Monadic Semantics, Team Logics and Substitution

A Thesis for Master's Degree in Logic, 30 credits

Orvar Lorimer Olsson
Supervisor: Fredrik Engström

UNIVERSITY OF GOTHENBURG
Faculty of Humanities
Department of Philosophy, Linguistics and Theory of Science

Abstract

In this thesis we investigate the issue of non-substitutionality of propositional team logics, in particular propositional dependence logic, by semantic means.

We establish a general language to discuss semantic systems for logics fundamentally based on consensus of truth in a class of objects. Importantly we admit a notion of logic that does not assume closure under substitution. We show how to express traditional semantic systems using this general language and describe a condition of being *truth-compositional* defined using a notion of *defining sets* for formulas. We then give an account of *propositional dependence logic* as described by Yang and Väänänen [YV16], and investigated by Lück [Lüc20] and Quadrellaro [Qua21].This is done using *valuational team semantics* based on sets of valuations as semantic objects. For valuational team semantics in general we describe the property of *flatness* and prove that a flat valuational team semantics always is equivalent to a standard valuational semantics. We then formulate the problem of *non-substitutionality* for logics based on the semantics and describe some of the consequences it has for the development and investigation of team logics.

The main part and contribution of this thesis develops a new semantic framework we call *monadic semantics*, constructed by considering a *universe* for which propositional variables are interpreted as unary predicates by *monadic models* and the interpretation of complex formulas are governed by a *monadic frame*. We show how monadic semantics are sufficient to express every logic with truth-compositional semantics, in the sense defined, by constructing *monadic dual semantics*. We also define the *maximal* and the *full* semantics of a monadic frame. We show how a notion of *monadic team semantics* corresponding to Yang and Väänänens construction can be identified as a particular type of monadic semantics and translate the results regarding flatness to this setting. With this semantic notion defined we utilise a definiton of *interpretation sets* for formulas to express notions of independence and generality of atoms in a semantics. With this in place we can prove connections between these properties of a semantics, and substitutionality of the logic it defines. In this way we can give a full semantic categorisation of whether a logic is closed under singular substitution, and both a necessary and a sufficient condition for unrestricted substitution in any logic with monadic semantics. As a direct consequence these categorisations imply that every maximal monadic semantics that correspond to a semantics of Yang and Väänänen's construction is either flat, or its logic is not closed under substitution. We reveal that the main reason for this is the treatment of atomic formulas in valuational team semantics.

In the last part of the thesis we use monadic semantics and introduce *natural teamifications* to construct a semantics defining a new logic we call *boolean teamified boolean logic*, `BTB`. We complement this by exhibiting a set of axioms that, when assumed in `BTB`, gives an up to typographical renaming conservative extension of propositional dependence logic. We also hint on strong connections between `BTB` and Girard's *linear logic* [GLR95].

Acknowledgements

I want to sincerely thank my supervisor Fredrik Engström. He introduced me to team semantics and after his initial suggestion of topic, I have pursued the work leading to this thesis. From my initial ideas he has helped me to hone in a suitable level of abstraction to adequately express my results, and asked the right questions to help bridge the gap between vague intuitions and formal proofs. I am also grateful, as the reader should also be, of his willingness to spend time on proof-reading, correcting typos, give typesetting suggestions, and point out when formulations were lacking in clarity or correctness. His efforts have helped me lift this text to a level of quality that otherwise would not have been possible. With that said, any mistakes or errors still remaining are solemnly my responsibility.

I further thank my examiner Simon Dobnik for his time, comments and interesting discussion from a different perspective.

I would also like to thank my fellow thesis writers in *The Thesis Trinity*: Josephine Dik and Johanna Wolff. Apart from the willingness to give comments on drafts of this thesis and a enjoyable opposition, we have shared two years of studies together. The social and mental support it has provided, throughout the years and in the final writing process, has been invaluable.

Finally I want to express my gratitude to the rest of masters and doctoral students, lecturers and professors in the logic group at Gothenburg University. I am honoured to have been a part of this warm community filled with care, coffee and companionship.

Contents

Chapter 1

Background and introduction

The modern study of the notion of *logic* has reached far beyond Aristotelian syllogisms. Many different approaches to the subject throughout the ages have not only shaped the language and methods in the theory of logic as it stands today, but also what it means when we say that system of deduction or notion of truth entailment is *logical*. Today a logician may study many types of logics with different prominent properties and motivations. A modern approach to logic uses so called *team semantics* with both first-order and propositional counterparts. These compositional semantics give rise to logics that can include notions of *dependence* and *independence* of variables in its language. Team semantics have been studied extensively in recent years, and a range of connections to other logics and fields of studies have been investigated [Abr+16]. These logics, however, are not closed under substitution, which is a necessary criterion for many treatments of the notion of logic [CE19]. In this thesis we study the propositional variant of team semantics. We seek to understand the issue of non-substitiutonality for logics with these semantics, and if it is possible to generalise the notion of team semantics in order to find a substitutional logic closely related to typical propositional team logics described in the literature [YV16; YV17]. We manage to do so by introducing what we call *monadic semantics*. This first chapter serves as an introduction to the main subject matters of this work, some historical background to how this discussion fits in the broader setting, and finally an overview of the structure of this thesis.

1.1 Truth semantics

Truth semantics is part of what defines a logic. The semantics gives some sort of interpretation or setting for which the truth of statements can be evaluated. A common construction of semantics is to collect a set of examples for which statements may hold true or not. Truth is then defined as the consensus of the set of examples so that a statement is considered true if and only if it is true for every example. This formulates a very open definition of what a semantics may be, and it is useful in practice to restrict a discussion to a more restricted class of semantics. Some restrictions, such as that of compositionality, ensures some computational simplicity, and others, such as that of algebraicity, allows for a strong mathematical categorisation of the semantic objects.

1.1.1 Compositional semantics

In a broader discussion of semantics for languages, there are many types of semantic notions and many methods to construct and motivate certain descriptions of semantics. For logical languages a prominent approach with history at least as far back as to Frege [PW10], is to utilise the often recursive nature of logical languages to define a *compositional semantics*. The idea is that the semantic meaning of a complex formula should be determined by the semantic meaning of its constituents. The exact definition of compositionality differs in different treatments, and there is a general discussion within linguistics to what extent and in exactly what way natural languages are compositional [PW10; Hod98]. For constructed logical languages, compositional semantics are desirable: Partly because it marries nicely with the often recursively defined formal languages, and partly for the increased computational simplicity of defining and determining the value of such semantics. The strive for a computational semantics was according to Hintikka [Hin98] a leading principle in Tarski's formulation of truth semantics for first-order logic and later developments in the founding of Model theory.

1.1.2 Algebraic semantics

The use of algebraic structures as a vehicle for semantic interpretations and computations of logics was pioneered by George Boole in the middle of the 19th century [Boo47] and followed prominent logicians and mathematicians such as Frege, Tarski, Heyting, and Halmos to name a few. The first uses of algebras in logic was as semantic models describing logical systems. When the field of *algebraic logic* matured in through the 20th century, the connection between logics and algebras was further used to fruitfully import results regarding the mathematical structures to the field of logic. A modern continuation of this type of study has resulted in the field of *abstract algebraic logic*, which focuses to a further extent on describing the nature of the connection between algebras and logic, and bares a lot of resemblance to the general approach to algebra called *universal algebra* [Fon16]. A result of all this work is that it can be very informative to find an algebraic description of a new logic under investigation. If this is possible, a vast machinery of theorems and methods, not only from logic but from the whole history mathematics, can become available.

1.2 Logic and substitutionality

Substitutionality, sometimes called *structurality*, is central to the notion of a logic. It is the property of a logic that if an entailment in a formal language is *true*, then it is true even if its atomic building blocks are reinterpreted to other formulas. This property is related to the idea that a statement that is *logically true* is so due to the *structure* of the statement rather than the content of its atoms. The idea of substitutionality is historically so central to the notion of logic that many scholars determine it essential for a system to be correctly coined *a logic*[Fon16; CE19]. Even though non-substitutional systems have been described, the centrality of this assumption bears with it that many of the general proofs, machinery and methods for analysing logics developed throughout history use substitutionality as an undisputed property. Regardless of any philosophical motivations, this makes it more desirable to work with substitutional logics.

1.3 Team semantics

The invention of team semantics is attributed to Hodges[AV09]. It springs from a longer discussion about how quantification works in natural language, mathematics and first-order logic. The intellectual development that brought this field to life can be prompted by observing how quantifiers may or may not interact in first-order logic. Consider the following statement in relation to group theory.[1]

$$\forall x \exists y \forall z \exists w (xzy = zw) \tag{1.1}$$

If we spell out the quantifiers in an English statement we get:

"For every element x we can choose an element y such that for every element z we can choose an element w (knowing x, y, z) such that the equation $xzy = zw$ holds true"

This is obviously true in any group structure, since for an arbitrary choice of x, y, z we can choose $w = z^{-1}xyz$. This choice is inherently dependent on all the values x, y, z, but this is not a problem, for at the stage when the choice of w is to be made, the interpretations of x, y, z are already fixed and known. The linear nature of first-order formulas enforces this *handing down* of information from the outer quantifiers to the inner.

If we instead demand that the choice of w should be independent of x and y, the story becomes very different. A mathematical way of expressing this criteria is that we can find a functional relation between z and w established independent of x and y. This can easily be expressed in second-order logic, allowing quantification over functions, as

$$\exists f \forall x \exists y \forall z (xzy = zf(z)). \tag{1.2}$$

The truth for this formula is significantly different from that of expression (1.1). Expression (1.2) is not true for group structures in general[2], but it can be established for certain classes. If the group is commutative (so that $xzy = xyz$), then we can choose $f \equiv 1$ and $y = x^{-1}$. The property of a group satisfying (1.2) is thus non-trivial and in particular significantly different from (1.1). We resorted

[1] The main importance is the structure of the quantifiers. Anyone not familiar with group theory hopefully can follow the discussion anyway

[2] As a simple counter example consider the symmetry group of an equilateral triangle S_3

to second-order logic to express the interdependency of the quantifiers in the statement. We need to allow quantification to not always depend on each other in a linear way, but be allowed to be treated independently, or in parallel to capture this type of statement. To be able to express this type of quantification general it is not enough with first-order logic [Hen61]. Since there are other drawbacks in using full second-order logic, it is desirable to find a smaller extension that can express this type of statements.

The first efforts in trying to do so is ascribed to Henkin who in the 1960's extended first-order logic with *branching quantifiers* also known as *Henkin quantifiers* [Hen61]. Our example sentence could then be written as

$$\begin{pmatrix}\forall x \exists y\\ \forall z \exists w\end{pmatrix}(xyz = zw)$$

This can be read as the quantifications $\forall x \exists y$ and $\forall z \exists w$ being treated in parallel with no interdependency. Formally the semantics is given using *Skolem functions* to achieve a sentence in the same style as (1.2).

In the end of 1980's Hintikka and Sandu [HS89] presented a new semantics based on games of imperfect information, and together with that a new notation using *slashed* quantifiers. They called this logic *Independence friendly* (IF-) Logic. In Hodges' version of the syntax [Hod07], called *slash logic*, a slashed quantifier is denoted on the form $(\exists x/y_1, \dots y_n)$, and interpreted as "there is a choice of x independent of knowledge of $y_1 \dots y_n$". In this notation our example statement would be written

$$\forall x \exists y \forall z (\exists w/x, y)(xzy = zw).$$

The game semantics given by Hintikka is however not compositional. He believed that no compositional semantics was possible for any logic that can treat quantifier independence.[3] He argued that this type of branching quantifiers are vital in game theory and also prominent in natural language, and took this as a reason to revise the logical foundations of mathematics in first-order logic, and step away from Tarski-type compositional definitions of truth [Hin98].

Soon after however, Hodges proved Hintikka wrong in this regard by presenting *Trump semantics* based on Hintikkas game theoretic semantics. By evaluating formulas for *trumps* being collection of *deals* that can be read as restricted set of variable assignments he manage to express a compositional semantics for IF-Logic.

Later Väänänen reinterpreted the setting and replaced the slashed quantifiers with a dependence predicate $\mathrm{dep}(x_1, \dots x_n, y)$ expressing that the value of y functionally depends on $x_1, \dots, x_n$, allowing the notion of dependence to be expressed separately from the quantification. In this notation our example sentence would be written

$$\forall x \exists y \forall z \exists w (xzy = zw \wedge \mathrm{dep}(z, w)).$$

He reinterpreted Hodges' semantics calling it *team-semantics* based on truth evaluation for *teams* where the members are valuations of variables. Väänänen's approach has been proven flexible enough to interpret several types of logics and extensions. Thus, considerable interest and a lot of work has been done by identifying different logics and analysing their complexities [Abr+16].

1.4 Propositional team semantics

Even though team semantics is historically motivated by first-order logic it has also been formulated for other logical systems, most important for us is propositional logics as developed by Yang and Väänänen [Yan14; YV16; YV17]. In this setting it is the valuations of propositions that form teams and the dependence of propositional truth values that are considered instead.

To express the notion of a propositional variable being dependent of other variables they introduce the *depenence atom* $=(p_{i_1}, \dots, p_{i_n}, p_j)$. To give semantics for this dependence atom it is not enough to evaluate for a single valuation. For every valuation every value is constant, and hence

[3]Hintikka and Sandu acknowledged that it is proven that it is possible to define compositional grammars for every language that can be described by finite means, and that such constructions can produce compositional semantics for IF logic. However, the semantics that are generated by these methods are considered, by Hintikka and Sandu, to be *unnatural* since they contain complex expressions treated as primitives [HS97].

every proposition is trivially dependent of every other valuation. In order to give a proper semantics we must lift the semantics to subsets of valuations called *teams*.

Consider the example of the following table defining a set of valuation $X = \{ s_1, \dots, s_4 \}$ of the propositional variables $p_1, \dots, p_4$:

	p_1	p_2	p_3	p_4
s_1	1	1	0	0
s_2	0	1	0	1
s_3	1	0	0	0
s_4	0	0	0	1

This can be interpreted as the four different tests of four different properties, or four different individuals votes on four different topics, etc. In this example we can see that for the collection of valuations X we have that all valuations that agree on p_1 also agree on p_4, but this does not hold for p_1 and p_2. Using dependence atoms we can state that $=(p_1, p_4)$ is true for the team X, but $=(p_1, p_2)$ is false. To give a logic for these types of dependence statements is what motivates the construction of propositional team semantics, and the standard semantic is constructed by truth for valuation teams as described above.

1.4.1 The problem

Propositional team semantics as presented by Yang and Väänänen is truth-compositional, but it exhibits other undesirable shortcomings. It expresses forms of propositional logics endowed with the capability of identifying interdependence between atomic propositions, and it may not come as a surprise that these logics fail to be substitutional. The first formulations of propositional team logic uses specific types of related atoms and are a priori not substitutional. But even when this is evaded, we see that all proper team logics described fail to be substitutional [Qua21; Lüc20]. This is a an obstacle in the study of these logics, since it renders them without many methods and techniques that are developed with substitutional logics in mind. Work has been done to try to investigate or mitigate these problems. Noticeable is Quadrellaro's work [Qua21] producing an *almost algebraisation* of a family of team logics.

1.5 The structure an contribution of this thesis

In this thesis we seek to understand the issue of non-substitutionality for propositional team logics and see if we can find ways to mitigate it. Our investigation will bring us to define new types of semantics where substitutionality of the logic can be connected to properties of its semantics. The resulting work can be divided into three parts spanning two chapters each:

First we set the stage. In Chapter 2 we formulate suitable notion of *semantics* and *logics*, such that they include standard semantic constructions, and are suitable for a discussion around *truth-compositionality* and *substitutionality*. In Chapter 3 we make an account of Yang and Väänänen's propositional team semantics, but given in our general notation for semantics, and describe valuational team semantics for a variant of propositional dependence logic. We further relate valuational team logics to single valuation semantics using the notion of *flatness* as investigated by Lück [Lüc20]. We then finish this part by presenting the non-substitutionality of our propositional dependence logic, and summarise some of the work that has been done on algebraisation and generalisations of team logics and semantics.

In the second part we make new tools to answer our questions. In Chapter 4 we define what we will call *monadic semantics*. These semantics can be seen as describing a universe viewed from a set of possible interpretations of the nature of its content. We show that this class of semantics is general enough to include logically equivalent semantics for standard Kripke frame semantics and valuational semantics, but also Yang and Väänänens team semantics. For the special case of *monadic team sematics* we further translate the main results from Chapter 3 to the new semantic setting. Chapter 5 includes the main results of this thesis. We introduce the notion of *interpretation sets* for the set of possible interpretations of a formula and use this to show that substitutional logics necessarily have monadic semantics with the certain distinguishable properties. As a consequence we observe that it is a consequence of the construction of Yang and Väänänens propositional team semantics, that none of their interesting logics can be substitutional. The categorisation of substitutionality in semantic criteria will not be complete and we close the chapter by reiterating the main results and listing the open problems still to be solved.

In the third part we use our results and look forward. In Chapter 6 we use the broader notion of monadic team and define a notion of *natural teamification*. We use this to construct a new substitutional team logic that has a direct connection to our account of propositional dependence logic. We also manage to give axioms recovering the propositional dependence logic and point to a possible strong connection between the new logic and Girard's linear logic [GLR95]. The last chapter concludes the thesis by reflecting on our main results and suggesting some possible extensions and generalisations that may be investigated in future work.

Throughout this thesis, a strong familiarity with the basics of standard set theory and mathematical notation is assumed.

Chapter 2

Language, semantics and logics

In this chapter we seek to give definitions for a general terminology to be able to discuss different semantics and logics in a coherent way. This will also outline the scope of the discussion of this thesis. To illustrate the definitions we will give some examples of semantics and logics that hopefully are familiar and should help to understand the notation and terminology. These examples will also be referred to throughout the thesis and be the starting point for upcoming constructions. With this in mind I encourage the reader to familiarise themselves with these examples even if the definitions and results in this chapter feel easy to follow.

2.1 Language

Any definition of a formal logical system starts by describing a formal language. Formal languages are strictly defined and it should be decidable what is a proper sentence of the language, also called a well formed formula. Different types of logics are formulated in different types of languages such as propositional logics constructed from atomic propositions and logical connectives, or first-order logics with terms, functions, predicates, connectives and quantifiers. The type of language will highly influence what type of logics that are describable.

This paper is focused on propositional logics. Since we are focused on compositional semantics we will restrict ourselves to recursively generated languages. A main focus is also to investigate substitutionality of the constructed logics. If a logic is substitutional, then any atom can take the place of any other atom and thus all atoms are, in respect to the logic, interchangeable or of the same *sort*. Since this will be our focus we restrict ourselves to investigate *one-sorted* languages in the sense that there is only one sort of atomic formulas. We fix a countable set of propositon symbols $\{p_i\}_{i\in\mathbb{N}}$ as atomic formulas in all languages described. We also define some additional terminology used when describing formulas.

Definition 2.1. We define a propositional language $\mathcal{L}$ by the following:

- The signature of $\mathcal{L}$ denoted $S(\mathcal{L})$ is a set of connectives with fixed arities, e.g. $(1^0, \neg^1, \to^2, \Box^1)$.
- The *set of formulas* of the language $\mathcal{L}$, also denoted by $\mathcal{L}$ and the context determines what is intended, is defined as the set of well formed formulas generated by a countable set of atomic symbols p_i, and the connectives of the signature of the language. this is given as an inductive definition by the following:
  - $p_i \in \mathcal{L}$ for all $i \in \mathbb{N}$ and are called *atomic formulas* or *propositional variables*
  - if $\psi_1 \dots \psi_n \in \mathcal{L}$ and $*^n \in S(\mathcal{L})$, then $*(\psi_1 \dots, \psi_n) \in \mathcal{L}$ and is called *a (complex) formula of $\mathcal{L}$*. We call $\psi_i \dots \psi_n)$ the *principal subformulas* of the complex formula, and $*^n$ is its *principal connective*.
- A *subformula* of a formula $\phi \in \mathcal{L}$ is defined recursively as follows: ψ is a subformula of ϕ if $\psi = \phi$, or ψ is a subformula of any of the principal subformulas of ϕ.
- The *(atomic) support* of a formula ϕ, denoted $\mathrm{Sup}(\phi)$, is the set of atomic formulas that appear in the formula, i.e. the atoms that are subformulas of ϕ. The support of a set of formulas $\mathrm{Sup}(\Gamma)$ is the set of atoms that support any of the formulas in Γ.
- The signature of a formula ϕ, denoted $\mathrm{Sing}(\phi)$ is the set of connectives that appear in the formula, i.e. the principal connectives of some subformula of ϕ.
- A *$*^n$-free formula* is a formula without the connective $*^n$ in its signature.
- The *$*^n$-free fragment of $\mathcal{L}$* is the set of $*^n$-free formulas in $\mathcal{L}$.

Throughout this work, lower case Greek letters ($\phi, \psi, \chi, \gamma \dots$) will be used to denote single formulas, and uppercase Greek letters ($\Gamma, \Delta, \Xi, \dots$) denote sets of formulas. $*^n$ is used to denote an arbitrary connective of arity n.

When a language is described, if the arities of the connectives are understood from the context they are omitted. Furthermore, we will often drop excessive parenthesis and use infix notation when it is customary. Thus we write $\phi \wedge \psi$ instead of $\wedge^2(\phi, \psi)$

Note 2.2. We use use the word *signature* to refer to the set of logical connectives. This is in contrast to how the word is used in first-order logic. A propositional language $\mathcal{L}$ can however also be viewed as an algebraic structure. From this perspective the set of formulas $\mathcal{L}$ constitutes the *free term algebra* of the connectives in the sense that two term are equal only if they are syntactically equal, and the set of connectives form the algebra signature of this algebra [Fon16]. We will partly use this notion of $\mathcal{L}$ as an algebra, called the *formula algebra* of the language, and draw a lot of connections to algebraic considerations throughout. Since we do not consider first-order logics in any meaningful extent and without suitable alternative, we have chosen to use the word *signature* in the algebraic sense despite the possible confusion.

2.1.1 Substitution

Substitution can often be a bit of a struggle to define formally in a concise way, in particular when there is need to handle variable bindings in formulas, or for simultaneous substitution of many terms. In the case of propositional languages however there is a very nice way to define substitution using an algebraic perspective, used in the abstract algebraic study of logics [Fon16]:

A mapping f between two algebras with the same signature is called a *homomorphism* if it distributes over all connectives, that is

$$f(*^n(\phi_1, \dots, \phi_n)) = *^n(f(\phi_1), \dots, f(\phi_n))$$

for all operations $*^n$ in the signature. If f is a homomorphism from and to the same algebra, it is known as an *endomorphism*. Consider the effect of an endomorphism $\sigma : \mathcal{L} \to \mathcal{L}$ on the formula algebra of a propositional language. By being a homomorphism, σ is fully defined by how it maps the atomic formulas p_i, and by $\mathcal{L}$ being a free term algebra, σ will map a formula ϕ to the formula obtained by simultaneously substituting every atom p_i in ϕ by the formula $\sigma(p_i)$. This shows that, for our purposes, the endomorphisms on the formula algebra exactly captures the behaviour of formula substitution.

Definition 2.3. [Fon16] For a propositional language $\mathcal{L}$, a *substitution* is an endomorphism $\sigma : \mathcal{L} \to \mathcal{L}$. For each $\phi \in \mathcal{L}$ the image formula $\sigma(\phi)$ is known as a *substitution instance* of ϕ. For a set of formulas Γ, we will write $\sigma(\Gamma)$ for the set of substitution instances. We will consequently use the letters σ and ρ to denote substitutions. When there is no risk of confusion, it is standard to omit the parenthesis and write $\sigma\phi$ and $\sigma\Gamma$.

This definition of substitution allows a substitution of an infinite number of variables simultaneously. For a single formula ϕ only a finite number of them will matter, but for a set of formulas Γ an infinite number of changes may take effect. It is useful to consider restricted forms of substitution such as singular substitution.

Definition 2.4. Given a language $\mathcal{L}$ a substitution $\sigma : \mathcal{L} \to \mathcal{L}$ is *a singular substitution* if there is an index j such that

$$\sigma p_i = p_i \quad \text{for all } i \neq j.$$

If $\sigma p_j = \psi$ we write $\phi(p_j/\psi)$ and $\Gamma(p_j/\psi)$ instead of $\sigma\phi$ and $\sigma\Gamma$.

2.2 Semantic truth

Throughout this thesis we will give several different meanings to the binary meta predicate $\vDash$. Since the usages are strongly related, and the different usages formally involve different domains, it is common to use the same symbol. We hope this definitional *overloading* is more informative than confusing, and that it should be clear from the context which definition applies.

We first give a general notion of semantics that will be used throughout this thesis. We separate the definition in two parts, the definiton of a semantic framework, and that of a semantics in such a framework.

Definition 2.5. A *semantic framework* $\mathscr{F}$ of a language $\mathcal{L}$ is a relation between a *semantic domain of semantic objects* $\mathscr{O}$ and the formulas of the language:

$$\mathscr{F} \subseteq \mathscr{O} \times \mathcal{L}.$$

For a semantic object $\mathbb{O} \in \mathscr{O}$, and formula $\phi \in \mathcal{L}$ we write $\mathscr{F} : \mathbb{O} \models \phi$ when $(\mathbb{O}, \phi) \in \mathscr{F}$ and say that ϕ *is true for* $\mathbb{O}$ by $\mathscr{F}$. Otherwise ϕ is *not true* for $\mathbb{O}$, and we write $\mathscr{F} : \mathbb{O} \not\models \phi$. For a set of formulas Γ we write $\mathscr{F} : \mathbb{O} \models \Gamma$ to mean that $\mathscr{F} : \mathbb{O} \models \gamma$ for all $\gamma \in \Gamma$. A semantic framework with semantic domain $\mathscr{O}$ may be referred to as *an $\mathscr{O}$-semantic framework*. If the choice of framework is understood from the context it is omitted from the notation.

Definition 2.6. An *$\mathscr{O}$-semantics* $\mathscr{S} = (\mathscr{F}, O)$ is an $\mathscr{O}$-semantic framework $\mathscr{F}$ together with a subset $O \subseteq \mathscr{O}$ of the semantic domain. For sets of formulas Γ and singular formulas ϕ, we define the *entailment of the semantics*, denoted $\mathscr{S} : \Gamma \models \phi$, by the following.

$$\mathscr{S} : \Gamma \models \phi \quad \text{if and only if for all } \mathbb{O} \in O: \text{ if } \mathscr{F} : \mathbb{O} \models \Gamma, \text{ then } \mathscr{F} : \mathbb{O} \models \phi.$$

When O is the full semantic domain of $\mathscr{F}$ we say that $\mathscr{S}$ is the semantics of the framework. When the choice of semantics is clear from the context it is omitted from the notation.

A truth semantics of this kind can be seen as a collection of semantic objects, that serve as individual counter examples for the truth of false formulas. The negative use of examples to exhibit falsehood is as old as Aristotle. The positive statement of truth, when no admissible example refutes, goes back at least to Tarski's formulations of truth semantics. These ideas about the role of a semantics are commonplace in many discussions about logic [CE19].

To illustrate what a semantics of this kind can be, we give some familiar examples. These will also be building blocks for further constructions throughout this thesis.

Valuation-semantics and classical propositional logic

The role of the semantic objects is to associate a truth value to the formulas in the language. The simplest and most direct type of semantics is *valuation-semantics*, were the semantic objects are a choice of truth value for each atomic proposition p_i, and the framework defines truth for complex formulas recursively. To express this formally we define a *valuation* as a characteristic function.

Definition 2.7. A *valuation* is a function $s : \mathbb{N} \to \{0, 1\}$. Let $\mathscr{V}$ denote the set of all truth valuations.

Using this we can give a semantics for classical propositional logic. We will denote this logic by CPL and therefore denote its valuational semantics by V_{CPL}.

Definition 2.8. Let $\mathcal{L}$ be the language with signature $(\bot, \neg, \wedge, \vee)$. We define the $\mathscr{V}$-semantics V_{CPL} on $\mathcal{L}$ as the semantics of the framework given by the following definitions:

- $s \models p_i$ if and only if $s(i) = 1$ for all propositional variables p_i,
- $s \not\models \bot$,
- $s \models \neg\phi$ iff $s \not\models \phi$,
- $s \models \phi \wedge \psi$ iff $s \models \phi$ and $s \models \psi$, and
- $s \models \phi \vee \psi$ iff $s \models \phi$ or $s \models \psi$.

This is the standard interpretation of these connectives, and a very commonly described semantics for classical propositional logic.

Kripke Semantics

Kripke semantics is a type of semantics that is commonly used in many branches of logic. I first state the definition of Kripke semantics the way it could be described in a textbook on modal logic (this is a mix of the definitions in [Zac19] and [CF08]). The semantic idea is conveyed by picturing a set of *possible worlds* with different valuations of atomic statements, together with a an accessibility notion describing how the worlds can access the truth in other worlds.

Definition 2.9 (Kripke Model)**.** A *Kripke model* M is a triple (W, R, V) where

- W is a nonempty set of *worlds*,
- R is a binary *accessibility relation* on W, and
- V is a function $V : W \to \mathscr{V}$ assigning to each world $w \in W$ a (binary) valuation $v_w : \mathbb{N} \to \{0, 1\}$.

A *pointed Kripke model* (M, w) is a model M together with a choice of a specific world w.

The pointed models are the building-blocks of Kripke semantics. Consider a simple modal language with signature $(\bot, \neg, \wedge, \vee, \Box, \Diamond)$.

Definition 2.10 (Modal pointed truth)**.** The standard Kripke semantics definition of truth in a model for the language with signature $(\bot, \neg, \wedge, \vee, \Box, \Diamond)$ in pointed worlds, denoted $M, w \Vdash \phi$, is defined recursively by the following rules:

- $M, w \Vdash p_i$ iff $v_w(i) = 1$,
- $M, w \nVdash \bot$ for all worlds $w \in W$,
- $M, w \Vdash \neg\phi$ iff $M, w \nVdash \phi$,
- $M, w \Vdash \phi \wedge \psi$ iff $M, w \Vdash \phi$ and $M, w \Vdash \psi$,
- $M, w \Vdash \phi \vee \psi$iff $M, w \Vdash \phi$ or $M, w \Vdash \psi$,
- $M, w \Vdash \Box\phi$ iff for all worlds w' such that $(w, w') \in R$: $M, w' \Vdash \phi$, and
- $M, w \Vdash \Diamond\phi$ iff there exists a world w' such that $(w, w') \in R$ and $M, w' \Vdash \phi$.

We write $M \vDash \phi$ if $M, w \vDash \phi$ for all $w \in M$, and $M, w \Vdash \Gamma$ if $M, w \Vdash \gamma$ for all $\gamma \in \Gamma$.

With this notion of truth we can define two different types of semantics called local and global semantics based on truth in whole models or in pointed models.

Definition 2.11. For a Kripke model M we say that Γ *entails* ϕ *locally* written

$$M : \Gamma \vDash_l \phi \text{ if and only if for all points } w \in M : \text{ if } M, w \Vdash \Gamma, \text{ then } M, w \Vdash \phi.$$

We say that M *entails* ϕ *globally* written

$$M : \Gamma \vDash_g \phi \text{ if and only if : when } M \Vdash \Gamma \text{ then } M \Vdash \phi.$$

A *global or local* Kripke semantics is then defined as the consensus of a class of Kripke models $\mathcal{M}$

$$\mathcal{M} : \Gamma \vDash_\# \phi \text{ if and only if } M : \Gamma \vDash_\# \phi \text{ for all } M \in \mathcal{M}$$

where # denotes l or g respectively.

It is directly clear that both global and local Kripke semantics form semantics according to our definition, where the semantic domains are the set of Kripke models and the set of pointed Kripke models, respectively.

Definition 2.12. The set of rules given in Definition 2.10 define, a *local Kripke framework* for a set of pointed Kripke models by setting

$$M, w \vDash \phi \text{ iff } M, w \Vdash \phi,$$

and a *global Kripke framework* for a set of Kripke models by setting

$$M \vDash \phi \text{ iff } M \Vdash \phi.$$

A *local or global Kripke sematics* is a semantics in a local or global Kripke framework respectively.

Note that in our definition, a local Kripke semantics need not be defined using a class of pointed Kripke models that exhaust all points in the included models. If the accessibility relation is a partial order we can for example define a semantics for only initial worlds $w \in M$ with respect to the order.

Algebraic frameworks

Algebraic structures have been used to interpret logics in many ways. Boolean Algebras and Heyting algebras for classical and intuitionistic logic respectively can be seen as prime examples of this type of investigation [GLR95]. There are many different ways an algebra can play a role for the semantics of a logic, and different authors use the notion of an *algebraic semantics* slightly differently. We will put our definition in relation to the terminology used by Font in his presentation of *Abstract Algebraic Logic* [Fon16].

Abstract Algebraic Logic can be seen as the effort of categorising the possible connections between logics and algebraic structures. It is done in a way that general results for properties of algebras as described in *universal algebra* can be translated into results regarding properties of associated logics. The establishment of such translations are called *bridge theorems*. In this investigation Font defines the notion of an *algebraic matrix*, and the logic it defines.

Definition 2.13. An *algebraic matrix* is a pair $(\mathcal{A}, T)$ where $\mathcal{A}$ is an algebra and and $T \subseteq |\mathcal{A}|$ is a subset of its elements. The logic of $(\mathcal{A}, T)$ for the language $\mathcal{L}$ with the same signature as the algebra, is the entailment relation $\vdash$ for $\mathcal{L}$ defined as follows:

$$(\mathcal{A}, T) : \Gamma \vdash \phi \text{ if and only if for all homomorphisms } h : \mathcal{L} \to \mathcal{A} : \text{ if } h(\Gamma) \subseteq T, \text{ then } h(\phi) \in T.$$

We can reinterpret this into our general notions of semantics by first describing the type of semantic domain, and then essentially let an algebraic matrix play the role of an *algebraic semantic framework*.

Definition 2.14. Let $\mathcal{A}$ be an algebra of the same signature as a propositional language $\mathcal{L}$. Then we define the following notions

- An $\mathcal{A}$-valuation is a function $h : \mathbb{N} \to \mathcal{A}$.
- For an $\mathcal{A}$-valuation h, let $h^* : \mathcal{L} \to \mathcal{A}$ denote the algebra homomorphism that is induces by h through the following definition for propositional variables

$$h^*(p_i) = h(i).$$

- An *algebraic framework* is a tuple $(\mathcal{A}, T)$ where $\mathcal{A}$ is an algebra and and $T \subseteq |\mathcal{A}|$ is a subset of its elements called the *Truth set* of the framework. Its semantic domain is the set of $\mathcal{A}$-valuations, and $(\mathcal{A}, T) : h \vDash \phi$ is defined as follows:

$$(\mathcal{A}, T) : h \vDash \phi \text{ if and only if } h^*(\phi) \in T.$$

- An *algebraic semantics* is a semantics in an algebraic framework.

Note that the logic of an algebraic framework $(\mathcal{A}, T)$ is identical to its logic when viewed as an algebraic matrix. The above definition of *algebraic semantics* does not however correspond to how the word is used by Font [Fon16].

We now move to give some notions to be able to analyse, categorise, and later compare different semantic frameworks and semantics.

2.2.1 Expressive power

One way to analyse a semantics is by, for each formula in the language, consider the sets of semantic objects for which it is true. In this way each formula ϕ defines a specific subset of the semantic domain, that can be seen as the semantic interpretation of the formula in the semantics. Inspired by Yang and Vännänen for the definition of this notion [YV16; YV17], we denote this set $[\![\phi]\!]$:

Definition 2.15. Given a semantics $\mathscr{S} = (\mathscr{F}, O)$ and a formula ϕ in its language let $[\![\phi]\!]_{\mathscr{S}}$ denote the set of semantic objects regarded by the semantics for which ϕ is true.

$$[\![\phi]\!]_{\mathscr{S}} = \{ \mathbb{O} \in O \mid \mathscr{F}, \mathbb{O} \vDash \phi \}$$

$[\![\phi]\!]_{\mathscr{S}}$ is called the set of objects *defined by* ϕ *in* $\mathscr{S}$.

For a set of formulas Γ we write $[\![\Gamma]\!]_{\mathscr{S}}$ for the collection of defined sets of objects by the formulas

$$[\![\Gamma]\!]_{\mathscr{S}} = \{ [\![\gamma]\!]_{\mathscr{S}} \bigm| \gamma \in \Gamma \}$$

We call this the collection of sets *defined by* Γ *in* $\mathscr{S}$, and the sets defined by $\mathcal{L}$, the sets *defined by the semantics*. When the semantics is clear from the context it is omitted from the notation.

From this definition we can directly see that we can equate the entailment notion of a semantics with a subset relation on its definable sets.

Theorem 2.16. *For a semantics $\mathscr{S} = (\mathscr{F}, O)$ we have the following equivalence*

$$\Gamma \vDash \phi \quad \textit{if and only if} \quad \bigcap_{\gamma \in \Gamma} [\![\gamma]\!] \subseteq [\![\phi]\!]$$

where $\bigcap \emptyset$ is defined as the full class of object O for the semantics.

The proof is direct from the definitions.

We will see how the identification of defined sets and expressibility is utilised by Yang and Väänänen, and Lück to compare semantics for different languages in Chapter 3. We will also use it when we construct new *monadic semantics* in Chapter 4, but first we will use it to help us define compositionality for semantic frameworks.

2.2.2 Compositionality

Exactly what the criteria of *compositionality* should mean is not fixed in the litterature. Pagin and Westerståhl accounts for several variants in their paper on the topic [PW10]. We choose to use what they call *the functional version of compositionality*. To state it fully in our setting we must first state a broader notion of semantics than just truth-semantics.

Definition 2.17. A *general semantics* for a language $\mathcal{L}$ is a function $\mu : \mathcal{L} \to \mathtt{M}$ from the formulas of the language to some set $\mathtt{M}$ of semantic meanings.

A general semantics μ is *compositional* if for every connective $*^n \in \mathrm{Sing}(\mathcal{L})$ there is a function $f_* : \mathtt{M}^n \to \mathtt{M}$ such that

$$\mu(*^n(\psi_1, \ldots, \psi_n)) = f_*(\mu(\psi_1), \ldots, \mu(\psi_n))$$

This type of compositionality is most clearly exemplified by the algebraic semantics described above, where the algebra can be viewed as an auxiliary set of meanings. However, as pointed out by Pagin and Westerståhl [PW10], if there is no restriction on the admittable sets of meaning, it is always possible do find a semantics that can be considered compositional in this sense. The notion of compositionality becomes more interesting if we restrict the possible sets of meaning in a noticeable way. Since we are focused on semantics for *truth* it is desirable with a useful notion of compositionality for truth. We can look to the expressiveness of a semantic framework to make this work. By associating the semantics of the framework with the defined-set function $[\![\cdot]\!]$ that can be viewed as a general semantics, we can identify a meaningful notion of *truth-compositionality* for a semantic framework.

Definition 2.18. A semantic framework $\mathscr{F}$ for the language $\mathcal{L}$ with semantic domain $\mathscr{O}$ is *truth-compositional* if for every connective $*^n \in \mathrm{Sing}(\mathcal{L})$ there is a function $f_* : \mathscr{P}(\mathscr{O})^n \to \mathscr{P}(\mathscr{O})$ such that

$$[\![*^n(\psi_1, \ldots, \psi_n)]\!] = f_*([\![\psi_1]\!], \ldots, [\![\psi_n]\!])$$

We may call the function f_* a *dual* to the connective $*^n$. A semantics with a truth-compositional framework is also called *truth-compositional.*

Note that the dual function is only uniquely determined on the sets expressed by the semantics, and thus it may not be unique as a function on $\mathscr{O}$. If however $[\![\mathcal{L}]\!]_{\mathscr{F}} = \mathscr{P}(\mathscr{O})$ it is unique and we confidently speak of f_* as *the dual* of $*^n$ for the framework $\mathscr{F}$

Theorem 2.19. *The semantics V_{CPL} and the local Kripke semantics as defined are truth-compositional.*

Proof. We note that truth in V_{CPL} and Kripke semantics for complex formulas are defined directly by the truth of the truth values of the principal subformulas for semantic object in a way completely determined by the principal connective of the formula. In the case of V_{CPL} it is easy to give explicit dual functions for the connectives as classic functions on sets:

$\bot :\ [\![\bot]\!] = \emptyset,$

$\neg :\ [\![\neg\phi]\!] = [\![\phi]\!]^C,$

$\wedge :\ [\![\phi \wedge \psi]\!] = [\![\phi]\!] \cap [\![\psi]\!]$, and

$\vee :\ [\![\phi \vee \psi]\!] = [\![\phi]\!] \cup [\![\psi]\!].$

These are the *boolean set operations*. □

Apart from categorising truth-functionality, we will come back to dual functions in Chapter 4 when we define a new type of semantics. In contrast to the above result we can observe that neither global Kripke semantics nor algebraic semantics are truth-compositional in general. This can be shown by the following counter-examples.

Example 2.20. We will construct a global Kripke semantics that is not truth-compositional. Consider a Kripke framework whose semantic domain $\mathscr{O}$ is the set of all models with two worlds w_1 and w_2 related with a singleton relation $R = \{(w_1, w_2)\}$, that is the relation from w_1 to w_2. This situation can be expressed by the following diagram:

$$w_1 \xrightarrow{R} w_2$$

Then for any such model M, if we evaluate the formula $\Box\bot$ we have

$$M, w_1 \nVdash \Box\bot \text{ but } M, w_2 \Vdash \Box\bot \text{ so that } M \nvDash \Box\bot.$$

Thus $[\![\Box\bot]\!] = \emptyset$. If we instead evaluate the formula $\Diamond\neg\bot$, we have the opposite results for the points, but the same truth value in the global model:

$$M, w_1 \Vdash \Diamond\neg\bot \text{ but } M, w_2 \nVdash \Diamond\neg\bot \text{ so that } M \nvDash \Diamond\neg\bot.$$

Hence $[\![\Diamond\neg\bot]\!] = \emptyset$. The conjunction of these formula however holds in both models, and thus is true in the global model:

$$M, w_1 \Vdash \Box\bot \vee \Diamond\neg\bot \text{ and } M, w_2 \Vdash \Box\bot \vee \Diamond\neg\bot \text{ so that } M \vDash \Box\bot \vee \Diamond\neg\bot.$$

Therefore $[\![\Box\bot \vee \Diamond\neg\bot]\!] = \mathscr{O}$. This means that any possible dual function $f_\vee$ of the disjunction $\vee$ will have to satisfy

$$f_\vee(\emptyset, \emptyset) = \mathscr{O}.$$

On the other hand however, if we consider $\Box\bot$ in disjunction with itself, it is clear that $M \nvDash \Box\bot \vee \Box\bot$ so that $[\![\Box\bot \vee \Box\bot]\!] = \emptyset$. This means that $f_\vee$ must simultaneously satisfy

$$f_\vee(\emptyset, \emptyset) = \emptyset$$

which of course is impossible. This proves that this global Kripke framework cannot be truth compositional.

Example 2.21. We can construct an algebraic framework that is not truth-compositional. We consider a language with the constant symbols $-1, 0$ and 1 together with a binary connective $*$. We then fix an algebra $\mathcal{A}$ with the three constants being distinct points and the binary function interpreted as standard multiplication. By choosing $\{1\}$ as truth set we get the algebraic framework $(\mathcal{A}, \{1\})$ with the set of $\mathcal{A}$-valuations $\mathscr{V}_{\mathcal{A}}$ as its semantic domain. Since we have included $-1, 0$ and 1 as constants in the language, their interpretations are fixed and do not change for any $\mathcal{A}$-valuation. We can therefore observe that $[\![-1]\!] = [\![0]\!] = [\![0 * 0]\!] = \emptyset$. However we also have $[\![-1 * -1]\!] = [\![1]\!] = \mathscr{V}_{\mathcal{A}}$, and with a similar argument as in the previous example, this proves that $(\mathcal{A}, \{1\})$ is not truth-compositional.

Note 2.22. Our notion of truth-compositionality is thus to some extent a restricted notion of substitutionality not including semantics that are evidently substitutional such as algebraic semantics. The reason for this is that in these semantics there is semantically important information lost when the $\mathcal{A}$-valuation for a formula is compressed to a binary value in relating it to the truth set of the semantics.

Observe however, that our definition of truth-compositionality makes it a property of the specific framework for a semantics. A semantics that is not truth-compositional may very well be practically identical to a truth-compositional one in an framework with an extended domain. We can for example start with an algebraic framework $(\mathcal{A}, T)$ with its semantic domain the set $\mathscr{V}_{\mathcal{A}}$ of $\mathcal{A}$-valuations. We can then extend this domain to $\mathscr{V}^+_{\mathcal{A}}$ by adding, for each $h \in \mathscr{V}_{\mathcal{A}}$, a marked copy of every element in the algebra, i.e.,

$$\mathscr{V}^+_{\mathcal{A}} = \mathscr{V}_{\mathcal{A}} \cup \{\, a_h \mid a \in |\mathcal{A}|,\ h \in \mathscr{V}_{\mathcal{A}} \,\}.$$

On this bigger domain we can define an extended framework $\mathscr{F}$ by defining truth for every new object a_h a formula to be true when the $\mathcal{A}$-valuation of the index induces a homomorphism for which ϕ is mapped to the element a of the algebra, i.e.,

$$\mathscr{F} : a_h \vDash \phi \ \text{ if and only if } \ h^*(\phi) = a$$

This framework on an extended domain does then have objects that *expose* the algebraic values of all valuations, and it is then possible to conjure up a dual function to every connective in the signature of the language, and thus the semantics $(\mathscr{F}, \mathscr{V}_{\mathcal{A}})$ which is practically identical to $(\mathcal{A}, T)$ can be shown to be truth-compositional.

For similar reasons we can also note that the full global Kripke framework, with every possible Kripke model in its semantic domain, can be technically considered truth-compositional. Since every subset of a Kripke model can be considered a Kripke model in its own right, we can create similarly elaborate dual functions as outlined for the algebraic case. In both these cases the resulting dual functions need to excessively utilise structures of the semantic domain not directly present in the formulation of the framework.

This all means however that our notion of truth-semantics is not stable under arbitrary extensions of the semantic domain. We still find it meaningful when we construct or discuss frameworks for a fixed intended semantic domain and preferably find well motivated dual functions.

2.2.3 Model existence and consistency

For a semantics $\mathscr{S} = (\mathscr{F}, O)$ consider a set of formulas Ξ that is not true according to any semantic object of the semantics, that is

$$\text{for all } \mathbb{O} \in O : \ \mathscr{F} : \mathbb{O} \nvDash \Xi.$$

Such set Ξ of formulas is said to be *inconsistent* in the semantics. We can observe that for any semantics as we have defined it, any inconsistent set of formulas entails every formula of the language:

$$\text{If } \Xi \text{ is inconsistent with } \mathscr{S} \text{ then } \mathscr{S} : \Xi \vDash \phi \text{ for all } \phi \in \mathcal{L}.$$

In many presentations the semantic framework will assign a constant connective $\perp$ that is false for every model. Then inconsistency is identified with the entailment of such constant:

$$\Xi \text{ is inconsistent with } \mathscr{S} \text{ if and only if } \mathscr{S} : \Xi \vDash \perp.$$

We want our semantic entailment notion to be able to express inconsistency without forcing the language necessarily to include an always inconsistent formula $\perp$. To express this we take inspiration from intuitionistic style of sequent calculus, and let inconsistency be denoted by an empty right hand side of the entailment symbol.

Definition 2.23. For every semantics $\mathscr{S}$ we extend the entailment notion by writing $\mathscr{S} : \Gamma \vDash$ with the right side empty whenever Γ is inconsistent in $\mathscr{S}$. We will refer to this entailment notion for a semantics as *the logic defined by the semantics*.

Notation. Throughout this thesis, when we write a general entailment $\Gamma \vDash \phi$, if it is not otherwise stated in the context, ϕ may be a formula, or the empty set. We also point out that for any substitution σ we have that $\sigma(\emptyset) = \emptyset$.

2.3 Logics

Since we restrict our notion of semantics to the above mentioned constructions, we can formulate properties of the entailment notions that arise from these constructions. This expresses a restriction on the class of logics that are relevant for us. In this thesis we will restrict our notion of logic to this class.

Theorem 2.24. *For every semantics, as described, the entailment $\vDash$ has the following properties:*

- *Identity/Reflexivity: If $\phi \in \Gamma$, then $\Gamma \vDash \phi$.*
- *Monotonicity: If $\Gamma \vDash \phi$ and $\Gamma \subseteq \Delta$, then $\Delta \vDash \phi$*
- *Cut/Transitivity: If $\Gamma \vDash \phi$ and $\Delta \vDash \gamma$ for every $\gamma \in \Gamma$, then $\Delta \vDash \phi$.*
- *Explosion: If $\Gamma \vDash$ then $\Gamma \vDash \phi$ for every formula $\phi \in \mathcal{L}$.*

The proof of these properties are direct from the *object class consensus* nature of the type of semantics we have defined.

Definition 2.25. Given a propositional language $\mathcal{L}$, a relation $\vdash$ between a set of formulas and none or one formula, i.e.,

$$\vdash \ \subseteq \mathscr{P}(\mathcal{L}) \times (\mathcal{L} \cup \{\emptyset\})$$

is said to be *a logic* if it satisfies the properties of $\vDash$ in Theorem 2.24. A set of formulas Γ is called *inconsistent* in the logic if $\Gamma \vdash$, otherwise it is *consistent*.

Theorem 2.26. *Every logic has a semantics.*

Proof. For every logic $\vdash$ we can define the *term semantics* for the logic as follows: The semantic domain of the term semantics, denoted $(\mathcal{L})_\vdash$ is the collection of all consistent sets of formulas with respect to the logic:

$$(\mathcal{L})_\vdash = \{ (\Gamma) \mid \Gamma \subseteq \mathcal{L} \text{ and } \Gamma \nvdash \} .$$

The parenthesis are added to discriminate between the semantic object (Γ) and the set of formulas Γ. We then define the term semantics to be the semantics of the framework $\mathrm{Term}(\vdash)$ defined by setting

$$\mathrm{Term}(\vdash) : (\Gamma) \vDash \phi \ \text{ if and only if } \ \Gamma \vdash \phi, \quad \text{for } \Gamma \text{ consistent in } \vdash .$$

It is clear by definition that for every Γ, ϕ we get for the semantics of the framework that

$$(\mathrm{Term}(\vdash), (\mathcal{L})_\vdash) : \Gamma \vDash \phi \ \text{ if and only if } \ \Gamma \vdash \phi.$$

This proves that $\vdash$ is the logic of the semantics $(\mathrm{Term}(\vdash), (\mathcal{L})_\vdash)$ □

This proves that our notion of logic directly corresponds to our notion of semantics. This is a weaker notion of logic than what is commonly used, since we do not demand that the logics are *substitutional*.

Definition 2.27. A logic $\vdash$ is *substitutional* when

$$\text{if} \quad \Gamma \vdash \phi \quad \text{then} \quad \sigma\Gamma \vdash \sigma\phi \ \text{ for every subsitution } \sigma \text{ on } \mathcal{L}.$$

A logic is *closed under singular substitutions* if the implication holds for all singular substitutions.

Substitutionality is also referred to as *structurality*. The notion *structure* has however a different prominent meaning in proof theory. The *structurality* or *sub-structurality* of a logic is then a property of a proof system, and is not related to if the logic is substitutional or not. To not confuse these notions, I prefer the more transparent terminology of *substitutionality* for the property we have in mind.

Substitutionality will be regarded a desirable property for a logic, but it will not be taken for granted. In Font's book on *abstract algebraic logic* [Fon16], substitutionality is an assumed property for all logics. It is then maybe not surprising that the logics of the semantics in our description that coincide with the logics of algebraic matrices as described by Font, are necessarily substitutional.

Theorem 2.28. *The logic of an algebraic framework is substitutional.*

Proof. Let $(\mathcal{A}, T)$ be an algebraic framework for the language $\mathcal{L}$. Assume that $(\mathcal{A}, T) : \Gamma \vDash \phi$. Then for all $\mathcal{A}$-valuations h,

$$\text{if } h^*(\Gamma) \subseteq T \text{ then } h^*(\phi) \in T.$$

Let σ be an arbitrary substitution on $\mathcal{L}$. We need to show that for every $\mathcal{A}$-valuations h,

$$\text{if } h^*(\sigma\Gamma) \subseteq T \text{ then } h^*(\sigma\phi) \in T.$$

For every h^* consider the composed homomorphism $h^* \circ \sigma$

$$h^* \circ \sigma(\phi) = h^*(\sigma(\phi))$$

then there is an $\mathcal{A}$-valuation σh such that $(\sigma h)^* = h^* \circ \sigma$. Then

$$(\sigma h)^*(\Gamma) = h^*(\sigma\Gamma), \text{ and } (\sigma h)^*(\phi) = h^*(\sigma\phi)$$

and thus by the assumption that $(\mathcal{A}, T) : \Gamma \vDash \phi$ we have that

$$\text{if } h^*(\sigma\Gamma) \subseteq T \text{ then } h^*(\sigma\phi) \in T.$$

This holds for all $\mathcal{A}$-valuation h and all substitutions σ on $\mathcal{L}$, and thus the logic of $(\mathcal{A}, T)$ is substitutional. □

2.3.1 Compactness and substitution

In our definition of a logic $\vdash$ it is defined for expressions $\Gamma \vdash \phi$ for arbitrary large sets Γ. In many logical systems only a finite subset of Γ actually plays a role for establishing the relation, and then by monotonicity it holds for the whole set. Such logics are called *compact*.

Definition 2.29. A logic $\vdash$ is *compact* if

$$\Gamma \vdash \phi \text{ if and only if there is a finite set } \Gamma' \subseteq \Gamma \text{ such that } \Gamma' \vdash \phi.$$

Compact logics are particularly well behaved, since Γ can practically be assumed to be finite. In particular they are substitutional if they are closed under singular substitution.

Theorem 2.30. *A compact logic is substitutional if and only if it is closed under singular substitution.*

Proof. Substitutionality always implies closed under singular substitution, so what is to be proven is the opposite direction.

Assume the logic $\vdash$ is compact and $\Gamma \vdash \phi$. Then there is some finite $\Gamma' \subseteq \Gamma$ such that $\Gamma' \vdash \phi$. Let σ be any substitution. Then since Γ' is finite, so is $\mathrm{Sup}(\Gamma, \phi)$. Therefore there exists a composition of finite number of singular substitutions $\sigma' = \rho_1 \circ \cdots \circ \rho_n$ such that

$$\sigma' p_i = \sigma p_i \ \text{ for all } \ p_i \in \mathrm{Sup}(\Gamma, \phi).$$

Thus if the logic is closed under singular substitution we get the following chain of implications

$$\Gamma' \vdash \phi \ \Rightarrow \ \rho_n\Gamma' \vdash \rho_n\phi \ \Rightarrow \ \ldots \ \Rightarrow \ \sigma'\Gamma' \vdash \sigma'\phi \ \Rightarrow \ \sigma\Gamma' \vdash \sigma\phi \Rightarrow \ \sigma\Gamma \vdash \sigma\phi$$

and the logic is substitutional. □

2.4 Different equivalences of semantics and logics

As part of our work we are defining new semantic frameworks to treat known logics. We are interested in when different semantics essentially describe the same logic. We can establish different kind of equivalence depending on if the logics and semantics share language or semantic domain.

2.4.1 Logically equivalent semantics

As part of our work we are defining new semantic frameworks to treat known logics. To express this relation we define the notion of logical equivalence of semantics:

Definition 2.31. Two semantics $\mathscr{S}, \mathscr{S}'$ for the same language are said to be *logically equivalent*, denoted $\mathscr{S} \equiv \mathscr{S}'$ if they define the same logic.

Logically equivalent semantics are semantics of the same language, but may have completely different semantic domains.

2.4.2 Translationally equivalent logics

Some notions of logics can be exhibited in multiple different languages. To be able to relate these logics we define the notion of *translationally equivalent*:

Definition 2.32. Two logics $\mathtt{L}, \mathtt{L}'$ for the languages $\mathcal{L}$ and $\mathcal{L}'$ respectively are *translationally equivalent*, denoted $\mathtt{L} \approx \mathtt{L}'$ if there exist *translations* $\tau : \mathcal{L} \to \mathcal{L}'$ and $\pi : \mathcal{L}' \to \mathcal{L}$ such that

$$\Gamma \vdash_{\mathcal{L}} \phi \text{ if and only if } \tau(\Gamma) \vdash_{\mathcal{L}'} \tau(\phi), \qquad \text{and} \qquad \Gamma \vdash_{\mathcal{L}'} \phi \text{ if and only if } \pi(\Gamma) \vdash_{\mathcal{L}} \pi(\phi).$$

Two translationally equivalent logics can in this sense exhibit each other via the translations, and are in this sense essentially the *same* logic. Note that in this definition the *translations* can be any function. When speaking of *translations between logics* it is sometimes assumed that translations are what we would describe as homomorphisms between the term algebras, but we do not intend any such requirement here.

2.4.3 Expressively equivalent semantics

Given two semantics on the same semantic domain we can compare their expressive power by comparing the sets of semantic objects they define. This definition is inherited from that of Yang and Väänänen [YV17].

Definition 2.33. For a fixed semantic type, let $\mathscr{S}$ and $\mathscr{S}'$ be semantics for the languages $\mathcal{L}$ and $\mathcal{L}'$ respectively. Then we say that the semantics are *expressively equivalent*, denoted $\mathscr{S} \approx \mathscr{S}'$ if

$$[\![\mathcal{L}]\!]_{\mathscr{S}} = [\![\mathcal{L}']\!]_{\mathscr{S}'}$$

The apparent clash in notation is explained by the following direct result

Theorem 2.34. *If two semantics are expressively equivalent, then their logics are translationally equivalent.*

Proof. Let $\mathscr{S} \approx \mathscr{S}'$. Since all propositional languages are countable we can enumerate the formulas in the languages and define a translation $\tau : \mathcal{L} \to \mathcal{L}'$ by mapping each formula $\phi \in \mathcal{L}$ to the first formula $\phi' \in \mathcal{L}'$ such that $[\![\phi]\!]_{\mathscr{S}} = [\![\phi']\!]_{\mathscr{S}'}$. Then by Theorem 2.16 we see that

$$\Gamma \vdash_{\mathcal{L}} \phi \text{ if and only if } \tau(\Gamma) \vdash_{\mathcal{L}'} \tau(\phi).$$

We define the translation $\pi : \mathcal{L}' \to \mathcal{L}$ symmetrically, and this proves the logics are translationally equivalent. $\square$

In this sense, two expressively equivalent semantics essentially defines the same logic. For this reason, when many different formulations of a logic are investigated for a fixed semantic domain, the logic as such is sometimes associated with the sets of objects it defines. This is done extensively by Yang and Väänänen in their treatment and descriptions of *propositional team logics* [YV16; YV17]. The next chapter of this thesis is dedicated to giving an account of some of these constructions.

Chapter 3

Team semantics for propositional logics

We give an introduction to *team semantics* for propositional logics as presented in the literature, but expressed within the terminology and notions presented in the previous chapter. We focus mainly on Yang and Väänänens development of semantics for *propositional dependence logic* [YV16], but also on some of the work by Quadrellaro [Qua21] and Lück [Lüc20]. We also define some definitions of important connectives for other team semantics as given in [YV17].

3.1 Valuational team semantics

The motivations for team semantics is to express dependence of the truth valuation of atomic propositions. We use the same definition of a truth-valuation and the set of valuations $\mathscr{V}$ as in Definition 2.7. The idea that a variable p_j may be *dependent* of a set of variables $p_{i_1}, \ldots, p_{i_k}$ is the notion that the possible value of p_j is determined by the specific values of $p_{i_1}, \ldots, p_{i_k}$. Dependency is thus not a property of a single valuation, but of a collection, or a *team* of valuations. The semantic object to evaluate in relation to is thus sets of valuations instead of individual ones.

Definition 3.1. A *valuation team* X is set of valuations $X \subseteq \mathscr{V}$. Let $\mathscr{VT}$ denote the set of all valuation teams, i.e

$$\mathscr{VT} = \mathscr{P}(\mathscr{V}).$$

$\mathscr{VT}$-semantic frameworks and semantics are also called *valuation team* semantic frameworks and semantics.

Using valuation teams as semantic objects we evaluate formulas with respect to them. For standard atomic formulas Yang and Väänänen defines truth by team consensus.

$$X \vDash p_i \text{ iff for all } s \in X : s(i) = 1.$$

They also use negation atoms $\neg p_i$ which are true for a team if the members agree on p_i being false.

$$X \vDash \neg p_i \text{ iff for all } s \in X : s(i) = 0.$$

To express dependence they introduce a *dependence atom* $=(p_{i_1}, \ldots, p_{i_k}, p_j)$ with the semantic truth definition

$$X \vDash =(p_{i_1}, \ldots, p_{i_k}, p_j) \text{ iff for all } s, s' \in X : \text{ if } s(i_1) = s'(i_1), \ldots s(i_k) = s'(i_k)], \text{ then } s(j) = s'(j)$$

Yang and Väänänen uses this together with different connectives to define multiple team logics. They then investigate their expressive power and show that many of these logics are expressively equivalent.

Including a dependence atom produces a multi-sorted language. We want to avoid having multiple sorts of atoms and thus will not present those languages in full. Given the restriction to VT-semantics we can express a one-sorted logic with the same expressive power. We will do so for propositional dependence logic. A one-sorted propositional dependence logic is not new, it is both outlined by Yang in her dissertation [Yan14], and used by Quadrellano [Qua21] and Lück [Lüc20].

3.1.1 Propositional dependence Logic

This is a logic intended to capture the notion of dependence of propositional variables. In their paper [YV16] Yang and Väänänen present different languages for this type of logic (named $\mathbf{PT}_0$, **PD**, $\mathbf{PD}^{\vee}$, **InqL** and **PID**), and prove them all expressively equivalent. For valuational team semantics every formula ϕ defines a collection of teams $[\![\phi]\!]$. Yang and Väänänen prove that the aforementioned logics have the following properties for their definable sets of teams:

- *Empty team property*: $\emptyset \in [\![\phi]\!]$ that is $\emptyset \vDash \phi$ for every ϕ.
- *Downwards closure*: If $X \in [\![\phi]\!]$ and $Y \subseteq X$, then $Y \in [\![\phi]\!]$.

They further identify the expressive power of these logics by proving that they exactly define every non-empty downwards closed collection of teams. We can thus use this to assert that a $\mathscr{VT}$-semantics expresses propositional logic when it has this exact expressive power.

Definition 3.2. A $\mathscr{VT}$-semantics for a language $\mathcal{L}$ expresses propositional dependence logic if the collection of subsets $\mathbb{X} \subseteq \mathscr{VT}$ it defines is exactly the set of non-empty collections of teams that are downwards closed. i.e

$$[\![\mathcal{L}]\!] = \{\, \mathbb{X} \subseteq \mathscr{VT} \mid \mathbb{X} \neq \emptyset, \text{ and if } X \in \mathbb{X}, \text{ and } Y \subseteq X, \text{ then } Y \in \mathbb{X} \,\}.$$

When choosing a semantics for propositional dependence logic we are free to choose any language and semantics as long as we show that it expresses the collection of non-empty downwards closed sets of teams. We will give a semantics for the language with the signature $S(\mathcal{L}) = (\bot, \neg, \wedge, \vee, \otimes)$ as it will suit us in our further development. This exact signature is not given in the literature as far as I know, but the interpretation of each connective can be found in the literature [YV17; YV16; Qua21; Lüc20].

Definition 3.3. Let $S(\mathcal{L}) = (\bot, \neg, \wedge, \vee, \otimes)$. Then let $\mathrm{VT_{PD}}$ denote the $\mathscr{VT}$-semantics of propositional dependence logic given by the following rules for the connectives:

- $X \vDash p_i$ iff for all $v \in X$, $v(i) = 1$.
- $X \vDash \bot$ iff $X = \emptyset$.
- $X \vDash \neg\phi$ iff for all non-empty subteams $Y \subseteq X,\ Y \nvDash \phi$.
- $X \vDash \phi \wedge \psi$ iff $X \vDash \phi$ and $X \vDash \psi$.
- $X \vDash \phi \vee \psi$ iff $X \vDash \phi$ or $X \vDash \psi$.
- $X \vDash \phi \otimes \psi$ iff there exists two subteams Y, Z where $Y \cup Z = X$ such that $Y \vDash \phi$ and $Z \vDash \psi$.

Most noteworthy are the definitions of $\wedge$ and $\otimes$. These connectives both play the role of a disjunction, but on different levels: $\phi \vee \psi$ holds for a team if one of the disjuncts hold for the whole team, whereas $\phi \otimes \psi$ holds for a team if it can be separated into two sub-teams such that each disjunct holds for each of them respectively. At a first glance it may seem that $\otimes$ is a new type of connective that needs team semantics to be definable, but we will learn later that it is actually $\vee$ in the presence of negation $\neg$ that produces the most new interesting properties of $\mathrm{VT_{PD}}$. First we prove that we have described a semantics with the desired expressive power.

Claim 3.4. *VT_{PD} expresses propositional dependence logic.*

Proof. We first argue that the expressive power of $\mathrm{VT_{PD}}$ includes all non-empty downwards closed sets of teams. This is proven by reference. One of the systems Yang and Väänänen describes in [YV16] is a system they call *Propositional dependence logic with intuitionistic disjunction* ($\mathbf{PD}^{\vee}$) [YV16] with the connectives $(\bot, \wedge, \vee, \otimes)$ defined the same way, and instead of a full negation $\neg$, only negative atoms $\neg p_i$ with semantics as described above. By this restriction the formulas of $\mathbf{PD}^{\vee}$ is a subset of the formulas of $\mathrm{VT_{PD}}$, and furthermore by comparing the definitions of negations it is clear that $\mathrm{VT_{PD}}$ and $\mathbf{PD}^{\vee}$ agree on the truth for every team and for all formulas in the language of $\mathbf{PD}^{\vee}$. It is therefore clear that the expressive power of $\mathrm{VP_{PD}}$ exceeds the expressive power of $\mathbf{PD}^{\vee}$. Thus, since $\mathbf{PD}^{\vee}$ expresses propositional dependence logic we have the desired inclusion. What is left to show is that there is nothing more expressed.

We show this by proving that for every formula $\phi \in \mathcal{L}$ we have that the set it defines $[\![\phi]\!]_{\mathrm{VT_{PD}}}$ is a non-empty downwards closed set of teams. The non-emptiness part is seen by observing that the empty team makes every formula true, thus every formula identifies a non-empty set of teams (containing at least the empty team). Downwards closedness is shown by induction over the complexity of the formula:

- *Base cases*: $X \vDash \bot$ iff $X = \emptyset$, so $\bot$ identifies a downwards closed set of connectives. For a proposition p_i assume $X \vDash p_i$ then by definition for all $v \in X, v(i) = 1$. Thus for any subteam $Y \subseteq X$ this also holds. Thus $Y \vDash p_i$.

- *Complex formulas*: Assume every formula of less complexity identifies a downwards closed set of teams. Then we consider cases for the type of main connective. we will not show every connective, only $\otimes$ and $\neg$ since these are the most interesting ones.

  $\otimes$: Assume $X \vDash \phi \otimes \psi$. Then there are subsets Z, W such that $Z \vDash \phi$ and $W \vDash \psi$. Then for any $Y \subseteq X$ define $Z' = Z \cap Y$ and $W' = W \cap Y$ so that $Z' \cup W' = Y$. Then by being subteams and induction hypothesis $Z' \vDash \phi$ and $W' \vDash \psi$ and thus $Y \vDash \phi \otimes \psi$.

  $\neg$: Assume $X \vDash \neg\phi$, Then for every subteam $Y \subseteq X$ we have $Y \nvDash \phi$. Let $Z \subseteq X$ be any subteam of X. To prove that $Z \vDash \neg\phi$ we need to prove that for every subteam $W \subseteq Z$ we have $W \nvDash \phi$, but by mere transitivity $W \subseteq X$. Thus this is already established.[1] □

We have chosen $\mathrm{VT_{PD}}$ since it is close to $\mathbf{PD}^{\vee}$ but is one-sorted by having the notion of negation extended to every formula. This extended negation will be important to us, and we can see some other results regarding it. Recall the rule for negation. Since the logic is downwards closed we see this is also equivalent with the set X not sharing any subset with a team for which ϕ is true:

$$X \vDash \neg\phi \text{ iff for all } Y \vDash \phi: \ Y \cap X = \emptyset. \tag{3.1}$$

From this we directly conclude that

$$X \vDash \neg\phi \wedge \phi \text{ if and only if } X = \emptyset$$

In this sense $\mathrm{VT_{PD}}$ defines an (internally) consistent logic.

Claim 3.5. *VT_{PD} is truth-compositional.*

Proof. This can be fairly easily assessed from the definitions, but we can also give explicit dual functions as identified by Lück [Lüc20] :

$\bot: \ [\![\bot]\!] = \{\emptyset\}$,

$\neg: \ [\![\neg\phi]\!] = \{\, X \mid \text{ for all } Y \in [\![\phi]\!] : X \cap Y = \emptyset \,\}$,

$\wedge: \ [\![\phi \wedge \psi]\!] = [\![\phi]\!] \cap [\![\psi]\!]$,

$\vee: \ [\![\phi \vee \psi]\!] = [\![\phi]\!] \cup [\![\psi]\!]$, and

$\otimes: \ [\![\phi \otimes \psi]\!] = \{\, X \cup Y \mid X \in [\![\phi]\!] \text{ and } Y \in [\![\psi]\!] \,\}$.

These dual functions however are not unique, a point we will notice again in Chapter 6. □

3.1.2 Connections to other logics

Even though propositional dependence logic and team semantics was formulated to capture the notion of dependence, it is somewhat surprisingly similar to other logics and semantic constructions. We will present connections to *inquisitive logic* and *linear logic.*

Inquisitive logic

Inquisitive logic is a logic that wants to reflect how natural languages are used to exchange information. In natural languages sentences can state facts, but they can also be questions - inquisitions asking for information. The idea of inquisitive logic is to describe a logic where formulas not only capture statements of facts, but may also be interpreted as *questions* or *inquisitions*. Inquisitive logic was first described by Groenendijk [Gro07]. The following description is based on the accounts

[1] Note that the proof for $\neg$ never used the induction hypothesis since establishing a negation being true for a formula is already connected to truths for subteams. This also shows in the *flatness* of negated formulas described below.

given by Yang [Yan14] and Quadrellaro [Qua21]. A basic notion for a general semantics for inquisitive semantics is an notion of *information states* which is a set of possible values for atomic formulas. From a formal perspective an *information state* is essentially a valuation team. In the same way as in valuational team semantics, truth semantics for inquisitive logic are given with this as the semantic object for the logic. In inquisitive semantics, an information state is viewed as a potential set of worlds and a proposition ϕ embodies both inquisitive and informative content. If $X \vDash \phi$ it is said that 'X settles the issue raised by ϕ'. Although inquisitive semantics has had completely different reading, it uses essentially valuational team semantics, just with different terminology. We can thus present propositional Inquisitive logic **InqL** in valuational team semantics by stating definitions for its connectives.

Definition 3.6. **InqL** is the semantics of the framework for the language with signature $(\bot, \wedge, \vee, \rightarrow)$, where $\bot, \wedge, \vee$ are defined as for $\mathrm{VT}_{\mathrm{PD}}$ and $\rightarrow$ is defined as follows

$$X \vDash \phi \rightarrow \psi \text{ iff for any subteam } Y \subseteq X: \text{ if } Y \vDash \phi, \text{ then } Y \vDash \psi.$$

The implication $\rightarrow$ then satisfies the deduction theorem

$$\mathbf{InqL}: \phi \vDash \psi \text{ if and only if } \mathbf{InqL}: \ \vDash \phi \rightarrow \psi$$

It was shown by Yang and Väänänen [YV16] that **InqL** expresses propositional dependence logic and thus it is expressively equivalent to $\mathrm{VT}_{\mathrm{PD}}$. Note also that if we define negation $\neg\phi$ as $\phi \rightarrow \bot$ this definition coincides with the definition of negation in $\mathrm{VT}_{\mathrm{PD}}$. We can also see by the downwards closure and the empty-set property, that the definition of $\rightarrow$ is equivalent with the following definition

$$X \vDash \phi \rightarrow \psi \text{ iff for any team } Y: \text{ if } Y \vDash \phi, \text{ then } Y \cap X \vDash \psi.$$

Linear logic

Linear logic is a logic introduced by Girard in [Gir87]. It is a *resource sensitive substructural logic* with strong motivations from proof theory and computation theory. The logic relates to classical propositonal logic, but with two types of disjunction and conjunction known as the *multiplicative* and *additive* variants. These reflect properties of their definition in the defining sequent calculus of linear logic [GLR95]. Being produced primarily with proof theory in mind, linear logic is not associated with a single semantic interpretation but different semantics are motivated by different connections, such as semantics by *proof nets*, or *monoidal categories* [Gir11]. Some of these semantics are semantics for linear formulas, and some are semantics of the full proof sructures. One early semantics for formulas of linear logic is *phase semantics*. We will not give a full account, but refer to Girard's exposition in [GLR95] for further details.

Definition 3.7. A *phase space* is a pair $(M, \bot)$, where M is a commutative monoid[2] and $\bot$ is a subset of M. We then give interpretation of all connectives of linear logic $(\bot, \top, 0, 1, \cdot^{\bot}, \multimap, \otimes, \&, \invamp, \oplus)$ as functions $\mathscr{P}(M)^n \rightarrow \mathscr{P}(M)$, in particular *linear implication* $\multimap$ is interpreted as

$$X \multimap Y = \{\, m \in M \mid \text{ for all } n \in X : m \cdot n \in Y \,\}.$$

A formula ϕ is then considered *true* in the phase space if ϕ is interpreted as a set that includes the neutral element 1 of the monoid.

Girard then proves that a formula is provable in linear logic if and only if it is true in every phase space [GLR95]. This then constitutes phase semantics as a semantics for formulas in linear logic.

Phase semantics and linear logic have a connection to dependence logic noted by Abramsky and Väänänen [AV09]. If we consider the set of valuation teams as a commutative monoid with respect to the set operation $\cup$ with unit element $\mathscr{V}$, then $\otimes$ in dependence logic exactly corresponds to multiplicative conjunction $\otimes$ in linear logic, and in particular we can define a corresponding connective $\multimap$ for valuational team semantics defined by Yang and Väänänen [YV17] by

$$X \vDash \phi \multimap \psi \text{ iff for any team } Y: \text{ if } Y \vDash \phi, \text{ then } Y \cup X \vDash \psi$$

[2] a *commutative monoid* is a set M together with a binary operation $\cdot$, that is commutative ($n \cdot m = m \cdot n$) and has a neutral element $1 \in M$ ($m \cdot 1 = m$ for all $m \in M$.)

This connection is however somewhat twisted, since $\otimes$, playing the role of a disjunction in dependence logic, is paired with its conjunctive name-sake in linear logic. We will discuss in Section 6.3 how a new team semantics and logic reveal a more straight-forward connection between linear logic through phase semantics, and team logics. As a precursor, observe the apparent duality between the definition of linear implication $\multimap$ in valuational team semantics above and the second definition of implication $\rightarrow$ for inquisitive logic.

3.1.3 Other connectives and team logics in the literature

For a further discussion of team logics we also list some other connectives as defined by Yang and Väänänen in [YV17]. Some of these are investigated further by Lück [Lüc20]. These serve mostly as further examples for how teams can be utilised to give semantics for more elaborate connectives.

Definition 3.8. Given valuation teams X, Y, Z and formulas ϕ, ψ in appropriate language, we give the following standard definitions for the connectives NE^0, ∇^1 and $\circledast^2$ in valuational team semantics:

NE : $X \vDash \mathbf{NE}$ iff $X \neq \emptyset$. We call **NE** the *non-emptiness constant*. This constant is interesting since if added or definable, the team semantics cannot have the empty team property.

∇ : $X \vDash \nabla\phi$ iff $X = \emptyset$ or there exists a non-empty subteam $Y \subseteq X$ such that $Y \vDash \phi$. ∇ is called the *might modality* or *non-emptyness operator* and considered in modal team logics [HS15].

$\circledast$: $X \vDash \phi \circledast \psi$ iff $X = \emptyset$ or there are *non-empty* subteams Y, Z where $Y \cup Z = X$ such that $Y \vDash \phi$ and $Z \vDash \psi$. $\circledast$ is called the *non-empty disjunction* and the relation between it, $\otimes$ and $\vee$ is extensively discussed by Lück [Lüc20].

Apart from just listing these connectives we also mention the logic called *strong propositional team logic* $\mathbf{PT}^+$ which is the logic for the connectives $(\bot, \mathrm{NE}, \wedge, \vee, \otimes)$ and atomic negation $\neg p_i$ as described above. Yang and Väänänen proves in [YV17] that this logic, in valuational team semantics, defines every possible collection of teams, and is thus an *expressively complete* $\mathscr{VT}$-semantics.[3] For more details and more exhaustive list of team logics , I refer to the aforementioned paper by Yang and Väänänen [YV17].

3.2 Relating $\mathscr{V}$- and $\mathscr{VT}$-semantics

We have now made a lift from valuational semantics to valuational team semantics. We are interested in what properties translates into the new setting, and what the relation between the two is. To study this relation is one of the focuses in Lücks dissertation on team logics [Lüc20]. The following results can in general be found there, but phrased in a different way. We start by identifying the *flatness* property also identified by Yang and Väänänen in [YV16; YV17].

3.2.1 Flatness property

Consider the semantics $\mathrm{VT}_{\mathrm{PD}}$ and $\mathrm{V}_{\mathrm{CPL}}$ as defined above and in Definition 2.8 respectively. By the definitions it is clear that a team X finds a propositional variable p_i true in $\mathrm{VT}_{\mathrm{PD}}$ if and only if every member $s \in X$ finds p_i true in V_{CPL}. We can express this fully in the valuational team semantics by mapping each valuation to its singleton team

$$X \vDash p_i \text{ iff for all } s \in X : \{s\} \vDash p_i.$$

This relation between valuation in a team and consensus of its singletons is called the flatness property by Yang and Väänänen [YV16]. Lück investigates this further in [Lüc20], and calls a connective *flatness preserving* if it connects flat formulas into a flat formula. We inherit these definitions:

[3] Technically this is not a $\mathscr{VT}$-semantics as we have defined it, since it is multi-sorted with negation atoms. It is fairly easy to see how the definitions of semantics can be modulated to allow multi-sorting, or that the corresponding semantics with negation as a connective as described above, also is expressively complete.

Definition 3.9. Given a $\mathscr{VT}$-semantics for a language $\mathcal{L}$, a formula ϕ is called *flat* if for all teams $X \in \mathscr{VT}$

$$X \vDash \phi \text{ iff for all } s \in X : \{s\} \vDash \phi.$$

A $\mathscr{VT}$-semantics is *flat* if every formula of the language is flat for it. A connective $*^n \in \mathcal{L}$ is *flatness preserving* in a semantics if $*(\phi_1, \dots, \phi_n)$ is flat whenever $\phi_1, \dots, \phi_n$ are.

We can then state the following theorem

Theorem 3.10. *For the semantics VT_{PD} the connectives $\bot, \neg, \wedge, \otimes$ are flatness preserving, but $\vee$ is not. Consequently, every $\vee$-free formula is flat.*

Proof. The flatness preserving is proven for each connective separately. Note however, since every definable set is downwards closed we know for all formulas ϕ:

$$\text{If } X \vDash \phi \text{ then } \{s\} \vDash \phi \text{ for all } s \in X.$$

What is needed to show for a formula to be flat is only the opposite implication.

$\bot$: $X \vDash \bot$ if an only if $X = \emptyset$. Hence $\bot$ is flat.

$\wedge$: Assume for all $s \in X : \{s\} \vDash \phi \wedge \psi$. Then by definition $\{s\} \vDash \phi$ and $\{s\} \vDash \psi$ for all $s \in X$. If ϕ and ψ are flat then $X \vDash \phi$ and $X \vDash \psi$, and thus $X \vDash \phi \wedge \psi$. Thus, if ϕ and ψ are flat, so is $\phi \wedge \psi$.

$\otimes$: Assume for all $s \in X : \{s\} \vDash \phi \otimes \psi$. Then by definition $\{s\} \vDash \phi$ or $\{s\} \vDash \psi$ for all $s \in X$. Now let Y be the set of all valuations making ϕ true, and Z the set of all valuations making ψ true:

$$Y = \{\, s \in X \mid \{s\} \vDash \phi \,\} \qquad Z = \{\, s \in X \mid \{s\} \vDash \psi \,\}.$$

It is clear that $Y \cup Z = X$. If ϕ and ψ are flat, then $Y \vDash \phi$ and $Z \vDash \psi$,and thus $X \vDash \phi \otimes \psi$. Hence if ϕ and ψ are flat, so is $\phi \otimes \psi$.

$\neg$: Assume for all $s \in X : \{s\} \vDash \neg\phi$. Then $\{s\} \nvDash \phi$ for all $s \in X$. Assume for contradiction that $X \nvDash \neg\phi$. Then there is some non-empty subset $Y \subseteq X$ such that $Y \vDash \phi$. By downwards closure then for all $s \in Y : \{s\} \vDash \phi$. Since Y is non-empty this would mean that for some $s \in X : \{s\} \vDash \phi$ and $\{s\} \nvDash \phi$ which is a contradiction. Hence $X \vDash \phi$ and thus $\neg\phi$ is flat *regardless of the flatness of the formula ϕ.*

To prove that $\vee$ is not flat, we give a simple counter example: It is clear from the above that every atomic formula p_i and its negation $\neg p_i$ are flat. It is also clear that for every valuation s we have that $\{s\} \vDash p_i \vee \neg p_i$. But it is easy to find teams X such that $X \nvDash p_i \vee \neg p_i$,hence $\vee$ is not flatness preserving.

The fact that all atomic formulas are flat proves the consequential statement in the theorem. □

Note 3.11. In the previous proof we saw specifically that *every* negated formula is flat. We can thus also define negation by the following:

$$X \vDash \neg\phi \text{ iff for all valuations } s \in X : \{s\} \nvDash \phi.$$

For flat formulas the truth of the formula in a team boils down to simple team consensus. As a consequence the collection of teams a flat formula defines is a powerset of a set of valuations. This identifies all flat formulas.

Theorem 3.12. *For any $\mathscr{VT}$-semantics : ϕ is flat if and only if $[\![\phi]\!]$ is a powerset.*

Proof. We prove the statement one direction at a time.

Assume ϕ is flat. Let S be the set of all valuations that makes ϕ true as singletons.

$$S = \{\, s \in \mathscr{V} \mid \{s\} \vDash \phi \,\}.$$

We claim that $\mathscr{P}(S) = [\![\phi]\!]$. If $X \in \mathscr{P}(S)$ then for all $s \in X : \{s\} \vDash \phi$ and thus by flatness $X \vDash \phi$ so that $X \in [\![\phi]\!]$. Thus $\mathscr{P}(S) \subseteq [\![\phi]\!]$. If $X \notin \mathscr{P}(S)$ then $X \backslash S$ is non-empty an thus there is some

$t \in X$ such that $\{t\} \nvDash \phi$. Then by flatness $X \nvDash \phi$ and thus $X \notin [\![\phi]\!]$. Hence $\mathscr{P}(S) \supseteq [\![\phi]\!]$ and thus $\mathscr{P}(S) = [\![\phi]\!]$.

Assume $[\![\phi]\!] = \mathscr{P}(S)$ for some set of valuations S we need to prove that ϕ is flat. Since every definable set is downwards closed it is enough to show that if $\{s\} \vDash \phi$ for all $s \in X$ then $X \vDash \phi$.

If $\{s\} \vDash \phi$, then $\{s\} \in \mathscr{P}(S)$ and thus $s \in S$. So if $\{s\} \vDash \phi$ for all $s \in X$ then $X \subseteq S$ and thus $X \in \mathscr{P}(S) = [\![\phi]\!]$. Thus $X \vDash \phi$ and we have proven that ϕ is flat. □

From this we directly conclude that if a $\mathscr{VT}$-semantics is flat, then there is an equally "nice" $\mathscr{V}$-semantics that is logically equivalent.

Theorem 3.13. *If a $\mathscr{VT}$-semantics VT is flat, then there is a $\mathscr{V}$-semantics V with the following inherited properties:*

$$\textit{If} \quad VT : X \vDash p_i \textit{ iff for all } s \in X : s(i) = 1, \quad \textit{then} \quad V : s \vDash p_i \textit{ iff } s(i) = 1$$

and if VT is truth-compositional, so is V.

Proof. If every formula is flat for VT, then for every ϕ there is a set of valuations S such that $[\![\phi]\!]_{\mathrm{VT}} = \mathscr{P}(S)$. We can then define a $\mathscr{V}$-semantics V by defining the set of valuations each formula satisfy to be the underlying set.

$$[\![\phi]\!]_{\mathrm{V}} = S \text{ where } S \text{ is such that } [\![\phi]\!]_{\mathrm{VT}} = \mathscr{P}(S).$$

We claim that V is logically equivalent to VT. This follows directly from the observation in Section 2.2.1 that for any truth-semantics $\mathscr{S}$

$$\mathscr{S} : \Gamma \vDash \phi \text{ if and only if } \bigcap_{\gamma \in \Gamma} [\![\gamma]\!]_{\mathscr{S}} \subseteq [\![\phi]\!]_{\mathscr{S}}$$

and the general set theory observations that

$$\bigcap_{i \in I} \mathscr{P}(S_i) = \mathscr{P}(\bigcap_{i \in I} S_i)$$

and that

$$\mathscr{P}(S) \subseteq \mathscr{P}(S') \text{ if and only if } S \subseteq S'.$$

We then get the following chain of equivalent statements:

$$\mathrm{VT} : \Gamma \vDash \phi \iff \bigcap_{\gamma \in \Gamma} [\![\gamma]\!]_{\mathrm{VT}} \subseteq [\![\phi]\!]_{\mathrm{VT}} \iff \bigcap_{\gamma \in \Gamma} \mathscr{P}([\![\gamma]\!]_{\mathrm{V}}) \subseteq \mathscr{P}([\![\phi]\!]_{\mathrm{V}}) \iff$$

$$\mathscr{P}(\bigcap_{\gamma \in \Gamma} [\![\gamma]\!]_{\mathrm{V}}) \subseteq \mathscr{P}([\![\phi]\!]_{\mathrm{V}}) \iff \bigcap_{\gamma \in \Gamma} [\![\gamma]\!]_{\mathrm{V}} \subseteq [\![\phi]\!]_{\mathrm{V}} \iff \mathrm{V} : \Gamma \vDash \phi$$

and thus V and VT are logically equivalent. Now we need to establish the "nice" properties of V.

Assume

$$\mathrm{VT} : X \vDash p_i \text{ iff for all } s \in X : s(i) = 1.$$

Let S be the set of all valuations evaluating i to 1,

$$S = \{ s \in \mathscr{V} \mid s(i) = 1 \}.$$

It is then clear that $[\![p_i]\!]_{\mathrm{VT}} = \mathscr{P}(S)$ and by construction $[\![p_i]\!]_{\mathrm{V}} = S$ and thus

$$\mathrm{V} : s \vDash p_i \text{ iff } s(i) = 1.$$

Assume VT is truth-compositional. Then for every connective $*^n$ in the language there is a function $f_* : \mathscr{VT}^n \to \mathscr{VT}$ such that

$$[\![*(\psi_1, \ldots, \psi_n)]\!]_{\mathrm{VT}} = f_*([\![\psi_1]\!]_{\mathrm{VT}}, \ldots, [\![\psi_n]\!]_{\mathrm{VT}}) \text{ for all } \psi_1 \ldots \psi_n \in \mathcal{L}.$$

Now, since VT is flat and $\mathscr{P}(\cdot)$ is an injective class function on sets we can define the function $\bar{f}_* : [\![\mathcal{L}]\!]^n_{\mathrm{V}} \to [\![\mathcal{L}]\!]_{\mathrm{V}}$ as follows

$$\bar{f}_*(S_1, \dots S_n) = S \text{ such that } \mathscr{P}(S) = f_*(\mathscr{P}(S_1), \dots, \mathscr{P}(S_n)) \text{ for all } S_1 \dots S_n \in [\![\mathcal{L}]\!]_{\mathrm{V}}.$$

From this follows directly that

$$[\![*(\psi_1, \dots, \psi_n)]\!]_{\mathrm{V}} = \bar{f}_*([\![\psi_1]\!]_{\mathrm{V}}, \dots, [\![\psi_n]\!]_{\mathrm{V}}) \text{ for all } \psi_1 \dots \psi_n \in \mathcal{L}.$$

So by considering arbitrary extensions of $\bar{f}_*$ to the full domain $\mathscr{V}$ for every connective $*$ in the languages proves that V is truth-compositional. □

This last theorem shows that if we only consider flat valuation team semantics, we have not gained the ability to express any additional logics which could not be expressed by valuational semantics alone. In this sence, it is the non-flatness preserving connectives such as $\vee$ in $\mathrm{VT}_{\mathtt{CPL}}$ that makes team semantics interesting.

3.3 Non-substitutionality and algebraisation

Propositional dependence logic has clear semantic motivations, and it is an interesting logic its own right, but also for its connection to other logics. It does however have the problem that none of the expressively equivalent logics in the literature is substitutional. This is also the case for $\mathrm{VT}_{\mathtt{PD}}$.

Theorem 3.14. *The logic of $VT_{\mathtt{PD}}$ is not substitutional.*

It is sufficient to exhibit a counter example. For future reference will we however prove a bit more than needed. We have observed that every negated formula is flat. We can then also prove the following claim:

Claim 3.15. *In $VT_{\mathtt{PD}}$: $\neg\neg\phi \vDash \phi$ if and only if ϕ is flat, in particular $\neg\neg p_i \vDash p_i$ for every $i \in \mathbb{N}$.*

Proof. For any valuation team X, assume $X \vDash \neg\neg\phi$. Then, since every negated formula is flat

$$\{s\} \vDash \neg\neg\phi \text{ for all } s \in X.$$

But by definiton of $\neg$ and the fact that the only non empty subteam of a singleton team is the team itself

$$\{s\} \vDash \phi \text{ for all } s \in X.$$

This statements is equivalent with $X \vDash \phi$ if and only if ϕ is flat. Thus, we have proven the claim. □

All we need to prove non-substitutionality of $\mathrm{VT}_{\mathtt{PD}}$ is a formula that is not flat. A simple example is $p_1 \vee \neg p_1$. For every singleton team $\{s\}$ it is clear that $\{s\} \vDash p_1 \vee \neg p_1$, but it is also evident that $p_1 \vee \neg p_1$ is not true in every valuation team. Hence $p_1 \vee \neg p_1$ is not flat.

Proof of Theorem 3.14 . $\mathrm{VT}_{\mathtt{PD}} : \neg\neg p_1 \vDash p_1$ but $\mathrm{VT}_{\mathtt{PD}} : \neg\neg(p_1 \vee \neg p_1) \nvDash p_1 \vee \neg p_1$. □

The non-substitutionality of the logic of $\mathrm{VT}_{\mathtt{PD}}$ is not fringe for team logics. It turns out that none of the logics with non-flat valuational team semantics presented by Yang and Väänänen, Lück, or Quadrellaro are substitutional [YV16; YV17; Lüc20; Qua21], and it begs the question if non-substitutionality is an inherent trait of team logics. Our motivation to construct a *monadic semantics* was to invensigate this issue, and in Section 5.3 we manage to give some answers.

3.3.1 Algebraisation as done by Quadrellaro

The lack of substitutionality for team logics has as one of its consequence that they inherently are not algebraisable in the sense of abstract algebraic logic as presented by Font [Fon16]. Quadrellaro makes an effort to adapt Font's machinery to a family of non-substitutional team logics [Qua21]. We can in general terms describe his adaption in our terminology. Quadrellaro defines an algebraic framework $F = (\mathcal{A}, \top)$ and a subset $C \subseteq \mathcal{A}$ called the *core*. He then produces the semantics $\mathscr{S} = (F, \mathcal{H}|_C)$ where $\mathcal{H}|_C$ is the set of $\mathcal{A} - valuations$ with image within the core C, that is

$$\mathcal{H}|_C = \{ h : \mathbb{N} \to \mathcal{A} \mid h(\mathbb{N}) \subseteq C \}.$$

By then defining certain properties of such pairs $(\mathcal{A}, C)$ of algebras and core-sets, Quadrellaro identifies a class of algebraic-semantics $\mathfrak{S}$ that together form a consensus semantics for the desired team logic.

$$\mathfrak{S} : \Gamma \vDash \phi \text{ if and only if for all } \mathscr{S} \in \mathfrak{S} : \ \mathscr{S} : \Gamma \vDash \phi.$$

It is the restriction of the $\mathcal{A} - valuations$ $\mathcal{H}|_C$ that makes this construction differ from the standard algebraisation construction, and the fact that such restriction is necessary is evident from Theorem 2.28 stating that the logic of an algebraic framework is always substitutional.

Quadrellaro manages to recover many of the important results and connections that can be expressed through abstract algebraic logic [Qua21; Fon16]. His approach does however also have some undesirable properties. From the fact that the set of $\mathcal{A} - valuations$ are restricted to a specific core-set C, the induced homomorphisms map every formula of the language to a designated *corner* of the algebra of the framework. Therefore, there are parts of the algebra that never play any role in the semantic interpretation of the logic. This has as a consequence that the algebras in the class of semantics $\mathfrak{S}$ are only partially defined and can have parts with arbitrary properties.

In the upcoming chapters of this thesis we look to define a more general team semantics and use it to try to diagnose the issue of non-substitutionality in team semantics. Then in Chapter 6 we describe a weaker substitutional team logic, for which propositional dependence logic and other team logics can be recovered. Our hope is that this new logic is more naturally suitable for algebraisation.

3.4 Comment on proof systems and more general team semantics

The existing work on propositional team semantics and team logics goes well beyond the results accounted for in this chapter. We have focused our presentation on standard valuational team semantics and the semantic formulation of propositional dependence logic. Even if it will not be noticeably addressed in this thesis, this account would be lacking without at least a mention of the extensive work on proof systems for propositional logics, and generalisations of the semantics they motivate. In [YV16] and [YV17] Yang and Väänänen present multiple natural deduction systems for different propositional team logics, while Quadrellaro [Qua21], using languages with implication, account for Hilbert systems. However, since the propositional team logics are not substitutional, these systems include rules and axioms that are not valid under arbitrary substitution, but instead have to include qualifiers on the included formulas for when the statement holds or the deduction rules can be used. This is a shortcoming of all these systems.

When a proof system is described it is common to consider weaker logics generated by omitting parts of the proof system and investigating the resulting logics and their possible semantics. This have directly given rise to the formulation of *intuitonistic team semantics* that can be given *intuitionistic Kripke team semantics* as accounted for in [Qua21]. These are semantics on Kripke models where the semantic object are *subsets* of the set of worlds, also called *teams* as their valuational counterparts. The study of propositional team logics and semantics generated in this fashion is not exhausted and is currently pursued in part by Yang [Yan22]

In this thesis we also seek to construct new semantics for propositional dependence logic, but with rather different motivations. Our goal is to investigate the problem of substitutionality, and the semantics we present will therefore not spring from any certain proof system or set of axioms. We will instead start by re-evaluating the role of propositional variables in the semantics.

Chapter 4

Monadic semantics

We present and motivate a new type of semantics that will be useful to discuss the issue of substitutionality. The purpose of this discussion is to examine team logics, but the results will be more general and apply to any logic with truth-compositional semantics.

In valuational semantics we consider valuations v as functional objects that evaluate the propositional variables p_i. We can then say that *the valuation deems the propositional variable true*, and write it as $v(i) = 1$. We can however reverse the roles. We can instead view the propositional variable p_i as a unary predicate describing some property according to some interpretation M. We then determine if a sample s has such a property, and say that *s has the property p_i in the interpretation M* and write $s \in p_i^M$. This approach is equally natural. We use this viewpoint and construct a semantics by letting each formula ϕ give rise to a unary predicate on a universe of things. We can then treat the entailment $\phi \vDash \psi$ as a statement of inclusion for the corresponding predicates. This idea is also apparent in what is known as *description semantics* investigated in for example [Dik22], but at least to my mind and limited knowledge, these semantics are developed in a very different direction than in the one we are heading. When choosing a name for the upcoming construction I have settled on the term *Monadic semantics* since unary predicates also can be called *monadics*. This may not be a perfect name, since one may think it uses the algebraic idea of a *monad*. It is however the name used in this thesis, and if anyone is interested enough to develop this further with new intuitions, a discussion of renaming is only welcomed.

When trying to describe my motivations and intentions for describing monadic semantics, I remembered a passage by Bertrand Russell in his book *Introduction to Mathematical Philosophy* [Rus19]. When discussing the notions of *similarity* and *likeness* for relations he writes:

> "We are led by such considerations to a problem which has, in mathematical philosophy, an importance by no means adequately recognised hitherto. Our problem may be stated as follows:—
>
> Given some statement in a language of which we know the grammar and the syntax, but not the vocabulary, what are the possible meanings of such a statement, and what are the meanings of the unknown words that would make it true?
>
> The reason that this question is important is that it represents, much more nearly than might be supposed, the state of our knowledge of nature. We know that certain scientific propositions—which, in the most advanced sciences, are expressed in mathematical symbols—are more or less true of the world, but we are very much at sea as to the interpretation to be put upon the terms which occur in these propositions. We know much more (to use, for a moment, an old-fashioned pair of terms) about the *form* of nature than about the *matter*. Accordingly, what we really know when we enunciate a law of nature is only that there is probably *some* interpretation of our terms which will make the law approximately true. Thus great importance attaches to the question: What are the possible meanings of a law expressed in terms of which we do not know the substantive meaning, but only the grammar and syntax? And this question is the one suggested above."

The sentiment and question here expressed by Russell lies very much in the heart of my idea, formulation and treatment of monadic semantics.

We consider a class of *models* that each represent possible *interpretations* of what type of things elements in a *universe* may be, represented as memberhood of a propositional variable. We then fix a language of connectives, and a *frame* describing rules that govern them, to create composite properties. In a frame we can then make the models interpret complex formulas in a way govern by these rules. A monadic semantics is then given by determining how a set of models on a frame agree on their interpretations of the things in the universe. In this chapter we encode this idea in formal definitions using the general terminology for semantics we have established. We can then show that Monadic semantics can represent any truth-compositional semantics and identify a particular class of monadic semantics that expresses team logics. In the next chapter we then lift the discussion by investigating the set of all possible interpretations of a formula in classes of models. We manage to relate structural properties in interpretations by classes of models on a frame with structural properties in the logics they define, in particular in regards to substitutionality. I see this investigation as being very much in line with Russell's question.

But first, the definitions.

4.1 Monadic frames and semantics

First we categorise the set of semantic objects.

Definition 4.1. A *monadic model* M is a valuation function $M : \mathbb{N} \to \mathscr{P}(U)$ for some set U called its *universe*. Let $\mathrm{Mod}(U)$ denote the collection of all monadic models with universe U. We call an element $s \in U$ a *sample*, a *point* or a sample-point from the universe. A pointed monadic model, is a pair M, s of a monadic model on a universe U together with a sample-point $s \in U$. Let $\mathscr{M}p(U)$ denote the set of all pointed monadic models of the universe U.

Definition 4.2. A *monadic frame* F of a language $\mathcal{L}$ is given by fixing a universe U_F and functional interpretations $*^n_F : U^n_F \to U_F$ for every connective $*^n \in \mathrm{Sing}(\mathcal{L})$.

Definition 4.3. For a monadic frame F and a monadic model $M \in \mathrm{Mod}(U_F)$ we define for every formula $\phi \in \mathcal{L}$ *its interpretation in* M *induced by* F, denoted ϕ^M_F. This is recursively defined as follows

- *Base case*: $p^M_{i_F} = M(i)$
- *Complex formula*: $(*^n(\phi_1, \dots, \phi_n))^M_F = *^n_F(\phi^M_{1F}, \dots, \phi^M_{nF})$

We define ϕ to be *true at the sample* $s \in U_F$ *in a model* M *for* F, denoted $F : M, s \vDash \phi$ by

$$F : M, s \vDash \phi \iff s \in \phi^M_F.$$

A monadic frame F thus defines an $\mathscr{M}p(U_F)$-semantic framework.

This wording is intensionally similar to the terminology of Kripke semantics. As will be illustrated below, if formulating a semantics for a modal language by imposing a relational structure on the universe, we create a Kripke frame as a special case of monadic frames. We have essentially generalised to a broader setting in order to consider other types of interactions between the samples/points of the domain than those that are naturally expressed by an accessibility relation. We could also say that we have rephrased from *possible worlds semantics* describing truths in possible points, to *possible interpretations semantics* describing possible interpretation for what is true in points.

Since we have restricted the idea of monadic frames to a type of framework where the connectives are functionally defined, every monadic frame gives rise to a truth-compositional semantic framework.

Theorem 4.4. *Every monadic frame defines a truth-compositional semantic framework.*

Proof. Let F be a monadic frame on the universe U for the language $\mathcal{L}$. For every formula ϕ, and every model $M \in \mathrm{Mod}(U)$ let $[\![\phi]\!]^M_F$ denote the restriction of the set ϕ defines to the set of pointed models that have M as their model

$$[\![\phi]\!]^M_F = \{ (N, s) \in [\![\phi]\!]_F \mid N = M \}.$$

If we have a collection of pointed models O, we can get a collection of points $P(O)$ by forgetting the models.

$$P(O) = \{ s \in U \mid \text{ for some } M \in \mathrm{Mod}(U) : (M, s) \in O \}$$

We can then directly observe, that $P([\![\phi]\!]^M_F) = \phi^M_F$. Using this can we for any connective $*^n$ give a dual function construction

$$[\![*^n(\psi_1, \dots, \psi_n)]\!]_F = \left\{ (M, s) \;\middle|\; s \in *^n_F(P([\![\psi_1]\!]^M_F), \dots, P([\![\psi_n]\!]^M_F)) \right\}$$

Even if it is a somewhat clunky definition of a dual function, it is sufficient to prove that F is truth-compositional. $\square$

With a type of semantic frameworks defined it is time to clarify what type of semantics we have in mind.

Definition 4.5. A *monadic semantics* is a pair $(F, \mathcal{M})$, where F is a monadic frame, and $\mathcal{M} \subseteq \mathrm{Mod}(U_F)$ is a collection of monadic models on its universe. The monadic semantics is then defined as the $\mathscr{M}p(U_F)$-semantics $(\mathscr{F}, O)$ where the framework $\mathscr{F}$ is determined by the frame F and $O = \{ (M, s) \mid M \in \mathcal{M} \text{ and } s \in S_F \}$. When $\mathcal{M} = \mathrm{Mod}(U_F)$ it may be omitted and we call the resulting monadic semantics *the semantics of the frame F*.

Note 4.6. Observe that not every $\mathscr{M}p(U_F)$-semantic framework correspond to a monadic frame, and not every semantics on a monadic frame is a monadic semantics. We have defined these types of semantics to capture the idea of formulas as unary predicates interpreted in a universe by a class of models.

Note 4.7. The entailment for a monadic semantics $(F, \mathcal{M})$ for a universe U is defined as before. We can however give some transparency to the setup by writing it out explicitly. The definition reeds

$$(F, \mathcal{M}) : \Gamma \vDash \phi \quad \text{if and only if for all } M \in \mathcal{M}, s \in U \text{ if } F : M, s \vDash \Gamma, \text{ then } F : M, s \vDash \phi.$$

We can however simplify the definition. Since $F : M, s \vDash \phi$ is defined as ϕ_F^M we directly see

$$(F, \mathcal{M}) : \Gamma \vDash \phi \quad \text{if and only if for all } M \in \mathcal{M} \text{ we have that } \bigcap \Gamma_F^M \subseteq \phi_F^M.$$

This gives a way to evaluate the entailment of a monadic semantics on the model level, without any need to consider specific sample points. We will use these two definitions interchangeably.

To illustrate the terminology, we will give some examples of monadic semantics. We will start of with multiple different monadic semantics for classical propositional logic (CPL).

Valuation-to-sample semantics VS_{CPL}

We want to give a monadic semantics for classical propositional logic with the signature $(\bot, \neg, \wedge, \vee)$. We can use the well known $\mathscr{V}$-semantics V_{CPL} given in Definition 2.8. One way of doing so is to consider the universe to be the set of valuations $\mathscr{V}$ and construct a frame F so that we get a distinct model M for which formulas are interpreted as the sets of valuations they define in V_{CPL}, that is

$$\phi_F^M = [\![\phi]\!]_{V_{\mathsf{CPL}}}.$$

Since V_{CPL} is truth-compositional this is easily done, by giving the functional interpretations of the connectives in the monadic frame by the dual functions for the connectives in V_{CPL}.

Definition 4.8. We define the semantics VS_{CPL}, called the *valuations-to-samples semantics for* CPL, as the monadic semantics $(F, \{M\})$ on the universe $U_F = \mathscr{V}$ where M is the model defined by setting

$$M(i) = \{ s \in \mathscr{V} \mid s(i) = 1 \} \quad \text{for all } i \in \mathbb{N}.$$

And the interpretation of the connectives $(\bot, \neg, \wedge, \vee)$ for the frame are as follows.

$\bot$: $\bot_F = \emptyset$

$\neg$: $\neg_F A = A^C$

$\wedge$: $A \wedge_F B = A \cap B$

$\vee$: $A \vee_F B = A \cup B$

VS_{CPL} is thus a semantics defined over a single model, and it is evident that $VS_{\mathsf{CPL}} \equiv V_{\mathsf{CPL}}$. Since the defined frame interprets the connectives as the standard boolean set functions, we call this *the boolean monadic frame on $\mathscr{V}$* and denote it $\mathfrak{B}_{\mathscr{V}}$. This method of constructing a monadic semantics from another semantics will be reviewed is Section 4.1.2.

Valuation-to-model semantics $\mathrm{VM}_{\mathrm{CPL}}$

An equally natural approach is to encode the valuations as models. We start by considering the singleton universe $\{\emptyset\}$ also known as $\mathbf{1}$.

$$U = \mathbf{1}.$$

Then by definition and the identification $\mathscr{P}(\mathbf{1}) = \mathbf{2}$

$$\mathrm{Mod}(\mathbf{1}) = \{ M \mid M : \mathbb{N} \to \mathscr{P}(\mathbf{1}) \} = \mathscr{V}$$

So that the set of models on this domain is exactly the set of valuations.

To get a logic equivalent to V_{CPL} all we need to do is define a frame with universe $\mathbf{1} = \{\emptyset\}$ and translate the definitions of the connectives as set functions on $\mathbf{2} = \{0, 1\}$. It should not be to anyone's surprise, that what we get is $\mathfrak{B}_{\mathbf{1}}$, the boolean frame on $\mathbf{1}$.

Definition 4.9. Let VM_{CPL} denote the monadic semantics of the boolean frame on the singleton set $\mathbf{1}$

$$VM_{\mathsf{CPL}} = (\mathfrak{B}_{\mathbf{1}}, \mathrm{Mod}(\mathbf{1})).$$

Clearly then $VM_{\mathsf{CPL}} \equiv VS_{\mathsf{CPL}} \equiv V_{\mathsf{CPL}}$.

Boolean monadic semantics

The previous two examples show how the V_{CPL} can be directly translated into monadic semantics in two different ways: with the valuations of samples in a single model, or with valuations as models on a singleton domain. Both these semantics are based on a boolean frame. They can be seen as *degenerate forms* of monadic semantics, since neither really captures the *monadic* idea of the setting, by having only one model, or having a singleton universe. If we however consider multiple models in larger domains, we find that we can use boolean frames to construct a whole family of semantics for CPL that more naturally fits a monadic reading.

Definition 4.10 ($B(U)$)**.** For any non empty set $S \neq \emptyset$ we define the *boolean monadic semantics* $B(U)$ for the universe U as the monadic semantics of the boolean monadic frame on U

$$B(U) = (\mathfrak{B}_U, \mathrm{Mod}(U)).$$

Note specifically that $B(\mathbf{1}) = VM_{\mathsf{CPL}}$

Claim 4.11. *For every non-empty set $U \neq \emptyset$,*

$$B(U) \equiv VM_{\mathsf{CPL}} \equiv VS_{\mathsf{CPL}} \equiv V_{\mathsf{CPL}}$$

Proof. We have already established the last two equivalences, and we are left to show that $B(U)$ is logically equivalent to the rest. We first prove that $B(U) : \Gamma \vDash \phi$ implies that $VM_{\mathsf{CPL}} : \Gamma \vDash \phi$.

Consider any model $M \in \mathrm{Mod}(\mathbf{1})$. This is confined to interpret every predicate either as the empty set, or the singleton set $\{\mathbf{0}\}$, which is the full universe. Then a model $M' \in \mathrm{Mod}(U)$ mimics this interpretation on the larger domain.

$$p_i^{M'} = \begin{cases} \emptyset & \text{if } p_i^M = \emptyset \\ U & \text{if } p_i^M = \mathbf{1} \end{cases}$$

We then have that for all samples $s \in U$ and all formulas ψ

$$B(U) : M', s \vDash \psi \ \text{ if and only if } \ VM_{\mathsf{CPL}} : M, \mathbf{0} \vDash \psi$$

This means that the set of models $\mathrm{Mod}(\mathbf{1})$ are represented in $\mathrm{Mod}(U)$. Thus if $\Gamma \vDash \phi$ holds for every model in $\mathrm{Mod}(U)$, it necessarily also holds for every model in $\mathrm{Mod}(\mathbf{1})$ and the implication between the entailments is proven.

For the other direction we prove that $\mathrm{V}_{\mathsf{CPL}} : \Gamma \vDash \phi$ implies that $B(U) : \Gamma \vDash \phi$.

Consider a model $M \in \mathrm{Mod}(U)$ and a sample $s \in U$. Given this point and this model we can construct a valuation v mimicking set memberhood of s in interpretations of atomic formulas by M:

$$v(i) = \begin{cases} 1 & \text{if } s \in M(i) \\ 0 & \text{if } s \notin M(i) \end{cases}$$

We can then see by working through the constructing arguments of these semantics that for any formula ψ

$$B(U) : M, s \vDash \psi \text{ if and only if } \mathrm{V}_{\mathsf{CPL}} : v \vDash \psi$$

thus every choice of model and sample can be represented as a valuation, so that if $\Gamma \vDash \phi$ holds for every valuation, it also holds for every sample in every model in $\mathrm{Mod}(U)$. This concludes the proof. □

By these examples we have not only shown that CPL can be expressed in this style of semantics, but it can be done in a myriad of ways. This hints to a versatility available in monadic semantics that may not be needed for classical propositional logic. We will see how this style of semantics may be more appropriately utilised. First we can note that monadic semantics also houses the local semantics of Kripke-frames.

Local Kripke frames

We can construct an monadic semantic *Kripke* frame F on a universe U_F for the language with signature $(\bot, \neg, \wedge, \vee, \Box, \Diamond)$ by fixing a relation $R_F \subseteq U_F \times U_F$ on the universe. To get a semantics for a modal logic we then define the connectives as follows

$\bot :\ \bot_F = \emptyset$

$\neg :\ \neg_F A = A^C$

$\wedge :\ A \wedge_F B = A \cap B$

$\vee :\ A \vee_F B = A \cup B$

$\Box :\ \Box_F(A) = \{\, s \in D \mid sR \subseteq A \,\}$

$\Diamond :\ \Diamond_F(A) = RA$

Here sR denotes the set of all samples related to s by R, i.e. $sR = \{\, t \in D \mid sRt \,\}$, and similarily $RA = \{\, s \in D \mid \text{ there exists } t \in A \text{ s.t. } sRt \,\}$. The monadic semantics $(F, \mathrm{Mod}(U_F))$ of the frame then coincides with a traditional definition of a local kripke-frame semantics.

We can also construct intuitionistic frame semantics for propositional logic by choosing the relation to be a pre-order (reflexive and transitive), defining an interpretation $\rightarrow_F$ for intuitionistic implication, and restricting the class of models to be monotone with respect to the order. It is most common to describe Modal logic semantics by considering classes of frames instead of a single one. This is a natural generalisation, but we postpone a general discussion of generalisations in Chapter 7 since it is not needed for the type of logics we are mainly focused on describing.

4.1.1 Maximal semantics

As exemplified, a logic does not in general have a unique monadic semantics, not even within the same frame as can be seen for classical propositional logic. However, as a consequence of the consensus style of semantics we have chosen, if two semantics $(F, \mathcal{M})$ and $(F, \mathcal{M}')$ on the same frame defines the same logic, then so does $(F, \mathcal{M} \cup \mathcal{M}')$. By this we can construct the maximal equivalent semantics of a frame:

Definition 4.12. Let $(F, \mathcal{M})$ be an monadic semantics for the logic L. Then the maximal equivalent monadic semantics, denoted $(F, \widehat{\mathcal{M}})$ is the semantics where the class of models is the union of the classes of all logically equivalent monadic semantics in the frame. i.e:

$$\widehat{\mathcal{M}} = \bigcup \{ \mathcal{N} \mid (F, \mathcal{N}) \equiv (F, \mathcal{M}) \}$$

$(F, \widehat{\mathcal{M}})$ is unique and is called *the maximal semantics of the logic* L *on the frame* F.

By construction a semantics is always logically equivalent to its maximal equivalent semantics, so coining this the maximal semantics of the logic on the frame is consistent with the rest of the terminology. A maximal semantics on a frame for a logic is arguably also the most *correct* representation of the logic on that frame, since the class of models of the semantics is then fully determined by the logic and the frame. This means that for a maximal semantics on a given frame, any interpretation of a formula that is excluded or included by some model is so for reasons inherent of the logic.

Example 4.13. For classical propositional logic, the maximal equivalent semantics of VS_{CPL} is the full boolean monadic semantics $B(\mathscr{V})$. It is also clear that every frame semantics is its own maximal semantics.

The use of maximal objects is standard in many parts of mathematics. Given a partial order of objects, by proving that if an object lacks a certain property we can construct a strictly greater object, we can prove that a maximal object must have the property even if it is not provable in general. Apart from the fact that it gives us a unique semantics of the logic for a frame, this is the way in which we intend to utilise the notion of maximal semantics in Section 5.3.[1]

4.1.2 Monadic dual for truth-compositional semantics

The construction of the monadic semantics VS_{CPL} from the valuation semantics $\mathrm{V}_{\mathrm{CPL}}$ was done in a very direct way. We used the defined sets and the fact that $\mathrm{V}_{\mathrm{CPL}}$ is the logic of a truth-compositional framework to utilise dual functions in our definition of a specific monadic frame on $\mathscr{V}$. This procedure can be mirrored for every truth-compositional semantics.

Theorem 4.14. *For every truth-compositional semantic framework there is a monadic semantics defining the same logic.*

Proof. Let $\mathscr{F}$ be a truth-compositional semantic framework for the language $\mathcal{L}$ with semantic domain $\mathscr{O}$. Then for every connective $* \in \mathrm{Sing}(\mathcal{L})$ we can choose a dual function f_*. Let $F_{\mathscr{F}}$ be the monadic frame with $\mathscr{O}$ as universe and $*_{F_{\mathscr{F}}} = f_*$ for each connective $* \in \mathrm{Sing}(\mathcal{L})$. We then define the monadic model $M_{\mathscr{F}}$ by the defined sets of the atomic formulas

$$M_{\mathscr{F}}(i) = [\![p_i]\!]_{\mathscr{F}} \text{ for all } i \in \mathbb{N}.$$

Then by induction over the complexity of formulas we see that the monadic semantic $(F_{\mathscr{F}}, \{M_{\mathscr{F}}\})$ is logically equivalent to the logic of $\mathscr{F}$. □

The monadic semantics $(F_{\mathscr{F}}, \{M_{\mathscr{F}}\})$ constructed as above can be called a *direct dual* of the framework $\mathscr{F}$. Its maximal equivalent semantics $(F_{\mathscr{F}}, \widehat{\{M_{\mathscr{F}}\}})$ is then called a *full dual* of $\mathscr{F}$. Only when the dual functions are unique, that is when $[\![\mathcal{L}]\!]_{\mathscr{F}} = \mathscr{P}(\mathscr{O})$, are the direct dual and full dual monadic semantics unique. With this terminology, $B(\mathscr{V})$ is *the* full dual monadic semantics of $\mathrm{V}_{\mathrm{CPL}}$.

4.2 Monadic team semantics

We are now interested in how to express team semantics building from monadic semantics. From Theorem 4.14 it is at least clear that this is possible for specific truth-compositional semantics, but we are interested in a coherent method for all team semantics.

[1] Note also that the definition of maximal semantics is fully constructive. There is no choice function needed, so the maximal semantics exists without *the axiom of choice* or *Zorn's lemma*.

Since sample points play a similar role in monadic semantics, as that of valuations in valuational semantics, we could consider constructing team logics by collecting collections of samples. In valuational semantics we needed to construct a new type of semantics, but maybe surprisingly, we can express semantics for teams as a *special case* of monadic semantics. A collection of samples X is a subset of the universe $X \subseteq U$, therefore it is also an element in the powerset of that universe. $X \in \mathscr{P}(U)$. Thus, if we define an monadic semantics whose universe is the powerset of some underlying set, we get a type of team semantics. we call this MT-frames, and MT-semantics.

Definition 4.15.

- An *MT-frame* is a monadic frame where the universe $U_F = \mathscr{P}(S)$, is the powerset of an underlying set S.
- An *MT-semantics* is then a pair $(F, \mathcal{M})$, where F is an MT-frame.

Since the samples in an MT-semantics are sets of points, we call them *teams* and denote them by capital letters X, Y, Z.

The definition above does not actually fully correspond to the valuational team semantics described by Yang and Väänänen. In their definitions they fixed the interpretation of propositional variables in teams to be consensus truth for the members, that is, for a valuation team X

$$X \vDash p_i \text{ if and only if for all } s \in X : \ s(i) = 1.$$

An atomic formula is thus always flat and defines a powerset of valuations by Theorem 3.12. We can call this the *flat atom assumption* of valuational team semantics. This also means that we can pick out the corresponding MT-semantics by restricting to the class of monadic models interpreting atoms as powersets. We call this *MT-semantics with powerset atoms*.

Definition 4.16 (MT^{PA}-semantics)**.**

- A *MT^{PA}-semantics* is an MT-semantics $(F, \mathcal{M})$ on a universe $\mathscr{P}(S)$, for which every model $\mathcal{M}$ interprets every atomic formula as the powerset of a subset of the underlying set S. i.e.

$$\text{for all } M \in \mathcal{M}, \text{and } i \in \mathbb{N} : \quad M(i) = \mathscr{P}(T) \quad \text{for some } T \subseteq S.$$

  The superscript PA stands for *powerset atoms*.
- Let $Mod^{PA}(\mathscr{P}(S))$ denote the collection of all models of this kind for a universe $\mathscr{P}(S)$.

Note that in this terminology, it is MT^{PA}-semantics that directly correlates to the standard valuational team semantics, and not the more general class. We will see though in the next chapter that this restriction is responsible for some of the peculiarities of team logics.

We are now ready to describe an MT-semantics for propositional dependence logic.

The semantics MT$_{\mathrm{PD}}$

We will define an MT^{PA}-semantics we call MT$_{\mathrm{PD}}$ for the language with signature $(\bot, \neg, \wedge, \vee, \otimes)$ equivalent to the valuational team semantics of propositional dependence logic VT$_{\mathrm{PD}}$ described in Section 3.1.1. Since VT$_{\mathrm{PD}}$ is truth-compositional we can utilise the dual functions described in the proof of that fact and generate an appropriate MT-frame.

Definition 4.17. Let F_{PD} be the monadic frame on the universe $\mathcal{VT} = \mathscr{P}(\mathcal{V})$ for the language $(\bot, \neg, \wedge, \vee, \otimes)$ defined by the following interpretation of the connectives, were A, B denotes collections of sample-teams:

$\bot : \ \bot_{F_{\mathrm{PD}}} = \{\emptyset\}$,

$\neg : \ \neg_{F_{\mathrm{PD}}} A = \{ X \mid \text{ for all } Y \in A, X \cap Y = \emptyset \}$,

$\wedge : \ A \wedge_{F_{\mathrm{PD}}} B = A \cap B$,

$\vee$: $A \vee_{F_{\mathtt{PD}}} B = A \cup B$, and

$\otimes$: $A \otimes_{F_{\mathtt{PD}}} B = \{ X \cup Y \mid X \in A \text{ and } Y \in B \}$.

We could then settle for a direct dual semantics with a single model, but in fact, we will make our semantics for the full set $Mod^{PA}(\mathscr{VT})$.

Definition 4.18. Let $\mathrm{MT}_{\mathtt{PD}}$ denote the MT^{PA}-semantics $(F_{\mathtt{PD}}, Mod^{PA}(\mathscr{VT}))$.

Claim 4.19. *$MT_{\mathtt{PD}}$ is logically equivalent to $VT_{\mathtt{PD}}$.*

Proof. The direct dual monadic model of $\mathrm{VT}_{\mathtt{PD}}$ is clearly in $Mod^{PA}(\mathscr{VT})$, so $\mathrm{MT}_{\mathtt{PD}} : \Gamma \vDash \phi$ implies $\mathrm{VT}_{\mathtt{PD}} : \Gamma \vDash \phi$. what is left to show is the opposite implication.

Consider any monadic model $M \in \mathrm{Mod}^{PA}(\mathscr{VT})$ and any sample-team $X \in \mathscr{VT}$. For all points $s \in X$ we can then find a valuation v_s^M mimicking the monadic memberhood of s in the valuation:

$$v_s^M(i) = \begin{cases} 1 & \text{if } s \in M(i) \\ 0 & \text{if } s \notin M(i) \end{cases} \qquad \text{for all } i \in \mathbb{N}.$$

If we then construct the valuation team $V_X^M = \{v_s^M\}_{s \in X}$ collecting all these mimicking valuations we can, by induction over complexity of formulas, prove that for every formula ψ

$$\mathrm{MT}_{\mathtt{PD}} : M, X \vDash \psi \text{ if and only if } \mathrm{VT}_{\mathtt{PD}} : V_X^M \vDash \psi.$$

Thus every pointed model in the monadic team semantics has an equivalent representation in the valuational team semantics. Therefore, if $\mathrm{VT}_{\mathtt{PD}} : \Gamma \vDash \phi$ then $\mathrm{MT}_{\mathtt{PD}} : \Gamma \vDash \phi$ and the two semantics are logically equivalent. □

It is worth noticing that for $\mathrm{MT}_{\mathtt{PD}}$ it is not important that its universe is constructed specifically on the sets of valuation functions $\mathscr{VT}$. For the interpretations of the connectives it only plays the role of a sufficiently large powerset. Choosing exactly $\mathscr{VT}$ makes the equivalence proof easier, and the construction can be done by dualisation of the valuational semantics, but from the perspective of the monadic semantics it only represents an arbitrary powerset.

In $\mathrm{MT}_{\mathtt{PD}}$ not only in the direct dual model, but truth in every model separately works very much like $\mathrm{VT}_{\mathtt{PD}}$ on its own. This means that the downwards closure for the defined sets of $\mathrm{VT}_{\mathtt{PD}}$, also holds for every model in $\mathrm{MT}_{\mathtt{PD}}$.

Theorem 4.20. *Let $(F, \mathcal{M}) = MT_{\mathtt{PD}}$. Then for every model $M \in \mathcal{M}$ and every formula ϕ, ϕ_F^M is a non-empty downwards closed set.*

Proof. For every model $M \in \mathrm{Mod}^{PA}(\mathscr{VT})$ and every atomic formula p_i we have that p_i^M is a non empty downwards closed set. The rest of the proof is to assert that this property is preserved for every connective in the language, and the proof is identical to that in Claim 3.4. □

We have thus created a monadic semantics for propositional dependence logic. When we have translated the notions of flatness, we will be able to further show in Theorem 4.26 that $\mathrm{MT}_{\mathtt{PD}}$ is in fact the full dual monadic semantics generated by the dual functions given.

4.2.1 Flatness in MT-semantics

We want to be able to show similar results about the relation between a monadic semantics and an MT-semantics as those established for their valuational counterparts. We first need to translate the notion of flatness.

Definition 4.21. For an MT-semantics $(F, \mathcal{M})$ a formula ϕ is considered *flat* if for every model $M \in \mathcal{M}$, every sample team $X \in U_F = \mathscr{P}(S)$:

$$M, X \vDash \phi \iff \text{for all } s \in X \text{ we have that } M, \{s\} \vDash \phi.$$

An MT-semantics is *flat* if every formula of the language is flat for it.

We then get the anticipated connection to the interpretations.

Theorem 4.22. *ϕ is flat if and only if ϕ^M is a powerset for every model $M \in \mathcal{M}$.*

Proof. The proof can be directly adapted from that of Theorem 3.12. □

From this we can easily define *flatness preserving* for MT-semantics too:

Definition 4.23. A connective $*^n$ is *flatness preserving* in an MT-frame F whith universe $\mathscr{P}(S)$ if its interpretation maps powersets to powersets, that is:

$$\text{for all } S_1, \ldots, S_n \subseteq S: \quad *^n_F\ (\mathscr{P}(S_1), \ldots, \mathscr{P}(S_n)) = \mathscr{P}(T) \quad \text{for some } T \subseteq S.$$

A connective is flatness preserving in an MT-semantics if it is on its frame.

We can now translate the results from Section 3.2.1 to MT-semantics.

Theorem 4.24. *For the semantics MT_{PD} the connectives $\bot, \neg, \wedge, \otimes$ are flatness preserving, but $\vee$ is not. Consequently, since all atomic formulas p_i are flat, so is every $\vee$-free formula.*

Proof. The proof can be directly translated from that of Theorem 3.10. □

Theorem 4.25. *For the semantics MT_{PD} every negated formula is flat. Furthermore for any formula ϕ, $\neg\neg\phi \vDash \phi$ if and only if ϕ is flat.*

Proof. This is observed and proven in the same way as in Note 3.11 and the proof of Claim 3.15. □

With this established we can finally prove the maximality of $\mathrm{MT}_{\mathrm{PD}}$.

Theorem 4.26. *MT_{PD} is a maximal monadic semantics and thus the full dual monadic semantics of VT_{PD} for the dual functions given.*

Proof. We know that in $\mathrm{VT}_{\mathrm{PD}}$ every atomic formula is flat, and thus $\mathrm{VT}_{\mathrm{PD}} : \neg\neg p_i \vDash p_i$ for all $i \in \mathbb{N}$. This means that for any MT-semantics $(F_{\mathrm{VT}_{\mathrm{PD}}}, \mathcal{M})$ on the frame $F_{\mathrm{VT}_{\mathrm{PD}}}$ that is logically equivalent must have flat atoms.

$$\text{For all } M \in \mathcal{M} \text{ and every } i \in \mathbb{N} \text{ we have that } M(i) = \mathscr{P}(T) \text{ for some } T \subseteq \mathscr{V}.$$

We can therefore conclude that $\mathcal{M} \subseteq \mathrm{Mod}^{PA}(\mathscr{V}\mathscr{T})$. Since $\mathrm{VT}_{\mathrm{PD}} = (F_{\mathrm{VT}_{\mathrm{PD}}}, \mathrm{Mod}^{PA}(\mathscr{V}\mathscr{T}))$ it is the maximal semantics on this frame logically equivalent to $\mathrm{VT}_{\mathrm{PD}}$. □

We can also reiterate the impotency of flat semantics in the monadic setting.

Theorem 4.27. *If an MT-semantics on the universe $\mathscr{P}(S)$ is flat it is logically equivalent to a monadic semantics with the underlying set S as its universe.*

Proof. If an MT-semantics $(F, \mathcal{M})$ on $\mathscr{P}S$ is flat, then it is logically equivalent to an MT-semantics on the same universe for which every connective is flatness preserving, so we may assume this is the case. It is also clear that $\mathcal{M} \subseteq \mathrm{Mod}^{PA}(\mathscr{P}(S))$, and thus, by the infectivity of the powerset operation, we can construct a monadic semantics $(F_S, \mathcal{M}_S)$ on S by *stripping of* the powerset function everywhere.

$$\mathcal{M}_S = \{\, M_S \in \mathrm{Mod}(S) \mid \text{there exists } M \in \mathcal{M} : \mathscr{P}(M_S(i)) = M(i) \text{ for all } i \in \mathbb{N} \,\}$$

and for every connective $*^n$ in the language

$$*^n_{F_S}(S_1, \ldots, S_n) = T \text{ such that } \mathscr{P}(T) = *^n_F(\mathscr{P}(S_1), \ldots, \mathscr{P}(S_1)) \text{ for all } S_1 \ldots S_n \subseteq S. \qquad \square$$

In this section we have both found an MT-semantics for propositional dependence logic and translated the main results presented about team logics from valuation team semantics. This shows to some extent that the definition of monadic semantics is flexible enough to express interesting logical ideas. In the next chapter we see how this new setting can help us prove more results, in particular in regards to substitutionality.

Chapter 5

Interpretation sets and semantic criteria for substitutionality

Valuational semantics and monadic semantics are similar in construction. In this chapter we see however that the monadic formulation gives birth to notions that would be much harder, if not impossible, to formulate in the valuational treatment. We present the notion of *interpretation sets* as the the vehicle for this discussion. Our main purpose for this is to analyse substitutionality. Note that all contructions and almost all results apply for *all* monadic semantics and not only monadic team semantics specifically, even if they motivated our research.

5.1 Interpretation sets

Valuational team semantics was created to give the possibility to talk about notions of dependence and independence of propositional variables. This forced the move from semantics based on single valuations to a semantics based on teams of valuations. Monadic semantics plays a similar role to truth-compositional semantics in general. Every truth-compositional semantics can give rise to a single model in a monadic frame, but if we treat a monadic semantic with multiple models, we can identify notions of dependence and independence for the *interpretations of formulas* in the models. This outer observation of dependence and independence will prove itself useful when semantically identifying substitutionality. We start by defining what we mean by the *interpretation set* of a formula in a monadic semantics.

Definition 5.1. Given a monadic semantics $(F, \mathcal{M})$ we define *the interpretation set* of a formula ϕ, denoted $\langle\!\langle\phi\rangle\!\rangle_F^{\mathcal{M}}$ to be the collection of interpretations of the formula by the models of the semantics, i.e.

$$\langle\!\langle\phi\rangle\!\rangle_F^{\mathcal{M}} = \left\{ \phi_F^M \;\middle|\; M \in \mathcal{M} \right\}$$

We define the interpretation set of a set of formulas Γ to be the union of the interpretation sets of the formulas

$$\langle\!\langle\Gamma\rangle\!\rangle_F^{\mathcal{M}} = \bigcup_{\gamma\in\Gamma} \langle\!\langle\gamma\rangle\!\rangle_F^{\mathcal{M}}$$

When the class and frame is clear from the context, the super- and subscripts are omitted.

The interpretation set of a formula tells you how general or restricted the possible different interpretations of it is in the semantics. This can reveal apparent structure in what sample points of the universe that may validate the formula in some model. We name some properties for formulas regarding their interpretation sets.

Definition 5.2. For a fixed monadic semantics we define the following properties of a formula ϕ :

- ϕ is *inconsistent* if $\langle\!\langle\phi\rangle\!\rangle = \emptyset$
- ϕ is *constant* if $\langle\!\langle\phi\rangle\!\rangle$ is a singleton or the empty set.
- ϕ is *broader* than a formula ψ if $\langle\!\langle\phi\rangle\!\rangle \supseteq \langle\!\langle\psi\rangle\!\rangle$
- ϕ is *weakly general* if it is broader than every formula i.e $\langle\!\langle\phi\rangle\!\rangle = \langle\!\langle\mathcal{L}\rangle\!\rangle$.

We say that a semantics has *weakly general atoms* if every atomic formula p_i is weakly general.

The motivation for this terminology is to see models as ways of interpreting predicates for a universe of sample objects. As example: If we see British people as interpreters of the English language, there are a certain set of things that different individuals would consider being a *cup* and a certain set of things that would be considered a *mug*. For some of these things there will be total agreement on their receptacle status, and for others it will not. The interpretation sets of the predicates *(is a cup)* and *(is a mug)* captures this information. If we then survey Americans or Australians instead, we might get a different set of interpretations.

The idea of weak generality is the notion that a formula can be interpreted as the same set as any other formula, albeit not by the same model. This means that a weakly general formula can play the role of any other formula by some interpretation. We will also give a stronger notion of generality in the next section.

Example 5.3. To exemplify this notion, we can see how it relates to the specific semantics we have defined so far.

- The direct dual semantics VS_{CPL} has only got a single model, so every formula is constant and no formula is weakly general.
- The boolean frame semantics $B(U)$ on any set U uses the full set of models, allowing propositional variables to be interpreted as any possible subset of the universe. It is therefore clear that these semantics have all got weakly general atoms.
- In the team semantics MT_{PD} all models interpret propositional variables as powersets, but an arbitrary formula may be interpreted as any non-empty downwards closed set. Therefore MT_{PD} has not got weakly general atoms.

By the two first examples we note that constancy of formulas and weak generality of atom may differ for logically equivalent semantics, showing that the logic does not determine these traits in general. We will however see that by using the notion of maximality, and choose a canonical semantics on a frame, that the notion of generality is connected to the substitutionality of the logic.

Using this terminology we can see that the restriction of the class of models in MT^{PA}-semantics give rise to a corresponding restriction on the interpretation sets of atomic formulas.

Note 5.4. An MT-semantics$(F, \mathcal{M})$ on the universe $\mathscr{P}(S)$ is an MT^{PA}-semantics if and only if

$$\text{if } X \in \langle\!\langle p_i \rangle\!\rangle, \text{ then } X = \mathscr{P}(T) \textit{ for some } T \subseteq S \qquad \text{for all } i \in \mathbb{N}.$$

The proof is direct from the definitions. This result does not look too surprising, but it directly follows that we can relate weak generality and flatness in MT^{PA}-semantics.

Theorem 5.5. *Every MT^{PA}-semantics with weakly general atoms is flat.*

Proof. Since MT^{PA}-semantics is defined by the restriction to models where atomic predicates interpreted as some powerset, these atoms being general means that every formula is interpreted as some powerset in every model of the semantics. Then by Theorem 4.22 the semantics is flat. □

5.2 Restricted subclasses and independence

Using the idea of collecting interpretations for a class of models $\mathcal{M}$ on a monadic frame, it will be useful to identify the subclass of models that agree on some interpretation. Identifying these restricted subclasses will allow us to observe when the interpretation of two formulas are *independent* or not. The formal definitions are as follows:

Definition 5.6 (restriction classes)**.** Let $(F, \mathcal{M})$ be a monadic semantics.

For every formula ϕ and every subset S of the universe, define the *subclass* $\mathcal{M}|_{\phi=S}$ as the restriction to models where ϕ is interpreted as S

$$\mathcal{M}|_{\phi=S} : \left\{ M \in \mathcal{M} \;\middle|\; \phi_F^M = S \right\}.$$

This may of course be an empty class. Let $\mathcal{M}|_{\{\phi_i=S_i\}_{i\in I}}$ denote the class restricted with multiple formulas. If $S = \phi^M$ for a model M, then abbreviate $\mathcal{M}|_{\phi^M} = \mathcal{M}|_{\phi=\phi_F^M}$. Similarly define $\mathcal{M}|_{\{\phi_i\}_{i\in I}^M}$. Note: If $M \in \mathcal{M}$, these classes are never empty.

Definition 5.7. Let $(F, \mathcal{M})$ be a monadic semantics.

We say that ϕ *is independent of* $\{\psi_i\}_{i\in I}$ in $(F, \mathcal{M})$ if

$$\langle\!\langle \phi \rangle\!\rangle^{\mathcal{M}|_{\{\psi_i^M\}_{i\in I}}} = \langle\!\langle \phi \rangle\!\rangle^{\mathcal{M}} \quad \text{for all } M \in \mathcal{M}$$

We say that $(F, \mathcal{M})$ has *independent atoms* if p_j is independent of $\{p_{i_k}\}_{k\in K}$ whenever $j \notin \{i_k\}_{k\in K}$. A semantics has *finitely independent atoms* if atoms p_j are independent of finite sets of atoms $\{p_{i_1}, \ldots, p_{i_n}\} \not\ni p_j$.

To construct a semantics with independent atoms is in line with the idea of a logic, especially a compositional one. It means that the choices of interpretation of one variable does not effect your possible choices of interpretations of the others. Semantics that do not have independent atoms can have deductions of the style

$$p_i \vDash p_j$$

for i ≠ j. This is true in some semantics. For example, if we consider models representing standard interpretations of English that interpret p_i as the predicate (Is a dog) and p_j as the predicate (Is a mammal), then the entailment above would hold. This would however not typically be consider this a *logical* deduction by most scholars [SP16; CE19]. We have allowed this type of entailment in our general notion of logic, but we can actively avoid such ill-behaved logics by demanding independent atoms in the semantics. This is extra important if we want to describe a logic that proves results about dependency and independency, since it would then not be desirable to have underlying dependencies in the semantics.

We use a similar construction as that of independence to define *strong generality* for atomic formulas. This is the property of an atom to be weakly general for every restricted model class that does not fix it.

Definition 5.8. An atom p_j is *strongly general* in the semantics $(F, \mathcal{M})$ if for all $\mathcal{M}' = \mathcal{M}|_{\{p_{i_j}=S_j\}_{j\in J}}$ where $p_k \notin \{p_{i_j}\}_{j\in J}$,

$$\langle\!\langle \mathcal{L} \rangle\!\rangle^{\mathcal{M}'} = \langle\!\langle p_k \rangle\!\rangle^{\mathcal{M}'}$$

and it is *finite strongly general* if it holds for finite J.

A semantics$(F, \mathcal{M})$ has *(finite) strongly general atoms* if all atoms are (finite) strongly general.

We describe a notion of *generic* atoms when any two unrestricted atoms can take the same interpretations. This captures a semantic notion of one-sortedness of atoms.

Definition 5.9. A semantics $(F, \mathcal{M})$ has *generic atoms*, if for all restriction classes $\mathcal{M}' = \mathcal{M}|_{\{p_{i_j}=S_j\}_{j\in J}}$ and all $p_{k_1}, p_{k_2} \notin \{p_{i_j}\}_{j\in J}$:

$$\langle\!\langle p_{k_1} \rangle\!\rangle^{\mathcal{M}'} = \langle\!\langle p_{k_2} \rangle\!\rangle^{\mathcal{M}'}$$

Clearly strongly general atoms are also generic. We will not discuss genericness in any extent, but want to highlight that we can find a semantic criteria that corresponds to our guiding idea of the logic being one-sorted. We have in this paper fixed the type of language to only be generated over a single set of atomic variables, but from the lack of assumed substitution of the logic, we can observe that this never was an actual restriction. By asserting generic atoms in the semantics we can actually enforce proper one-sortedness.

Corollary 5.10. *The following results follow directly:*

- *If a monadic semantics has independent atoms, then each atomic formula p_i is independent of every formula ϕ it does not support ($p_i \notin Sup(\phi)$).*
- *If a monadic semantics has strongly general atoms it has weakly general and generic atoms.*
- *If a monadic semantics has independent and weakly general atoms it has strongly general atoms.*
- *The full semantics of a monadic frame has always got independent and general atoms.*

When a semantics has independent atoms, since weak and strong generality for atoms coincide, we drop the qualifier from the terminology and say that a semantic has *independent general atoms*.

Example 5.11. We can see how these notions relate to the monadic semantics we have described so far.

- The direct dual semantics VS_{CPL} has only got a single model, so the restriction classes are always either empty or the whole class of the semantics. Therefore VS_{CPL} has trivially got independent atoms. We also see that every atomic formula is interpreted as its own distinct subset of the universe by the model. VS_{CPL} has thus not got generic atoms. The same will hold in general for every direct dual monadic semantics, and it may convince us that these are not particularly nice semantics for our purposes.

- The boolean frame semantics $B(U)$ on any set U allows propositional variables to be interpreted as any possible subset of the universe, and any combination of interpretations of different atomic formulas is represented in the set of models. It is therefore clear that these semantics have independent and general atoms.

- In the team semantics MT_{PD} every allowed interpretation of atomic formulas are presented together with every possible interpretation of other atomic formulas. Therefore all atoms are independent. Furthermore, there is no discernable differences between treatment of different atoms and it is therefore clear that they are generic. MT_{PD} has thus got independent and generic atoms but, as observed before, not general.

It is natural to wonder if strong generality for atoms also implies independence. This is not the case however, which can be shown by the following degenerate example.

Example 5.12. Let $\mathcal{L}$ be the language of empty signature, that is, all formulas are of the form p_i. A monadic frame for this language is then just a choice of universe. We define a frame F by choosing a singleton universe:

$$U_F = \{1\}.$$

Now consider the two models that map every index to the empty set or the full universe respectively.

$$M_0(i) = \emptyset \quad \text{and} \quad M_1(i) = \{1\} \quad \text{ for all } i \in \mathbb{N}.$$

Then the semantics $(F, \{M_0, M_1\})$ has strongly general atoms, but they are not independent.

The logic of $(F, \{M_0, M_1\})$ can be fully described as follows

$$p_i \nvdash \ , \ \nvdash p_i, \text{ and } p_i \vdash p_j \quad \text{ for every formula } p_i, p_j \in \mathcal{L}.$$

We call this the *meaningful almost inconsistent logic* of the empty language[1]. It is also evident that $(F, \{M_0, M_1\})$ is maximal.

We can however find another property that is connected to strong generality and equivalent for the finite counterparts.

Definition 5.13. A monadic semantics $(F, \mathcal{M})$ has the *(finite) atomification property* if for any (finite) restriction set of atoms $R = \{p_{i_k}\}_{k \in K}$, any formula ϕ fully supported by this set, i.e $\text{Sup}(\phi) \subseteq R$ and any $p_j \notin R$ we have that :

for every model $M \in \mathcal{M}$ there is a model $M' \in \mathcal{M}$ such that $p_i^M = p_i^{M'}$ for all $p_i \in R$ and $\phi_F^M = p_j^{M'}$.

The idea is that if a semantics has the atomification property, then for any model M and formula ϕ we can find a model M' for which the interpretation of ϕ is the same as that of a *fresh* propositional variable.

Theorem 5.14. *If a monadic semantics has strongly general atoms it has the atomification property.*

[1]The notion of an almost inconsistent logics is used by Font [Fon16] as pathological examples. But since he does not allow an empty right hand side of formulas, he does not consider demanding $p_i \nvdash$. I call this the *meaningful* logic because the atoms are not inconsistent on their own, and thus have meaning in the logic. With what we have defined can we also identify additional semantics for degrees of inconsistency in their logics: $(F, \{M_0\})$ where $p_i \vdash$ and $\nvdash p_i$, $(F, \{M_1\})$ where $p_i \nvdash$ and $\vdash p_i$, and $(F, \emptyset)$ where $p_i \vdash$ and $\vdash p_i$ for every $i \in \mathbb{N}$.

Proof. Let $(F, \mathcal{M})$ be a monadic semantics, M be a model in $\mathcal{M}$, $R = \{p_{i_k}\}_{k\in K}$ be a restriction set of atoms and ϕ a formula fully supported by this set. Let $p_j \notin R$ be a *fresh* propositional variable.

Assume $(F, \mathcal{M})$ has strongly general atoms. Then

$$\phi_F^M \in \langle\langle \mathcal{L} \rangle\rangle^{\mathcal{M}|_{R^M}} = \langle\langle p_j \rangle\rangle^{\mathcal{M}|_{R^M}}$$

and thus there exists $M' \in \mathcal{M}|_{R^M}$ such that $p_j^M = \phi_F^M$. It also holds that $p_i^M = p_i^{M'}$ for all $p_i \in R$. □

Theorem 5.15. *A monadic semantics has finite strongly general atoms if and only if it has the finite atomification property.*

Proof. For one direction the proof is the same as for the previous theorem, but for only finite restriction sets

For the opposite direction, we first observe by finite atomification that for any $p_j, p_k \notin \{p_{i_1} \dots p_{i_n}\}$, we have for every model $M \in \mathcal{M}'$ a model $M' \in \mathcal{M}'$ such that $p_j^M = p_k^{M'}$ and thus $\langle\langle p_j \rangle\rangle^{\mathcal{M}'} = \langle\langle p_k \rangle\rangle^{\mathcal{M}'}$. Now we want to show, that for all $\phi \in \mathcal{L}$ and all $p_j \notin \{p_{i_1} \dots p_{i_n}\}$ we have

$$\langle\langle \phi \rangle\rangle^{\mathcal{M}'} \subseteq \langle\langle p_j \rangle\rangle^{\mathcal{M}'}.$$

By the first observation it is sufficient to prove this under the assumption that $p_j \notin Sup(\phi)$. Take as restriction set $\{p_{i_1} \dots p_{i_n}\} \cup Sup(\phi) \not\ni p_j$. Then finite atomification applies and for any model $M \in \mathcal{M}'$ there is a model $M' \in \mathcal{M}'$ such that $\phi^M = p_j^{M'}$, and hence $\langle\langle \phi \rangle\rangle^{\mathcal{M}'} \subseteq \langle\langle p_j \rangle\rangle^{\mathcal{M}'}$ and the semantic has finite strongly general atoms. □

The proof of the equivalence only works in the finite case simply because for a variable p_j an arbitrary restriction set can be $\{p_{i_k}\}_{k\neq j} \not\ni p_j$. Then this exhausts all propositional variables and there is no way to find two atoms not present in the restriction set. There might be some clever way around this problem in a proof, but since we cannot assume all atoms are interpreted in the same way semantically, it is dangerous to try to add or rearrange the propositional variables, and I have not been able to find a way to circumvent this issue.

5.3 Semantic criteria for substitutionality

Our main goal when developing the notions of interpretation sets is to categorise in the monadic semantics the substitutionality of its logic. Some semantic properties will only ensure that its logic is closed under singular substitution, but this also means that they still identify substitutionality in compact logics. We will give results both for full substitutionality and its singular counterpart.

5.3.1 Singular substitutions in maximal semantics

Theorem 5.16. *If a monadic semantics has a logic that is closed under singular substitutions, then its maximal semantics has strongly general atoms.*

Proof. We prove the contrapositive statement.
Assume that the maximal semantics $(F, \widehat{\mathcal{M}})$ does not have strongly general atoms. Then there is some p_j, some restriction set $R = \{S_{i_k}\}_{k\in K}$ with $\{i_k\}_{k\in K} \not\ni j$, and some ψ such that for $\mathcal{M}' = \widehat{\mathcal{M}}|_{\{p_{i_k} = S_{i_k}\}_{k\in K}}$

$$\langle\langle \psi \rangle\rangle^{\mathcal{M}'} \not\subseteq \langle\langle p_i \rangle\rangle^{\mathcal{M}'}.$$

Claim. *The logic of $(F, \widehat{\mathcal{M}})$ is not closed under singular substitution.*

Proof of claim. For the goal of a contradiction, assume the claim is false and the logic is closed under single substitution. Then $\Gamma \vDash \phi$ implies $\Gamma(p_i/\psi) \vDash \phi(p_i/\psi)$ for every choice of Γ, ϕ. Since $\langle\langle \psi \rangle\rangle^{\mathcal{M}'} \not\subseteq \langle\langle p_i \rangle\rangle^{\mathcal{M}'}$, there is some $M \in \mathcal{M}'$ such that

$$\psi^M \notin \langle\langle p_i \rangle\rangle^{\widehat{\mathcal{M}}}.$$

Let us then define the model $M' \in \mathcal{M}'$ with all atoms apart from p_j the same as for M, but assign $p_j^{M'} = \psi^M$:

$$M'(i) = \begin{cases} \psi & \text{if } i = j \\ M(i) & \text{if } i \neq j \end{cases}$$

Now construct the extended class $\widehat{\mathcal{M}}^+ = \widehat{\mathcal{M}} \cup \{M'\}$. We show that $(F, \widehat{\mathcal{M}}^+) \equiv (F, \widehat{\mathcal{M}})$. First, since $\widehat{\mathcal{M}}^+ \supseteq \widehat{\mathcal{M}}$ it is clear that if $(F, \widehat{\mathcal{M}}^+) : \Gamma \vDash \phi$ then $(F, \widehat{\mathcal{M}}) : \Gamma \vDash \phi$. For the opposite direction, assume $(F, \widehat{\mathcal{M}}) : \Gamma \vDash \phi$. Then by assumption for single substitutions

$$(F, \widehat{\mathcal{M}}) : \Gamma(p_j/\psi) \vDash \phi(p_j/\psi).$$

So for $M \in \mathcal{M}$ in particular

$$(F, \{M\}) : \Gamma(p_j/\psi) \vDash \phi(p_j/\psi).$$

But then by construction

$$(F, \{M'\}) : \Gamma(p_j) \vDash \phi(p_j)$$

for the altered model M'. Then by the definition of $\widehat{\mathcal{M}}^+$ and consensus semantics:

$$(F, \widehat{\mathcal{M}}^+) : \Gamma \vDash \phi.$$

Since this holds for any choice of Γ, ϕ we have proven that

$$(F, \widehat{\mathcal{M}}^+) \equiv (F, \widehat{\mathcal{M}})$$

By construction also $\langle\!\langle p_j \rangle\!\rangle^{\widehat{\mathcal{M}}^+} \supsetneq \langle\!\langle p_j \rangle\!\rangle^{\widehat{\mathcal{M}}}$ and thus $\widehat{\mathcal{M}}^+ \supsetneq \widehat{\mathcal{M}}$, which contradicts the maximality of $\widehat{\mathcal{M}}$. Under the initial assumption of $\widehat{\mathcal{M}}$ not having strongly general atoms, we have only assumed the logic to be closed under singular substitution, hence it cannot hold. □

By this we have proven the contrapositive of the theorem statement. □

Corollary 5.17. *If an MT^{PA}-semantics of a substitutional logic is maximal, it is flat.*

Proof. Since strong generality implies weak generality, this follows directly from the above theorem and Theorem 5.5. □

The above theorem provides a necessary condition in maximal monadic semantics for when the logic is closed under singular substitution. We can however also show that this condition is sufficient in every monadic semantics.

Theorem 5.18. *If a monadic semantics has strongly general atoms its logic is closed under singular substitution.*

Similarly, if a monadic semantics has finite strongly general atoms and its logic is compact, then it is substitutional.

Proof. Assume $\Gamma(p_i/\psi) \nvDash_{\mathcal{M}} \phi(p_i/\psi)$, then there is some M such that $\bigcap_{\gamma\in\Gamma}\gamma(p_i/\psi)^M \supsetneq \phi(p_i/\psi)^M$. Consider the class $\mathcal{M}' = \mathcal{M}|_{(\mathrm{Sup}(\Gamma,\phi)-p_i)^M}$. Since p_i is strongly general $\langle\!\langle\psi\rangle\!\rangle_{\mathcal{M}'} \subseteq \langle\!\langle p_i \rangle\!\rangle_{\mathcal{M}'}$. Then there is a model $M' \in \mathcal{M}'$ such that $\bigcap_{\gamma\in\Gamma}\gamma(p_i)^{M'} \supsetneq \phi(p_i)^{M'}$. Hence $\Gamma(p_i) \nvDash_{\mathcal{M}} \phi(p_i)$ This proves the first statement. For the second statement it is enough to observe that if the logic is compact we may assume Γ is finite. Then finite strongly general atoms suffices for the same proof. □

Corollary 5.19. *A maximal monadic semantics has strongly general atoms if and only if its logic is closed under singular substitution.*

Also, if a maximal monadic semantics has a compact logic, it has strongly general atoms whenever they are finite strongly general.

Proof. Both statements follow directly from Theorem 5.16 and Theorem 5.18 together. □

We have thus fully categorised when a logic is closed under singular substitutions as a criteria on its maximal monadic semantics.

5.3.2 Full substitutionality

The previous result also identifies full substitutionality under the assumption that the logic is compact, but not in general. For this I think we would need a stronger condition than just strongly general atoms. Otherwise we would have also proven that substitutionality is equivalent to closure under singular substitution. In this thesis we will however not be able to provide an example of a semantic with strongly general atoms for which the logic is not substitutional. Such example would have to define a non-compact logic, and the peculiarities of such considerations is outside the scope of this thesis. We will unfortunately not manage to completely categorise full substitutionality. We will be able to give a sufficient condition that we can also easily ensure.

The conditions we identify are less natural than previous properties, but we will see that we can still ensure to find semantics for which they hold. For the following definition recall the definition of a model in a monadic frame as a function $\mathbb{N} \to \mathscr{P}(U)$ where U is the universe of the frame. We also point out the interpretation of a countable product of sets $\prod_{i\in\mathbb{N}} A_i$, as the set of functions $\{ f \mid f(i) \in A_i \text{ for all } i \in \mathbb{N} \}$.

Definition 5.20. A monadic semantics $(F, \mathcal{M})$ with universe U has a *product-class of models* if $\mathcal{M} = \prod_{i\in\mathbb{N}} A_i$ for some collection of $A_i \subseteq \mathscr{P}(U)$.

We can directly incorporate the notion of interpretation with this definition

Theorem 5.21. *A monadic semantics $(F, \mathcal{M})$ has a product-class of models if and only if*

$$\mathcal{M} = \prod_{i\in\mathbb{N}} \langle\!\langle p_i \rangle\!\rangle.$$

Proof. The proof is direct from the definitions. □

The property of having a product-class of models thus implies that it is enough to determine the possible interpretations of every atomic formula independently in order to identify the class of models for the semantics. It is in fact a strengthening of the requirement to have independent atoms. It is also not a rare property, since it is always satisfied for the full semantics of a monadic frame. These are the statements of the following theorems.

Theorem 5.22. *Every full semantics of a monadic frame has a product-class of models.*

Proof. We observe directly that for any universe U, we have that $\mathrm{Mod}(U) = \mathscr{P}(U)^{\mathbb{N}}$. □

Theorem 5.23. *If a monadic semantics has a product-class of models, then it has independent atoms. This implication is strict.*

Proof. Assume a monadic semantics $(F, \mathcal{M})$ has a product-class of models. For any model $M \in \mathcal{M}$ and index $i \in \mathbb{N}$ let $\mathcal{M}' = \mathcal{M}|_{\{p_j^M\}_{j\neq i}}$. It is then clear that, for any restriction set of atoms $\{p_{j_k}\}_{k\in K} \not\ni p_i$ we have that $\mathcal{M}' \subseteq \mathcal{M}|_{\{p_{j_k}^M\}_{k\in K}}$ and since $\mathcal{M}$ is a product class we can ensure that

$$\langle\!\langle p_i \rangle\!\rangle^{\mathcal{M}'} = \langle\!\langle p_i \rangle\!\rangle^{\mathcal{M}}.$$

This proves that the same holds for every atomic restriction set not including p_i. Since the choice of model and atom was arbitrary we have proven that the semantics has independent atoms.

To prove that this implication is strict we present an example of a semantics with independent atoms that has not got a product-class of models. Let the universe be the singleton $U = \{1\}$. Consider the set of models that interpret every index after some finite point as the empty set!

$$\mathcal{M} = \{ M \mid \text{for some } N \in \mathbb{N} \text{ we have that } M(i) = \emptyset \text{ for all } i > N \}.$$

This is clearly not a product-class, since every index is mapped to both $\emptyset$ and $\{1\}$ but $\mathcal{M}$ does not include for example the model M_1 such that $M_1(i) = \{1\}$ for all $i \in \mathbb{N}$ What is left to show is that a monadic semantics with $\mathcal{M}$ as model class has independent atoms. This is seen by observing that for any index i and any model $M \in \mathcal{M}$ there is also a model $M' \in \mathcal{M}$ such that $\{ M(i), M'(i) \} = \{ \emptyset, \{1\} \} = \mathscr{P}(U)$. This is enough to convince that us that a semantics on the universe $\{1\}$ with $\mathcal{M}$ as class of models has independent atoms. □

We can also see that for the case when a monadic semantics has both a product-class of models and general atoms[2], we can give an extra nice identification of the product-class:

Theorem 5.24. *A monadic semantics $(F, \mathcal{M})$ has a product-class of models and general atoms if and only if $\mathcal{M} = \langle\!\langle \mathcal{L} \rangle\!\rangle^{\mathbb{N}}$.*

Proof. This follows directly from the definitions and Theorem 5.21. □

Note: Having a product-class of models and general atoms does not imply that the class of models $\mathcal{M}$ includes every model in $\mathrm{Mod}(U)$, since we may very well have that $\langle\!\langle \mathcal{L} \rangle\!\rangle \neq \mathscr{P}(U)$.

Example 5.25. All the boolean frames semantics $B(U)$ have both a product-class of models and general atoms, since they are full frame semantics. Also the MT-semantics MT_{PD} has got a product-class of models, since we have

$$\mathrm{Mod}^{PA}(\mathscr{P}(S)) = \{\, \mathscr{P}(T) \mid T \subset S \,\}^{\mathbb{N}}.$$

It does however not have general atoms as already observed. This also shows that just because a product-class of models can be expressed as the full set of functions $\mathbb{N} \to D$ for some D, does not imply that the semantics has got general atoms.

The reason for us to introduce the notion of product-classes is that we then can ensure the following property result regarding substitutions:

Theorem 5.26. *Let $(F, \mathcal{M})$ be a monadic semantics that has a product-class of models and general atoms. Then for any model $M \in \mathcal{M}$ and any substitution σ there is a model $M' \in \mathcal{M}$ such that for all $\psi \in \mathcal{L}$*

$$(\sigma\psi)_F^M = \psi_F^{M'}$$

We call M' the σ-model of M and denote it σM.

Proof. Define M' by setting $p_i^{M'} = (\sigma p_i)^M$ for all i. Then clearly the equalities of the theorem holds. We only need to show that $M' \in \mathcal{M}$, but this follows directly by Theorem 5.24 since $M' \in \langle\!\langle \mathcal{L} \rangle\!\rangle^{\mathbb{N}}$. □

From this we get the desired criteria of substitutionality.

Corollary 5.27. *If an monadic semantics has a product-class of models and general atoms, then it is substitutional. Consequently, the full semantics of a monadic frame always defines a substitutional logic.*

Proof. Assume $\sigma\Gamma \nvDash \sigma\phi$ then there is a model M such that $\bigcap \sigma\Gamma^M \not\subseteq \sigma\phi^M$. Now, since we have a product-class of models and general atoms we can by Theorem 5.26 find a σ-model $\sigma M \in \mathcal{M}$ of M. Then clearly $\bigcap \Gamma^{\sigma M} \not\subseteq \phi^{\sigma M}$ which proves that $\Gamma \nvDash \phi$ and we have proven the contrapositive of substitutionality. □

By this we have a sufficient condition for substitutionality. The restriction to semantics with product-classes of models feels quite technical, even if it is connected to the idea of independence. It is desirable to find logical properties that categorise either, but we will not manage to do so here. It is not possible in general since every direct dual monadic semantics has a singleton class of models which is trivially a product-class. We can still hope to find logical connection to maximal semantics but we will not be able to do so in this thesis. A good candidate logical property would be a type of *traceability* condition– something in the line of:

(Under certain circumstances): if $\Gamma \nvDash$, and $\nvDash \phi$, then $\Gamma \vDash \phi$ if and only if $\mathrm{Sup}(\Gamma) \cap \mathrm{Sup}(\phi) \neq \emptyset$.

This type of condition would be desirable for well behaved proof systems, since it asserts that for non-trivial deductions, there is some shared content between assumptions and conclusions. The exact formulation is a bit tricky however, since we need to account for inconsistent-like logical constants. We have not manage to develop this further, and it might be that such a formulation would need to include further generalisations of the semantic framework as discussed in Chapter 7.

[2] No qualifier for the generality is needed since it also has independent atoms.

5.3.3 Main results and open problems

We can summarise our main results and the open problems connected as follows.

Main results

- A maximal monadic semantics has strongly general atoms if and only if its logic is closed under singular substitution.
- If a monadic semantics has a product-class of models and general atoms, its logic is substitutional

This has the following most important consequences

- A maximal monadic semantics of a compact logic has strongly general atoms if only if the logic is substitutional.
- A maximal monadic semantics of a compact logic has the atomification property if and only if the logic is substitutional.
- If a maximal monadic semantics has a product-class of models, it has general atoms if and only if the logic is substitutional.
- If an MT^{PA}-semantics of a substitutional logic is maximal, it is flat.
- The full frame semantics of a monadic frame is always substitutional.

This goes a long way to categorise substitutionality in monadic semantics. It also answers the main question about team semantics that lead us on this path: If we argue that the maximal semantics is the correct semantic representation in a given frame, then there cannot exist any non-flat, substitutional team logic correctly expressed in standard valuational team semantics. The main issue is not any choice of connectives, but the interpretation of atomic formulas.

Open problems

Some important and still missing results for full categorisation of substitutionality in monadic semantics are the following:

- To disprove (or prove) that every logic closed under singular substitution that has a monadic semantics is substitutional.
- To give a monadic semantic condition identifying when its logic is fully substitutional, possibly restricted to maximal semantics.
- To identify logical conditions identifying when its (maximal) monadic semantics has independent atoms.
- To identify logical conditions identifying when its (maximal) monadic semantics have a product-class of models and general atoms, or alternatively,
- to identify logical conditions identifying when its (maximal) monadic semantics has a product-class of models.

Categorising having a product-class of models in the logic would be sufficient to give categorisation of having a product-class and general atoms.

This is how far we will get for now. In the next chapter we use these results, and let them guide us in defining a new substitutional team logic.

Chapter 6

A new team logic BTB

In this chapter we construct an MT-semantics for a substitutional team logic we will call BTB. This logic generalises propositional dependence logic in the sense that the semantics includes the same operations as the interpretations of connectives in MT_{PD} and we can find a set of axioms recovering propositional dependence logic in BTB.

When looking for a substitutional logic, we choose to look for the full semantics of a monadic frame. This guarantees that the semantics we create is maximal with a product-class of models and general atoms, and hence the substitutionality of the logic by Corollary 5.27. With this in mind we first take a step back and see how functions on a set naturally defines a related function on its powerset. Using this we can from any monadic frame construct a related MT-frame we call its *teamification*.

6.1 Natural teamification

Consider an MT-semantics with universe $U = \mathscr{P}(S)$. Then every sample-team X is a set of elements from the underlying set S. We can therefore for any function f on S naturally define a function on U by collecting the results of applying f to every possible combination of elements from sample-teams in U. When the starting function f is the interpretation of a connective in some monadic frame, we call the resulting function its *natural teamification* when interpreting a related connective in an MT-frame.[1]

Definition 6.1. For any function $f : S^n \to S$ we define the function $(f) : \mathscr{P}(S)^n \to \mathscr{P}(S)$ for every $A_1, \dots A_n \subseteq S$ as

$$(f)^n(A_1, \dots, A_n) = \{\, f^n(a_1, \dots, a_n) \mid a_1 \in A_1, \dots, \text{ and } a_n \in A_n \,\}.$$

If a monadic frame F with universe S gives an interpretation of a connective $*^n$ as $*^n_F$. Then the function $(*_F)^n$ gives an interpretation of an n-ary connective on the universe $\mathscr{P}(S)$. We call this *the natural teamification* of $*^n_F$.

When we intend to use the interpretation of a connective $*^n$ in a monadic frame to define a connective as its natural teamification in an MT-frame, we will preferably choose the natural teamification to be the interpretation of the connective denoted $(*)^n$, or when typographically available $\circledast^n$.

Example 6.2. We can observe that the interpretation of $\otimes$ and $\perp$ in $\mathrm{MT}_{\mathrm{PD}}$ are the natural teamification of $\vee$ and $\perp$ in the boolean frame $\mathfrak{B}_{\mathscr{V}}$. In our notation we would hence prefer to change the language and instead of $\otimes$ and $\perp$ use the connectives $\circledvee$ and $\circledperp$ with this interpretation.

6.1.1 Teamification of a monadic frame

Using the idea of natural teamification we can teamify every connective in a language for a frame, and then get a team semantics on the powerset of the original frame. We call this the *teamification* of the frame.

Definition 6.3. Let F be a monadic frame with universe U for the language $\mathcal{L}$. Then its *teamification*, denoted (F) is an MT-frame on the universe $\mathscr{P}(U)$ for the language $\mathcal{L}^\circ$ with the language and frame defined by setting

$$\mathrm{Sing}(\mathcal{L}^\circ) = \{\, \circledast^n \;\big|\; *^n \in \mathrm{Sing}(\mathcal{L}) \,\} \quad \text{and} \quad \circledast^n_{(F)} = (*_F)^n \text{ for all } \circledast^n \in \mathrm{Sing}(\mathcal{L}^\circ).$$

[1] Note that this usage of the word *teamification* is incompatible with the usage of the word by Lück [Lüc20]. The usage of the word is still justified, and this motivates the additional qualifier *natural*.

The reason to rename the connectives is that when a frame F interprets all connectives as set functions on the universe U, they can be reinterpreted as the same set functions on $\mathscr{P}(U)$. We can then make an MT-frame on the doubled up language by pooling the original and teamified definitions of connectives without any resulting conflicts in the namespace. This will be utilised in the construction of the logic BTB.

6.2 Semantic definition of BTB

We observe that $\mathrm{MT_{PD}}$ has interpretations of connectives that over all coincide with boolean operators and teamified boolean operators. The only outlier is negation $\neg$. We want to construct a logic that define these operators, so a motivated start is to build from a combined boolean and teamified boolean frame. Maybe surprising is that the logic we then construct is exactly what we need, even for negation. For transparency, I have chosen to call this logic *the boolean teamified boolean logic* BTB. In order to not be in conflict with the notation of the negation $\neg$ in $\mathrm{MT_{PD}}$, to which we return later, we will denote the boolean and teamified boolean negations $-$ and $\ominus$ instead.

Definition 6.4 (BTB)**.** Let the language have the signature $(\bot^0, -^1, \wedge^2, \vee^2, \obot^2, \ominus^2, \owedge^2, \ovee^2)$. The interpretation of the connectives in the frame F_{BTB} on the universe $\mathscr{VT}$ are defined as follows with $F = F_{\mathrm{BTB}}$:

- The *external* connectives : $\bot, -, \wedge, \vee$ are defined by the boolean interpretations:

$$\begin{aligned}
\bot &: \ \bot_F = \emptyset \\
- &: \ -_F A = A^C \\
\wedge &: \ A \wedge_F B = A \cap B \\
\vee &: \ A \vee_F B = A \cup B
\end{aligned}$$

- The *internal* connectives :$\obot, \ominus, \owedge, \ovee$ are defined as the teamified boolean interpretations:

$$\begin{aligned}
\obot &: \ \obot_F = \{\emptyset\} \\
\ominus &: \ \ominus_F A = \{\, a^C \mid a \in A \,\} \\
\owedge &: \ A \owedge_F B = \{\, a \cap b \mid a \in A, b \in B \,\} \\
\ovee &: \ A \ovee_F B = \{\, a \cup b \mid a \in A, b \in B \,\}
\end{aligned}$$

Let MT_{BTB} be the MT-semantics of this frame, i.e. $MT_{\mathrm{BTB}} = (F_{\mathrm{BTB}}, \mathrm{Mod}(\mathscr{VT}))$. We call its logic the *boolean teamified-boolean logic* BTB.

Note that the universe $\mathscr{VT}$ is not used as a set of teams of valuations, but just as a sufficently large powerset and to easier connect this new semantics to MT_{PD} that has this universe for the connection to VT_{PD}. To emphasise the unimportance of the specific set for the semantics itself, we will denote this universe with a generic $U = \mathscr{P}(S)$ in many of the expressions in this chapter.

By choosing the full semantics of a frame we know without extra effort that the logic is well behaved:

Theorem 6.5. *MT_{BTB} has product-class of models and general atoms, and* BTB *is substitutional.*

Proof. MT_{BTB} is a full frame semantics, and thus, this follows from Corollary 5.27. □

We see that these interpretations of connectives include direct counterparts for every connective interpreted in $\mathrm{MT_{PD}}$ apart from negation $\neg$. We do however also get the additional connective $\owedge$ whose interpretation we have not encountered until now. If we remember that traditional team semantics have the flat atom assumption, and that every formula ϕ is interpreted as a non-empty downwards closed set in every model of $\mathrm{MT_{PD}}$, we can see by the following theorem why this interpretation of a connective $\owedge$ has been overlooked.

Theorem 6.6. *If $A, B \subseteq U$ are non-empty downwards closed sets of teams, then using the interpretations of both conjunctions $\wedge, \owedge$ in MT_{BTB} the resulting sets coincide, that is*

$$A \wedge_{F_{\mathrm{BTB}}} B = A \owedge_{F_{\mathrm{BTB}}} B$$

Proof. For any team $X \in U$ we have that

$$X \in A \owedge_{F_{\mathtt{BTB}}} B \quad \text{if and only if there is } Y \in A, Z \in B \text{ such that } X = Y \cap Z.$$

This implies that $X \subseteq Y$ and $X \subseteq Z$. If then A and B are downwards closed they both have X as a member. Evidently then $X \in A \cap B$, and thus

$$A \wedge_{F_{\mathtt{BTB}}} B \supseteq A \owedge_{F_{\mathtt{BTB}}} B.$$

For the other direction we just need to note that $X = X \cap X$ for every team X. □

This means that if this interpretation of $\owedge$ is added to $\mathrm{MT}_{\mathrm{PD}}$ it would just coincide with that of $\wedge$. In other words: both $\wedge_{F_{\mathtt{BTB}}}$ and $\owedge_{F_{\mathtt{BTB}}}$ are dual functions to the interpretation of $\wedge$ in VT_{PD}.

6.2.1 Definable connectives

From the interpretations and definitions of the connectives in $MT_{\mathtt{BTB}}$ we are motivated to define additional operators. Some are the usual suspects such as $\top$ and $\rightarrow$, and some are useful as they are motivated by the semantics and will help us to define $\neg$ so that the interpretation in $MT_{\mathtt{BTB}}$ agrees with that in $\mathrm{MT}_{\mathrm{PD}}$.

Constants

We have chosen to build from a boolean language with the single constant $\bot$ called the *bottom element*. It is then customary to define the top element $\top$ as $-\bot$. When we move to BTB however we have both an internal and external bottom element and negation. This gives rise to additional naturally defined constants. We name these constants after how they are interpreted in our defining semantics $MT_{\mathtt{BTB}}$.

Definition 6.7. For the logic BTB we define the constant connectives $(\top, \circledtop, \mathrm{NE}, \mathrm{NA})$ with naming motivated by the interpretations in the frame $F = F_{\mathtt{BTB}}$ with its universe denoted $U = \mathscr{P}(S)$ as follows:

- $\top = -\bot$. Then $\top_F = U$ and is called *the external top*.
- $\circledtop = \ominus\circledbot$. Then $\circledtop_F = \{S\}$ and is called *the internal top*.
- $\mathrm{NE} = -\circledbot$. Then $\mathrm{NE}_F = \{\emptyset\}^C = \{\, X \mid X \neq \emptyset \,\}$ and is called the *non-emptiness constant*.
- $\mathrm{NA} = -\circledtop$. Then $\mathrm{NA}_F = \{S\}^C = \{\, X \mid X \neq S \,\}$ and is called the *not-all constant*.

These are all the constants interpreted as distinct in $MT_{\mathtt{BTB}}$ definable from $\bot, \circledbot, -, \ominus$ alone, since adding more of the negations does not construct any new sets.

Deductive implication

In classical propositional logic with the signature $(\bot, -, \wedge, \vee)$ we can define implication by setting

$$\phi \rightarrow \psi = -\phi \vee \psi.$$

For this implication the deduction theorem holds. Since the same connectives in $MT_{\mathtt{BTB}}$ have got a standard boolean interpretation, the same definition suffices to get the same result in BTB.

Definition 6.8. For the logic BTB we include *exterior implication* $\rightarrow$ by defining $\phi \rightarrow \psi = -\phi \vee \psi$

Theorem 6.9. *For the logic* BTB *the following deduction theorem holds:*

$$\textit{For all } \Gamma, \phi, \psi \textit{ we have that } \Gamma, \phi \vdash \psi \textit{ if and only if } \Gamma \vdash \phi \rightarrow \psi.$$

Proof. Using the semantics $MT_{\mathtt{BTB}}$ together with the observation in Note 4.7 we see that

$$MT_{\mathtt{BTB}} : \Gamma, \phi \vDash \psi \text{ if and only if for all models } M \text{ we have } \bigcap \Gamma^M \cap \phi^M \subseteq \psi^M$$

Then by standard set computations we get for each model the following chain of equivalences were $A = \bigcap \Gamma^M, B = \phi^M, C = \psi^M$:

$$\begin{array}{rcll} A \cap B & \subseteq & C & \Leftrightarrow \\ B^C \cup (A \cap B) & \subseteq & B^C \cup C & \Leftrightarrow \\ (B^C \cup A) \cap (B^C \cup B) & \subseteq & B^C \cup C & \Leftrightarrow \\ B^C \cup A & \subseteq & B^C \cup C & \Leftrightarrow \\ A & \subseteq & B^C \cup C & \end{array}$$

where the last equivalence summarises the two implications

$$B^C \cup A \subseteq B^C \cup C \;\Rightarrow\; A \subseteq B^C \cup C$$

and

$$A \subseteq B^C \cup C \;\Rightarrow\; B^C \cup A \subseteq B^C \cup (B^C \cup C) = (B^C \cup B^C) \cup C = B^C \cup C.$$

Thus, for every model M of $MT_{\mathtt{BTB}}$ and with F denoting $F_{\mathtt{BTB}}$:

$$\bigcap \Gamma^M_F \cap \phi^M_F \subseteq \psi^M_F \text{ if and only if } \bigcap \Gamma^M_F \subseteq \neg_F \phi^M_F \vee_F \psi^M_F = (\neg\phi \vee \psi)^M_F$$

which holds if and only if $MT_{\mathtt{BTB}} : \Gamma \vDash \phi \to \psi$. This concludes the proof. □

Closures

We have in mind to find propositional dependence logic using BTB, and since in $MT_{\mathtt{PD}}$, every formula is interpreted as a downwards closed set, it will be useful to define a *downward closure* operation $\downarrow$ interpreted as the operation identifying every team in the downwards closure of a set of teams:

$$\downarrow_F A = \{\, X \mid \text{ there exists } Y \in A \text{ such that } X \subseteq Y \,\}.$$

This set however can also be identified as all teams that are the result of an intersection of a member of A and an arbitrary team in the universe U:

$$\{\, X \cap Y \mid X \in A \text{ and } Y \in U \,\}.$$

This can be exactly expressed in $MT_{\mathtt{BTB}}$ by the external top and internal conjunction as $\top_F \owedge_F A$, and hence we use this as our definition of the downwards closure operation.

Definition 6.10. For the logic BTB we define the *downwards closure operation* $\downarrow$ as $\downarrow\phi = \top \owedge \phi$.

Similarly, we can construct the *upwards closure operation* $\uparrow$ so that for the interpretation by $MT_{\mathtt{BTB}}$, $\uparrow_F A$ is the set of all teams that has a subteam in A:

$$\uparrow_F A = \{\, X \mid \text{ there exists } Y \in A \text{ such that } Y \subseteq X \,\} = \{\, X \cup Y \mid X \in A \text{ and } Y \in U \,\}.$$

This can symmetrically to $\downarrow$ be defined in $MT_{\mathtt{BTB}}$ by the external top and internal disjunction as $\top_F \ovee_F A$.

Definition 6.11. For the logic BTB we define the *upwards closure operation* $\uparrow$ as $\uparrow\phi = \top \ovee \phi$.

We are now ready to find our desired negation.

Negation

The interpretation for the negation $\neg$ in MT_{PD} is not the same as either the interpretation of external or internal negation in MT_{BTB}. It is however definable. We start by recalling the goal interpretation.

$$\neg_{F_{\text{PD}}} A = \{ X \mid \text{for all } Y \in A : X \cap Y = \emptyset \} .$$

This is the set of teams that do not share any non-empty subset with any team in A. To define the same interpretation in MT_{BTB} we can work through this wording step by step. To reduce clutter in the expressions we move the subscript F_{BTB} from the connectives in the formula and add a subscript F to the equality sign between the formula and the explicit interpretation.

We start by finding every subteam of teams in A, that is, the downwards closure of A:

$$\downarrow A =_F \{ X \cap Y \mid X \in U, Y \in A \} ,$$

where U denotes the full universe $\mathscr{V}\mathscr{T}$. We now want to identify the non-empty subsets of this set. That is achieved by intersecting with the interpretation of the non-emptiness constant.

$$\text{NE} \wedge \downarrow A =_F \{ X \cap Y \mid X \in U, Y \in A \text{ and } X \neq \emptyset \} .$$

The interpretation of the upwards closure of this set is the set of teams that share non-empty subset with some member of A:

$$\uparrow(\text{NE} \wedge \downarrow A) =_F \{ X \mid \text{ there exists } Y \in A : X \cap Y \neq \emptyset \} .$$

This is the complement of the set we were looking for. We obtain the desired set by exterior negation.

$$-\uparrow(\text{NE} \wedge \downarrow A) =_F \{ X \mid \text{ for all } Y \in A : X \cap Y = \emptyset \} .$$

Thus we have a definition of $\neg$ in BTB. To distinguish it from the other negations, we call this the *flattening* negation.

Definition 6.12. We define *the flattening negation* $\neg$ in BTB as $\neg\phi = -\uparrow(\text{NE} \wedge \downarrow\phi)$

This may not be the cleanest definition possible, but with it we have identified exactly identical interpretations in the frame of MT_{BTB} of all interpretations of connectives in the frame of MT_{PD}. The name *flattening* is given by the following expected observation.

Theorem 6.13. *In the semantics MT_{BTB} every formula of the form $\neg\phi$ is flat. Furthermore, for all formulas ϕ we have that $\neg\neg\phi \vDash \phi$ if and only if ϕ is flat.*

Proof. This is the same result as Theorem 4.25 stated for MT_{BTB} instead of MT_{PD}. □

With some naturally extended notations we conclude:

Theorem 6.14. *For F_{BTB} the frame of MT_{BTB}, F_{PD} the frame of MT_{PD}, and $\mathcal{L}_{\text{PD}}$ the language with signature $(\bot, \neg, \wedge, \vee, \otimes)$ we have for every model $M \in Mod(\mathscr{V}\mathscr{T})$ and every formula $\phi \in \mathcal{L}_{\text{PD}}$:*

$$\phi^M_{F_{\text{PD}}} = \phi[\bot/\oplus, \otimes/\circledvee]^M_{F_{\text{BTB}}}$$

where $\phi[\bot/\oplus, \otimes/\circledvee]$ denotes the result of substituting every $\bot$ with $\oplus$ and every $\otimes$ with $\circledvee$ in ϕ, and $\neg$ is interpreted as defined in BTB.

This means that if we restrict the semantics to the class of models $\text{Mod}^{PA}(\mathscr{V}\mathscr{T})$ we can express MT_{PD} using the boolean teamified boolean frame F_{BTB}.

Theorem 6.15. *For F_{BTB} the frame of MT_{BTB}, $\mathcal{M}^{PA} = Mod^{PA}(\mathscr{V}\mathscr{T})$, and $\mathcal{L}_{\text{PD}}$ the language with signature $(\bot, \neg, \wedge, \vee, \otimes)$ we have for all $\{\Gamma, \phi\} \subseteq \mathcal{L}_{\text{PD}}$:*

$$MT_{\text{PD}} : \Gamma \vDash \phi \ \text{ if and only if } \ (F_{\text{BTB}}, \mathcal{M}^{PA}) : \Gamma \vDash \phi[\bot/\oplus, \otimes/\circledvee]$$

where $\Gamma \vDash \phi[\bot/\oplus, \otimes/\circledvee]$ denotes the result of substituting every $\bot$ with $\oplus$ and every $\otimes$ with $\circledvee$ in $\Gamma \vDash \phi$, and $\neg$ is interpreted as defined in BTB.

We have thus by a relatively simple construction achieved an extended monadic frame housing a semantics conservatively extending propositional dependence logic. This semantics is however defined on the class of models $\text{Mod}^{PA}(\mathscr{V}\mathscr{T})$ and thus cannot be substitutional. We would like to find axioms for a full frame semantics that can single out this logic. But to do so we need a modal operation.

6.2.2 Definable $\Box$ and axiomatic recovering of propositional dependence logic

We have shown that the frame $F_{\texttt{BTB}}$ of $MT_{\texttt{BTB}}$ extends the frame of $MT_{\texttt{PD}}$, so that the same interpretations of connectives can be recovered and the frames agree on the interpretations of every formula in every model. By Theorem 6.13, Theorem 4.22 and Note 4.7 we can identify the models in $\mathrm{Mod}^{PA}(U)$ with regards to the interpretations of formulas in $F = F_{\texttt{BTB}}$ in the following way

$$(\neg\neg p_i)^M_F \subseteq {p_i}^M_F \text{ for all } i \in \mathbb{N} \text{ if and only if } \quad M \in \mathrm{Mod}^{PA}(U)$$

And for BTB we can by the deduction theorem also state

$$(\neg\neg p_i \to p_i)^M_F = U \text{ for all } i \in \mathbb{N} \text{ if and only if } \quad M \in \mathrm{Mod}^{PA}(U)$$

which is equivalent to the following statement

Theorem 6.16. *For every model $M \in Mod(U)$*

$$MT_{\texttt{BTB}} : M, X \models \neg\neg p_i \to p_i \textit{ for all } i \in \mathbb{N} \textit{ and all } X \in U \textit{ if and only if } \quad M \in Mod^{PA}(U).$$

This is however an identification that uses every team in the universe to evaluate the properties of the model. To capture such evaluation in a formula evaluated for a single team we would need an operator that works like a modal operator $\Box$ defined for the universal relation on the universe, so that for any formula ϕ and any sample X in the universe U:

$$M, X \models \Box\phi \quad \text{if and only if} \quad M, Y \models \phi \quad \text{for all} \quad Y \in U.$$

Such an operator would have to have the following interpretation in a monadic frame F:

$$\Box_F A = \begin{cases} U & \text{if } A = U \\ \emptyset & \text{otherwise.} \end{cases}$$

We are therefore inclined to extend the frame and add such operator to BTB. However (as a surprise at least to me) such an operator is actually definable in $MT_{\texttt{BTB}}$ by the connectives already given. To show this, we consider the interpretations in the frame $F = F_{\texttt{BTB}}$ and start by observing that for the closure operations $\downarrow$ and $\uparrow$, the empty set of teams is an isolated fixpoint:

$$\downarrow_F A = \emptyset \quad \text{if and only if } A = \emptyset \quad \text{and} \quad \uparrow_F A = \emptyset \text{ if and only if } A = \emptyset.$$

We also see that every other downwards closure of a set necessarily includes the empty team

$$\{\emptyset\} \in \downarrow_F A \quad \text{if and only if } A \neq \emptyset,$$

and that the upwards closure of a set is the full universe exactly when the set includes the empty team:

$$\uparrow_F A = U \quad \text{if and only if } \{\emptyset\} \in A.$$

Composing these two closure operations, we get the following interpretation:

$$\uparrow\downarrow_F A = \begin{cases} \emptyset & \text{if } A = \emptyset \\ U & \text{otherwise} \end{cases}$$

This results in an operator that is the converse of what we are looking for. We can then get our desired operation by complementing the initial set and the result:

$$-\uparrow\downarrow-_F A = \begin{cases} U & \text{if } A = U \\ \emptyset & \text{otherwise} \end{cases}$$

This gives a definition of the desired $\Box$ operator in BTB and as a bonus, of an existential operator $\Diamond$ as well:

Definition 6.17. We define *the existential and universal operators* $\Diamond$ and $\Box$ in BTB as $\Diamond A = \uparrow\downarrow A$ and $\Box A = -\Diamond -A$.

We then have the following theorem:

Theorem 6.18. *For the semantics $MT_{\texttt{BTB}}$ with universe $U = \mathscr{VT}$ we have for every model M, sample team X and formula ϕ:*

$$M, X \vDash \Box\phi \quad \textit{if and only if} \quad M, Y \vDash \phi \quad \textit{for all} \quad Y \in U.$$

Proof. The proof is exhibited in the discussion leading up to the definition of $\Box$ in BTB. $\Box$

The operator $\Box$ serves as a quantification over every team in the universe. With this operation identified we can define the set of formulas asserting that a model has flat atoms:

Definition 6.19. Let *the flat atom axioms*, denoted *FAA*, be the following set of formulas

$$FAA = \{ \Box(\neg\neg p_i \to p_i) \}_{i \in \mathbb{N}} .$$

We can then state and prove the following lemma and theorem:

Lemma 6.20. *For the semantics $MT_{\texttt{BTB}}$ with universe $U = \mathscr{VT}$ we have for every model M and sample team X:*

$$M, X \vDash FAA \quad \textit{if and only if} \quad M \in Mod^{PA}(U).$$

Proof. By combining Theorem 6.16 and Theorem 6.18, and the definition of *FAA* this follows directly. $\Box$

Theorem 6.21. *For $\mathcal{L}_{\texttt{PD}}$ the language with signature $(\bot, \neg, \wedge, \vee, \otimes)$ we have for all $\{\Gamma, \phi\} \subseteq \mathcal{L}_{\texttt{PD}}$:*

$$MT_{\texttt{PD}} : \Gamma \vDash \phi \quad \textit{if and only if} \quad MT_{\texttt{BTB}} : FAA, \Gamma \vDash \phi[\bot/\operp, \otimes/\ovee]$$

where $\Gamma \vDash \phi[\bot/\operp, \otimes/\ovee]$ denotes the result of substituting every $\bot$ with $\operp$ and every $\otimes$ with $\ovee$ in $\Gamma \vDash \phi$, and $\neg, \to \Box$ are interpreted as defined in BTB.

Proof. This follows directly from the lemma, and Theorem 6.15. $\Box$

We have thus to some extent found propositional dependence logic axiomatisable in the substitutional team logic BTB. Note however that this set of axioms is very much *not* closed under substitution, in the sense that $\sigma FAA \not\subseteq FAA$ for almost all substitutions σ.

Other team logics

We constructed BTB in order to give a substitutional logic housing propositional dependence logic in a axiomatisable way. This logics is however much more expressive and we can observe that the interpretations of connectives in $MT_{\texttt{BTB}}$ also coincide with the dual functions of the additional connectives in the team logic $\mathbf{PD}^{\vee}$ mentioned in Section 3.1.3. This means that the formulas *FAA* suffice to axiomatise even this logic in BTB. I am convinced that most of the logics presented by Yang and Väänänen in [YV17] can be expressed in BTB in a similar way with these *flat atom axioms*. This will however have to be addressed in future work.

6.3 New connection to linear logic

In Section 3.1.2 we mentioned a connection between propositional dependence logic and linear logic observed by Abramsky and Väänänen, by interpreting $\mathscr{VT}$ as a commutative monoid using to $\cup$ as operation [AV09]. However, if we instead in our new found framework, interpret $\mathscr{VT}$ as a commutative monoid over $\cap$ with unit element $\{\emptyset\}$, we find that the tensor $\otimes$ of this phase semantics for linear logic correspond to our interpretation of internal conjunction $\owedge$ in $MT_{\texttt{BTB}}$. With this reading the corresponding linear implication is exactly the same as the implication defined for the team semantics of *inquisitive logic* also in Section 3.1.2. This connection is much more straight forward than the one previously observed, since it identifies a conjunction with a conjunction instead of disjunction. This connection was however not readily observable in valuational team semantics, since the interpretation of $\owedge$ coincides with $\wedge$ for downwards closed teams and the connective was hidden from view. I do believe the connection between linear logic and BTB is much stronger still, and I think it is possible to fully identify all the connectives of linear logic with those of BTB, where the multiplicative connectives of linear logic are paired with the internal connectives of BTB, and the additive with the external. The exponential operator ! of linear logic is also well paired with the $\square$ of BTB. To elaborate on this connection we can consider a reading of linear logic that interprets formulas as consumable resources [GLR95]. With such a reading, the formula $A \otimes B$ is considered a resource that can be split into two resources, one of A, and one of B. For BTB interpreted in $MT_{\mathtt{BTB}}$, an element of $A \owedge B$ is a token that indicates at least one element in A and one of B must exist in the universe. In this way we can see how *having a resource* for linear logic may correspond to *know to exist* for BTB. Unfortunately, without providing a full account of linear logic, it is not possible to give any deeper description of this connection than mere pointers. A full presentation will therefore have to be left for another time.

Chapter 7

Conclusions and further work

7.1 Conclusion

This thesis was motivated by a struggle to understand why every propositional team logic defined based on the team semantic construction introduced by Yang and Väänänen, either fail to be substitutional, or does not need such team construction for its semantics. To answer this question we needed to first take a general position and define what we will mean by *a semantics* and *a logic*. The general notion of semantics we define is based on the little disputed idea of a semantics consisting of a collection of all possible refutations. We also account for a notion of truth-compositionality which indicates that the semantic content of a formula can be fully reduced to its truth or falsity status in the domain of semantic objects. The notion of logic we define is close to a standard notion as it is used in the field of abstract algebraic logic but weakened to allow for non-substitutional logics. We actively chose to include non-substitutional logics since this was the main property under investigation with respect to propositional team logics which we present in a chapter of its own. We observe that this notion of logic is perfectly adequate for the notion of semantics, in that every semantics defines a logic and every logic can be given a semantics. This may seem quite striking, since even though the notion of semantics we describe may be considered fairly unproblematic, the lack of substitutionality for something called *a logic* is strongly opposed [CE19]. In hindsight we may say, that if one wants to enforce substitutionality as a criteria for the notion of logic, our definition of semantics has to be, to some extent, *underdefined*. For the specific case of propositional dependence logic we describe how the lack of substitutionality in the logic enforces undesirable restrictions on its proof systems and how Quadrellaro's algebraisation efforts identifies a class of algebraic structures, of algebras with partially unrestricted regions. This class of algebras, can then also be described as some extent underdefined.

In this thesis however, in order to investigate these notions further, we introduce a new construction we call *monadic semantics*. Monadic semantics are constructed to capture the idea of possible interpretations of basic notions constructing complex terms through well described operations. This class of structures is shown to be general enough to be able to give equivalent semantics to every truth-compositional semantics, in particular we find how to give monadic semantics equivalent to Yang and Väänänen's valuational team semantics by the right choice of semantic universe, and a strict restriction on the interpretation of atomic formulas. The class of monadic semantics is however not only broad and general, it proves structured enough to exhibit significant results regarding the connection between properties of a logic, and its possible monadic semantics. In particular we show that logics closed under singular substitution only can have maximal monadic semantics for which the propositional variables are the most general formulas with least fixed interpretation in the semantics. Furthermore, if we instead assume any monadic semantics where the atomic formulas are general and the class of models can be expressed as the set of elements in a product of subsets, we are guaranteed that the resulting logic is fully substitutional. The full semantic categorisation is not achieved, but these results are sufficient if we are concerned with compact logics. Furthermore, this proves that the problem of non-substitutionality of Yang and Väänänens team logics is directly linked to their semantic treatment of atomic formulas, which gives rise to a corresponding restriction of interpretations in its monadic counterpart.

With the problem properly identified we have also revealed a way forward for substitutional team logics. We introduce the idea of natural teamification, which allows us to use the description of any monadic semantics in order to define a teamified version, semantics to a different logic. By merging a standard and teamified version of a monadic semantics for classical propositional logic we contruct the *boolean teamified boolean logic* $\mathtt{BTB}$. This logic turns out to be surprisingly expressive and we manage to find a set of formulas that can be assumed to recover Yang and Väänänen's propositional dependence logic conservatively, up to typography, in $\mathtt{BTB}$. $\mathtt{BTB}$ can also be argued to have strikingly strong conections to linear logic, an important logic for computer theory and proof theory alike. For me this strongly indicates that the logic $\mathtt{BTB}$ should be of interest to investigate further. And, even if not achieved in this thesis, well-formulated proof theories and algebraisations should not be hard to find.

7.2 Further work

The study of team logics is a fertile field still under much development. In the writing of this thesis I sought out to understand the problem of substitutionality in team logics, and if possible give a substitutional team logic housing propositional dependence logic. We have manage to do both of these things to some degree. In the process we have given a machinery to describe and compare a big range of possible semantic frameworks, and specialised tools for truth-compositional semantics. This research is by no means exhausted by this thesis. The most glaring holes are the open problems posed in Section 5.3.3, and the connections to linear logic for BTB still awaiting description. Beyond this there are many ways to extend this work. I would like to use the bulk of this last chapter to outline some further work that I see can be done developing these results and methods. The specific presentation and construction we used in this thesis was intentionally geared to a discussion about substitutionality and team semantics. Different directions of either specification or further generalisations are naturally possible and may be useful for related considerations. We list a few with some of the discussions they may give rise to.

7.2.1 Possible algebraisation and proof theory for BTB

The development of proof theory and algebraisation of the logic BTB was not addressed in this thesis. Since the logic is substitutional, and by the simple construction from boolean frames, I suspect that such results are not too far removed. A lot of inspiration can probably be taken from the proof systems for propositional dependence logic by Yang, Väänänen, and Quadrellaro [YV16; YV17; Qua21], but with the reminder that in BTB we need to split the notion of conjunction into two separate connectives. Furthermore, if the connection to linear logic proclaimed in Section 6.3 is as strong as I believe, there is a great amount of work that can be utilised from that field as well [GLR95].

7.2.2 A dependence logic in BTB with substitution closed axioms

A still lurking problem, from a proof theoretic viewpoint, with the axioms FAA for propositional dependence logic in BTB, is that they are not closed under substitution. This hinders any chance for a proof system that admits fully schematic rules and axioms, but this must naturally be the case since they are to define a non-substitutional logic. We are thus inclined to look for other axioms defining logic that captures *as much as possible* of propositional dependence logic, but without sacrificing substitutionality. This turns out to be quite possible. If we remember that dependence logic is described as the logic for which every formula defines a downwards closed set, we are naturally led to look at axioms of the type $\Box(\downarrow\phi \to \phi)$ asserting that the interpretation of a formula already includes its downwards closure. When this axiom scheme is assumed in BTB we expect its logic to agree with propositional dependence logic for expressions that are true for every of its substitution instances. The formal description and proof of these statements were, however, not able to be finalised within the bounds of this thesis.

7.2.3 Multi-frame monadic semantics

We have only defined monadic semantics for individual frames. It is of course possible construct a consensus semantics by pooling many monadic frames. Different ways of doing so will lend themselves to different types of analysis.

On different universes

We mentioned in Section 4.1 when presenting monadic semantics that the formulation of monadic frames are strongly influenced by that of Kripke frames. Kripke semantics are generally defined by pooling multiple frames with essentially the same interpretation of the connectives but on different universes. We can do a similar construction for monadic semantics in general. In modal logic with Kripke semantics it becomes important to identify certain classes of frames to identify types of logics [Zac19]. It seems like a natural step to consider logics for classes of monadic frames in the same fashion as in Kripke semantics. We could then, for example, collect all boolean monadic frames and talk about *the boolean monadic semantics* for classical propositional logic. When using Kripke semantics for some monadic logics it is possible to show that every refutable formula is refuted in a model of bounded size in its defining class [Zac19]. Similar results would be interesting for logics with monadic semantics.

On the same universe

The idea of logical constants gives rise to the semantics being restricted to a frame with fixed interpretation of connectives. We could also completely disregard this elevated role and pool multiple frames on the same universe with different interpretations of the connectives. Much of the constructions about interpretation sets, atomic generality and substitution still hold and only get slightly more involved. In the same way as going from valuational semantics to monadic semantics allowed us to talk about dependence and independence of interpretations of atomic formulas, this type of semantics would enable similar dependence notions for the *logical connectives*. This might be a way to formulate a proper *traceability condition* for logics as outlined in the end of Section 5.3.2.

7.2.4 Probabilistic extensions

When Yang and Väänänen first introduce propositional team semantics they contrast their usage of valuation teams to one where members are counted with multiplicities to be able to consider probability weightings of the members [YV16]. This multi-team approach has for example been used to give semantics for *probabilistic team logic* and *quantum team logic*, described in [HPV15] and explored in for example [Fok20]. Such probabilistic considerations would also suit monadic semantics. Since a monadic semantics is constructed with respect to a universe, one can easily achieve a probabilistic generalisation by imposing a probability measure on the universe or, in the case MT-semantics, its underlying set. Such semantics could without hesitation consider uncountable universes with non-discrete measures. Since monadic semantics is ripe for such generalisations, it seems worthwhile to give them a thought.

7.2.5 Higher order and mixed teamification

The lift from monadic semantics to monadic team semantics is made by the choice of the universe to be a powerset. We also automatically get new semantics and logics on the higher order universe by means of *teamification*. This can be repeated to produce *second order* team semantics and *double teamification*. There is however no limit for this process. We can thus construct higher order team semantics to arbitrary degree, and someone with a Cantorian mind might even want to expand the process to the infinite. I do not see what interesting questions such a process would pose or answer, but I find the idea of *ordinal rank team logics* intriguing.

In this *hierarchy of teamification* each semantics would be defined for a universe of elements, all of equal set theoretic rank. We may want to think of structures with elements of mixed rank, that can then also be members of each other. We are then moving to a setting that can be described by a universe consisting of arbitrary sets, ordered by the two relations $\in$ and $\subseteq$. The structure of this universe is thus ordered by a reflexive and transitive relation $\subseteq$, and a relation $\in$ that much more depends on the specific choice of sets. This resembles the Kripke semantic structures used for *intuitonistic modal logic* discussed in for example [Wol22].

7.2.6 Multimodal $*$-adic semantics and first-order logic

We have described monadic team semantics where the semantic object is a model together with a set. If we instead of sets treated ordered sequences, either by extension or by set theoretic encoding and the ideas in the previous Section, we can get a $*$*-adic semantics* in that it is a semantics for predication of arbitrary arity. By then constructing semantics with modalities as discussed in Section 6.2.2, we get semantics for a type of logics somewhere in the borderland of multimodal logics and first-order logics. If such connections are possible this might give some insights into first-order logics in general.

7.2.7 Manyvalued semantics

We have in standard logical fashion focused on binary semantics of truth. We could generalise to a semantics notion allowing multiple semantic meanings. To formally define this we can draw inspiration from constructions developed by Kripke for *formal theories of truth* [Kri76] and define *semantic valuation scheme* as a generalisation of truth-semantic frameworks.

Definition 7.1. a *semantic valuation function* is a function $f : \mathscr{O} \to \mathtt{M}$ from a semantic domain $\mathscr{O}$ to a set of semantic meanings $\mathtt{M}$.

Definition 7.2. Let $\mathcal{L}$ be a language and $\mathscr{O}$ a semantic domain. A *semantic valuation scheme* for $\mathcal{L}$ on $\mathscr{O}$ is a function $\mathcal{F} : \mathscr{O} \times \mathcal{L} \to \mathtt{M}$ to a set of semantic meanings $\mathtt{M}$.

That is, the semantic valuation scheme maps semantic objects and formulas to semantic meanings not nessecarily restricted to the binary choice of *true* and *false*. The notion of *truth-compositionality* then naturally generalises to a notion of $\mathtt{M}$-*compositionality* instead.

Using these definition, we can in a similar way adapt the whole notion of monadic semantics and the analytic machinery of interpretation sets. If we combine this with the multi-frame generalisations described in Section 7.2.3 we arrive at a very general discussion of semantics for recursively generated languages. This might be too general for most logical needs, but may be useful in a broader linguistic discussion.

7.2.8 Non-consensus semantics

We have outlined a general notion of semantic truth as the consensus value for a set of semantic objects. This is a standard approach of defining semantics from classes of logics. We could of course do otherwise. We can directly consider semantics where a formula is true based on the *existence* of an affirming semantic object. This can directly be seen as identical to a semantics of *refutation* with a formula refuted if they are so by any semantic object. It can also be seen as semantics for *provable truth* where a theorem is provable if there exists a proof. I do not know how much of the constructions in this thesis for consensus semantics that translate in a natural way to other notions of semantics, but it would be interesting to try.

I would not dare to claim that these suggestions represent the most important or most interesting possible continuations of this work, nor that they would actually be fruitful in generating new interesting questions. I would however be happy for any possibility to continue any of these thoughts.

Bibliography

[Abr+16] Samson Abramsky et al. *Dependence Logic: Theory and Applications*. Birkhäuser, 2016.

[AV09] Samson Abramsky and Jouko Väänänen. "From if to bi". In: *Synthese* 167.2 (2009), pp. 207–230.

[Boo47] George Boole. *The mathematical analysis of logic*. Philosophical Library, 1847.

[CE19] Daniel Cohnitz and Luis Estrada-González. *An Introduction to the Philosophy of Logic*. Cambridge University Press, 2019.

[CF08] Nino B Cocchiarella and Max A Freund. *Modal logic: an introduction to its syntax and semantics*. Oxford University Press, 2008.

[Dik22] Josephine Femke Dik. "Proof theory of circular description logics". MA thesis. University of Gothenburg, 2022.

[Fok20] Jelle Tjeerd Fokkens. "On the Reduction of Quantum Teams". MA thesis. University of Gothenburg, 2020.

[Fon16] Josep Maria Font. *Abstract Algebraic Logic: An Introductory Textbook*. Studies in logic and the foundations of mathematics. College Publications, 2016. ISBN: 9781848902077. URL: `https://books.google.se/books?id=-lHPjwEACAAJ`.

[Gir11] Jean-Yves Girard. *The Blind Spot: lectures on logic*. European Mathematical Society, 2011.

[Gir87] Jean-Yves Girard. "Linear logic". In: *Theoretical computer science* 50.1 (1987), pp. 1–101.

[GLR95] JY Girard, Y Lafont, and L Regnier. "Advances in Linear Logic". In: *London Mathematical Society Lecture Notes* 222 (1995).

[Gro07] Jeroen Groenendijk. "Inquisitive semantics: Two possibilities for disjunction". In: *International Tbilisi Symposium on Logic, Language, and Computation*. Springer. 2007, pp. 80–94.

[Hen61] Leon Henkin. "Some Remarks on Infinitely Long Formulas". In: *Infinitistic Methods. Proceedings of the symposium on foundations of mathematics*. Pergamon press, 1961, pp. 167–183.

[Hin98] Jaakko Hintikka. *The principles of mathematics revisited*. Cambridge University Press, 1998.

[Hod07] Wilfrid Hodges. "Logics of imperfect information: why sets of assignments". In: *Interactive Logic* 1 (2007), pp. 117–134.

[Hod98] Wilfrid Hodges. "Compositionality is not the problem". In: *Logic and logical philosophy* 6 (1998), pp. 7–33.

[HPV15] Tapani Hyttinen, Gianluca Paolini, and Jouko Väänänen. "Quantum team logic and Bell's inequalities". In: *The Review of Symbolic Logic* 8.4 (2015), pp. 722–742.

[HS15] Lauri Hella and Johanna Stumpf. "The expressive power of modal logic with inclusion atoms". In: *arXiv preprint arXiv:1509.07204* (2015).

[HS89] Jaakko Hintikka and Gabriel Sandu. "Informational Independence as a Semantical Phenomenon". In: *Logic, Methodology and Philosophy of Science VIII*. Ed. by R. H. J. E. Fenstad and I. T. Frolov. Elsevier, 1989, pp. 571–589.

[HS97] Jaakko Hintikka and Gabriel Sandu. "Game-theoretical semantics". In: *Handbook of logic and language*. Elsevier, 1997, pp. 361–410.

[Kri76] Saul Kripke. "Outline of a theory of truth". In: *The journal of philosophy* 72.19 (1976), pp. 690–716.

[Lüc20] Martin Lück. "Team logic: axioms, expressiveness, complexity". PhD thesis. Hannover: Institutionelles Repositorium der Leibniz Universität Hannover, 2020.

[PW10] Peter Pagin and Dag Westerståhl. “Compositionality I: Definitions and variants”. In: *Philosophy Compass* 5.3 (2010), pp. 250–264.

[Qua21] Davide Emilio Quadrellaro. “On Intermediate Inquisitive and Dependence Logics: An Algebraic Study”. In: *arXiv preprint arXiv:2104.00981* (2021).

[Rus19] Bertrand Russell. *Introduction to Mathematical Philosophy*. London: Allen and Unwin, New York: The Macmillan Company, 1919.

[SP16] Vladimír Svoboda and Jaroslav Peregrin. “Logically incorrect arguments”. In: *Argumentation* 30.3 (2016), pp. 263–287.

[Wol22] Johanna Wolff. “What’s the name of the game?: Dialogue Game Semantics for Intuitionistic Modal Logic with Strict Implication”. MA thesis. University of Gothenburg, 2022.

[Yan14] Fan Yang. “On extensions and variants of dependence logic: A study of intuitionistic connectives in the team semantics setting”. PhD thesis. 2014.

[Yan22] Fan Yang. *Generalized propositional team semantics*. Presentation at online conference: *Logic4Peace: fundraising online Logic event for Peace*. 2022.

[YV16] Fan Yang and Jouko Väänänen. “Propositional logics of dependence”. In: *Annals of Pure and Applied Logic* 167.7 (2016), pp. 557–589.

[YV17] Fan Yang and Jouko Väänänen. “Propositional team logics”. In: *Annals of Pure and Applied Logic* 168.7 (2017), pp. 1406–1441.

[Zac19] Richard Zach. *Boxes and Diamonds: An Open Introduction to Modal Logic*. 2019.