\Diamond$ are each others dual and is obviously sound.

As for the rules. The (MP) rule is sound as well. Just imagine the situation where you know that when A happens, B happens. Furthermore you know that A has happened. It now must be the case that B has happened as well.

The (S) rule uses the notion of a substitution. A substitution simultaneously replaces all the occurrences of a specified propositional letter with a formula. (S) captures the fact that the validity of a formula does not rely on a specific valuation. The (UG) rule is sound as well. If a formula φ is valid it will be impossible to falsify it on any world, hence $\Box\varphi$ must be valid as well.

2.4 Summary

Mathematical morphology is a formalism that is widely used in image processing. It consists of 2 basic operators, the dilation and the erosion. Using these operations many image operators can be defined. Furthermore, Mathematical Morphology has a solid algebraic base. The dilation and erosion can be defined in terms of operators on a complete lattice. Hence, the framework of mathematical morphology can be lifted to an arbitrary complete lattice.

Modal logic can be seen as the logic of relations. Modal formulas can isolate families of frames that contain the same property. After isolating the set of frames, one can ask oneself what the set of formulas is that is valid on this set of frames. Furthermore, one can ask oneself how one can create this set of frames.

In the chapters to come we investigate the connection between Mathematical Morphology and modal logic and show how these two fields can be combined into a theory of qualitative spatial reasoning.

Chapter 3

The Morpho-Language

At first sight there are several connections between Mathematical Morphology and logic. First, the dilation and erosion are adjuncts of each other. In logic the concept of adjunction is know under the name residuation. Linear logic is a logic that captures such residuals. Second, is the link between Mathematical Morphology and modal logic, arrow logic in particular (for a comprehensive overview of arrow logic see [43]). Arrow logic is a modal logic where the definitions of the modalities resemble the way in which translation invariant operators can be written. [3, 24] Finally, Isabelle Bloch finds a link between mathematical morphology and modal logic by taking the neighborhood relation as the frame relation [15, 2].

The focus of the this treatment lies on the link between Mathematical Morphology and arrow logic. To this end we introduce a new modal language, the morpho-language based on a modal logic containing a binary modality.

The remainder of this chapter has the following form. In Section 1, the intuitive connection between Mathematical Morphology and modal (arrow) logic is specified. In Section 2, the language and semantics of the morpho-language are introduced, an axiomatization for the logic is given and it is shown that it is complete. Finally, in Section 3, hybrid logic is introduced and an hybrid axiomatization of the morpho-language is given.

3.1 Mathematical Morphology and Logic

Arrow logic is a modal logic that consist of 3 modalities. A nullary, a unary and a binary modality. Together, they constitute the following language:

Definition 3.1.1 Arrow logic: The language of arrow logic consists of the following

$$\varphi := p|e|\neg\varphi|\varphi \vee \psi| \otimes \varphi|\varphi\hat{\oplus}\psi$$

Where $p \in \textsc{prop}$.

The other connectives are defined using the usual shorthands. As for the duals, define $\otimes\varphi = \neg\underline{\otimes}\neg\varphi$ and $\varphi\hat{\oplus}\psi = \neg(\neg\varphi\underline{\hat{\oplus}}\neg\psi)$. Although arrow logic was created to be the basic modal logic of arrows, we do not go into the details of the arrow semantics. We only use the semantics that have been defined in [43] to point at the similarity with the definition of translation invariant dilation.

Definition 3.1.2 A frame for arrow logic consists of three relations (I, R, C), thus a model is a tuple of the form $\mathcal{M} = (W, I, R, C, \mathcal{V})$.

Before we give the semantics of the relations I, R and C, we first given the semantics of the arrow logic, given these relations.

Definition 3.1.3 Semantics: Given a model $\mathcal{M}$ and a world w in W, arrow logic has the following semantics

$\mathcal{M}, w \Vdash p$	iff	$w \in \mathcal{V}(p)$, with $p \in$ PROP
$\mathcal{M}, w \Vdash e$	iff	$(w) \in I$
$\mathcal{M}, w \Vdash \neg\varphi$	iff	$\mathcal{M}, w \nVdash \varphi$
$\mathcal{M}, w \Vdash \varphi \vee \psi$	iff	$\mathcal{M}, w \Vdash \varphi$ or $\mathcal{M}, w \Vdash \psi$
$\mathcal{M}, w \Vdash \otimes\varphi$	iff	there exists a $v \in W$ such that $(w, v) \in R$ and $\mathcal{M}, v \Vdash \varphi$
$\mathcal{M}, w \Vdash \varphi\hat{\oplus}\psi$	iff	there exists $v, v' \in W$ such that $(w, v, v') \in C$, $\mathcal{M}, v \Vdash \varphi$ and $\mathcal{M}, v' \Vdash \psi$

Just as for the basic modal language, we can again lift the valuation $\mathcal{V}$ to the level of formulas. For now, we are only interested in the definition for the binary operator $\hat{\oplus}$. It has the form

$$\mathcal{V}(\varphi\hat{\oplus}\psi) = \{w | \exists v, v' : (w, v, v') \in C \text{ and } v \in \mathcal{V}(\varphi), v' \in \mathcal{V}(\psi)\} \tag{3.1}$$

Given a group $(l, +, -, e)$, in which l is the universe, $+$ the binary group operator, $-$ the operator that finds the reverse element and e the identity element, suppose that we give the relations I, R and C the following semantics:

- $(x, y, z) \in C$ if $x = y + z$
- $(x, y) \in R$ if $x = -y$
- $(x) \in I$ if $x = e$

We can now rewrite Equation 3.1 as follows:

$$\mathcal{V}(\varphi\hat{\oplus}\psi) = \{x | \exists y, z : x = y + z, y \in \mathcal{V}(\varphi), z \in \mathcal{V}(\psi)\} = \{y + z | y \in \mathcal{V}(\varphi), z \in \mathcal{V}(\psi)\} \tag{3.2}$$

Notice the similarities between the definition of $\oplus$ in Definition 2.1.3 and equation 3.2 in the case that the supremum is the union and the infimum is equal to the intersection. This similarity is at the heart of the link between Mathematical Morphology and modal logic. However, just using the axioms for a standard normal modal logic is not enough. In the semantics defined in Definition 3.1.3, the relations should adhere to the axioms of the group. In a basic normal modal logic however, this is not the case. Several axioms have to be added that define certain properties of the relations. Below two different approaches to do this are explained.

3.2 Morpho-Language

We have just shown the link between arrow logic and mathematical morphology. Now, let us be more formal. Suppose we have a commutative group $G = (W, +, -, e)$ and $\mathcal{P}(G)$ it's complex group. One wants to find an axiomatization that properly captures the properties of the group G.

In doing so several problems occur. One of the properties of the composition operator $+$ is that it is a binary function. A binary function should be total on the domain it is defined on. This means that for each $a, b \in W$ there should be a $c \in W$ such that $c = a + b$. In traditional modal logic this is not definable. However, in [25] a general method is devised to axiomatize algebraic structures using the difference operator.

Using this method, a modal axiomatization of several algebraic structures can be obtained via a translation of first order axioms to modal axioms. One starts by defining a signature σ. A signature is a set of function symbols, relation symbols and constants together with their arity. Using this signature, a structure can be defined.

Definition 3.2.1 A group structure In the case of a group, $\sigma = (+, -, e)$. A σ-structure contains a non-empty set, function symbols, constants and relations corresponding to σ. Given a σ-structure $G = (W, +, -, e)$, one can look at the power set of W. Lifting the σ operators to the power set, the complex σ-structure $\mathcal{P}(G)$ is obtained.

$$\mathcal{P}(G) = \langle \mathcal{P}(W), \emptyset, \text{—}, \cup, \hat{\oplus}, \otimes, \langle e \rangle \rangle$$

where, for $X, X_1, X_2 \subseteq W$,

- $X_1 \hat{\oplus} X_2 = \{x_1 + x_2 | x_1 \in X_1, x_2 \in X_2\}$
- $\otimes X = \{-x | x \in X\}$
- $\langle e \rangle = \{e\}$

On a set the difference operator $\langle \neq \rangle$ can be defined. In short, this operator finds all the elements that are not equal to the elements in the argument. For example, $\langle \neq \rangle A$ with A a singleton set returns $\overline{A}$, the complement of A. However, given a set that contains more elements, the whole universe is returned.

Example 3.2.2 For example, look at the set $\{a, b, c\}$. Given the subset $\{a, b\}$, we want to have $\langle \neq \rangle \{a, b\}$. We thus want to have all the elements in $\{a, b, c\}$ that are different from either a or b. Obviously, c is different from either a or b, but a is different from b and b is different from a. Thus $\langle \neq \rangle \{a, b\} = \{a, b, c\}$.

Definition 3.2.3 Difference operator: Given an arbitrary set W and a subset $A \subseteq W$, the *difference* operator is defined as

$$\langle \neq \rangle A = \{b \in W | \exists a \in A, b \neq a\}$$

Adding the difference operator to a complex algebra creates a differentiated complex algebra. From here on assume that a complex algebra is enriched with the difference operator.

From the σ-signature we can create a modal language $\mathcal{L}_\sigma$ containing 4 modalities. A binary for the binary function, a unary for the unary function, a unary modality representing the difference operator and a nullary modality for the constant.

Definition 3.2.4 Morpho-language $\mathcal{L}_\sigma$: The well formed sentences of the language $\mathcal{L}_\sigma$ are:

$$\varphi := \bot | p | e | \neg\varphi | \varphi \vee \psi | \otimes \varphi | \varphi \hat{\oplus} \psi | D\varphi$$

where $p \in$ PROP. The connectives are defined using the usual definitions. As for the duals, $\underline{\otimes}\varphi = \neg \otimes \neg\varphi$, $\varphi \underline{\hat{\oplus}} \psi = \neg(\neg\varphi \hat{\oplus} \neg\psi)$ and $\underline{D}\varphi = \neg D \neg\varphi$.

In Chapter 2 the semantics of a modal logic is defined using the concept of a model and a frame. In the case of $\mathcal{L}^\sigma$ the semantics can be found by looking at the complex σ-algebra. Given a complex σ-algebra $\mathcal{P}(G)$, one can create a frame $\mathcal{F}_\sigma$:

$$\mathcal{F}_\sigma = (W, \neq, C, R, I)$$

where $C = \{(x_1 + x_2, x_1, x_2) | \forall x_1, x_2 \in W\}$, $R = \{(-x, x) | \forall x \in W\}$ and $I = \{e\}$ and $\neq = \{(x, y) | x \neq y\}$. Using this knowledge one arrives at the following definition of a model and the semantics of the connectives

Definition 3.2.5 $\mathcal{M}_\sigma$: Given a complex σ-algebra $\mathcal{P}(G)$, a model $\mathcal{M}_\sigma$ can be obtained through the following definition $\mathcal{M}_\sigma = (\mathcal{F}_\sigma, \mathcal{V})$ with $\mathcal{V} : \text{PROP} \mapsto \mathcal{P}(W)$.

Definition 3.2.6 Semantics of $\mathcal{L}_\sigma$: Given a model $\mathcal{M}_\sigma$ and a world w in $\mathcal{M}_\sigma$ the semantics is defined as:

$\mathcal{M}, w \Vdash \bot$		is never true
$\mathcal{M}, w \Vdash p$	iff	$w \in \mathcal{V}(p)$, with $p \in \text{PROP}$
$\mathcal{M}, w \Vdash e$	iff	$(w) \in I$
$\mathcal{M}, w \Vdash \neg\varphi$	iff	$\mathcal{M}, w \nVdash \varphi$
$\mathcal{M}, w \Vdash \varphi \vee \psi$	iff	$\mathcal{M}, w \Vdash \varphi$ or $\mathcal{M}, w \Vdash \psi$
$\mathcal{M}, w \Vdash \otimes\varphi$	iff	there exists a $v \in W$ such that $(w, v) \in R$ and $\mathcal{M}, v \Vdash \varphi$
$\mathcal{M}, w \Vdash \varphi \hat{\oplus} \psi$	iff	there exists $v, v' \in W$ such that $(w, v, v') \in C$, $\mathcal{M}, v \Vdash \varphi$ and $\mathcal{M}, v' \Vdash \psi$
$\mathcal{M}, w \Vdash D\varphi$	iff	there exists a $v \in W$ such that $v \neq w$ and $\mathcal{M}, v \Vdash \varphi$

Satisfiability and validity are defined as usual.

The final goal of the method is to give an axiomatization of a class of σ-structures $\mathbf{C}$. From a class $\mathbf{C}$, a class of complex σ-algebras can be created, denoted by $\mathbf{C}^*$. The family of frames accompanying $\mathbf{C}^*$ is denoted by F^*. Before going to the class of σ-structures axiomatized by the group axioms, first an axiomatization is given for the class of all σ-structures.

The axiomatization consists of two parts. First, a set of inference rules is given. Some of these have already been introduced, namely the uniform substitution, modus ponens and necessitation. The remaining set of rules is introduced to give the difference operator the proper meaning. Second, a set of axioms is given that define certain properties of the modalities and how they interact with each other.

Before giving the axiomatization, two modalities need to be defined that are used in the axiomatization. Both modalities are defined using the difference modality D. The first modality that is defined is called the universal modality.

Definition 3.2.7 Universal modality: Given the difference operator D, the universal modality U is defined as:

$$U\varphi := \varphi \wedge \neg D\neg\varphi$$

The purpose of the universal modality is to define global truth. That is, a formula of the form $U\varphi$ is true in a model if and only if φ is true everywhere in the model. The dual of U, E is defined as $E\varphi = \neg U\neg\varphi$ and means that φ is true somewhere in the model.

The other modality that must be defined is the *only* modality. This modality has the power to state that a formula is true in only one point in the model.

Definition 3.2.8 Only modality Given the difference operator D, the only modality $\mathcal{O}$ is defined as:

$$\mathcal{O}\varphi := \varphi \wedge \neg D\varphi$$

We are now ready to state the axiomatization of the class of all σ-structures.

Definition 3.2.9 Inference rules: The inference rules that are necessary for a complete axiomatization are the following:

Uniform substitution(SUB): $\dfrac{\varphi}{sub(\varphi)}$

Where $sub(\varphi)$ is the result of an application of any uniform substitution of formulas for variables in φ.

Modens Ponens(MP):	$\dfrac{\varphi, \varphi \to \psi}{\psi}$	
Necessitation for U(N(U)):	$\dfrac{\varphi}{U\varphi}$	
Witness rule ($W_{\underline{\otimes}}$):	$\dfrac{\varphi \to (\underline{\otimes}(\mathcal{O}p \to \psi))}{\varphi \to (\underline{\otimes}\psi)}$	for some p not occurring in φ or ψ
Witness rule ($W_{\hat{\oplus}_1}$):	$\dfrac{\varphi \to ((\mathcal{O}p \to \psi_1)\underline{\hat{\oplus}}\psi_2)}{\varphi \to \psi_1\underline{\hat{\oplus}}\psi_2}$	for some p not occurring in φ, ψ_1 or ψ_2
Witness rule ($W_{\hat{\oplus}_2}$):	$\dfrac{\varphi \to (\psi_1\underline{\hat{\oplus}}(\mathcal{O}p \to \psi_2))}{\varphi \to \psi_1\underline{\hat{\oplus}}\psi_2}$	for some p not occurring in φ, ψ_1 or ψ_2

Definition 3.2.10 General Axioms: The following axioms are needed to arrive at a complete axiomatization of a structure containing a binary and a unary operator and a constant.

($K_{\underline{\hat{\oplus}},r}$)	$p\underline{\hat{\oplus}}(q \to r) \to ((p\underline{\hat{\oplus}}q) \to (p\underline{\hat{\oplus}}r))$
($K_{\underline{\hat{\oplus}},l}$)	$(p \to q)\underline{\hat{\oplus}}r \to ((p\underline{\hat{\oplus}}r) \to (q\underline{\hat{\oplus}}r))$
($K_{\otimes}$)	$\otimes(p \to q) \to (\otimes p \to \otimes q)$
(Dual$_{\otimes}$	$\otimes p \to \neg \otimes \neg p$
(Dual$_{\hat{\oplus}}$)	$p\hat{\oplus}q \to \neg(\neg p\hat{\oplus}\neg q)$
(D_1)	$p \vee \underline{D}\neg\underline{D}p$
(D_2)	$DDp \to (A \vee Dp)$
($U_{\underline{\otimes}}$)	$Up \to \underline{\otimes}p$
($U_{\underline{\hat{\oplus}}_1}$)	$Up \to p\underline{\hat{\oplus}}q$
($U_{\underline{\hat{\oplus}}_2}$)	$Up \to q\underline{\hat{\oplus}}p$
(e_1)	$E\mathcal{O}e$
(e_2)	$e \to \mathcal{O}e$
($\otimes_1$)	$E\mathcal{O}p \to E \otimes \mathcal{O}p$
($\otimes_2$)	$\otimes\mathcal{O}p \to \mathcal{O} \otimes \mathcal{O}p$
($\hat{\oplus}_1$)	$E\mathcal{O}p \wedge E\mathcal{O}q \to E(\mathcal{O}p\hat{\oplus}\mathcal{O}q)$
($\hat{\oplus}_@$)	$\mathcal{O}p\hat{\oplus}\mathcal{O}q \to \mathcal{O}(\mathcal{O}p\hat{\oplus}\mathcal{O}q)$

Before giving the completeness proof, first an explanation of the axioms is in order. The axioms $K_{\underline{\hat{\oplus}},r}$, $K_{\underline{\hat{\oplus}},r}$, $K_{\underline{\otimes}}$ and the duals are given in order for $\mathcal{L}_\sigma$ to be a normal modal logic. The axioms D_1, D_2, $U_{\underline{\otimes}}$, $U_{\underline{\hat{\oplus}}_1}$ and $U_{\underline{\hat{\oplus}}_2}$ are given for the axiomatization of the difference operator. Next, the e_1 and e_2 axioms say that e exists and that e is unique. The axioms $\otimes_1$, $\otimes_2$, $\hat{\oplus}_1$ and $\hat{\oplus}_2$ make sure that $\forall x \exists y Rxy$ and $\forall xy \exists z Czxy$.

Theorem 2.20 in [25] tells us that this axiomatization is complete for the class of all σ-structures. The next step is to define an axiomatization for the class of σ-structures adhering to the group axioms:

associativity	$\forall abc((a+b)+c = a+(b+c))$
neutral element	$\forall a(a+e = e+a = a)$
inverse element	$\forall a(a+(-a) = e)$
commutativity	$\forall ab((a+b = b+a))$

The method from [25] gives a translation to translate first order universal formulas to formulas from $\mathcal{L}_\sigma$. However, the translation assumes that the universal first order formula does not contain any nested operators. That is, the atomic formulas are of the form $x = x_1$, $x = e$, $x = x_1 + x_2$ and $x = -x_1$, where x, x_1, x_2 are variables. This assumption has no influence on the generality of the translation because every universal formula can be unnested to a new and equivalent universal formula containing no nestings.

Definition 3.2.11 Translation τ: In the following definition an *open* formula is a formula without quantifiers. A *universal* formula is a formula of the form $\forall x_1 ... \forall x_n \psi(x_1, ..., x_n)$ with $\psi(x_1, ..., x_n)$ an *open* formula. The translation τ is defined as follows, with $(x_1, ..., x_n)$ variables:

For atomic formula:

- $\tau(x = x_1) = U(\mathcal{O}x \wedge \mathcal{O}x_1)$
- $\tau(x = e) = U(\mathcal{O}x \wedge e)$
- $\tau(x = x_1 + x_2) = U(\mathcal{O}x \wedge (\mathcal{O}x_1 \hat{\oplus} \mathcal{O}x_2))$
- $\tau(x = -x_1) = U(\mathcal{O}x \wedge \otimes \mathcal{O}x_1)$

For open formulae:

- $\tau(\neg\varphi) = \neg\tau(\varphi)$
- $\tau(\varphi \wedge \psi) = \tau(\varphi) \wedge \tau(\psi)$

For universal formula:
Let $\varphi = \forall x_1 ... \forall x_n \psi(x_1, ..., x_n)$ where ψ is an open formula. Then

- $\tau(\psi) = U(\mathcal{O}x_1) \wedge ... \wedge U(\mathcal{O}x_n) \rightarrow \tau(\psi(x_1, ..., x_n))$

Before giving the translation of the universal group axioms, they must first be transformed into unnested formulae

associativity	$\forall abcx_1x_2x_3x_4(b + c = x \wedge a + x_1 = x_2 \wedge a + b = x_3 \wedge x_3 + c = x_4 \rightarrow x_2 = x_4)$
neutral element	$\forall ax(x = e \rightarrow a + x = a \wedge x + a = e)$
inverse element	$\forall ax_1x_2(x_1 = e \wedge x_2 = -a \rightarrow a + x_2 = x_1 \wedge x_2 + a = x_1)$
commutativity	$\forall abx(a + b = x \rightarrow b + a = x)$

Using the translation τ we arrive at the following modal axioms.

Definition 3.2.12 Modal group axioms: The axioms below allow us to arrive at a complete axiomatization of the group structure.

associativity	$U(\mathcal{O}a) \wedge U(\mathcal{O}b) \wedge U(\mathcal{O}c) \wedge U(\mathcal{O}x_1) \wedge U(\mathcal{O}x_2) \wedge U(\mathcal{O}x_3) \rightarrow$ $\neg U(\mathcal{O}x_1 \wedge (\mathcal{O}b \hat{\oplus} \mathcal{O}c)) \vee \neg U(\mathcal{O}x_2 \wedge (\mathcal{O}a \hat{\oplus} \mathcal{O}x_1)) \vee \neg U(\mathcal{O}x_3 \wedge (\mathcal{O}a \hat{\oplus} \mathcal{O}b)) \vee$ $\neg U(\mathcal{O}x_4 \wedge (\mathcal{O}x_3 \hat{\oplus} \mathcal{O}c)) \vee U(\mathcal{O}x_2 \wedge \mathcal{O}x_4)$
neutral element	$U(\mathcal{O}a) \wedge U(\mathcal{O}x) \rightarrow \neg U(\mathcal{O}x \wedge e) \vee U(\mathcal{O}a \wedge (\mathcal{O}a \hat{\oplus} \mathcal{O}x)) \vee U(\mathcal{O}a \wedge (\mathcal{O}x \hat{\oplus} \mathcal{O}a))$
inverse element	$U(\mathcal{O}a) \wedge U(\mathcal{O}x_1) \wedge U(\mathcal{O}x_2) \rightarrow \neg U(\mathcal{O}x_1 \wedge e) \vee \neg U(\mathcal{O}x - 2 \wedge \underline{\otimes} \mathcal{O}a) \vee$ $U(\mathcal{O}x_1 \wedge (\mathcal{O}a \hat{\oplus} \mathcal{O}x_2)) \vee U(\mathcal{O}x_1 \wedge (\mathcal{O}x_2 \hat{\oplus} \mathcal{O}a))$
commutativity	$U(\mathcal{O}a) \wedge U(\mathcal{O}x) \rightarrow \neg U(\mathcal{O}x \wedge (\mathcal{O}a \hat{\oplus} \mathcal{O}b)) \vee U(\mathcal{O}x \wedge (\mathcal{O}b \hat{\oplus} \mathcal{O}a))$

The completeness of the above described axiomatization comes from theorem 3.2 in [25], as formalized in the theorem below.

Theorem 3.2.13 *The axioms defined in Definition 3.2.12 together with the axioms defined in Definition 3.2.10 and the inference rules defined in Definition 3.2.9 are complete for the class of σ-structures that adhere to the group axioms.*

The proof of this theorem follows directly from theorem 3.2 in [25].

The next step that we want to take is to create a resolution calculus to perform automated reasoning. For a logic containing the difference operator nothing is known about creating and implementing a theorem prover. Above that, the axioms that are obtained through the above method are very un-intuitive and unwieldy. Fortunately, there is another way of extending modal logic that does allow adaptation to a resolution calculus. Namely by moving to a modal logic called hybrid logic.

3.3 Hybrid Logic

Just as modal logic extends the language of propositional logic, hybrid logic extends the language of modal logic. Modal logic is a nice formalism, but it lacks the ability to point to a specific world in a model. In order to repair this, hybrid logic was invented [13]. The difference between modal and hybrid logic lies in the nominals. A nominal is a new type of proposition, with the property that it can only be true in one world. From here on, i, j, k denote nominals. More formally, given a nominal i, $\mathcal{V}(i) = \{w\}$ for some $w \in W$. The valuation of a nominal always consists of a singleton set.

Just as in the previous modal language, the universal modality is again part of the language. However, because in hybrid logic the difference operator is not needed, the universal modality must be added to the language.

Having added the nominals, the morpho-language is redefined in the following way.

Definition 3.3.1 $\mathcal{H}(\mathcal{E})\mathcal{M}$**:** Let NOM denote the set of all nominals and ATOM = NOM ∪ PROP, then a sentence in the morpho-language is:

$$\varphi := |a|\neg\varphi|\varphi \vee \psi|e| \otimes \varphi|\varphi\hat{\oplus}\psi|E\varphi$$

With $a \in$ ATOM, $i \in$ NOM and $\psi \in \mathcal{H}(E)M$.

The morpho-language has the following semantics

Definition 3.3.2 Model A hybrid model $\mathcal{M}$ is a structure $\mathcal{M} = (W, C, R, I, \mathcal{V})$ such that $C \subseteq W \times W \times W$, $R \subseteq W \times W$ and $I \subset W$. Furthermore, $\mathcal{V}$: ATOM $\mapsto \mathcal{P}(W)$ is a valuation such that $|\mathcal{V}(i)| = 1$ for every $i \in$ NOM.

Definition 3.3.3 Morpho-semantics: Given a hybrid model $\mathcal{M}$ and a world w in $\mathcal{M}$ the morpho-semantics is:

$\mathcal{M}, w \Vdash p$	iff	$w \in \mathcal{V}(p)$, for $p \in$ ATOM
$\mathcal{M}, w \Vdash e$	iff	$(w) \in I$
$\mathcal{M}, w \Vdash \neg\varphi$	iff	$\mathcal{M}, w \nVdash \varphi$
$\mathcal{M}, w \Vdash \varphi \vee \psi$	iff	$\mathcal{M}, w \Vdash \varphi$ or $\mathcal{M}, w \Vdash \psi$
$\mathcal{M}, w \Vdash \otimes\varphi$	iff	there exists a $v \in W$ such that $(w, v) \in R$ and $\mathcal{M}, v \Vdash \varphi$
$\mathcal{M}, w \Vdash \varphi\hat{\oplus}\psi$	iff	there exists $v, v' \in W$ such that $(w, v, v') \in C$, $\mathcal{M}, v \Vdash \varphi$ and $\mathcal{M}, v' \Vdash \psi$
$\mathcal{M}, w \Vdash E\varphi$	iff	there exists a $v \in W$ such that $\mathcal{M}, v \Vdash \varphi$

For a frame to be a frame based on a group structure, the relations I, R and C must be given a proper meaning through axioms. Where in the method defined in [25] a direct translation is used, in axiomatizing the hybrid language another approach is used. Remember that there is a special class of modal formulas that define a family of frames that adhere to a specific first order property. In the following the necessary axioms are introduced that represent the desired properties.

total$_{\hat{\oplus}}$)	$Ei \wedge Ej \rightarrow Ei\hat{\oplus}j$
(unique$_{\hat{\oplus}}$)	$E(i \wedge j_1\hat{\oplus}j_2) \wedge E(k \wedge j_1\hat{\oplus}j_2) \rightarrow E(j \wedge k)$
(Ass1)	$(i\hat{\oplus}j)\hat{\oplus}k \rightarrow i\hat{\oplus}(j\hat{\oplus}k)$
(Ass2)	$i\hat{\oplus}(j\hat{\oplus}k) \rightarrow (i\hat{\oplus}j)\hat{\oplus}k$
(comm)	$i\hat{\oplus}j \rightarrow j\hat{\oplus}i$
(rev$_1$)	$\otimes i \rightarrow \underline{\otimes} i$
(rev$_2$)	$\underline{\otimes} j \rightarrow \otimes j$
(rev$_2$)	$i\hat{\oplus}(\otimes i) \rightarrow e$
(rev$_3$)	$e \rightarrow i\hat{\oplus}(\otimes i)$
(id$_1$)	$i\hat{\oplus}e \rightarrow i$
(id$_2$)	$i \rightarrow i\hat{\oplus}e$

Table 3.1: The morpho-logic axioms

We can distinguish 3 types of axioms: the axioms that define the properties for $\hat{\oplus}$, the axioms that define the properties for $\otimes$ and the axioms that define how e, $\hat{\oplus}$ and $\otimes$ cooperate. All the axioms are grouped in Table 3.1 First the axioms that define the properties of $\hat{\oplus}$. $\hat{\oplus}$ must axiomatize a binary function which is total. A function is total if it is defined on every element of it's domain. Furthermore, a binary function maps every input to a unique element in the domain. Thus we have the following two axioms:

(total$_{\hat{\oplus}}$) $\quad Ei \wedge Ej \rightarrow Ei\hat{\oplus}j$
(unique$_{\hat{\oplus}}$) $\quad E(i \wedge j_1\hat{\oplus}j_2) \wedge E(k \wedge j_1\hat{\oplus}j_2) \rightarrow E(j \wedge k)$

Furthermore, $\hat{\oplus}$ is both associative and commutative:

(Ass1) $\quad (i\hat{\oplus}j)\hat{\oplus}k \rightarrow i\hat{\oplus}(j\hat{\oplus}k)$
(Ass2) $\quad i\hat{\oplus}(j\hat{\oplus}k) \rightarrow (i\hat{\oplus}j)\hat{\oplus}k$
(comm) $\quad i\hat{\oplus}j \rightarrow j\hat{\oplus}i$

To give the proper meaning to the modality $\otimes$, it should represent a total function as well. This is captured by the following 2 axioms:

(rev$_1$) $\quad \underline{\otimes} j \rightarrow \otimes j$
(rev$_2$) $\quad \otimes i \rightarrow \underline{\otimes} i$

The axiom (rev$_1$) defines the fact that $\forall x \exists y Rxy$. That is, R is total. The second axiom defines the following property: $\forall xyz Rxy \wedge Rxz \rightarrow y = z$. As for the interplay between $\hat{\oplus}$, $\otimes$ and e we have the following axioms, inspired by the first order group axioms:

(rev$_2$) $\quad i\hat{\oplus}(\otimes i) \rightarrow e$
(rev$_3$) $\quad e \rightarrow i\hat{\oplus}(\otimes i)$
(id$_1$) $\quad i\hat{\oplus}e \rightarrow i$
(id$_2$) $\quad i \rightarrow i\hat{\oplus}e$

From here on the set of axioms (total$_{\hat{\oplus}}$) through (id$_2$), defined in Table 3.1, is denoted by Σ_{morpho}. Because the morpho-logic contains nominals, we need to redefine what is meant with completeness with respect to a family of formulas. A set of formulas is a normal hybrid logic if it adheres to the following Definition [40].

Definition 3.3.4 Normal hybrid logic: Given a similarity type τ, a normal hybrid logic over τ

contains the following axioms:

(CT)	φ, for all classical tautologies φ
(Dual$_\otimes$)	$\otimes p \rightarrow \neg\underline{\otimes}\neg p$
(Dual$_{\hat{\oplus}}$)	$p\hat{\oplus}q \rightarrow \neg(\neg p\underline{\hat{\oplus}}\neg q)$
(Dual$_U$)	$Ep \leftrightarrow \neg U\neg p$
(K$_{\underline{\otimes}}$)	$\underline{\otimes}(p \rightarrow q) \rightarrow (\underline{\otimes}p \rightarrow \underline{\otimes}q)$
(K$_{\underline{\hat{\oplus}}_r}$)	$p\underline{\hat{\oplus}}(q \rightarrow r) \rightarrow (p\underline{\hat{\oplus}}q \rightarrow p\underline{\hat{\oplus}}r)$
(K$_{\underline{\hat{\oplus}}_l}$)	$(p \rightarrow r)\underline{\hat{\oplus}}(q) \rightarrow (p\underline{\hat{\oplus}}q \rightarrow r\underline{\hat{\oplus}}q)$
(K$_U$)	$U(p \rightarrow q) \rightarrow (Up \rightarrow Uq)$
(Ref$_E$)	$p \rightarrow Ep$
(Trans$_E$)	$EEp \rightarrow Ep$
(Sym$_E$)	$p \rightarrow UEp$
(Incl$_\otimes$)	$\otimes p \rightarrow Ep$
(Incl$_{\hat{\oplus}}$	$p\hat{\oplus}q \rightarrow Ep \wedge Eq$
(Nom$_E$)	$E(i \wedge p) \rightarrow U(i \rightarrow p)$

Furthermore, it should be closed under application of the following derivation rules:

(MP)	If $\vdash \varphi \rightarrow \psi$ and $\vdash \varphi$ then $\vdash \psi$
(Nec$_{\underline{\otimes}}$)	If $\vdash \varphi$ then $\vdash \underline{\otimes}\varphi$
(Nec$_{\underline{\hat{\oplus}}}$)	If $\vdash \varphi$ then $\vdash \varphi\underline{\hat{\oplus}}\psi$ for an arbitrary formula ψ
(Nec$_U$)	If $\vdash \varphi$ then $\vdash U\varphi$
(Subst)	If $\vdash \varphi$ then $\vdash \sigma\varphi$, where σ is a substitution that uniformly replaces proposition letters by formulas and nominals by nominals.
(Name)	If $\vdash i \rightarrow \varphi$ then $\vdash \varphi$ for i not occurring in φ
(Paste$_{E_{\underline{\otimes}}}$)	If $\vdash E(i \wedge \otimes j) \wedge E(j \wedge \varphi) \rightarrow \psi$, then $\vdash E(i \wedge \otimes\varphi) \rightarrow \psi$, for $i \neq j$ and j not occurring in φ
(Paste$_{E_{L\underline{\hat{\oplus}}}}$)	If $\vdash E(i \wedge j\hat{\oplus}\xi) \wedge E(j \wedge \varphi) \rightarrow \psi$ then $\vdash E(i \wedge \varphi\underline{\hat{\oplus}}\xi \rightarrow \psi$ for $i \neq j$ and j not occuring in φ
(Paste$_{E_{R\underline{\hat{\oplus}}}}$)	If $\vdash E(i \wedge \xi\hat{\oplus}j) \wedge E(j \wedge \varphi) \rightarrow \psi$ then $\vdash E(i \wedge \xi\underline{\hat{\oplus}}\varphi \rightarrow \psi$ for $i \neq j$ and j not occuring in φ

We say that a formula φ is deducible from a set of formulas Γ, $\vdash_\Gamma \varphi$, if there exists a finite sequence of formulas, ending in φ such that each formula of the sequence either is contained in Γ, is an axiom or can be obtained from previous formulas in the sequence by applying the rules (MP), (UG), (Nec$_{\underline{\otimes}}$), (Nec$_{\underline{\hat{\oplus}}}$), (Nec$_U$), (Subst), (Name), (Paste$_{E_{\underline{\otimes}}}$) and (Paste$_{E_{L\underline{\hat{\oplus}}}}$) and (Paste$_{E_{R\underline{\hat{\oplus}}}}$).

Definition 3.3.5 $K^+_{\mathcal{H}(E)}M\Sigma$: Given a set of $\mathcal{H}(E)M$-formulas, $K^+_{\mathcal{H}(E)}M\Sigma$ is the smallest set containing the axioms defined in Definition 3.3.4 and Σ and closed under the rules defined in Definition 3.3.4. $K_{\mathcal{H}(E)}M\Sigma$ is equal to $K^+_{\mathcal{H}(E)}M\Sigma$, but not closed under the rules (Name), (Paste$_{E_{\underline{\otimes}}}$) and (Paste$_{E_{L\underline{\hat{\oplus}}}}$) and (Paste$_{E_{R\underline{\hat{\oplus}}}}$).

Definition 3.3.6 Completeness: Let S be a family of frames. A logic Λ is *strongly complete* with respect to S if for any set of formulas $\Gamma \cup \{\varphi\}$, if $\Gamma \Vdash_S \varphi$ then $\Gamma \vdash_\Lambda \varphi$. That is, if Γ semantically entails φ on S, then φ is Λ-deducible from Γ.

A logic Γ is *weakly complete* with respect to S if for any set of formulas, if $S \Vdash \varphi$ then $\vdash_\Lambda \varphi$. Λ is strongly complete (weakly complete) with respect to a single structure $\mathcal{S}$ if Λ is strongly complete (weakly complete) with respect to $\{\mathcal{S}\}$.

Before the completeness result is given note that none of the axioms contain propositional variables. In other words, all axioms contain only nominals. A formula that contains only nominals is called a *pure formula* . The fact that the completeness theorem can be given is due to the fact that all the formulas are pure.

Theorem 3.3.7 $K^{+}_{\mathcal{H}(E)}M\Sigma_{morpho}$ *is strongly complete with respect to the set of frames defined by* Σ_{morpho}.

Proofsketch. As we have already seen, all the axioms are pure formulas. One can show that every pure formula is di-persistent. Thus, given Theorem 5.3.16 in [40], which can be generalized to the hybrid arrow logic, we have that the axioms are complete for the family of frames that they define. For the proof, see Appendix D. QED

We have seen two types of axiomatization of the morphological dilation. One in modal logic and one in hybrid logic. The first axiomatization is given because the steps that are taken to go from the first order axioms to an axiomatization are very clear and understandable. However, the axiomatization gives us very long and not easy to read set of axioms. In hybrid logic the axiomatization is much more readable and intuitive. Above all, much more is known about automated theorem proving using hybrid logic.

3.4 Summary

The link between mathematical morphology and modal logic presented in [4] is a link between the translation invariant operators and arrow logic. But arrow logic itself is not strong enough to axiomatize Mathematical Morphology. To be able to do this, the difference operator has to be added to the language, together with the difference axioms and the (WIT) derivation rules.

However, automated theorem proving using the difference operator is not well known. Hence, it is more convenient to axiomatize the language using hybrid logic. Hybrid logic is an extension of modal logic using nominals. Nominals behave like propositions, with the restriction that they can be true at only one point in the model. Furthermore, the universal modality E is introduced. Due to the fact that $\hat{\oplus}$ must be a total function, the logic must be able to represent the fact that a formula must be true somewhere in the model. This can be done using the universal modality. Using hybrid logic we arrive at a complete axiomatization of mathematical morphology, creating the morpho-language.

Chapter 4

Resolution in the morpho-language

What does the link between Mathematical Morphology and modal logic add to the knowledge and understanding of Mathematical Morphology? The morpho-logic consists of a set of axioms together with a set of derivation rules. Combining these, new formulas can be generated that are valid in all the models. This means that properties of Mathematical Morphology can be generated automatically. A simple example is the fact that the dilation distributes over the $\vee$, which is a general property of every normal modal logic. The problem with using the derivation rules is that one can only *generate* formulas from the axioms. Hence, it is not obvious how to prove that a certain formula is in fact a valid formula. For this purpose several general techniques have been developed that work quite well with a lot of different logics. One of these techniques is called resolution [32] and is the focus of what follows. Other theorem proving techniques for hybrid logic exist, but to the best of the authors knowledge resolution is the most matured form of reasoning for hybrid logics. Hence this form of reasoning is chosen to pursue further.

The remainder of this chapter is organized as follows. In Section 1 resolution is explained in more detail. In Section 2 a resolution calculus is introduced for the morpho-logic. In Section 3 an implementation of the resolution calculus for the morpho-logic is discussed.

4.1 Resolution

Resolution is a method that tries to answer the question of satisfiability. More precisely, it tries to answer the question of negative satisfiability. It aims at finding a proof that shows that a formula or set of formulas is not satisfiable. This means that there is no model such that this formula or set of formulas is true in some world in the model. Resolution has first been introduced by J.A. Robinson in [32].

Using resolution, the question whether a formula φ is valid can be answered by searching for a proof that $\neg\varphi$ is not satisfiable. If the negation of a formula is always false, the formula itself must always be true and thus must be valid. Before we can introduce resolution, several basic concepts have to be recalled. Note that the definitions assume that propositional resolution is used. After propositional resolution is explained, it is extended to cope with the modalities present in modal logic.

4.1.1 Basic concepts

A resolution theorem prover takes as input a formula. This formula however, is in a special kind of form. This form is called the conjunctive normal form,

Definition 4.1.1 Conjunctive normal form: A formula φ is in *conjunctive* normal form if it is of the form

$$(\psi_{1,1} \vee ... \vee \psi_{1,m}) \wedge ... \wedge (\psi_{n,1} \vee ... \vee \psi_{n,k})$$

With $\psi_{i,j}$ either a propositional variable or a negated propositional variable. Thus, a formula is in conjunctive normal form if it is a conjunction of disjunctions. In the modal case, a formula that has the form $\Box\varphi$ or $\Diamond\varphi$ is also considered a literal because it cannot be decomposed in terms of $\vee$ and $\wedge$ any further.

Note that for every formula an equivalent formula in conjunctive normal form exists. Thus w.l.g we can assume that every formula is in conjunctive normal form. However, a formula is not represented as one formula. A formula is split into sub-formulas.

Definition 4.1.2 Clauses and literals: A formula φ is called a *literal* if it is a propositional variable. A formula φ is called a *negated variable* if it is a negated propositional literal. These literals are used to form a *clause* . A clause is a set of literals and a clause is satisfied if one of the literals is satisfied. The clauses are grouped together into a set of clauses. This set of clauses is satisfied if all the clauses in the set are satisfied.

4.1.2 Propositional resolution

The basic principal of propositional resolution, actually, of resolution in general is the following. Suppose one has two clauses. One clause of the form $\{p, q\}$ and one clause of the form $\{\neg p, q\}$. Since both clauses must be true at the same time, and the truth values of the 2 literals p and $\neg p$ are opposite one can combine both clauses while removing the opposing literals. This is due to the fact that one of p and $\neg p$ must be false, thus at least one of the other entries in one of the clauses must be true. Therefore we can ignore the formulas p and $\neg p$ and focus on the rest of the clauses. This operation gives us a clause of the form $\{q\}$. The rule that belongs to this operation is called the resolution rule and has the following form:

Definition 4.1.3 Resolution rule: Given 2 clauses $\{q_1, ..., q_i, p, q_{i+1}, ..., q_n\}$ and $\{r_1, ..., r_j, \neg p, r_{j+1}, ..., r_m\}$ with $\{q_1, ..., q_i, q_{i+1}, ..., q_n, r_1, ..., r_j, r_{j+1}, ..., r_m\}$ literals and $p \in \textsc{prop}$ we have the following rule

$$\frac{\{q_1, ..., q_i, p, q_{i+1}, ..., q_n\} \quad \{r_1, ..., r_j, \neg p, r_{j+1}, ..., r_m\}}{\{q_1, ..., q_i, q_{i+1}, ..., q_n, r_1, ..., r_j, r_{j+1}, ..., r_m\}}$$

The result of the resolution step is called the *resolvent*. *Resolving* two clauses means applying the resolution rule to the two clauses. Note that in each resolution step only one literal may be resolved. It is not allowed to resolve 2 literals in one step.

But how does one go from such a simple rule to proving whether a formula is satisfiable? The basic idea is the following. The first step of resolution is to bring all the formulas into conjunctive normal form and then creating a set of clauses. All these clauses must simultaneously be satisfied in a model for the original formula. By using the resolution rule, one tries to deduce the empty clause. Arriving at the empty clause means that there is a clause that must be satisfied but that does not contain anything that can be satisfied. Hence the set of clauses cannot be satisfied.

Example 4.1.4 For example, look at the formula $(p \to q) \wedge (q \to r) \wedge p \wedge \neg r$. Rewritten in conjunctive normal form this formula has the form $(\neg p \vee q) \wedge (\neg q \vee r) \wedge p \wedge \neg r$, giving the following clauses: $\{\neg p, q\}, \{\neg q, r\}, \{p\}, \{\neg r\}$. Applying the resolution rule, one creates the resolution tree shown in Figure 4.1

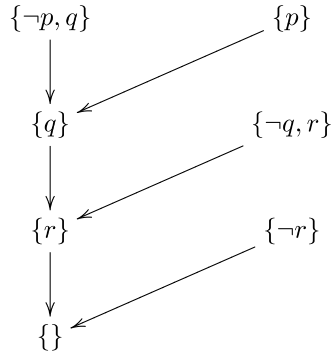


Figure 4.1: An example of a resolution proof

Definition 4.1.5 Refutation: A *refutation* of a set of clauses is a sequence of applications of the refutation rule such that the end result is the empty clause.

Resolution is used to proof the validity of a sentence in the following manner.

Definition 4.1.6 Proof: Suppose one wants to proof that a formula φ is valid. This is done by applying resolution on $\neg\varphi$. If $\neg\varphi$ has a refutation it is valid. This is denoted by $\vdash_{propres} \varphi$. If a formula φ follows logically from a set of formula Γ, one can proof this by finding a refutation for the clausal form of $\Gamma \cup \{\neg\varphi\}$. This is denoted by $\Gamma \vdash_{propres} \varphi$

To show that resolution really does the trick, recall the following proposition. For the proof we refer to [32].

Proposition 4.1.7 *Given a set of clauses* Σ*,* Σ *is propositionally unsatisfiable iff there exists a refutation of* Σ*.*

Resolution is not limited to propositional logic but can be applied to a wide array of logics including modal logic and hybrid logic. In these cases the simple resolution rule is not enough and additional rules have to be defined.

4.2 Morpho-resolution

As mentioned above, resolution is not limited to propositional logic. In [6, 23] a resolution calculus for modal logic is introduced that allows for modal resolution. Furthermore, [6] also introduces a resolution calculus for hybrid logic.

4.2.1 Hybrid resolution

With propositional resolution the resolution rule is sufficient. But in modal and hybrid logic this resolution rule is not sufficient any more. Suppose that we have two modal clauses $\{\Diamond p\}$ and $\{\Box\neg p\}$. The former clause says that there is an accessible world that makes p true while the latter clause tells us that each accessible world makes $\neg p$ true. Obviously this cannot be the case, but the resolution rule cannot be applied on these two clauses. A way of bringing the resolution "inside" is needed. One way to do this is by using labels (see [6]). Fortunately hybrid logic contains a property that enables us to bring the resolution inside in a very natural way. The nominals in the language can be used as labels of worlds.

First of all, all the formulas in a set of clauses need to be in the form $@_i\varphi$. This has no consequences for the result of the prover because if a formula $@_i\varphi$ is satisfiable, φ is satisfiable as well and vice-versa. Second, we need to transform the formulas to negation normal form.

Definition 4.2.1 Negation normal form: A formula is in *negation normal form* when only propositional letters and nominals occur in the scope of a negation.

One can transform a formula to negated normal form by using the function nf. nf is an operation that takes as input a formula and outputs an equivalent formula in negated normal form.

$$
\begin{array}{lcl}
nf(p) & = & p \\
nf(\neg p) & = & p \\
nf(\neg\neg\varphi) & = & \varphi \\
nf(\neg \otimes (\varphi)) & = & \underline{\otimes} nf(\neg\varphi) \\
nf(\neg\underline{\otimes}(\varphi)) & = & \otimes nf(\neg\varphi) \\
nf(\neg\varphi\hat{\oplus}\psi) & = & (nf(\neg\varphi))\underline{\hat{\oplus}}(nf(\neg\psi)) \\
nf(\neg(\varphi \vee \psi)) & = & (nf(\neg\varphi) \wedge nf(\neg\psi)) \\
nf(\neg(\varphi \wedge \psi)) & = & (nf(\neg\varphi) \vee nf(\neg\psi)) \\
nf(\neg @_i\varphi) & = & @_i nf(\neg\varphi)
\end{array}
$$

Given a set of clauses, Σ, we create a new set of clauses $\Sigma' = \{@_i nf(\varphi) | \varphi \in \Sigma\}$. Let us have another look at the previous clauses. They now get the form $\{@_i\Diamond p\}$ and $\{@_i\Box\neg p\}$. Formula $@_i\Diamond p$ tells us that there should be a world accessible from the world labelled by i that makes p true. We can label this new world by adding a new nominal j and the following clauses: $\{@_i\Diamond j\}$ and $\{@_j p\}$. The formula $\Box\neg p$ tells us that each world accessible from the world with label i makes $\neg p$ true. Combine this with the clause $\{@_i\Diamond j\}$ and one can see that $\{@_j\neg p\}$ must be the case as well. This brings us to the rules ($\Diamond$) and ($\Box$) in Table 4.2. One now has both $@_i p$ and $@_i\neg p$. On these two formulas the normal resolution rule can be applied, resulting in the empty clause. The normal resolution rule is represented by rule (R).

Observe that a formula of the form $@_i@_j\varphi$ is equivalent to $@_j\varphi$. This brings us to the @-rule.

In using the above introduced rules ($\Diamond$) and ($\Box$) one creates a new formula $\{@_j\varphi\}$. It is possible that φ is not in conjunctive normal form and not of the form $\Diamond\varphi$ or $\Box\varphi$. When this happens there are two possibilities. Either it is a conjunction or a disjunction. To be able to handle these situations the ($\wedge$) and ($\vee$) rules in Table 4.2 are introduced.

The rules introduced so far are sufficient to take care of the modalities and the connectives (see [6]). But what about the satisfaction operator? Consider the following situation. Suppose we have the set of clauses $\{\{@_i\Diamond j\}, \{@_j p\}, \{@_j k\}, \{@_k\neg p\}\}$. Since a nominal can point to only one world k and j must point to the same world, thus this set of clauses is not satisfiable. But with the present rules the empty clause is not derivable. What happens is that the satisfaction operator introduces a form of equational reasoning. To see why, let us translate the formula $@_i j$ to first order logic. We then get the formula $\exists x(x = i \wedge x = j)$ which can be simplified to the formula $i = j$.

In first order logic the way to deal with equational reasoning in resolution is by using paramodulation [9, 17]. In first order logic a set of clauses is said to be equationally satisfiable if resolution with the axioms shown in Table 4.1 does not produce the empty clause.

Performing resolution with these additional clauses produces a lot of unnecessary resolution steps. To solve this problem, in [17] paramodulation is introduced to replace reasoning with the symmetry and transitivity axioms. In the case of hybrid logic the paramodulation rule amounts to the (PARAM)-rule in Table 4.2. This rule basically replaces every occurrence of a nominal i with the equivalent nominal

reflexivity:	$x = x$
symmetry:	$x = y \rightarrow y = x$
transitivity:	$x = y \wedge y = z \rightarrow x = z$

Table 4.1: equational axioms

j. To be able to reason with the reflexivity axiom, the (REF) rule must be introduced. To allow reasoning with $@_i j$ if $@_j i$ occurs in the clause set, the (SYM) rule is introduced.

$$(\wedge)\ \frac{Cl \cup \{@_i\varphi \wedge \psi\}}{\begin{array}{c} Cl \cup \{@_i\varphi\} \\ Cl \cup \{@_i\psi\} \end{array}} \qquad (\vee)\ \frac{Cl \cup \{@_i\varphi \vee \psi\}}{Cl \cup \{@_i\varphi, @_i\psi\}}$$

$$(\text{RES})\ \frac{Cl_1 \cup \{@_i\varphi\} \quad Cl_2 \cup \{@_i\neg\varphi\}}{Cl_1 \cup Cl_2}$$

$$(\Box)\ \frac{Cl_1 \cup \{@_i\Diamond j\} \quad Cl_2 \cup \{@_i\Box\varphi\}}{Cl_1 \cup Cl_2 \cup \{@_j\varphi\}} \qquad (\Diamond)\ \frac{Cl \cup \{@_i\Diamond\varphi\}}{\begin{array}{c} Cl \cup \{@_i\Diamond j\} \\ Cl \cup \{@_j\varphi\} \end{array}}$$

$$(\text{PARAM})\ \frac{Cl_1 \cup \{@_i j\} \quad Cl_2 \cup \{\varphi(i)\}}{Cl_1 \cup Cl_2 \cup \{\varphi(i/j)\}}$$

$$(\text{REF})\ \frac{Cl \cup \{@_i\neg i\}}{Cl} \qquad (@)\ \frac{Cl \cup \{@_i @_j\varphi\}}{Cl \cup \{@_j\varphi\}}$$

$$(\text{SYM})\ \frac{Cl \cup \{@_i j\}}{Cl \cup \{@_j i\}}$$

Table 4.2: Hybrid rules.

The following proposition tells us that the rules presented above are sufficient to perform resolution on a hybrid logic.

Proposition 4.2.2 *A set of clauses Σ is unsatisfiable iff the closure of Σ has a refutation using the rules presented in Table 4.2.*

The proof of this proposition can be found in [6]. Here we simply give a sketch.

The idea behind the proof is that one can create a tree-structure from the original set of clauses Σ in such a way that if no empty clause occurs in the tree a model for the original set of clauses can be constructed from the tree. Furthermore, if the tree contains an empty clause this empty clause can be constructed by the use of the resolution rules for hybrid logic.

4.2.2 Resolution rules for the morpho-logic

The rules defined in Table 4.2 are sufficient for resolution on the basic hybrid logic. The morpho logic contains another set of modalities, hence we need to adjust the rules to take care of these modalities.

Furthermore, these modalities satisfy a set of axioms. These axioms must be taken into consideration as well.

Before looking at the axioms, we consider the modalities of the morpho language. The morpho language contains three modalities, the $\otimes$, the $\hat{\oplus}$ and the E. The first is a unary modality and can be taken care of by using the rules defined in Table 4.2. The $\hat{\oplus}$ on the other hand, is a binary modality and requires some new rules. The rule for the $\hat{\oplus}$ modality introduces a new relation in the same manner that the $\otimes$ rule does. The $\underline{\hat{\oplus}}$ rule on the other hand works a little differently. $\varphi\underline{\hat{\oplus}}\psi$ means that for each relation $j_1\hat{\oplus}j_2$ it is either the case that $@_{j_1}\varphi$ or $@_{j_2}\psi$. This is captured by putting both formulas in one clause and making sure that at least one of them is true. Creating two clauses would be to strong, because then both formulas should be satisfiable.

The third modality, the universal modality, has the following meaning. $E\varphi$ is true if φ is true somewhere in the model. Thus, a point must be introduced that makes φ true, meaning that if a clause of the form $\{E\varphi\} \cup C$ is found, a clause of the form $\{@_i\varphi\} \cup C$ must be created for i a new nominal. $A\varphi$ is true if φ is true everywhere in the model. Thus, if clause of the form $\{A\varphi\} \cup C_1$ is found combined with a clause of the form $\{\psi\} \cup C_2$ then a clause $\{@_i\varphi\} \cup C_1 \cup C_2$ must be formed for every nominal i that occurs in ψ.

Finally, note that $@_iE\varphi$ and $E\varphi$ are equivalent (the same holds for A). For this reason the $(@E)$ and $(@A)$ rules are introduced.

The rules for the $\otimes$, $\hat{\oplus}$, E and there respective dualities are shown in Table 4.3.

$$(A) \quad \frac{Cl_1 \cup \{A\varphi\} \quad Cl_2 \cup \{\psi\}}{Cl_1 \cup Cl_2 \cup \{@_i\varphi\}}\text{, where } i \text{ occurs in } \psi \qquad (E) \quad \frac{Cl \cup \{E\varphi\}}{Cl \cup \{@_i\varphi\}}\text{, where i is new}$$

$$(\underline{\otimes}) \quad \frac{Cl_1 \cup \{@_i\underline{\otimes}\varphi\} \quad Cl_2 \cup \{@_i \otimes (j)\}}{Cl_1 \cup Cl_2 \cup \{@_j\varphi\}} \qquad (\otimes) \quad \begin{array}{c} Cl \cup \{@_i \otimes \varphi\} \\ \hline Cl \cup \{@_i \otimes (j)\} \\ Cl \cup \{@_j\varphi\} \end{array}\text{, where j is new}$$

$$(@E) \quad \frac{Cl \cup \{@E\varphi\}}{Cl \cup \{E\varphi\}} \qquad (@A) \quad \frac{Cl \cup \{@A\varphi\}}{Cl \cup \{A\varphi\}}$$

$$(\underline{\hat{\oplus}}) \quad \frac{Cl_1 \cup \{@_i\varphi\underline{\hat{\oplus}}\psi\} \quad Cl_2 \cup \{@_i(j_1\hat{\oplus}j_2)\}}{Cl_1 \cup Cl_2 \cup \{@_{j_1}\varphi, @_{j_2}\psi\}}$$

$$(\hat{\oplus}) \quad \begin{array}{c} Cl \cup \{@_i(\varphi\hat{\oplus}\psi)\} \\ \hline Cl \cup \{@_i(j_1\hat{\oplus}j_2)\} \\ Cl \cup \{@_{j_1}\varphi\} \\ Cl \cup \{@_{j_2}\psi\} \end{array}\text{, where } j_1 \text{ and } j_2 \text{ are new}$$

Table 4.3: Resolution rules for the morpho modalities.

To incorporate the axioms a new set rules must be introduced. The reason for this is the following. The rules defined above make sure that the model that can be found by the construction used in the proof is a normal hybrid model. However, this model does not necessarily satisfy the morpho-axioms as defined in Table 3.1.

The rules that are needed are defined in Table 4.4. Next, we go over every rule and show what it's purpose is. Before we go further, note that the rules assume that there is a clause of the form $\{@_ie\}$ in the original set of clauses. This is done to make more efficient rules possible.

$$(\mathrm{Rev}_1)\quad \frac{Cl_1 \cup \{@_i \underline{\otimes} \varphi\}}{Cl_1 \cup \{@_i \otimes \varphi\}} \qquad (\mathrm{Rev}_2)\quad \frac{Cl_1 \cup \{@_i \otimes \varphi\}}{Cl_1 \cup \{@_i \underline{\otimes} \varphi\}}$$

$$(\mathrm{Rev}_{3_1})\quad \frac{Cl_1 \cup \{@_i j_1 \hat{\oplus} j_2\} \quad Cl_2 \cup \{@_{j_2} \otimes j_1\} \quad Cl_3 \cup \{@_k e\}}{Cl_1 \cup Cl_2 \cup Cl_3 \cup \{@_i k\}}$$

$$(\mathrm{Rev}_{3_2})\quad \frac{Cl_1 \cup \{@_i e\} \quad CL_2 \cup \{@_j \varphi\}}{Cl_1 \cup Cl_2 \cup \{@_i j \hat{\oplus} (\otimes j)\}}$$

$$(\mathrm{Id}_1)\quad \frac{Cl_1 \cup \{@_{j_1} e\} \quad Cl_2 \cup \{@_i j_1 \hat{\oplus} j_2\}}{Cl_1 \cup Cl_2 \cup \{@_i j_2\}}$$

$$(\mathrm{Id}_2)\quad \frac{Cl_1 \cup \{@_i \varphi\} \quad Cl_2 \cup \{@_j e\}}{Cl_1 \cup \{@_i (j \hat{\oplus} i)\}}$$

$$(\mathrm{Ass}_1)\quad \frac{Cl_1 \cup \{@_i j_1 \hat{\oplus} j_2\} \quad Cl_2 \cup \{@_{j_2} s_1 \hat{\oplus} s_2\}}{\begin{array}{c} Cl_1 \cup Cl_2 \cup \{@_i z \hat{\oplus} s_2\} \\ Cl_1 \cup Cl_2 \cup \{@_z j_1 \hat{\oplus} s_1\} \end{array}} \quad \text{with } z \in \textsc{nom} \text{ a new nominal}$$

$$(\mathrm{Ass}_2)\quad \frac{Cl_1 \cup \{@_i j_1 \hat{\oplus} j_2\} \quad Cl_2 \cup \{@_{j_1} s_1 \hat{\oplus} s_2\}}{\begin{array}{c} Cl_1 \cup Cl_2 \cup \{@_i s_1 \hat{\oplus} z\} \\ Cl_1 \cup Cl_2 \cup \{@_z s_2 \hat{\oplus} j_2\} \end{array}} \quad \text{with } z \in \textsc{nom} \text{ a new nominal}$$

$$(\mathrm{Comm})\quad \frac{Cl_1 \cup \{@_i j_1 \hat{\oplus} j_2\}}{Cl_1 \cup \{@_i j_2 \hat{\oplus} j_1\}}$$

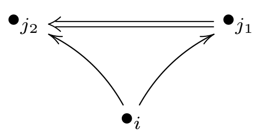
$$(\mathit{total})\quad \frac{Cl_1 \cup \{\varphi\} \quad Cl_2 \cup \{\psi\}}{Cl_1 \cup Cl_2 \cup \{E i \hat{\oplus} j\}}\ \text{, for } i \text{ occurs in } \varphi \text{ and } j \text{ occurs in} \psi$$

$$(\mathit{unique})\quad \frac{Cl_1 \cup \{@_i j_1 \hat{\oplus} j_2\} \quad Cl_2 \cup \{@_k j_1 \hat{\oplus} j_2\}}{Cl_1 \cup Cl_2 \cup \{@_i k\}}$$

Table 4.4: Additional resolution rules

The first two rules, (rev_1) and (rev_2) are defined to take care of the uniqueness and totality of the R relation. The rules are just a simple translation of the axioms rev_1 and rev_2, by replacing the arrow in the axiom with the bar in the rule.

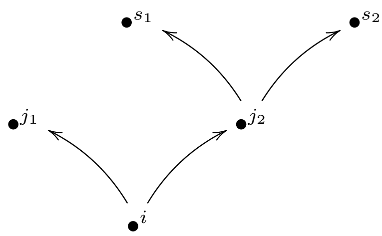
As for the rules (rev_{3_1}) and (rev_{3_2}), they are inspired by the (rev_3) axiom. (rev_{3_1}) tells us that if there are clauses of the form $\{@_i j_1 \hat{\oplus} j_2\} \cup C_1$, $\{@_{j_2} \otimes j_1\} \cup C_2$ and $\{@_k e\} \cup C_3$ then there must be a clause $\{@_i k\} \cup C_1 \cup C_2 \cup C_3$. If one is talking in terms of models, the following is the case. Suppose that there is a structure of the form



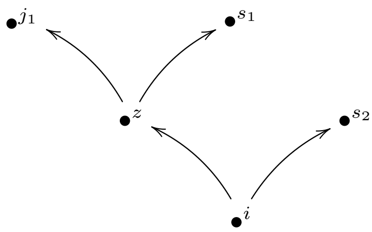
and further more there is a point in the model, named by nominal k, that satisfies e then this world is also named by the nominal i. The (rev_{3_2}) rule tells us that if there is a clause of the form $\{@_i e\} \cup C_1$ and a clause of the form $\{@_j \varphi\} \cup Cl_2$, then there must be a clause of the form $\{@_i j \hat{\oplus} (\otimes j)\} \cup C_1 \cup C_2$. In terms of models, if there is a point in the model that satisfies e, then this world is in a C relation with every other world in the model and the reverse of this world. Do note that we actually only take into account worlds of which we have some knowledge, the rest are not important for the resolution.

The rules (id_1) and (id_2) take care of the (id) axiom. The (id_1) rule tells us that if there are clauses of the form $\{@_{j_1} e\} \cup Cl_1$ and $\{@_i j_1 \hat{\oplus} j_2\}$, then there must be a clause of the form $\{@_i j_2\}$. As for the (id_2) rule, it tells us that if there are clauses of the form $\{@_i \varphi\} \cup Cl_1$ and $\{@_j e\} \cup Cl_2$ then there must be a clause of the form $\{@_i j \hat{\oplus} i\} \cup Cl_1 \cup Cl_2$.

The (ass_1) and (ass_2) axioms supply the structure that is being demanded by the associativity axiom. Each time a structure of the form



is encountered the rule (ass_1) adds one new nominal z such that the following structure emerges.



The (ass_2) rule does the same for the situation where the left side is branched.

The (comm) rule implements the commutativity axiom by switching the nominals. All the formulas that are true at either one of the end points of the branch are immediately transferred to the other one.

The last two rules make sure that C models a binary function. The first rule, (total), makes sure that if two worlds occur in a model, then they are connected. The second rule tells us that if two worlds are connected with a third world through the relation C, then this third world is the only world through which they are connected. Note that the (total) rule probably poses a big problme for the prover due to the fact that it can use it's own output to create new nominals.

The question now is whether the rules presented in Table 4.4 are adequate. In other words, is it true that if the empty clause cannot be found a model of the original clause set exists that satisfies the morpho-axioms. The answer lies in the following theorem.

Theorem 4.2.3 *The rules presented in Table 4.4, together with the rules presented for the morpho-modalities in Table 4.3 are refutationally complete with respect to the morpho axioms (i.e. given a set of clauses Σ a refutation can be found if and only if Σ is satisfiable with respect to the morpho axioms).*

The proof of this theorem lies outside the scope of this thesis, but can be found in Appendix B. Below follows a brief sketch of the proof.

The proof follows a modified version of the proof for the rules in Table 4.2 given in [7]. We begin with a set of clauses Σ. Then the closure is taken of this set of clauses and one can proof that if this set of clauses does not contain the empty clause, a model can be found who's frame satisfies all the axioms.

4.3 HyLoRes

The rules as presented in Table 4.2 are implemented using the theorem prover called HyLoRes [8]. The prover is written in Haskell, a functional programming language and uses the "Given clause" algorithm [44]. The first version of HyLoRes, version 1.0, implements precisely the rules given in Table 4.2. Version 2.0 of the prover implements an adjusted set of rules, presented in [7]. The difference lies in the generation of new nominals and the selection of a formula from a clause. This generation is restricted in such a way that the prover always converges. The proof given in Appendix B already takes this difference into account. The underlying algorithm (Algorithm 1) does not change.

```
input: init: set of clauses
var   : new, clauses, inuse, inactive: set of clauses
var   : given:clauses

clauses := {};
new := init;
simplify(&new, inuse∪inactive∪clauses);
if {} ∈new then
|  return unsatisfiable
end
clauses := computeComplexity(new);
while clauses ≠ {} do
|  given = select(clauses);
|  clauses = clauses - {given};
|  while subsumed(given, inuse) do
|  |  if clauses = {} then
|  |  |  return satisfiable
|  |  else
|  |  |  given = selectclauses;
|  |  |  clauses = clauses - {given};
|  |  end
|  end
|  simplify(&inuse, given);
|  new = infer(inuse, given, &inactive);
|  if {} ∈new then
|  |  return unsatisfiable
|  end
|  clauses = clauses∪computeComplexity(new);
end
```

The function *simplify* performs subsumption deletion and the function *computeComplexity* determines the complexity of each clause through some predefined function. This complexity is used by the function *select* to pick a given clause, but after five times the oldest clause in *clauses* is chosen to prevent starvation. *infer* applies the resolution rules to the given clause and each clause in *inuse*: if the $\wedge, \vee$ or $\diamond$ are applied the given clause is added to *inactive* because applying the rule again to the same clause is redundant.

Algorithm 1: Given clause algorithm [8]

Intuitively, the algorithm works as follows. As input it receives the set of clauses of which satisfiability has to be tested. These clauses are stored in the variable *new*. After that the complexity of all the clauses is computed and they are put into the variable *clauses*. The loop starts by selecting a clause from *clauses* using a selection function. On this clause all the rules that can be applied are applied and the results are put into *new*. Note that some of the rules are binary or ternary and thus require two or three inputs. This second or third input is found in the *inuse* variable. After all the rules have been applied it is checked whether the empty clause has been derived. If this is the case then the original set of clauses is unsatisfiable. If this is not the case the complexity of the new clauses is computed and they are put into *clauses*.

4.3.1 Implementation

The prover is implemented in Haskell as an extension of HyLoRes. The program consists of several modules of which the most important ones are:

- HyLoRes: this module is the main module of the program. It takes care of the initialisation and reads in the set of clauses that is used as input. Furthermore, it initiates the main loop of the program.

- Resolve: this module contains the main loop. It selects a clause from *clauses* and finds the rules that can be applied on the clause. After application it collects the results and based on that it decides the action to be taken next.

- Formula: this module contains all the functions and data structures that are used to represent and work with formulas.

- Rules: this module contains all the rules that can be applied on the given clause

- ClauseRepository: this module contains all the clauses that are being used in the program. It contains the data structure that contains the *new*, *clause* and *inuse* data sets.

Let us informally describe how the program operates. When starting the program, the HyLoRes module initialises the variables and creates the instances of the data structures that keep track of the formulas and clauses that will be generated during the process. After the initialisation, HyLoRes starts the Resolve module that contains the main loop. In this main loop, each time a clause is picked from *clauses*. This is done via a selection function that can be selected in the initialisation period. Every sixth loop, the oldest clause is picked to prevent starvation. The selection procedure can look at the number of formulas in the clause, number of literals appearing in the clause, the maximal depth of the formulas and the minimum prefix level of the formulas in the clause.

After the clause is selected, the formula in the clause that will be considered for resolution is selected via a pre-specified selection function. Using this formula, Resolve decides which rules can be applied on the formula. For each application of a binary rule, the set of clauses that are potential candidates for resolution are selected from the InUse repository. While applying the rule it is decided whether a formula is an actual candidate for application.

After application of all the rules Resolve checks whether the empty clause has been derived. If this is the case the program is terminated. If not the newly generated clauses are put into *clauses*. it is checked, however, whether the newly generated clause is subsumed by another clause. If this is the case the clause is deleted, because it will not add any new information.

After all the newly created clauses are processed, the loop starts all over again. Do note that if there are no clauses left in *clauses*, the process terminates.

4.4 HyLoMorphRes

Hylores is designed to handle unary modalities. However, the morpho-language contains both a binary modality, a universal modality and a constant. Furthermore, the morpho-logic contains several axioms that need to be taken into account in proving theorems. This means that in adjusting the prover to cope with the morpho-logic, the following adjustments need to be made:

- the support for a binary modality has to be implemented,
- the support for the unary modality has to be implemented,
- the rules for the new binary modality have to be implemented,
- the rules for the axioms presented in Table 4.4 have to be implemented.

4.4.1 Adding support for the binary modalities

The original software was designed for the unary modalities $\Diamond$ and $\Box$. In adjusting the implementation of HyLoRes to fit the requirements for the morpho-language, this unary modality is chosen to represent the $\otimes$ and $\underline{\otimes}$ modalities. This leaves us with the binary modalities $\hat{\oplus}$ and $\underline{\hat{\oplus}}$ and the universal modalities A and E.

The first step in implementing the support is to adjust the HyloParse.hs and HyloLex.hs modules that process the input of the program. The HyloParse module is created using the parser generator Happy (*http://www.haskell.org/happy*). Adding new tokens for the $\hat{\oplus}$, $\underline{\hat{\oplus}}$, A and E operators is sufficient.

The module HyloLex uses the parsed tokens to create the data structures in which the formulas are captured. This is done in a case statement and adding a new case for the new tokens is sufficient. The functions that are used to create the data structures, and the definition of the data structure Formula are found in the module *Formula.hs*. This module also contains the code for calculating the order of formulas and the code for creating new formulas and nominals from old formulas. Hence, this module needs to be adjusted as well.

Every formula belongs to a certain family. For example, the formula $@_i\underline{\otimes}j$ belongs to the family of relations Relnm. Adding two new modalities means that five new families must be added. These new families are PRelnm, AtPBox, AtPDia, Ef and Uf. Finally, the module PrettyPrint.hs needs to be adjusted so that the correct information can be given to the user.

As for adding the rules, this must be done by adjusting *Rules.hs* and *RuleMetadata.hs*. *Rules.hs* contains the implementation for the different rules and *RuleMetadata.hs* contains some additional data that is needed for the implementation in *Rules.hs*.

In implementing the rules, one rule was needed for the PBox family and one for the PDia family. Furthermore, two rules needed to be implemented in order for the paramodulation to be effective. For the universal modality four more rules needed implementation.

All the rules are applied in Resolve.hs, the module that contains the main loop. This is done by a function that checks what family the selected formula is in and according to that information applies a number of rules on this formula. Thus the appropriate families must be added to this function and furthermore, it must be defined which families of formulas can be used as second argument to the binary rules. This last part must be done in the module ClauseRepository.hs.

4.4.2 Adding support for the additional resolution rules

Adding support for the binary relations is only one part of what must be done. The additional rules that take care of the extra requirements of the morpho-language must be implemented as well.

This means that Rule.hs, RuleMetadata.hs, Resolve.hs, ClauseRepository.hs and Formula.hs must be changed.

The first thing that one can notice is that some of the rules require more that two arguments. In Resolve different handlers are used when unary or binary rules are needed, hence a new handler must be implemented that can handle the application of a ternary rule.

Second, in Rules.hs the ternary rule must be introduced as a new type of rule. After these prerequisites are met, the rules can be implemented. This also means adjusting Formula.hs because that is the only place in which formulas can be manipulated and new formulas can be created. The actual contents of the type Formula is not known outside of this module.

It is not necessary to keep track of the application of rules, because every time a new clause is created this is recorded by the program and hence a clause will only occur once in the set *clauses*.

4.4.3 Summary

Several methods exists to automatically prove whether a certain formula or set of formulas is valid in a certain logic. We have chosen the method of resolution to create a theorem prover that is able to prove theorems for the morpho-logic. For this purpose a new resolution calculus is created. This calculus is implemented in an existing theorem prover, called HyLoRes. The prover has been extend in such a way that it can now cope with binary modalities, the universal modality and the extra requirements for the morpho-logic.

Chapter 5

Preliminary Evaluation

In this chapter we draw preliminary conclusions regarding the performance of the implementation described in Chapter 4. The evaluation of the prover is divided into two parts. First, we investigate the interaction between the different rules. Sometimes the output of a rule can directly be used by another rule. We are interested what the impact of this is on the performance. Second, we investigate the overall complexity of the prover i.e., we try to see what the influence of the complexity of the input is on the performance of the prover. In this setting, with complexity we mean the modal depth of the formulas. Because HyLoResMorph is, to our knowledge, the only prover that is capable of proving theorems of the morpho language it is not possible to compare the performance of the prover with other provers.

The remainder of this chapter is structured as follows. In Section 1, the experimental setup is defined and the evaluation measures are explained. In Section 2, the experimental results are presented and in Section 3, the results are discussed.

5.1 Experimental setup

Two aspects of the prover are evaluated in the experiments. The first part of the experimentation focuses on the interaction between the individual rules. The second part focuses on the overall performance of the prover.

As for the first part of the experimentation, the interaction between the rules is tested in the following manner. In running the prover, one can define which rules are to be activated by the prover and which rules are not to be activated by the prover. For example, one can define whether the commutativity rule is used by the prover or not. Belonging to each rule is a formula that corresponds to the frame property that the rule must enforce. First, the performance of each individual rule is tested by applying the prover on the accompanying formula, with the addition that only the rule to be tested can be used by the prover. Second, the prover is instructed to use all the rules defined in Table 4.4. Then again the prover is used to prove the validity of the formulas belonging to the different individual rules. Finally, the prover is instructed to use all rules but the (total) rule to again prove the validity of the individual axioms. This last test is done because we believe that the total rule generates a wide range of unnecessary clauses, which affects the runtime.

To measure the performance, the cpu-time that the prover takes to find an answer is measured. Each measurement is repeated 10 times and the average of the 10 measurements is taken as the measure of evaluation. The results have been obtained from experiments on a Dell computer containing an Intel Pentium 4 3,00 GHz processor and 1 GB of memory running Fedora 3.

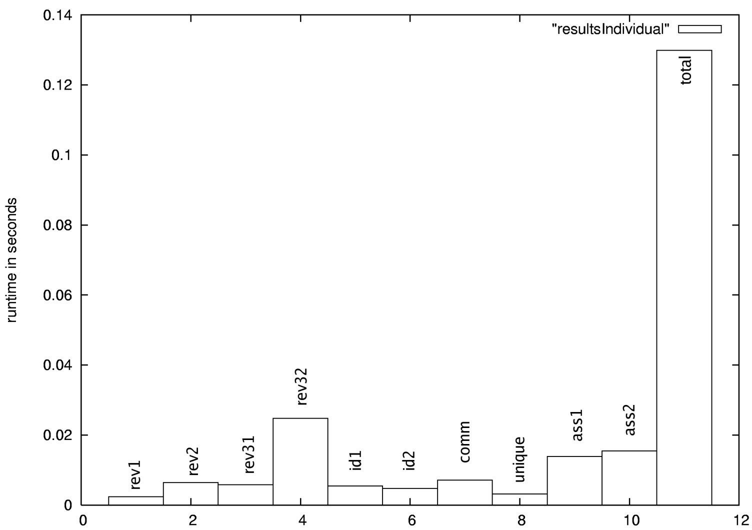
The second part of the experimentation consists of testing the overall performance of the prover. This is done by giving the prover formulas of different complexity. With complexity the depth of the formulas is meant, i.e., the number of modalities that occur in the formula. For the test, 16 formulas of increasing modal depth have been generated. Four formulas of depth 0, four of depth 1, four of depth 2 and four of depth 3. As a measure of evaluation the cpu-time that the prover needs to find an answer is measured. Due to the possible non-termination a time-out of 15 minutes in cpu-time has been put in place.

5.2 Results

First, we provide the results of the experiments to evaluate the interaction between the axioms. Next the results for the overall performance of the prover are presented.

5.2.1 Interaction

Figure 5.1: The individual rules.



The results of the three different tests are presented in the Figures 5.1, 5.2 and 5.3. On the vertical axes the cpu-time is shown. Each bar represents one of the axioms belonging to the rules. One can see in Figure 5.1 that the prover needs more time to prove the validity of the (total) axiom than to prove the other axioms. Another thing one can observe is that if all the rules are turned on (see Figure 5.2), the overall time needed to find an answer increases as well, but not uniformly. For example, the time needed to compute an answer for the rev32, id1 and comm axioms increases significantly more than the time needed to compute an answer for the other axioms. The time needed to find an answer for the ass1, ass2 and total axioms increases so much that they are of the charts. To illustrate this, the time needed to compute an answer for the (total) axiom is approximately 200 seconds. As for the ass1 and ass2 axioms, an out of memory message is given before an answer is reached. Figure 5.3 shows that turning off the (total) rule reduces the needed time significantly, except for the id1 and comm axioms. Figure 5.3 also shows that the id1 and comm rules need significantly more time compared to

Figure 5.2: The interaction between all the rule.s

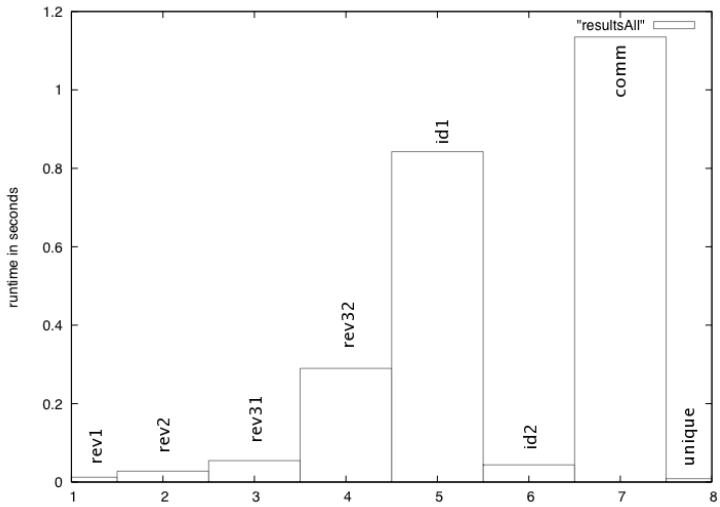
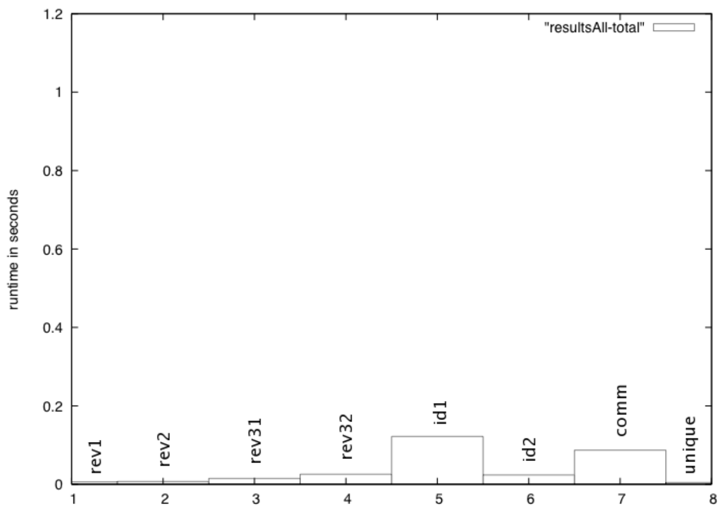


Figure 5.3: The interaction between all the rules except the (total) rule.



the situation shown in Figure 5.1. The situation for the ass1, ass2 and total rules is the same as in Figure 5.2.

5.2.2 Overall performance

16 formulas have been used in the test. These formulas are reported in Appendix C. Of these 16 formulas only 3 formulas gave an answer within a second. In proving the other formulas, the prover reached the 15 minute timeout before an answer could be given. The results are given in Table 5.1

Modal Depth	Formula	elapsed time
Modal Depth 0	Formula 1	Time out
	Formula 2	$3.0e^{-3}$ s
	Formula 3	Time out
	Formula 4	Time out
Modal Depth 1	Formula 1	Time out
	Formula 2	Time out
	Formula 3	$5.5992e^{-2}$ s
	Formula 4	Time out
Modal Depth 2	Formula 1	4.252354 s
	Formula 2	Time out
	Formula 3	Time out
	Formula 4	Time out
Modal Depth 3	Formula 1	Time out
	Formula 2	Time out
	Formula 3	Time out
	Formula 4	Time out

Table 5.1: Overall performance results

5.3 Discussion

Clearly, performance is an issue for a prover such as HyLoRes. One can discern two different problems. The first problem lies in the fact that if a set of clauses is unsatisfiable the prover does not terminate. The second problem is the explosion of time-consumption if all the rules are used simultaneously.

The first problem is due to the total rule. The total rule takes as input formulas of the form φ and ψ. Each nominal i that occurs in φ is combined with a nominal j that occurs in ψ to create the formula $Ei \oplus j$. This new formula then creates a new nominal, on which the total rule can again be applied. A loop unrolls that does not stop unless the empty clause is found. Figure 5.1 is a good illustration of this. As one can see, the prover needs far more time to prove the total axiom than it needs to prove another axiom. The results of the overall performance test also illustrate the non-termination. The three formulas that did give an answer where satisfiable, the other thirteen formulas where not. Therefore, they would never stop if the timeout would not have been placed.

The second problem is due to the interaction between several rules. In this interaction many clauses are generated that are not needed for the actual proof. This over-generation of clauses makes it harder for the prover to find the proper clauses that are needed for the proof and has several causes. We pinpoint the causes by looking at the Figures 5.2 and 5.3. The first observation that one can make is that the performance shown in Figure 5.2 is about ten times as low as the performance shown in Figure 5.3. The only difference is that in Figure 5.3 the total rule is not used. Thus the total rule is a big cause of over-generation.

The second observation that one can make is that the rev32, id1 and comm axioms see a greater increase in needed time than the other axioms (not looking at ass1, ass2 and total). To see which rules are actually used in the proof we turn our attention to Tables 5.2, 5.3 and 5.4. In these tables the rules that are used in the corresponding proofs are shown, together with the number of times each rule is invoked. One can see that the rev32, id2, comm, ass1 and ass2 rules are the rules that have the most invocations. Looking at the specific rules one can see why this is the case.

First, we note that the rev32 and id2 rules take a formula of the form $@_j\varphi$ and a formula of the form $@_ie$. They respectively create formulas of the form $@_ij \oplus (\otimes j)$ and $@_ji \oplus j$. The former formula can be decomposed in $@_i j \oplus s$ and $@_s \otimes j$, enabling the application of the comm rule, and again the id2 rule and the rev32 rule. After this, we have the formulas $@_i j \oplus s$ and $@_s i \oplus j$, which can be used by the ass1 rule, which introduces a new nominal z and a new formula of the form $@_z \varphi$. On this formula again id2 and rev32 can be applied, starting the loop all over again.

Looking at the Tables 5.2, 5.3, 5.4 one must also note that the usage of the total rule has little effect on the usage of the other rules. However, because the total rule can "feed" itself, i.e., it can indirectly be applied on its own result, thus yielding a high number of applications.

We end by making the observation that the runtimes that are given in Table 5.1 are increasing. Although no graph is made, one can see that the needed time increases very fast. Where a formula of depth 0 needs a millisecond, a formula of depth 1 needs ten milliseconds and a formula of depth 2 needs 4 seconds. Without generalizing, one might say that it is possible that the time-consumption could grow exponentially with the complexity of the formula.

Summarizing, the main problem in the interaction between the rules is the creation of new nominals. Thus, if one wants to improve the performance of the prover, the creation of new nominals must be controlled further. For example, note that in the application of the ass1 rule on formulas of the form $@_i j_1 \oplus j_2$ and $@_{j_2} s_1 \oplus s_2$, new formulas of the form $@_i z \oplus s_2$ and $@_z j_1 \oplus s_1$ are created. On these two formulas the ass2 rule can be applied, creating $@_i j_1 \oplus z_1$ and $@_{z_1} s_1 \oplus s_2$. By application of the unique rule z_1 and j_2 must point to the same world, thus if one can make the calculus recognize this situation one can stop the generation of a wide range of unnecessary rules.

Rules	applications (all without total)	applications (all)
R_Box rule:	5	5
R_PBox rule:	1	1
R_PDia rule:	8	9
R_Par rule:	33	48
R_ParRel rule:	1	1
R_Conj rule:	1	1
R_Rev1 rule:	3	3
R_Rev2 rule:	3	4
R_Rev31 rule:	4	5
R_Rev32 rule:	58	74
R_Id1 rule:	8	12
R_Id2 rule:	34	40
R_Comm rule:	16	26
R_Ass1 rule:	2	4
R_Ass2 rule:	3	6
R_Udia:	-	18
R_Uniq:	-	1
R_total rule:	-	646

Table 5.2: Rule application for the rev32 axiom

5.4 Summary

The theorem prover discussed in chapter 4 is tested. The results from these tests show that the main performance problem of the prover is due to the (total) rule. Although this prover is far from perfect, it is a first step in finding a way of automated reasoning with the morpho-logic.

Rules	applications (all without total)	applications (all)
R_ResP rule:	1	1
R_Box rule:	7	6
R_PDia rule:	13	10
R_Par rule:	35	20
R_ParPRel rule:	2	-
R_Conj rule:	1	1
R_Rev1 rule:	6	6
R_Rev2 rule:	6	6
R_Rev31 rule:	7	5
R_Rev32 rule:	85	65
R_Id1 rule:	12	9
R_Id2 rule:	63	54
R_Comm rule:	33	21
R_Ass1 rule:	39	23
R_Ass2 rule:	56	55
R_Udia:	-	16
R_Uniq:	6	-
R_total rule:	-	1068

Table 5.3: Rule application for the Id1 axiom

Rules	applications (all without total)	applications (all)
R_ResP rule:	2	2
R_Box rule:	6	7
R_PBox rule:	7	6
R_PDia rule:	7	7
R_Par rule:	15	8
R_Conj rule:	1	1
R_Rev1 rule:	6	6
R_Rev2 rule:	6	6
R_Rev31 rule:	6	5
R_Rev32 rule:	58	52
R_Id1 rule:	11	9
R_Id2 rule:	55	55
R_Comm rule:	23	17
R_Ass1 rule:	32	32
R_Ass2 rule:	53	73
R_Udia:	-	22
R_Uniq:	4	1
R_total rule:	-	1509

Table 5.4: Rule application for the Comm axiom

Chapter 6

The Morpho-Language Landscape

6.1 Introduction

The Morpho-language defined in Chapter 3 is able to talk about Mathematical Morphology's basic constituents: dilation and erosion. Just as Mathematical Morphology is more than simply applying the dilation and erosion, the morpho-language can express more than just the dilation and erosion. Most applications of MM are build up out of several dilations and erosions with several specific structuring elements. One group of these applications are the filters. Filters are operations on an image that, the name already gives it away, filter out some information captured in the image. There are several classes of images, namely binary, grey scale and color images. The morpho-language can be applied to all of them but for reasons of simplicity we will only look at the filters for binary images. We are interested in filters that we can express in the morpho-language. That is, if we can express the filters in the language, we can reason with them and we can automatically see what the properties of the filters are. It turns out that not all the filters can be expressed in the morpho-language. Therefore we will also try to see what extensions of the language will be needed to be able to express these filters

Filters are only one interesting application of MM. In images a lot of spatial information is stored. Therefore we could also consider the intuitive spatial relations between regions and the role that MM and the morpho-language can play in finding and representing these relations. Qualitative Spatial Reasoning (QSR) deals with the formalization of space. It tries to find a formal language that is suited to represent spatial concepts and can be used to reason with these concepts. MM is a formalism that searches for geometric properties of shapes. These geometric properties can be used in QSR. It would be interesting to see to what extend the morpho-language is suited for QSR and which spatial concepts can be expressed in the language.

The remainder of this chapter is structured as follows. In Section 2, we first explain what filters are and what different properties they have. We identify which filters can be represented in the morpho-language and which cannot. According to this we identify the extensions that we need in order to be able to express the remaining filters. In Section 3, we recall the basic concepts behind QSR, what has already been done in the field and what role MM has already played in it, which spatial relations can be expressed in the morpho-language and what extensions need to be introduced to express further concepts such as nearness. Finally, in Section 4 we suggest a formalization for the extensions overviewed in the present morpho-language landscape.

6.2 Binary filters

Morphological filters are operations on images that filter out certain information contained in the image. For example, openings and closings are very basic filters that can filter out irregularities in an image by removing regions that cannot contain a certain structuring element, respectively filling holes that are smaller than a certain structuring element. Openings and closings are useful because when an image is recorded, noise could cause certain unconnected regions to be connected or certain connected regions to be unconnected. In this context, a region is connected if for each set of points (x, y) in the region, there is a path from x to y that does not leave the region.

More formally, a filter f is an operator $f : V \mapsto V$ that takes as input an element of V and returns an element of V. In the case of binary images, $V = \mathcal{P}(\mathbb{R}^2)$. Filters can be classified by using the following properties:

- **translation invariant**: A filter f is translation invariant if, given a translation t, for all A $f(t(A)) = t(f(A))$. That is, first translating an image and than applying the filter yields the same result as first applying the filter and than translating the result.
- **increasing**: A filter f is increasing if for all A and B, given the fact that $A \subseteq B$, $f(A) \subseteq f(B)$. For example, dilation and erosion are both increasing operators.
- **extensive**: A filter f is extensive if for all A, $A \subseteq f(A)$. In other words, the resulting region is always bigger than the original region.
- **anti-extensive**: A filter f is anti-extensive if for all A, $f(A) \subseteq A$. In other words, the resulting region always lies inside the original region.
- **idempotence**: A filter f is idempotent if applying the filter twice yields the same result as applying the filter once. More formally, for all A, $f(f(A)) = f(A)$.

In [22] a set of binary filters is presented. This set of binary filters is a set that is widely used in practice in the fields of medical imaging, mineralogy and OCR. We only look at filters presented in [22].

6.2.1 Expressible filters

Some filters are expressible in the morpho-language. The constituents of the morpho-language are, besides the dilation and erosion, the reverse operator, the Boolean complement, the intersection and the union. The question is which filters in [22] are expressible in the morpho-language. All the expressible binary filters are collected in Table 6.1. We divide the filters in translation invariant filters and filters that are not translation invariant.

Translation invariant filters

Let us first consider the opening and closing. The opening and closing are used to filter out noise from the image. What kind of noise is filtered depends on the structuring element that is used. The opening has the form

$$A \circ B = (A \hat{\ominus} B) \hat{\oplus} B$$

The closing has the form

$$A \bullet B = (A \hat{\oplus} B) \hat{\ominus} B$$

Where $A \hat{\ominus} B = \neg(\neg A \hat{\oplus} \otimes B)$. The opening is extensive, idempotent and increasing. The closing also is idempotent and increasing but anti-extensive.

From the opening and closing the opening top-hat and closing top-hat can be constructed. These two filters give what is removed in the opening or added in the closing respectively. The opening top-hat has the form $A\hat{\circ}B = A - A \circ B$. If we translate this to the morpho-language we get

$$A\hat{\circ}B = A \wedge \neg(A \circ B)$$

The closing top-hat has the form $A\hat{\bullet}B = (A \circ B) - A$. If we translate this to the morpho-language we get

$$A\hat{\bullet}B = (A \bullet B) \wedge \neg A$$

The opening top-hat is anti-extensive, but the closing top-hat has non of the above mentioned properties.

There are three types of filters that give a boundary of a shape. These are the internal boundary $A - (A \ominus B)$, the external boundary $(A \oplus B) - A$ and the morphological gradient $(A \oplus B) - (A \ominus B)$. In the morpho-language we can express these filters in the following manner. The internal boundary is equivalent to the expression

$$A \wedge \neg(A\hat{\ominus}B)$$

the external boundary to the expression

$$(A\hat{\oplus}B) \wedge \neg A$$

and the morphological gradient to the expression

$$(A\hat{\oplus}B) \wedge \neg(A\hat{\ominus}B)$$

As for the properties of these filters, the internal boundary is anti-extensive. The other boundary operators do not have any of the properties listed in Section 6.2.

The next operator we express in the morpho-language is the hit-or-miss transform. This operator is mainly used to filter out objects that do have certain properties but lack others. The form of the hit-and-miss transform is $A \circledast T = ((A \ominus E) - (\bar{A} \ominus F))$ where $T = (E, F)$ and $E \subset \bar{F}$. The fact that E is a subset of $\bar{F}$ means that E and F are disjoint. The expression

$$A \circledast T = ((A\hat{\ominus}E) \wedge \neg(\bar{A}\hat{\ominus}F)) \wedge U(E \rightarrow \neg F)$$

where $T = (E, F)$ is the morpho-language equivalent of this filter. T is called the hit-or-miss template. The function of $U(E \rightarrow \neg F)$ is to capture the fact that E must be a subset of $\bar{F}$.

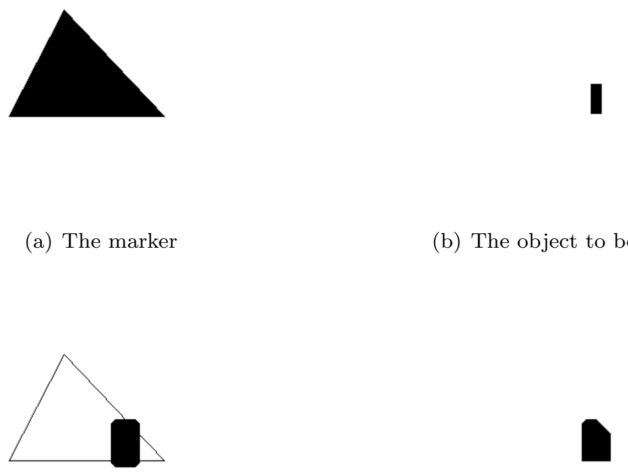
Using the hit-or-miss transform we can create the thinning filter. The thinning filter takes a shape and removes the outer layer of the shape. The exact form of this outer layer is dependent on the structuring element that is being used. In MM, thinning is defined as $A \odot T = A - (A \circledast T)$. In the morpho-language this definition is equivalent to

$$A \odot T = A \wedge \neg(A \circledast T)$$

Sequential thinning is the successive application of a sequence of thinnings. Given a set of hit-or-miss templates, $T_1, T_2, ..., T_n$, sequential thinning is $(((A \odot T_1) \odot T_2) ... \odot T_n)$.

Non translation invariant filters

Interestingly, only two expressible filters are not translation invariant. These two filters are the conditional dilation and conditional erosion. Both are increasing operators and have the form $A \oplus_C B = (A \oplus B) \cap C$ and $(A \ominus_C B) = (A \ominus B) \cap C$ respectively. The goal of these two operators is to restrict the range of the operator. Where the result of the normal dilation/erosion can range over the entire $\mathbb{R}^2$, the range of the conditional dilation/erosion will be in C. This is illustrated in Figure 6.1. In the morpho-language these operators get the following form:



(a) The marker (b) The object to be dilated

(c) The dilation and the marker together (d) The conditional dilation

Figure 6.1: The conditional dilation

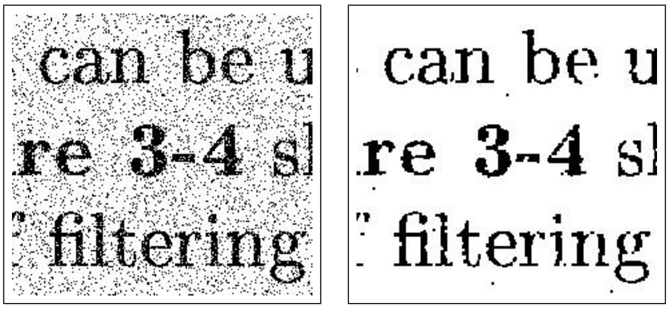
- conditional dilation: $(A \hat{\oplus}_C B) = (A \hat{\oplus} B) \wedge C$
- conditional erosion: $(A \hat{\ominus}_C B) = (A \hat{\ominus} B) \wedge C$

filter	MM	MM-Logic
opening $A \circ B$	$(A\hat{\ominus}B)\hat{\oplus}B$	$(A\hat{\ominus}B)\hat{\oplus}B$
closing $A \bullet B$	$(A\hat{\oplus}B)\hat{\ominus}B$	$(A\hat{\oplus}B)\hat{\ominus}B$
opening top-hat $A\hat{\circ}B$	$A - (A\hat{\ominus}B)\hat{\oplus}B$	$A \wedge \neg((A\hat{\ominus}B)\hat{\oplus}B)$
closing top-hat $A\hat{\bullet}B$	$(A\hat{\oplus}B)\hat{\ominus}B - A$	$(A\hat{\oplus}B)\hat{\ominus}B \wedge \neg A$
internal boundary	$A - (A\hat{\ominus}B)$	$A \wedge \neg(A\hat{\oplus}B)$
external boundary	$(A\hat{\oplus}B) - A$	$(A\hat{\oplus}B) \wedge \neg A$
morphological gradient	$(A\hat{\oplus}B) - (A\hat{\ominus}B)$	$(A\hat{\oplus}B) \wedge \neg(A\hat{\ominus}B)$
hit-or-miss transform $A \circledast T$	$((A\hat{\ominus}E) - (\bar{A}\hat{\ominus}F))$	$((A\hat{\ominus}E) \wedge \neg(\bar{A}\hat{\ominus}F)) \wedge U(E \rightarrow \neg F)$
thinning $A \odot B$	$A - A \circledast B$	$A \wedge \neg(A \circledast B)$
conditional dilation $A\hat{\oplus}_C B$	$(A\hat{\oplus}B) \cap C$	$(A\hat{\oplus}B) \wedge C$
conditional erosion $A\hat{\ominus}_C B$	$(A\hat{\ominus}B) \cap C$	$(A\hat{\ominus}B) \wedge C$

Table 6.1: all expressible filters.

6.2.2 Non expressible filters

The list of filters that has been discussed above is not an exhaustive list. There exist more morphological filters that can be applied on binary images. Unfortunately, these filters cannot be expressed



(a) Picture with noise

(b) After the salt and pepper filter has been used

Figure 6.2: The salt and pepper filter.

in the morpho language.

Translation invariant filters

It turns out that there is only one translation invariant filter described in [22] that is not expressible in the morpho-language. This is the Alternating Sequence Filter (ASF). Informally, the ASF is a filter that alternates the openings and closings. Each time a combination of opening and closing is performed the structuring element is enlarged by dilating it with the original structuring element. The ASF has the form $ASF_B^n(S) = ((((S \bullet B) \circ B) \bullet 2B) \circ 2B)... \bullet nB) \circ nB)$. nB is a shortcut for $(((B \oplus_1 B) \oplus_2 B)... \oplus_{n-1} B)$ in which $\oplus_m$ tells us that it is the m^{th} application of the dilation. Just as the opening and closing are used to filter out noise, the ASF is used to filter out noise. One nice example is the salt-n-pepper filter, see Figure 6.2. It filters out both white and black noise. In order to represent this filter in the morpho-language, we need to be able to count the number of applications of the dilation.

Non translation invariant filters

The size-n geodesic dilation and erosion are two filters that serve as building blocks for further filter applications. Their forms are $S \oplus_T^n B$ and $S \ominus_T^n B$ respectively. $\oplus_T^n$ means that the conditional dilation is performed n times.

Using these two operations we can reconstruct parts of an image with the so called inf-geodesic and sup-geodesic reconstruction:

- inf-geodesic reconstruction: $T \triangle_B S = (S \oplus_T^\infty B)$
- sup-geodesic reconstruction: $T \bigtriangledown_B S = (S \ominus_T^\infty B)$

The S works as a marker and T is the picture from which we want to reconstruct. What one does is select several points in an image (the marker), and find the connected components of these points by successively dilating them. Since we restrict the outcome to the original picture, at some point the dilations will have no more effect. In theory the dilation is performed infinitely, but in practice one can stop when there are no further changes. These reconstructions assume that the marker is given.

It is also possible to find the markers in the image automatically. The following 3 filters do exactly that.

- reconstructive opening: $A \circ_E B = A \triangle_E (A \circ B)$
- reconstructive opening top-hat: $A \hat{\circ}_E B = A - (A \circ_E B)$
- reconstructive closing: $A \bullet_E B = (A \bullet B) \bigtriangledown_E A$

In the case of the reconstructive opening, the marker is found by opening the original image A with a structuring element B. This filter can be used to detect shapes with certain properties in an image. The reconstructive opening top-hat can be used to delete shapes with certain properties from an image. The reconstructive closing is a connected operator. In short this means that it makes the image coarser. It deletes all the regions in the complement of the image that are not fully filled by the closing.

All the reconstructions are based on the size-n geodesic dilation and erosion. What needs to be added to the morpho-language is therefore the ability to **count** the number of occurrences of a conditional dilation/erosion.

On can also combine several reconstructive openings/closing. This way one creates a reconstructive τ-opening. What one does is, given a set of structuring elements $\{B_1, ...B_n\}$, find the reconstructive opening of all these B_i with the same image. $\{B_1, ..., B_n\}$ is called the base. This results in the following operation:

$$\bigcup_{i=1}^{n} A \circ_E B_i$$

The reconstructive radial opening is a special instance of such a reconstructive τ-opening. The base is a set of linear structuring elements of varying angle. This brings us to the following property that would be nice to have. The ability to say something about the shape of the structuring elements. Or, more generally, to be able to specify properties of regions. For example, one might want to define a sphere or a directional element.

Two other filters that need the ability to specify the shape of a region are the bounded dilation and erosion. Both are a special instance of the conditional dilation and erosion respectively in which the condition is the region of the image S, denoted by $V[S]$.

In summary, there are filters that cannot be expressed by the morpho-language. If we want to incorporate them we need to add to the language the ability to count the number of applications of a conditional dilation or erosion and we should have the ability to specify properties of a region. For example, we want to be able to say that a region denotes a circle, or that it is convex.

6.3 Qualitative Spatial Reasoning

In our every day life we make use of spatial reasoning. We use it to navigate through a room, we use it to recognize objects and so on. The field of Qualitative Spatial Reasoning (QSR) deals with the formalization of spatial reasoning. In order to reason about space one needs to have a language in which one can represent spatial concepts and an inference mechanism. Finding an appropriate language has been the mayor focus of QSR in the last decades. For an overview of spatial reasoning see [19] and [36].

One of the most influential QSR languages has been the RCC (Region Connected Calculus) from [31] and the RCC-8 calculus defined in [30]. In Figure 6.3 the eight different relations are shown. One can see them as the spatial equivalent of the Allen relations ([5], [41]) for time. The RCC-8 language is based on the notion of connection. In [20] three definitions of connection are given. Just to illustrate the concept of connection (see also Figure 6.4) we give one of these definitions:

- $C(x, y) \Leftrightarrow x \cap c(y) \neq \emptyset$ or $c(x) \cap y \neq \emptyset$

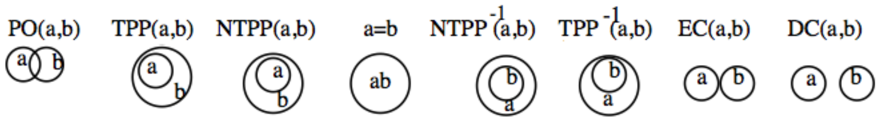


Figure 6.3: The RCC-8 relations

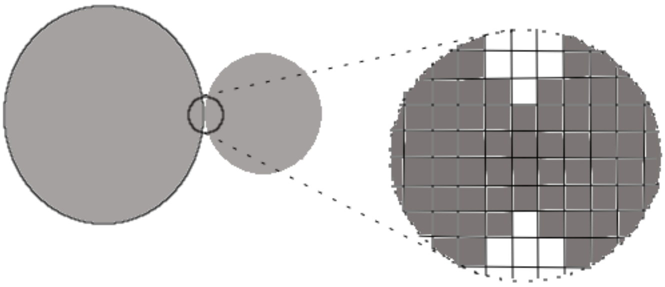


Figure 6.4: The notion of connectedness

Here $c(x)$ denotes the closure of the region x. This definition gives a set-theoretical notion of connectedness. Using the notion of connectedness, one can define relations shown in Table 6.2

$DC(x,y)$	$\neg C(x,y)$
$P(x,y)$	$\forall z[C(z,x) \rightarrow C(z,y)]$
$PP(x,y)$	$P(x,y) \cap \neg P(y,x)$
$x = y$	$P(x,y) \cap P(y,x)$
$O(x,y)$	$\exists z[P(z,x) \cap P(z,y)]$
$PO(x,y)$	$O(x,y) \cap \neg P(x,y) \cap \neg P(y,x)$
$DR(x,y)$	$\neg O(x,y)$
$EC(x,y)$	$C(x,y) \cap \neg O(x,y)$
TPP	$PP(x,y) \cap \exists z[EC(z,x) \cap EC(z,y)]$
$NTPP(x,y)$	$PP(x,y) \cap \neg \exists z[EC(z,x) \cap EC(z,y)]$
$P^{-1}(x,y)$	$P(y,x)$
$PP^{-1}(x,y)$	$PP(y,x)$
$TPP^{-1}(x,y)$	$TPP(y,x)$
$NTPP^{-1}(x,y)$	$NTPP(y,x)$

Table 6.2: the RCC relations

The $DC(x,y)$ predicate is true if two regions are disconnected. In the set-theoretic case this amounts to x and y having no points in common. $P(x,y)$ is true if the region x is a part of region y. Do note that, according to the definition, it is still possible that $x = y$. The $PP(x,y)$ predicate is true if x is a proper part of y. In other words, if $PP(x,y)$ holds, then $x \subset y$. The meaning of $=$ is obvious. Two objects are the same if they are part of each other.

The predicate $O(x, y)$ is true if x and y have points in common. In other words, x and y overlap. $PO(x, y)$ defines the partial overlap, hence if $PO(x, y)$ holds $x \neq y$ is true. The $DR(x, y)$ predicate is true if two regions are distinct. They do not overlap.

The $EC(x, y)$ predicate is true if x and y are externally connected. In other words, they touch each other. The $TPP(x, y)$ predicate states that x is a tangential proper part of y. In other words, x touches the edge of y while lying inside y. $NTPP(x, y)$ is true if x does not touch the edge of y and lies inside y. Finally, the $P^{-1}, PP^{-1}, TPP^{-1}$ and $NTPP^{-1}$ predicates are the inverse of P, PP, TPP and $NTPP$ respectively.

The RCC-8 is a first order language. In [10] modal logic is used to encode the tractable fragment of the RCC relations. It turns out that one can also define the RCC-8 relations using the notion of interior. The interior of a region consists of all the points that are not connected, in the topological sense, with the background. From [37] and [38] it is known that the modal logic S4 has a topological interpretation. In this interpretation the $\Box$ denotes the interior of a formula and the $\Diamond$ denotes the closure of a formula. In [10] the RCC-8 relations are defined using the S4 modal logic with the topological interpretation.

The modal logic S4 is a normal modal logic, see Chapter 2, extended with the following axioms

T $\Box\varphi \rightarrow \varphi$

4 $\Box\varphi \rightarrow \Box\Box\varphi$

Building on the S4 logic, in [45] a logic of metric and topology is formed. The topological part of the logic is created by incorporating the S4 logic from [10]. The metric part of the logic is handled by the distance logic defined in [27] in which the modality $\forall^{\leq a}$ is introduced with the following semantics:

$$\mathcal{M}, x \Vdash \forall^{\leq a}\varphi \text{ iff } \mathcal{M}, y \Vdash \varphi \text{ for all } y \text{ s.t. } d(x, y) \leq a$$

The distance measure $d(x, y)$ should satisfy the following axioms

- $d(x, y) = 0$ if $x = y$
- $d(x, z) \leq d(x, y) + d(y, z)$
- $d(x, y) = d(y, x)$

These axioms define a metric. As already stated in [2], $\forall^{\leq a}\varphi$ using the Euclidean metric is equivalent to dilating φ with a ball of size a.

In [2] another notion of connectedness is used to define the RCC-8 relations. The idea behind it is that by taking a specific structuring element C one defines the connectedness. One then says that two regions x and y are connected if $(x\hat{\oplus}C) \cap y \neq \emptyset$. For example, if looking at the $\mathbb{R}^2$, one takes the unit circle as C. A region now is connected to all the space that lies one unit outside the region. This way another modal logic of RCC is defined, this time using the dilation as it's primitive.

But these RCC-8 relations are not the only link between QSR and Mathematical Morphology. In [16] Mathematical Morphology is used to define the notion of betweenness. They look into several definitions of betweenness, both crisp and fuzzy, and see what notion is most appropriate for what situation. In [34] a notion of convex-hull is defined using specific dilations and erosions. The convex-hull of a region A is the region one gets if you close A under the condition that for each combination of points (x, y), the line combining x and y must be contained in the region. Examples of convex regions are circles and rectangles. Informally, one can get the convex-hull of a region A by stretching an elastic string around the object and releasing it, see Figure 6.5. The region surrounded by the string defines the convex-hull of A. in [2] several other links between QSR and Mathematical morphology are given.

(a) Concave object (b) The convex hull

Figure 6.5: The convex hull

6.3.1 RCC-8

Spatial relations as defined in the RCC-8 calculus can be expressed in the Morpho-language. For this we use the the dilation as a primitive to model a form of connectedness. One can say that two objects A and B are C-connected if $(A \oplus C) \cap B$ is not empty. In this expression, C defines the connectivity. This means that, given a point x all the points that are covered by placing the origin of C over x are connected to x. For example C could be a 1×1 region, a diamond region or even a circle. Using this idea, we can define the RCC-8 relations in the morpho-language as shown in Table 6.3

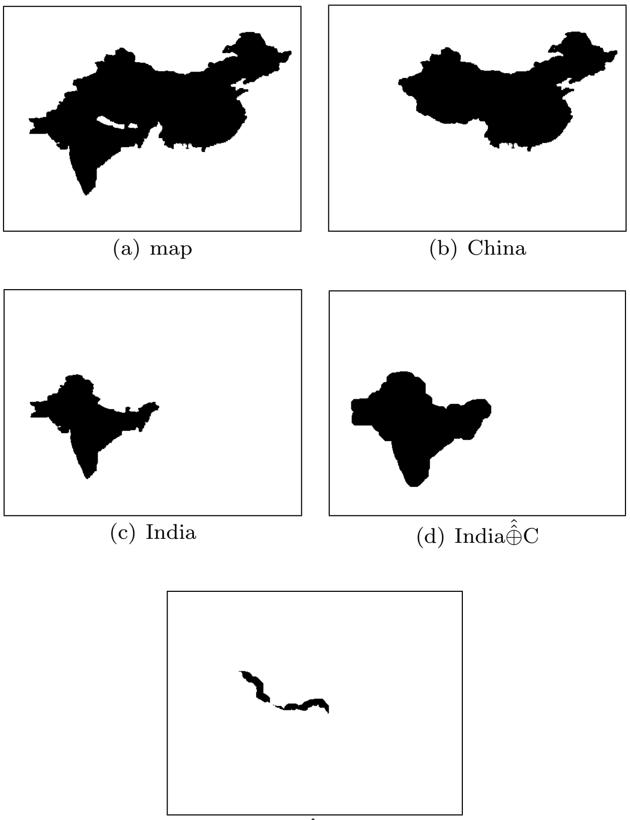
$DC(x, y)$	$U\neg(x \wedge y)$
$EC(x, y)$	$E(((x\hat{\oplus}C) \wedge y))$
$PO(x, y)$	$\neg U\neg(x \wedge y) \wedge \neg U(x \to y) \wedge \neg U(y \to x)$
$x = y$	$U(x \leftrightarrow y)$
$TPP(x, y)$	$U(x \to y) \wedge E(x \wedge y) \wedge \neg U(y \to x) \wedge \neg U((x\hat{\oplus}C) \to y)$
$NTPP(x, y)$	$U(x \to y) \wedge E(x \wedge y) \wedge \neg U(y \to x) \wedge U((x\hat{\oplus}C) \to y)$
$TPP^{-1}(x, y)$	$U(y \to x) \wedge E(x \wedge y) \wedge \neg U(x \to y) \wedge \neg U((y\hat{\oplus}C) \to x)$
$NTPP^{-1}(x, y)$	$U(y \to x) \wedge E(x \wedge y) \wedge \neg U(x \to y) \wedge U((y\hat{\oplus}C) \to x)$

Table 6.3: The RCC-8 relations in the morpho language

Incidentally, there is a difference between the definitions in [2] and the definitions above. In [2] one needs to add that a formula is consistent. In the morpho-language this is captured by the U-modality, which states that if $U\varphi$ holds, φ is true every where in the model. In Figure 6.6 it is shown how the dilation can be used to find the EC relation in a picture containing two objects.

6.3.2 Further Spatial Concepts

So far we have introduced a language for representing and reasoning with certain spatial relations. But there are more spatial relations that might be useful to represent.



(a) map (b) China

(c) India (d) India$\hat{\hat{\oplus}}$C

(e) (India$\hat{\hat{\oplus}}$C)$\wedge$China

Figure 6.6: Finding the EC* relation between China and India.

Relative size

Using the concept of relative size, one can state that one region is larger or smaller than another region. One way of defining this is saying that a region, say A, is smaller than a region B if there exists a translation t and a rotation r such that translating A with t, creating A_t and rotating A_t with r, creating A_{tr}, $A_{tr} \subseteq B$.

One thus needs two separate concepts, translation and rotation, to properly define the concept of relative size. In mathematical morphology one can define the former but not the latter. Namely, one can translate a region by dilating it with a singleton. Thus, for objects having the same shape and orientation, we can define the notion relative size in the morpho-language in the following manner

Definition 6.3.1 Relative size: given two regions x and y, x smaller than y, denoted by $ST(x, y)$ if and only if the following is true

$$Ei \wedge (TPP(x, y) \vee NTPP(x, y))$$

The use of TPP and $NTPP$ is due to readability, and should be replaced with the corresponding morpho-language formulas when used.

Metric

Although a metric is a quantitative property of space, it is important to be able to talk about distances between objects. In mathematical morphology one can measure distances between objects by dilating with for example the unit circle C. The distances between two objects is then equivalent to the number of dilations that is needed to reach the other shape. If we denote the distance between regions A and B with $d(A, B)$, then $d(A, B) = n$ iff $(A \oplus^n C) \cap B \neq \emptyset$ and $(A \oplus^{n-1} C) \cap B = \emptyset$. In order for one to be able to represent this, one must be able to count the number of dilations. In order for d to be a metric it should also adhere to the axioms for a metric above.

Nearness

Besides the quantitative distance measure, one can also consider a relative distance measure. Namely the concept of nearness as defined in [41]. It is captured by the predicate $n(x, y, z)$. If $n(x, y, z)$ holds, it means that the distance between x and y is smaller than the distance between x and z. The main ingredient is the definition of relative distance. Informally, the concept of nearness can be defined using dilation in the following manner. Given a connectivity C, dilate x with C until you reach either y or z. If y is reached first we have $n(x, y, z)$. If z is reached first we have that $n(x, z, y)$. In Figure 6.7 shows the situation where $n(A, B, C)$ holds.

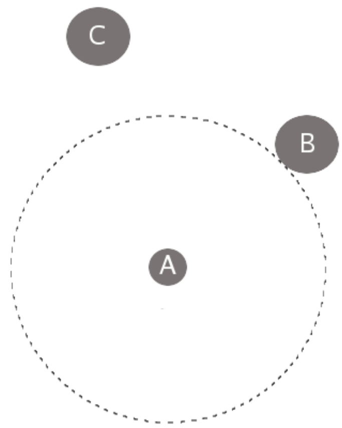


Figure 6.7: Nearness: object B is closer to A than C

Therefore in the morpho-language we need a way of controlling the number of dilations that are being applied. Again, we see that the ability to count the number of applications of a dilation or erosion is necessary. Using the notation from the previous section, we define nearness as follows:

Definition 6.3.2 Nearness Given regions A, B, C and structuring element S, one can define that A is closer to B than to C as follows:

$$\exists n \text{ s.t. } \neg U((A \hat{\oplus}^n S) \wedge C) \wedge U((A \hat{\oplus}^n S) \wedge B)$$

From the definition we remark the need for a way to quantify over the number n.

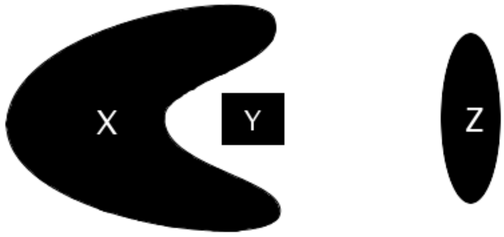


Figure 6.8: An example of where the above given definition of betweenness in terms of dilations fails.

Betweenness

Using this concept of nearness one can define a form of betweenness as done in [4]. What we want is a relation $B(x, y, z)$ that is true if y is between x and z. One way to look at this, is by saying that y is in between x and z if $n(x, y, z)$ and $n(z, y, x)$. We thus can define the concept of betweenness in the following manner

$$B(x, y, z) \text{ iff } n(x, y, z) \wedge n(z, y, x)$$

In the end, what one wants is a definition that captures the intuitive notion of between. Unfortunately the above definition does not give an intuitive notion of betweenness. A good example of where the above definition fails is shown in Figure 6.8.

Convex-hull

In [34], a method is described to find the convex hull of a shape using the morphological dilation and erosion. What is done is the following. One takes a set of half planes with a certain orientation. For each of these half planes one calculates the closing of the shape with this half plane. In the end one calculates the intersection of all these half planes. If one needs to express the convex hull, one needs the ability to define properties of shapes. In this case we need to be able to define the fact that a region is a half plane, and that it has a certain direction.

In summary, the concepts we need to add to our language in order to be able to express more spatial concepts are the following. We need the ability to *count* the number of applications of a dilation or an erosion and the number of applications of conditional dilations and erosions. We need to be able to *quantify* over numerical variables and we need to be able to *define geometric properties* of regions.

6.4 Extending the morpho-language

From the overview that we just presented, we conclude that there are three properties that need to be added to the language. These are the ability to count the number of applications of a specific dilation, the necessity of quantifying over the number denoting the number of dilations and the use of regions with specific geometric properties.

6.4.1 Counting

The ability of counting is necessary both for the representation of a large group of morphological filters, for enabling the definition of relative distance and for giving the possibility of defining a metric on the underlying space. By the following notation

$$A \oplus^n B$$

we represent the dilation of A with B repeated n times. For example, $A \oplus^3 B$ is equivalent to $(((A \oplus B) \oplus B) \oplus B)$. One can look at $\oplus^n$ as a new modality, but then the fact that $A \oplus^n B$ is equivalent to $A \oplus B^n$ where B^n denotes that B is dilated with itself n times tells us that it's behavior is as the usual dilation. Hence the same axioms that apply to the usual dilation also apply to $\oplus^n$. Furthermore, one must define the interaction between different n. This could be done by adding the following axioms:

- $A \hat{\oplus} B \rightarrow A \hat{\oplus}^n B$ for every $n > 1$
- $A \hat{\oplus}^n B \rightarrow A \oplus^m B$ for all n, m s.t. $n < m$

Note that adding these axioms implies that $e \rightarrow B$. Whether this follows from the axioms we do not know.

The counting ability significantly increases the expressive power and allows for the representation of many more filters. Interestingly, this opens a new issue of completeness for the language which is beyond the scope of the present treatment. The question now arises if this is enough for defining both relative distance and metric distance.

The answer to the first part of the question, concerning relative distance, is positive. The only property a relative distance should have is transitivity. If B is closer to A than C, and C is closer to A than D, it should be the case that B is closer to A than D. This is the case with Definition 6.3.2 of relative distance. Suppose that $n(A, B, C)$ and that S is the structuring element. That means that there is an n such that $A \oplus^n S \cap B$ is nonempty and $A \oplus^n S \cap C$ is empty. Furthermore, suppose that $n(A, C, D)$, which gives us that there is an m such that $A \oplus^m S \cap C \neq \emptyset$ and $A \oplus^m S \cap D = \emptyset$. This gives us that $A \oplus^n S \rightarrow A \oplus^m S$. Towards a contradiction suppose that $n(A, D, B)$. Thus there must be an l such that $A \oplus^l S \cap D \neq \emptyset$ and $A \oplus^l S \cap B = \emptyset$ which gives us that $A \oplus^m S \rightarrow A \oplus^l S$. From this we can conclude that $A \oplus^n S \rightarrow A \oplus^l S$. However, then it cannot be the case that $A \oplus^l S \cap B = \emptyset$.

For metric distance however it is not yet clear whether these axioms are enough. Further research must be conducted to find an answer to this question.

6.4.2 Specifying geometric properties of a region

Sometimes a concept or filter can only be expressed in mathematical morphology if one has the ability to specify certain geometric properties of a region. Since a structuring element is nothing more than a region in space, specifying geometric properties of a region allows one to specify geometric properties of a structuring element.

Most work on spatial reasoning has been done in First Order Logic. When one wants to define a property of a region one defines a new predicate and axioms to make sure that the predicate has the intended meaning. For example, in [21] first the notion of congruence is defined by some axioms. Two regions are in the congruence relation, if they have the same shape. That is, one can translate and rotate one shape such that it perfectly fits on the other. Using this relation and the notion of part-hood, the concept of a sphere is defined using to the following definitions

- $\mathrm{CGOP}xyz \equiv \exists x'(\mathrm{CG}xx' \wedge \mathrm{O}x'y \wedge \mathrm{O}x'z)$

- $x \preceq y \equiv \forall z_1 z_2[CGOPxz_1z_2 \rightarrow CGOPyz_1z_2]$[1]
- $x \prec y \equiv x \preceq y \wedge \neg(y \preceq x)$
- $\text{MAXCGOP}x \equiv \forall y[\text{PP}xy \rightarrow x \prec y]$
- $\text{CGOSUM}xy \equiv \forall v[(\text{CG}vx \wedge \text{O}vx) \rightarrow \text{P}vy] \wedge \neg\exists v[\text{P}vy \wedge \forall w[(\text{CG}wx \wedge \text{O}wx) \rightarrow \text{DR}wv]]$
- $\text{S}x \equiv (\text{MAXCGOP}x \wedge \exists y[\text{CGOSUM}yx])$

Just as in the RCC language, this language uses the connectivity as a basic predicate, from which O, overlap, is defined. The congruence primitive is denoted by CG. With CGOP, a congruence overlapping part is defined. CGOP(xyz) holds if there is a shape congruent to x that overlaps with y and z. The MAXCGOP defines the CGOP that has a maximal surface. Finally, the CGOSUM predicate holds if y is the sum of all regions that both overlap and are congruent with x.

What the axioms do is that they create a hierarchy of regions such that all the regions where the maximum distance between two points in the region is the same are at the same level. They then define the maximal element of this group. The CGOSUM predicate can only hold if x is connected, and hence x should be a sphere. The proof can be found in [11].

In modal logic one cannot use predicates. Furthermore, Modal Logic is inherently local. One can add global operators that can look to the entire model, but it is not possible to single out a specific part of the model. Hence, there is reason to believe that it is not possible to define properties of regions in the morpho-language. Thus, in order to incorporate the ability to specify properties of regions one must look at First Order Logic.

6.4.3 Quantification over the number of dilations

In the definition of nearness one needs the ability to quantify over the number of applications of a dilation. Just as in the previous section this goes beyond the expressive power of modal logic. In First Order Logic however, quantification is a rather important part of the language, and hence it is reasonable to believe that one must again look at First Order Logic to incorporate the notions that one wants to express.

6.5 Summary

We have seen that certain filters are expressible in the morpho-language, but also that there are filters that are not expressible. Furthermore, we have seen that we can express the RCC-calculus in the morpho-language. Other spatial concepts like nearness and convexity are not expressible in the basic morpho-language. We have identified which operators are needed for expressing certain spatial concepts and creating given morphological filters. Above all, the ability to count the number of applications of an operator is needed.

[1] In [21] the x and y are switched, as can be seen from [11] this is a typo

Chapter 7

Conclusion

The main purpose of this thesis is to investigate the link between Mathematical Morphology and logic, especially with respect to Qualitative Spatial Reasoning. This investigation can be subdivided into two main themes. First the link between Mathematical Morphology and logic must be formalized. Second, the consequences and applications of the link must be investigated.

To formalize the link between Logic and Mathematical Morphology we develop a new hybrid logic language that models the behaviour of the morphological dilation and erosion. In investigating the consequences and applications of this link a new language to perform QSR is defined and a resolution calculus is created.

In Chapter 3 we have seen that the basic modal language is not powerful enough to axiomatize the semantics that belongs to the morpho-logic. It cannot express the notion of a singleton. Two ways are discussed in which the modal language can be enriched in such a way that a singleton can be defined. First, the difference operator can be added to the language. Using this difference operator a method is given by [25] that automatically gives a complete axiomatization of the language. Second, the modal language can be extended by adding nominals and the universal modality. Again a complete axiomatization is given.

The latter method has been used in the rest of the thesis due to the higher usability, mainly in the realm of automated theorem proving.

In investigating the consequences and applications we have focused on two aspects. First, we focused on the automated reasoning capabilities of the morpho logic. We created a resolution calculus and implemented this in the HyLoRes theorem prover. Unfortunately, we found that the performance of the prover as it is, is not sufficient to actually use it in real life applications. More research is needed to explore possible improvements of the prover.

Second, we looked at the expressive power of the language. We found that there is a group of morphological filters that can be expressed in the morpho language. However, there also exists a large family of filters that is not expressible in the language presented in this paper. Some extensions of the language are proposed, but no formalizations are given.

More satisfactory is the application of the morpho-language to the field of QSR. First, several spatial notions that cannot be represented in the morpho-language are discussed. It is analysed what is needed to define these concepts. Second, it is discussed which notions can be expressed.

Due to the geometric application of mathematical morphology it is interesting to see what properties of space, more specifically of the regions inside a space, could be captured by the morpho logic. We found that the concepts expressed in the RCC calculus can also be expressed in the morpho logic. Even better, the morpho logic is more expressible that the RCC calculus in that it can also express a weak notion of relative size.

In the introduction of this thesis the following question was asked: "How does the link help Mathematical Morphology and can it help us in reasoning about space?"

The second part of the question has been answered by introducing the RCC-8 encoding in the morpho-language. This result has also been published in [1]. The answer to the first part of the question can be further subdivided into two parts. First, the morpho-logic can be used to analyse properties of Mathematical Morphology. Several properties could be described in the morpho-language. These descriptions can then be used to test whether some morphological operations have certain properties. However, we have seen that not all morphological operators can be represented by the morpho-language. Hence the analysis would not be complete and universally applicable. The second part of the answer has to do with the application area of Mathematical Morphology. By presenting a spatial language based on the morphological operator, both in this thesis and in [15], it has been shown that morphology can help interpreting pictures in a spatial manner. We have seen that several spatial concepts can be expressed in the morpho-language itself, but, since Mathematical Morphology is richer more spatial properties can be represented. It is thus interesting to see whether Mathematical Morphology is useful, not just as an image processing technique, but also as a technique that is helpful in analysing the spatial content of an image.

Appendix A

Algebra: main definitions

In this appendix some basic algebraic concepts that are used throughout the thesis are discussed. Algebra is commonly viewed as the study of algebraic structures. For a thorough review of the algebraic concepts discussed see [18].

Definition A.0.1 Algebraic structure An algebraic structure is a set $\mathcal{L}$ together with a collection of operators upon this set $\mathcal{L}$. An $n-$ary operation f on $\mathcal{L}$ is a function that takes n elements of $\mathcal{L}$ and returns a single element of $\mathcal{L}$.

The operators of an algebraic structure can be defined as adhering to certain axioms. For example, a binary operator $+$ is called associative if $\forall x, y, z \in \mathcal{L}\,, x + (y + z) = (x + y) + z$. One example of an algebraic structure whose nature is limited by several axioms is a group.

Definition A.0.2 Group A structure $(A, *, ^{-1}, e)$ is called a group if A is a non-empty set and $*$ is a binary operator $A \times A \rightarrow A$, $^{-1}$ a unary operator on A and e a constant satisfying the following axioms

- associativity: for all $a, b, c \in A$ we have that $a * (b * c) = (a * b) * c$
- For all $a \in A$ we have that $a * e = e * a = a$
- For all $a \in A$ we have that $a * a^{-1} = a^{-1} * a = e$. Where a^{-1} is called the inverse of a.

A structure that on top of these axioms also satisfies

$$\forall x, y\, x + y = y + x$$

is called a commutative (abelian) group.

Besides defining operators on a set, the elements inside the set can be related with each other. The standard way of doing this is by defining an order on the set. For example take the set of natural numbers $\mathcal{N}$, then $<$ is a natural ordering on the set of natural numbers. More formally, an order is defined as follows.

Definition A.0.3 Partial order and partially ordered sets
Consider a set $\mathcal{L}$. A binary relation $\leq$ on $\mathcal{L}$ is called a partial order relation if it has the following properties

- reflexive: for any $a \in \mathcal{L}$, $a \leq a$

- antisymmetric: for any $a, b \in \mathcal{L}$, if $a \leq b$ and $b \leq a$ then $a = b$
- transitive: for any $a, b, c \in \mathcal{L}$, if $a \leq b$ and $b \leq c$ then $a \leq c$.

We call $(\mathcal{L}, \leq)$ a partially ordered set.

Using the ordering on a set one can define an upper bound and a lower bound of an arbitrary subset. For example take the set $A = \{0, 1, 2, 3, 4, 5\}$. It is obvious that $B = \{1, 2, 3\}$ is a subset of A. The upper bound of B is 3 and the lower bound is 1.

This example might give the impression that finding an upper bound or a lower bound is easy because it is always contained in the set. However, this is not always the case. To see this, look at the following example. Define a set $A = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}\}$ with the subset relation as it's ordering. Furthermore, take $B = \{\{2\}, \{3\}\}$ a subset of A. In this case the upper bound of B is $\{2, 3\}$ and the lower bound of B is $\{\}$. This brings us to the following definition of an upper bound and a lower bound and the notions of greatest lower bound and smallest upper bound.

Definition A.0.4 Upper bound and lower bound

Given some $l, u \in \mathcal{L}$ and $K \subseteq \mathcal{L}$ we say that l is a lower bound of K if for every $k \in K$ we have that $l \leq k$. We say that u is an upper bound if for every $k \in K$ we have that $k \leq u$.

Definition A.0.5 Infimum and supremum

We define the infimum as the greatest lower bound and supremum as the least upper bound. The supremum of K is denoted by $\bigvee K$, the infimum of K by $\bigwedge K$.

Using the previous definitions we now define the concept of a complete lattice:

Definition A.0.6 Complete Lattice

We will say that the partially ordered set $(\mathcal{L}, \leq)$ is a complete lattice if every non-void subset K of $\mathcal{L}$ has a supremum and an infimum.

Some examples of complete lattices are:

- The powerset of a given set with the usual inclusion relation. The supremum is given by the union, the infimum by the intersection.
- The set $\overline{\mathbb{R}} = \mathbb{R} \cup \{+\infty, -\infty\}$ with the usual order $\leq$. The infimum is the lowest number, the supremum the largest. Note that $\mathbb{R}$ by itself does note denote a lattice, because the set $\mathbb{R}$ has no lower of upper bound at all.

A complete lattice [12] has two elements that are of great importance. These are the so called universal bounds O and I. I is the greatest element and O is the least element. That is, for each $a \in \mathcal{L}$ we have that $O \leq a \leq I$. Furthermore, we have that $O = \bigvee \emptyset$ and $I = \bigwedge \emptyset$.

We have the following well-known characterization of complete lattices(theorem 3 [12]):

Proposition A.0.7 *Let $(\mathcal{L}, \leq)$ be a partially ordered set. Then the following three statements are equivalent:*

- *$\mathcal{L}$ is a complete lattice*

- *$\mathcal{L}$ has a least element O and every subset of $\mathcal{L}$ has a supremum*
- *$\mathcal{L}$ has a greatest element I and every subset of $\mathcal{L}$ has an infimum*

In the algebraic theory of mathematical morphology the concept of a dual has an important place. Every lattice has a so called dual. For example, given a complete lattice $\mathcal{L}$ together with an ordering $\leq$, it is known that the partially ordered set $(\mathcal{L}, \geq)$ is a complete lattice as well. Furthermore, we have that for any $K \subseteq \mathcal{L}$

$$\bigvee_{(\mathcal{L},\geq)} K = \bigwedge_{(\mathcal{L},\leq)} K$$

$$\bigwedge_{(\mathcal{L},\geq)} K = \bigvee_{(\mathcal{L},\leq)} K$$

The universal bounds O and I are interchanged. Thus, the dual lattice of $(\mathcal{L}, \leq)$ is $(\mathcal{L}, \geq)$. The duality principle tells us that for every property, definition or statement on $(\mathcal{L}, \leq)$ there is a dual one on $(\mathcal{L}, \geq)$, but with O, I, $\bigvee$ and $\bigwedge$ interchanged.

The last concept that needs to be explained is the concept of an isomorphism. An isomorphism can be used to compare structures with each other.

Definition A.0.8 Isomorphism

Given two complete lattices $(\mathcal{L}, \leq)$ and $(\mathcal{L}', \leq)$, an isomorphism from $\mathcal{L}$ to $\mathcal{L}'$ is a bijection $\varphi : \mathcal{L} \to \mathcal{L}'$ such that for any $X, Y \in \mathcal{L}$, $X \leq Y$ if and only if $\varphi(X) \leq \varphi(Y)$. A function φ is a bijection if, for every Y there is at most one X s.t. $\varphi(X) = Y$ (injective) and for all Y there is at least one X s.t. $\varphi(X) = Y$ (surjective). In other words, φ is one-to-one and total. An isomorphism from $(\mathcal{L}, \leq)$ to $(\mathcal{L}, \leq)$ is called an automorphism.

Appendix B

Resolution for modal-morpho-logics

B.1 introduction

In [7] a resolution calculus for the basic hybrid logic is introduced. Here this resolution calculus is extended to work with the morpho language.

The morpho language is a hybrid language containing a nullary, unary and binary relation on the set of worlds. The language is based on a link between hybrid logic and mathematical morphology and therefore the underlying frame of every model of this language must adhere to the group axioms.

B.2 The logic

The morpho logic is a hybrid logic [13] containing a constant e, a unary modality $\otimes$ and a binary modality $\hat{\oplus}$ together with the global modality E.

Definition B.2.1 Morpho language Given a countably infinite set of proposition letters PROP, a countably infinite set of nominals NOM and ATOM = PROP $\cup$ NOM, let $\mathcal{L}$ be the set of morpho-logic formulas. Then a formula of the morpho language is defined by

$$\varphi := p|i|e|\varphi \vee \psi|\varphi \wedge \psi|\neg\varphi| \otimes \varphi|\underline{\otimes}\varphi|\varphi\hat{\oplus}\psi|\varphi\underline{\hat{\oplus}}\psi|@_i\varphi|E\varphi|A\varphi$$

with $p \in PROP$ and $i \in$ NOM. A formula of the form $@_i\varphi$ is called an @-formula. A formula of the form $E\varphi$ or $A\varphi$ is called a global formula.

A model for the morpho-language is defined as follows

Definition B.2.2 Model A model $\mathcal{M}$ for the morpho language is a tuple $(W, I, R, C, \mathcal{V})$ such that W is a countable (possible infinite) set of worlds, $I \subseteq W$ a unary relation on W, $R \subseteq W \times W$ a binary relation on W and $C \subseteq W \times W \times W$ a ternary relation on W. Furthermore, $\mathcal{V}$ is a valuation function $\mathcal{V}$: ATOM $\mapsto \mathcal{P}(W)$ such that for each $i \in$ NOM, $\mathcal{V}$ maps i to a singleton set.

Given an abelian group $(W, +, -, e)$ over W the relations I, R and C are defined as follows

- $(u, v, w) \in C$ iff $u = v + w$
- $(v, w) \in R$ iff $v = -w$
- $(w) \in I$ iff $w = e$

We are now ready to define the semantics for the morpho-language.

Definition B.2.3 Semantics Given a morpho model $\mathcal{M}$ and a world w in $\mathcal{M}$ the morpho semantics is defined as follows.

$M, w \Vdash p$	iff	$w \in V(p)$
$M, w \Vdash i$	iff	$w \in V(i)$
$M, w \Vdash e$	iff	$(w) \in I$
$M, w \Vdash \neg\varphi$	iff	$M, w \nVdash \varphi$
$M, w \Vdash \varphi \vee \psi$	iff	$M, w \Vdash \varphi$ or $M, w \Vdash \psi$
$M, w \Vdash \varphi \wedge \psi$	iff	$M, w \Vdash \varphi$ and $M, w \Vdash \psi$
$M, w \Vdash @_i\varphi$	iff	there is a world v s.t. $M, v \Vdash i$ and $M, v \Vdash \varphi$
$M, w \Vdash \otimes\varphi$	iff	$\exists v \in W$ s.t. $(w, v) \in R$ and $M, v \Vdash \varphi$
$M, w \Vdash \underline{\otimes}\varphi$	iff	$\forall v \in W$ $(w, v) \in R$ implies that $M, v \Vdash \varphi$
$M, w \Vdash \varphi\hat{\oplus}\psi$	iff	$\exists v, u \in W$ s.t. $(w, v, u) \in C$ and $M, v \Vdash \varphi$ and $M, u \Vdash \psi$
$M, w \Vdash \varphi\underline{\hat{\oplus}}\psi$	iff	$\forall v, u \in W$ $(w, v, u) \in C$ implies that $M, v \Vdash \varphi$ or $M, u \Vdash \psi$
$M, w \Vdash E\varphi$	iff	$\exists v \in W$ and $M, v \Vdash \varphi$
$M, w \Vdash A\varphi$	iff	$\forall v \in W$ we have that $M, v \Vdash \varphi$

B.3 Hybrid logic and resolution

The calculus given below is based on the calculus introduced in [6]. Before we can introduce the calculus, a number of assumptions and definitions need to be introduced, based on the definitions given in [6].

The calculus assumes that all the formulas are given in negation normal form, i.e. the negation operator can only be applied on atoms. This changes the definition of the language slightly, giving us the following language

Definition B.3.1 $\mathcal{L}$ in negated normal form
A formula φ is in $\mathcal{L}^{nnf}$ if it follows the following recursive definition

$$\varphi := a|\neg a|e|\varphi \wedge \psi|\varphi \vee \psi| \otimes \varphi|\underline{\otimes}\varphi|\varphi\hat{\oplus}\psi|\varphi\underline{\hat{\oplus}}\psi|E\varphi|A\varphi$$

with $a \in$ ATOM and $\varphi, \psi \in \mathcal{L}^{nnf}$.

Just as in First Order resolution, the calculus represents formulas as clauses. A clause in this case is a set of arbitrary $\mathcal{L}^{nnf}$ @-formulas and global formulas representing a disjunction of formulas. Allowing only @-formulas and global formulas in the clauses does not in any way affect the expressibility of the calculus. If a formula φ is satisfiable in some model, then $@_i\varphi$ is satisfiable as well and the satisfiability of a global formula does not depend on the world on which it is located.

Given a formula $\varphi \in \mathcal{L}^{nnf}$, $Clset = \{\{@_i\varphi\}\}$ for i an arbitrary nominal not occurring in φ if φ is not a global formula and $Clset = \{\varphi\}$ if it is a global formula. We define $Clset^*(\varphi)$ as the smallest set that includes $Clset(\varphi)$ and $\{@_{i_e}e\}$ and is closed under the application of the calculus given in Table B.1 and Table 4.4. We thus assume that there must be a world that satisfies the identity element. We need this in order for some of the rules defined in Table 4.4 to be effective.

To prove refutational completeness we need to show that φ is unsatisfiable given the morpho-semantics if and only if $Clset^*(\varphi)$ contains the empty clause.

In order for an implementation of the calculus to be effective a way must be found to define which clauses must be used in the calculations and which formulas in the clauses are candidates for application of the rules. For this purpose selection functions and an ordering on formulas are introduced.

An ordering on formulas must posses certain properties to make sure that it does not affect the refutational completeness of the calculus. An ordering possessing these properties is an admissible ordering.

Definition B.3.2 Ordering
An ordering is a binary relation that is both transitive and reflexive. Furthermore, an ordering is *total* if for any two elements φ and ψ, either $\varphi > \psi$ or $\varphi > \psi$ is the case. An ordering is *well-founded* if there exists no infinite chain such that $\varphi_1 > \varphi_2 > \varphi_3 > \ldots$. In other words, there must be a smallest element.

By $\varphi[\psi]_p$ it is indicated that in the formula φ, the proposition letter on position p is replaced with ψ. An ordering $>$ has the *subformula property* if $\varphi[\psi]_p > \psi$ whenever $\varphi[\psi]_p \neq \psi$. An ordering is a *rewrite ordering* when $\varphi[\psi_1]_p > \varphi[\psi_2]_p$ iff $\psi_1 > \psi_2$. A well-founded rewrite ordering is a *reduction ordering.* If, on top of that, an ordering also satisfies the subformula property it is called a *simplification ordering.*

Definition B.3.3 Admissible ordering [7] An ordering $>$ over $\mathcal{L}^{nnf}$ is admissible if it is a total simplification ordering satisfying the following conditions for all $\varphi, \psi \in \mathcal{L}^{nnf}$ and all $i, j, i_1, i_2, j_1, j_2 \in$ NOM:

A1 $\varphi > i$ for all $\varphi \notin$ NOM

A2 if $\varphi > \psi$, then $@_i\varphi > @_i\psi$

A3 if $\otimes i$ is a proper subformula of φ, then $\varphi > \otimes j$

A4 if Ei is a proper subformula of φ, then $\varphi > Ej$

A5 if $i_1 \hat{\oplus} i_2$ is a proper subformula of φ, then $\varphi > j_1 \hat{\oplus} j_2$

A6 $\underline{\otimes} i > \otimes j$

A7 $Ai > Ej$

A8 $i_1 \underline{\hat{\oplus}} i_2 > j_1 \hat{\oplus} j_2$

A9 $A\varphi > @_i\varphi$

A10 $E\varphi > @_i\varphi$

Given the above definition of an admissible ordering, the following ordering on $\mathcal{L}^{nnf}$ can be defined. The ordering is based on a lexicographic path ordering.

Definition B.3.4 Admissible ordering over $\mathcal{L}^{nnf}$ [7] Given a hybrid signature $S = \langle \{p_i | i \in \mathcal{N}\}, \{n_i | i \in \mathcal{N}\} \rangle$, let O be the set $S \cup \{e, \neg, \wedge, \vee, @, \otimes, \underline{\otimes}, \hat{\oplus}, \underline{\hat{\oplus}}, A, E\}$ and define the precedence relation $> \subseteq O \times O$ as the transitive closure of the set

$$\{(A,E),(E,@),(@,\neg),(\neg,\wedge),(\wedge,\vee),(\vee,\underline{\hat{\oplus}}),(\underline{\hat{\oplus}},\hat{\oplus}),(\hat{\oplus},\underline{\otimes}),(\underline{\otimes},\otimes)\}\cup$$
$$\{(\otimes,p_i),(p_i,n_j)|i,j\in\mathcal{N}\}\cup$$
$$\{(p_i,p_j),(n_i,n_j)|i>j\}$$

By definition, $>$ is total, irreflexive and well-founded. Let $>_{lpo}$ be the lexicographic path ordering over $\mathcal{L}^{nnf}$ that uses $>$ as precedence. It follows that $>_{lpo}$ must be a total simplification ordering. Finally, define $>_h$ as

$$\varphi >_h \psi \text{ iff } \begin{cases} \text{size}(\varphi) > \text{size}(\psi), \text{ or} \\ \text{size}(\varphi) = \text{size}(\psi) \text{ and } \varphi >_{lpo} \psi \end{cases}$$

where size(φ) is the number of operators in φ.

Proposition B.3.5 $>_h$ *is an admissible ordering.*

Proof. First, we prove that $>_h$ is a total simplification ordering. It is easy to see that $>_h$ is a total ordering. Suppose we have two formula's φ and ψ. If they differ in size it is the case that either $\varphi >_h \psi$ or $\psi >_h \varphi$. If they are of equal size, the fact that $>_{lpo}$ is a total order makes that either $\varphi >_h \psi$ or $\psi >_h \varphi$, and thus $>_h$ is a total order. To see that $>_h$ is a simplification ordering, note that if size($\varphi[\psi]_p$) > size(ψ), then $\varphi[psi]_p >_h \psi$. Combined with the fact that $>_{lpo}$ is a simplification ordering this gives us that $>_h$ is a total simplification ordering as well.

Second, we must proof that $>_h$ possesses the properties A1 - A10. We treat them one by one.

A1 If size(φ) > 1 then by definition $\varphi >_h i$. If size(φ) = 1 then by definition of $>_{lpo}$ $\varphi >_h i$.

A2 Suppose that $\varphi >_h \psi$ for some arbitrary formulas φ and ψ. Then it either is that case that size(φ)$>_h$size(ψ), or size(φ) = size(ψ) and $\varphi >_{lpo} \psi$. In the former, it is also the case that size($@_i\varphi$)$>_h$size($@_i\psi$) and thus $@_i\varphi >_h @_i\psi$. In the latter case, we have that have by definition that $@_i\varphi >_{lpo} @_i\psi$, and thus $@_i\varphi >_h @_i\psi$.

A3 If $\otimes i$ is a proper subformula of φ, it is the case that size(φ) > size($\otimes i$), and thus size(φ) > size($\otimes j$). This means that size(φ)$>_h$ size($\otimes j$).

A4 This proof follows the same reasoning as for A3

A5 This proof follows the same reasoning as for A3

A6 It is the case that size($\underline{\otimes} i$)=size($\otimes i$). Furthermore, by definition $\underline{\otimes} i >_{lpo} \otimes i$ and thus $\underline{\otimes} i >_h \otimes i$.

A7 The proof for A7 follows the same reasoning as A6.

A8 The proof for A8 follows the same reasoning as A6.

A9 By definition of $>_{lpo}$ it is the case that $A\varphi >_{lpo} @_i\varphi$, and thus $A\varphi >_h @_i\varphi$.

A10 By definition of $>_{lpo}$ it is the case that $E\varphi >_{lpo} @_i\varphi$, and thus $E\varphi >_h @_i\varphi$.

QED

One can lift the defined ordering to clauses in the following way. Given a clause C and a clause D, $C > D$ if and only if $C \neq D$ and if there is some $\varphi \in D$ such that $\varphi \notin C$ then there is a $\psi \in C$ such that $\psi > \varphi$.

In first order logic, a selection function is defined in such a way that it selects only negative formulas. Because the clauses in this calculus can contain arbitrary @-formulas the definition of a negative literal must be redefined as follows

Definition B.3.6 Negative literal First, we define the set of positive literals PLIT to be the following set. PLIT := $@_ij | @_ip | @_i \otimes j | @_ij_1 \hat{\oplus} j_2$, for $i, j_1, j_2 \in$ NOM and $p \in$ PROP. The set of *negative literals* is defined as the complement of PLIT.

Using this definition, a selection function is defined as follows

Definition B.3.7 Selection function [7] A function S from clauses to clauses is a selection function iff for every clause C we have that $S(C) \subseteq C$, $|S(C)| \leq 1$ and $S(C) \cap \text{PLIT} = \emptyset$

B.4 Refutational completeness

The proof of refutational completeness is based on the proof in [9, 7]. The main idea behind the proof lies in the generation of Herbrand models. In applying the rules to the set of clauses, a candidate model is build. If the original set is satisfiable this should lead to a model, if it is not satisfiable to a contradiction.

The application of the rules displayed in Table B.1 are used to create a proper hybrid model. The rules displayed in Table 4.4 make sure that the model that is found adheres to the group rules.

The following definitions and theorems are taken from [7]. For N a hybrid model, let diag(N) be the set $\text{diag}(N) = \{\varphi|\varphi \in \text{PLIT and } N \Vdash \varphi\} \cup \{\neg\varphi|\varphi \in \text{PLIT and } N \nVdash \varphi\}$. A model is named if each world in the domain satisfies a nominal.

Theorem B.4.1 (Scott's Isomorphism Theorem.) *Let M and N be two countable, named hybrid models. Then M and N are isomorphic iff $M \Vdash$ diag(N).*

Using Theorem 1, we can define a Herbrand model in the following manner.

Definition B.4.2 Herbrand Model Let $S = \langle \text{PROP}, \text{NOM} \rangle$ be a hybrid signature. A hybrid Herbrand model for $\mathcal{L}$ over S is any set $H \subseteq \text{PLIT}$

From a hybrid herbrand model an hybrid model can be build using the following definition.

Definition B.4.3 Given a hybrid Herbrand model H, let $\sim_H$ be the minimum equivalence relation over NOM that extends the set $\{(i,j)|@_i j \in H\}$. We now define the hybrid model uniquely determined by H as $\langle W^H, I^H, R^H, C^H, \mathcal{V}^H \rangle$ where

$$\begin{array}{lcl} W^H & = & \text{NOM}/_{\sim_H} \\ I^H & = & \{([j])|@_j e \in H\} \\ R^H & = & \{([i],[j])|@_i \otimes j \in H\} \\ C^H & = & \{([i],[j_1],[j_2])|@_i j_1 \hat{\oplus} j_2 \in H\} \\ \mathcal{V}^H(p) & = & \{[i]|@_i p \in H\}, p \in \text{PROP} \\ \mathcal{V}^H(i) & = & \{[i]\}, i \in \text{NOM} \end{array}$$

where $\text{NOM}/_{\sim_H}$ is the set consisting of equivalence classes of $H/_{\sim_H}$ and $[i]$ is the equivalence class assigned to i by $\sim_H$.

Using theorem B.4.1 we can proof the following theorem.

Theorem B.4.4 *Given Γ and a set of @-formulas of $\mathcal{L}$ over a signature $S = \langle \text{PROP}, \text{NOM} \rangle$, Γ has a hybrid model if and only if it has a hybrid Herbrand model over the signature $S = \langle \text{PROP}, \text{NOM} \cup \text{NOM}' \rangle$, where NOM' is a numerable set disjoint from NOM.*

Proof. Suppose that Γ has a hybrid model $\mathcal{M}$ over the signature S. Take H to be $H = \text{diag}(\mathcal{M}) \cap \text{PLIT}$. Use the construction in Definition B.4.3 to create a model $\mathcal{M}'$. By definition it satisfies diag($\mathcal{M}$) and by theorem B.4.1 it is isomorphic with $\mathcal{M}$. Hence it is also a model for Γ.

For the other direction, suppose that $\mathcal{M}$ is a hybrid Herbrand model for Γ, then trivially it has a hybrid model. QED

As already mentioned above, the idea behind the proof is that by applying the rules on a set of clauses N a candidate model for N is build. Before a definition of the candidate model can be given the following definitions are needed.

Definition B.4.5 σ_H Given a hybrid Herbrand interpretation H, the following substitution of nominals to nominals can be defined.

$$\sigma_H = \{i \mapsto j | i \sim_H j \wedge (\forall k)(k \sim_h j \rightarrow k \geq j\}.$$

The purpose of σ_H is to replace every occurrence of a nominal with the least nominal of it's class. That is, for each equivalence class, the least nominal is taken as it's representation.

Definition B.4.6 SIMP The set of simple formulas over $\mathcal{L}^{nnf}$ is defined as follows

$$\text{SIMP} := @_i j(\text{ with } i > j)|@_i p|@_i \neg a|@_i \otimes j|@_i \underline{\otimes} \varphi|@_i j_1 \hat{\oplus} j_2|@_i \varphi \hat{\underline{\oplus}} \psi|A\varphi$$

where $>$ is admissible, $i, j \in \text{NOM}, p \in \text{PROP}, a \in \text{ATOM}$ and $\varphi, \psi \in \mathcal{L}^{nnf}$.

A model for N is a model that satisfies at least one formula from each clause of N. Hence, to build a candidate model a method is needed to pick specific formulas from N that can be used to build a formula. Through Theorem B.4.4 we know that we only have to look at formulas from PLIT. The following definitions are used to create a candidate model. The idea is that one works through N, each time choosing the smallest clause not yet looked at. From this clause one chooses the maximal formula. If this formula is both in PLIT and not yet satisfied by the model constructed so far, add it to the Herbrand model. Formally, this is defined in the following manner. Despite the fact that the definitions below are given separately, they must be viewed as a whole.

Definition B.4.7 H_C Let C be a clause (not necessarily in N), name H_C the hybrid Herbrand interpretation given by $\bigcup_{C>D} \epsilon_D$

Definition B.4.8 Reduced form Let C be a clause and φ its maximal formula. If $\varphi \in$ SIMP and either a) $\varphi \in$ PLIT and $\varphi = \varphi\sigma_{H_C}$, or b) $\varphi = @_i \underline{\otimes} \psi, \varphi = @_i \psi_1 \hat{\underline{\oplus}} \psi_2$, $\varphi = A\psi$ or $\varphi = @_i \neg a$ and $\varphi = \varphi\sigma_{H_C}$ then we say that both φ and C are in *reduced form*.

Definition B.4.9 ϵ_C Let C be a clause (not necessarily in N). If it simultaneously is the case that a) $C \in N$, b) C is in reduced form, c) The maximal formula in C is in PLIT, d) C is false under H_C, and e) $S(C) = \emptyset$ then $\epsilon_C = \{\varphi\}$, where φ is the maximal formula in C. Otherwise ϵ_C is empty.

We are now ready to define a candidate model for a set of clauses N.

Definition B.4.10 Candidate model A candidate model for N is defined as $H_N = \bigcup_{C \in N} \epsilon_C$.

This definition however does not capture exactly what, according to the morpho-semantics is a candidate model. Namely, a candidate model must satisfy all the group axioms. Therefore, the following proposition is needed.

Proposition B.4.11 *Given a set of clauses N. If N is closed under the rules presented in Table 4.4 then $H_N = \langle W^H, I^H, R^H H, C^H, \mathcal{V}^H \rangle$ can be extended to a model that satisfies the group axioms as defined in Definition B.2.2.*

Proof. We begin by giving the extension. The only two relations that need to be changed are R^H and C^H. The first is due to the fact that there can be a world $[i]$ that has no R-successors, the second is due to the fact that it can be the case that a world $[i]$ such that there is no formula φ in N such that $\varphi = @_i\psi$.

$R' = R^H \cup \{([i][i]) | \forall j \in \text{NOM}\{@_i j\} \notin N\}$

$C' = C^H \cup \{([i],[j],[i]),([i],[i],[j]) | \{@_j e\} \in N, \text{ for all } \varphi \text{ s.t. } i \text{ occurs in} \varphi, \varphi \neq @_i\psi\}$

Using the above defined relations H'_N can be defined as follows

$$H'_N = \langle W^H, I^H, R', C', \mathcal{V}^H\rangle$$

Next, it must be shown that H'_N satisfies the group axioms as defined in Definition B.2.2. First, we look at the properties for the C' relation. After that the R' relation is considered. Finally the interaction between C', R' and I^H is taken care of. Assume that N is closed under the rules.

The first property that C' must satisfy is associativity. Assume that are some worlds $[i],[j_1],[j_2],[s_1],[s_2]$ such that $([i],[j_1],[j_2]) \in C'$ and $([j_1],[s_1],[s_2]) \in C'$. In order for C' to be associative it must be the case that there is some world $[z]$ such that $([i],[s_1],[z]) \in C'$ and $([z],[s_2],[j_2]) \in C'$. Two cases can now occur. First, it can be the case that there are clauses in N that produce $@_i j_1 \hat{\oplus} j_2$ and $@_{j_1} s_1 \hat{\oplus} s_2$. But this means that we can apply the (Ass1) rule, producing $@_i z \hat{\oplus} s_2$ and $@_z j_1 \hat{\oplus} s_1$. This means that, by definition there is a world $[z]$ such that $([i],[z],[s_2]) \in C'$ and $([z],[j_1],[s_1]) \in C'$.

Second, it can be the case that $([i],[i],[j]) \in C'$ and $([i],[j],[i]) \in C'$ with $[j] \in I^H$ and for each formula φ occuring in N, $\varphi \neq @_i\varphi$. From $[j] \in I^H$ one can conclude that $\{@_j e\} \in N$. By closure under rule (Id$_2$) it is the case that $\{@_j j \hat{\oplus} j\} \in N$ and thus $([j],[j],[j]) \in C'$, satisfying the associativity condition in the case where $[i] = [j_1] = [s_1]$ and $[j] = [j_2] = [s_2] = [z]$. In the case that $[i] = [j_1] = [s_2] = [z]$ and $[j] = [j_2] = [s_1]$ we automatically satisfy the condition for associativity. The reasoning for the (Ass2) rule is the same. Hence C' is associative.

Second, it must be the case that C' is commutative, i.e. $([s],[i],[j]) \in C^H$ implies that $([s],[j],[i]) \in C^H$. So suppose that there are worlds $[s],[i]$ and $[j]$ such that $([s],[i],[j]) \in C'$. Again we can discern two cases. First, assume that there must be a clause such that it produces $@_s i \hat{\oplus} j$. By closure of N it must be the case that there is some clause that produces $@_s j \hat{\oplus} i$. This means that $([s],[j],[i]) \in C'$. In the case that $[s] = [i]$, $[j] \in I^H$ and for all formula φ occuring in N it is the case that $\varphi \neq @_i\psi$ for arbitrary ψ, by construction we have that $([i],[j],[i])$. Thus C' is commutative.

Third, it must be the case that $\forall x, y \exists z\, C'zxy$. Suppose that there are worlds $[j_1],[j_2]$, then by definition there is some clause that produces φ and j_1 occurs in φ and there is some clause that produces ψ such that j_2 occurs in ψ. This means that the (total) rule can be applied in combination with the $(\hat{\oplus})$ rule, creating a clause that produces $@_i j_1 \hat{\oplus} j_2$ which means that $([i],[j_1],[j_2]) \in C'$. Thus, H_N already satisfies this condition and H'_N is equal to H_N.

Finally, it should be the case that $\forall xyzv\, C'xyz \wedge C'vyz \rightarrow v = x$, i.e. $i \hat{\oplus} j$ can be true in only one world. Suppose that there are worlds $[i],[j],[s_1],[s_2]$ such that $([i],[s_1],[s_2]) \in C'$ and $([j],[s_1],[s_2]) \in C'$. This means that there is a clause that produces $@_i s_1 \hat{\oplus} s_2$ and a clause that produces $@_j s_1 \hat{\oplus} s_2$. By application of (unique$_{\hat{\oplus}}$) it must be the case that there is some clause that produces $@_i j$. Again, H'_N is equal to H_N.

As for the R^H relation, it should be the case that $\forall x \exists y R'xy$ and $\forall xyz(R'xy \wedge R'xz \rightarrow y = z)$. Looking at the first condition, suppose that there is some world $[i]$ and no $[j]$ such that $([i],[j]) \in R'$). This means that $\{@_i \otimes j\} \notin N$ for all j. But by definition this means that $([i],[i]) \in R'$. Thus R' satisfies the first condition.

The second condition is false if there are distinct worlds $[i],[j_1],[j_2]$ such that $([i],[j_1]) \in R'$ and $([i],[j_2]) \in R'$. But this means that $\{@_i \otimes j_1\} \in N$ and $\{@_i \otimes j_2\} \in N$. By closure under the (Rev$_2$) rule it is the case that $\{@_i \underline{\otimes} j_1\} \in N$ and by closure under the $(\underline{\otimes})$ rule it is the case that $\{@_{j_1} j_2\} \in N$. But since $[j_1]$ and $[j_2]$ are distinct worlds this gives us a contradiction.

As for the interaction between C' and R', we have the following two properties. First, it must be the case that it if there are worlds $[i], [j_1]$ and$[j_2]$ such that $([i], [j_1], [j_2]) \in C'$ and $([j_1]) \in I'$ it must be the case that $[i] = [j_2]$. To show that this is the case, assume that there are worlds $[i], [j_1]$ and$[j_2]$ in H_N such that $([i], [j_1], [j_2]) \in C'$ and $([j_1]) \in I^H$. By definition it must be the case that there is a clause in N that produces $@_i j_1 \hat{\oplus} j_2$ and furthermore it must be the case that there is a clause that produces $@_{j_1} e$. By closure of N under the (Id_1 rule) it must be the case that there is a clause that produces $@_i j_2$. In the case where there is no formula of the form $\{@_i \varphi\} \in N$ by construction it is the case that $([i], [j_2], [i]) \in C'$.

Second, it must be the case that if there is some world $[i]$ and there is some world $[j]$ such that $([j]) \in I^H$, then $([i], [j], [i]) \in C'$. To see that H_N satisfies this property, assume that there is some world $[i]$ and there is some world $[j]$ such that $([j]) \in I^H$. Then by definition there must be a clause that produces $@_j e$. Furthermore, there must be a clause that produces φ such that i occurs in φ, with either $\varphi = @_i \psi$ or $\varphi \neq @_i \psi$ for some formula ψ. In the former case, by closure of N under (Id_2) it must be the case that there is some clause that produces $@_i j \hat{\oplus} i$ and by construction $([i], [j], [i]) \in C'$ and H'_N is equal to H_N. In the latter case we have, by construction, that $([i], [j], [i]) \in C'$.

Thus H_N can be extended to a model H'_N such that it satisfies the group axioms. QED

Definition B.4.12 Counterexample If a clause C is false under the extension H'_N of H_N, we say that C is a counterexample of H_N.

Proposition B.4.13 *Given a set of clauses N and $C \in N$ the minimum counterexample of H_N, with respect to an admissible ordering $>$. If $C \neq \emptyset$, then there exists an inference using one of the rules of the calculus such that:*

1. *C is the main premise*
2. *the side premise (when present) is productive*
3. *all the consequences are smaller, with respect to $>$, than C and at least one of them is a counterexample of I_N*

Proof. The first thing to be proved is that every rule in Table B.1 produces at least one element that is smaller. The (RES) and (REF) rules trivially satisfies this condition because the both delete the maximal formulas in the preconditions.

As for the ($\wedge$), ($\vee$), ($\underline{\otimes}$), ($\underline{\hat{\oplus}}$), ($@E$), ($@A$) and ($@$) rules, each consequence of these rules is a subformula of the main premise. Because an admissible ordering must posses the subformula property, all these rules will satisfy requirement 3.

As for the (REF), ($\otimes$) and ($\hat{\oplus}$) , suppose that φ is the distinguished formula of the main clause. For every consequent ψ it is the case that size(φ) > size(ψ). Hence ($\otimes$) and ($\hat{\oplus}$) satisfy requirement 3 as well. According to $A9$ and $A10$ it must be the case that $A\varphi > @_i \varphi$ and $E\varphi > @_i \varphi$, hence the rules (A) and (E) satisfy requirement 3. Looking at the (PARAM) rule, the rewrite property and the fact that $j > i$ tells us that $\varphi(i) > \varphi(i/j)$. Finally, the rule (SYM) satisfies requirement 3 because of the rewrite ordering property.

Second, suppose that C is as described in the proposition. Furthermore, assume that φ is the maximal formula of C. In proving requirements 1 and 2 we have several possibilities for φ. First, suppose that $\varphi \notin$ SIMP, then either ($\wedge$), ($\vee$), ($\otimes$), ($\hat{\oplus}$), (E), ($@E$) or ($@A$) can be applied and the proposition is trivially true.

Next, suppose that $\varphi \in$ SIMP and is not in reduced form. This means that there must be some i occurring in φ that is not the least nominal in it's equivalence class. Thus there must be some D

that produces $@_i j$ such that $i > j$. The (PARAM) rule can be applied to C and D showing that the wanted inference exists.

If φ is in reduced form it cannot be in PLIT, because than it would be produced and would be true in H_N. Thus φ must be one of the following:

- $\varphi = @_i \underline{\otimes} \psi$. For φ to be false, there must be some clause in N that produces $@_i \otimes j$ such that $@_j \varphi$ is false. But this means that we can use $(\underline{\otimes})$ to create a new clause containing $@_j \varphi$ which then again is a counterexample to H_N.
- $\varphi = @_i \psi_1 \hat{\underline{\oplus}} \psi_2$. For φ to be false, there must be some clause in N that produces $@_i j_1 \hat{\oplus} j_2$ such that $@_{j_1} \psi_1$ and $@_{j_2} \psi_2$ are both false. But then $(\hat{\underline{\oplus}})$ can be used to create a clause that contains both $@_{j_1} \psi_1$ and $@_{j_2} \psi_2$ and is thus a counterexample to H_N.
- $\varphi = A\psi$. For φ to be false there must be some clause in N that produces $@_i \xi$ and $@_i \psi$ is false in H_N. But then (A) can be applied creating a new clause containing $@_i \psi$ which is again a counter example to H_N.
- $\varphi = @_i \neg a$. If $a = i$, then the (REF) rule can be used to create a new and smaller counterexample. If $a \neq i$, there must be some clause in N that produces $@_i a$ and hence we can use the (RES) rule to produce a new and smaller counterexample of H_N.

QED

Proposition B.4.14 *If N is saturated with respect to the calculus, then H'_N and H_N satisfy exactly the same clauses in N.*

Proof. By construction, H'_N and H_N differ only in the relations C' and C^H and R' and R^H.

In the case of R' and R^H it can be the case that there is some world $[i]$ such that there is no world $[j]$ such that $([i],[j]) \notin R^H$. This means that there is some clause that produces φ, with φ containing i, but no clause of the form $@_i \otimes j$ is produced. Changing R^H such that $([i],[i]) \in R'$ can only have affect on the satisfiability of a formula of the form $@_i \underline{\otimes} \varphi$. But by closure of N under (Rev_1) and ($\otimes$) however, it must be the case that there are clauses that produce $@_i \otimes j$ and $@_j \varphi$. This means that there is a world $[j]$ such that $([i],[j]) \in R^H$ making H'_N and H_N agree.

As for the case where C^H is adjusted, assume that there are some worlds $[i]$ and $[j]$ such that $([j]) \in I^H$. Furthermore, suppose that there is no clause that produces a formula of the form $@_i \varphi$. This means that $([i],[j],[i])$ must be added to C^H, creating C'. This move can only affect the satisfiability of a formula of the form $@_i \varphi \hat{\underline{\oplus}} \psi$, but by assumption there is no such clause in N. QED

Using proposition B.4.13, refutational completeness can be proved in the following manner.

Theorem B.4.15 *If N is saturated with respect to the calculus and it does not contain the empty clause, then there is a model $\mathcal{M}$ in terms of the Definition B.2.2 such that it satisfies N.*

Proof. If N is saturated with respect to the calculus then it is closed under the rules defined in Table 4.4. Thus H_N can be extended to a model in terms of the Definition B.2.2. Also, via proposition B.4.13 we know that if N contains a counterexample it should also contain the empty clause because this is the smallest counterexample to which no rule can be applied. Hence every clause in N is satisfied in H_N. Thus, by Proposition B.4.14, H'_N is also a model of N. Hence, if N is saturated with respect to the calculus and it does not contain the empty clause a model for N can be found. QED

$$(\wedge)\ \frac{Cl \cup \{@_i\varphi \wedge \psi\}}{\begin{array}{c}Cl \cup \{@_i\varphi\}\\ Cl \cup \{@_i\psi\}\end{array}} \qquad (\vee)\ \frac{Cl \cup \{@_i\varphi \vee \psi\}}{Cl \cup \{@_i\varphi, @_i\psi\}}$$

$$(\text{RES})\ \frac{Cl_1 \cup \{@_i p\} \quad Cl_2 \cup \{@_i \neg p\}}{Cl_1 \cup Cl_2}$$

$$(\underline{\otimes})\ \frac{Cl_2 \cup \{@_i \otimes (j) \quad Cl_1 \cup \{@_i \underline{\otimes} \varphi\}\}}{Cl_1 \cup Cl_2 \cup \{@_j\varphi\}} \qquad (\otimes)\ \frac{Cl \cup \{@_i \otimes \varphi\}}{\begin{array}{c}Cl \cup \{@_i \otimes (j)\}\\ Cl \cup \{@_j\varphi\}\end{array}} \ \begin{array}{l}\text{for a new } j \in \text{NOM}\\ \text{and } \varphi \notin \text{NOM}\end{array}$$

$$(\underline{\hat{\oplus}})\ \frac{Cl_2 \cup \{@_i(j_1 \hat{\oplus} j_2) \quad Cl_1 \cup \{@_i\varphi \underline{\hat{\oplus}} \psi\}\}}{Cl_1 \cup Cl_2 \cup \{@_{j_1}\varphi, @_{j_2}\psi\}}$$

$$(\hat{\oplus})\ \frac{Cl \cup \{@_i(\varphi \hat{\oplus} \psi)\}}{\begin{array}{c}Cl \cup \{@_i(j_1 \hat{\oplus} j_2)\}\\ Cl \cup \{@_{j_1}\varphi\}\\ Cl \cup \{@_{j_2}\psi\}\end{array}} \ \begin{array}{l}\text{for new } j_1, j_2 \in \text{NOM}\\ \text{and } \varphi, \psi \notin \text{NOM}\end{array}$$

$$(E)\ \frac{Cl \cup \{E\varphi\}}{Cl \cup \{@_i\varphi\}} \ \text{for a new } i \in \text{NOM} \qquad (A)\ \frac{Cl_1 \cup \{@_i\psi\} \quad Cl_2 \cup \{A\varphi\}}{Cl_1 \cup Cl_2 \cup \{@_i\varphi\}}$$

$$(@E)\ \frac{Cl \cup \{@_i E\varphi\}}{Cl \cup \{E\varphi\}} \qquad (@A)\ \frac{Cl \cup \{@_i A\varphi\}}{Cl \cup \{A\varphi\}}$$

$$(\text{PARAM})\ \frac{Cl_1 \cup \{@_j i\} \quad Cl_2 \cup \{\varphi(j)\}}{Cl_1 \cup Cl_2 \cup \{\varphi(j/i)\}} \ \begin{array}{l}\text{if } j > i \text{ and}\\ \varphi(j) > @_j i\end{array}$$

$$(\text{SYM})\ \frac{Cl \cup \{@_j i\}}{Cl \cup \{@_i j\}} \ \text{if } i > j$$

$$(\text{REF})\ \frac{Cl \cup \{@_i \neg i\}}{Cl} \qquad (@)\ \frac{Cl \cup \{@_i @_j \varphi\}}{Cl \cup \{@_j\varphi\}}$$

Restrictions: Assume an admissible ordering $>$ and a selection function S. In the following, φ and ψ are the formulas explicitly displayed in the rules. The main premise of each rule is the rightmost, the other premise(in rules with two premises) is the side premise.

- If $C = C' \cup \{\varphi\}$ is the main premise, then either $S(C) = \{\varphi\}$ or, $S(C) = \emptyset$ and $\{\varphi\} < C'$
- If $D = D' \cup \{\psi\}$ is the side premise, then $\{\psi\} > D'$ and $S(D) = \emptyset$

Table B.1: Resolution calculus $R^{os}[\mathcal{L}^{nnf}]$

Appendix C

Formulas used for evaluation

C.1 Formulas of modal depth 0

- $(\neg @_{n_1} n_3) \wedge (@_{n_4} \neg n_2) \wedge (\neg @_{n_2} n_1) \wedge (\neg @_{n_2} \neg p_3) \wedge (\neg @_{n_4} p_5)$
- $(@_{n_1} \neg n_4) \wedge (\neg @_{n_3} \neg p_2) \wedge n_4 \wedge (@_{n_3} \neg n_1) \wedge (@_{n_5} \neg n_3)$
- $(\neg @_{n_1} n_4) \wedge n_5 \wedge (\neg @_{n_4} \neg p_4) \wedge (\neg @_{n_3} n_2) \wedge (@_{n_1} n_3)$
- $(\neg @_{n_3} n_1) \wedge (@_{n_4} \neg p_4) \wedge (@_{n_5} \neg p_4) \wedge (\neg @_{n_4} p_1) \wedge (\neg @_{n_1} \neg n_2)$

C.2 Formulas of modal depth 1

- $p_1 \wedge (\neg @_{n_3} \neg n_4 \oplus p_2) \wedge p_1 \wedge (\neg @_{n_1} \neg p_2 \oplus \neg p_4) \wedge (\neg @_{n_4} \neg(\neg n_1 \oplus \neg p_5))$
- $\neg(\underline{\otimes}(@_{n_5} \neg p_5)) \wedge \neg(\underline{\otimes} \neg @_{n_4} \neg p_2) \wedge \neg((@_{n_2} \neg n_4) \oplus (@_{n_2} p_5)) \wedge \neg(\neg p_3 \oplus \neg p_3) \wedge \underline{\otimes}(@_{n_2} p_2)$
- $\neg(\underline{\otimes} \neg @_{n_4} \neg n_3) \wedge (\neg @_{n_5} \neg n_1 \oplus n_1) \wedge (@_{n_1} \neg n_4) \oplus (@_{n_5} \neg n_1) \wedge (@_{n_5} n_2) \oplus \neg(@_{n_4} \neg p_2) \wedge (@_{n_5} \neg p_5) \oplus (@_{n_5} p_5)$
- $\underline{\otimes}(\neg @_{n_2} \neg p_4) \wedge \neg((\neg @_{n_4} p_1) \oplus (@_{n_1} \neg p_1)) \wedge \neg \underline{\otimes}(@_{n_5} p_1) \wedge \neg \underline{\otimes} n_4 \wedge \neg \underline{\otimes} n_3$

C.3 Formulas of modal depth 2

- $(@_{n_4}((\underline{\otimes} \neg n_3) \oplus \neg(n1 \oplus \neg n_5))) \wedge (@_{n_4} \neg \underline{\otimes} \neg \underline{\otimes} \neg p_2) \wedge \underline{\otimes}((@_{n_5} \neg n_1) \oplus (@_{n_3} p_2)) \wedge \neg \underline{\otimes\otimes}(@_{n_1} p_1) \wedge \neg \underline{\otimes} \neg((@_{n_3} \neg p_2) \oplus (@_{n_1} \neg p_3))$
- $\neg \underline{\otimes\otimes}(\neg @_{n_5} \neg n_1) \wedge \neg(\neg(\underline{\otimes} @_{n_3} n_1) \oplus ((\neg @_{n_3} n_2) \oplus (@_{n_1} \neg p_3))) \wedge \neg((\neg @_{n_3} \neg(p_4 \oplus \neg p_4)) \oplus (@_{n_2} \neg(p_3 \oplus \neg p_2)))) \wedge (@_{n_2} \neg(\neg(n_1 \oplus n_4) \oplus (\underline{\otimes} n_2))) \wedge \underline{\otimes} \neg(\neg(@_{n_1} p_5) \oplus (@_{n_5} \neg p_4))$
- $((@_{n_1} \neg(\neg p_5 \oplus p_5)) \oplus ((\neg @_{n_2} n_4) \oplus \neg(@_{n_1} \neg p_2))) \wedge (\neg @_{n_4}((\underline{\otimes} n_2) \oplus (\underline{\otimes} \neg p_1))) \wedge \neg((\underline{\otimes} @_{n_1} \neg n_4) \oplus ((@_{n_4} n_5) \oplus (\neg @_{n_2} n_1))) \wedge (\neg @_{n_3} \underline{\otimes\otimes} \neg p_3) \wedge \neg(\underline{\otimes} \neg(@_{n_3}(p_5 \oplus p_2)))$
- $(\neg((@_{n_3} n_5) \oplus (\neg @_{n_1} n_3)) \oplus ((\neg @_{n_2} n_4) \oplus \neg(@_{n_1} \neg p2))) \wedge (\neg @_{n_4} \underline{\otimes}(n_3 \oplus \neg p_3) \wedge (\neg @_{n_1} \neg((\neg p_2 \oplus \neg n_2) \oplus \neg(\underline{\otimes} \neg n_2))) \wedge \neg \underline{\otimes\otimes}(@_{n_4} n_2) \wedge (@_{2} \neg((\neg n_3 \oplus \neg n_4) \oplus \neg(p_5 \oplus \neg p_3))))$

C.4 Formulas of modal depth 3

- $(\neg(\underline{\otimes}((\neg n_1 \oplus \neg p_2) \oplus (\neg p_3 \oplus \neg n_4))) \wedge ((@_{n_3}\neg((\underline{\otimes}\neg n_4) \oplus (n_3 \oplus \neg p_4))) \wedge (\neg((\neg(@_{n_5}\neg p_4) \oplus (\underline{\otimes} p_1)) \oplus ((@_{n_4} n_3) \oplus \neg(\underline{\otimes} p_4))) \wedge ((\underline{\otimes}\neg(\underline{\otimes}(p_4 \oplus \neg n_3))) \wedge \neg(\underline{\otimes}((@_{n_4}\neg p_5) \oplus \neg(@_{n_3} n_5)))))))$
- $(\neg(@_{n_2}(\underline{\otimes}(\neg p_2 \oplus \neg p_3))) \wedge ((\underline{\otimes}((\underline{\otimes} p_3) \oplus (@_{n_4}\neg n_2))) \wedge (n_1 \wedge (\neg(\underline{\otimes}\neg_2) \wedge (\neg(@_{n_1}\neg(\underline{\otimes}\neg p_4)) \oplus (@_{n_3}\neg(\neg p_5 \oplus n_2)))))))$
- $((@_{n_5}(\neg(\neg p_4 \oplus \neg n_1) \oplus \neg(\underline{\otimes} n_5))) \wedge (\neg((\underline{\otimes}\neg(\neg p_3 \oplus p_3)) \oplus \neg(\underline{\otimes}\neg(\neg n_3 \oplus \neg n_3))) \wedge (\neg((\underline{\otimes}(\underline{\otimes} n_1)) \oplus \neg(@_{n_2}(p_5 \oplus \neg p_1))) \wedge ((\underline{\otimes}(@_{n_5}(p_3 \oplus n_5))) \wedge \neg('neg((\underline{\otimes}\neg p_1) \oplus (\underline{\otimes}\neg n_2)) \oplus \neg(@_{n_2}\neg(\neg n_4 \oplus p_1)))))))$
- $(((@_{n_3}\neg(\underline{\otimes} n_5)) \oplus \neg(\underline{\otimes}\neg(n_2 \oplus p_1))) \wedge (\neg(\neg(\underline{\otimes}(@_{n_1} p_3)) \oplus \neg(@_{n_2}(\neg p_3 \oplus n_2))) \wedge (@_{n_4}\neg(\underline{\otimes}(\neg n_2 \oplus \neg n_1))) \wedge ((@_{n_4}\neg(\underline{\otimes}\neg(\underline{\otimes}\neg n_5))) \wedge n_3)))$

Appendix D

Completeness of pure morpho formulas

In Chapter 3 a proofsketch is given for Theorem 3.3.7. In this appendix the full proof is presented. What we want is to prove that a set of pure morpho-formulas Σ, $K^{+}_{\mathcal{H}(E)}M\Sigma$ is complete for the set of frames F_{Σ} that Σ defines. The proof that is given here is based on the proof given in [40], adjusted for the morpho-language. First, we proof that every pure $\mathcal{H}(E)$ formula is di-persistent. We then prove that, for every set of morpho formulas Σ, $K^{+}_{\mathcal{H}(E)}M\Sigma$ is complete for the set of discrete, two-sorted general frames that it defines. We assume that the reader is familiar with the terms di-persistence, general frames, discreteness, and descriptiveness. For reference, see [40, 14].

Proposition D.0.1 *Every pure $\mathcal{H}(E)M$ formula is di-persistent.*

Proof. Contraposition is used to prove the above proposition. Given a pure formula φ and a discrete two-sorted general frame $\mathcal{F}$, suppose that $\mathcal{F}, \mathcal{V}, w \not\Vdash \varphi$ for some arbitrary valuation $\mathcal{V}$ and some world w. We want to create an admissible valuation $\mathcal{V}'$ such that $\mathcal{F}, \mathcal{V}', w \not\Vdash \varphi$.

We define $\mathcal{V}'$ as follows. For each nominal i, $\mathcal{V}'(i) = \mathcal{V}(i)$. For every $p \in$ PROP we define $\mathcal{V}'(p) = X$ for some arbitrary admissible set X. Because φ is pure the valuation of the propositional variables does not affect the satisfiability. Furthermore, because $\mathcal{F}$ is discrete, every singleton set is admissible. Hence, for each nominal i, $\mathcal{V}(i)$ is admissible. Combined with the fact that for all the propositional variables an admissible set was chosen, $\mathcal{V}'$ is admissible. All that is left is to check whether $\mathcal{F}, \mathcal{V}', w \not\Vdash \varphi$. Because $\mathcal{V}'$ and $\mathcal{V}$ agree on the nominals and φ is a pure nominal this is automatically the case. QED

Before we can continue we first need a more general completeness theorem for a general normal modal logic $K_{\mathcal{M}}$.

Theorem D.0.2 *Let Γ be any set of modal formulas. $K_{\mathcal{M}}\Gamma$ is complete for the class of descriptive general frames defined by Γ.*

Proof. This result is a direct consequence of theorem 5.69 from [14]. QED

We then use the previous Theorem to prove a similar completeness result for a normal hybrid logic $K_{\mathcal{H}(E)}M$.

Proposition D.0.3 *Let Σ be a set of $\mathcal{H}(E)M$ formulas. $K_{\mathcal{H}(E)}M\Sigma$ is sound and strongly complete for the class of descriptive two-sorted frames defined by Σ.*

Proof. We only proof completeness. Temporarily, we shall adopt a purely modal perspective on $\mathcal{H}(H)M$ for this proof. Nominals are treated like constants and the global modality is seen as a normal modal operator. A non-standard frame will get the following structure $\mathcal{F} = (W, C, R, I, R_E, (S_i)_{i\in\text{NOM}})$ where R_E is a binary relation on W interpreting the modality E and $S_i \subseteq W$ interprets the nominal i. Non-standard general frames and non-standard models are defined in the same manner.

Suppose that Γ is a $K_{\mathcal{H}(E)}M\Sigma$ consistent set of $\mathcal{H}(E)M$ formulas. Then by Theorem D.0.2, Γ is satisfiable on a descriptive non-standard general frame $\mathcal{F} = (W, C, R, I, R_E, (S_i)_{i\in\text{NOM}}, A)$ such that $\mathcal{F} \Vdash K_{\mathcal{H}(E)}M$. Without loss of generality, we may assume that $\mathcal{F}$ is point-generated.

Recall that $K_{\mathcal{H}(E)}M$ contains the distribution axiom and necessitation rule for E, as well as the following axioms

$p \to Ep$	$\forall x\, R_E xx$
$EEp \to Ep$	$\forall xyz\, (R_E xy \wedge_E yz \to R_E xz)$
$p \to AEp$	$\forall xy\, (R_E xy \to R_E yx)$
$\otimes p \to Ep$	$\forall xy\, (Rxy \to R_E xy)$
$p \hat{\oplus} q \to Ep \wedge Eq$	$\forall xyz\, (Cxyz \to R_E xy \wedge R_E xz)$
Ei	$\forall x \exists\, y(R_E xy \wedge S_i y)$
$E(i \wedge p) \to A(i \to p)$	$\forall xyz\, (R_E xy \wedge R_E xz \wedge S_i y \wedge S_i z \to y = z)$

Each of the axioms is in Shalqvist form. Their first-order correspondents are indicated as well. By d-persistence, each of these formulas is valid on the underlying (non-standard) Kripke frame of $\mathcal{F}$. Together with the fact that $\mathcal{F}$ is point generated, this implies that $R_E = W \times W$ and $|S_i| = 1$ for each $i \in \text{NOM}$.

Let $\mathcal{F}' = (W, C, R, I, A, B)$ with $B = \bigcup_{i\in\text{NOM}} S_i$. Using the fact that $K_{\mathcal{H}(E)}M\Sigma$ is closed under substitution it is easy to show that $\mathcal{F}' \Vdash \Sigma$. As for the satisfiability of Γ. Suppose that $(\mathcal{F}, V), w \Vdash \Gamma$ for some world w and admissible valuation V. Define V' to be a valuation such that $V'(p) = V(p)$ for $p \in \text{PROP}$ and $V'(i) = S_i$ for $i \in \text{NOM}$. Because every $S_i \in \mathbb{A}$, V' is admissible as well. It is obvious that $(\mathcal{F}, V'), w \Vdash \Gamma$.

Finally, $\mathcal{F}'$ is a descriptive two-sorted general frame. QED

Lemma D.0.4 *The following rule is derivable in $K^+_{\mathcal{H}(E)}M\Sigma$:*

If $\vdash E(i \wedge j_1 \hat{\oplus} j_2) \wedge E(j_1 \wedge \varphi_1) \wedge E(j_2 \wedge \varphi_2) \to \psi$ then $\vdash E(i \wedge \varphi_1 \hat{\oplus} \varphi_2) \to \psi$, provided that $i \neq j_1$ or $i \neq j_2$ and j_1 and j_2 do not occur in φ or ψ.

Proof. Suppose that $K^+_{\mathcal{H}(E)}M \vdash E(i \wedge j_1 \hat{\oplus} j_2) \wedge E(j_1 \wedge \varphi_1) \wedge E(j_2 \wedge \varphi_2) \to \psi$. By propositional reasoning, we can write the latter formula as $E(i \wedge j_1 \hat{\oplus} j_2 \wedge E(j_1 \wedge \varphi_1) \to (E(j_2 \wedge \varphi_2) \to \psi)$ which is thus provable in $K^+_{\mathcal{H}(E)}M$. Using (Paste$_{E_{L\hat{\oplus}}}$) we have that $K^+_{\mathcal{H}(E)}M \vdash E(i \wedge \varphi_1 \hat{\oplus} j_2) \to (E(j_2 \wedge \varphi_2) \to \psi)$. By propositional reasoning the latter formula can be written as $E(i \wedge \varphi_1 \hat{\oplus} j_2) \wedge E(j_2 \varphi_2) \to \psi$ which is is thus provable in $K^+_{\mathcal{H}(E)}M$. Using (Paste$_{E_{R\hat{\oplus}}}$) we have that $K^+_{\mathcal{H}(E)}M \vdash E(i \varphi_1 \hat{\oplus} \varphi_2) \to \psi$. QED

Lemma D.0.5 *Every $K^+_{\mathcal{H}(E)}M\Sigma$-consistent set Γ can be extended to a maximal $K^+_{\mathcal{H}(E)}M\Sigma$-consistent set Γ^+ such that*

1. *One of the elements of Γ^+ is a nominal*
2. *For all $E(i \wedge \otimes\varphi) \in \Gamma^+$ there is a nominal j such that $E(i \wedge \otimes j) \in \Gamma^+$ and $E(j \wedge \varphi) \in \Gamma^+$*
3. *For all $E(i \wedge \varphi \hat{\oplus} \psi) \in \Gamma^+$ there are nominals j_1 and j_2 such that $E(i \wedge j_1 \hat{\oplus} j_2) \in \Gamma^+$, $E(j_1 \wedge \varphi) \in \Gamma^+$ and $E(j_2 \wedge \psi) \in \Gamma^+$.*

Proof. First, we extend the language with new nominals in such a way that we can ensure that a countably infinite number of nominals do not occur in Γ, while preserving consistency. With $(i_n)_{n\in\mathbb{N}}$ we denote a enumeration of a countable infinite set of nominals not occurring in Γ, and by $(\varphi_n)_{n\in\mathbb{N}}$ we denote an enumeration of all the $\mathcal{H}(E)M$-formulas in the extended language.

We first extend Γ with a new nominal such that property 1. of Γ^+ is accounted for. So, let Γ^0 denote $\Gamma \cup \{i_0\}$. That Γ^0 is consistent can be seen through the following. Suppose, towards a contradiction, that Γ^0 is not consistent. Then there exist $\psi_1, ..., \psi_n \in \Gamma$ such that $\vdash_{K_{\mathcal{H}(E)}M} i_0 \to \neg(\psi_1 \wedge ... \wedge \psi_n)$. Since i_0 does not occur in $\psi_1, ..., \psi_n$, by (name) it is the case that $\vdash_{K_{\mathcal{H}(E)}M} \neg(\psi_1 \wedge ... \wedge \psi_n)$, telling us that Γ is inconsistent. But this cannot be, since we assumed that Γ was consistent. Hence Γ^0 must be consistent.

Next, for $k \in \mathbb{N}$, define Γ^{k+1} as follows. If $\Gamma^k \cup \{\varphi_k\}$ is $K^+_{\mathcal{H}(E)}M\Sigma$-inconsistent, then $\Gamma^{k+1} = \Gamma^k$. If $\Gamma^k \cup \{\varphi_k\}$ is $K^+_{\mathcal{H}(E)}M\Sigma$-consistent however, then

1. $\Gamma^{k+1} = \Gamma^k \cup \{\varphi_k\}$ if φ_k is not of the form $E(i \wedge \otimes\psi)$ or $W(i \wedge \psi_1 \hat{\oplus} \psi_2)$.
2. $\Gamma^{k+1} = \Gamma^k \cup \{\varphi_k, E(i \wedge \otimes j), E(j \wedge \psi\}$ if φ_k is of the form $E(i \wedge \otimes\psi)$ with j the first new nominal not occurring in Γ^k or φ_k.
3. $\Gamma^{k+1} = \Gamma^k \cup \{\varphi_k, E(i \wedge j_1 \hat{\oplus} j_2), E(j_1 \wedge \psi_1), E(j_1 \wedge \psi_2)\}$ if φ_k is of the form $E(i \wedge \psi_1 \hat{\oplus} \psi_2)$ with j_1, j_2 the first new nominals not occurring in Γ^k or φ_k.

Each step preserves consistency. If Γ^K is $K^+_{\mathcal{H}(E)}M\Sigma$-consistent, then so is Γ^{k+1}. The first step is trivial. For the second and third case it is shown below that they also preserve consistency.

As for the second case, let $\Gamma^k \cup \{\varphi_k\}$ be $K^+_{\mathcal{H}(E)}M\Sigma$-consistent and suppose that φ_k is of the form $E(i \wedge \otimes\psi)$. Furthermore, suppose that, towards a contradiction, $\Gamma^{k+1} = \Gamma^k \cup \{\varphi_k, E(i \otimes j), E(j \wedge \psi\}$ is $K^+_{\mathcal{H}(E)}M\Sigma$-inconsistent. Then there are $\psi_1, ..., \psi_n \in \Gamma^k$ such that $\vdash_{K_{\mathcal{H}(E)}M} (E(i \wedge \otimes j) \wedge E(j \wedge \psi)) \to \neg(\psi_1 \wedge ... \wedge \psi_n)$. It follows by (Paste$_{E_\otimes}$) that $\vdash_{K_{\mathcal{H}(E)}M} \varphi_k \to \neg(\psi_1 \wedge ... \wedge \psi_n)$, contradicting the fact that $\Gamma^k \cup \{\varphi_k\}$ is $K^+_{\mathcal{H}(E)}M\Sigma$-consistent.

As for the third case, let $\Gamma^k \cup \{\varphi_k\}$ be $K^+_{\mathcal{H}(E)}M\Sigma$-consistent and suppose that φ_k is of the form $E(i \wedge \psi_1 \hat{\oplus} \psi_2)$. Furthermore, suppose that, towards a contradiction, $\Gamma^{k+1} = \Gamma^k \cup \{\varphi_k, E(i \wedge j_1 \hat{\oplus} j_2), E(j_1 \wedge \psi_1, E(j_1 \wedge \psi_2)\}$ is $K^+_{\mathcal{H}(E)}M\Sigma$-inconsistent. Then there are $\xi_1, ..., \xi_n \in \Gamma^k$ such that $\vdash_{K^+_{\mathcal{H}(E)}M\Sigma} (E(i \wedge j_1 \hat{\oplus} j_2) \wedge E(j_1 \wedge \psi_1) \wedge E(j_2 \wedge \psi_2)) \to \neg(\xi_1 \wedge ... \wedge \xi_n)$. It follows by Lemma D.0.4 that $\vdash_{K^+_{\mathcal{H}(E)}M\Sigma} \varphi_k \to \neg(\xi_1 \wedge ... \wedge \xi_n)$. But this contradicts the fact that $\Gamma^k \cup \{\varphi_k\}$ is $K_{\mathcal{H}(E)}M$-consistent. Thus it must be the case that Γ^{k+1} is $K_{\mathcal{H}(E)}M$-consistent.

Because consistency is preserved in every stage, $\Gamma^+ = \bigcup_{n<\omega} \Gamma^n$ is $K_{\mathcal{H}(E)}M$-consistent as well.

QED

Where in modal logic descriptiveness is a useful concept, in hybrid logic there do not exist a lot of descriptive formulas. Therefore, we must extend our definition of descriptiveness with the following addition, giving us the notion of strongly descriptive.

Definition D.0.6 Strongly descriptive two-sorted general frame A two-sorted general frame $(W, C, R, I, \mathbb{A}, \mathbb{B})$ is strongly descriptive if it is descriptive and it satisfies the following further conditions:

1. For all $X \in \mathbb{A}$, if $X \neq \emptyset$ then $X \cap \mathbb{B} \neq \emptyset$.
2. For all $X \in \mathbb{A}$, if $\{v \in X | v \in I\} \neq \emptyset$ then $\{v \in X | v \in I\} \cap \mathbb{B} \neq \emptyset$.

3. For all $X \in \mathbb{A}$ and $w \in \mathbb{B}$, if $\{v \in X | Rwv\} \neq \emptyset$ then $\{v \in X | Rwv\} \cap \mathbb{B} \neq \emptyset$.
4. For all $X, Y \in \mathbb{A}$ and $w \in \mathbb{B}$, if $\{v \in X | Cwvv' \wedge v' \in Y\} \neq \emptyset$ and $\{v' \in Y | Cwvv' \wedge v \in X\} \neq \emptyset$ then $\{v \in X | Cwvv' \wedge v' \in Y\} \cap \mathbb{B} \neq \emptyset$ and $\{v' \in Y | Cwvv' \wedge v \in X\} \cap \mathbb{B} \neq \emptyset$.

Definition D.0.7 *$dsf\mathcal{F}$* Given a strongly descriptive two-sorted general frame $\mathcal{F} = (W, C, R, I, \mathbb{A}, \mathbb{B})$, let $dsf\mathcal{F} = (\mathbb{B}, (C \cap \mathbb{B} \times \mathbb{B} \times \mathbb{B}), (R\mathbb{B} \times \mathbb{B}), (I \cap \mathbb{B}), \mathbb{B})$.

It is clear from the definition that each $dsf\mathcal{F}$ is discrete.

Proposition D.0.8 *For all strongly descriptive two-sorted general frames $\mathcal{F}$ and $\mathcal{H}(E)M$-formulas φ, $\mathcal{F} \Vdash \varphi$ iff $dsf\mathcal{F} \Vdash \varphi$*

Proof.
$\Rightarrow$: We prove this direction via contraposition. Let $\mathcal{F} = (W, C, R, I, \mathbb{A}, \mathbb{B})$ be a strongly descriptive two-sorted general frame, and suppose that $(dsf\mathcal{F}, V), v \nVdash \varphi$ for some admissible valuation V and world $v \in \mathbb{B}$. Let V' be any admissible valuation for $\mathcal{F}$ such that $V(p) = V'(p) \cap \mathbb{B}$ for $p \in$ PROP and $V(i) = V'(i)$ for $i \in$ NOM. From the definition of $dsf\mathcal{F}$ it can be seen that such a valuation exists. Because $\mathcal{F}$ is strongly descriptive it must be the case that for each non-empty $V(p)$, $V(p) \cap \mathbb{B}$ is non-empty as well. A straightforward induction argument establishes that for all $\mathcal{H}(E)M$-formulas ψ, $(\mathcal{F}, V'), v \Vdash \psi$ iff $(dsf\mathcal{F}, V), v \Vdash \psi$. Only the cases for $e, \otimes\xi$, $\xi_1 \hat{\oplus} \xi_2$ are shown here:

- $\psi = e$. Suppose that $(\mathcal{F}, V'), v \Vdash e$. Because $v \in \mathbb{B}$ it is also the case that $(\mathcal{F}, V), v \Vdash e$ and hence $(dsf\mathcal{F}, V), v \Vdash e$. The other direction follows from the fact that $V(e) \subseteq V'(e)$.
- $\psi = \otimes\xi$. Suppose that $(\mathcal{F}, V'), v \Vdash \otimes\xi$. This means that there is some $w \in W$ such that $(v, w) \in R$ and $w \in V'(\xi)$. Thus $\{w \in V'(\xi) | (v, w) \in R\}$ is non-empty. Hence, there must be a world $v' \in \mathbb{B}$ such that $v' \in V'(\xi)$ and $(v, v') \in R$. Thus $(\mathcal{F}, V'), v' \Vdash \xi$. By induction it must be the case that $(dsf\mathcal{F}, V), v' \Vdash \xi$, thus $(dsf\mathcal{F}, V), v \Vdash \otimes\xi$. The other direction follows from the fact that $V(e) \subseteq V'(e)$.
- $\psi = \xi_1 \hat{\oplus} \xi_2$. Suppose that $(\mathcal{F}, V'), v \Vdash \xi_1 \hat{\oplus} \xi_2$. This means that there are some $w, w' \in W$ such that $(v, w, w') \in C$, $w \in V'(\xi_1)$ and $w' \in V'(\xi_2)$. Thus $\{w \in V'(\xi_1) | (v, w, w') \in C \wedge v' \in V'(\xi_2)\}$ and $\{w' \in V'(\xi_2) | (v, w, w') \in C \wedge v \in V'(\xi_1)\}$ are non-empty. Hence there are $v', v'' \in \mathbb{B}$ such that $(v, v', v'') \in C$, $v' \in V'(\xi_1)$ and $v'' \in V'(\xi_2)$. Thus we can conclude that $(\mathcal{F}, V'), v' \Vdash \xi_1$ and $(\mathcal{F}, V'), v'' \Vdash \xi_2$. By induction it is the case that $(dsf\mathcal{F}, V), v' \Vdash \xi_1$ and $(dsf\mathcal{F}, V), v'' \Vdash \xi_2$, giving us that $(dsf\mathcal{F}, V), v \Vdash \xi_1 \hat{\oplus} i_2$. The other direction follows from the fact that $V(\xi_1) \subseteq V'(\xi_1)$ and $V(\xi_2) \subseteq V'(\xi_2)$.

It follows that $(\mathcal{F}, V'), v \nVdash \varphi$, and hence $\mathcal{F} \nVdash \varphi$.

$\Leftarrow$: We again prove this direction via contraposition. Let $\mathcal{F} = (W, C, R, I, \mathbb{A}, \mathbb{B})$ be a strongly descriptive two-sorted general frame, and suppose that $(\mathcal{F}, V), w \nVdash \varphi$ for some admissible valuation V and world $w \in W$. It follows from the first property in definition D.0.6 that $(\mathcal{F}, V), v \nVdash \varphi$ for some $v \in \mathbb{B}$. Let V' be a valuation such that $V'(p) = V(p) \cap \mathbb{B}$ for $p \in$ PROP and $V'(i) = V(i)$ for $i \in$ NOM. By definition V' is an admissible valuation for $dsf\mathcal{F}$. Next, a straightforward induction argument shows that for all $\mathcal{H}(E)M$-formulas ψ and for all worlds $u \in \mathbb{B}$, $(\mathcal{F}, V), u \Vdash \psi$ iff $(dsf\mathcal{F}, V'), u \Vdash \psi$. Only the cases for $e, \otimes\xi$ and $\xi_1 \hat{\oplus} \xi_2$ are shown here:

- $\psi = e$ Given an arbitrary $u \in \mathbb{B}$, assume that $(\mathcal{F}, V), u \Vdash e$. This means that $u \in V(e)$. By definition $u \in V'(e)$ and thus $(dsf\mathcal{F}, V'), u \Vdash e$. The other direction follows from the fact that $V'(e) \subseteq V(e)$.

- $\psi = \otimes\xi$. Given an arbitrary $u \in \mathbb{B}$, assume that $(\mathcal{F}, V), u \Vdash \otimes\xi$. This means that there is a $u' \in W$ such that $(u, u') \in R$ and $(\mathcal{F}, V), u' \Vdash \xi$. Thus $\{v \in V(\xi) | (u, v \in R)\} \neq \emptyset$. Therefore there must by a $u'' \in \mathbb{B}$ such that $(u, u'') \in R$ and $u'' \in V(\xi)$. Thus, $(\mathcal{F}, V), u'' \Vdash \xi$. by induction we have that $(dsf\mathcal{F}, V'), u'' \Vdash \xi$ and thus $(dsf\mathcal{F}, V'), u \Vdash \otimes\xi$. As for the other direction, suppose that $(dsf\mathcal{F}, V'), u \Vdash \otimes\xi$. This means that there is a $u' \in \mathbb{B}$ such that $(u, u') \in R$ and $u' \in V'(\xi)$. By induction we also have that $(\mathcal{F}, V'), u' \Vdash \xi$, and thus $(\mathcal{F}, V'), u \Vdash \otimes\xi$.

- $\psi = \xi_1 \hat{\oplus} \xi_2$. Given an arbitrary $u \in \mathbb{B}$, assume that $(\mathcal{F}, V), u \Vdash \xi_1 \hat{\oplus} \xi_2$. This means that there are $u', u'' \in W$ such that $(u, u', u'') \in C$, $(\mathcal{F}, V), u' \Vdash \xi_1$ and $(\mathcal{F}, V), u'' \Vdash \xi_2$. Thus $\{v \in V(\xi_1) | (u, v, v') \in C \wedge v' \in V(\xi_2)$ and $\{v' \in V(\xi_2) | (u, v, v') \in C \wedge v \in V(\xi_1)$ are non-empty. This means that there are $v, v' \in \mathbb{B}$ such that $(u, v, v') \in C$, $(\mathcal{F}, V), v \Vdash \xi_1$ and $(\mathcal{F}, V), v' \Vdash \xi_2$. By induction it is the case that $(dsf\mathcal{F}, V'), v \Vdash \xi_1$ and $(dsf\mathcal{F}, V'), v' \Vdash \xi_2$, thus $(dsf\mathcal{F}, V'), u \Vdash' xi_1 \hat{\oplus} \xi_2$. As for the other direction, assume that $(dsf\mathcal{F}, V'), u \Vdash \xi_1 \hat{\oplus} \xi_2$. This means that there are $u', u'' \in \mathbb{B}$ such that $(u, u, u'') \in C$, $u' \in V(\xi_1)$ and $u'' \in V(\xi_2)$. By induction we have that $(\mathcal{F}, V'), u' \Vdash \xi_1$ and $(\mathcal{F}, V'), u'' \Vdash \xi_2$. Thus it is the case that $(\mathcal{F}, V'), u \Vdash \xi_1 \hat{\oplus} \xi_2$.

It follows that $(dsf\mathcal{F}, V'), v \nVdash \varphi$ and thus $dsf\mathcal{F} \nVdash \varphi$. QED

Theorem D.0.9 *Let Σ be a set of $\mathcal{H}(E)M$-formulas. $K^+_{\mathcal{H}(E)}M\Sigma$ is strongly sound and complete for the class of strongly descriptive two-sorted general frames defined by Σ.*

Proof. Let Γ be an arbitrary $K^+_{\mathcal{H}(E)}M\Sigma$-consistent set of formulas. Let Γ^+ be the maximal consistent set extending Γ obtained from Lemma D.0.5. Through Proposition D.0.3 we obtain a descriptive two-sorted general frame $\mathcal{F}$ such that $\mathcal{F} \Vdash \Sigma$ and Γ^+ is satisfiable on $\mathcal{F}$. Remains to prove that $\mathcal{F}$ is strongly descriptive. First, we can assume that the frame used in proposition D.0.3 is a point generated subframe of the canonical (non-standard) general frame, generated from Γ^+. Thus, by definition $\mathbb{A} = \{\hat{\psi}|$ for ψ a $\mathcal{H}(E)M$ consistent formula$\}$, where $\hat{\psi}$ is the set of MCSs that contains ψ. Furthermore, by definition of $\mathcal{F}$, for each nominal i there is exactly one MCS that contains i, denoted by Γ_i, and for each nominal i, $\Gamma_i \in \mathbb{B}$.

By Lemma D.0.5, for each ψ there is a MCS containing a nominal, thus for each $X \in \mathbb{A}$ there is a $\Gamma \in X$ such that Γ contains a nominal j, thus $\Gamma = \Gamma_j$. This means that $X \cap \mathbb{B} \neq \emptyset$ if $X \neq \emptyset$.

As for the second property, this is proven via controposition. So, given some $X \in \mathbb{A}$ and $w \in \mathbb{B}$ with $i \in w$, suppose that $\{v \in X | (w, v) \in R\} \cap \mathbb{B} = \emptyset$. This means that there is no $v \in \mathbb{B}$ such that $j \in v$ and $\otimes j \in w$. By definition of $\mathcal{F}$ it cannot be the case that $E(i \wedge \otimes j) \in \Gamma^+$, thus it cannot be the case that $E(i \wedge \varphi) \in \Gamma^+$ for some φ. This means that w cannot have any successors and therefore $\{v \in X | (w, v) \in R\} = \emptyset$ as well.

The third property is also proven via controposition. Given some $X, Y \in \mathbb{A}$ and $w \in \mathbb{B}$ with $i \in w$, suppose that $\{v \in X | (w, v, v') \in C \wedge v' \in Y\} \cap \mathbb{B} = \emptyset$ and $\{v' \in Y | (w, v, v') \in C \wedge v \in X\} \cap \mathbb{B} = \emptyset$. This means that there are no $v, v' \in \mathbb{B}$ with $j_1 \in v$ and $j_2 \in v'$ for $j_1, j_2 \in \text{NOM}$ such that $j_1 \hat{\oplus} j_2 \in w$. This means that it cannot be the case that $E(i \wedge j_1 \hat{\oplus} j_2) \in \Gamma^+$. By construction of Γ^+ it thus is the case that there are no formula φ_1, φ_2 such that $E(i \wedge \varphi_1 \hat{\oplus} \varphi_2)$, thus w does not have any C-successors. Therefore $\{v \in X | Cwvv' \wedge v' \in Y\} = \emptyset$ and $\{v' \in Y | Cwvv' \wedge v \in X\} = \emptyset$.

QED

Theorem D.0.10 $K^+_{\mathcal{H}(\mathcal{E})}M\Sigma$ *is strongly sound and complete for the class of discrete two-sorted general frames defined by Σ, where Σ is any set of $\mathcal{H}(E)M$ formulas.*

Proof. Let Γ be any $K_{\mathcal{H}(E)}M$-consistent set of formulas. Pick a new nominal i. By the $(Name)$ rule, $\Gamma \cup \{i\}$ is $K_{\mathcal{H}(E)}M$-consistent as well. Hence, by Proposition D.0.9, $\Gamma \cup \{i\}$ is satisfiable on a strongly descriptive two-sorted general frame $\mathcal{F} = (W, C, R, I, \mathbb{A}, \mathbb{B})$ with $\mathcal{F} \Vdash \Sigma$. Let $(\mathcal{F}, V), w \Vdash \Gamma \cup \{i\}$ for V an admissible valuation. Note that $w \in \mathbb{B}$. Let V' be the valuation such that $V'(p) = V(p) \cap \mathbb{B}$

for $p \in$ PROP and $V'(i) = V(i) \cap \mathbb{B}$ for $i \in$ NOM. Let $dsf\mathcal{F}$ be the discrete subframe of f defined by definition D.0.7. By the same proof used in Proposition D.0.8 it is the case that $(dsf\mathcal{F}, V'), w \Vdash \Gamma$. By Proposition D.0.8 it is the case that $dsf\mathcal{F} \Vdash \Sigma$. Hence Γ is satisfiable on the class of discrete two-sorted general frames defined by Σ. QED

Through the combination of D.0.1 and Theorem D.0.10 we have the following.

Corollary D.0.11 *Let Σ be any set of pure $\mathcal{H}(E)M$-formulas. Then $K^{+}_{\mathcal{H}(E)}M\Sigma$ is strongly complete for the class of frames defined by Σ.*

Bibliography

[1] M. Aiello and B. Ottens. The mathematical morpho-logical view on reasoning about space. In *International Joint Conference on Artificial intelligence (IJCAI) 2007*, 2007. To appear.

[2] M. Aiello, I. Pratt-Hartmann, and J. van Benthem, editors. *Handbook of Spatial Logics*. Kluwer–Springer, 2007. To appear.

[3] M. Aiello and J. van Benthem. Logical patterns in space. In *Words, proofs and Diagrams*. CSLI publications, Stanford, 1999.

[4] M. Aiello and J. van Benthem. A modal walk through space. *Journal of Applied Non-Classical Logics*, 12(3/4):319–363, 2002.

[5] James F. Allen. Maintaining knowledge about temporal intervals. *Commun. ACM*, 26(11):832–843, 1983.

[6] C. Areces, H. de Nivelle, and M. de Rijke. Resolution in modal, description and hybrid logic. *Journal of Logic and Computation*, 11(5):717–736, 2001. Special Issue on Hybrid Logics. C. Areces and Blackburn, P. (eds.).

[7] C. Areces and D. Gorín. Ordered resolution with selection for h(@). In F. Baader and A. Voronkov, editors, *LPAR 2004: 11th Conference on Logic for Programming, Artificial Intelligence and Reasoning*, volume 3452 of *LNCS*, pages 125–141, Montevideo, Uruguay, 2004. Springer.

[8] C. Areces and J. Heguiabehere. Hylores: A hybrid logic prover based on direct resolution. In *Proceedings of Advances in Modal Logic*, Toulouse, France, 2002.

[9] L. Bachmair and H. Ganzinger. Equational reasoning in saturation-based theorem proving. In Wolfgang Bibel and Peter H. Schmidt, editors, *Automated Deduction: A Basis for Applications. Volume I, Foundations: Calculi and Methods*. Kluwer Academic Publishers, Dordrecht, 1998.

[10] B. Bennett. Modal logics for qualitative spatial reasoning. *Journal of the Interest Group in Pure and Applied Logic (IGPL)*, 4(1), 1996.

[11] Brandon Bennett, Anthony G. Cohn, Paolo Torrini, and Shyamanta M. Hazarika. A foundation for region-based qualitative geometry. In *Proceedings of the European Conference on Artificial Intelligence*, pages 204–208, 2000.

[12] G. Birkhoff. *Lattice Theory*. Amer. Math. Soc., 1967. 3rd ed., Colloquium Publications.

[13] P. Blackburn. Representation, reasoning, and relational structures: A hybrid logic manifesto. *Logic Journal of the IGPL*, 8(3):339–365, 2000.

[14] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge University Press, 2001.

[15] I. Bloch. Modal logics based on mathematical morphology for qualitative spatial reasoning. *Journal of Applied Non-Classical Logics*, 12(3-4):399–424, 2002.

[16] I. Bloch, O. Colliot, and R. Marcondes Cesar Junior. On the ternary spatial relation "between". *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 36(2):312–327, 2006.

[17] D. Brand. Proving theorems with the modification method. *SIAM J. Computing*, 4:412 – 430, 1975.

[18] S. Burris and H. Sankappanavar. *A Course in Universal Algebra*. Springer, 1981.

[19] A G Cohn and S M Hazarika. Qualitative spatial representation and reasoning: An overview. *Fundamenta Informaticae*, 46(1-2):1–29, 2001.

[20] A. G. Cohn and A. C. Varzi. Modes of connection. In *Spatial Information Theory - Cognitive and Computational Foundations of Geographic Information Science*, volume 1661, pages 299–314. Springer-Verlag, 1999.

[21] M. Cristani, A. G. Cohn, and B. Bennett. Spatial locations via morpho-mereology. In A. G. Cohn, F. Giunchiglia, and B. Selman, editors, *KR2000: Principles of Knowledge Representation and Reasoning*, pages 15–25, San Francisco, 2000. Morgan Kaufmann.

[22] E.R. Dougherty and R.A. Lotufo. *Hands-on Morphological Image Processingods in Imaging*. SPIE Press, Bellingham, WA, 2003.

[23] P. Enjalbert and L. F. del Cerro. Modal resolution in clausal form. *Theor. Comput. Sci.*, 65(1):1–33, 1989.

[24] M. Fujio and I. Bloch. Non-Classical Logic via Mathematical Morphology. Technical Report 2004D010, Ecole Nationale Supérieure des Télécommunications, dec 2004.

[25] V. Goranko and D. Vakareloc. Universal algebra and modal logic. In *Advances in Modal Logic, vol II*, pages 265–292. CLSI Publications, Stanford, California, 2000.

[26] H. Heijmans and C. Ronse. The algebraic basis of mathematical morphology: I. Dilations and erosions. *Computer Vision, Graphics, and Image Processing*, 50:245–295, 1990.

[27] O. Kutz, H. Sturm, N. Suzuki, F. Wolter, and M. Zakharyaschev. Axiomatizing distance logics. *Journal of Applied Non-Classical Logic*, 12(3-4):425–440, 2002.

[28] G. Matheron. *Elements pour une theorie des milieux poreux*. Masson, Paris, 1976.

[29] G. Matheron and J. Serra. The birth of mathematical morphology. In H. Talbot and R. Beare, editors, *Proceedings of VIth International Symposium on Mathematical Morphology*, pages 1–16, Sydney, Australia, 2002. Commonwealth Scientific and Industrial Research Organisation.

[30] D. A. Randell, Z. Cui, and A. Cohn. A spatial logic based on regions and connection. In B. Nebel, C. Rich, and W. Swartout, editors, *KR'92. Principles of Knowledge Representation and Reasoning: Proceedings of the Third International Conference*, pages 165–176. Morgan Kaufmann, San Mateo, California, 1992.

[31] D. A. Randell, Z. Cui, and A. G. Cohn. An interval logic for space based on connection. In *ECAI '92: Proceedings of the 10th European conference on Artificial intelligence*, pages 394–398, New York, NY, USA, 1992. John Wiley & Sons, Inc.

[32] J. A. Robinson. A machine-oriented logic based on the resolution principle. *J. ACM*, 12(1):23–41, 1965.

[33] J. Serra. *Image Analysis and Mathematical Morphology Vol I.* Ac Press, London, 1982.

[34] P. Soille. Grey scale convex hulls: definition, implementation, and application. In H. Heijmans and J. Roerdink, editors, *Mathematical Morphology and its Applications to Image and Signal Processing*, volume 12 of *Computational Imaging and Vision*, pages 83–90. Kluwer Academic Publishers, Dordrecht, 1998. Proc. of ISMM'98.

[35] P. Soille. *Morphological Image Analysis*. Springer-Verlag, 1999.

[36] O. Stock, editor. *Spatial and Temporal Reasoning*. Kluwer Academic Publishers, 1997.

[37] A. Tarski. Some theorems about the sentenial calculi of Lewis and Heyting. *Journal of Symboloc Logic*, 1948.

[38] A. Tarski. Sentential calculus and topology. In *Logic, Semantics, Meta-mathematics*. Oxford Clarendon Press, 1956.

[39] A. Tarski. What is elementary geometry. In L. Henkin, P. Suppes, and A. Tarski, editors, *The Axiomatic Method, with Special Reference to Geometry and Physics*, pages 16–29. Studies in Logic and the Foundations of Mathematics, 1959.

[40] B. ten Cate. *Model theory for extended modal languages*. PhD thesis, University of Amsterdam, 2005. ILLC Dissertation Series DS-2005-01.

[41] J. van Benthem. *The logic of time*. Reidel, Dordrecht, 1983.

[42] J. van Benthem, H. van Ditmarsch, J. Ketting, and W. Meyer Viol. *Logica voor Informatici*. Addison-Wesley, Amsterdam, 2003.

[43] Y. Venema. A crash course in arrow logic. In Maarten Marx, Lázló Pólos, and Michael Masuch, editors, *Arrow Logic and Multimodal Logic*, pages 3–34. CLSI Publications, Stanford, California, 1996.

[44] A. Voronkov. Algorithms, datastructures, and other issues in efficient automated deduction. In *IJCAR '01: Proceedings of the First International Joint Conference on Automated Reasoning*, pages 13–28, London, UK, 2001. Springer-Verlag.

[45] F. Wolter and M. Zakharyaschev. A logic for metric and topology. *Journal of Symbolic Logics*, 70(3):795 – 828, 2005.