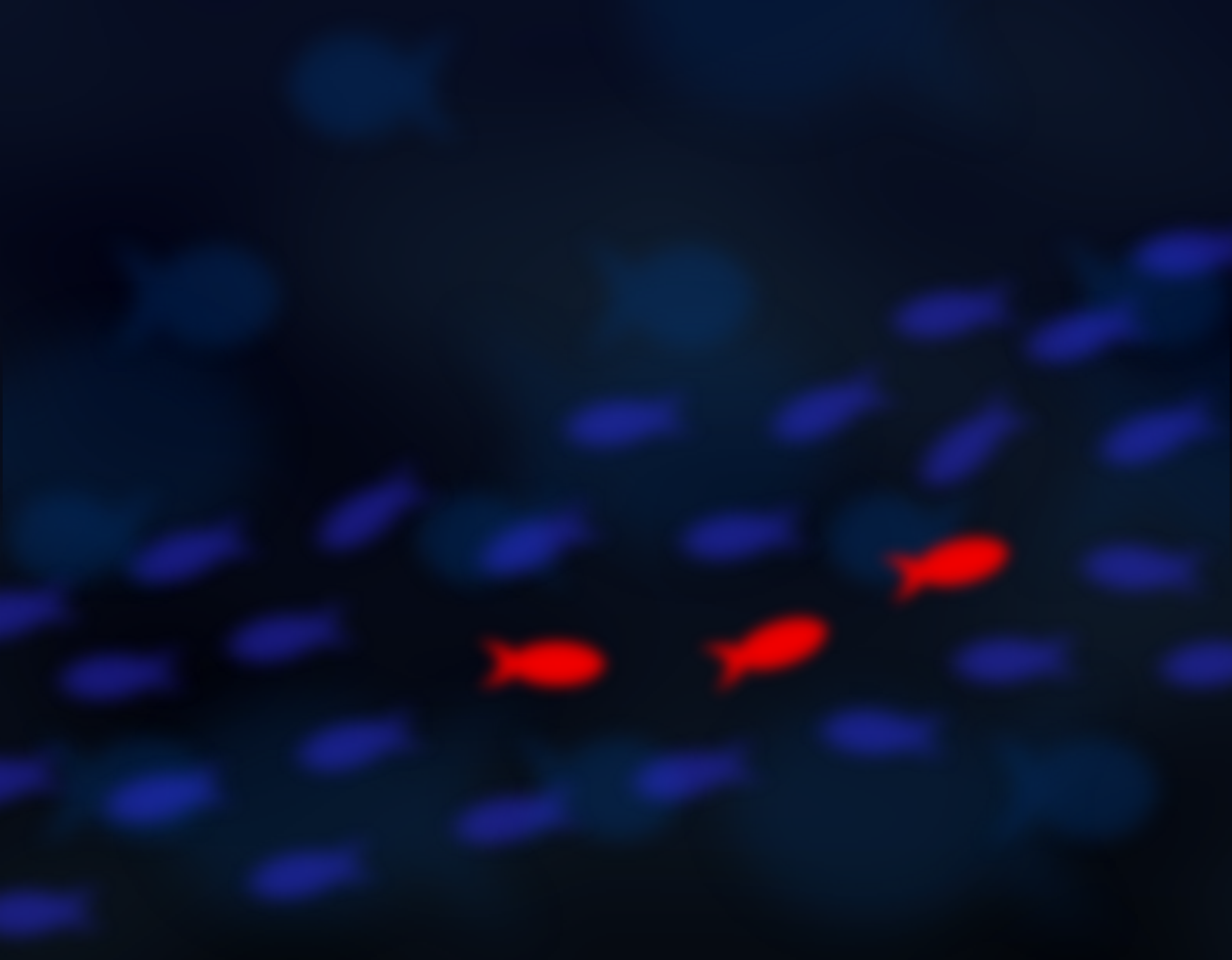
Team Logic

Axioms
Expressiveness
Complexity



Martin Lück

Team Logic
Axioms, Expressiveness, Complexity

Von der Fakultät für Elektrotechnik und Informatik
der Gottfried Wilhelm Leibniz Universität Hannover
zur Erlangung des akademischen Grades
Doktor der Naturwissenschaften (Dr. rer. nat.)
genehmigte

Dissertation

von Herrn

M. Sc. Martin Lück

geboren am 15.12.1988 in Güstrow

Hannover

2020

Referent:	Prof. Heribert Vollmer, Leibniz Universität Hannover
Korreferenten:	Prof. Lauri Hella, Universität Tampere (FI) Prof. Juha Kontinen, Universität Helsinki (FI)
Tag der Promotion:	09.01.2020

Acknowledgments

First of all, I wish to sincerely thank my supervisor, Professor Heribert Vollmer. I am indebted to him for granting me both the trust and liberty to pursue my own ideas in my area of research, as well as the time to develop these over the years and finally write this thesis. Also, I thank him for his deep, motivating and inspiring lectures that drew me to the field of theoretical computer science from the beginning of my studies.

Next, I am thankful to my Finnish colleagues from the logic groups of Helsinki and Tampere for the enjoyable work together, for their hospitality and for many fruitful discussions. In particular, I want to thank Juha Kontinen, Miika Hannula, Jonni Virtema, Lauri Hella, and Miikka Vilander. I am especially grateful to Professor Juha Kontinen and Professor Lauri Hella for their advice and for agreeing to review this thesis on our hike at the Dagstuhl seminar on Logics for Dependence and Independence.

Also, I thank my fellow students and now-colleagues from Hannover, Anselm Haak, Fabian Müller and Maurice Chandoo for many nice discussions, as well as Arne Meier, Irina Schindler, Yasir Mahmood, Timon Barlag and Rahel Becker from the Institute for Theoretical Computer Science.

Finally, I want to thank my sister Maren and my girlfriend Lisa for their unbreakable support and their patience in often challenging times.

Zusammenfassung

Im letzten Jahrzehnt wurde die Teamsemantik kontinuierlich zu einem vielseitigen und mächtigen Rahmen entwickelt, um Begriffe wie Abhängigkeit und Unabhängigkeit mit den Mitteln formaler Logik beschreiben zu können. Bisher lag der Fokus dabei auf der Vielzahl nichtklassischer Atome, die Beziehungen zwischen einzelnen semantischen Einheiten beschreiben, wozu neben dem bekannten Abhängigkeitsatom auch Atome der Unabhängigkeit, des Ein- oder Ausschlusses, der Nichtleerheit sowie verschiedene Zählatome gehören.

Das Ziel dieser Arbeit ist hingegen, die Natur der Teamsemantik auch und gerade ohne solche Atome zu untersuchen. Im ersten Teil beschäftigen wir uns dafür mit grundlegenden Fragen, wie etwa: Wann genau ist eine Logik eine teamsemantische Erweiterung einer klassischen Logik? Welche Freiheitsgrade gibt es beim Definieren einer solchen Erweiterung? Hierfür analysieren wir existierende Teamlogiken und zeigen mögliche Ansätze zur formalen Beantwortung dieser Fragen auf.

Im Rest der Arbeit betrachten wir mit der *Aussagenteamlogik* $\mathsf{PL}(\sim)$, der *Modalteamlogik* $\mathsf{ML}(\sim)$ und der *Prädikatenteamlogik* $\mathsf{FO}(\sim)$ drei konkrete Teamlogiken ohne nichtklassische Atome, die aber unter den Boole'schen Verknüpfungen abgeschlossen sind. Wir untersuchen sie hinsichtlich dreier zentraler Fragestellungen aus dem Bereich der mathematischen Logik, nämlich Ausdrucksstärke, Berechnungskomplexität und Axiomatisierbarkeit. Dabei verwenden wir auch abstrakte Resultate aus dem ersten Teil.

Ein wichtiger Teil der Arbeit ist das Ergebnis, dass $\mathsf{ML}(\sim)$ nichtelementare Komplexität besitzt. Für den Beweis übertragen wir Begriffe aus der Modelltheorie auf die Teamlogik und konstruieren einen Schwerebeweis in mehreren Schritten, wobei wir als Zwischenresultate festhalten, dass Eigenschaften wie Bisimilarität und Kanonizität in $\mathsf{ML}(\sim)$ auf gewisse Weise effizient definierbar sind. Durch Übertragung der Filtrationsmethode auf Teamsemantik finden wir anschließend jedoch auch elementar entscheidbare Fragmente.

Danach wird $\mathsf{FO}(\sim)$ bezüglich Komplexitätsfragen betrachtet, wobei sich das Zwei-Variablen-Fragment wie im klassischen Fall als entscheidbar herausstellt, ebenso wie das *Guarded Fragment* $\mathsf{GF}(\sim)$, das wir analog zum klassischen Fragment GF einführen. Des Weiteren wird im Bereich der Modelltheorie eine Variante des Ultraproduktsatzes von Łoś bewiesen, aus dem beispielsweise auch der Kompaktheitssatz für $\mathsf{FO}(\sim)$ folgt.

Zuletzt entwickeln wir ein modulares Beweissystem für die genannten Logiken $\mathsf{PL}(\sim)$, $\mathsf{ML}(\sim)$ und $\mathsf{FO}(\sim)$. Wir zeigen insbesondere, dass sich die besondere Disjunktion der Teamlogik wie ein modaler Operator axiomatisieren lässt. Für die Vollständigkeit des Systems spielt auch die Widerlegungsvollständigkeit auf der Ebene der Literale eine wichtige Rolle, die für Teamlogiken im Gegensatz zu klassischer Logik nicht trivial gegeben ist. Es werden zwei Möglichkeiten vorgestellt, wie sie für obige Logiken dennoch erreicht werden kann.

Schlagworte: Team-Semantik; Axiome; Ausdrucksstärke; Komplexität

2010 MSC: 03B45; 03B60; 03B70; 03C20; 68Q15; 68Q17

Abstract

In the last decade, team semantics has been continuously developed into a flexible and powerful framework to describe concepts of dependence and independence by means of formal logic. Until now, research mostly focused on the manifold non-classical atoms that express relationships between the semantical units. Besides the prominent dependence atom, among these there also are atoms of independence, inclusion, exclusion, non-emptiness as well as various counting atoms.

The aim of this thesis is instead to study the nature of team semantics on its own and especially without such atoms. In the first part, we consider basic questions, such as: What makes a logic a team-semantical extension of a classical logic? Which degrees of freedom exist when defining such an extension? For this, we analyze existing team logics and point out possible approaches to answer these questions formally.

In the remainder of the thesis, we study *propositional team logic* $\mathsf{PL}(\sim)$, *modal team logic* $\mathsf{ML}(\sim)$ and *first-order team logic* $\mathsf{FO}(\sim)$, which are three concrete team logics without non-classical atoms, yet which are closed under the Boolean operations. We investigate three central questions from the area of mathematical logic, that is, expressive power, computational complexity, and axiomatizability. For this, we also utilize abstract results from the first part.

An important part of the thesis is the result that $\mathsf{ML}(\sim)$ has non-elementary complexity. For the proof, we generalize concepts of model theory to team logic and in several steps construct a hardness proof, with key steps such as showing that bisimilarity and canonicity are, in a certain sense, efficiently definable in $\mathsf{ML}(\sim)$. That being said, by transferring the well-known filtration method to team semantics, we also find elementarily decidable fragments.

Afterwards, the complexity aspects of $\mathsf{FO}(\sim)$ are studied. Its two-variable fragment turns out to be decidable as in the classical case, and likewise for the *Guarded Fragment* $\mathsf{GF}(\sim)$ that we introduce analogously to the classical fragment GF. Furthermore, as a model-theoretic result, we prove a variant of Łoś's ultraproduct theorem, which entails, for example, the compactness theorem for $\mathsf{FO}(\sim)$.

Finally, we develop a modular proof system for the mentioned logics $\mathsf{PL}(\sim)$, $\mathsf{ML}(\sim)$ and $\mathsf{FO}(\sim)$. In particular, we show that the special disjunction of team logic can be axiomatized like a modal operator. In order to prove that this system is complete, the refutation completeness on the level of literals plays a crucial role, which, in contrast to classical logics, does not trivially hold for team logics. We present two methods to achieve this property for the above logics nonetheless.

Keywords: Team semantics; axioms; expressiveness; complexity

2010 MSC: 03B45; 03B60; 03B70; 03C20; 68Q15; 68Q17

Contents

List of Tables and Figures

1 Introduction

Logic as a formalization of reasoning goes back to the ancient Greek philosophers. Their *syllogisms* are a special form of deductive arguments. Consider this classical example:

$$\begin{array}{ll} & \text{Socrates is a human.} \\ & \text{All humans are mortal.} \\ \hline \therefore & \text{Socrates is mortal.} \end{array}$$

If one believes that both premises are true, that is, that Socrates is a human and that all humans are mortal, then one should also believe that the conclusion is true, that is, Socrates is mortal. On the other hand, not all such arrangements of statements conform to valid reasoning:

$$\begin{array}{ll} & \text{Every hippo is a mammal.} \\ & \text{Some mammals lay eggs.} \\ \hline \therefore & \text{Some hippos lay eggs.} \end{array}$$

Although both premises are true, the conclusion is not. We call such reasoning *unsound*. One task of a logician is to model *sound* deductive rules and collect these in a *logical system*, also called *axiom system*, *formal system* or simply *logic*. In modern *first-order predicate logic*, the above examples could be written as *formulas* like

$$\varphi \;=\; \big(\mathsf{Human}(\mathsf{Socrates}) \land \forall x(\mathsf{Human}(x) \to \mathsf{Mortal}(x))\big) \to \mathsf{Mortal}(\mathsf{Socrates})$$

and

$$\begin{aligned}\psi \;=\; &\big(\forall x(\mathsf{Hippo}(x) \to \mathsf{Mammal}(x)) \land \exists y(\mathsf{Mammal}(y) \land \mathsf{LaysEggs}(y))\big) \\ &\qquad \to \exists z(\mathsf{Hippo}(z) \land \mathsf{LaysEggs}(z)).\end{aligned}$$

We agree on φ as being true, and on ψ as false. The concept of *truth* and *falsity* is determined by *semantics*, which roughly speaking is a set of instructions on how to assign a meaning to each formula. In modern logic, valid inferences are often given only implicitly by means of a semantics, which then poses the question of an *axiomatization*, a description of the logic as a set of valid derivation rules, or *axioms*.

In this thesis we consider so-called *team semantics*. In a nutshell, team semantics was introduced in order to equip logics of imperfect information with a *compositional* semantics. Briefly, a semantics is compositional if the meaning of a complex formula is solely determined by the meaning of its constituents. The modern formulation of this principle is often accredited to Frege [34] (cf. [75]) and Boole [11], although it is not undisputed in philosophy and linguistics (cf., e.g., [120]). Compositionality often is a desirable property of formal logic, for example it comes with the principle of *full abstraction*, which states that replacing a constituent by an equivalent one cannot change the meaning of the overall formula. For our example, this means that

Σωκράτης είναι άνθρωπος.
Minden ember halandó.

∴ Sokrates on kuolevainen.

is a valid deduction, as we only replaced statements by equivalent ones.

Today, the fields of mathematics and computer science are unimaginable without formal logic. During the 19th and 20th century, logic was extensively developed and ultimately became the foundation of modern mathematics. In computer science, logic is ubiquitous in fields like digital systems, programming, software verification and artificial intelligence.

As established independently by Church [15] and Turing [134], there is no algorithm for the *Entscheidungsproblem* (decision problem) of first-order logic, the problem of computing whether a given formula is true or false. Driven by this groundbreaking result, logicians have studied numerous logical systems for decades and compared their expressive power (*which properties can be expressed in a logic?*) and algorithmic complexity (*how difficult is its decision problem?*). In this thesis, we will classify several logics with team semantics with respect to these questions, and also present axiomatizations for them.

1.1 Team logic

1.1.1 History of team semantics

The first appearance of team semantics has to be accredited to Wilfrid Hodges [70, 71] and his work on logics of imperfect information.

In logic and linguistics, there have been a number of attempts to formally capture the notions of dependence and independence that occur in natural language, science or statistics. One early approach are *partially ordered quantifiers*, or *branching quantifiers*, by Henkin [63, 92]. Branching quantifiers allow formulas such as for example

$$\exists x \begin{Bmatrix} \forall y\, \exists z \\ \forall w\, \exists u \end{Bmatrix} \forall v\ \varphi(x,y,z,w,u,v)$$

which means "there is an x such that for all y there is a z, and for all w there is a u, such that then for all v the statement φ holds." In particular, the value of z must be chosen depending only on the value of y, and likewise for u and w (this can be formally defined via Skolem functions). Jaakko Hintikka [65] stressed the importance and naturality of branching quantifiers in language, and proposed a formal semantics borrowed from the mathematical area of game theory. Authors such as Barwise [7] noted that branching quantifiers are not *compositional* in Frege's sense (cf. [76]). Indeed, there is no way to interpret the above formula classically by removing one quantifier at a time.

As an alternative to partially ordered quantifiers, Hintikka and Sandu [68] proposed "slashed" quantifiers $(\exists x / \forall y_1, \ldots, \forall y_n)$, meaning "there exists an x independent of the choice of $y_1, \ldots, y_n$." They called their resulting logic *independence-friendly logic* (IF), in which the order of syntactic elements nicely corresponds to that of the players' moves in the game-theoretic semantics. A quantifier $(\exists x / \forall y_1, \ldots, \forall y_n)$ hereby corresponds to a move where the player picks x without knowing the values of the y_i. Accordingly, IF-formulas are interpreted as games of imperfect information. Our example can be expressed in IF as

$$(\exists x)(\forall y)(\exists z)(\forall w)(\exists u / \forall y)(\forall v)\varphi(x, y, z, w, u, v)$$

with z quantified as usual. Note that, as z appears outside the scope of $\forall w$, it is already independent of w it by default.

Barwise, Hintikka and Sandu posed the challenge to find a compositional semantics for logics of this kind. As a response, Hodges [70, 71] proposed his *trump semantics* based on Hintikka's game-theoretic semantics. The gist of it are objects called *trumps*, which are collections of *deals* (read: first-order assignments[1]) that render a formula true in a uniform way. For example, in $\mathbb{N}$, the set $T = \{x \mapsto 2, x \mapsto 3\}$ is a trump of the formula $x > 1$, but $T' = \{x \mapsto 1\}$ is not. Adding imperfect information to the picture, T is not a trump of $(\exists y / \forall x)\,(x = y)$, since there is no way to uniformly pick a y *independently of* x such that $x = y$ throughout T. But T' is a trump of the formula, because the question of dependence does not manifest over single assignments.

Jouko Väänänen [135] introduced *dependence logic* $\mathsf{FO}(\mathsf{dep})$ as a new approach to formalize (in-)dependence in first-order logic. Instead of the cumbersome slashed operators, he added a new atomic formula called *dependence atom*, written $\mathsf{dep}(t_1, \ldots, t_n; t_{n+1})$, which states that the value of the term t_{n+1} depends on the values of the terms $t_1, \ldots, t_n$ (and on nothing else). For $n = 0$, then $\mathsf{dep}(t)$ means that t is constant. For instance, T' satisfies $\mathsf{dep}(x)$, but T does not. In dependence logic, our example formula becomes

$$\exists x\, \forall y\, \exists z\, \forall w\, \exists u\, \forall v\, \big(\varphi(x, y, z, w, u, v) \land \mathsf{dep}(x, w; u)\big).$$

With the atom $\mathsf{dep}(x, w; u)$ we state that u depends only on x and w, which is the same as to say that it is independent of everything else, in particular y and z.

Väänänen appealed to the intuition of assignments as "agents" acting together in a "team", and thereby coined the name of this semantics. Just like IF, dependence logic suffers from flaws inherent to imperfect information, such as the failure of the law of excluded middle. In the team $\{x \mapsto 1, x \mapsto 2\}$, neither $x \neq 1$ nor $x = 1$ are true, because the first assignment violates $x \neq 1$ and the second one violates $x = 1$. The empty team

[1]The syntax of IF also provides constructs such as $\varphi\ (\lor / \forall x)\ \psi$, in which the player chooses a disjunct independently of x, and similarly for conjunction. Accordingly, deals in trump semantics also determine the choice at such connectives. We omit this detail here.

$\emptyset$, on the other hand, satisfies every $\mathsf{FO}(\mathsf{dep})$-formula, including $x \neq 1$ and $x = 1$. This phenomenon is referred to as the *empty team property*. Such peculiarities of the semantics have led to the study of a plethora of variants and extensions of dependence logic.

1.1.2 Logics of dependence and independence

Väänänen's approach has turned out as a powerful and flexible foundation for logics of dependence and independence. In the spirit of the dependence atom, Grädel and Väänänen [49] proposed the *independence atom* $\vec{t} \perp \vec{t}'$ which states that the values of $\vec{t}$ are independent of those of $\vec{t}'$ (in the sense that every value of $\vec{t}$ together with every value of $\vec{t}'$ occurs in some assignment in the team). Galliani [37] introduced *inclusion* atoms $\vec{t} \subseteq \vec{t}'$ (every value of $\vec{t}$ must occur in the team as a value of $\vec{t}'$) and *exclusion* atoms $\vec{t} \mid \vec{t}'$ (the values of $\vec{t}$ and $\vec{t}'$ are disjoint). The logics resulting from adding the respective atoms are called $\mathsf{FO}(\perp)$, $\mathsf{FO}(\subseteq)$, and so on. Too many other non-classical atoms to list them all have been studied, such as *non-dependence* $\neq(\vec{t}; \vec{t}')$ [123], *non-emptiness* NE [142], *totality* ALL [1, 41], and others concerning the cardinality of teams [41, 46].

Team semantics has proliferated into several other logical systems such as propositional, modal, or temporal logic. Here, seminal work is due to Yang [142], Väänänen [136] and Krebs et al. [90]. Numerous papers have appeared concerning the expressive power and computational complexity of all combinations of the above logics with the different non-classical atoms. We will list only some of them (e.g., in Tables 2.1 and 5.4). Common to all of the results is that team logics may have vastly higher complexity than their classical counterparts. For example, $\mathsf{FO}(\mathsf{dep})$ is not axiomatizable and its decision problem is non-arithmetical [135]. In this thesis, we will mostly focus on decidable team logics, but even those have non-elementary complexity (Chapters 4 and 5).

In proposition-based logics, which do not have terms, non-classical atoms work differently. Namely, they are *truth-functional*. For instance, the dependence atom is of the form $\mathsf{dep}(\varphi_1, \ldots, \varphi_n; \varphi_{n+1})$, with the φ_i being *formulas*, not terms, and the atom does only make statements about their truth values; in this case the truth of φ_{n+1} should only depend on the truth of $\varphi_1, \ldots, \varphi_n$.

We give two simple examples for practical applications in cryptography (for more sophisticated ones, see [83]). The atom $\mathsf{dep}(p_1, \ldots, p_n; p_{\mathsf{admin}})$ specifies that whether or not a user has administrator permissions depends on some "password" encoded by propositions $p_1 \cdots p_n$. By this, we can stipulate in a specification, without revealing the values of the p_i, that there is such a function determining p_{admin}. As another example, with the independence atom $c_1 \cdots c_m \perp k_1 \cdots k_n$ we can express that a ciphertext space is independent from the key space, i.e., every key $k_1 \cdots k_n$ can produce every ciphertext $c_1 \cdots c_m$, and one cannot deduce information about one from the other.

1.1.3 Negation

When Hintikka [64, 66] introduced the Boolean negation (which we write as $\sim$) to the IF-setting, he restricted it to be applied to whole sentences. The reason for this lies in the game-theoretical semantics, where truth is defined via winning strategies. The sentence $\sim\varphi$ means "there is no winning strategy for φ", and as such it makes little sense to place $\sim$ in front of the constituents of φ, which due to the lack of compositionality have no meaning on their own. Compare this with $\neg$, which in the game semantics is defined as the players switching roles, and which consequently can be placed in formulas arbitrarily.

Väänänen [135] embraced Boolean negation as a proper connective in his *Team Logic* TL, in our notation written $\mathsf{FO}(\mathsf{dep}, \sim)$. The Boolean negation $\sim$ satisfies the law of excluded middle, and allows to overcome the empty team property such that formulas can be genuinely unsatisfiable. Furthermore, it restores the duality of satisfiability and validity, which are not necessarily dual criteria in team logic, and allows a deduction theorem for team logic (cf. Chapter 6).

Negation was introduced to propositional team logic by Hannula et al. [53] and Yang and Väänänen [144], to modal team logic by Müller [119] and Kontinen et al. [82], and to temporal team logic by Krebs et al. [91]. With arbitrary negation, both the complexity and the expressiveness of the logic increase tremendously. For example, the Boolean disjunction ($\owedge$) becomes easily expressible, whereas in ordinary team logics it is not. Propositional team logic $\mathsf{PL}(\sim)$ and modal team logic $\mathsf{ML}(\sim)$, which are PL and ML in team semantics with $\sim$ added, become expressively complete for their respective semantics [82, 144]. Unlike for modal logic ML, the complexity of $\mathsf{ML}(\sim)$ is non-elementary, and for so-called *synchronous* team-semantical LTL, the decision problem even becomes equivalent to third-order arithmetic [103].

With negation, the truth-functional non-classical atoms are definable in terms of other connectives. In a recent joint work with Miikka Vilander [108], we showed that they are in fact uniformly definable via formulas of polynomial size. Consequently, large parts of this thesis regarding team logic without non-classical atoms easily carry over to the respective extension by such atoms, e.g., in questions of complexity and succinctness.

1.1.4 Applications

We briefly present a few of the practical applications of team semantics in the literature, which are all based on different interpretations of teams. For each area, we also refer the reader to some seminal publications.

Statistics. A team can be seen as a set of (possible or actual) outcomes of a random experiment. From this point of view, Durand et al. [27] defined a *probabilistic team* as a team equipped with a distribution determining the probability of each of its members. A similar notion is that of a *measure team* [72, 73]. Clearly, dependence and independence are of utmost importance in statistics, and with team logic there is a powerful system to express these connections syntactically. In the same vein, Hyttinen et al. [72, 73] introduced a variant of team logic called *quantum team logic* to support a logical analysis of quantum theory, and presented an axiomatization of it.

Database theory. The most intuitive interpretation of a team is perhaps that of a database table consisting of rows, where each row (assignment) maps columns (variables) to values. Often, rows in a database table can appear duplicated. Hence, tables are modeled as *bags*, i.e., multisets, and a corresponding *bag semantics* has been introduced to team logic [26]. Moreover, a database usually contains multiple tables, and team semantics was generalized accordingly to *polyteam semantics* in which a family of teams is considered simultaneously, one representing each table in the database [54]. Dependency notions are a crucial part of database theory, and dependence logic and its variants seem ideal as a logical approach. Indeed, dependence atoms are long-known under the name *functional dependencies*, which have been axiomatized by Armstrong [6] (cf. Abramsky et al. [1]). Likewise, the independence atom expresses a property known as *multivalued dependency* [57] in database theory, and the inclusion atom essentially is a *foreign key constraint* [57].

Epistemic, doxastic and inquisitive interpretations. A team can also be interpreted as a set of possible worlds, or of possible states of reality. This setting is related to the probabilistic one in the sense that it reasons about uncertainty, but also supports, for example, modeling belief and belief updates [40]. The interpretation is that a classical formula α is deemed true only if it holds in every possible reality, that is, in each assignment of the team. For example, epistemically, the formula $p \vee \neg p$ means "I know that p is true or false", which is true. But it is not the same as $p \otimes \neg p$, which means "I know that p or I know that not p", or more succinctly, "I know whether p holds." The latter is false in the team $\{p \mapsto 0, p \mapsto 1\}$, where either value of p is deemed possible. Then again, $p \otimes {\sim} p$ means "Either I know that p or I don't", which is true.

Another related logic is *inquisitive logic*, which allows to formalize questions [16]. There, we read $p \otimes \neg p$ as "is p true?". That p is either true or false is expressed by $p \vee \neg p$, but as this does not settle the question of which one is the case, we have $p \vee \neg p \not\models p \otimes \neg p$. In team semantics, it thus becomes possible to formalize information exchange between agents.

1.2 Contributions

This thesis can be divided into two parts. The first part consists of Chapter 3, in which we study team logic from a novel abstract perspective. The second part consists of Chapters 4 to 6, where we consider propositional, modal and first-order team logic. We investigate central open questions and present results concerning their expressive power, computational complexity and axiomatizability.

In what follows, we elaborate on both parts in more detail.

1.2.1 Abstract team logic

In Chapter 3, we introduce the concept of *abstract connectives* and study team semantics in an algebraic setting. While past research in the area mostly focused on a particular flavor (like propositional, modal, or first-order) of team logic, combined with only one or two non-classical atoms, we instead study such logics in a uniform approach. We classify the common variants of team logic and identify similar features among them, and propose a formal framework that covers the existing formalisms. This approach has several benefits.

Firstly, we are able to prove recurring results abstractly instead of being tied to a specific logic. This does apply not only to well-known properties such as *flatness*, but also to a new result proved in this thesis we call the *collapse theorem*. It states that, under certain assumptions, every formula is equivalent to a Boolean combination of classical formulas, and as such is a cornerstone in the *expressiveness* part of this thesis.

Secondly, classical logic often admits multiple team-logical generalizations, such as *lax* and *strict* semantics [37], or *synchronous* and *asynchronous* temporal semantics [91]. When a logician devises a new team semantics based on an existing logic, he or she might wish that the resulting logic is well-behaved and tractable. From this point of view, logics such as $\mathsf{PL}(\sim)$, $\mathsf{ML}(\sim)$ and $\mathsf{FO}(\sim)$ have nice and in fact nearly identical semantics, so it seems there are conventions on which authors have silently agreed. In this light, our framework provides some orientation on how to "teamify" a given classical connective.

Thirdly, the abstract approach puts us in the position to consider team logic from a more philosophical perspective. We can ask questions such as, what does it mean for a logic to be a team-semantical extension of a given classical formalism? Is a team semantics necessarily based on a classical logic?

Previous research. Algebraization, the formulation of semantics in terms of universal algebra, is a whole area of logic (see, e.g., Andréka et al. [5]), and has been pursued also for independence-friendly logic. Hintikka [67] argued that the propositional part of (extended) IF, that is, its restriction to the operations $\wedge, \vee, \neg$ and $\sim$, gives rise to a *closure algebra*, which are a special case of a Boolean algebra with operators (BAO). Later, Mann [111] proposed an algebraization of full IF. Also, an early paper by Abramsky and Väänänen [2] suggests that Hodges's semantics naturally results as an instance of a more general algebraic system that combines intuitionistic and linear logic.

Although we employ algebraic notations, we do not pursue a full algebraic description of a single logic. Instead, we utilize universal algebra as a framework to define arbitrary connectives in, and to prove results on these abstract grounds that are common to all team logics. For this reason, our efforts have to be understood as orthogonal to existing results in the area of algebraic logic.

Results and organization. After some basic definitions (Section 3.1), in Section 3.2 we define the central concept of *teamification*, which is a relation between a "classical" connective $\triangle$ and its "team-counterpart" ∇. Roughly speaking, it means that the power set operation $\wp$ acts as a *homomorphism* between the algebras spanned by $\triangle$ and ∇. It turns out that this is a natural concept that covers virtually all team-logical connectives, and has a number of interesting implications.

In the area of universal algebra, Boolean algebras with operators (BAOs) are prevalent.

They provide an abstract framework for modal logic and its variants and offer a rich model theory, such as the famous Jónsson-Tarski Representation Theorem [77], which in a sense is the modal extension of Stone's Representation Theorem [138]. In Section 3.3, we lay the foundations for applications to team logic by showing that many team-logical connectives are in fact operators in this sense.

The most prominent operator is the diamond $\Diamond$ of standard modal logic. Interestingly, in team logic, operators also cover pairs like $\wedge/\vee$, $\Diamond/\Box$ and $\exists/\forall$ which classically are duals of each other. In this light, we introduce the notion of *weak duality* between pairs of operators, which is a concept that does not exist in classical semantics. We also provide two characterizations of weak duality.

In Section 3.4, we study an important subclass of operators we call *transversals*. Intuitively, a transversal is an operator with the restriction that the "successor teams" of a team T are completely determined by the successors of the elements of T. Not only is this a well-behaved class of operators with nice properties, in fact this concept is again natural and ubiquitous among existing team-logical connectives.

Afterwards, in Section 3.5, we continue to formalize another recurring pattern, namely the concept of strict and lax operators. We present a general definition that again covers the existing connectives.

Next, in Section 3.6, we show that all connectives of propositional, modal and first-order team logic but $\neg$ and $\sim$ fall into a subclass we call *standard transversals*. We propose this class as a canonical "teamification" of connectives. In combination with our definition of laxness, we prove the main theorem of this chapter, the collapse theorem. As mentioned before, it states that every formula is equivalent to a Boolean combination of flat formulas. This has in fact been shown for propositional [144], modal [82] and first-order logic [38] in lax semantics by means of different methods, but it seems that this connection was never noticed or stated explicitly. We generalize this fact and prove it solely based on the assumption that all involved connectives besides $\neg$ and $\sim$ are lax standard transversals.

We conclude the chapter with an outlook (Section 3.8) on how these definitions can lead to well-behaved team semantics of temporal logic, and with some remarks (Section 3.9) on future research possibilities.

1.2.2 Propositional, modal, and first-order team logic

The second part of this thesis focuses on three concrete team logics, namely *propositional team logic* $\mathsf{PL}(\sim)$, *modal team logic* $\mathsf{ML}(\sim)$, and finally *first-order team logic* $\mathsf{FO}(\sim)$. Moreover, propositional team logic also comes in a quantified variant, $\mathsf{QPL}(\sim)$, which is the team analog to the well-known quantified Boolean formulas.

The above logics do not possess any non-classical atoms (so $\mathsf{FO}(\sim)$ is not the same as Väänänen's $\mathsf{TL} = \mathsf{FO}(\mathsf{dep}, \sim)$), and at first sight seem to be no proper team logics and pointless to study. We give several justifications to consider them nevertheless.

First of all, truth-functional atoms like $\mathsf{dep}(\varphi_1, \ldots, \varphi_n; \varphi_{n+1})$ *can* be expressed in these logics. As indicated earlier, these atoms can be translated to formulas over $\wedge$, $\vee$ and $\sim$, and indeed efficiently so [108], which includes the atoms of dependence, independence, inclusion, exclusion and anonymity. Our translation in [108] crucially relies on negated disjunctions $\vee$, and indeed in the same paper, we showed that the translation is necessarily exponential if $\vee$ may only occur positively.

In first-order logic, we can distinguish between truth-functional atoms like $\mathsf{dep}(\varphi;\psi)$ and proper dependence atoms like $\mathsf{dep}(x;y)$ which range over individuals, and similarly for independence atoms, inclusion atoms and so on. In the inquisitive interpretation, $\mathsf{dep}(x) \wedge R(x)$ corresponds to the question "which x satisfies R?" (with perhaps infinitely many possible answers), whereas $\mathsf{dep}(\varphi)$ means "is φ true?". Essentially, any matter with only finitely many choices can already be expressed in $\mathsf{FO}(\sim)$.

From the proof-theoretic perspective, it is actually *easier* to study, say, $\mathsf{ML}(\sim)$ than all the separate fragments of modal dependence logic, modal inclusion logic etc. independently. This is because standard tools like the deduction theorem become available if the negation is present, but also because connectives such as $\vee$ and $\Diamond$ are easier to axiomatize in the form of their universally quantifying dual connectives.

Finally, our results on these logics also yield new complete problems for rarely studied complexity classes.

Previous research. We provide some background on the logics $\mathsf{PL}(\sim)$, $\mathsf{QPL}(\sim)$, $\mathsf{ML}(\sim)$ and $\mathsf{FO}(\sim)$; a formal introduction can be found in Chapter 2. For $\mathsf{PL}(\sim)$ and $\mathsf{QPL}(\sim)$, the computational complexity of their central decision problems of satisfiability and validity is settled as ATIME-ALT(exp, poly)-complete [53, 56].

For $\mathsf{ML}(\sim)$, while its model checking problem is known to be PSPACE-complete [119], the satisfiability problem was only known to be ATIME-ALT(exp, poly)-hard, as it contains $\mathsf{PL}(\sim)$ as a fragment. The exact complexity of the satisfiability and validity problems of $\mathsf{ML}(\sim)$ was a major open question [28, 58, 82, 119].[1] It is resolved in Chapter 4 as completeness for the non-elementary class TOWER(poly), which corresponds to runtime that is a tower of stacked exponentials of polynomial height.

The logic $\mathsf{FO}(\sim)$ has not yet been addressed much in questions of computational complexity. We show that its complexity coincides with that of FO, i.e., its decision problem is Σ^0_1-complete. Furthermore, we also continue a line of previous research on decidable fragments. In a series of papers, Kontinen et al. [79, 80, 81] showed that the satisfiability problem of the two-variable fragments of $\mathsf{FO}(\mathsf{dep})$, $\mathsf{FO}(\perp)$ and $\mathsf{FO}(\subseteq)$ are all elementarily decidable. However, their method relies on a translation to existential second-order logic, and thus fails if arbitrary negation is allowed. In Chapter 5, we show by a different approach that the two-variable fragment of $\mathsf{FO}(\sim)$ is decidable, albeit non-elementary, and similarly its *guarded fragment* $\mathsf{GF}(\sim)$ which we introduce in this thesis.

Concerning the expressiveness of $\mathsf{PL}(\sim)$, $\mathsf{ML}(\sim)$ and $\mathsf{FO}(\sim)$, the most relevant existing results are the following. Kontinen et al. [82] characterized the $\mathsf{ML}(\sim)$-definable proper-

[1] Hella et al. [58] already observed that a non-elementary bound for $\mathsf{ML}(\sim)$ could be achieved via a notion of "team-bisimulation and Hintikka-types", which presumably means proving a finite model property for some fixed non-elementary size bound and then using a brute force algorithm. The algorithm presented here is not so much different.

ties as those that are invariant under k-*team-bisimulation* for some k. One implication of this is that ML(∼) is equivalent to the Boolean closure of ML. Galliani [38] proved that the expressiveness of FO(∼) coincides with that of FO when restricted to sentences. For his result, he proved that FO(∼)-formulas can be written as Boolean combinations of flat formulas as well; using the fact that ∼ and ¬ are equivalent (over non-empty teams) when applied to sentences, one can transform FO(∼)-sentences into pure FO. For PL(∼), a similar normal form was established by Yang and Väänänen [144], when they proved that PL(∼) is expressively complete.

These normal forms also follow from our collapse theorem in Chapter 3, and they also play a major role in the computational complexity (Chapter 5) and the axiomatizations of the respective logics (Chapter 6).

For certain fragments of the above logics, axiom systems have been proposed. Sano and Virtema [124] and Yang [141] presented complete systems for propositional and modal dependence logic, as well as Yang and Väänänen [143] for propositional dependence logic, and for a fragment of propositional team logic [144]. In the first-order case, it is well-known that Väänänen's dependence logic FO(dep) and team logic FO(dep, ∼) are not axiomatizable [135]. However, partial axiomatizations have been found, such as for all FO-formulas entailed by a set of FO(dep)-formulas [78, 86], or for isolated independence or inclusion atoms [52]. Recently, Kontinen and Yang [89] proposed a novel first-order team logic called FOT, whose expressive power coincides with FO in an analog way as FO(dep) does with existential second-order logic SO(∃), and proved that it is axiomatizable.

Yet, common to all these proof systems is that they do not support the Boolean negation ∼, and the full logics PL(∼), ML(∼) and FO(∼) have previously resisted any attempt of axiomatization.

Contributions. In Chapter 4, we prove that the satisfiability and validity problem of ML(∼) are complete for TOWER(poly), which to the author's best knowledge is the first result of a team logic that is decidable but not elementary. We also show that the fragments $\mathsf{ML}_k(\sim)$ of bounded modal depth k are complete for classes we call ATIME-ALT($\exp_{k+1}$, poly). Key to this approach are *canonical models*, which are a standard tool in modal logics. In Section 4.1, we adapt this notion to team semantics, and prove that such models exist for ML(∼). Afterwards, we proceed with the lower bounds. In Section 4.2 to 4.4, we show that ML(∼) can, in a certain sense, *efficiently enforce* canonical models. In Section 4.5 and 4.6, we encode computations of non-elementary length into such large models. Moreover, we also generalize the result to strict semantics (Section 4.7) and to the common frame classes of modal logic (Section 4.8). Finally, in Section 4.9, we adapt the well-known filtration method to modal team logic and by this obtain a non-trivial fragment of ML(∼) that has only elementary complexity.

In Chapter 5, we focus on FO(∼). By application of the collapse theorem, we obtain results on the complexity of its model checking, validity and satisfiability problem in Section 5.1. Furthermore, we study its two-variable fragment $\mathsf{FO}^2(\sim)$ and introduce a new fragment called *guarded team logic* GF(∼). In analogy to the classical fragments FO^2

and GF of FO, we prove that both $FO^2(\sim)$ and $GF(\sim)$ are decidable. Next, in Section 5.2, we show that well-known standard translation from ML to FO can be generalized to team semantics with minor adaptations. Lastly, in Section 5.3, we study $FO(\sim)$ from the perspective of model theory and prove a variant of Łoś's ultraproduct theorem. Roughly speaking, it states that a structure and its ultrapower satisfy the same first-order formulas. We adapt this to team logic, and for this introduce a suitable notion of ultraproducts of *teams*. As an implication of this, we conclude that the compactness theorem does not hold only for $FO(\sim)$, but also for certain extensions of it by non-classical atoms.

Finally, in Chapter 6, we present a modular proof system for team logic, which we subsequently adapt to $PL(\sim)$, $ML(\sim)$ and $FO(\sim)$. First, we axiomatize the Boolean operators in Section 6.2, and proceed with the other operators in Section 6.3. We conclude with some remarks on the empty team (Section 6.4) and compare our approach with the existing axiomatizations of fragments of team logics and its fragments (Section 6.5).

1.3 Further notes

Prerequisites. The reader is assumed to be familiar with elementary logic. In particular, basic knowledge on propositional logic, modal logic and first-order logic in the usual Kripke and Tarski semantics is helpful throughout the thesis. Moreover, basic complexity theory is required for Chapters 4 and 5. We refer the reader to standard textbooks on logic [10, 31] and complexity theory [130].

No previous knowledge on team logic is required, and we formally introduce it in Chapter 2. We also define novel complexity classes considered in this thesis, and agree on some mathematical standard notation.

Publications. Chapter 3 has not been published previously. Chapter 4 up to Section 4.6 is based on a conference publication [101], with Sections 4.7 and 4.8 added in an extended journal article [102]. The final Section 4.9 appeared as a single paper [105].

Most of Chapter 5 is based on a single conference publication [104], but the part on model theory (Section 5.3) is new, as well as our proof for the complexity of $FO(\sim)$. Moreover, the team-semantical guarded fragment $GF(\sim)$ is introduced in this thesis for the first time.

Finally, Chapter 6 is based on a journal article [100] which again extends a preliminary conference publication [99]. As part of this thesis, it was completely rewritten, and many proofs have been significantly shortened and simplified. This applies both to the derivations in the appendix and to proofs on the meta-level, for instance the necessary compactness theorem is a corollary of the ultraproduct theorem (Section 5.3), whereas it required a different proof in the original paper [100]. We also now take the connectives $\wedge$, $\vee$, $\Diamond$ and $\exists$ as primitives instead of their duals $\multimapdotboth$, $\multimap$, $\triangle$ and $!$, and hence simplify the notation and follow conventions in the literature more consistently.

2 Preliminaries

In this chapter, we agree on some standard notation, and in Section 2.1 lay the ground for the complexity theoretical aspects of this thesis. Afterwards, we provide a formal background on team logic in Section 2.2.

The set of non-negative integers $\{0, 1, 2, \ldots\}$ is denoted by ω or $\mathbb{N}$. We write $[n]$ for the range $\{1, \ldots, n\}$, and $|X|$ for the cardinality of the set X. A set with one element is called *singleton*. The set of all finite sequences of elements of X, or *words*, is X^*. If $x = x_1 \cdots x_n$ is a word, then $|x| := n$ also denotes the length of x. If $Y, Y' \subseteq X^*$, then $Y \circ Y' := \{ww' \mid w \in Y, w' \in Y'\}$.

The power set of X is written $\wp(X)$ or $\wp X$. The set $\wp^+(X) := \wp(X) \setminus \{\emptyset\}$ is the set of all non-empty subsets of X, $\wp^1(X) := \{\{x\} \mid x \in X\}$ is the set of all singleton subsets of X, and $\wp^{<\omega}(X)$ is the set of all finite subsets of X.

The domain of a function f is denoted by $\operatorname{dom} f$. If $X \subseteq \operatorname{dom} f$, then $f \restriction X$ is the restriction of f to the domain X. The *inverse* of a function $f\colon X \to Y$ is the function $f^{-1}\colon Y \to \wp X$ with $f^{-1}(y) = \{x \in X \mid f(x) = y\}$.

Let $(X_i)_{i \in I}$ be a family of sets. A *choice function* for $(X_i)_{i \in I}$ is a function $f\colon I \to \bigcup_{i \in I} X_i$ such that $f(i) \in X_i$ for all $i \in I$. The set of all choice functions for $(X_i)_{i \in I}$ is denoted by $\prod_{i \in I} X_i$, that is, $\prod_{i \in I} X_i := \{f\colon I \to \bigcup_{i \in I} X_i \mid \forall i \in I : f(i) \in X_i\}$. For example, if $I = [n]$, then a choice function is just an n-tuple in the Cartesian product $X_1 \times \cdots \times X_n$. Throughout this thesis, we assume the axiom of choice, which is equivalent to the statement that $\prod_{i \in I} X_i$ is non-empty iff X_i is non-empty for all $i \in I$.

2.1 Complexity theory

We assume that the reader is familiar with basic complexity theoretic concepts such as Turing machines, (un-)decidability, $\mathcal{O}$-notation, time and space complexity, non-determinism, reductions, hardness and completeness. For an introduction to these topics, the reader is referred to standard literature [130].

Recall that a *reduction* from a problem $A \subseteq \Sigma^*$ to a problem $B \subseteq \Delta^*$ is a computable function $f\colon \Sigma^* \to \Delta^*$ such that $x \in A \Leftrightarrow f(x) \in B$ for all $x \in \Sigma^*$. We write $A \leqslant B$ if there is some reduction from A to B. If the function f is additionally computable in time $n^{\mathcal{O}(1)}$, then it is called *polynomial time reduction*, and we write $A \leqslant^{\mathrm{P}}_{\mathrm{m}} B$ if A is polynomial time reducible to B. Similarly, if f is computable in space $\mathcal{O}(\log n)$, then it is called *logspace reduction* and we write $A \leqslant^{\log}_{\mathrm{m}} B$. A problem B is *hard* for a complexity class $\mathcal{C}$ with respect to a type of reduction $\prec$ if $A \in \mathcal{C}$ implies $A \prec B$, and *complete* for $\mathcal{C}$ if B is hard for $\mathcal{C}$ and $B \in \mathcal{C}$. If nothing else is stated, hardness and completeness in this thesis are w. r. t. $\leqslant^{\log}_{\mathrm{m}}$.

Turing machines. We briefly remind the reader of some basic definitions. A (single-tape) Turing machine is a tuple $M = (Q, \Sigma, \Gamma, \delta, q_0, \flat)$ with the usual components, that is, a state set Q, an input alphabet Σ, a tape alphabet $\Gamma \supseteq \Sigma$, a transition relation $\delta \subseteq Q \times \Gamma \times Q \times \Gamma \times \{L, R, N\}$, an initial state $q_0 \in Q$ and a blank symbol $\flat \in \Gamma \setminus \Sigma$. A *configuration* of M is a string $C \in \Gamma^* \circ (Q \times \Gamma) \circ \Gamma^*$, with the successor relation $C \vdash_M C'$ defined as usual. M is called *deterministic* if every configuration has at most one successor configuration.

For the special type of *alternating Turing machines*, we follow Chandra et al. [13] (see also Sipser [130, Ch. 10.3]).

A (single-tape) alternating Turing machine is a tuple $M = (Q, \Sigma, \Gamma, \delta, q_0, \flat, g)$ where $(Q, \Sigma, \Gamma, \delta, q_0, \flat)$ is a Turing machine and $g \colon Q \to \{\exists, \forall, \mathrm{acc}, \mathrm{rej}\}$. Suppose that configurations of M do not form a loop w. r. t. $\vdash_M$.

We define the sets $Q_\exists, Q_\forall, Q_{\mathrm{acc}}, Q_{\mathrm{rej}}$ by $Q_i = \{q \in Q \mid g(q) = i\}$ and call these the *existential*, *universal*, *accepting* and *rejecting* states of M, respectively. Let $C = w(q, a)v$ be a configuration, where $w, v \in \Gamma^*$, $a \in \Gamma$ and $q \in Q$. Then C is *accepting* if $q \notin Q_{\mathrm{rej}}$ and additionally one of the following holds: Either $q \in Q_{\mathrm{acc}}$, or $q \in Q_\exists$ and there is an accepting successor configuration C' of C, or $q \in Q_\forall$ and all successor configurations C' of C are accepting. An input $x = x_1 \cdots x_n \in \Sigma^*$ is *accepted by* M if the *initial configuration* $(q_0, x_1)x_2 \cdots x_n$ is accepting. A *non-deterministic* Turing machine is an alternating Turing machine where $g(q) \neq \forall$ for all $q \in Q$.

A Turing machine M runs in *time* $t \colon \mathbb{N} \to \mathbb{N}$ if for every n and every input x with $|x| = n$ every computation path from the initial configuration on x reaches an accepting or rejecting state after at most $t(n)$ steps. M runs in *space* $s(n)$ if every reachable configuration has length at most $s(n)$, and with $a(n)$ *alternations* if on every computation path there are at most $a(n)$ transitions from an existential to a universal state or vice versa.

For a function $f \colon \mathbb{N} \to \mathbb{N}$, the complexity classes $\textsc{Time}(f)$ and $\textsc{Space}(f)$ contain all decision problems that are decidable by some deterministic Turing machine in time resp. space $\mathcal{O}(f)$. The classes $\textsc{NTime}(f)$ and $\textsc{NSpace}(f)$ resp. $\textsc{ATime}(f)$ and $\textsc{ASpace}(f)$ are defined analogously via non-deterministic resp. alternating machines. Finally, $\textsc{ATime-Alt}(f, g)$ contains the problems decidable by an alternating machine that simultaneously runs in time $\mathcal{O}(f(n))$ and with $g(n)$ alternations.

If $\mathcal{F}$ is a set of functions (such as $n^{\mathcal{O}(1)} = \{n, n^2, n^3, \ldots\}$), then $\textsc{Time}(\mathcal{F}) := \bigcup_{f \in \mathcal{F}} \textsc{Time}(f)$, and similarly for the other classes. Some prominent complexity classes are the following:

$$
\begin{aligned}
\mathrm{L} &:= \textsc{Space}(\log n) & \textsc{PSpace} &:= \textsc{Space}\left(n^{\mathcal{O}(1)}\right) \\
\mathrm{P} &:= \textsc{Time}\left(n^{\mathcal{O}(1)}\right) & \textsc{ExpTime} &:= \textsc{Time}\left(2^{n^{\mathcal{O}(1)}}\right) \\
\mathrm{NP} &:= \textsc{NTime}\left(n^{\mathcal{O}(1)}\right) & \textsc{NExpTime} &:= \textsc{NTime}\left(2^{n^{\mathcal{O}(1)}}\right) \\
\mathrm{AP} &:= \textsc{ATime}\left(n^{\mathcal{O}(1)}\right) & \textsc{2ExpTime} &:= \textsc{Time}\left(2^{2^{n^{\mathcal{O}(1)}}}\right)
\end{aligned}
$$

By Chandra et al. [13], $\mathrm{AP} = \textsc{PSpace}$. In particular, the above classes are ordered by inclusion as follows:

$$\mathrm{L} \subseteq \mathrm{P} \subseteq \mathrm{NP} \subseteq \mathrm{AP} = \mathrm{PSPACE} \subseteq \mathrm{EXPTIME} \subseteq \mathrm{NEXPTIME} \subseteq 2\mathrm{EXPTIME}.$$

(Non-)Elementary complexity. For this thesis, we require several other complexity classes. Let $\exp_0(n) := n$ and $\exp_{k+1}(n) := 2^{\exp_k(n)}$. A function $f\colon \mathbb{N} \to \mathbb{N}$ (resp. a problem A) is *elementary* if it is computable (decidable) in time $\mathcal{O}(\exp_k(n))$ for some fixed $k \in \mathbb{N}$, otherwise it is *non-elementary*.

Definition 2.1. $\mathrm{ELEMENTARY} := \mathrm{TIME}(\exp_{\mathcal{O}(1)}(n))$.

Hence the elementary problems are precisely those decidable by a deterministic Turing machine in time $f(n)$ for some elementary function $f\colon \mathbb{N} \to \mathbb{N}$.

Definition 2.2. Let $k \geqslant 0$. Then

$$\mathrm{ATIME\text{-}ALT}(\exp_k, \mathrm{poly}) := \mathrm{ATIME\text{-}ALT}\left(\exp_k\left(n^{\mathcal{O}(1)}\right), \mathcal{O}\left(n^{\mathcal{O}(1)}\right)\right).$$

In other words, this class contains the problems decidable by an alternating Turing machine with at most $p(n)$ alternations and runtime at most $\exp_k(p(n))$, for some polynomial p. Note that for $k = 0$ this class coincides with $\mathrm{AP} = \mathrm{PSPACE}$. For $k = 1$, k is often omitted, so that this class is also known as $\mathrm{ATIME\text{-}ALT}(\exp, \mathrm{poly})$ [56].

Some decision problems we consider are decidable, but not elementary. We can locate them in the following class, proposed by Schmitz [125]:

Definition 2.3.

$$\mathrm{TOWER} := \bigcup_{\substack{f\colon \mathbb{N}\to\mathbb{N} \\ f \text{ elementary}}} \mathrm{TIME}(\exp_{f(n)}(1)).$$

A suitable type of reduction for this class is the *elementary reduction*. An elementary reduction from A to B is an elementary function f such that $x \in A \Leftrightarrow f(x) \in B$. We write $A \leqslant_m^{\mathrm{elem}} B$ if there exists an elementary reduction from A to B.

Proposition 2.4 ([125])**.** TOWER *is closed under* $\leqslant_m^{\mathrm{elem}}$.

For our purposes, however, already a weaker complexity class is sufficient.

Definition 2.5. $\mathrm{TOWER}(\mathrm{poly}) := \mathrm{TIME}\left(\exp_{n^{\mathcal{O}(1)}}(1)\right)$.

Hence $\mathrm{TOWER}(\mathrm{poly})$ is the class of problems decidable by a deterministic Turing machine in time $\exp_{p(n)}(1)$ for some polynomial p. The reader may verify that both $\mathrm{ATIME\text{-}ALT}(\exp_k, \mathrm{poly})$ and $\mathrm{TOWER}(\mathrm{poly})$ are closed under $\leqslant_m^{\mathrm{P}}$ and $\leqslant_m^{\log}$.

There are several problems in TOWER that also have non-elementary lower bounds (cf. Schmitz [125] and the survey of Meyer [114]). To name a few, these include the satisfiability problem of separated first-order logic [133, 140], the equivalence problem for star-free regular expressions [132], the first-order theory of finite trees [17], the theory of weak monadic second-order logic with one successor [115, 121], or, more recently, the satisfiability problem of modal separation logic [23]. It is not hard to check that all these listed problems are in fact in TOWER(poly).[1] For the latter two, indeed also TOWER-hardness under $\leqslant_m^{\text{elem}}$ is claimed [23], which trivially implies TOWER(poly)-hardness and thus completeness. Yet, we have to be careful: In a sense, $\leqslant_m^{\text{elem}}$ is too "coarse" as a reduction for TOWER(poly), which can be seen by the argument below. For this reason, we only use $\leqslant_m^{\text{P}}$ and $\leqslant_m^{\log}$ as reductions in this thesis.

Proposition 2.6. TOWER(poly) *is not closed under* $\leqslant_m^{\text{elem}}$.

Proof. We claim that every $\leqslant_m^{\text{elem}}$-hard set for TOWER(poly) is also $\leqslant_m^{\text{elem}}$-hard for TOWER. This proves the proposition as follows. Assume that TOWER(poly) is closed under $\leqslant_m^{\text{elem}}$, and let A be any TOWER(poly)-hard problem (such A exists; see also Theorem 4.32 in this thesis). In combination with our claim this would imply TOWER $\subseteq$ TOWER(poly), contradiction to the time hierarchy theorem (see, e.g., Sipser [130, Cor. 9.11]).

Now we prove the claim, so assume A is $\leqslant_m^{\text{elem}}$-hard for TOWER(poly). We have to show that it is also TOWER-hard, so let $B \in$ TOWER be arbitrary. We show $B \leqslant_m^{\text{elem}} A$.

B is decidable in time $\exp_{r(n)}(1)$ for some elementary r. Define the set

$$C := \{x\#0^{r(|x|)} \mid x \in B\},$$

where $0^{r(|x|)}$ is a string of $r(|x|)$ zeroes. First, we show that $C \in$ TOWER(poly). Consider the algorithm that first checks that the input z is of the form $x\#0\cdots0$, computes $r(|x|)$ in elementary time, and checks whether $z = x\#0^{r(|x|)}$. These steps take time that is elementary in $|z|$. Then the algorithm verifies that $x \in B$, which takes time $\exp_{r(|x|)}(1) \leqslant \exp_{|z|}(1)$. Hence $C \in$ TOWER(poly).

By assumption, $C \leqslant_m^{\text{elem}} A$ via an elementary reduction f. But clearly also $B \leqslant_m^{\text{elem}} C$ by the elementary reduction $g\colon x \mapsto x\#0^{r(|x|)}$. As a consequence, the function $h := f \circ g$ is a reduction from B to A. Now h is computable in time $\exp_{k_1}(\exp_{k_2}(n)) = \exp_{k_1+k_2}(n)$ for fixed $k_1, k_2 \geqslant 0$ depending on f and g, and hence again elementary. □

In this thesis, we present several problems that are $\leqslant_m^{\log}$-complete for TOWER(poly), which to the best knowledge of the author is the first explicit completeness result for this class under an "efficient" reduction.

2.2 Team logic

We proceed with the formal definition of propositional, modal and first-order team logic, and mention existing results regarding their computational complexity.

The syntax of team logics usually coincides with that of classical logic, with the exception of non-classical *atoms of dependency* and the Boolean negation $\sim$. In this thesis,

[1] Rabin [121] erroneously claims an elementary upper bound; this is observed and corrected by Meyer [115].

we use Greek letters $\varphi, \psi, \theta, \ldots$ for formulas, Latin letters $x, y, z, w, \ldots$ for variables and $t, u, \ldots$ for terms. Atomic propositions are written $p, q, r, \ldots$. Furthermore, $\alpha, \beta, \gamma, \ldots$ are reserved for classical formulas and $\varphi, \psi, \theta, \ldots$ for arbitrary team-logical formulas.

A *fragment* $\mathcal{L}'$ of a logic $\mathcal{L}$ is a logic with the same semantics, but with only a subset of formulas available. In other words, it is a syntactic restriction of $\mathcal{L}$.

Given a formula φ, we let $|\varphi|$ denote the *size* or *length* of the formula φ over some suitable encoding. In this thesis, every atomic formula is counted as having length one. The set $\mathsf{sub}(\varphi)$ is the set of all *subformulas* defined by the usual recursion.

2.2.1 First-order logic

Classical first-order logic. A *vocabulary* or *first-order language* σ is a (possibly infinite) set of function symbols f and *relation symbols* or *predicates* P, each with finite, non-negative arity $\mathrm{ar}(f)$ and $\mathrm{ar}(P)$, respectively. We also assume a countably infinite set $\mathsf{Var} = \{x_1, x_2, \ldots\}$ of *first-order variables*. The set of σ*-terms* is defined in the standard way by composition of function symbols and variables. A vocabulary σ is called *relational* if it contains no function symbols. If t is a σ-term, then $\mathsf{Var}(t)$ is the set of variables occurring in t.

Formulas of classical *first-order logic* σ-FO are given by the grammar

$$\alpha ::= P\vec{t} \mid t_1 = t_2 \mid \top \mid \bot \mid \neg\alpha \mid \alpha \wedge \alpha \mid \alpha \vee \alpha \mid \exists x\, \alpha \mid \forall x\, \alpha,$$

where $P \in \sigma$ is a predicate, $x \in \mathsf{Var}$, $\vec{t} = (t_1, \ldots, t_{\mathrm{ar}(P)})$ and $t_1, t_2, \ldots$ are σ-terms.

We use the usual abbreviations $\alpha \to \beta := \neg\alpha \vee \beta$ and $\alpha \leftrightarrow \beta := (\alpha \to \beta) \wedge (\beta \to \alpha)$. If $\vec{t} = (t_1, \ldots, t_n)$ and $\vec{u} = (u_1, \ldots, u_n)$ are tuples of terms, then we use the shorthand $\vec{t} = \vec{u}$ for the formula $\bigwedge_{i=1}^{n} t_i = u_i$.

If α is a formula, then $\mathsf{Fr}(\alpha)$ and $\mathsf{Var}(\alpha)$ denote the set of free resp. of all variables in α, defined in the standard way. A *sentence* is a formula with no free variables.

First-order logic σ-FO is evaluated in σ*-structures*, which are pairs $\mathcal{A} = (A, \sigma^{\mathcal{A}})$, where A is a non-empty set called *domain* of $\mathcal{A}$, often written $|\mathcal{A}|$, and $\sigma^{\mathcal{A}}$ maps each symbol $S \in \sigma$ to a function resp. relation $S^{\mathcal{A}}$ of suitable arity that interprets S, i.e., $f^{\mathcal{A}} \colon A^r \to A$ for an r-ary function symbol f, and $P^{\mathcal{A}} \subseteq A^r$ for an r-ary predicate P. We often identify a structure $\mathcal{A}$ and its domain if the meaning is clear.

Let $X \subseteq \mathsf{Var}$. A function $s \colon X \to \mathcal{A}$ is called an *assignment*. If $s \colon X \to \mathcal{A}$ and $\mathrm{dom}\, s \supseteq \mathsf{Var}(t)$, then $t\langle \mathcal{A}, s\rangle$ is the usual evaluation of the term t in $\mathcal{A}$ under the assignment s. That is, if $t = x$ is a variable, then $t\langle \mathcal{A}, s\rangle = s(x)$, and if $t = f(t_1, \ldots, t_n)$ for a function symbol f and terms $t_1, \ldots, t_n$, then $t\langle \mathcal{A}, s\rangle = f^{\mathcal{A}}(t_1\langle \mathcal{A}, s\rangle, \ldots, t_n\langle \mathcal{A}, s\rangle)$. For tuples $\vec{t} = (t_1, \ldots, t_n)$, we define $\vec{t}\langle \mathcal{A}, s\rangle$ as $(t_1\langle \mathcal{A}, s\rangle, \ldots, t_n\langle \mathcal{A}, s\rangle)$.

If $s \colon X \to \mathcal{A}$ and $x \in \mathsf{Var}$, then $s^x_a \colon X \cup \{x\} \to \mathcal{A}$ is the updated assignment that sends x to a and all $y \in X \setminus \{x\}$ to $s(y)$.

We will often suppress the vocabulary σ in the notation if it does not matter. Next, we define the classical satisfaction relation $(\mathcal{A}, s) \vDash \alpha$, known as *Tarski semantics*.

$$
\begin{aligned}
(\mathcal{A}, s) &\vDash P\vec{t} && \Leftrightarrow \vec{t}\langle \mathcal{A}, s\rangle \in P^{\mathcal{A}},\\
(\mathcal{A}, s) &\vDash t_1 = t_2 && \Leftrightarrow t_1\langle \mathcal{A}, s\rangle = t_2\langle \mathcal{A}, s\rangle,\\
(\mathcal{A}, s) &\vDash \neg\alpha && \Leftrightarrow (\mathcal{A}, s) \nvDash \alpha,\\
(\mathcal{A}, s) &\vDash \top && \quad \text{always},\\
(\mathcal{A}, s) &\vDash \bot && \quad \text{never},\\
(\mathcal{A}, s) &\vDash \alpha \wedge \beta && \Leftrightarrow (\mathcal{A}, s) \vDash \alpha \text{ and } (\mathcal{A}, s) \vDash \beta,\\
(\mathcal{A}, s) &\vDash \alpha \vee \beta && \Leftrightarrow (\mathcal{A}, s) \vDash \alpha \text{ or } (\mathcal{A}, s) \vDash \beta,\\
(\mathcal{A}, s) &\vDash \exists x\, \alpha && \Leftrightarrow (\mathcal{A}, s^x_a) \vDash \alpha \text{ for some } a \in \mathcal{A},\\
(\mathcal{A}, s) &\vDash \forall x\, \alpha && \Leftrightarrow (\mathcal{A}, s^x_a) \vDash \alpha \text{ for all } a \in \mathcal{A}.
\end{aligned}
$$

A *model* of a σ-formula α is a pair $(\mathcal{A}, s)$ such that $(\mathcal{A}, s) \vDash \alpha$. If the assignment s does not matter, i.e., if α is a sentence, we sometimes simply call $\mathcal{A}$ a model of α.

First-order team logic. Next, we introduce team semantics and *first-order team logic* σ-$\mathsf{FO}(\sim)$. The set of σ-$\mathsf{FO}(\sim)$-*formulas* (or simply σ-*formulas*) is given by the grammar

$$\varphi ::= P\vec{t} \mid \neg P\vec{t} \mid t_1 = t_2 \mid \neg t_1 = t_2 \mid \top \mid \bot \mid {\sim}\varphi \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \exists x\, \varphi \mid \forall x\, \varphi,$$

where $P \in \sigma$ is a predicate, $\vec{t} = (t_1, \ldots, t_{\mathrm{ar}(P)})$, $x \in \mathsf{Var}$ and $t_1, t_2, \ldots$ are σ-terms. Note that $\neg$ now can only occur at the level of literals, that is, the $\sim$-free fragment of σ-$\mathsf{FO}(\sim)$ consists precisely of all σ-FO-formulas in negation normal form.[1]

Definition 2.7. Let $X \subseteq \mathsf{Var}$. A *team in $\mathcal{A}$ with domain* X is a set T of assignments $s\colon X \to \mathcal{A}$.

If T is a team with domain $X \supseteq Y$, then its *restriction* to Y is $T\restriction Y := \{s\restriction Y \mid s \in T\}$. If t is a term and T is a team with domain $X \supseteq \mathsf{Var}(t)$, then $t\langle \mathcal{A}, T\rangle := \{t\langle \mathcal{A}, s\rangle \mid s \in T\}$, and $\vec{t}\langle \mathcal{A}, T\rangle := \{\vec{t}\langle \mathcal{A}, s\rangle \mid s \in T\}$. If $\mathcal{A}$ is understood, we write only, e.g., $t\langle s\rangle$ or $t\langle T\rangle$.

Like Tarski semantics, the compositional nature of team semantics is based on assignment updates. If T is a team in $\mathcal{A}$ with domain X, then $f\colon T \to \wp^+(\mathcal{A})$ is called a *supplementing function* of T. It extends (or modifies) T to the *supplementing team* $T^x_f := \{\, s^x_a \mid s \in T, a \in f(s)\,\}$ with domain $X \cup \{x\}$. If $f(s) = A$ is constant, we write T^x_A for T^x_f. If $f(s) = |\mathcal{A}|$, then f is also called *duplicating function* and $T^x_{\mathcal{A}}$ is the *duplicating team*.

A pair $(\mathcal{A}, T)$ is *admissible* for a σ-formula φ if $\mathcal{A}$ is a σ-structure and T is a team with domain X for some $X \supseteq \mathsf{Fr}(\varphi)$. We evaluate σ-$\mathsf{FO}(\sim)$-formulas φ on $(\mathcal{A}, T)$ as follows, where t_1, t_2 are terms, $\vec{t}$ is a tuple of terms, $P \in \sigma$ is a predicate, and $x \in \mathsf{Var}$:

$$
\begin{aligned}
(\mathcal{A}, T) &\vDash P\vec{t} && \Leftrightarrow \forall s \in T : \vec{t}\langle \mathcal{A}, s\rangle \in P^{\mathcal{A}}, \text{ or equivalently, } \vec{t}\langle \mathcal{A}, T\rangle \subseteq P^{\mathcal{A}},\\
(\mathcal{A}, T) &\vDash \neg P\vec{t} && \Leftrightarrow \forall s \in T : \vec{t}\langle \mathcal{A}, s\rangle \notin P^{\mathcal{A}}, \text{ or equivalently, } \vec{t}\langle \mathcal{A}, T\rangle \cap P^{\mathcal{A}} = \emptyset,
\end{aligned}
$$

[1] For logics with $\sim$, Väänänen [135] originally employed the symbols ! for $\forall$ and $\otimes$ for $\vee$, so that $\forall$ could denote the $\sim$-dual of $\exists$, and $\vee$ the Boolean disjunction. However, we follow the notation of Galliani [38] instead in order to simplify statements such as the flatness property (see below).

$$
\begin{aligned}
(\mathcal{A},T) &\vDash t_1 = t_2 && \Leftrightarrow \forall s \in T : t_1\langle \mathcal{A}, s\rangle = t_2\langle \mathcal{A}, s\rangle,\\
(\mathcal{A},T) &\vDash \neg t_1 = t_2 && \Leftrightarrow \forall s \in T : t_1\langle \mathcal{A}, s\rangle \neq t_2\langle \mathcal{A}, s\rangle,\\
(\mathcal{A},T) &\vDash \top && \quad\text{always},\\
(\mathcal{A},T) &\vDash \bot && \Leftrightarrow T = \emptyset,\\
(\mathcal{A},T) &\vDash {\sim}\psi && \Leftrightarrow (\mathcal{A},T) \nvDash \psi,\\
(\mathcal{A},T) &\vDash \psi \wedge \theta && \Leftrightarrow (\mathcal{A},T) \vDash \psi \text{ and } (\mathcal{A},T) \vDash \theta,\\
(\mathcal{A},T) &\vDash \psi \vee \theta && \Leftrightarrow \exists S, U \subseteq T \text{ such that } T = S \cup U,\, (\mathcal{A},S) \vDash \psi, \text{ and } (\mathcal{A},U) \vDash \theta,\\
(\mathcal{A},T) &\vDash \exists x\, \psi && \Leftrightarrow (\mathcal{A},T^x_f) \vDash \psi \text{ for some } f\colon T \to \wp^+(\mathcal{A}),\\
(\mathcal{A},T) &\vDash \forall x\, \psi && \Leftrightarrow (\mathcal{A},T^x_{\mathcal{A}}) \vDash \psi.
\end{aligned}
$$

As abbreviations, we use the *Boolean disjunction* $\varphi \varovee \psi := {\sim}({\sim}\varphi \wedge {\sim}\psi)$, the *non-emptiness atom* $\textsc{ne} := {\sim}\bot$, and *strong absurdity* $⫫ := {\sim}\top$. Sometimes also the duals $\varphi \otimes \psi := {\sim}({\sim}\varphi \vee {\sim}\psi)$, $\triangle\varphi := {\sim}\Diamond{\sim}\varphi$ and $!x\, \varphi := {\sim}\exists x\, {\sim}\varphi$ are used.

A *model* of a formula φ is a pair $(\mathcal{A},T)$ such that $(\mathcal{A},T) \vDash \varphi$. We say that φ *entails* ψ, in symbols $\varphi \vDash \psi$, if $(\mathcal{A},T) \vDash \varphi$ implies $(\mathcal{A},T) \vDash \psi$ for all $(\mathcal{A},T)$ that are admissible for both φ and ψ. If $\varphi \vDash \psi$ and $\psi \vDash \varphi$, then φ and ψ are said to be *equivalent*, in symbols $\varphi \equiv \psi$. A formula φ is *satisfiable* if it has at least one model, and φ is *valid* if every $(\mathcal{A},T)$ admissible for φ is a model of φ.

Basic properties. If φ, ψ, θ are formulas, then $\varphi[\psi/\theta]$ is the formula obtained from φ by replacing every occurrence of the subformula ψ by θ (but no occurrences that appear newly as part of θ). Formally, $\varphi[\psi/\theta] := \theta$ if $\varphi = \psi$, and otherwise

$$
\varphi[\psi/\theta] := \begin{cases}
\varphi & \text{if } \varphi \text{ is atomic},\\
{\sim}(\varphi'[\psi/\theta]) & \text{if } \varphi = {\sim}\varphi',\\
\neg(\varphi'[\psi/\theta]) & \text{if } \varphi = \neg\varphi',\\
\varphi_1[\psi/\theta] \circ \varphi_2[\psi/\theta] & \text{if } \varphi = \varphi_1 \circ \varphi_2 \text{ and } \circ \in \{\wedge, \vee\},\\
\partial x\, (\varphi'[\psi/\theta]) & \text{if } \varphi = \partial x\, \varphi' \text{ and } \partial \in \{\exists, \forall\},\, x \in \mathsf{Var}.
\end{cases}
$$

Team semantics satisfies the *full abstraction principle*:

Proposition 2.8 ([135])**.** *Let φ, ψ, θ be $\mathsf{FO}({\sim})$-formulas. If $\psi \equiv \theta$, then $\varphi \equiv \varphi[\psi/\theta]$.*

Classical formulas have the *flatness* property.[1]

Definition 2.9 (Flatness)**.** A $\mathsf{FO}({\sim})$-formula φ is *flat* if, for all admissible $(\mathcal{A},T)$, it holds that $(\mathcal{A},T) \vDash \varphi$ if and only if $(\mathcal{A},\{s\}) \vDash \varphi$ for all $s \in T$.

Proposition 2.10 ([135])**.** *Every FO-formula is flat. Moreover, if $\alpha \in \mathsf{FO}$, then $(\mathcal{A},T) \vDash \alpha$ if and only if $(\mathcal{A},s) \vDash \alpha$ (i.e., in Tarski semantics) for all $s \in T$.*

[1]The term first appeared with Hodges [70], who defined $\downarrow\varphi$ as the *flattening* of the formula φ. In his notation, flat formulas φ are precisely those where $\varphi \equiv \downarrow\varphi$.

A σ-formula φ is *downward closed* (resp. *upward closed*) if for every σ-structure $\mathcal{A}$ and teams $S \subseteq T$ (resp. $S \supseteq T$) it holds that $(\mathcal{A}, T) \vDash \varphi$ implies $(\mathcal{A}, S) \vDash \varphi$. It is *union closed* if for every σ-structure $\mathcal{A}$ and set $\mathcal{T}$ of teams in $\mathcal{A}$ it holds that $\forall T \in \mathcal{T} : (\mathcal{A}, T) \vDash \varphi$ implies $(\mathcal{A}, \bigcup \mathcal{T}) \vDash \varphi$. It has the *empty team property* if $(\mathcal{A}, \emptyset) \vDash \varphi$ for every σ-structure $\mathcal{A}$.

Proposition 2.11 ([135]). *Every* FO*-formula is downward closed, union closed and has the empty team property.*

Observe that union closure implies the empty team property, and that flatness is equivalent to combined downward and union closure. Also, a formula φ is downward closed (upward closed) if and only if $\sim\varphi$ is upward closed (downward closed).

Negation. It is useful to add the classical negation operator $\neg$ as a primitive connective to team logic. Most authors allow $\neg$ only in front of atoms, with $\neg\alpha$ defined merely as an abbreviation for the formula resulting from pushing $\neg$ inwards to the atomic level using the classical equivalences $\neg(\alpha \wedge \beta) \equiv \neg\alpha \vee \neg\beta$, $\neg(\alpha \vee \beta) \equiv \neg\alpha \wedge \neg\beta$, $\neg\top \equiv \bot$, $\neg\bot \equiv \top$, $\neg\exists x\alpha \equiv \forall x\neg\alpha$, and $\neg\forall x\alpha \equiv \exists x\neg\alpha$ (cf., e.g., [85, 135]).

Originally, Hodges [70, 71] included $\neg$ in his semantics as the combination of two operators called "game negation" and "flattening". The definition of $\neg$ in his terms is the following.

$$(\mathcal{A}, T) \vDash \neg\varphi \quad \Leftrightarrow \quad \forall s \in T : (\mathcal{A}, \{s\}) \nvDash \varphi. \tag{$\star$}$$

This definition has also been used by Yang et al. [74, 141, 144] for the purpose of defining *uniform substitution* inside formulas such as $\neg p$, as well as by Kuusisto [93] for his *double team semantics*.

With $\neg$ as a primitive defined as in ($\star$), we avoid the limitation of formulas to negation normal form. It is easy to prove that this does not change the semantics [141]. Alternatively, $\neg\varphi$ is definable as an abbreviation:

$$\neg\varphi \equiv {\sim}\big(\top \vee (\textsc{ne} \wedge {\sim}(\top \vee (\textsc{ne} \wedge {\sim}\varphi)))\big)$$

This formula states that every non-empty subteam contains a non-empty subteam satisfying $\sim\varphi$, which is precisely the case if every singleton satisfies $\sim\varphi$. (A formal proof for this is given as part of Section 4.2.2). Consequently, we can assume $\neg$ as part of the syntax whenever it is convenient to do so.

The following abbreviation by Galliani [41] and Kontinen and Nurmi [84] is useful as well. For $\alpha \in$ FO and $\varphi \in$ FO($\sim$), define $\alpha \hookrightarrow \varphi := \neg\alpha \vee (\alpha \wedge \varphi)$, and if T is a team in $\mathcal{A}$, then we call $T_\alpha := \{ s \in T \mid (\mathcal{A}, \{s\}) \vDash \alpha \}$ the *team* T *conditioned to* α.

Proposition 2.12. $(\mathcal{A}, T) \vDash \alpha \hookrightarrow \varphi$ *if and only if* $(\mathcal{A}, T_\alpha) \vDash \varphi$.

Proof. Clear. See also Galliani [41, Lem. 16]. □

Non-classical atoms. We proceed with the common non-classical atoms for team semantics. In what follows, let $\vec{t}, \vec{u}, \vec{v}$ be tuples of terms.

The *dependence atom* $\mathsf{dep}(\vec{t};\vec{u})$ is due to Väänänen [135]. Grädel and Väänänen [49] introduced the *independence atom* $\vec{t} \perp_{\vec{u}} \vec{v}$. Finally, Galliani [37] defined[1] the *inclusion atom* $\vec{t} \subseteq \vec{u}$ and *exclusion atom* $\vec{t} \mid \vec{u}$, where $\vec{t}$ and $\vec{u}$ are tuples of equal length. The semantics of these atoms is as follows.

$$\begin{aligned}
(\mathcal{A},T) \vDash \mathsf{dep}(\vec{t};\vec{u}) \quad &\Leftrightarrow \quad \forall s,s' \in T : (\vec{t}\langle s\rangle = \vec{t}\langle s'\rangle \Rightarrow \vec{u}\langle s\rangle = \vec{u}\langle s'\rangle),\\
(\mathcal{A},T) \vDash \vec{u} \perp_{\vec{t}} \vec{v} \quad &\Leftrightarrow \quad \forall s,s' \in T : \big(\vec{t}\langle s\rangle = \vec{t}\langle s'\rangle \Rightarrow\\
&\qquad \exists s'' \in T : \vec{t}\vec{u}\langle s\rangle = \vec{t}\vec{u}\langle s''\rangle \text{ and } \vec{v}\langle s'\rangle = \vec{v}\langle s''\rangle\big),\\
(\mathcal{A},T) \vDash \vec{t} \subseteq \vec{u} \quad &\Leftrightarrow \quad \forall s \in T : \exists s' \in T : \vec{t}\langle s\rangle = \vec{u}\langle s'\rangle,\\
(\mathcal{A},T) \vDash \vec{t} \mid \vec{u} \quad &\Leftrightarrow \quad \forall s,s' \in T : \vec{t}\langle s\rangle \neq \vec{u}\langle s'\rangle.
\end{aligned}$$

The atom $\mathsf{dep}(\vec{t};\vec{u})$ is also known as $=(\vec{t};\vec{u})$ in the literature. For a single term u, the atom $\mathsf{dep}(u)$ is called *constancy atom*, as it means that u is constant (since it depends on nothing).

The extension of FO by the respective class of atoms is called *dependence logic* $\mathsf{FO}(\mathsf{dep})$, *independence logic* $\mathsf{FO}(\perp)$, *inclusion logic* $\mathsf{FO}(\subseteq)$ and *exclusion logic* $\mathsf{FO}(\mid)$. The logics $\mathsf{FO}(\mathsf{dep},\sim)$, $\mathsf{FO}(\perp,\sim)$ and so on extend $\mathsf{FO}(\sim)$ analogously.

A desirable property of logic is *locality*, which means that the truth of a formula depends only on the assignments to its free variables.

Definition 2.13. A formula φ is *local* if $(\mathcal{A},T) \vDash \varphi \Leftrightarrow (\mathcal{A},T{\upharpoonright}\mathrm{Fr}(\varphi))$ for all admissible $(\mathcal{A},T)$. We say that a logic is local if all its formulas are.

Proposition 2.14 ([37, 49, 135])**.** *Let* $\mathbf{D} \in \{\mathsf{dep},\perp,\subseteq,\mid\}$. *Then* $\mathsf{FO}(\mathbf{D},\sim)$ *is local.*

Proof. Easily proven by induction on the formula φ (cf. [135, Lem. 3.27], [37, Thm. 2.2]). The base case of atomic formulas and the inductive step for each connective work as in Galliani's proof [37, Thm. 4.22], to which the $\sim$-case can be added in the obvious way. ☐

Proposition 2.15 ([37, 135])**.** $\mathsf{FO}(\mathsf{dep})$ *and* $\mathsf{FO}(\mid)$ *are downward closed and have the empty team property, but are not union closed.* $\mathsf{FO}(\subseteq)$ *is union closed and has the empty team property, but is not downward closed.* $\mathsf{FO}(\perp)$ *has the empty team property, but is neither downward closed nor union closed.*

We also follow the notation of Kontinen et al. [82] and use the formula $\mathrm{E}\alpha$, for $\alpha \in \mathsf{FO}$, which states that some assignment in the current team satisfies α. It can be defined as $\sim\neg\alpha$ or as $\top \vee (\textsc{ne} \wedge \alpha)$.

Lax and strict semantics. Galliani [37] noticed that there are formalisms under which locality fails, notably for so-called *strict* semantics, as opposed to *lax* semantics. While these originally were two alternative semantics for certain connectives, in this thesis, we introduce them as separate syntactic elements, namely *strict splitting* $\dot\vee$ and the *strict*

[1] Although he claims that the notation is due to Grädel.

existential quantifier $\dot\exists$, as opposed to the *lax* operators $\vee$ and $\exists$ introduced before. Their definition is (with the lax connectives restated to compare):

$$\begin{aligned}
(\mathcal{A}, T) \vDash \psi \dot\vee \theta &\Leftrightarrow \exists S, U \subseteq T \text{ such that } T = S \cup U,\ S \cap U = \emptyset, \text{ and}\\
&\qquad (\mathcal{A}, S) \vDash \psi \text{ and } (\mathcal{A}, U) \vDash \theta,\\
(\mathcal{A}, T) \vDash \psi \vee \theta &\Leftrightarrow \exists S, U \subseteq T \text{ such that } T = S \cup U,\ (\mathcal{A}, S) \vDash \psi \text{ and } (\mathcal{A}, U) \vDash \theta,\\
(\mathcal{A}, T) \vDash \dot\exists x\, \varphi &\Leftrightarrow (\mathcal{A}, T^x_f) \vDash \varphi \text{ for some } f\colon T \to \wp^1(\mathcal{A}),\\
(\mathcal{A}, T) \vDash \exists x\, \varphi &\Leftrightarrow (\mathcal{A}, T^x_f) \vDash \varphi \text{ for some } f\colon T \to \wp^+(\mathcal{A}).
\end{aligned}$$

For $\dot\vee$, the condition $S \cap U = \emptyset$ is new, and for $\dot\exists$, only $\wp^+$ was changed to $\wp^1$.

In Väänänen's original definition [135], there was no distinction between lax and strict semantics. This is because for dependence logic, and in general for downward closed logics, both are equivalent. This also applies to downward closed $\mathsf{FO}(\sim)$-formulas:

Proposition 2.16. *Let $\varphi, \psi \in \mathsf{FO}(\sim)$ such that φ is downward closed. Then $\varphi \vee \psi \equiv \varphi \dot\vee \psi$ and $\exists x\varphi \equiv \dot\exists x\varphi$.*

Proof. Clearly $\varphi \dot\vee \psi$ entails $\varphi \vee \psi$ and $\dot\exists x\varphi$ entails $\exists x\varphi$ by definition. If conversely $(\mathcal{A}, T) \vDash \varphi \vee \psi$ via subteams $S, U \subseteq T$ such that $S \cup U = T$, $(\mathcal{A}, S) \vDash \varphi$ and $(\mathcal{A}, U) \vDash \psi$, then we instead split T into the subteams $T \setminus U$ and U. Since $T \setminus U \subseteq S$ and φ is downward closed, this proves $(\mathcal{A}, T) \vDash \varphi \dot\vee \psi$.

Likewise, suppose $(\mathcal{A}, T) \vDash \exists x\varphi$ via some supplementing function $f\colon T \to \wp^+(\mathcal{A})$. By the axiom of choice, there also is some function $f'\colon T \to \wp^1(\mathcal{A})$. Then $T^x_{f'} \subseteq T^x_f$. By downward closure, we obtain $(\mathcal{A}, T^x_{f'}) \vDash \varphi$, so $(\mathcal{A}, T) \vDash \dot\exists x\varphi$ is witnessed by f'. □

In this thesis, where we assume negation $\sim$ as part of the logic, the distinction between strict and lax semantics becomes apparent already for simple formulas such as $\mathsf{E}\top \vee \mathsf{E}\top \not\equiv \mathsf{E}\top \dot\vee \mathsf{E}\top$, where the former defines non-emptiness, but the latter means that the team contains at least two assignments. If nothing else is stated, the notation $\mathsf{FO}(\cdots)$ will always refer to lax connectives and not include $\dot\vee$ and $\dot\exists$.

Decision problems. Let us define the central decision problems of logic in the setting of team semantics.

Definition 2.17. Let Ψ be a set of formulas. The *satisfiability problem* of Ψ is the set

$$\mathrm{SAT}(\Psi) := \{\varphi \in \Psi \mid \varphi \text{ is satisfiable}\}.$$

Definition 2.18. Let Ψ be a set of formulas. The *validity problem* of Ψ is the set

$$\mathrm{VAL}(\Psi) := \{\varphi \in \Psi \mid \varphi \text{ is valid}\}.$$

Definition 2.19. Let Ψ be a set of formulas. The *model checking problem* of Ψ is the set

$$\mathrm{MC}(\Psi) := \{(\mathcal{A}, T, \varphi) \mid \varphi \in \Psi,\ (\mathcal{A}, T) \text{ admissible for } \varphi, \text{ and } (\mathcal{A}, T) \vDash \varphi\}.$$

Note that the satisfiability problem becomes trivial for logics with the empty team property. For this reason, in the literature on team semantics, the empty team is usually excluded from the definition of satisfiability [37, 49, 135]. However, since we have NE, both problems are equally hard: A formula φ is satisfiable iff $\top \vee \varphi$ is satisfied in some non-empty team, and conversely φ is satisfied in some non-empty team iff NE $\wedge\, \varphi$ is satisfiable at all.

Galliani [37], Grädel and Väänänen [49], and Väänänen [135] showed that sentences of inclusion, exclusion, independence and dependence logic all have the same expressive power, namely that of existential second-order logic sentences, and that there are effective translations in both directions.

The classes Σ_2 and Π_2 are the second level of the *Lévy hierarchy* [95].

Theorem 2.20 ([37, 49, 135])**.** *Let* $\mathbf{D} \in \{\mathsf{dep}, \perp, \subseteq, |\}$*. Then the satisfiability problem of* $\mathsf{FO}(\mathbf{D})$ *restricted to non-empty teams is* Π^0_1*-complete, and the satisfiability problem of* $\mathsf{FO}(\mathbf{D}, \sim)$ *is* Σ_2*-complete. The validity problem of both* $\mathsf{FO}(\mathbf{D})$ *and* $\mathsf{FO}(\mathbf{D}, \sim)$ *is* Π_2*-complete.*

Theorem 2.21 ([44])**.** *For any* $\mathbf{D} \in \{\mathsf{dep}, \perp, \subseteq, |\}$*, the model checking problem of* $\mathsf{FO}(\mathbf{D})$ *is* NExpTime*-complete.*

2.2.2 Modal logic

Classical modal logic. Fix a countably infinite set $\mathsf{Prop} := \{p, q, r, \ldots\}$ of propositional variables. Formulas of modal logic ML are built by the grammar

$$\alpha ::= p \mid \top \mid \perp \mid \neg\alpha \mid \alpha \wedge \alpha \mid \alpha \vee \alpha \mid \Diamond\alpha \mid \Box\alpha \qquad (p \in \mathsf{Prop}).$$

The set of propositional variables occurring in a formula α is $\mathsf{Prop}(\alpha)$. Next, we briefly recall classical Kripke semantics. Let $\Phi \subseteq \mathsf{Prop}$ be finite. A *Kripke structure* (*over* Φ) is a tuple $\mathcal{K} = (W, R, V)$, where W is a set of *worlds* or *points*, (W, R) is a directed graph called *frame* of $\mathcal{K}$, and $V \colon \Phi \to \wp W$ is the *valuation*. Occasionally, by slight abuse of notation, we use the inverse mapping $V^{-1} \colon W \to \wp\Phi$. A *pointed structure* (*over* Φ) is a pair $(\mathcal{K}, w)$ where $\mathcal{K} = (W, R, V)$ is a Kripke structure (over Φ) and $w \in W$. ML-formulas are evaluated in *Kripke semantics*, where $p \in \mathsf{Prop}$ and $(\mathcal{K}, w)$ is a pointed structure:

$$\begin{aligned}
(\mathcal{K}, w) &\vDash p && \Leftrightarrow w \in V(p),\\
(\mathcal{K}, w) &\vDash \top && \ \text{always},\\
(\mathcal{K}, w) &\vDash \perp && \ \text{never},\\
(\mathcal{K}, w) &\vDash \neg\alpha && \Leftrightarrow (\mathcal{K}, w) \nvDash \alpha,\\
(\mathcal{K}, w) &\vDash \alpha \wedge \beta && \Leftrightarrow (\mathcal{K}, w) \vDash \alpha \text{ and } (\mathcal{K}, w) \vDash \beta,\\
(\mathcal{K}, w) &\vDash \alpha \vee \beta && \Leftrightarrow (\mathcal{K}, w) \vDash \alpha \text{ or } (\mathcal{K}, w) \vDash \beta,\\
(\mathcal{K}, w) &\vDash \Diamond\alpha && \Leftrightarrow \exists v \in W : Rwv \text{ and } (\mathcal{K}, v) \vDash \alpha,\\
(\mathcal{K}, w) &\vDash \Box\alpha && \Leftrightarrow \forall v \in W : \text{if } Rwv, \text{ then } (\mathcal{K}, v) \vDash \alpha.
\end{aligned}$$

We sometimes omit $\mathcal{K}$ and write only $w \vDash \alpha$.

Modal team logic. Formulas of *modal team logic* $\mathsf{ML}(\sim)$ are defined by the grammar

$$\varphi ::= p \mid \neg p \mid \top \mid \bot \mid {\sim}\varphi \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \Diamond\varphi \mid \Box\varphi,$$

where again $p \in \mathsf{Prop}$. The semantics of $\mathsf{ML}(\sim)$ is defined on pairs $(\mathcal{K}, T)$ called *structure with team* (*over* Φ), where $\mathcal{K} = (W, R, V)$ is a Kripke structure (over Φ) and $T \subseteq W$ is called *team in* $\mathcal{K}$. A structure with team $(\mathcal{K}, T)$ is *admissible* for a formula φ if $\mathcal{K}$ is a Kripke structure over Φ for some $\Phi \supseteq \mathsf{Prop}(\varphi)$.

The *image* of a team T is $RT := \{v \mid w \in T, (w, v) \in R\}$. For $w \in W$, we simply write Rw instead of $R\{w\}$. Sometimes we use the "iterated" image, defined by $R^0T := T$ and $R^{i+1}T := RR^iT$. An R-*successor team* (or simply *successor team*) of T is a team S such that $S \subseteq RT$ and $T \subseteq R^{-1}S$, where $R^{-1} := \{(v, w) \mid (w, v) \in R\}$. Intuitively, S is formed by picking at least one R-successor of every world in T, which means that every $w \in T$ has a successor in S, and every $v \in S$ has a predecessor in T. Equivalently, S is a successor team of T if and only if $S = \bigcup_{w \in T} f(w)$ for some $f \in \prod_{w \in T} \wp^+(Rw)$. The evaluation of $\mathsf{ML}(\sim)$-formulas is now as follows, where $(\mathcal{K}, T)$ is an admissible pair and $p \in \mathsf{Prop}$:

$$
\begin{aligned}
(\mathcal{K}, T) &\vDash p && \Leftrightarrow \forall w \in T : (\mathcal{K}, w) \vDash p, \text{ or equivalently, } T \subseteq V(p),\\
(\mathcal{K}, T) &\vDash \neg p && \Leftrightarrow \forall w \in T : (\mathcal{K}, w) \nvDash p, \text{ or equivalently, } T \cap V(p) = \emptyset,\\
(\mathcal{K}, T) &\vDash \top && \text{ always},\\
(\mathcal{K}, T) &\vDash \bot && \Leftrightarrow T = \emptyset,\\
(\mathcal{K}, T) &\vDash {\sim}\psi && \Leftrightarrow (\mathcal{K}, T) \nvDash \psi,\\
(\mathcal{K}, T) &\vDash \psi \wedge \theta && \Leftrightarrow (\mathcal{K}, T) \vDash \psi \text{ and } (\mathcal{K}, T) \vDash \theta,\\
(\mathcal{K}, T) &\vDash \psi \vee \theta && \Leftrightarrow \exists S, U \subseteq T \text{ such that } T = S \cup U, (\mathcal{K}, S) \vDash \psi, \text{ and } (\mathcal{K}, U) \vDash \theta,\\
(\mathcal{K}, T) &\vDash \Diamond\psi && \Leftrightarrow (\mathcal{K}, S) \vDash \psi \text{ for some successor team } S \text{ of } T,\\
(\mathcal{K}, T) &\vDash \Box\psi && \Leftrightarrow (\mathcal{K}, RT) \vDash \psi.
\end{aligned}
$$

Like in the classical case, we sometimes omit $\mathcal{K}$ and write only $T \vDash \varphi$. As for first-order team logic, let $\textsc{ne} := {\sim}\bot$, $⫫ := {\sim}\top$ and $E\alpha := {\sim}\neg\alpha$. (We can again consider negation $\neg$ as part of the syntax where this is convenient.)

Basic properties. The definition of flatness is similar to the first-order case:

Definition 2.22 (Flatness)**.** An $\mathsf{ML}(\sim)$-formula φ is *flat* if, for all admissible $(\mathcal{K}, T)$, it holds that $(\mathcal{K}, T) \vDash \varphi$ if and only if $(\mathcal{K}, \{w\}) \vDash \varphi$ for all $w \in T$.

The definitions of satisfiability, validity, model checking, entailment, substitution, downward/upward closure, union closure and the empty team property are also analogous.

Proposition 2.23 ([128])**.** *Every* ML*-formula is flat. Moreover, if* $\alpha \in \mathsf{ML}$*, then* $(\mathcal{K}, T) \vDash \alpha$ *if and only if it holds that* $(\mathcal{K}, w) \vDash \alpha$ *(in Kripke semantics) for all* $w \in T$.

The *modal depth* $\mathsf{md}(\varphi)$ of a formula φ is recursively defined:

$$\begin{aligned}
\mathsf{md}(p), \mathsf{md}(\top), \mathsf{md}(\bot) &:= 0\\
\mathsf{md}({\sim}\varphi), \mathsf{md}(\neg\varphi) &:= \mathsf{md}(\varphi)\\
\mathsf{md}(\varphi \wedge \psi), \mathsf{md}(\varphi \vee \psi) &:= \max\{\mathsf{md}(\varphi), \mathsf{md}(\psi)\}\\
\mathsf{md}(\Diamond\varphi), \mathsf{md}(\Box\varphi) &:= \mathsf{md}(\varphi) + 1
\end{aligned}$$

ML_k and $\mathsf{ML}_k({\sim})$ are the fragments of ML and $\mathsf{ML}({\sim})$ with modal depth $\leqslant k$, respectively. If the propositions are restricted to a fixed set $\Phi \subseteq \mathsf{Prop}$ as well, then the fragment is denoted by ML_k^Φ or $\mathsf{ML}_k^\Phi({\sim})$, respectively.

The *bisimulation* relation $\rightleftharpoons_k^\Phi$ captures the expressive power of modal logic [9, 43].

Definition 2.24. Let $\Phi \subseteq \mathsf{Prop}$ and $k \geqslant 0$. For $i \in \{1,2\}$, let $(\mathcal{K}_i, w_i)$ be a pointed structure over $\Phi'_i \supseteq \Phi$, where $\mathcal{K}_i = (W_i, R_i, V_i)$. Then $(\mathcal{K}_1, w_1)$ and $(\mathcal{K}_2, w_2)$ are (Φ, k)-*bisimilar*, in symbols $(\mathcal{K}_1, w_1) \rightleftharpoons_k^\Phi (\mathcal{K}_2, w_2)$, if

- $\forall p \in \Phi\colon w_1 \in V_1(p) \Leftrightarrow w_2 \in V_2(p)$,
- and if $k > 0$,
  - $\forall v_1 \in R_1 w_1\colon \exists v_2 \in R_2 w_2\colon (\mathcal{K}_1, v_1) \rightleftharpoons_{k-1}^\Phi (\mathcal{K}_2, v_2)$ (*forward* condition),
  - $\forall v_2 \in R_2 w_2\colon \exists v_1 \in R_1 w_1\colon (\mathcal{K}_1, v_1) \rightleftharpoons_{k-1}^\Phi (\mathcal{K}_2, v_2)$ (*backward* condition).

The next result is standard:

Theorem 2.25 ([43, Thm. 32])**.** *Let $\Phi \subseteq \mathsf{Prop}$ be finite and $k \geqslant 0$. For $i \in \{1,2\}$, let $(\mathcal{K}_i, w_i)$ be a pointed structure over $\Phi'_i \supseteq \Phi$, where $\mathcal{K}_i = (W_i, R_i, V_i)$. Then the following statements are equivalent:*

(1) $\forall \alpha \in \mathsf{ML}_k^\Phi\colon (\mathcal{K}_1, w_1) \vDash \alpha \Leftrightarrow (\mathcal{K}_1, w_2) \vDash \alpha$.

(2) $(\mathcal{K}_1, w_1) \rightleftharpoons_k^\Phi (\mathcal{K}_2, w_2)$.

Furthermore, so-called *characteristic formulas* or *Hintikka formulas* represent the bisimulation equivalence classes syntactically.

Proposition 2.26 ([43, Thm. 32])**.** *Let $\Phi \subseteq \mathsf{Prop}$ be finite, $k \geqslant 0$, and let $(\mathcal{K}, w)$ be a pointed structure over $\Phi' \supseteq \Phi$. Then there is a formula $\zeta \in \mathsf{ML}_k^\Phi$ such that for all pointed structures $(\mathcal{K}', w')$ we have $(\mathcal{K}, w) \rightleftharpoons_k^\Phi (\mathcal{K}', w')$ if and only if $(\mathcal{K}', w') \vDash \zeta$.*

The notion of bisimulation was lifted to team semantics by Hella et al. [60] and Kontinen et al. [82, 83]:

Definition 2.27. Let $\Phi \subseteq \mathsf{Prop}$ be and $k \geqslant 0$. For $i \in \{1,2\}$, let $(\mathcal{K}_i, T_i)$ be a structure with team over $\Phi'_i \supseteq \Phi$. Then $(\mathcal{K}_1, T_1)$ and $(\mathcal{K}_2, T_2)$ are (Φ, k)-*team-bisimilar*, $(\mathcal{K}_1, T_1) \rightleftharpoons_k^\Phi (\mathcal{K}_2, T_2)$, if

- $\forall w_1 \in T_1\colon \exists w_2 \in T_2\colon (\mathcal{K}_1, w_1) \rightleftharpoons_k^\Phi (\mathcal{K}_2, w_2)$, and

- $\forall w_2 \in T_2 \colon \exists w_1 \in T_1 \colon (\mathcal{K}_1, w_1) \rightleftharpoons_k^\Phi (\mathcal{K}_2, w_2)$.

If no confusion can arise, we will also refer to teams T_1, T_2 that are (Φ, k)-team-bisimilar simply as (Φ, k)-bisimilar.

Proposition 2.28. *Let $\Phi \subseteq \mathsf{Prop}$ be finite, and $k \geqslant 0$. For $i \in \{1, 2\}$, let $(\mathcal{K}_i, w_i)$ be a pointed structure over $\Phi'_i \supseteq \Phi$, where $\mathcal{K}_i = (W_i, R_i, V_i)$. The following statements are equivalent:*

(1) $\forall \alpha \in \mathsf{ML}_k^\Phi \colon (\mathcal{K}_1, w_1) \vDash \alpha \Leftrightarrow (\mathcal{K}_2, w_2) \vDash \alpha$,

(2) $(\mathcal{K}_1, w_1) \rightleftharpoons_k^\Phi (\mathcal{K}_2, w_2)$, *that is,* $(\mathcal{K}_1, w_1)$ *and* $(\mathcal{K}_2, w_2)$ *are* (Φ, k)*-bisimilar,*

(3) $(\mathcal{K}_1, \{w_1\}) \rightleftharpoons_k^\Phi (\mathcal{K}_2, \{w_2\})$, *that is,* $(\mathcal{K}_1, \{w_1\})$ *and* $(\mathcal{K}_2, \{w_2\})$ *are* (Φ, k)*-team-bisimilar,*

and if $k > 0$,

(4) $(\mathcal{K}_1, w_1) \rightleftharpoons_0^\Phi (\mathcal{K}_2, w_2)$ *and* $(\mathcal{K}_1, R_1 w_1) \rightleftharpoons_{k-1}^\Phi (\mathcal{K}_2, R_2 w_2)$.

Proof. (1) $\Leftrightarrow$ (2) is just Theorem 2.25. (2) $\Leftrightarrow$ (3) follows from Definition 2.27. For $k > 0$, we first show that (2)+(3) $\Rightarrow$ (4). Clearly, $(\mathcal{K}_1, w_1) \rightleftharpoons_0^\Phi (\mathcal{K}_2, w_2)$ follows from (2). By Hella et al. [60, Lem. 3.3], (3) implies $(\mathcal{K}_1, R_1 w_1) \rightleftharpoons_{k-1}^\Phi (\mathcal{K}_2, R_2 w_2)$.

Finally, we show (4) $\Rightarrow$ (2). Suppose $(\mathcal{K}_1, w_1) \rightleftharpoons_0^\Phi (\mathcal{K}_2, w_2)$ and $(\mathcal{K}_1, R_1 w_1) \rightleftharpoons_{k-1}^\Phi (\mathcal{K}_2, R_2 w_2)$. Then to show $(\mathcal{K}_1, w_1) \rightleftharpoons_k^\Phi (\mathcal{K}_2, w_2)$, it is sufficient to prove the *forward* and *backward* conditions of Definition 2.24. Suppose $v_1 \in R_1 w_1$. Since $(\mathcal{K}_1, R_1 w_1) \rightleftharpoons_{k-1}^\Phi (\mathcal{K}_2, R_2 w_2)$, by Definition 2.27 there exists $v_2 \in R_2 w_2$ such that $(\mathcal{K}_1, v_1) \rightleftharpoons_{k-1}^\Phi (\mathcal{K}_2, v_2)$, proving the *forward* condition. The *backward* condition is symmetric. □

This means that the *forward* and *backward* condition from Definition 2.24 can be equivalently stated in terms of team-bisimilarity of the respective image teams.

Characteristic formulas exist also in the team setting:

Theorem 2.29 ([82, Prop. 3.10])**.** *Let $\Phi \subseteq \mathsf{Prop}$ be finite and $k \geqslant 0$, and let $(\mathcal{K}, T)$ be a Kripke structure with team over $\Phi' \supseteq \Phi$ such that T is non-empty. There is an $\mathsf{ML}_k^\Phi(\sim)$-formula φ such that for all admissible Kripke structures with team $(\mathcal{K}', T')$, for T' non-empty, we have $(\mathcal{K}, T) \rightleftharpoons_k^\Phi (\mathcal{K}', T')$ if and only if $(\mathcal{K}', T') \vDash \varphi$.*

Moreover, a characterization similar to Theorem 2.25 exists for team-bisimilarity. The following theorem slightly extends Kontinen et al. [82, Prop. 2.8 and 3.10].

Proposition 2.30. *Let $\Phi \subseteq \mathsf{Prop}$ be finite, and $k \geqslant 0$. Let $(\mathcal{K}_i, T_i)$ be a structure with team for $i \in \{1, 2\}$. Then the following statements are equivalent:*

(1) $\forall \alpha \in \mathsf{ML}_k^\Phi \colon (\mathcal{K}_1, T_1) \vDash \alpha \Leftrightarrow (\mathcal{K}_2, T_2) \vDash \alpha$,

(2) $\forall \varphi \in \mathsf{ML}(\sim)_k^\Phi \colon (\mathcal{K}_1, T_1) \vDash \varphi \Leftrightarrow (\mathcal{K}_2, T_2) \vDash \varphi$,

(3) $(\mathcal{K}_1, T_1) \rightleftharpoons_k^\Phi (\mathcal{K}_2, T_2)$.

Logic	Satisfiability	Validity	References
$\mathsf{PL}(\mathsf{dep})$	NP	NEXPTIME	[96, 139]
$\mathsf{ML}(\mathsf{dep})$	NEXPTIME	NEXPTIME	[51, 128]
$\mathsf{PL}(\bot)$	NP	NEXPTIME-hard, in Π_2^E	[55]
$\mathsf{ML}(\bot)$	NEXPTIME	Π_2^E-hard	[51, 83]
$\mathsf{PL}(\subseteq)$	EXPTIME	co-NP	[55]
$\mathsf{ML}(\subseteq)$	EXPTIME	co-NEXPTIME-hard	[58, 59]
$\mathsf{PL}(\sim)$	ATIME-ALT(exp, poly)	ATIME-ALT(exp, poly)	[53, 56]
$\mathsf{ML}_k(\sim)$	ATIME-ALT($\exp_{k+1}$, poly)	ATIME-ALT($\exp_{k+1}$, poly)	Thm. 4.32
$\mathsf{ML}(\sim)$	TOWER(poly)	TOWER(poly)	Thm. 4.32

Table 2.1: Complexity of satisfiability and validity of propositional and modal logics of dependence, independence, inclusion, and full team logic. Entries are completeness results unless stated otherwise. For logics without $\sim$, the satisfiability problem excludes the empty team.

Proof. An easy induction yields that $(\mathcal{K}_1, \emptyset)$ and $(\mathcal{K}_2, \emptyset)$ satisfy the same $\mathsf{ML}(\sim)$-formulas. Moreover, a team satisfies $\bot$ iff it is empty. Consequently, the above statements are all true if $T_1 = T_2 = \emptyset$, and are all false if exactly one of the teams is empty. So w.l.o.g. let T_1 and T_2 be non-empty. Then (3) $\Rightarrow$ (2) is due to Kontinen et al. [82, Prop. 2.8]. (2) $\Rightarrow$ (1) is clear. Finally, (1) $\Rightarrow$ (3) in shown as follows: Suppose $w_1 \in T_1$. Let ζ be the characteristic ML_k^Φ-formula from Proposition 2.26, i.e., we have $(\mathcal{K}_2, w_2) \vDash \zeta$ iff $(\mathcal{K}_1, w_1) \rightleftharpoons_k^\Phi (\mathcal{K}_2, w_2)$. Then obviously $(\mathcal{K}_1, T_1) \vDash \mathsf{E}\zeta$, so $(\mathcal{K}_1, T_1) \nvDash \neg\zeta$, and consequently $(\mathcal{K}_2, T_2) \nvDash \neg\zeta$ by (1). Hence $(\mathcal{K}_2, T_2) \vDash \mathsf{E}\zeta$. This means that there exists $w_2 \in T_2$ such that $(\mathcal{K}_1, w_1) \rightleftharpoons_k^\Phi (\mathcal{K}_2, w_2)$. As w_1 was arbitrary, and the argument is symmetric, we obtain $(\mathcal{K}_1, T_1) \rightleftharpoons_k^\Phi (\mathcal{K}_2, T_2)$. □

Note that the team analog of (4) in Proposition 2.28 is not equivalent: It is possible that $(\mathcal{K}_1, T_1) \rightleftharpoons_0^\Phi (\mathcal{K}_2, T_2)$ and $(\mathcal{K}_1, R_1T_1) \rightleftharpoons_{k-1}^\Phi (\mathcal{K}_2, R_2T_2)$, but $(\mathcal{K}_1, T_1) \not\rightleftharpoons_k^\Phi (\mathcal{K}_2, T_2)$.

Lax and strict semantics. Similarly to first-order logic, the modal connectives $\vee$ and $\Diamond$ comes in strict and lax variants. Let $\mathcal{K} = (W, R, V)$ be a Kripke structure and $T \subseteq W$.

$$(\mathcal{K}, T) \vDash \psi \mathbin{\dot\vee} \theta \Leftrightarrow \exists S, U \subseteq T \text{ such that } T = S \cup U,\ S \cap U = \emptyset, \text{ and } (\mathcal{K}, S) \vDash \psi \text{ and } (\mathcal{K}, U) \vDash \theta,$$

$$(\mathcal{K}, T) \vDash \dot\Diamond\varphi \quad \Leftrightarrow \exists f \in \prod_{w \in T} \wp^1(Rw) \text{ such that } \left(\mathcal{K}, \bigcup_{w \in T} f(w)\right) \vDash \varphi.$$

We call the team $\bigcup_{w \in T} f(w)$ *strict successor team* if $f(w)$ is a singleton for every w; otherwise it is called *lax successor team*.

It is noteworthy that truth in strict semantics is *not* invariant under bisimulation. An example is the formula $\mathsf{E}\top \mathbin{\dot\vee} \mathsf{E}\top$, which states that the team has at least two elements. However, a result analogous to Proposition 2.16 holds.

Logic	Model Checking	Reference
$\mathsf{PL}(\mathsf{dep})$	NP	[29]
$\mathsf{ML}(\mathsf{dep})$	NP	[29]
$\mathsf{QPL}(\mathsf{dep})$	NExpTime	[53]
$\mathsf{PL}(\perp)$	NP	[56]
$\mathsf{ML}(\perp)$	NP	[83]
$\mathsf{QPL}(\perp)$	NExpTime	[53]
$\mathsf{PL}(\subseteq)$	P	[58]
$\mathsf{ML}(\subseteq)$	P	[58]
$\mathsf{QPL}(\subseteq)$	ExpTime	[53]
$\mathsf{PL}(\sim)$	PSpace	[56]
$\mathsf{ML}(\sim)$	PSpace	[119]
$\mathsf{QPL}(\sim)$	ATime-Alt(exp, poly)	[53]

Table 2.2: Complexity of model checking of propositional and modal logics of dependence, independence, and inclusion, and with negation. All entries are completeness results.

Proposition 2.31. *Let $\varphi, \psi \in \mathsf{ML}(\sim)$ such that φ is downward closed. Then $\varphi \vee \psi \equiv \varphi \dot{\vee} \psi$ and $\Diamond\varphi \equiv \dot{\Diamond}\varphi$.*

Proof. The proof for $\dot{\vee}$ is identical to Proposition 2.16. For $\dot{\Diamond}$, suppose $(\mathcal{K}, T) \vDash \Diamond\varphi$ via some successor team S of T. By the axiom of choice, there is some function $f\colon T \to S$ such that $f(w) \in Rw$ for each $w \in T$. The team $\{f(w) \mid w \in T\} \subseteq S$ is now a strict successor team of T and by downward closure satisfies φ, so $(\mathcal{K}, T) \vDash \dot{\Diamond}\varphi$. Conversely, $(\mathcal{K}, T) \vDash \Diamond\varphi$ again trivially follows from $(\mathcal{K}, T) \vDash \dot{\Diamond}\varphi$. □

With modal team logic, strict semantics was studied, e.g., by Hella et al. [58, 59] and Hella and Stumpf [61]. In these works, the underlying team logic has been enriched by non-downward closed atoms such as inclusion or independence atom, analogously as for first-order logic. As we will not study these here, and they are similarly defined as the first-order atoms, we will skip their definitions here.

Decision problems. The complexity of modal logic equipped with the different atoms has been subject to several studies (cf. Tables 2.1 and 2.2). For the model checking problem of $\mathsf{ML}(\sim)$, the complexity is as follows:

Theorem 2.32 ([119])**.** *The model checking problem for $\mathsf{ML}(\sim)$ is PSpace-complete. Moreover, it is decidable by an alternating Turing machine that runs in polynomial time and with polynomially many alternations in $|\varphi|$, where $(\mathcal{K}, T, \varphi)$ is the input.*

However, the complexity of satisfiability and validity has been open [28, 58, 82, 119]. We consider it in Chapter 4. We do not explicitly consider extensions $\mathsf{ML}(\mathbf{D}, \sim)$ by the various atoms $\mathbf{D} \in \{\mathsf{dep}, \perp, \subseteq, \mid\}$; these logics can be translated to $\mathsf{ML}(\sim)$ in logarithmic space [108] and consequently the same completeness results apply. For instance, the constancy atom $\mathsf{dep}(\alpha)$, where α is a classical formula, is equivalent to $\alpha \varovee \neg\alpha$, and $\mathsf{dep}(\alpha_1, \ldots \alpha_n; \alpha_{n+1})$ is equivalent (cf. [56]) to ${\sim}\Big(\top \vee \bigwedge_{i=1}^{n} \mathsf{dep}(\alpha_i) \wedge {\sim}\mathsf{dep}(\alpha_{n+1})\Big)$.

2.2.3 Propositional team logic

Propositional team logic (resp. propositional dependence logic) has been first studied by Yang [142] and subsequently by Yang and Väänänen [143, 144]. Since it is closely related to modal team logic, we introduce it only briefly.

Syntactically, $\mathsf{PL}(\sim)$ coincides with $\mathsf{ML}_0(\sim)$, i.e., modal team logic without modalities, just like PL coincides with ML_0. Its semantics is based on *propositional teams*.

Definition 2.33. Let $\Phi \subseteq \mathsf{Prop}$. A *team with domain* Φ is a set $T \subseteq (\Phi \to \{0, 1\})$.

Given a $\mathsf{PL}(\sim)$-formula φ, a team is *admissible* if it has domain $\Phi \supseteq \mathsf{Prop}(\varphi)$. The satisfaction relation is similar to $\mathsf{ML}(\sim)$:

$$\begin{aligned}
T &\vDash p && \Leftrightarrow \forall s \in T : s(p) = 1,\\
T &\vDash \neg p && \Leftrightarrow \forall s \in T : s(p) = 0,\\
T &\vDash \top && \quad \text{always},\\
T &\vDash \bot && \Leftrightarrow T = \emptyset,\\
T &\vDash {\sim}\psi && \Leftrightarrow T \nvDash \psi,\\
T &\vDash \psi \wedge \theta && \Leftrightarrow T \vDash \psi \text{ and } T \vDash \theta,\\
T &\vDash \psi \vee \theta && \Leftrightarrow \exists S, U \subseteq T \text{ such that } T = S \cup U,\ S \vDash \psi, \text{ and } U \vDash \theta.
\end{aligned}$$

Adding propositional quantifiers $\exists p, \forall p$ results in the logic $\mathsf{QPL}(\sim)$, a team-semantical analog to quantified Boolean formulas:

$$\begin{aligned}
T &\vDash \exists p\, \varphi &&\Leftrightarrow\ T^p_f \vDash \varphi \text{ for some } f \colon T \to \wp^+(\{0, 1\}),\\
T &\vDash \forall p\, \varphi &&\Leftrightarrow\ T^p_{\{0,1\}} \vDash \varphi.
\end{aligned}$$

These teams are defined as in the first-order case, i.e., $T^p_f := \{s^p_b \mid s \in T, b \in f(s)\}$ and $T^p_{\{0,1\}} := \{s^p_0, s^p_1 \mid s \in T\}$, where s^p_b is the assignment that maps p to b and all $q \neq p$ to $s(q)$. Again, there are also strict connectives:

$$\begin{aligned}
T &\vDash \psi \,\dot{\vee}\, \theta &&\Leftrightarrow\ \exists S, U \subseteq T \text{ such that } T = S \cup U,\ S \cap U = \emptyset,\ S \vDash \psi \text{ and } U \vDash \theta,\\
T &\vDash \dot{\exists} p\, \varphi &&\Leftrightarrow\ T^p_f \vDash \varphi \text{ for some } f \colon T \to \{\{0\}, \{1\}\}.
\end{aligned}$$

Analogously to before, PL resp. QPL corresponds to the $\sim$-free fragment of $\mathsf{PL}(\sim)$ resp. $\mathsf{QPL}(\sim)$, and the logics $\mathsf{PL}(\mathbf{D})$, $\mathsf{PL}(\mathbf{D}, \sim)$, $\mathsf{QPL}(\mathbf{D})$ and $\mathsf{QPL}(\mathbf{D}, \sim)$, for non-classical atoms $\mathbf{D}$, are defined as expected. Concerning the complexity of the above logics, we refer the reader to Tables 2.1 and 2.2 for the relevant completeness results.

3 Abstract team logic

Since the seminal work of Väänänen [135], numerous team-logical formalisms have been proposed, most importantly in first-order logic [135], modal logic [136], propositional logic [142] and temporal logic [90]. Moreover, these logics have been augmented with a plethora of non-classical atoms of dependency [37, 41, 49, 123, 135] and non-classical types of disjunction, implication, and other connectives [2, 61, 123], to only list a few.

In this chapter, we present a framework in which we systematically classify existing team logics in an abstract fashion. We will not consider the concrete atoms and connectives mentioned above, but rather abstractly gather those that have similar properties in classes. While it turns out that the different manifestations of team semantics exhibit many similarities, at the same time, the framework is be flexible enough to allow concepts such as "lax" and "strict" semantics [37], or "synchronous" and "asynchronous" semantics [91].

First, in Section 3.1, some basic definitions are introduced. In Section 3.2 we propose a natural transformation from classical connectives to team-logical ones to which we refer as *teamification*. As the next step, we then introduce the notion of *flatness equivalent* operator pairs (i.e., that behave equivalently on flat formulas), such as lax and strict disjunction, $\vee$ and $\dot{\vee}$. We also distinguish between *strong* and *weak duality*. Section 3.3 deals with connectives that are *operators*, which includes all connectives of modal, propositional and first-order team logic but $\neg$ and $\sim$. Moreover, in Section 3.4, we study a subclass of operators called *transversals* which is well-behaved and yet surprisingly abundant in team logics. As an attempt to generalize strict and lax semantics, we consider the further restriction towards *standard transversals* (Section 3.6) and *relaxations* thereof (Section 3.5). In Section 3.7, we prove our collapse theorem inside our framework; namely that logics such as $\mathsf{PL}(\sim)$, $\mathsf{ML}(\sim)$ and $\mathsf{FO}(\sim)$ collapse to Boolean combinations of flat formulas. We also discuss in Section 3.8 how our framework can be applied to temporal logic. Finally, we conclude with some remarks in Section 3.9.

3.1 Basic definitions

We start with a compositional, algebraic definition of semantics.

Definition 3.1 (Signature)**.** A *signature* is a (possibly infinite) set τ of symbols $\triangle$, called *connectives*, each having a finite *arity* $\mathrm{ar}(\triangle) \in \omega$. A connective with arity zero is an *atom*.

Definition 3.2 (Algebra)**.** Let τ be a signature. A τ-*algebra* is a pair $\mathcal{A} = (X, (f_\triangle)_{\triangle \in \tau})$ where X is a non-empty set, the *carrier* of $\mathcal{A}$, and $f_\triangle \colon X^{\mathrm{ar}(\triangle)} \to X$ for each $\triangle \in \tau$. The $f_\triangle$ are called *operations on* X.

Definition 3.3 (Formula)**.** The set $\mathcal{F}_\tau$ of τ-*formulas* is built as follows. If $\varphi_1, \ldots, \varphi_r$ are τ-formulas and $\triangle \in \tau$ is r-ary, then $\triangle(\varphi_1, \ldots, \varphi_r)$ is a τ-formula (including the case $r = 0$ where $\triangle$ is an atom). Nothing else is a τ-formula.

Formulas themselves constitute an algebra, namely that where each r-ary $\triangle \in \tau$ induces an r-ary composition operation. This is known as *term algebra* (cf. [24, p. 21]).

Definition 3.4. The *length* $|\varphi|$ of a τ-formula φ is inductively defined as

$$|\nabla(\psi_1, \ldots, \psi_r)| := 1 + \sum_{i \in [r]} |\psi_i|.$$

Definition 3.5. The set of *subformulas* $\mathsf{sub}(\varphi)$ of φ is inductively defined as

$$\mathsf{sub}(\nabla(\psi_1, \ldots, \psi_r)) := \{\nabla(\psi_1, \ldots, \psi_r)\} \cup \bigcup_{i \in [r]} \mathsf{sub}(\psi_i).$$

Note that always $|\mathsf{sub}(\varphi)| \leqslant |\varphi|$.

We sometimes identify an algebra $A = (X, (f_\triangle)_{\triangle \in \tau})$ with its carrier and write, e.g., $x \in A$ instead of $x \in X$. We adapt the infix notation where this is common, that is, $x \wedge y$ and $x \vee y$ for $\wedge(x, y)$ and $\vee(x, y)$. Also, if no confusion can arise, we identify the symbol $\triangle$ and the map $f_\triangle$ and sometimes write $\triangle^A$ or even just $\triangle$ instead of $f_\triangle$.

Definition 3.6 (Evaluation of formulas)**.** For a τ-algebra A and τ-formula φ, we inductively define the element $[\![\varphi]\!]^A \in A$ as

$$[\![\triangle(\varphi_1, \ldots, \varphi_{\mathrm{ar}(\triangle)})]\!]^A = \triangle^A([\![\varphi_1]\!]^A, \ldots, [\![\varphi_{\mathrm{ar}(\triangle)}]\!]^A)$$

for all $\triangle \in \tau$ and τ-formulas $\varphi_1, \ldots, \varphi_{\mathrm{ar}(\triangle)}$.

We sometimes refer to the elements of an algebra A as *properties*, and say that φ *defines* the property $[\![\varphi]\!]^A$.

Definition 3.7 (Homomorphism)**.** Let $A = (X, (f_\triangle)_{\triangle \in \tau})$ and $B = (Y, (g_\triangle)_{\triangle \in \tau})$ be τ-algebras. A *homomorphism* from A to B is a map $h \colon X \to Y$ such that

$$h(f_\triangle(x_1, \ldots, x_{\mathrm{ar}(f_\triangle)})) = g_\triangle(h(x_1), \ldots, h(x_{\mathrm{ar}(f_\triangle)}))$$

for all $\triangle \in \tau$ and elements $x_1, \ldots, x_{\mathrm{ar}(f_\triangle)} \in X$.

Remark. If A is a τ-algebra, then $[\![\cdot]\!]^A$ is the unique homomorphism from $\mathcal{F}_\tau$ to A (cf. [24]). If $[\![\cdot]\!]^A$ is injective, then there are no non-trivial logical equivalences.[1] If $[\![\cdot]\!]^A$ is surjective, then $\mathcal{F}_\tau$ is expressively complete, i.e., every property is definable by a formula.

Note that *signature* in logic often means "first-order language". However, this is crucially different from the signature of an algebra. Our definition, where everything

[1] If there is at least one atom and any Boolean connective, then $[\![\cdot]\!]^A$ is not injective.

built from τ is a formula, is completely obliviously of concepts like terms or variables. To interpret first-order logic algebraically, we require a new atom in τ for each atomic FO-formula. Needless to say, there are more sophisticated approaches to *algebraize* first-order logic (see, e.g., the very good introduction by Andréka et al. [5]), but this is not necessary for our purposes.

Before continuing with definitions, we present an exhaustive list of how connectives of PL, ML and FO are defined algebraically.

Example 3.8 (Algebraization of syntax)**.** We list the signatures of propositional, modal and first-order (team) logic.

- The signature of PL is $\tau_{\mathsf{PL}} := \tau_{\mathrm{Bool}} \cup \mathsf{Prop}$, where $\tau_{\mathrm{Bool}} := \{\wedge, \vee, \neg, \top, \bot\}$. Note that every proposition $p \in \mathsf{Prop}$ is an atom.
- The signature of QPL is $\tau_{\mathsf{QPL}} := \tau_{\mathsf{PL}} \cup \{\exists p, \forall p \mid p \in \mathsf{Prop}\}$. In other words, the quantifiers translate to infinitely many unary connectives, one for each p.
- The signature of ML is $\tau_{\mathsf{ML}} := \tau_{\mathsf{PL}} \cup \{\Diamond, \Box\}$.
- The signature of σ-FO is $\tau_{\sigma\text{-}\mathsf{FO}} := \tau_{\mathrm{Bool}} \cup \{\exists x, \forall x \mid x \in \mathsf{Var}\} \cup \mathrm{Atom}_\sigma$, where $\mathrm{Atom}_\sigma := \{\alpha \in \sigma\text{-}\mathsf{FO} \mid \alpha \text{ atomic }\}$. For example, $0 = 1$ is an atom if 0 and 1 are σ-terms.
- The signature of the corresponding *team logic* $\mathcal{L}(\sim)$ (without dependency atoms), with $\mathcal{L}$ being PL, QPL, ML or σ-FO, is then $\tau_{\mathcal{L}(\sim)} := \tau_{\mathcal{L}} \cup \{\sim\}$.

Note that the first-order dependence atom $\mathsf{dep}(t_1, \ldots, t_n; t_{n+1})$ is atomic and hence a *nullary* connective, while the truth-functional atom $\mathsf{dep}(\alpha_1, \ldots, \alpha_n; \alpha_{n+1})$ can be defined as an $(n+1)$-ary connective.

To algebraize the semantics of the above logics, we are interested in algebras with power sets as carriers. This is because a *property* P usually refers to a collection of states, assignments, etc. For example, in modal logic, $\Diamond$ is often defined as

$$(\mathcal{K}, w) \vDash \Diamond\varphi \Leftrightarrow \exists v \in Rw : (\mathcal{K}, v) \vDash \varphi$$

for a Kripke structure $\mathcal{K} = (W, R, V)$. Algebraically, this definition now becomes

$$\Diamond(U) := \{ w \in W \mid Rw \cap U \neq \emptyset \}.$$

Intuitively, if $U \subseteq W$ is the set of worlds where a formula φ is true, then $\Diamond(U)$ is the set of worlds where $\Diamond\varphi$ is true.

Example 3.9 (Algebraization of classical semantics)**.** Let $\tau_{\mathsf{PL}}, \tau_{\mathsf{QPL}}, \tau_{\mathsf{ML}}$ and $\tau_{\sigma\text{-}\mathsf{FO}}$ be as in Example 3.8.

- PL is evaluated on the set $\mathsf{Prop} \to \{0, 1\}$ of propositional assignments. This corresponds to the unique τ_{PL}-algebra A with carrier $\wp(\mathsf{Prop} \to \{0, 1\})$ and
  - $\wedge^A(U_1, U_2) = U_1 \cap U_2$,

  - $\vee^A(U_1, U_2) = U_1 \cup U_2$,
  - $\neg^A(U) = (\mathsf{Prop} \to \{0,1\}) \setminus U$,
  - $\top^A = \mathsf{Prop} \to \{0,1\}$,
  - $\bot^A = \emptyset$,
  - $p^A = \{s \colon \mathsf{Prop} \to \{0,1\} \mid s(p) = 1\}$ for $p \in \mathsf{Prop}$.

- For QPL, we add for each $p \in \mathsf{Prop}$
  - $\exists p^A(U) = \{s \colon \mathsf{Prop} \to \{0,1\} \mid \{s_0^p, s_1^p\} \cap U \neq \emptyset\}$,
  - $\forall p^A(U) = \{s \colon \mathsf{Prop} \to \{0,1\} \mid \{s_0^p, s_1^p\} \subseteq U\}$.

- ML is evaluated on Kripke structures $\mathcal{K} = (W, R, V)$. The τ_{ML}-algebra A induced by $\mathcal{K}$ has carrier $\wp W$ and
  - $\wedge, \vee, \neg, \top$ and $\bot$ are defined analogously to PL,
  - $p^A = V(p)$ for $p \in \mathsf{Prop}$,
  - $\Diamond^A(U) = \{w \in W \mid Rw \cap U \neq \emptyset\}$,
  - $\Box^A(U) = \{w \in W \mid Rw \subseteq U\}$.

- Finally, let $\mathcal{A}$ be a σ-structure for a first-order language σ. The semantic units in first-order logic are, as for propositional logic, *assignments*, hence the $\tau_{\sigma\text{-}\mathsf{FO}}$-algebra A corresponding to $\mathcal{A}$ has the carrier $\wp(\mathsf{Var} \to \mathcal{A})$. Besides the Boolean connectives, we now have
  - $\alpha^A = \{s \colon \mathsf{Var} \to \mathcal{A} \mid (\mathcal{A}, s) \vDash \alpha\}$ for atomic formulas α,
  - $\exists x^A(U) = \{s \colon \mathsf{Var} \to \mathcal{A} \mid \{s_a^x \mid a \in \mathcal{A}\} \cap U \neq \emptyset\}$ for $x \in \mathsf{Var}$,
  - $\forall x^A(U) = \{s \colon \mathsf{Var} \to \mathcal{A} \mid \{s_a^x \mid a \in \mathcal{A}\} \subseteq U\}$ for $x \in \mathsf{Var}$.

Note that the carrier of an algebra is not determined by the encoded structure, but can also depend on the specific semantics. For example, LTL-formulas are modeled by Kripke structures $\mathcal{K} = (W, R, V)$, but the corresponding carrier is the set of sets of infinite *traces* in $\mathcal{K}$, $\wp(R^\omega)$, and not $\wp W$ as in modal logics.

Also note that for the sake of simplicity we deviate from the definitions in Chapter 2 and w.l.o.g. assume that the domain of an first-order assignment is always Var, and that every Kripke structure is over the set Prop of all propositions.

Team semantics

Logics with team semantics enjoy closure properties such as *flatness*, *union closure* or *downward closure*. To be able to talk about these properties, we require carriers that are *double* power sets.

Definition 3.10 (Team algebra). A τ-*team algebra* is a τ-algebra with carrier $\wp\wp X$, where X is a set.

Example 3.11 (Algebraization of team semantics)**.** Under team semantics, the connectives can be algebraically defined as follows.

- In PL, we have the unique τ-team algebra A with carrier $\wp\wp(\mathsf{Prop} \to \{0,1\})$ and
  - $\wedge^A(\mathcal{T}_1, \mathcal{T}_2) = \mathcal{T}_1 \cap \mathcal{T}_2$,
  - $\vee^A(\mathcal{T}_1, \mathcal{T}_2) = \{T_1 \cup T_2 \mid (T_1, T_2) \in \mathcal{T}_1 \times \mathcal{T}_2\}$ (lax semantics),
  - $\dot{\vee}^A(\mathcal{T}_1, \mathcal{T}_2) = \{T_1 \cup T_2 \mid (T_1, T_2) \in \mathcal{T}_1 \times \mathcal{T}_2 \text{ and } T_1 \cap T_2 = \emptyset\}$ (strict semantics),
  - $\neg^A(\mathcal{T}) = \{T \subseteq \mathsf{Prop} \to \{0,1\} \mid \forall s \in T : \{s\} \notin \mathcal{T}\}$,
  - $\top^A = \wp(\mathsf{Prop} \to \{0,1\})$,
  - $\bot^A = \{\emptyset\}$,
  - $p^A = \wp\{s \in \mathsf{Prop} \to \{0,1\} \mid s(p) = 1\}$.

  For QPL, the quantifiers become
  - $\exists p^A(\mathcal{T}) = \{T \mid \exists f \colon T \to \wp^+(\{0,1\}) : T^p_f \in \mathcal{T}\}$ (lax semantics),
  - $\dot{\exists} p^A(\mathcal{T}) = \{T \mid \exists f \colon T \to \wp^1(\{0,1\}) : T^p_f \in \mathcal{T}\}$ (strict semantics),
  - $\forall p^A(\mathcal{T}) = \{T \mid T^p_{\{0,1\}} \in \mathcal{T}\}$.
- For ML, the team algebra A corresponding to a Kripke structure (W, R, V) has carrier $\wp\wp W$, and
  - $\wedge, \vee, \neg, \top$ and $\bot$ are analogous to PL,
  - $p^A = \wp V(p)$,
  - $\Diamond^A(\mathcal{T}) = \{T \mid \exists f \in \prod_{w \in T} \wp^+(Rw) : \bigcup_{w \in T} f(w) \in \mathcal{T}\}$ (lax semantics),
  - $\dot{\Diamond}^A(\mathcal{T}) = \{T \mid \exists f \in \prod_{w \in T} \wp^1(Rw) : \bigcup_{w \in T} f(w) \in \mathcal{T}\}$ (strict semantics),
  - $\Box^A(\mathcal{T}) = \{T \mid RT \in \mathcal{T}\}$.
- Finally, for σ-FO, given a first-order structure $\mathcal{A}$, the corresponding team algebra A has carrier $\wp\wp(\mathsf{Var} \to \mathcal{A})$, and
  - $\wedge, \vee, \neg, \top$ and $\bot$ are again analogous to before,
  - $\alpha^A = \wp\{s \colon \mathsf{Var} \to \mathcal{A} \mid (\mathcal{A}, s) \vDash \alpha\}$, if $\alpha \in \sigma$-FO is atomic,
  - $\exists x^A(\mathcal{T}) = \{T \mid \exists f \colon T \to \wp^+(A) : T^x_f \in \mathcal{T}\}$ (lax semantics),
  - $\dot{\exists} x^A(\mathcal{T}) = \{T \mid \exists f \colon T \to \wp^1(A) : T^x_f \in \mathcal{T}\}$ (strict semantics),
  - $\forall x^A(\mathcal{T}) = \{T \mid T^x_{\mathcal{A}} \in \mathcal{T}\}$.

Observe that we use $\neg$ as a logical primitive instead of considering $\neg\alpha$ as some formula in negation normal form, already for the simple reason that negation normal forms do not necessarily exist in arbitrary τ-algebras.

It is straightforward to algebraize other team-logical connectives, for example

- $\sim\mathcal{T} = (\wp\wp X) \setminus \mathcal{T}$ for the contradictory negation, where $\wp\wp X$ is the carrier,

- $\mathsf{dep}(t_1, \ldots, t_n; t_{n+1}, \ldots, t_{n+m})^A = \{T \mid \forall s, s' \in T : (\forall i \in [n] : t_i\langle s\rangle = t_i\langle s'\rangle) \Rightarrow (\forall i \in [m] : t_{n+i}\langle s\rangle = t_{n+i}\langle s'\rangle)\}$ for the first-order dependence atom.

Next, we algebraically define the common closure properties of team logic. Fix a team algebra A and let $\mathcal{T} \in A$ be a property, i.e., a set of teams.

Definition 3.12 (Coherence)**.** $\mathcal{T}$ is k-*coherent*, for $k \in \mathbb{N}$, if for all teams $T \in A$ it holds that $T \in \mathcal{T}$ if and only if $T' \in \mathcal{T}$ for all subteams $T' \subseteq T$ of size $|T'| = k$.

Definition 3.13 (Flatness)**.** $\mathcal{T}$ is *flat* if it is 1-coherent.

Definition 3.14 (Closure properties)**.** $\mathcal{T}$ is *union closed* if, for any set $\mathcal{T}' \subseteq \mathcal{T}$ of teams, $\bigcup \mathcal{T}' \in \mathcal{T}$. $\mathcal{T}$ is *downward closed* if $T \in \mathcal{T}$ and $T' \subseteq T$ implies $T' \in \mathcal{T}$.

A formula φ is called k-coherent, flat, union closed or downward closed (w. r. t. A) if $[\![\varphi]\!]^A$ is.

Proposition 3.15. *$\mathcal{T}$ is flat if and only if it is union closed and downward closed.*

Proof. For "$\Leftarrow$", if $\mathcal{T}$ is downward closed, then $T \in \mathcal{T}$ implies $\{s\} \in \mathcal{T}$ for all $s \in T$, and if it is union closed, then $\{s\} \in \mathcal{T}$ for all $s \in T$ implies $T \in \mathcal{T}$. Hence $\mathcal{T}$ is then flat.

For "$\Rightarrow$", if $\mathcal{T}$ is flat, we show that the other two properties hold. If $\mathcal{T}' \subseteq \mathcal{T}$, then $\{s\} \in \mathcal{T}$ for all $s \in T \in \mathcal{T}'$; so in particular $\{s\} \in \mathcal{T}$ for all $s \in \bigcup \mathcal{T}'$. But then $\bigcup \mathcal{T}' \in \mathcal{T}$ by flatness. Likewise, if $T \in \mathcal{T}$ and $T' \subseteq T$, then $\forall s \in T : \{s\} \in \mathcal{T}$, so in particular $\forall s \in T' : \{s\} \in \mathcal{T}$. Consequently, $T' \in \mathcal{T}$ again by flatness. □

Corollary 3.16. *$\mathcal{T}$ is flat iff $\mathcal{T} = \wp \bigcup \mathcal{T}$ iff $\mathcal{T}$ is a power set.*

The flatness of formulas is usually proved by induction on the syntax, exploiting that each connective in a sense "preserves flatness." This can be made formal as follows.

Definition 3.17 (Flatness preserving)**.** An operation $\triangle \colon A^r \to A$ *preserves flatness* in a team algebra A if $\triangle(\mathcal{T}_1, \ldots, \mathcal{T}_r)$ is flat whenever $\mathcal{T}_1, \ldots, \mathcal{T}_r \in A$ are flat properties.

The team-logical variants of the common classical connectives preserve flatness.

Proposition 3.18. *The connectives $\neg$, $\wedge$, $\vee$, $\dot\vee$, $\Box$, $\Diamond$, $\dot\Diamond$, $\forall x$, $\exists x$ and $\dot\exists x$ preserve flatness.*

The above is seen as part of typical inductive proofs of flatness in the literature, so we will not include a proof here. However, we prove a stronger statement (Corollary 3.35) later in this chapter.

Proposition 3.19. *Let $A = (\wp\wp X, (f_\triangle)_{\triangle \in \tau})$ be a τ-team algebra. If a τ-formula φ contains only connectives $\triangle \in \tau$ such that $f_\triangle$ preserves flatness, then $[\![\varphi]\!]^A$ is flat.*

Proof. Straightforward by induction on the syntax of φ. □

Often it seems natural to define a connective in a flatness preserving manner, such as the above, but in other cases this is neither obvious nor desirable. For instance, Krebs et al. [91] defined two team semantics for LTL, a *synchronous* and an *asynchronous* one. Only the asynchronous connectives preserve flatness, but some properties can only be defined synchronously.

Another example is the work of Grädel and Hegselmann [46] on first-order logic with counting. They proposed two counting mechanisms for team semantics: *counting quantifiers* $\exists^{\geqslant\mu}x$ as flatness preserving connectives on teams, and *forking atoms* $\vec{u} \triangleleft^{\geqslant\mu} \vec{v}$, which are non-flat atoms.

3.1.1 Duality

The idea of *duality* of connectives is tightly connected to negation. In team logic, there are two distinct notions of duality, each induced by its own negation. We refer to duality w. r. t. $\neg$ as *weak* and to duality w. r. t. $\sim$ as *strong*. [1]

Definition 3.20 (Complement)**.** Let A be a τ-algebra with carrier $\wp X$. The *complement* of a property $\mathcal{P} \subseteq X$ is the property $\mathcal{P}^{\complement} := X \setminus \mathcal{P}$.

Definition 3.21 (Strong dual)**.** Let A be a τ-algebra with carrier $\wp X$ and $\triangle \colon (\wp X)^r \to \wp X$. The *strong dual* $\triangle^{\complement}$ of $\triangle$ is defined by $\triangle^{\complement}(U_1, \ldots, U_r) := \triangle(U_1^{\complement}, \ldots, U_r^{\complement})^{\complement}$ for all $U_1, \ldots, U_r \subseteq X$.

Proposition 3.22 (Symmetry)**.** $\nabla = \triangle^{\complement}$ *if and only if* $\triangle = \nabla^{\complement}$.

Proof. Clear. □

We proceed with the second type of duality, *weak duality*, which exists only in team semantics. Fix a τ-team algebra $A = (\wp\wp X, (f_\triangle)_{\triangle \in \tau})$. Symbols $\triangle, \nabla, \ldots$ will denote operations on A with some arity $r \geqslant 0$.

Recall the definition of $\neg$ as a logical primitive in Chapter 2:

$$(\mathcal{A}, T) \vDash \neg\varphi \quad \Leftrightarrow \quad \forall s \in T : (\mathcal{A}, \{s\}) \nvDash \varphi.$$

Accordingly, the algebraic definition is:

$$\neg\mathcal{T} := \{\, T \in A \mid \forall s \in T : \{s\} \notin \mathcal{T} \,\}$$

For flat properties $\mathcal{T}$, $\neg\mathcal{T}$ is simple to compute:

Proposition 3.23. *Let* $\mathcal{T} \in A$ *be flat. Then* $\neg\mathcal{T} = \wp(X \setminus \bigcup \mathcal{T})$.

[1] It is not easy to come up with sensible and intuitive names for the two kinds of negation. In the literature, $\neg$ is called *game negation*, *dual negation*, *intuitionistic negation*, or *strong negation*, while we call it *weak* negation. The connective $\sim$ is called *classical negation*, *contradictory negation*, *Boolean negation* or *weak negation*, while we call it *strong negation*. (So only the non-classical negation is allowed to appear in classical formulas.) We simply call $\neg$ weak and $\sim$ strong because, unlike $\sim$, $\neg$ does not allow to define one connective in terms of its dual, as we show at the end of the section.

Proof. Due to Corollary 3.16, for the team $T = \bigcup \mathcal{T}$ we have $\mathcal{T} = \wp T$. Note that $s \in T$ iff $\{s\} \subseteq T$ iff $\{s\} \in \wp T$. Hence

$$\begin{aligned} \neg\mathcal{T} &= \{ T' \mid \forall s \in T' : \{s\} \notin \mathcal{T} \} && \text{(def. } \neg) \\ &= \{ T' \mid \forall s \in T' : \{s\} \notin \wp T \} \\ &= \{ T' \mid \forall s \in T' : s \notin T \} \\ &= \{ T' \mid T' \subseteq X \setminus T \} \\ &= \wp(X \setminus T) \end{aligned}$$

□

Proposition 3.24. *For each $\mathcal{T} \in A$, $\neg\mathcal{T}$ is flat.*

Proof. The condition $\forall s \in T : \{s\} \notin \mathcal{T}$ is downward and union closed w. r. t. T. □

Corollary 3.25. *The connective $\neg$ preserves flatness, and $\neg\neg\mathcal{T} = \mathcal{T}$ for all flat $\mathcal{T}$.*

Next, we proceed by defining weak duality. Note that for it we consider only singletons. We come to the reason for that later.

Definition 3.26 (Weak duality)**.** Let $\triangle, \nabla \colon (\wp\wp X)^r \to \wp\wp X$. Then ∇ is a *weak dual* of $\triangle$ if

$$\{s\} \in \triangle(\mathcal{T}_1, \ldots, \mathcal{T}_r) \Leftrightarrow \{s\} \notin \nabla(\neg\mathcal{T}_1, \ldots, \neg\mathcal{T}_r)$$

for all $s \in X$ and flat $\mathcal{T}_1, \ldots, \mathcal{T}_r \in A$.

Proposition 3.27 (Symmetry)**.** *If ∇ is a weak dual of $\triangle$, then $\triangle$ is a weak dual of ∇.*

Proof. For all flat properties $\mathcal{T}_1, \ldots, \mathcal{T}_r \in A$ and $s \in X$,

$$\begin{aligned} & \{s\} \in \nabla(\mathcal{T}_1, \ldots, \mathcal{T}_r) \\ \Leftrightarrow\ & \{s\} \in \nabla(\neg\neg\mathcal{T}_1, \ldots, \neg\neg\mathcal{T}_r) && \text{(Corollary 3.25)} \\ \Leftrightarrow\ & \{s\} \notin \triangle(\neg\mathcal{T}_1, \ldots, \neg\mathcal{T}_r) && \text{(by assumption)} \end{aligned}$$

□

Proposition 3.28. *Every operation $\triangle$ has a weak dual.*

Proof. We apply symmetry and show that $\triangle$ is the weak dual of

$$\nabla(\mathcal{T}_1, \ldots, \mathcal{T}_r) := \{ \{s\} \in \wp X \mid \{s\} \notin \triangle(\neg\mathcal{T}_1, \ldots, \neg\mathcal{T}_r) \}.$$

As clearly $\{s\} \in \nabla(\mathcal{T}_1, \ldots, \mathcal{T}_r)$ iff $\{s\} \notin \triangle(\neg\mathcal{T}_1, \ldots, \neg\mathcal{T}_r)$, the proposition follows. □

Above, we used the fact that weak duality is defined only w. r. t. *flat* arguments $\mathcal{T}_1, \ldots, \mathcal{T}_r$. Without this, we could construct operations $\triangle$ with no weak dual. By Cantor's theorem, $|A| = |\wp\wp X| > |\wp X| = |\{\mathcal{T} \in A \mid \mathcal{T} \text{ flat}\}| \geqslant |\{\neg\mathcal{T} \mid \mathcal{T} \in A\}|$, so $\neg$ cannot be injective. This means there are properties $\mathcal{T} \neq \mathcal{T}'$ with $\neg\mathcal{T} = \neg\mathcal{T}'$, and Definition 3.26 would be impossible to satisfy. On flat properties $\mathcal{T}$ however, $\neg$ is a bijection (Proposition 3.23), so this issue cannot occur.

The observant reader may also wonder why weak duality is defined only on the level of singletons. Another natural definition of duality would be the stronger condition

$$\triangle(\mathcal{T}_1, \ldots, \mathcal{T}_r) = \neg\nabla(\neg\mathcal{T}_1, \ldots, \neg\mathcal{T}_r).$$

However, the weaker one is easier to prove and suffices for our purposes, since both definitions are equivalent for flatness preserving connectives:

Proposition 3.29. *If ∇ and $\triangle$ are flatness preserving and weak duals of each other, then*

$$\neg\triangle(\mathcal{T}_1, \ldots, \mathcal{T}_r) = \nabla(\neg\mathcal{T}_1, \ldots, \neg\mathcal{T}_r)$$

and

$$\triangle(\mathcal{T}_1, \ldots, \mathcal{T}_r) = \neg\nabla(\neg\mathcal{T}_1, \ldots, \neg\mathcal{T}_r)$$

for all flat $\mathcal{T}_1, \ldots, \mathcal{T}_r \in \mathcal{A}$.

Proof. First observe that $\neg\triangle(\mathcal{T}_1, \ldots, \mathcal{T}_r)$ and $\nabla(\neg\mathcal{T}_1, \ldots, \neg\mathcal{T}_r)$ are flat by Proposition 3.24 and since ∇ preserves flatness. Then for all teams T,

$$\begin{aligned}
& T \in \neg\triangle(\mathcal{T}_1, \ldots, \mathcal{T}_r) \\
\Leftrightarrow\ & \forall s \in T : \{s\} \notin \triangle(\mathcal{T}_1, \ldots, \mathcal{T}_r) && \text{(def. } \neg\text{)} \\
\Leftrightarrow\ & \forall s \in T : \{s\} \in \nabla(\neg\mathcal{T}_1, \ldots, \neg\mathcal{T}_r) && \text{(def. weak dual)} \\
\Leftrightarrow\ & T \in \nabla(\neg\mathcal{T}_1, \ldots, \neg\mathcal{T}_r). && \text{(flatness)}
\end{aligned}$$

The other equivalence is analogous. □

Observe that any operation $\triangle$ has exactly one strong dual, but may have multiple weak duals. For instance, $\wedge$ has the strong dual $\otimes$ but (among others) the weak duals $\vee$ and $\dot\vee$. The same holds for $\Box$ and $\Diamond/\dot\Diamond$, or $\forall x$ and $\exists x/\dot\exists x$.

The terminology "weak" is not completely unjustified: Since weak duals are not unique, and $\neg\mathcal{T}$ is always flat, $\neg$ cannot be used to define a connective by its weak dual. While $\wedge$ and $\vee$ are strong duals and thus interdefinable in classical logic, in team semantics they become weak duals and are not definable in terms of each other, and for this reason both need to be included as primitives in team logic. The same holds for the pairs $\Diamond/\Box$ and $\exists/\forall$.

3.2 Teamification

Team semantical connectives such as $\wedge, \vee, \forall, \exists, \Box, \Diamond$ are carefully defined in the literature in order to preserve flatness. Now imagine we are given some logical system and want to build a "faithful" team semantics for it from scratch:

> **Given:** An arbitrary logical connective $\triangle$
> **Question:** What is a "natural" connective ∇ that satisfies
> $$T \vDash \nabla(\alpha_1, \ldots, \alpha_r) \;\Leftrightarrow\; \forall s \in T : s \vDash \triangle(\alpha_1, \ldots, \alpha_r) \text{ for all flat } \alpha_1, \ldots, \alpha_r?$$

This is not trivial, and connectives may have multiple team-semantical, seemingly equally valid alternatives (think of $\exists$ and $\dot{\exists}$, where $\dot{\exists}$ was introduced first [135]).

In this section, we propose a rule called *teamification*[1] to lift classical operations to flatness preserving team semantics. We will show that the standard connectives obey it, and prove several abstract results based on it.

The fundamental idea is that the power set operator $\wp$ is a natural *homomorphism* between the corresponding algebras.

Definition 3.30 (Teamification)**.** An operation $\nabla\colon (\wp\wp X)^r \to \wp\wp X$ is a *teamification* of an operation $\triangle\colon (\wp X)^r \to \wp X$ if

$$\wp\triangle(T_1, \ldots, T_r) = \nabla(\wp T_1, \ldots, \wp T_r)$$

for all $T_1, \ldots, T_r \subseteq X$.

Definition 3.31. A τ-team algebra $B = (\wp\wp X, (g_\triangle)_{\triangle\in\tau})$ is a *teamification* of a τ-algebra $A = (\wp X, (f_\triangle)_{\triangle\in\tau})$ if $g_\triangle$ is a teamification of $f_\triangle$ for all $\triangle \in \tau$, or equivalently, if $\wp$ is an algebra homomorphism from A to B.

Proposition 3.32. (1) *Every operation has multiple teamifications.*

(2) *There are operations that are not the teamification of any operation.*

(3) *If ∇ is the teamification of some operation $\triangle$, then that $\triangle$ is unique.*

Proof. For (1), simply defining $\nabla(\wp T_1, \ldots, \wp T_r) := \wp\triangle(T_1, \ldots, T_r)$ yields a teamification, with ∇ acting arbitrarily on arguments that are not power sets. As $\emptyset$ is not a power set, $\nabla(\emptyset, \ldots, \emptyset)$ can take for instance the distinct values $\emptyset$ (the set of no teams) and $\{\emptyset\}$ (the set containing only the empty team), which yields two distinct teamifications.

For (2), if $\nabla(\wp T_1, \ldots, \wp T_r) = \emptyset$ then ∇ cannot be a teamification.

For (3), suppose that ∇ is a teamification of $\triangle$ and $\triangle'$. Then

$$\wp\triangle(T_1, \ldots, T_r) = \nabla(\wp T_1, \ldots, \wp T_r) = \wp\triangle'(T_1, \ldots, T_r)$$

for all $T_1, \ldots, T_r \in \wp X$. But $\wp$ is injective, so $\triangle = \triangle'$. ☐

Next, we show that the standard team-logical connectives are teamifications, that is, $\wp$ is a homomorphism from the corresponding classical semantics to the team semantics. Recall that the usual atomic formulas are represented as nullary connectives.

[1] The term stems from a recent paper by Galliani [39], where he referred, without formal definition, to a process of "lifting" classical compositional semantics to team semantics.

Theorem 3.33. *The operations* $\top, \bot, \wedge, \vee, \dot\vee, \neg, \Diamond, \dot\Diamond, \Box, \exists x, \dot\exists x, \forall x$, $p \in \mathrm{Prop}$, *and all atomic first-order formulas are teamifications of the corresponding classical connectives.*

Proof. While we usually use the same symbol for both, in this proof for the sake of readability the classical operations will be denoted by $\triangle_c$ and the team-semantical counterparts by $\triangle_t$. We show for all operations that $\triangle_t$ is a teamification of $\triangle_c$, i.e.,

$$\wp\triangle_c(T_1, \ldots, T_r) = \triangle_t(\wp T_1, \ldots, \wp T_r)$$

for all teams $T_1, \ldots, T_r$. Let us start with atoms. Since all the mentioned atoms obey flatness,

$$T \in \triangle_t \Leftrightarrow \forall s \in T : s \in \triangle_c \Leftrightarrow T \subseteq \triangle_c \Leftrightarrow T \in \wp\triangle_c.$$

In other words, $\wp\triangle_c = \triangle_t$. As an example, we have $\bot_t = \{\emptyset\} = \wp\emptyset = \wp\bot_c$. We proceed with the Boolean connectives $\wedge, \vee, \neg$.

- For conjunction, we have to show $\wedge_t(\wp T_1, \wp T_2) = \wp\wedge_c(T_1, T_2)$.

$$\begin{aligned}
& T' \in \wedge_t(\wp T_1, \wp T_2) \\
\Leftrightarrow\ & T' \in \wp T_1 \cap \wp T_2 && \text{(team semantics of } \wedge) \\
\Leftrightarrow\ & T' \subseteq T_1 \text{ and } T' \subseteq T_2 \\
\Leftrightarrow\ & T' \subseteq T_1 \cap T_2 \\
\Leftrightarrow\ & T' \subseteq \wedge_c(T_1, T_2) && \text{(classical semantics of } \wedge) \\
\Leftrightarrow\ & T' \in \wp\wedge_c(T_1, T_2).
\end{aligned}$$

- For disjunction, we have to show $\vee_t(\wp T_1, \wp T_2) = \wp\vee_c(T_1, T_2)$ (which was already observed by Abramsky and Väänänen [2]).

$$\begin{aligned}
& T' \in \vee_t(\wp T_1, \wp T_2) \\
\Leftrightarrow\ & \exists V_1 \in \wp T_1, V_2 \in \wp T_2 : T' = V_1 \cup V_2 && \text{(team semantics of } \vee) \\
\Leftrightarrow\ & \exists V_1 \subseteq T_1, V_2 \subseteq T_2 : T' = V_1 \cup V_2 \\
\Leftrightarrow\ & T' \subseteq T_1 \cup T_2 \\
\Leftrightarrow\ & T' \subseteq \vee_c(T_1, T_2) && \text{(classical semantics of } \vee) \\
\Leftrightarrow\ & T' \in \wp\vee_c(T_1, T_2).
\end{aligned}$$

  For strict disjunction $\dot\vee$, the union $V_1 \cup V_2$ in the second and third line has to be disjoint, but then the fourth line is still equivalent.

- For negation, we have to show $\neg_t(\wp T) = \wp\neg_c(T)$.

$$\begin{aligned}
& T' \in \neg_t(\wp T) \\
\Leftrightarrow\ & \forall s \in T' : \{s\} \notin \wp T && \text{(team semantics of } \neg) \\
\Leftrightarrow\ & \forall s \in T' : s \notin T
\end{aligned}$$

$$\Leftrightarrow T' \subseteq X \setminus T$$
$$\Leftrightarrow T' \subseteq \neg_c(T) \qquad \text{(classical semantics of } \neg)$$
$$\Leftrightarrow T' \in \wp\neg_c(T).$$

Next, we consider modal logic.

- For diamond, we show $\Diamond_t(\wp T) = \wp\Diamond_c(T)$.

$$T' \in \Diamond_t(\wp T)$$
$$\Leftrightarrow \exists S \in \wp T : S \text{ successor team of } T' \qquad \text{(team semantics of } \Diamond)$$
$$\Leftrightarrow \exists S \subseteq T : \exists f \in \prod_{w \in T'} \wp^+(Rw) : S = \bigcup_{w \in T'} f(w)$$

For "⇒", the existence of f clearly implies the weaker statement below. For "⇐", to construct f and hence S we apply the axiom of choice to pick an element from each of the non-empty sets $Rw \cap T$.

$$\Leftrightarrow \forall w \in T' : Rw \cap T \neq \emptyset$$
$$\Leftrightarrow \forall w \in T' : w \in \Diamond_c(T) \qquad \text{(classical semantics of } \Diamond)$$
$$\Leftrightarrow T' \in \wp\Diamond_c(T)$$

For strict semantics, f additionally is a function from $\prod_{w \in T'} \wp^1(Rw)$. The above argument for the construction of S in the "⇐" direction however yields precisely such a function.

- For box, we show $\Box_t(\wp T) = \wp\Box_c(T)$.

$$T' \in \Box_t(\wp T)$$
$$\Leftrightarrow RT' \in \wp T \qquad \text{(team semantics of } \Box)$$
$$\Leftrightarrow RT' \subseteq T$$
$$\Leftrightarrow \forall s \in T' : Rs \subseteq T$$
$$\Leftrightarrow \forall s \in T' : s \in \Box_c(T) \qquad \text{(classical semantics of } \Box)$$
$$\Leftrightarrow T' \subseteq \Box_c(T)$$
$$\Leftrightarrow T' \in \wp\Box_c(T)$$

Finally, the quantifiers behave analogously to the modal connectives.

- Existential quantification:

$$T' \in \exists x_t(\wp T)$$
$$\Leftrightarrow \exists S \in \wp T : S \text{ supplementing team of } T' \qquad \text{(team semantics of } \exists x)$$
$$\Leftrightarrow \exists f \in T' \to \wp^+(\mathcal{A}) : (T')^x_f \subseteq T$$

$$\begin{aligned}
&\Leftrightarrow \forall s \in T' : \exists a \in \mathcal{A} : s^x_a \in T \\
&\Leftrightarrow \forall s \in T' : s \in \exists x_c(T) && \text{(classical semantics of } \exists x) \\
&\Leftrightarrow T' \subseteq \exists x_c(T) \\
&\Leftrightarrow T' \in \wp\exists x_c(T)
\end{aligned}$$

For strict semantics, $f(s)$ must be a singleton for each s. This is handled like for the strict $\Diamond$.

- Universal quantification:

$$\begin{aligned}
& T' \in \forall x_t(\wp T) \\
&\Leftrightarrow (T')^x_{\mathcal{A}} \in \wp T && \text{(team semantics of } \forall x) \\
&\Leftrightarrow (T')^x_{\mathcal{A}} \subseteq T \\
&\Leftrightarrow \forall s \in T' : \forall a \in \mathcal{A} : s^x_a \in T && \text{(def. of duplicating team)} \\
&\Leftrightarrow \forall s \in T' : s \in \forall x_c(T) && \text{(classical semantics of } \forall x) \\
&\Leftrightarrow T' \subseteq \forall x_c(T) \\
&\Leftrightarrow T' \in \wp\forall x_c(T)
\end{aligned}$$ □

The next result is a central characterization of teamification.

Theorem 3.34. *A map $\nabla \colon (\wp\wp X)^r \to \wp\wp X$ is a teamification if and only if it preserves flatness.*

Proof. "$\Rightarrow$": Let $\nabla \colon (\wp\wp X)^r \to \wp\wp X$ be the teamification of $\triangle$, where $\triangle \colon (\wp X)^r \to \wp X$, and let $\mathcal{T}_1, \ldots, \mathcal{T}_r \subseteq \wp X$ be flat properties. By Corollary 3.16, $\mathcal{T}_i = \wp \bigcup \mathcal{T}_i$. By definition of teamification, then $\nabla(\wp \bigcup \mathcal{T}_1, \ldots, \wp \bigcup \mathcal{T}_r) = \wp\triangle(\bigcup \mathcal{T}_1, \ldots, \bigcup \mathcal{T}_r)$, which is a power set and hence flat.

"$\Leftarrow$": Let $\nabla \colon (\wp\wp X)^r \to \wp\wp X$ preserve flatness. We define an operation $\triangle \colon (\wp X)^r \to \wp X$ such that, for arbitrary $T_1, \ldots, T_r \subseteq X$, we have $\wp\triangle(T_1, \ldots, T_r) = \nabla(\wp T_1, \ldots, \wp T_r)$. The $\wp T_i$ are flat, so by assumption, $\nabla(\wp T_1, \ldots, \wp T_r)$ is flat. Hence it is also of the form $\wp U$ for some team $U \subseteq X$. Now, to define $\triangle$, we let $\triangle(T_1, \ldots, T_r) := U$. □

This yields an alternative proof of Proposition 3.18 by means of Theorem 3.33:

Corollary 3.35. *The operations $\top$, $\bot$, $\wedge$, $\vee$, $\dot\vee$, $\neg$, $\Diamond$, $\dot\Diamond$, $\Box$, $\exists x$, $\dot\exists x$ and $\forall x$ preserve flatness.*

We return to the original question at the beginning of this section, and as follows relate a semantics and its teamification.

Proposition 3.36. *Let the τ-team algebra B be a teamification of a τ-algebra A. Then, for all τ-formulas φ, $[\![\varphi]\!]^B = \wp[\![\varphi]\!]^A$. Moreover, $T \in [\![\varphi]\!]^B$ iff $\forall s \in T : \{s\} \in [\![\varphi]\!]^B$ iff $\forall s \in T : s \in [\![\varphi]\!]^A$, for all teams T.*

Proof. Note that the second claim follows easily from the first: For all teams T, $T \in [\![\varphi]\!]^B$ iff $T \subseteq [\![\varphi]\!]^A$ iff $\forall s \in T : s \in [\![\varphi]\!]^A$. Setting $T = \{s\}$ furthermore proves the equivalence to $\forall s \in T : \{s\} \in [\![\varphi]\!]^B$.

We prove the first claim by induction on the syntax of φ. Let $\varphi = \triangle(\psi_1, \ldots, \psi_r)$, where $\triangle \in \tau$ and the proposition already holds for the τ-formulas ψ_i. Let $\triangle_A$ or $\triangle_B$ denote the respective operation in the algebra A or B.

$$\begin{aligned}
[\![\varphi]\!]^B &= \triangle_B([\![\psi_1]\!]^B, \ldots, [\![\psi_r]\!]^B) && \text{(def. } [\![\cdot]\!]) \\
&= \triangle_B\left(\wp[\![\psi_1]\!]^A, \ldots, \wp[\![\psi_r]\!]^A\right) && \text{(induction hypothesis)} \\
&= \wp\triangle_A\left([\![\psi_1]\!]^A, \ldots, [\![\psi_r]\!]^A\right) && \text{(def. teamification)} \\
&= \wp[\![\varphi]\!]^A. && \text{(def. } [\![\cdot]\!]) \quad \square
\end{aligned}$$

Teamification also turns strong into weak duality, as is shown below.

Theorem 3.37. *Let $\triangle_1, \triangle_2 \colon (\wp X)^r \to \wp X$ and $\nabla_1, \nabla_2 \colon (\wp\wp X)^r \to \wp\wp X$. For $i \in \{1, 2\}$, let ∇_i be a teamification of $\triangle_i$. Then $\triangle_1$ and $\triangle_2$ are strong duals of each other if and only if ∇_1 and ∇_2 are weak duals of each other.*

Proof. "$\Rightarrow$": Suppose $\triangle_1$ and $\triangle_2$ are strongly dual. Let $\mathcal{T}_1, \ldots, \mathcal{T}_r \subseteq \wp X$ be flat properties and $T_1, \ldots, T_r$ teams such that $\mathcal{T}_i = \wp T_i$. For weak duality, we have to show that $\{s\} \notin \nabla_1(\mathcal{T}_1, \ldots, \mathcal{T}_r) \Leftrightarrow \{s\} \in \nabla_2(\neg\mathcal{T}_1, \ldots, \neg\mathcal{T}_r)$, or equivalently, by Proposition 3.23, that

$$\{s\} \notin \nabla_1(\wp T_1, \ldots, \wp T_r) \Leftrightarrow \{s\} \in \nabla_2(\wp(X \setminus T_1), \ldots, \wp(X \setminus T_r)).$$

This follows by

$$\begin{aligned}
\{s\} \notin \nabla_1(\wp T_1, \ldots, \wp T_r) &\Leftrightarrow \{s\} \notin \wp\triangle_1(T_1, \ldots, T_r) && \text{(def. teamification)} \\
&\Leftrightarrow s \notin \triangle_1(T_1, \ldots, T_r) \\
&\Leftrightarrow s \in \triangle_2(X \setminus T_1, \ldots, X \setminus T_r) && \text{(def. strong duality)} \\
&\Leftrightarrow \{s\} \in \wp\triangle_2(X \setminus T_1, \ldots, X \setminus T_r) \\
&\Leftrightarrow \{s\} \in \nabla_2(\wp(X \setminus T_1), \ldots, \wp(X \setminus T_r)). && \text{(def. teamification)}
\end{aligned}$$

"$\Leftarrow$": Let ∇_1 and ∇_2 be weakly dual and let $T_1, \ldots, T_r \subseteq X$ and $s \in X$ be arbitrary. For strong duality, we have to show that $s \notin \triangle_1(T_1, \ldots, T_r) \Leftrightarrow s \in \triangle_2(X \setminus T_1, \ldots, X \setminus T_r)$:

$$\begin{aligned}
s \notin \triangle_1(T_1, \ldots, T_r) &\Leftrightarrow \{s\} \notin \wp\triangle_1(T_1, \ldots, T_r) \\
&\Leftrightarrow \{s\} \notin \nabla_1(\wp T_1, \ldots, \wp T_r) && \text{(def. teamification)} \\
&\Leftrightarrow \{s\} \in \nabla_2(\neg\wp T_1, \ldots, \neg\wp T_r) && \text{(def. weak duality)} \\
&\Leftrightarrow \{s\} \in \nabla_2(\wp(X \setminus T_1), \ldots, \wp(X \setminus T_r)) && \text{(Proposition 3.23)} \\
&\Leftrightarrow \{s\} \in \wp\triangle_2(X \setminus T_1, \ldots, X \setminus T_r) && \text{(def. teamification)} \\
&\Leftrightarrow s \in \triangle_2(X \setminus T_1, \ldots, X \setminus T_r). && \square
\end{aligned}$$

Based on the results of this section, we return to one question from Chapter 1:

> Is a team semantics necessarily based on a classical logic?

This obviously depends on the meaning of "is based on". At least if we interpret it as being a teamification in the sense of Definition 3.31, we are in the position to give a formal answer.

Theorem 3.38. *Let* B *be a* τ*-team algebra. Then* B *is a teamification of some* τ*-algebra* A *if and only if* $\triangle^B$ *is flatness preserving for all* $\triangle \in \tau$.

Proof. Immediately from Theorem 3.34. □

3.3 Operators

In the algebraic world, a unary *operator* $\triangle$ (in a set algebra) is an operation that is

normal: $\triangle(\emptyset) = \emptyset$

additive: $\triangle(U \cup U') = \triangle(U) \cup \triangle(U')$

monotone: $U \subseteq U' \Rightarrow \triangle(U) \subseteq \triangle(U')$ (which follows from the first two).

Remarkably, many team-logical connectives behave like operators. In the literature, a Boolean algebra augmented with operators is called BAO (*Boolean algebra with operators*) [138]. The most common operators $\triangle$ are those generated by a binary relation R such that $s \in \triangle(U) \Leftrightarrow \exists s' \in U : (s, s') \in R$, such as the classical $\Diamond$ from modal logic.

By the famous Jónsson-Tarski Representation Theorem [77], every BAO is isomorphic to one where the operators are generated in this way. For this reason, in this thesis we restrict ourselves to such operators.[1]

Definition 3.39 (Operator)**.** A map $\triangle\colon (\wp X)^r \to \wp X$ is an r*-ary operator* if there is a relation $R \subseteq X^{r+1}$ *generating* $\triangle$, meaning

$$u \in \triangle(U_1, \dots, U_r) \;\Leftrightarrow\; \exists (u, u_1, \dots, u_r) \in R : u_1 \in U_1, \dots, u_r \in U_r$$

for all $U_1, \dots, U_r \subseteq X$ and $u \in X$.

Our definition meets the above criteria of an operator:

Proposition 3.40. *Let* $\triangle\colon (\wp X)^r \to \wp X$ *be an operator. Then* $\triangle$ *is*

- *additive:* $\bigcup_{U_i \in \mathcal{C}} \triangle(U_1, \dots, U_r) = \triangle(U_1, \dots, U_{i-1}, \bigcup \mathcal{C}, U_{i+1}, \dots, U_r)$ *for all* $\mathcal{C} \subseteq \wp X$,
- *normal:* $\triangle(U_1, \dots, U_{i-1}, \emptyset, U_{i+1}, \dots, U_r) = \emptyset$,
- *monotone:* $\triangle(U_1, \dots, U_r) \subseteq \triangle(U_1, \dots, U_{i-1}, V, U_{i+1}, \dots, U_r)$ *for all* $U_i \subseteq V \subseteq X$,

where $U_1, \dots, U_r \subseteq X$ *and* $i \in [r]$.

Proof. Standard (see, e.g., Givant [42, Theorem 1.2]). □

[1] A generating relation exists if the BAO has *atoms* in the order-theoretic sense. Otherwise, those can be obtained by an ultrafilter construction. For details, see Givant [42] or Venema [138].

As logical laws, these are well-known in, e.g., classical modal logic as $\Diamond\varphi \vee \Diamond\psi \equiv \Diamond(\varphi \vee \psi)$, $\Diamond\bot \equiv \bot$, and $\varphi \vDash \psi \Rightarrow \Diamond\varphi \vDash \Diamond\psi$.

Proposition 3.41. *The relation generating an operator is unique.*

Proof. Suppose that $\triangle\colon (\wp X)^r \to \wp X$ is generated by both R and R'. Then by definition, the following are equivalent for all sets $U_1, \ldots, U_r \subseteq X$ and $u \in X$:

(1) $u \in \triangle(U_1, \ldots, U_r)$

(2) $\exists u_1 \in U_1 \cdots \exists u_r \in U_r : (u, u_1, \ldots, u_r) \in R$

(3) $\exists u_1' \in U_1 \cdots \exists u_r' \in U_r : (u, u_1', \ldots, u_r') \in R'$

This implies $R = R'$ as follows: Suppose $(u_0, u_1, \ldots, u_r) \in R$. Then $u_0 \in \triangle(\{u_1\}, \ldots, \{u_r\})$ by (2) $\Rightarrow$ (1). Hence, $(u_0, u_1, \ldots, u_r) \in R'$ by (1) $\Rightarrow$ (3). This proves $R \subseteq R'$, with the other direction being symmetric. □

The unique $(r+1)$-ary relation generating an r-ary operator $\triangle$ is denoted by $\mathcal{R}_\triangle$. Given an $(r+1)$-ary relation R, every element $u \in X$ induces an r-ary relation,

$$Ru := \{\, (u_1, \ldots, u_r) \mid (u, u_1, \ldots, u_r) \in R \,\}.$$

These can be interpreted as "successor tuples" of u w. r. t. R. Note that in the case $r = 1$ there are only two possible relations Ru, namely $\emptyset$ and $\{\varepsilon\}$ (where ε is the empty tuple).

Observe that the usual semantics of polyadic modal operators (cf. Blackburn and van Benthem [9]),

$$w \vDash \triangle(\varphi_1, \ldots, \varphi_r) \;\Leftrightarrow\; \exists (v_1, \ldots, v_r) \in \mathcal{R}_\triangle w \text{ such that } \forall i \in [r] : v_i \vDash \varphi_i,$$

precisely mirrors Definition 3.39 on the level of formulas.

Example 3.42. Not only $\Diamond$, but also the classical $\wedge$ and $\exists x$ are operators. For instance, $\wedge$ is generated by $\mathcal{R}_\wedge = \{(s, s, s)\}$. In a first-order structure $\mathcal{A}$, the relation $\mathcal{R}_{\exists x}$ is equal to $\approx_x$, where $s \approx_x s'$ if there exists a such that $s' = s^x_a$.

Curiously, in team semantics, a lot more connectives become operators, even those that are classically strong duals of each other.

Theorem 3.43. *In classical semantics, $\wedge$, $\Diamond$ and $\exists x$ are operators, $\vee$, $\Box$ and $\forall x$ are strong duals of operators, and $\neg$ is neither. In team semantics, $\wedge$, $\vee$, $\dot\vee$, $\Box$, $\Diamond$, $\dot\Diamond$, $\forall x$, $\exists x$, $\dot\exists x$ are operators, while $\neg$ and $\sim$ are not.*

Proof. For the classical operators, see above. For team semantics,

- $\mathcal{R}_\wedge T = \{(T, T)\}$,
- $\mathcal{R}_\vee T = \{\, (T_1, T_2) \mid T = T_1 \cup T_2 \,\}$,
- $\mathcal{R}_{\dot\vee} T = \{\, (T_1, T_2) \mid T = T_1 \uplus T_2 \,\}$,

- $\mathcal{R}_{\Diamond}T = \{ \bigcup_{w\in T} f(w) \mid f \in \prod_{w\in T} \wp^{+}(Rw) \}$ (in a Kripke structure (W, R, V)),
- $\mathcal{R}_{\dot\Diamond}T = \{ \bigcup_{w\in T} f(w) \mid f \in \prod_{w\in T} \wp^{1}(Rw) \}$,
- $\mathcal{R}_{\Box}T = \{ RT \}$,
- $\mathcal{R}_{\exists x}T = \{ T^x_f \mid f \in \prod_{s\in T} \wp^{+}(\mathcal{A}) \}$ (in a first-order structure $\mathcal{A}$),
- $\mathcal{R}_{\dot\exists x}T = \{ T^x_f \mid f \in \prod_{s\in T} \wp^{1}(\mathcal{A}) \}$,
- $\mathcal{R}_{\forall x}T = \{ T^x_{\mathcal{A}} \}$.

On the other hand, $\neg$ and $\sim$ are for example not monotone, and hence no operators. □

Proposition 3.44. *Every nullary operation $\triangle$ is an operator, and $\mathcal{R}_{\triangle} = \triangle$.*

Proof. Straightforward from Definition 3.39. □

An operator $\triangle$ is *functional* if $|\mathcal{R}_{\triangle}u| = 1$ for every u.

Theorem 3.45 (Self-dual modalities). *Let $\triangle\colon (\wp X)^r \to \wp X$ be a non-nullary operator and $X \neq \emptyset$. The following statements are equivalent:*

(1) *$\triangle^{\complement}$ is also an operator.*

(2) *$\triangle$ is unary and functional.*

(3) *$\triangle$ is strongly self-dual.*

Proof. (1) to (2): Suppose $\triangle$ and $\triangle^{\complement}$ are operators. For the sake of contradiction, assume they have arity two (or higher; the proof is similar then). Any operator is normal (Proposition 3.40), so $\triangle(\emptyset, \mathcal{P}) = \triangle(\mathcal{P}, \emptyset) = \emptyset$ for any $\mathcal{P} \subseteq X$. But the strong dual of $\triangle$ is also an operator and hence normal, so $\triangle(\mathcal{P}, X) = X$ for any $\mathcal{P}$. But then $\emptyset = \triangle(\emptyset, X) = X$, contradiction to the assumption $X \neq \emptyset$. So $\triangle$ can only be unary.

Next, we show that $\triangle$ is functional, that is, that $\mathcal{R}_{\triangle}u$ is a singleton for all $u \in X$. First, assume that $\mathcal{R}_{\triangle}u$ is empty. Then $u \notin \triangle\mathcal{P}$ for any $\mathcal{P} \subseteq X$, in particular $u \notin \triangle X$, so by duality $u \in \triangle^{\complement}\emptyset$, contradiction to normality. So $\mathcal{R}_{\triangle}u \neq \emptyset$, and by symmetry $\mathcal{R}_{\triangle^{\complement}} \neq \emptyset$, for all $u \in X$. Second, if $v \neq v'$ and $v, v' \in \mathcal{R}_{\triangle}u$, then

$$\begin{aligned} u \in \triangle\{v\} \cap \triangle\{v'\} &\Leftrightarrow u \notin \triangle^{\complement}(X \setminus \{v\}) \cup \triangle^{\complement}(X \setminus \{v'\}) \\ &\Leftrightarrow u \notin \triangle^{\complement}((X \setminus \{v\}) \cup (X \setminus \{v'\})) = \triangle^{\complement}X = (\triangle\emptyset)^{\complement} = \emptyset^{\complement} = X, \end{aligned}$$

contradiction to additivity. Hence $\triangle$ (and in fact also $\triangle^{\complement}$) is unary and functional.

(2) to (3): If $\triangle$ is unary and functional, then $\triangle$ is strongly self-dual:

$$\begin{aligned} u \in \triangle(U) &\Leftrightarrow \mathcal{R}_{\triangle}u \cap U \neq \emptyset && \text{(def. operator)} \\ &\Leftrightarrow \mathcal{R}_{\triangle}u \subseteq U && \text{(since } \mathcal{R}_{\triangle}u \text{ is a singleton)} \\ &\Leftrightarrow \mathcal{R}_{\triangle} \cap (X \setminus U) = \emptyset \\ &\Leftrightarrow u \notin \triangle(X \setminus U) && \text{(def. operator)} \end{aligned}$$

(3) to (1): Trivial. □

Example 3.46. Besides $\neg$ and $\sim$, there are also less trivial examples for non-operators, for instance in classical linear temporal logic LTL. The connectives F ("future") and X ("next time") are operators, but the binary U ("until") is not, since $\bot \mathsf{U} \varphi$ is equivalent to φ and not $\bot$.[1]

Every connective has a weak dual, but does every *operator* also have a weakly dual *operator*? In the next subsection, we give a characterization of weak duality to answer this question affirmatively.

3.3.1 Characterizations of weak duality and flatness equivalence

Fix r-ary operators $\triangle, \nabla \colon (\wp\wp X)^r \to \wp\wp X$. In the context of weak duality, we are interested in the relations $\mathcal{R}_\triangle\{s\}$ and $\mathcal{R}_\nabla\{s\}$ for singleton teams $\{s\} \subseteq X$.

As we are working in team semantics, the elements of $\mathcal{R}_\triangle\{s\}$ are r-*tuples of teams*, i.e., tuples $(X_i)_{i\in[r]} = (X_1, \dots, X_r)$ where $X_1, \dots, X_r \subseteq X$. We omit the subscript and just write (X_i) for $(X_i)_{i\in[r]}$. The following abbreviations will also be important. For r-tuples (X_i) and (Y_i), $(X_i) \subseteq (Y_i)$ means that $X_i \subseteq Y_i$ for all i. If $\mathcal{X}$ is a set of r-tuples, then $\bigcup \mathcal{X}$ is the component-wise union $(Y_i)_{i\in[r]}$ defined by

$$Y_i := \bigcup \left\{ X_i \;\middle|\; (X_i)_{i\in[r]} \in \mathcal{X} \right\}.$$

In the next two lemmas, we give an upper and a lower bound, respectively, on what (tuples of) teams may be in $\mathcal{R}_\nabla$ for ∇ to be a weak dual of $\triangle$. So-called *hitting vectors*, a generalization of hitting sets, play a crucial role in this.

Definition 3.47. Let $\mathcal{X} \subseteq (\wp X)^r$. A *hitting vector* of $\mathcal{X}$ is an r-tuple $(H_i)_{i\in[r]} \in (\wp X)^r$ that has a non-empty intersection with every tuple $(X_i) \in \mathcal{X}$, meaning $X_i \cap H_i \neq \emptyset$ for at least one $i \in [r]$. The set of all hitting vectors of $\mathcal{X}$ is $\mathsf{hv}(\mathcal{X})$.

Observe that $\bigcup \mathcal{X}$ itself is a hitting vector of $\mathcal{X}$ (and in fact the maximal one) iff $\mathcal{X}$ has at least one hitting vector, iff $(\emptyset, \dots, \emptyset) \notin \mathcal{X}$.

The characterization of weak duality is given in two steps. Recall that weak duality means that exactly one of $\triangle(\alpha_1, \dots, \alpha_r)$ and $\nabla(\neg\alpha_1, \dots, \neg\alpha_r)$ is true for each singleton. In two lemmas, we first define a "maximal dual candidate" ∇ for $\triangle$, maximal in the sense that $\triangle(\alpha_1, \dots, \alpha_r)$ and $\nabla(\neg\alpha_1, \dots, \neg\alpha_r)$ cannot be simultaneously true; and a "minimal candidate" ∇ where always at least one of them is true.

Lemma 3.48 (Maximal weak dual)**.** *The following are equivalent for all* $T \subseteq X$*:*

(1) $T \notin \nabla(\mathcal{T}_1, \dots, \mathcal{T}_r) \cap \triangle(\neg\mathcal{T}_1, \dots, \neg\mathcal{T}_r)$ *for all flat team properties* $\mathcal{T}_1, \dots, \mathcal{T}_r \subseteq \wp X$.

(2) *For all* $(T_i) \in \mathcal{R}_\nabla T$ *there is a hitting vector* (X_i) *of* $\mathcal{R}_\triangle T$ *such that* $(X_i) \subseteq (T_i)$.

[1] In temporal logic, $\varphi \mathsf{U} \psi$ means "along the current trace $(s_0, s_1, \dots)$, there exists $n \geqslant 0$ such that $(s_n, s_{n+1}, \dots) \vDash \psi$ and $(s_k, s_{k+1}, \dots) \vDash \varphi$ for all $k < n$."

Proof. (1) to (2): Let $(T_1, \dots, T_r) \in \mathcal{R}_\nabla T$ be arbitrary. We have to find a hitting vector (X_i) of $\mathcal{R}_\triangle T$ such that $(X_i) \subseteq (T_i)$. In order to apply the assumption (1), we require flat team properties $\mathcal{T}_1, \dots, \mathcal{T}_r$. Here we simply choose $\mathcal{T}_i := \wp T_i$ (cf. Corollary 3.16). Now $T \in \nabla(\wp T_1, \dots, \wp T_r)$, since clearly $T_i \in \wp(T_i)$. By (1), $T \notin \triangle(\neg\wp T_1, \dots, \neg\wp T_r)$.

But this means that for every $(S_i) \in \mathcal{R}_\triangle T$ there is some i such that $S_i \notin \neg\wp T_i = \wp(X \setminus T_i)$ (cf. Proposition 3.23), i.e., $S_i \nsubseteq X \setminus T_i$. Equivalently, S_i contains an element $s' \in T_i$. Because (S_i) was arbitrary, the tuple $(X_i)_{i \in [r]}$ defined by

$$X_i := \left\{ s' \;\middle|\; s' \in S_i \cap T_i \text{ and } (S_i) \in \mathcal{R}_\triangle T \right\}$$

is a hitting vector of $\mathcal{R}_\triangle T$, and obviously by definition $(X_i) \subseteq (T_i)$.

(2) to (1): For the sake of contradiction, assume there are flat properties $\mathcal{T}_1, \dots, \mathcal{T}_r \subseteq \wp X$ such that both $T \in \nabla(\mathcal{T}_1, \dots, \mathcal{T}_r)$ and $T \in \triangle(\neg\mathcal{T}_1, \dots, \neg\mathcal{T}_r)$, witnessed by tuples $(T_i) \in \mathcal{R}_\nabla T$ such that $\forall i : T_i \in \mathcal{T}_i$ and $(S_i) \in \mathcal{R}_\triangle T$ such that $\forall i : S_i \in \neg\mathcal{T}_i$. By (2), there is a hitting vector (X_i) of $\mathcal{R}_\triangle T$ such that $(X_i) \subseteq (T_i)$. In particular, (S_i) must be hit, i.e., for some i, $X_i \cap S_i \neq \emptyset$. But observe that we also have $X_i \subseteq T_i \subseteq \bigcup \mathcal{T}_i$, and by Proposition 3.23 simultaneously $S_i \subseteq (X \setminus \bigcup \mathcal{T}_i)$, so $X_i \cap S_i = \emptyset$, contradiction. □

Lemma 3.49 (Minimal weak dual). *The following are equivalent for all* $T \subseteq X$*:*

(1) $T \in \nabla(\mathcal{T}_1, \dots, \mathcal{T}_r) \cup \triangle(\neg\mathcal{T}_1, \dots, \neg\mathcal{T}_r)$ *for all flat team properties* $\mathcal{T}_1, \dots, \mathcal{T}_r \subseteq \wp X$.

(2) *For every hitting vector* (X_i) *of* $\mathcal{R}_\triangle T$ *there is* $(T_i) \in \mathcal{R}_\nabla T$ *such that* $(T_i) \subseteq (X_i)$.

Proof. (1) to (2): Let (X_i) be an arbitrary hitting vector of $\mathcal{R}_\triangle T$. In order to to apply (1), we pick the flat property $\mathcal{T}_i := \neg\wp X_i$ (cf. Corollary 3.16). Next, we show that $T \notin \triangle(\neg\wp X_1, \dots, \neg\wp X_r)$. For this, note that any tuple $(T_i) \in \mathcal{R}_\triangle T$ satisfying $\forall i : T_i \in \neg\wp X_i$ would contradict (X_i) being a hitting vector, since by Proposition 3.23, $T_i \in \neg\wp X_i$ iff $T_i \subseteq X \setminus X_i$ iff $T_i \cap X_i = \emptyset$. Accordingly, $T \notin \triangle(\neg\wp X_1, \dots, \neg\wp X_r)$. In consequence, by (1), $T \in \nabla(\wp X_1, \dots, \wp X_r)$. In other words, there is some $(T_i) \in \mathcal{R}_\nabla T$ such that $T_i \subseteq X_i$ for all $i \in [r]$, as desired.

(2) to (1): Let $\mathcal{T}_1, \dots, \mathcal{T}_r$ be flat team properties. If $T \in \triangle(\neg\mathcal{T}_1, \dots, \neg\mathcal{T}_r)$, then we are done, so assume the contrary. This means that for every $(T_i) \in \mathcal{R}_\triangle T$ there is some $i \in [r]$ such that $T_i \notin \neg\mathcal{T}_i$. By flatness of $\neg\mathcal{T}_i$, equivalently T_i contains an element s' such that $\{s'\} \notin \neg\mathcal{T}_i$, hence $\{s'\} \in \mathcal{T}_i$. We gather these elements in sets

$$X_i := \left\{ s' \;\middle|\; s' \in T_i, \{s'\} \in \mathcal{T}_i \text{ and } (T_i) \in \mathcal{R}_\triangle T \right\}.$$

By construction, for every $(T_i) \in \mathcal{R}_\triangle T$ there is some i such that $X_i \cap T_i \neq \emptyset$. In other words, (X_i) is a hitting vector of $\mathcal{R}_\triangle T$. Due to (2), there must be $(T_i) \in \mathcal{R}_\nabla T$ such that $(T_i) \subseteq (X_i)$. Moreover, $\forall i \in [r] : X_i \in \mathcal{T}_i$, since $\mathcal{T}_i$ contains all singletons $\{s\}$ for $s \in X_i$ and is flat. This implies $T_i \in \mathcal{T}_i$ by downward closure of $\mathcal{T}_i$ (Proposition 3.15). As a consequence, we have $T \in \nabla(\mathcal{T}_1, \dots, \mathcal{T}_r)$ witnessed by (T_i). □

We can restate the above two lemmas more concisely in terms from order theory. If X is a family of (tuples of) sets, then a subfamily $Y \subseteq X$ is called *coinitial* if for all $x \in X$

there exists $y \in Y$ such that $y \subseteq x$.[1]

Theorem 3.50 (Weak dual characterization). *The following statements are equivalent:*

(1) *$\triangle$ and ∇ are weak duals.*

(2) *For all $s \in X$, the families $\mathcal{R}_\nabla\{s\}$ and $\mathsf{hv}(\mathcal{R}_\triangle\{s\})$ are mutually coinitial.*

(3) *For all $s \in X$, the families $\mathcal{R}_\triangle\{s\}$ and $\mathsf{hv}(\mathcal{R}_\nabla\{s\})$ are mutually coinitial.*

Proof. By symmetry (Proposition 3.27) it suffices to consider only (1) and (2). Their equivalence results from the above two lemmas as follows. First, recall that ∇ and $\triangle$ are weak duals if for all flat team properties $\mathcal{T}_1, \dots, \mathcal{T}_r$, any singleton $\{s\}$ is contained in exactly one of $\nabla(\mathcal{T}_1, \dots, \mathcal{T}_r)$ and $\triangle(\neg\mathcal{T}_1, \dots, \mathcal{T}_r)$. Now $\mathcal{R}_\nabla\{s\}$ and $\mathsf{hv}(\mathcal{R}_\triangle\{s\})$ being mutually coinitial is, by these lemmas, equivalent to $\{s\} \notin \nabla(\mathcal{T}_1, \dots, \mathcal{T}_r) \cap \triangle(\neg\mathcal{T}_1, \dots, \neg\mathcal{T}_r)$ and $\{s\} \in \nabla(\mathcal{T}_1, \dots, \mathcal{T}_r) \cup \triangle(\neg\mathcal{T}_1, \dots, \neg\mathcal{T}_r)$, i.e., to weak duality of $\triangle$ and ∇. $\square$

Corollary 3.51. *Every operator has a weakly dual operator.*

Proof. Given an operator $\triangle$, define a weakly dual operator ∇ by $\mathcal{R}_\nabla T := \mathsf{hv}(\mathcal{R}_\triangle T)$. $\square$

Example 3.52. Let us list some concrete team-logical operators and confirm that they are weakly dual. Recall from p. 44 that $\mathcal{R}_\wedge T = \{(T, T)\}$. On singletons, consequently

$$\mathcal{R}_\wedge\{s\} = \{\, (\{s\}, \{s\}) \,\}.$$

What are the possible hitting vectors? This is the set

$$\mathsf{hv}(\mathcal{R}_\wedge\{s\}) = \{\, (\emptyset, \{s\}), (\{s\}, \emptyset), (\{s\}, \{s\}) \,\}$$

which is precisely $\mathcal{R}_\vee\{s\}$. Conversely, $\{(\{s\}, \{s\})\}$ is the only hitting vector of the above set. So $\vee$ is weakly dual to $\wedge$. By contrast, the strict splitting,

$$\mathcal{R}_{\dot\vee}\{s\} = \{\, (\emptyset, \{s\}), (\{s\}, \emptyset) \,\}$$

is only a subset of hitting vectors. But it is not arbitrary; it is the smallest coinitial subset of hitting vectors, that is, all other hitting vectors contain one of its elements w. r. t. $\subseteq$. Let us stress that the only hitting vector of either $\mathcal{R}_\vee\{s\}$ or $\mathcal{R}_{\dot\vee}\{s\}$ is $(\{s\}, \{s\})$, which coincides with $\mathcal{R}_\wedge\{s\}$. This explains why $\vee$ comes in strict and lax flavors, but $\wedge$ does not.

As another example, consider modal team logic. If $w \in W$ is a world in a Kripke frame (W, R), then $\mathcal{R}_\Box\{w\} = \{Rw\}$. The hitting vectors (or rather hitting sets) are precisely all non-empty subsets, i.e., the elements of $\wp^+(Rw)$. Indeed, $\mathcal{R}_\Diamond\{w\} = \wp^+(Rw)$ and $\mathcal{R}_{\dot\Diamond}\{w\} = \wp^1(Rw)$, where the latter again contains only the minimal hitting sets of $\{Rw\}$. In the converse direction, again $\mathcal{R}_\Box\{w\} = \{Rw\}$ is the only possible hitting set of $\wp^+(Rw)$ and/or $\wp^1(Rw)$, which explains why $\Box$ also has no strict and lax variants.

Note that for successorless worlds we can have $\mathcal{R}_\Box\{w\} = \{Rw\} = \{\emptyset\}$ (as a formula, $\Box\bot$ is then true). Then there are no hitting vectors at all, i.e., $\mathcal{R}_\Diamond\{w\} = \emptyset$ (so $\neg\Diamond\top$ is true),

[1] The dual notion, *cofinality*, is common, for example in set theory.

since the empty set cannot be hit anywhere. However, unlike $\{\emptyset\}$, $\emptyset$ *has* a hitting set, namely $\emptyset$! Formally, we have to distinguish $\mathsf{hv}(\emptyset) = \{\emptyset\}$ and $\mathsf{hv}(\{\emptyset\}) = \emptyset$.

Remark. Not all duals $\triangledown$ of an operator $\vartriangle$ are necessarily operators, not even if $\vartriangle$ and $\triangledown$ preserve flatness. For example, let $\triangledown$ be a flatness preserving weak dual of $\vartriangle$, but set $\triangledown(\emptyset) \neq \emptyset$. Since $\emptyset$ is not a flat property, this changes neither the fact that $\triangledown$ is flatness preserving, nor the weak duality. But $\triangledown$ is not normal and hence not an operator.

Closely related to the concept of weak duality is that of *flatness equivalence*. For example, given flat formulas α_1, α_2, we have the equivalence $\alpha_1 \dot\vee \alpha_2 \equiv \alpha_1 \vee \alpha_2$. In this subsection, we characterize this sort of equivalence in a similar fashion as weak duality. In what follows, let $\vartriangle, \triangledown \colon (\wp\wp X)^r \to \wp\wp X$ again be operators.

Definition 3.53. The operators $\vartriangle, \triangledown \colon (\wp\wp X)^r \to \wp\wp X$ are *flatness equivalent* if

$$\{s\} \in \vartriangle(\mathcal{T}_1, \dots, \mathcal{T}_r) \Leftrightarrow \{s\} \in \triangledown(\mathcal{T}_1, \dots, \mathcal{T}_r)$$

for all flat team properties $\mathcal{T}_1, \dots, \mathcal{T}_r \subseteq \wp X$ and $s \in X$.

Proposition 3.54. *Flatness equivalence is an equivalence relation, i.e., reflexive, symmetric, and transitive.*

Again, a stronger natural statement would be $\vartriangle(\mathcal{T}_1, \dots, \mathcal{T}_r) = \triangledown(\mathcal{P}_1, \dots, \mathcal{P}_r)$, but:

Proposition 3.55. *If $\vartriangle$ and $\triangledown$ preserve flatness and are flatness equivalent, then*

$$\vartriangle(\mathcal{T}_1, \dots, \mathcal{T}_r) = \triangledown(\mathcal{T}_1, \dots, \mathcal{T}_r)$$

for all flat team properties $\mathcal{T}_1, \dots, \mathcal{T}_r \subseteq \wp X$.

Proof. If $\vartriangle$ and $\triangledown$ preserve flatness, then $\vartriangle(\mathcal{T}_1, \dots, \mathcal{T}_r)$ and $\triangledown(\mathcal{T}_1, \dots, \mathcal{T}_r)$ are flat. Consequently, $T \in \vartriangle(\mathcal{T}_1, \dots, \mathcal{T}_r)$ iff $\{s\} \in \vartriangle(\mathcal{T}_1, \dots, \mathcal{T}_r)$ for all $s \in T$ iff $\{s\} \in \triangledown(\mathcal{T}_1, \dots, \mathcal{T}_r)$ for all $s \in T$ iff $T \in \triangledown(\mathcal{T}_1, \dots, \mathcal{T}_r)$. □

The next theorem is analogous to Theorem 3.50.

Theorem 3.56. *The following statements are equivalent:*

(1) *$\vartriangle$ and $\triangledown$ are flatness equivalent.*

(2) *For all $s \in X$, the families $\mathcal{R}_\vartriangle\{s\}$ and $\mathcal{R}_\triangledown\{s\}$ are mutually coinitial.*

(3) *For all $s \in X$ the families $\mathsf{hv}(\mathcal{R}_\vartriangle\{s\})$ and $\mathsf{hv}(\mathcal{R}_\triangledown\{s\})$ are mutually coinitial.*

Proof. (1) to (2): Let $s \in X$ and $(T_i) \in \mathcal{R}_\triangledown\{s\}$. To show that $\mathcal{R}_\vartriangle\{s\}$ is coinitial in $\mathcal{R}_\triangledown\{s\}$, we have to find some $(S_i) \in \mathcal{R}_\vartriangle\{s\}$ such that $(S_i) \subseteq (T_i)$. As in the proofs of Lemma 3.48 and 3.49, we use the flat properties $\wp T_1, \dots, \wp T_r$. By assumption, $\{s\} \in \triangledown(\wp T_1, \dots, \wp T_r)$, and by (1), $\{s\} \in \vartriangle(\wp T_1, \dots, \wp T_r)$. This is witnessed by some $(S_i) \in \mathcal{R}_\vartriangle\{s\}$. But then (S_i) is our desired tuple, since $S_i \in \mathcal{T}_i = \wp T_i$ implies $S_i \subseteq T_i$. Since T_i was arbitrary, $\mathcal{R}_\vartriangle\{s\}$ is coinitial in $\mathcal{R}_\triangledown\{s\}$. The above argument is symmetric, which proves (2).

(2) to (3): Let $s \in X$ and $(X_i) \in \mathsf{hv}(\mathcal{R}_\nabla\{s\})$ be arbitrary. The other direction of (3) is again symmetric, so we define a tuple $(Y_i) \in \mathsf{hv}(\mathcal{R}_\triangle\{s\})$ by

$$Y_i := \{ s' \mid s' \in U_i \cap X_i \text{ for some } (U_i) \in \mathcal{R}_\triangle\{s\} \}$$

where obviously $(Y_i) \subseteq (X_i)$. It remains to show that (Y_i) is actually a hitting vector of $\mathcal{R}_\triangle\{s\}$. Suppose the contrary, that is, for some $(U_i) \in \mathcal{R}_\triangle\{s\}$ and all i it holds $U_i \cap X_i = \emptyset$, i.e., $U_i \subseteq X \setminus X_i$. By (2) there is a family $(S_i) \in \mathcal{R}_\nabla\{s\}$ such that $(S_i) \subseteq (U_i)$, hence also $S_i \subseteq X \setminus X_i$ for all i. But by definition, (X_i) must hit (S_i), contradiction.

(3) to (1): Let $\mathcal{T}_1, \ldots, \mathcal{T}_r \subseteq \wp X$ be arbitrary flat team properties, and let $s \in X$. To prove that $\triangle$ and ∇ are flatness equivalent, we assume $\{s\} \in \triangle(\mathcal{T}_1, \ldots, \mathcal{T}_r)$ but $\{s\} \notin \nabla(\mathcal{T}_1, \ldots, \mathcal{T}_r)$ and derive a contradiction. By symmetry, this suffices.

First of all, $\mathcal{T}_i = \wp T_i$ for teams T_i (Corollary 3.16). By assumption, for every $(S_i) \in \mathcal{R}_\nabla\{s\}$ there must exist i such that $S_i \not\subseteq T_i$, i.e., $S_i \cap (X \setminus T_i) \neq \emptyset$. This means that

$$X_i := \{ s' \mid s' \in S_i \cap (X \setminus T_i) \text{ and } (S_i) \in \mathcal{R}_\nabla\{s\} \}$$

defines a hitting vector (X_i) of $\mathcal{R}_\nabla\{s\}$. By (3), there is a corresponding hitting vector (Y_i) of $\mathcal{R}_\triangle\{s\}$ such that $(Y_i) \subseteq (X_i)$. Note that $Y_i \cap T_i = \emptyset$ for all i by construction of X_i. Since we assumed $\{s\} \in \triangle(\mathcal{T}_1, \ldots, \mathcal{T}_r)$, there is a witnessing tuple $(U_i) \in \mathcal{R}_\triangle\{s\}$ such that $\forall i : U_i \subseteq T_i$. But (Y_i) has to hit (U_i), contradiction. □

Corollary 3.57. *$\vee$ is flatness equivalent to $\dot{\vee}$. $\Diamond$ is flatness equivalent to $\dot{\Diamond}$. $\exists x$ is flatness equivalent to $\dot{\exists} x$.*

Proof. By definition of $\dot{\vee}$ and $\vee$, clearly $\mathcal{R}_{\dot{\vee}}\{s\} \subseteq \mathcal{R}_\vee\{s\}$, and a set is coinitial in all of its subsets. It remains to show that $\mathcal{R}_{\dot{\vee}}\{s\}$ is coinitial in $\mathcal{R}_\vee\{s\}$. But for any $(T_1, T_2) \in \mathcal{R}_\vee\{s\}$ it holds that $(T_1, T_2 \setminus T_1) \in \mathcal{R}_{\dot{\vee}}\{s\}$. As $(T_1, T_2 \setminus T_1) \subseteq (T_1, T_2)$, then $\mathcal{R}_\vee\{s\}$ is also coinitial in $\mathcal{R}_{\dot{\vee}}\{s\}$. The same holds for $\mathcal{R}_\Diamond\{s\} = \wp^+(Rs)$ and $\mathcal{R}_{\dot{\Diamond}}\{s\} = \wp^1(Rs)$, since $\wp^1(Rs) \subseteq \wp^+(Rs)$, and since every non-empty set $S \in \wp^+(Rs)$ contains some singleton $\{s\}$ which is in $\wp^1(Rs)$. The quantifiers $\exists x$ and $\dot{\exists} x$ are analogous. □

Finally, we conclude this section with a characterization of flatness equivalence that is analogous to Theorem 3.37 and works also for non-operators.

Theorem 3.58. *For $i \in \{1, 2\}$, let $\triangle_i : (\wp X)^r \to \wp X$, and let $\nabla_i : (\wp\wp X)^r \to \wp\wp X$ be a teamification of $\triangle_i$. Then ∇_1 and ∇_2 are flatness equivalent if and only if $\triangle_1 = \triangle_2$.*

Proof. "$\Rightarrow$": Let ∇_1 and ∇_2 be flatness equivalent. By Theorem 3.34, they are also flatness preserving, and hence $\nabla_1(\mathcal{T}_1, \ldots, \mathcal{T}_r) = \nabla_2(\mathcal{T}_1, \ldots, \mathcal{T}_r)$ for all flat $\mathcal{T}_i$ (Proposition 3.55). By definition of teamification, for all teams $T_1, \ldots, T_r \subseteq X$ we have $\wp\triangle_i(T_1, \ldots, T_r) = \nabla_i(\wp T_1, \ldots, \wp T_r)$. But then $\wp\triangle_1(T_1, \ldots, T_r) = \wp\triangle_2(T_1, \ldots, T_r)$ for all $T_1, \ldots, T_r \subseteq X$. As $\wp$ is injective, $\triangle_1 = \triangle_2$.

"$\Leftarrow$": We have to prove $\{s\} \in \nabla_1(\wp T_1, \ldots, \wp T_r) \Leftrightarrow \{s\} \in \nabla_2(\wp T_1, \ldots, \wp T_r)$ for all teams $T_1, \ldots, T_r \subseteq X$. But this follows immediately from $\nabla_1(\wp T_1, \ldots, \wp T_r) = \wp\triangle_1(T_1, \ldots, T_r) = \wp\triangle_2(T_1, \ldots, T_r) = \nabla_2(\wp T_1, \ldots, \wp T_r)$. □

3.4 Transversals

We saw that the common team-logical connectives are operators (Theorem 3.43), but in fact they satisfy an even stronger property: The sets $\mathcal{R}_\nabla T$ depend solely on the sets $\mathcal{R}_\nabla\{s\}$ for individuals $s \in T$. Loosely speaking, to find a successor team of T, we traverse all $s \in T$ and pick one of the possible "successor teams" of each $\{s\}$, and afterwards take the union of all these. We call such operators *transversals*.

Definition 3.59 (Transversal)**.** A *transversal* is an operator $\triangle\colon (\wp\wp X)^r \to \wp\wp X$ that satisfies the following equation for all $T \subseteq X$:

$$\mathcal{R}_\triangle T = \left\{ \bigcup\{f(s) \mid s \in T\} \;\middle|\; f \in \prod_{s\in T} \mathcal{R}_\triangle\{s\} \right\}. \tag{$\star$}$$

We briefly explain this equation. For every s, $\mathcal{R}_\triangle\{s\}$ contains r-tuples of teams. $\prod_{s\in T}\mathcal{R}_\triangle\{s\}$ is the set of all choice functions f that for each $s \in T$ pick exactly one successor team of $\{s\}$, and so $f(s)$ is a specific r-tuple of teams, with $\bigcup\{f(s) \mid s \in T\}$ being the (component-wise) union over all $s \in T$.

The above equation *uniquely determines* a transversal in terms of the sets $\mathcal{R}_\nabla\{s\}$ for singletons $\{s\}$. Next, we show that the common operators are transversals.

Theorem 3.60. *The operators* $\wedge, \vee, \dot\vee, \Box, \Diamond, \dot\Diamond, \forall x, \exists x, \dot\exists x$ *are transversals.*

Proof. We prove ($\star$) for each of these operators.

- $\wedge$: Recall that $\mathcal{R}_\wedge T = \{\, (T, T) \,\}$, in particular $\mathcal{R}_\triangle\{s\} = \{\, (\{s\}, \{s\}) \,\}$. Now $\prod_{s\in T}\mathcal{R}_\wedge\{s\}$ contains only one choice function f, namely $f(s) = (\{s\}, \{s\})$. We conclude

$$\mathcal{R}_\wedge T = \{(T, T)\} = \left\{\bigcup\{\, (\{s\}, \{s\}) \mid s \in T \,\}\right\} = \left\{\bigcup\{\, f(s) \mid s \in T \,\}\right\}.$$

- $\vee$: Disjunction is generated by $\mathcal{R}_\vee T = \{\, (T_1, T_2) \mid T_1 \cup T_2 = T \,\}$. On singletons, hence $\mathcal{R}_\vee\{s\} = \{\, (\{s\}, \emptyset), (\emptyset, \{s\}), (\{s\}, \{s\}) \,\}$. We prove that ($\star$) is satisfied.

  For "$\subseteq$", suppose $T = T_1 \cup T_2$. Then we pick the choice function f defined by $f(s) = (\{s\}, \emptyset)$ if $s \in T_1 \setminus T_2$, by $f(s) = (\emptyset, \{s\})$ if $s \in T_2 \setminus T_1$, and $f(s) = (\{s\}, \{s\})$ if $s \in T_1 \cap T_2$. Then clearly $\bigcup_{s\in T} f(s) = (T_1, T_2)$.

  For "$\supseteq$", let $f \in \prod_{s\in T}\{\, (\{s\}, \emptyset), (\emptyset, \{s\}), (\{s\}, \{s\}) \,\}$. Define teams

$$\begin{aligned} S_1 &:= \{\, s \in T \mid f(s) = (\{s\}, \emptyset) \,\} \\ S_2 &:= \{\, s \in T \mid f(s) = (\emptyset, \{s\}) \,\} \\ S_3 &:= \{\, s \in T \mid f(s) = (\{s\}, \{s\}) \,\}. \end{aligned}$$

  Then clearly $\bigcup_{s\in T} f(s) = (S_1 \cup S_3, S_2 \cup S_3)$. But since $T = S_1 \cup S_2 \cup S_3$, also $T = (S_1 \cup S_3) \cup (S_2 \cup S_3)$. As a consequence, $(S_1 \cup S_3, S_2 \cup S_3) \in \mathcal{R}_\vee T$.

- $\dot\vee$: The proof is similar to $\vee$, but with $S_3 = \emptyset$.

- $\Box$: Let $\mathcal{K} = (W, R, V)$ be a Kripke structure and $T \subseteq W$ a team. Then $\mathcal{R}_\Box T = \{ RT \}$. Consequently, $\mathcal{R}_\Box\{s\} = \{ Rs \}$. Again, the choice function f is unique, and obviously, $RT = \bigcup_{s\in T} Rs = \bigcup_{s\in T} f(s)$.
- $\Diamond$: Let $\mathcal{K}$ and T be as above. Then $\Diamond$ is generated by the relation $\mathcal{R}_\Diamond$ where $S \in \mathcal{R}_\Diamond T$ iff S is a successor team of T. The successor teams of a singleton $\{s\}$ are exactly the non-empty subsets of its image: $\mathcal{R}_\Diamond\{s\} = \wp^+(Rs)$. Moreover, by definition of $\Diamond$, the successor teams S are precisely those of the form $\bigcup_{s\in T} f(s)$ for some $f \in \prod_{s\in T} \wp^+(Rs)$. Hence $(\star)$ holds.
- $\dot\Diamond$: Analogous to $\Diamond$, with $\wp^1(Rs) = \mathcal{R}_{\dot\Diamond}\{s\}$.
- $\forall x$: Let $\mathcal{A}$ be a first-order structure, $T \subseteq \mathrm{Var} \to \mathcal{A}$ a team, and $x \in \mathrm{Var}$. Analogously to $\Box$, now $\mathcal{R}_{\forall x} T = \{ T^x_{\mathcal{A}} \}$, and on singletons, $\mathcal{R}_{\forall x}\{s\} = \{ \{s\}^x_{\mathcal{A}} \}$. By definition, $T^x_{\mathcal{A}} = \bigcup_{s\in T}\{s\}^x_{\mathcal{A}}$, so $T^x_{\mathcal{A}} = \bigcup_{s\in T} f(s)$ for the unique choice function f.
- $\exists x$: Analogously to $\Diamond$. Let $\mathcal{A}, x, T$ be as above. By the semantics of $\exists$, $S \in \mathcal{R}_{\exists x} T$ iff $S = T^x_h$ for some $h\colon T \to \wp^+(\mathcal{A})$. So for singletons $\{s\}$, $S \in \mathcal{R}_{\exists x}\{s\}$ iff $S = \{s\}^x_A$ for some $A \in \wp^+(\mathcal{A})$ iff $S \in \wp^+(\{s\}^x_{\mathcal{A}})$. For arbitrary teams T, then

$$\begin{aligned}
& S \in \mathcal{R}_{\exists x} T \\
\Leftrightarrow\; & S = T^x_h \text{ for some } h\colon T \to \wp^+(\mathcal{A}) && \text{(team semantics of } \exists) \\
\Leftrightarrow\; & S = \{s^x_a \mid s \in T, a \in h(s)\} \text{ for some } h\colon T \to \wp^+(\mathcal{A}) && \text{(def. suppl. team)} \\
\Leftrightarrow\; & S = \{s' \mid s' \in f(s), s \in T\} \text{ for some } f \in \prod_{s\in T} \wp^+(\{s\}^x_{\mathcal{A}}) \\
\Leftrightarrow\; & S = \bigcup_{s\in T} f(s) \text{ for some } f \in \prod_{s\in T} \wp^+(\{s^x_{\mathcal{A}}\}).
\end{aligned}$$

- $\dot\exists x$: Analogous to $\exists x$, with $\wp^+$ replaced by $\wp^1$. $\Box$

From the above picture, the atoms are missing. For atoms however, transversals and flat operators coincide.

Theorem 3.61. *Let $\triangle \subseteq \wp X$ be flat. Then $\triangle$ is a transversal.*

Proof. First, $\triangle$ is an operator (Proposition 3.44). Next, note that $T \in \mathcal{R}_\triangle$ iff the empty tuple ε is in $\mathcal{R}_\triangle T$. In particular, for every team T, either $\mathcal{R}_\triangle T = \{\varepsilon\}$ or $\mathcal{R}_\triangle T = \emptyset$. We distinguish between two cases:

- $T \in \triangle$, i.e., $\mathcal{R}_\triangle T = \{\varepsilon\}$: By flatness, $\{s\} \in \triangle$ for all $s \in T$, so $\mathcal{R}_\triangle\{s\} = \{\varepsilon\}$. The only choice function $f \in \prod_{s\in T} \mathcal{R}_\triangle\{s\}$ is the constant function $f(s) = \varepsilon$. But then

$$\mathcal{R}_\triangle T = \{\varepsilon\} = \left\{ \bigcup\{\varepsilon \mid s \in T\} \right\} = \left\{ \bigcup\{f(s) \mid s \in T\} \;\middle|\; f \in \prod_{s\in T} \mathcal{R}_\nabla\{s\} \right\},$$

so $(\star)$ holds.

- $T \notin \triangle$: By flatness, $\{s_0\} \notin \triangle$ for some $s_0 \in T$, hence $\mathcal{R}_\triangle\{s_0\} = \emptyset$. Then the set $\prod_{s \in T} \mathcal{R}_\triangle\{s\}$ of choice functions is empty, as there is nothing to choose for s_0. However, also $\mathcal{R}_\triangle T = \emptyset$, so $(\star)$ holds again. ☐

Remarkably, virtually all common team-logical connectives are transversals (see Table 3.1), which suggests that they constitute a natural class of connectives in team semantics. At the very least, they are a tool to straightforwardly lift classical operators to teams in a way that is guaranteed to preserve flatness, as we will see in the next theorem.

Theorem 3.62. *Every transversal preserves flatness.*

Proof. Let $\triangle$ be a transversal and $\mathcal{T}_1, \ldots, \mathcal{T}_r \subseteq \wp X$ flat team properties. We have to show that $\mathcal{T}_0 := \triangle(\mathcal{T}_1, \ldots, \mathcal{T}_r)$ is flat, i.e.,

$$T \in \mathcal{T}_0 \Leftrightarrow \forall s \in T : \{s\} \in \mathcal{T}_0.$$

For "$\Rightarrow$", assume $T \in \mathcal{T}_0$. Then there exists a tuple $(T_i) \in \mathcal{R}_\triangle T$ such that $T_i \in \mathcal{T}_i$ for all $i \in [r]$. As $\triangle$ is a transversal, there is a choice function $f \in \prod_{s \in T} \mathcal{R}_\triangle\{s\}$ such that $(T_i) = \bigcup\{f(s) \mid s \in T\}$. For each s, call the corresponding tuple $(T_1^s, \ldots, T_r^s) := f(s)$. Then we can write T_i as $\bigcup_{s \in T} T_i^s$. In particular $T_i^s \subseteq T_i$, and since flatness implies downward closure, we have $T_i^s \in \mathcal{T}_i$. But then $\{s\} \in \mathcal{T}_0$ via the tuple $(T_1^s, \ldots, T_r^s)$ in $\mathcal{R}_\triangle\{s\}$.

For "$\Leftarrow$", assume $\{s\} \in \mathcal{T}_0$ for all $s \in T$. Again for every $s \in T$, $\{s\} \in \mathcal{T}_0$ is witnessed by some tuple $(T_1^s, \ldots, T_r^s) \in \mathcal{R}_\triangle\{s\}$ such that $T_i^s \in \mathcal{T}_i$. Let $U_i := \bigcup_{s \in T} T_i^s$. Flatness implies union closure, hence also $U_i \in \mathcal{T}_i$. As the function f mapping each $s \in T$ to $(T_1^s, \ldots, T_r^s)$ is an element of $\prod_{s \in T} \mathcal{R}_\triangle\{s\}$, by definition of a transversal, (U_i) must be in $\mathcal{R}_\triangle T$. But as we also showed $U_i \in \mathcal{T}_i$ for each i, we can conclude $T \in \mathcal{T}_0$. ☐

Observe that this constitutes another proof for Proposition 3.18.

Not every flatness preserving operator is a transversal. If we define $\mathcal{R}_\triangle T = \{T\}$ for every non-empty team T, but $\mathcal{R}_\triangle \emptyset = \wp X$, then $\triangle$ is a flatness preserving unary operator, but in a transversal the only successor team of $\emptyset$ can be $\emptyset$.

Example 3.63. The universal modality $\boxed{u}$ is defined in a Kripke structure (W, R, V) as $w \vDash \boxed{u}\varphi \Leftrightarrow \forall v \in W : v \vDash \varphi$. For a team semantics, we could naively set $\mathcal{R}_{\boxed{u}} T = \{W\}$ for all T, analogously to $\Box$. However, this would result in $\boxed{u}$ not being flatness preserving, since then, e.g., $\emptyset \nvDash \boxed{u}\bot$ in non-empty structures. Instead, we define it as a transversal, where the natural choice is $\mathcal{R}_{\boxed{u}}\{w\} := \{W\}$ for all $w \in W$. Then $\mathcal{R}_{\boxed{u}}\emptyset = \{\emptyset\}$ and $\mathcal{R}_{\boxed{u}} T = \{W\}$ for $T \neq \emptyset$. This yields a flatness preserving semantics of $\boxed{u}$, namely:

$$T \vDash \boxed{u}\varphi \Leftrightarrow \begin{cases} \text{always} & \text{if } T = \emptyset \\ W \vDash \varphi & \text{if } T \neq \emptyset \end{cases}$$

3.5 Relaxations

Some transversals considered so far come in flatness equivalent pairs, like $\vee/\dot\vee$, $\Diamond/\dot\Diamond$, $\exists/\dot\exists$, of so-called *lax* and *strict* variants. For others like $\wedge, \Box, \forall$, no such distinction exists.

Logic	Transversal ∇		$\mathcal{R}_\nabla\{s\}$
All	$\wedge$	Conjunction	$\{\,(\{s\},\{s\})\,\}$
	$\vee$	Lax disjunction	$\{\,(\{s\},\emptyset),(\emptyset,\{s\}),(\{s\},\{s\})\,\}$
	$\dot\vee$	Strict disjunction	$\{\,(\{s\},\emptyset),(\emptyset,\{s\})\,\}$
Modal	$\Box$	Box	$\{\,Rs\,\}$
	$\Diamond$	Lax diamond	$\wp^+(Rs)$
	$\dot\Diamond$	Strict diamond	$\wp^1(Rs)$
	p	Proposition	$\begin{cases}\{\varepsilon\} & \text{if } s \in V(p)\\ \emptyset & \text{else}\end{cases}$
	$\neg p$	Negated proposition	$\begin{cases}\emptyset & \text{if } s \in V(p)\\ \{\varepsilon\} & \text{else}\end{cases}$
First-order	$\forall x$	Universal quantifier	$\{\,\{s\}^x_{\mathcal{A}}\,\}$
	$\exists x$	Lax existential quantifier	$\wp^+(\{s\}^x_{\mathcal{A}})$
	$\dot\exists x$	Strict existential quantifier	$\wp^1(\{s\}^x_{\mathcal{A}})$
Temporal	X	Nexttime	$\{\,\{s[1]\}\,\}$
	F^a	Future (async.)	$\{\,\{s[k]\} \mid k \geqslant 0\,\}$
	G^d	Globally	$\{\,\{s[k] \mid k \geqslant 0\}\,\}$

Table 3.1: Transversals. For modal team logic, in a Kripke structure (W, R, V), the carrier is $\wp\wp W$. For first-order logic, if $\mathcal{A}$ is a first-order structure, the carrier is $\wp\wp(\mathsf{Var} \to \mathcal{A})$. For the temporal operators, cf. Section 3.8.

In this section, we study the connection between these pairs formally.

Definition 3.64 (Relaxation)**.** The *relaxation* of an operator $\triangle\colon (\wp\wp X)^r \to \wp\wp X$ is the operator $\triangle^\cup\colon (\wp\wp X)^r \to \wp\wp X$ defined by

$$\mathcal{R}_{\triangle^\cup} T := \left\{ \bigcup \mathcal{X} \;\middle|\; \mathcal{X} \in \wp^+(\mathcal{R}_\triangle T) \right\}.$$

In other words, the "$\triangle^\cup$-successors" of a team are the non-empty unions of its "$\triangle$-successors".

Example 3.65. Observe that $\vee = \dot\vee^\cup$, $\Diamond = \dot\Diamond^\cup$ and $\exists x = \dot\exists x^\cup$. For example, $\mathcal{R}_{\dot\vee} T = \{(T_1, T_2) \mid T = T_1 \cup T_2, T_1 \cap T_2 = \emptyset\}$ and $\mathcal{R}_\vee T = \{(T_1, T_2) \mid T = T_1 \cup T_2\}$, but any pair (T_1, T_2) with $T = T_1 \cup T_2$ can be written as the union of two *disjoint* pairs from $\mathcal{R}_{\dot\vee} T$, namely $(T_1, T_2 \setminus T_1) \cup (T_1 \setminus T_2, T_2)$.

On the other hand, we have $\Box = \Box^\cup$, $\wedge = \wedge^\cup$, and $\forall x = \forall x^\cup$. From this perspective, "lax" connectives are those that cannot be further "relaxed".

Definition 3.66 (Strict and lax)**.** An operator $\triangle$ is *lax* if it is its own relaxation. Otherwise it is *strict*.

This definition of strictness is subtle, and ultimately a bit unsatisfying, as the following argument shows. Fix some lax operator ∇, and suppose we want to define a *strictmost* operator corresponding to ∇. For this, we could define a partial order $\leqslant$ on all connectives flatness equivalent to ∇ such that $\triangle_1 \leqslant \triangle_2$ if and only if $\mathcal{R}_{\triangle_1}\mathsf{T} \subseteq \mathcal{R}_{\triangle_2}\mathsf{T}$ for all T. Then intuitively $\leqslant$ means "is stricter than". Analogously to $\dot{\vee}$, $\dot{\Diamond}$, and $\dot{\exists}$, we could now call those operators *strict* that are $\leqslant$-minimal. However, this does not work in general. For example, suppose we have the transversal $\Diamond^\infty$ defined by $\mathcal{R}_{\Diamond^\infty}\{s\} = \{S \subseteq Rs \mid S \text{ is infinite}\}$. $\Diamond^\infty$ is lax, but what is the "strictmost" operator it corresponds to? The relation $\leqslant$ is in general not well-founded, so there is no hope in finding such an operator.

We proceed with some basic properties of relaxations.

Theorem 3.67 (Relaxation laws). *Let $\triangle$ be an operator. Then the following hold:*

(1) $\mathcal{R}_\triangle \mathsf{T} \subseteq \mathcal{R}_{\triangle^\cup}\mathsf{T}$ *for any* T (*monotonicity*).

(2) $\triangle^{\cup\cup} = \triangle^\cup$ (*idempotency*).

(3) *If* $\mathcal{Y} \in \wp^+(\mathcal{R}_{\triangle^\cup}\mathsf{T})$, *then* $\bigcup \mathcal{Y} \in \mathcal{R}_{\triangle^\cup}\mathsf{T}$ (*relaxations are closed under non-empty union*).

Proof. (1): If $(U_i) \in \mathcal{R}_\triangle\mathsf{T}$, then also $(U_i) \in \mathcal{R}_{\triangle^\cup}\mathsf{T}$, since $(U_i) = \bigcup\{(U_i)\}$ and $\{(U_i)\} \in \wp^+(\mathcal{R}_\triangle\mathsf{T})$. (2) follows from (1) and (3): We have $\mathcal{R}_{\triangle^\cup}\mathsf{T} \subseteq \mathcal{R}_{\triangle^{\cup\cup}}\mathsf{T}$ by (1); and if $(U_i) \in \mathcal{R}_{\triangle^{\cup\cup}}\mathsf{T}$, then $(U_i) = \bigcup \mathcal{Y}$ for some $\mathcal{Y} \in \wp^+(\mathcal{R}_{\triangle^\cup}\mathsf{T})$, so already $(U_i) \in \mathcal{R}_{\triangle^\cup}\mathsf{T}$ by (3).

Finally, we prove (3). Let $\mathcal{Y} \in \wp^+(\mathcal{R}_{\triangle^\cup}\mathsf{T})$ be arbitrary. We need to show $\bigcup\mathcal{Y} \in \mathcal{R}_{\triangle^\cup}\mathsf{T}$. By definition of $\triangle^\cup$, every $(U_i) \in \mathcal{Y}$ is of the form $\bigcup\mathcal{X}$ for some non-empty set $\mathcal{X} \subseteq \mathcal{R}_\triangle\mathsf{T}$ of tuples. Clearly $\mathcal{X}$ contains only tuples (S_i) such that $(S_i) \subseteq (U_i)$. But then

$$(U_i) = \bigcup\mathcal{X} \subseteq \bigcup\left\{ (S_i) \in \mathcal{R}_\triangle\mathsf{T} \;\middle|\; (S_i) \subseteq (U_i) \right\} \subseteq (U_i),$$

so as a consequence, $\bigcup\mathcal{X} = \bigcup\left\{ (S_i) \in \mathcal{R}_\triangle\mathsf{T} \;\middle|\; (S_i) \subseteq (U_i) \right\}$, which in turn implies

$$\begin{aligned}\bigcup\mathcal{Y} &= \bigcup\left\{ \bigcup\left\{ (S_i) \in \mathcal{R}_\triangle\mathsf{T} \mid (S_i) \subseteq (U_i) \right\} \;\middle|\; (U_i) \in \mathcal{Y} \right\}\\ &= \bigcup\left\{ (S_i) \in \mathcal{R}_\triangle\mathsf{T} \;\middle|\; \exists (U_i) \in \mathcal{Y} : (S_i) \subseteq (U_i) \right\}.\end{aligned}$$

Let $\mathcal{X}^\star := \left\{ (S_i) \in \mathcal{R}_\triangle\mathsf{T} \;\middle|\; \exists (U_i) \in \mathcal{Y} : (S_i) \subseteq (U_i) \right\}$. Then $\bigcup\mathcal{Y} = \bigcup\mathcal{X}^\star$, and $\mathcal{X}^\star \in \wp(\mathcal{R}_\triangle\mathsf{T})$. It remains to show that $\bigcup\mathcal{Y}$ is actually a *non-empty* union of tuples in $\mathcal{R}_\triangle\mathsf{T}$, which is not clear if $\bigcup\mathcal{X}^\star = \bigcup\mathcal{Y} = \emptyset^r$. But in this case $\mathcal{Y} = \{\emptyset^r\}$ (by choice of $\mathcal{Y}$, $\mathcal{Y} \neq \emptyset$), and hence $\emptyset^r \in \mathcal{R}_{\triangle^\cup}\mathsf{T}$, which in turn requires that also $\emptyset^r \in \mathcal{R}_\triangle\mathsf{T}$. ☐

Interestingly, relaxation preserves our previous classification of connectives. By definition, the relaxation is an operator, but we can show that it is indeed flatness equivalent, and furthermore the relaxation of a transversal is again a transversal.

Proposition 3.68. *An operator $\triangle$ and its relaxation $\triangle^\cup$ are flatness equivalent.*

Proof. To show flatness equivalence, we use its order-theoretic characterization (Theorem 3.56). For this, we have to show that $\mathcal{R}_\triangle\{s\}$ and $\mathcal{R}_{\triangle^\cup}\{s\}$ are mutually coinitial for arbitrary singletons $\{s\}$. By Theorem 3.67, $\mathcal{R}_\triangle\{s\} \subseteq \mathcal{R}_{\triangle^\cup}\{s\}$, so $\mathcal{R}_{\triangle^\cup}\{s\}$ is trivially coinitial in $\mathcal{R}_\triangle\{s\}$. For the other direction, let $(S_i) \in \mathcal{R}_{\triangle^\cup}\{s\}$. Then $(S_i) = \bigcup \mathcal{X}$ for some non-empty $\mathcal{X} \subseteq \mathcal{R}_\triangle\{s\}$. In consequence, there is $(U_i) \in \mathcal{R}_\triangle\{s\}$ such that $(U_i) \subseteq (S_i)$. As (S_i) is arbitrary, $\mathcal{R}_\triangle\{s\}$ is also coinitial in $\mathcal{R}_{\triangle^\cup}\{s\}$. □

Theorem 3.69. *The relaxation of a transversal is a transversal.*

Proof. Let $\triangle$ be a transversal. We have to show that

$$\mathcal{R}_{\triangle^\cup}T = \left\{ \bigcup_{s\in T} g(s) \;\middle|\; g \in \prod_{s\in T} \mathcal{R}_{\triangle^\cup}\{s\} \right\} \tag{$\star$}$$

for an arbitrary team T. We prove both directions of $(\star)$.

"$\subseteq$": Suppose $(S_i) \in \mathcal{R}_{\triangle^\cup}T$. Then $(S_i) = \bigcup \mathcal{X}$ for some non-empty $\mathcal{X} \subseteq \mathcal{R}_\triangle T$. Since $\triangle$ is a transversal, every $(U_i) \in \mathcal{X}$ is the image of a choice function from T, i.e., $(U_i) = \bigcup_{s\in T} f(s)$ for some $f \in \prod_{s\in T} \mathcal{R}_\triangle\{s\}$. The idea to show that the whole tuple (S_i) is the image of some choice function as well is to form the "union" of all previous choice functions that do not exceed (S_i). Formally, we choose

$$g(s) := \bigcup \left\{ (V_i) \in \mathcal{R}_\triangle\{s\} \;\middle|\; (V_i) \subseteq (S_i) \right\}.$$

First, we show that this is an actual choice function for $\triangle^\cup$, i.e., for all s, $g(s) \in \mathcal{R}_{\triangle^\cup}\{s\}$. By definition of $\triangle^\cup$, $\mathcal{R}_{\triangle^\cup}\{s\}$ contains any non-empty union of tuples of $\mathcal{R}_\triangle\{s\}$, and hence $g(s)$, provided some $(V_i) \subseteq (S_i)$ exists in $\mathcal{R}_\triangle\{s\}$ such that the above union is actually non-empty. Let $(U_i) \in \mathcal{X}$. As (U_i) itself is the image of a choice function from T, that is, $(U_i) = \bigcup_{s\in T} f(s)$, the tuple $(V_i) := f(s) \subseteq (U_i) \subseteq (S_i)$ is in $\mathcal{R}_\triangle\{s\}$.

Next, we prove that $(S_i) = \bigcup_{s\in T} g(s)$. For the $\supseteq$-direction, obviously $g(s) \subseteq (S_i)$ for all s. For the $\subseteq$-direction, we show that $(U_i) \subseteq \bigcup_{s\in T} g(s)$ for all $(U_i) \in \mathcal{X}$. As before, given (U_i), there is f such that $(U_i) = \bigcup_{s\in T} f(s)$. By definition of a transversal, $f(s) \in \mathcal{R}_\triangle\{s\}$ and $f(s) \subseteq (U_i)$ for all $s \in T$, hence $f(s) \subseteq g(s)$. Consequently, $(U_i) = \bigcup_{s\in T} f(s) \subseteq \bigcup_{s\in T} g(s)$.

"$\supseteq$": Let $g \in \prod_{s\in T} \mathcal{R}_{\triangle^\cup}\{s\}$ and $(U_i) := \bigcup_{s\in T} g(s)$. To show $(U_i) \in \mathcal{R}_{\triangle^\cup}T$, we demonstrate that (U_i) is a non-empty union of tuples in $\mathcal{R}_\triangle T$, which suffices by definition of $\triangle^\cup$. First, for arbitrary $u \in U_i$, $i \in [r]$, we identify a tuple $(V_i) \in \mathcal{R}_\triangle T$ such that $u \in V_i$ and $(V_i) \subseteq (U_i)$. This shows that (U_i) can be written as the union of such tuples. The union can be empty in case $(U_i) = \emptyset^r$, but we handle that case below. Let now $i \in [r], u \in U_i$. As $(U_i) = \bigcup_{s\in T} g(s)$, there must exist $s \in T$ such that $(\emptyset^{i-1}, \{u\}, \emptyset^{r-i}) \subseteq g(s)$. However, $g(s) \in \mathcal{R}_{\triangle^\cup}\{s\}$, so $g(s)$ itself is a non-empty union of tuples in $\mathcal{R}_\triangle\{s\}$. This means that there is $(V_i) \in \mathcal{R}_\triangle\{s\}$ with $u \in V_i$ and $(V_i) \subseteq g(s) \subseteq (U_i)$, as desired. Finally, we consider the case where $(U_i) = \emptyset^r$. Then $g(s)$ must be $\emptyset^r$ for all s, too. In particular, $\emptyset^r \in \mathcal{R}_{\triangle^\cup}\{s\}$ for all $s \in T$, which is only possible if $\emptyset^r \in \mathcal{R}_\triangle\{s\}$. Recall that $\triangle$ is a transversal, so this results in $\emptyset^r$ also being in $\mathcal{R}_\triangle T$, and ultimately in $\mathcal{R}_{\triangle^\cup}T$ by definition of $\triangle^\cup$. □

Example 3.70. The team connectives considered so far suggest that always either for $\triangle$ or its dual the lax and strict connectives coincide (such as with $\wedge$, $\Box$ and $\forall$). However, it is also possible that *both* sides have distinct strict and lax variants, even for unary operators. Consider the operators $\otimes_s$, $\otimes_l$, $\odot_s$ and $\odot_l$ satisfying

$$\begin{aligned}
&\mathcal{R}_{\otimes_s}\{s\} = \{\{s_1\}, \{s_2, s_3\}\}\\
&\mathcal{R}_{\otimes_l}\{s\} = \{\{s_1\}, \{s_2, s_3\}, \{s_1, s_2, s_3\}\}\\
&\mathcal{R}_{\odot_s}\{s\} = \{\{s_1, s_2\}, \{s_1, s_3\}\}\\
&\mathcal{R}_{\odot_l}\{s\} = \{\{s_1, s_2\}, \{s_1, s_3\}, \{s_1, s_2, s_3\}\}.
\end{aligned}$$

Then by the order-theoretic (hitting vector) characterization, $\otimes_s$ and $\otimes_l$ are strict resp. lax duals of $\odot_s$ and $\odot_l$, while $\odot_s$ and $\odot_l$ are strict resp. lax duals of both $\otimes_s$ and $\otimes_l$.

3.6 Strict and lax standard transversals

We saw that transversals are a natural and sensible restriction of operators that reduce the work of defining an operator $\triangle$ to the simpler task of defining $\mathcal{R}_\triangle\{s\}$ for singletons $\{s\}$. Nevertheless, there is still some degree of freedom in defining $\mathcal{R}_\triangle\{s\}$, for example lax and strict variants of the same operator (cf. Example 3.52).

We propose a canonical definition on how to "teamify" arbitrary operators. In what follows, let $\triangle\colon (\wp X)^r \to \wp X$ be a (classical) operator.

Definition 3.71. The *strict standard transversal* of $\triangle$ is the transversal $\nabla\colon (\wp\wp X)^r \to \wp\wp X$ defined by

$$\mathcal{R}_\nabla\{s\} := \left\{ \left(\{s_i\}\right) \;\middle|\; (s_i) \in \mathcal{R}_\triangle s \right\}.$$

The *strict dual standard transversal* is the transversal $\nabla'\colon (\wp\wp X)^r \to \wp\wp X$ defined by

$$\mathcal{R}_{\nabla'}\{s\} := \left\{ \left(\{s_i \mid (s_j) \in f^{-1}(i)\}\right)_{i\in[r]} \;\middle|\; f \in \prod_{(s_j)\in\mathcal{R}_\triangle s} [r] \right\}.$$

The *lax* (*dual*) *standard transversal* of $\triangle$ is the relaxation of its strict (dual) standard transversal.

This definition seems complicated, but for unary $\triangle$ boils down to:

$$\begin{aligned}
\mathcal{R}_\nabla\{s\} &= \wp^1(\mathcal{R}_\triangle s)\\
\mathcal{R}_{\nabla^\cup}\{s\} &= \wp^+(\mathcal{R}_\triangle s)\\
\mathcal{R}_{\nabla'}\{s\} = \mathcal{R}_{\nabla'^\cup}\{s\} &= \left\{\mathcal{R}_\triangle s\right\}
\end{aligned}$$

Let us explain the intuition behind the general definition: If s' is a successor of s, then $\{s'\}$ should be a "successor team" of $\{s\}$ (and likewise for higher arities). For the dual standard transversal, essentially we want ∇' to be a weak dual of ∇, as we then obtain (by Theorem 3.37) a teamification of $\triangle^\complement$ for free. For this, we follow the characterization of weak duality via hitting vectors (Theorem 3.50). To form a hitting vector of the family

$\mathcal{R}_\nabla\{s\}$, for every tuple $(S_i) \in \mathcal{R}_\nabla\{s\}$ we pick a number $i \in [r]$ and include an element of the i-th component, which ensures that (S_i) is hit. Indeed, every choice function $f \in \prod_{(s_i)\in\mathcal{R}_\triangle s}[r]$ encodes a hitting vector. Finally, $\{s_i \mid (s_j)_{j\in[r]} \in f^{-1}(i)\}$ is simply the set of all elements that have been selected from the i-th components, and thus is the i-th component of the hitting vector.

Therefore, as shown in Table 3.1:

Corollary 3.72. *The operators* $\wedge, \Diamond, \exists x$ (*resp.* $\wedge, \Diamond, \dot{\exists} x$) *are the lax* (*resp. strict*) *standard transversals of* $\wedge, \Diamond, \exists x$. *The operators* $\vee, \Box, \forall x$ (*resp.* $\dot{\vee}, \Box, \forall x$) *are the lax* (*resp. strict*) *dual standard transversals of* $\wedge, \Diamond, \exists x$.

In what follows, we assume $\triangle\colon (\wp X)^r \to \wp X$ and $\nabla, \nabla'\colon (\wp\wp X)^r \to \wp\wp X$. We still have to prove that this definition actually produces teamifications of $\triangle$ and $\triangle^{\complement}$.

Theorem 3.73. *Let* $\triangle$ *be an operator. The strict and lax standard transversals of* $\triangle$ *are teamifications of* $\triangle$. *The strict and lax dual standard transversals of* $\triangle$ *are teamifications of* $\triangle^{\complement}$.

For the proof, we require a lemma:

Lemma 3.74. *A flatness preserving operator* ∇ *is the teamification of the connective* $\triangle$ *defined by* $\triangle(T_1, \ldots, T_r) := \{ s \mid \exists (S_i) \in \mathcal{R}_\nabla\{s\} : (S_i) \subseteq (T_i) \}$.

Proof. We have to show that $\wp\triangle(T_1, \ldots, T_r) = \nabla(\wp T_1, \ldots, \wp T_r)$. As ∇ is flatness preserving, $\nabla(\wp T_1, \ldots, \wp T_r)$ is a power set. Hence it suffices to show that $s \in \triangle(T_1, \ldots, T_r)$ iff $\{s\} \in \nabla(\wp T_1, \ldots, \wp T_r)$. But these are both equivalent to $\exists (S_i) \in \mathcal{R}_\nabla\{s\} : (S_i) \subseteq (T_i)$. ☐

Proof of Theorem 3.73. Let ∇ be the strict standard transversal of $\triangle$. First of all, the relaxation of ∇ is a transversal (Theorem 3.69), and flatness equivalent to it (Proposition 3.68). Both preserve flatness (Theorem 3.62) and hence are teamifications of the same connective $\triangle'$ (Theorems 3.34 and 3.58). For this reason, if we show that $\triangle = \triangle'$, then we prove the statement for both the strict and lax standard transversal. We show $\triangle = \triangle'$ as follows:

$$\begin{aligned}
& s \in \triangle(T_1, \ldots, T_r) && \\
\Leftrightarrow\ & \exists (s_i) \in \mathcal{R}_\triangle s : \forall i : s_i \in T_i && (\triangle \text{ is operator})\\
\Leftrightarrow\ & \exists (S_i) \in \mathcal{R}_\nabla\{s\} : (S_i) \subseteq (T_i) && (\text{def. standard transversal})\\
\Leftrightarrow\ & s \in \triangle'(T_1, \ldots, T_r) && (\text{Lemma 3.74})
\end{aligned}$$

Next, we proceed with the dual standard transversal ∇'. By the same arguments as above, the strict variant suffices. Again, ∇' is the teamification of some connective $\triangle''$. We have the equivalences:

$$\begin{aligned}
s \in \triangle^{\complement}(T_1, \ldots, T_r) &\Leftrightarrow \forall (s_i) \in \mathcal{R}_\triangle s : \exists i \in [r] : s_i \in T_i && (\triangle \text{ is operator})\\
s \in \triangle''(T_1, \ldots, T_r) &\Leftrightarrow \exists (S_i) \in \mathcal{R}_{\nabla'}\{s\} : (S_i) \subseteq (T_i) && (\text{Lemma 3.74})
\end{aligned}$$

So we need to show for all s that

$$\Big(\forall (s_i) \in \mathcal{R}_\triangle s : \exists i \in [r] : s_i \in T_i\Big) \;\Leftrightarrow\; \Big(\exists (S_i) \in \mathcal{R}_{\nabla'}\{s\} : (S_i) \subseteq (T_i)\Big).$$

For "$\Rightarrow$", assume that $\forall (s_i) \in \mathcal{R}_\triangle s : \exists i \in [r] : s_i \in T_i$. Then there is a choice function f mapping each element of $\mathcal{R}_\triangle s$ to a number $i := f((s_i)) \in [r]$, formally $f \in \prod_{(s_i) \in \mathcal{R}_\triangle}[r]$, such that $f((s_i)) = i$ implies $s_i \in T_i$. By definition of ∇', then the tuple (S_i), where $S_i = \{\, s_i \mid (s_i) \in f^{-1}(i) \,\}$, is in $\mathcal{R}_\nabla\{s\}$. Also, $S_i \subseteq T_i$ for all i by the choice of f.

For "$\Leftarrow$", suppose $(S_i) \in \mathcal{R}_\nabla\{s\}$ exists such that $(S_i) \subseteq (T_i)$. Then by definition of ∇', the tuple (S_i) must result from some $f \in \prod_{(s_i) \in \mathcal{R}_\triangle s}[r]$, such that $S_i = \{s_i \mid (s_i) \in f^{-1}(i)\}$ for all $i \in [r]$. In particular, it holds that $f((s_i)) = i$ implies $s_i \in T_i$. Hence f witnesses the left-hand side of the above equivalence. □

Corollary 3.75. *The (strict or lax) dual standard transversal of $\triangle$ is a weak dual of the (strict or lax) standard transversal of $\triangle$.*

Proof. Immediately by Theorem 3.37. □

Once more, atoms are well-behaved. Recall that every nullary connective is an operator (Proposition 3.44), and, if it preserves flatness, a transversal (Theorem 3.61). This can be strengthened as follows:

Proposition 3.76. *Given a nullary connective $\triangle$, its strict and lax standard transversal coincide and are its unique teamification.*

Proof. First, for any two teamifications ∇_1, ∇_2 of $\triangle$, we have $\nabla_1 = \wp\triangle = \nabla_2$ by definition of teamification (Definition 3.30), which proves uniqueness. As both strict and lax standard transversals are teamifications by Theorem 3.73, they coincide. □

This implies that the team semantics of atomic formulas in propositional, modal or first-order logic is precisely their standard transversal.

We conclude this section with illustrations of the interplay of the various relations such as teamification, weak and strong duality, and relaxation of connectives. In Figure 3.2 we show these relations for an arbitrary operator $\triangle$. In Figures 3.3 to 3.5, this is shown specifically for propositional, modal and first-order team logic.

3.7 Quasi-flatness

In this section, we consider a generalization of flatness called *quasi-flatness*. A property is quasi-flat if it is the Boolean combination of flat properties. Some logics can only define quasi-flat properties, which has several nice implications, for instance it allows to lift a classical decision procedure to a team-semantical decision procedure (see Chapter 5). As a counter-example, the dependence atom $\mathrm{dep}(x; y)$ is not quasi-flat.

Before we come to the results, we proceed with a formal definition. As before, let A be a τ-team algebra with carrier $\wp\wp X$. Let $\mathcal{C} \subseteq \wp\wp X$, i.e., $\mathcal{C}$ is a collection of team

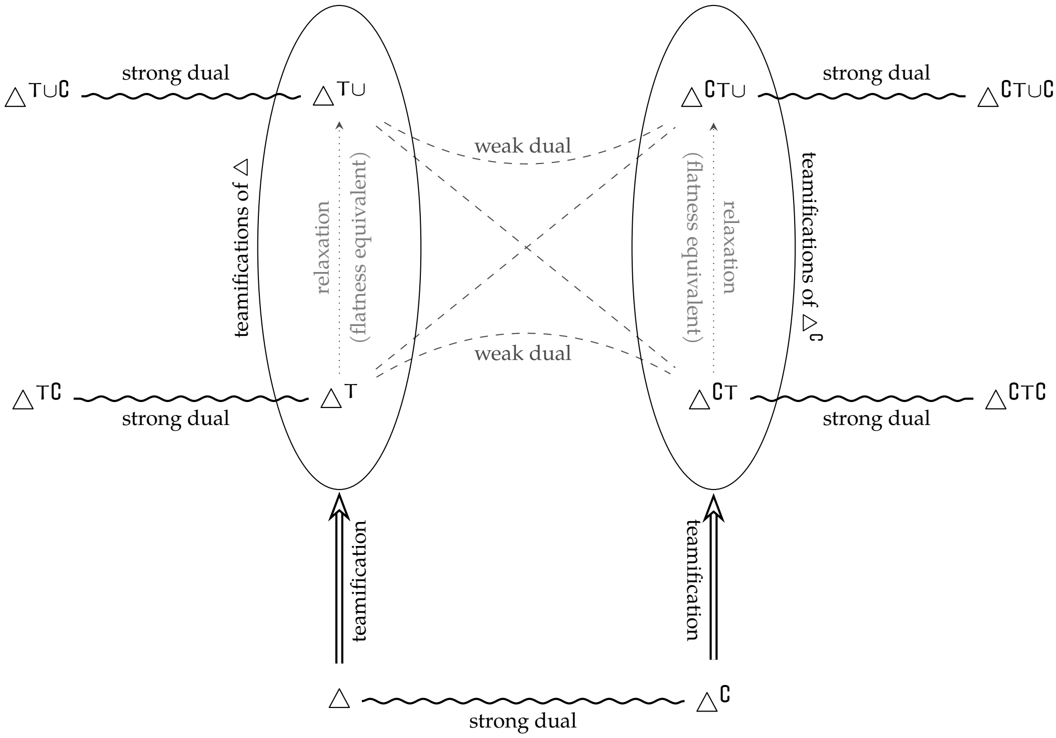


Figure 3.2: An operator △, its strong dual △ᶜ, the standard transversals △ᵀ, △ᶜᵀ, the duals △ᵀᶜ, △ᶜᵀᶜ of those, and the relaxations △ᵀ∪, △ᶜᵀ∪ with their duals.

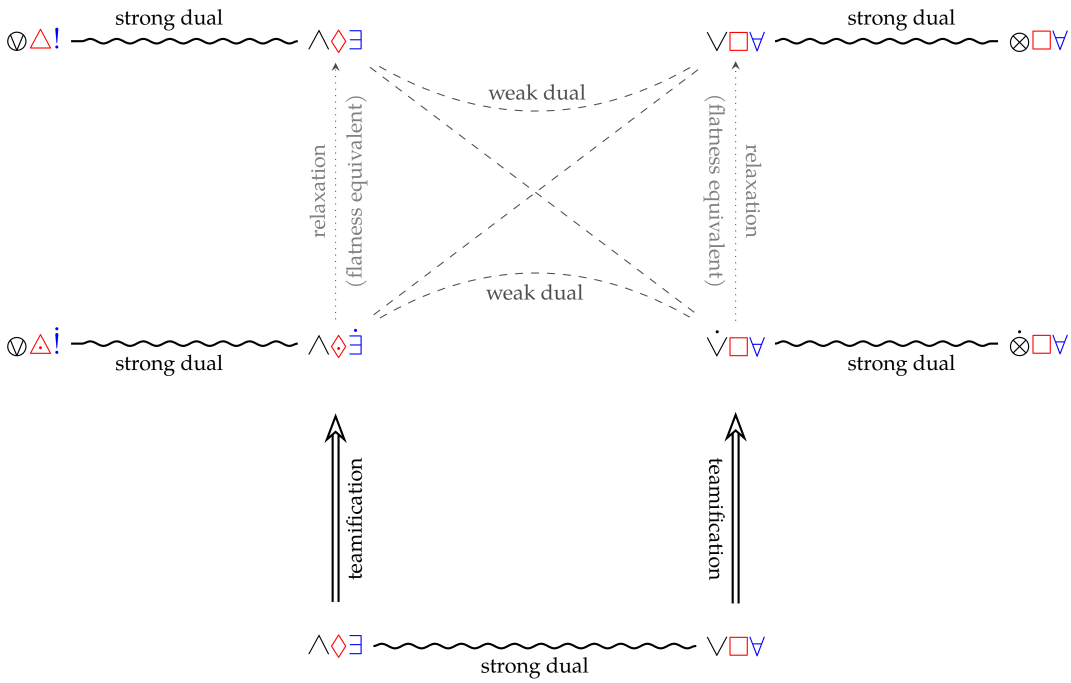


Figure 3.3: Standard teamifications of common team-logical connectives.

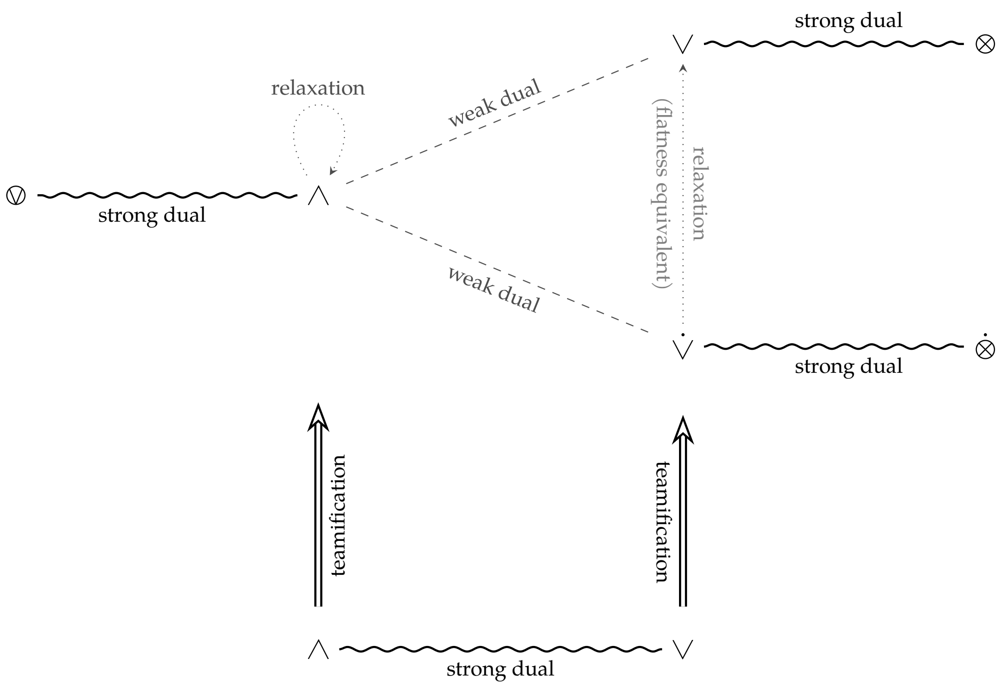


Figure 3.4: Standard teamifications of Boolean connectives.

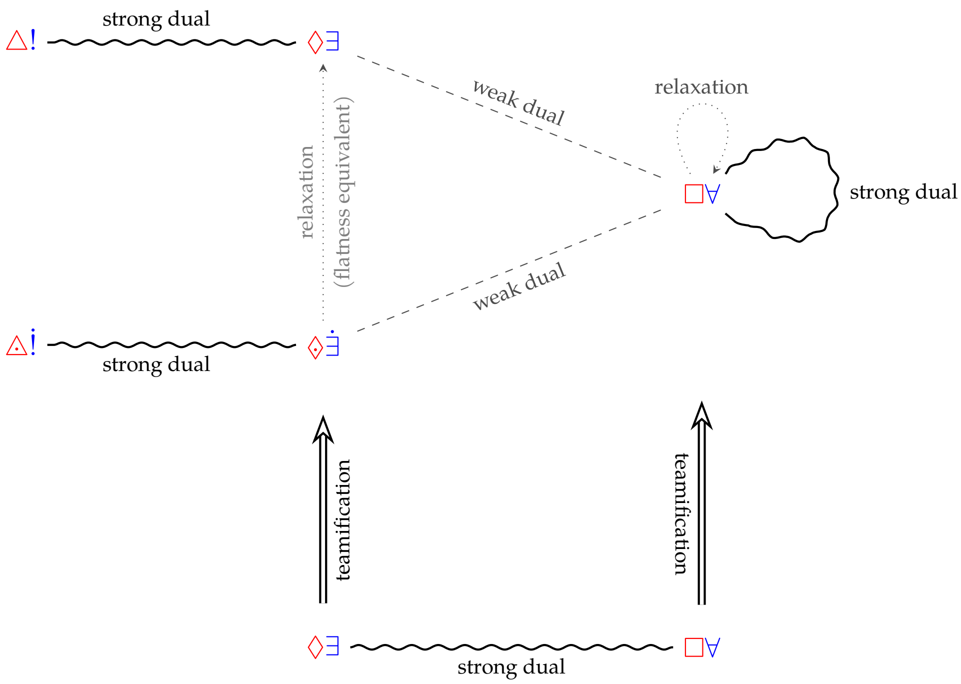


Figure 3.5: Standard teamifications of modal and first-order quantifiers.

properties. The *Boolean closure* $\mathcal{B}(\mathcal{C})$ of $\mathcal{C}$ is $\mathcal{B}(\mathcal{C}) := \bigcup_{n\in\mathbb{N}} \mathcal{C}_n$ with $\mathcal{C}_0 := \mathcal{C}$ and $\mathcal{C}_{n+1} := \{\mathcal{T}_1 \cap \mathcal{T}_2,\ \mathcal{T}_1 \cup \mathcal{T}_2,\ \wp X \setminus \mathcal{T}_1 \mid \mathcal{T}_1, \mathcal{T}_2 \in \mathcal{C}_n\}$.

If for example $\mathcal{C} = \{[\![\varphi]\!]^A, [\![\psi]\!]^A\}$, then $\mathcal{B}(\mathcal{C})$ contains all properties that are definable by a finite combination of φ and ψ with $\wedge, \otimes$ and $\sim$.

The collection of all *flat* properties is written $\mathcal{C}_{\text{flat}}$.

Definition 3.77 (Quasi-flatness)**.** A property $\mathcal{T} \in \wp\wp X$ is *quasi-flat* if it is in $\mathcal{B}(\mathcal{C}_{\text{flat}})$.

Obviously flatness implies quasi-flatness. Quasi-flat properties permit a "disjunctive normal form":

Proposition 3.78. *Every quasi-flat $\mathcal{T} \in \wp\wp X$ is of the form $\bigcup_{i\in[n]} \left(\mathcal{P}_i \setminus \bigcup_{j\in J_i} \mathcal{Q}_{i,j}\right)$ for some finite $n \in \mathbb{N}$, finite sets $J_i \subseteq \mathbb{N}$ and flat team properties $\mathcal{P}_i, \mathcal{Q}_{i,j} \in \wp\wp X$.*

Proof. By assumption, $\mathcal{T}$ is a Boolean combination of flat team properties. By de Morgan's laws and distributive laws, it can be written as

$$\bigcup_{i\in[n]} \left(\bigcap_{k\in K_i} \mathcal{P}_{i,k} \cap \bigcap_{k\in K'_i} \mathcal{Q}^{\complement}_{i,k} \right)$$

for flat $\mathcal{P}_{i,k}, \mathcal{Q}_{i,k}$. As the the intersection of flat properties is flat (the intersection of zero properties being $\top$), we obtain

$$\mathcal{P} = \bigcup_{i \in [n]} \left(\mathcal{P}_i \setminus \bigcup_{j \in J_i} \mathcal{Q}_{i,j} \right)$$

for finite n and J_i and flat properties $\mathcal{P}_i, \mathcal{Q}_{i,j}$. □

How do we show that a property is quasi-flat? Due to the compositional semantics, we are interested in connectives that preserve quasi-flatness, analogously to flatness preserving connectives (Definition 3.17).

Definition 3.79. Let $\triangle\colon (\wp\wp X)^r \to \wp\wp X$. Then $\triangle$ *preserves quasi-flatness* if $\triangle(\mathcal{T}_1, \ldots, \mathcal{T}_r)$ is quasi-flat for all quasi-flat properties $\mathcal{T}_1, \ldots, \mathcal{T}_r \subseteq \wp X$.

Indeed all properties definable in $\mathsf{PL}(\sim)$, $\mathsf{ML}(\sim)$ and $\mathsf{FO}(\sim)$ are quasi-flat because their connectives fall under the above definition. For these logics, this boils down to the following "distributive laws", which have been proven in [100]:

$$(\alpha \wedge \bigwedge_{i=1}^{n} \mathsf{E}\beta_i) \vee (\alpha' \wedge \bigwedge_{i=1}^{m} \mathsf{E}\beta'_i) \equiv (\alpha \vee \alpha') \wedge \bigwedge_{i=1}^{n} \mathsf{E}(\alpha \wedge \beta_i) \wedge \bigwedge_{i=1}^{m} \mathsf{E}(\alpha' \wedge \beta'_i)$$

for all flat formulas $\alpha, \alpha', \beta_1, \ldots, \beta_n, \beta'_1, \ldots, \beta'_n$,

$$\Diamond(\alpha \wedge \bigwedge_{i=1}^{n} \mathsf{E}\beta_i) \equiv \Diamond\alpha \wedge \bigwedge_{i=1}^{n} \mathsf{E}\Diamond(\alpha \wedge \beta_i) \qquad \Box(\alpha \wedge \bigwedge_{i=1}^{n} \mathsf{E}\beta_i) \equiv \Box\alpha \wedge \bigwedge_{i=1}^{n} \mathsf{E}\Diamond(\alpha \wedge \beta_i)$$

for all flat formulas $\alpha, \beta_1, \ldots, \beta_n$, and likewise for first-order team logic,

$$\exists x(\alpha \wedge \bigwedge_{i=1}^{n} \mathsf{E}\beta_i) \equiv \exists x\alpha \wedge \bigwedge_{i=1}^{n} \mathsf{E}\exists x(\alpha \wedge \beta_i) \qquad \forall x(\alpha \wedge \bigwedge_{i=1}^{n} \mathsf{E}\beta_i) \equiv \forall x\alpha \wedge \bigwedge_{i=1}^{n} \mathsf{E}\exists x(\alpha \wedge \beta_i).$$

In each case, the disjunction, modality or quantifier outside a quasi-flat formula can be distributed over the conjunction in such a way that we obtain a Boolean combination of flat formulas. (They all distribute also over disjunction because they are operators.) Next, in the main theorem of this section, we show that in fact *all* (lax) standard transversals of arbitrary arity permit distributive laws like the above, and hence preserve quasi-flatness.

Theorem 3.80. *All lax standard transversals and lax dual standard transversals preserve quasi-flatness.*

The proof of Theorem 3.80 will be split into a series of lemmas below.

Lemma 3.81 (Union closure for lax transversals)**.** *Let ∇ be a lax transversal. Then $(S_i) \in \mathcal{R}_\nabla T_1$ and $(U_i) \in \mathcal{R}_\nabla T_2$ implies $(S_i \cup U_i) \in \mathcal{R}_\nabla(T_1 \cup T_2)$.*

Proof. Let T_1, T_2 be arbitrary teams and let $(S_i)_{i\in[r]} \in \mathcal{R}_\nabla T_1$ and $(U_i)_{i\in[r]} \in \mathcal{R}_\nabla T_2$. We have to show that $(S_i \cup U_i)_{i\in[r]} \in \mathcal{R}_\nabla(T_1 \cup T_2)$. First, by definition of a transversal, there are choice functions $f \in \prod_{s\in T_1} \mathcal{R}_\nabla\{s\}$ and $g \in \prod_{s\in T_2} \mathcal{R}_\nabla\{s\}$ such that for all i,

$$\begin{aligned} S_i &= \bigcup \{X_i \mid (X_1, \dots, X_r) = f(s), s \in T_1\}, \\ U_i &= \bigcup \{Y_i \mid (Y_1, \dots, Y_r) = g(s), s \in T_2\}. \end{aligned} \tag{$\star$}$$

From this, we define a new choice function h on the team $T_1 \cup T_2$ as follows:

$$h(s) := \begin{cases} f(s) & \text{if } s \in T_1 \setminus T_2 \\ g(s) & \text{if } s \in T_2 \setminus T_1 \\ f(s) \cup g(s) & \text{if } s \in T_1 \cap T_2 \end{cases}$$

To see that h is a valid choice function, in symbols $h \in \prod_{s\in T_1\cup T_2} \mathcal{R}_\nabla\{s\}$, observe that $f(s) \cup g(s)$ is an element of $\mathcal{R}_\nabla\{s\}$. This is due to the fact that by Theorem 3.67, $\mathcal{R}_\nabla T$ is closed under non-empty union for all T. Finally, we show that the tuple $(S_i \cup U_i)_{i\in[r]}$ is actually generated by h and hence is in $\mathcal{R}_\nabla T$. In other words, if

$$S_i \cup U_i = \bigcup \{Z_i \mid (Z_1, \dots, Z_r) = h(s) \mid s \in T_1 \cup T_2\}$$

holds for all i.

"$\subseteq$": Suppose $s \in S_i$ (the case $s \in U_i$ works analogously). Then by $(\star)$ there exists $s' \in T_1$ such that $f(s') = (X_1, \dots, X_r) \in \mathcal{R}_\nabla\{s'\}$ and $s \in X_i$. But $h(s') \supseteq f(s')$.

"$\supseteq$": Suppose $s \in Z_i$ for $(Z_1, \dots, Z_r) = h(s')$ and some $s' \in T_1 \cup T_2$. We show $s \in S_i \cup U_i$.

- If $s' \in T_1 \setminus T_2$, then $Z_i = X_i$, where $(X_i)_{i\in[r]} = f(s')$.
- If $s' \in T_2 \setminus T_1$, then $Z_i = Y_i$, where $(Y_i)_{i\in[r]} = g(s')$.
- If $s' \in T_1 \cap T_2$, then $Z_i = X_i \cup Y_i$ by definition of h, where $(X_i)_{i\in[r]} = f(s')$ and $(Y_i)_{i\in[r]} = g(s')$. Consequently, $s \in X_i$ or $s \in Y_i$. □

In all cases, $s \in S_i \cup U_i$ due to $(\star)$.

We use the abbreviation $E\mathcal{P} := \sim\neg\mathcal{P}$, so $T \in E\mathcal{P}$ if some $\{s\} \subseteq T$ is in $\mathcal{P}$. Also recall that NE is the property of all non-empty teams, and $\mathcal{P}_1 \vee \mathcal{P}_2 = \{T_1 \cup T_2 \mid T_1 \in \mathcal{P}_1, T_2 \in \mathcal{P}_2\}$.

Lemma 3.82. *For all flat properties $\mathcal{P}, \mathcal{Q}_1, \dots, \mathcal{Q}_n$,*

$$\mathcal{P} \vee \bigvee_{i=1}^{n} (\mathcal{P} \cap \mathcal{Q}_i \cap \textsc{ne}) \;=\; \mathcal{P} \cap \bigcap_{i=1}^{n} E\mathcal{Q}_i$$

Proof. "$\subseteq$": If $T \in \mathcal{P} \vee \bigvee_{i=1}^{n} (\mathcal{P} \cap \mathcal{Q}_i \cap \textsc{ne})$. The splitting disjunction is witnessed by teams $T_0 \cup \dots \cup T_n = T$ such that $T_0 \in \mathcal{P}$ and, for $i > 0$, $T_i \in \mathcal{P} \cap \mathcal{Q}_i$ and $T_i \neq \emptyset$. Hence $T_i \in E\mathcal{Q}_i$. Now by union closure, $T \in \mathcal{P}$, and by downward closure, $T_i \in E\mathcal{Q}_i \Rightarrow T_i \notin \neg\mathcal{Q}_i \Rightarrow T \notin \neg\mathcal{Q}_i \Rightarrow T \in E\mathcal{Q}_i$.

"$\supseteq$": Suppose $T \in \mathcal{P} \cap \bigcap_{i=1}^{n} E\mathcal{Q}_i$. The $E\mathcal{Q}_i$ are witnessed by elements $s_1, \ldots, s_n \in T$ such that $\{s_i\} \in \mathcal{Q}_i$. Also, by downward closure, $\{s_i\} \in \mathcal{P}$. Consequently, $\{s_i\} \in \mathcal{P} \cap \mathcal{Q}_i \cap \textsc{ne}$. Now $T = T \cup \{s_1\} \cup \cdots \cup \{s_n\}$ witnesses $T \in \mathcal{P} \vee \bigvee_{i=1}^{n} (\mathcal{P} \cap \mathcal{Q}_i \cap \textsc{ne})$. □

The remainder of the proof now consists of two lemmas, one for standard transversals and one for dual standard transversals.

Lemma 3.83. *Lax standard transversals preserve quasi-flatness.*

Proof. Let $\nabla^{\cup}$ be the relaxation of the strict standard transversal ∇ of an operator $\triangle$. Let $\mathcal{T}_1, \ldots, \mathcal{T}_r$ be quasi-flat properties. We have to show that $\nabla^{\cup}(\mathcal{T}_1, \ldots, \mathcal{T}_r)$ is quasi-flat as well, i.e., equivalent to a Boolean combination of flat properties. By assumption, each $\mathcal{T}_i$ is equivalent to a Boolean combination of flat properties. By Proposition 3.78, there are finite $I, J_i, K_{(i,j)} \subseteq \mathbb{N}$ and flat properties $\mathcal{P}_{(i,j)}, \mathcal{Q}_{(i,j,k)}$ such that

$$\mathcal{T}_i = \bigcup_{j \in J_i} \left(\mathcal{P}_{(i,j)} \cap \bigcap_{k \in K_{(i,j)}} E\mathcal{Q}_{(i,j,k)} \right).$$

Next, recall that operators distribute over Boolean disjunction (Proposition 3.40), so it suffices to establish that properties of the form

$$\mathcal{V} = \nabla^{\cup} \left(\mathcal{P}_1 \cap \bigcap_{j \in J_1} E\mathcal{Q}_{(1,j)}, \ldots, \mathcal{P}_r \cap \bigcap_{j \in J_r} E\mathcal{Q}_{(r,j)} \right)$$

are quasi-flat. To prove this, we show that $\mathcal{V}$ is equivalent to a Boolean combination of flat formulas, viz.

$$\mathcal{V}' = \nabla^{\cup}(\mathcal{P}_1, \ldots, \mathcal{P}_r) \cap \bigcap_{i \in [r], j \in J_i} E\nabla^{\cup}(\mathcal{P}_1, \ldots, \mathcal{P}_{i-1}, \mathcal{P}_i \cap \mathcal{Q}_{(i,j)}, \mathcal{P}_{i+1}, \ldots, \mathcal{P}_r).$$

As E is short for $\sim\neg$, $\mathcal{V}'$ is a Boolean combination of flat properties. We prove that it equals $\mathcal{V}$.

$\mathcal{V} \subseteq \mathcal{V}'$: Let $T \in \mathcal{V}$ via $(S_1, \ldots, S_r) \in \mathcal{R}_{\nabla^{\cup}} T$ such that $S_i \in \mathcal{P}_i \cap \bigcap_{j \in J_i} E\mathcal{Q}_{(i,j)}$ for all $i \in [r]$. First of all, this witnesses $T \in \nabla^{\cup}(\mathcal{P}_1, \ldots, \mathcal{P}_r)$, as $S_i \in \mathcal{P}_i$. For the other $\cap$-conjuncts, fix $i \in [r]$ and $j \in J_i$. There exists $s' \in S_i$ such that $\{s'\} \in \mathcal{P}_i \cap \mathcal{Q}_{(i,j)}$. Because $\nabla^{\cup}$ is a transversal, there must exist a corresponding $s \in T$ with tuple $(U_k)_{k \in [r]} \in \mathcal{R}_{\nabla^{\cup}}\{s\}$ such that $s' \in U_i$ and $(U_k) \subseteq (S_k)$. Because (U_k) is a union of tuples in $\mathcal{R}_{\nabla}\{s\}$, there must exist some $(V_k) \in \mathcal{R}_{\nabla}\{s\}$ as well such that $s' \in V_i$ and $(V_k) \subseteq (U_k)$. By definition of a strict standard transversal (Definition 3.71), each V_k is a singleton, which implies $V_i = \{s'\}$ and therefore $V_i \in \mathcal{P}_i \cap \mathcal{Q}_{(i,j)}$. As also $V_k \in \mathcal{P}_k$ for all k, $\{s\} \in \nabla(\mathcal{P}_1, \ldots, \mathcal{P}_{i-1}, \mathcal{P}_i \cap \mathcal{Q}_{(i,j)}, \mathcal{P}_{i+1}, \ldots, \mathcal{P}_r)$. Since ∇ and $\nabla^{\cup}$ are flatness equivalent (Proposition 3.68), we conclude $T \in E\nabla^{\cup}(\mathcal{P}_1, \ldots, \mathcal{P}_{i-1}, \mathcal{P}_i \cap \mathcal{Q}_{(i,j)}, \mathcal{P}_{i+1}, \ldots, \mathcal{P}_r)$.

$\mathcal{V}' \subseteq \mathcal{V}$: Assume $T \in \mathcal{V}'$. First of all, $T \in \nabla^{\cup}(\mathcal{P}_1, \ldots, \mathcal{P}_r)$ implies there is a tuple $(S_i)_{i\in[r]} \in \mathcal{R}_{\nabla^{\cup}}T$ such that $S_i \in \mathcal{P}_i$ for all $i \in [r]$. Likewise, by the E-conjuncts, and flatness equivalence of ∇ and $\nabla^{\cup}$, for every $i \in [r], j \in J_i$ there exists $s_{(i,j)} \in T$ such that

$$\{s_{(i,j)}\} \in \nabla(\mathcal{P}_1, \ldots, \mathcal{P}_{i-1}, \mathcal{P}_i \cap \mathcal{Q}_{(i,j)}, \mathcal{P}_{i+1}, \ldots, \mathcal{P}_r),$$

which is witnessed by some tuple $(U_k^{(i,j)})_{k\in[r]} \in \mathcal{R}_{\nabla}\{s_{(i,j)}\}$. Every $U_k^{(i,j)}$ is a singleton (Definition 3.71), so $U_i^{(i,j)} \in \mathrm{NE} \cap \mathcal{P}_i \cap \mathcal{Q}_{(i,j)}$. Together, we obtain

$$T \cup \bigcup_{\substack{i\in[r]\\ j\in J_i}} \{s_{(i,j)}\} \in \nabla^{\cup}(\mathcal{P}_1, \ldots, \mathcal{P}_r) \vee \bigvee_{\substack{i\in[r]\\ j\in J_i}} \nabla(\mathcal{P}_1, \ldots, \mathcal{P}_i \cap \mathcal{Q}_{(i,j)} \cap \mathrm{NE}, \ldots, \mathcal{P}_r),$$

but since $s_{(i,j)} \in T$ for all $i \in [r], j \in J_i$,

$$T \in \nabla^{\cup}(\mathcal{P}_1, \ldots, \mathcal{P}_r) \vee \bigvee_{\substack{i\in[r]\\ j\in J_i}} \nabla(\mathcal{P}_1, \ldots, \mathcal{P}_i \cap \mathcal{Q}_{(i,j)} \cap \mathrm{NE}, \ldots, \mathcal{P}_r)$$

from which we can by $\mathcal{R}_{\nabla}T \subseteq \mathcal{R}_{\nabla^{\cup}}T$ conclude

$$T \in \nabla^{\cup}(\mathcal{P}_1, \ldots, \mathcal{P}_r) \vee \bigvee_{\substack{i\in[r]\\ j\in J_i}} \nabla^{\cup}(\mathcal{P}_1, \ldots, \mathcal{P}_i \cap \mathcal{Q}_{(i,j)} \cap \mathrm{NE}, \ldots, \mathcal{P}_r). \tag{$\star$}$$

Furthermore, by Lemma 3.81,

$$\nabla^{\cup}(\mathcal{T}_1, \ldots, \mathcal{T}_r) \vee \nabla^{\cup}(\mathcal{U}_1, \ldots, \mathcal{U}_r) \subseteq \nabla^{\cup}(\mathcal{T}_1 \vee \mathcal{U}_1, \ldots, \mathcal{T}_r \vee \mathcal{U}_r). \tag{$\star\star$}$$

As $\mathcal{P} \vee \cdots \vee \mathcal{P} = \mathcal{P}$, we obtain from $(\star)$ and $(\star\star)$ that

$$\begin{aligned} T &\in \nabla^{\cup}\left(\mathcal{P}_1 \vee \bigvee_{j\in J_1} (\mathcal{P}_1 \cap \mathcal{Q}_{1,j} \cap \mathrm{NE}), \ldots, \mathcal{P}_r \vee \bigvee_{j\in J_r} (\mathcal{P}_r \cap \mathcal{Q}_{r,j} \cap \mathrm{NE})\right) \\ &= \nabla^{\cup}\left(\mathcal{P}_1 \cap \bigcap_{j\in J_1} \mathrm{E}\mathcal{Q}_{1,j}, \ldots, \mathcal{P}_r \bigcap_{j\in J_r} \mathrm{E}\mathcal{Q}_{r,j}\right) = \mathcal{V}. \qquad \text{(Lemma 3.82)} \end{aligned}$$ □

After relaxations of standard transversals (such as $\Diamond, \exists$), we proceed with relaxations of *dual standard transversals* such as $\vee, \Box, \forall$.

Lemma 3.84. *Lax dual standard transversals preserve quasi-flatness.*

Proof. The proof is analogous to the case of standard transversals. Let $\nabla^{\cup}$ be the relaxation of the dual standard transversal ∇ of $\triangle$. As before, we need only to prove properties of the form

$$\mathcal{W} = \nabla^{\cup}\left(\mathcal{P}_1 \cap \bigcap_{j\in J_1} \mathrm{E}\mathcal{Q}_{(1,j)}, \dots, \mathcal{P}_r \cap \bigcap_{j\in J_r} \mathrm{E}\mathcal{Q}_{(r,j)}\right)$$

to be quasi-flat where $\mathcal{P}_i, \mathcal{Q}_{i,j}$ are flat. We show that $\mathcal{W}$ equals

$$\mathcal{W}' = \nabla^{\cup}(\mathcal{P}_1, \dots, \mathcal{P}_r) \cap \bigcap_{i\in[r], j\in J_i} \mathrm{E}\neg\nabla^{\cup}(\bot^{i-1}, \neg(\mathcal{P}_i \cap \mathcal{Q}_{(i,j)}), \bot^{r-i}).$$

We again proceed in two steps.

$\mathcal{W} \subseteq \mathcal{W}'$: Let $T \in \mathcal{W}$ via some $(S_k)_{k\in[r]} \in \mathcal{R}_{\nabla^{\cup}}T$. Clearly, $T \in \nabla^{\cup}(\mathcal{P}_1, \dots, \mathcal{P}_r)$. Next, fix $i \in [r]$ and $j \in J_i$. As $\nabla^{\cup}$ is a transversal (Theorem 3.69), there is again some $s \in T$ such that

$$\{s\} \in \nabla^{\cup}\left(\mathcal{P}_1, \dots, \mathcal{P}_{i-1}, \mathcal{P}_i \cap \mathrm{E}\mathcal{Q}_{(i,j)}, \mathcal{P}_{i+1}, \dots, \mathcal{P}_r\right)$$

witnessed by a tuple $(U_k) \in \mathcal{R}_{\nabla^{\cup}}\{s\}$. In particular, there is $s' \in U_i$ such that $\{s'\} \in \mathcal{P}_i \cap \mathcal{Q}_{(i,j)}$ and furthermore $(V_k)_{k\in[r]} \in \mathcal{R}_{\nabla}\{s\}$ such that $s' \in V_i$.

Now tuples in $\mathcal{R}_{\nabla}\{s\}$ are minimal hitting vectors of $\mathcal{R}_{\triangle}s$ (Definition 3.71), so for at least one tuple $(u_k)_{k\in[r]} \in \mathcal{R}_{\triangle}s$ it holds that $u_i = s'$. If ∇' denotes the strict standard transversal of $\triangle$, then consequently $\{s\} \in \nabla'(\top^{i-1}, \mathcal{P}_i \cap \mathcal{Q}_{(i,j)}, \top^{r-i})$, so by the weak duality of ∇' and $\nabla^{\cup}$ (Corollary 3.75) we obtain

$$T \in \mathrm{E}\neg\nabla^{\cup}(\bot^{i-1}, \neg(\mathcal{P}_i \cap \mathcal{Q}_{(i,j)}), \bot^{r-i}).$$

$\mathcal{W}' \subseteq \mathcal{W}$: Let $T \in \mathcal{W}'$. By assumption we have $T \in \nabla^{\cup}(\mathcal{P}_1, \dots, \mathcal{P}_r)$. Our goal is for each $i \in [r]$ and $j \in J_i$ to identify some subteam $T^{(i,j)}$ of T that satisfies

$$T^{(i,j)} \in \nabla(\mathcal{P}_1, \dots, \mathcal{P}_{i-1}, \mathcal{P}_i \cap \mathrm{E}\mathcal{Q}_{(i,j)}, \mathcal{P}_{i+1}, \dots, \mathcal{P}_r). \qquad (\star)$$

Since $T = T \cup \bigcup_{i\in[r], j\in J_i} T^{(i,j)}$, this suffices to prove $T \in \mathcal{W}$ as follows. From

$$T \in \nabla^{\cup}(\mathcal{P}_1, \dots, \mathcal{P}_r) \vee \bigvee_{i\in[r], j\in J_i} \nabla(\mathcal{P}_1, \dots, \mathcal{P}_{i-1}, \mathcal{P}_i \cap \mathrm{E}\mathcal{Q}_{(i,j)}, \mathcal{P}_{i+1}, \dots, \mathcal{P}_r)$$

and since $\mathcal{R}_{\nabla} \subseteq \mathcal{R}_{\nabla^{\cup}}$ and $\vee$ is monotone, we conclude

$$\begin{aligned} &\subseteq \nabla^{\cup}(\mathcal{P}_1, \dots, \mathcal{P}_r) \vee \bigvee_{i\in[r], j\in J_i} \nabla^{\cup}(\mathcal{P}_1, \dots, \mathcal{P}_{i-1}, \mathcal{P}_i \cap \mathrm{E}\mathcal{Q}_{(i,j)}, \mathcal{P}_{i+1}, \dots, \mathcal{P}_r) \\ &\subseteq \nabla^{\cup}(\mathcal{P}_1 \vee \bigvee_{j\in J_1} (\mathcal{P}_1 \cap \mathrm{E}\mathcal{Q}_{(1,j)}), \dots, \mathcal{P}_r \vee \bigvee_{j\in J_r} (\mathcal{P}_r \cap \mathrm{E}\mathcal{Q}_{(r,j)})) \qquad \text{(Lemma 3.81)} \end{aligned}$$

$$= \nabla^{\cup}\left(\mathcal{P}_1 \cap \bigcap_{j\in J_1} E\mathcal{Q}_{(1,j)}, \dots, \mathcal{P}_r \cap \bigcap_{j\in J_r} E\mathcal{Q}_{(r,j)}\right) = \mathcal{W}. \qquad \text{(Lemma 3.82)}$$

Next, we show that the subteams $T^{(i,j)}$ as in $(\star)$ indeed exist. Fix $i \in [r]$ and $j \in J_i$. By definition of $\mathcal{W}'$, there exists an element $s \in T$ such that

$$\{s\} \in \neg\nabla^{\cup}(\bot^{i-1}, \neg(\mathcal{P}_i \cap \mathcal{Q}_{(i,j)}), \bot^{r-i}).$$

Let again ∇' denote the strict standard transversal of $\triangle$, then by weak duality of ∇ and ∇' (Corollary 3.75)

$$\{s\} \in \nabla'(\top^{i-1}, \mathcal{P}_i \cap \mathcal{Q}_{(i,j)}, \top^{r-i}).$$

This is witnessed by some tuple in $\mathcal{R}_{\nabla'}\{s\}$, which by definition of strict standard transversal (Definition 3.71) is of the form $(\{t_1\}, \dots, \{t_r\}) \in \mathcal{R}_{\nabla'}\{s\}$ for some tuple $(t_1^{(i,j)}, \dots, t_r^{(i,j)}) \in \mathcal{R}_{\triangle}s$. In particular, $\{t_i\} \in \mathcal{P}_i \cap \mathcal{Q}_{(i,j)}$. We require t_i later.

Next, we show that $\{s\}$ is now our desired subteam $T^{(i,j)}$. By assumption we have $T \in \nabla^{\cup}(\mathcal{P}_1, \dots, \mathcal{P}_r)$, and so due to flatness $\{s\} \in \nabla(\mathcal{P}_1, \dots, \mathcal{P}_r)$. This is witnessed by some tuple $(V_k) \in \mathcal{R}_{\nabla}\{s\}$ generated by a choice function $f \in \prod_{(s_k)\in\mathcal{R}_{\triangle}s}[r]$, in the sense that $V_k = \{s_k \mid (s_1, \dots, s_r) \in f^{-1}(k)\}$. In order to satisfy $(\star)$, we need to add a singleton satisfying $\mathcal{Q}_{(i,j)}$ to V_i. To do so, we modify f to incorporate t_i as follows. Define g like f but set $g(t_1, \dots, t_r) := i$. This results in a new hitting vector (U_k) of $\mathcal{R}_{\nabla'}s$, and hence tuple in $\mathcal{R}_{\nabla}\{s\}$. Furthermore, $U_k \subseteq V_k$ for $k \neq i$, but $U_i = V_i \cup \{t_i\}$. As a consequence, $U_k \in \mathcal{P}_k$ by downward closure and $U_i \in \mathcal{P}_i \cap E\mathcal{Q}_i$ by union closure, which witnesses $(\star)$. □

With Lemmas 3.83 and 3.84, this concludes the proof of Theorem 3.80.
We use the following definition for algebras that enjoy quasi-flatness.

Definition 3.85 (Nice). Let $\{\wedge, \owedge, \sim, \neg, \top, \bot\} \subseteq \tau$. A τ-team algebra a *nice* if $\wedge, \owedge, \sim, \neg, \top$ and $\bot$ have their usual interpretations in A, and $\triangle^A$ is a lax standard transversal or lax dual standard transversal for all $\triangle \in \tau \setminus \{\wedge, \owedge, \sim, \neg, \top, \bot\}$.

This definition also covers propositional, modal and first-order team logic (in lax semantics).

Theorem 3.86. *If A is a nice τ-team algebra, then $[\![\varphi]\!]^A$ is quasi-flat for every τ-formula φ.*

Proof. By induction. We show that all connectives preserve quasi-flatness. For the Boolean operations $\wedge$, $\owedge$ and $\sim$ this is trivial. $\top$ and $\bot$ are already flat, as is $\neg\varphi$ for every formula φ, so these connectives also trivially preserve quasi-flatness. For all other connectives, this follows from Theorem 3.80. □

3.7.1 Two normal forms for quasi-flatness

We introduce two normal forms of formulas, and show that every formula that defines a quasi-flat property can be brought into such a normal form. Call a formula φ *strongly flat* if it has only flat subformulas, including itself. Recall that $\mathsf{E}\gamma = {\sim}\neg\gamma$.

Definition 3.87 (($\otimes\wedge$)-normal form and ($\otimes\vee$)-normal form)**.** Let φ be a τ-formula. We call φ in ($\otimes\wedge$)-*normal form* if

$$\varphi = \bigotimes_{i\in I}\left(\alpha_i \wedge \bigwedge_{j\in J_i} \mathsf{E}\beta_{i,j}\right)$$

for finite sets I and J_i and strongly flat τ-formulas $\alpha_i, \beta_{i,j}$ for all $i \in I$ and $j \in J_i$. It is in ($\otimes\vee$)-*normal form* if

$$\varphi = \bigotimes_{i\in I}\left(\alpha_i \vee \bigvee_{j\in J_i} (\beta_{i,j} \wedge \textsc{ne})\right)$$

for I, J_i, $\alpha_i, \beta_{i,j}$ as above.

Lemma 3.88 (Change of normal form)**.** *Let $\tau \supseteq \{\wedge, \otimes, \sim, \neg, \top, \bot\}$. Then for every τ-formula φ in ($\otimes\wedge$)-normal form (resp. ($\otimes\vee$)-normal form) there is a formula ψ in ($\otimes\vee$)-normal form (resp. ($\otimes\wedge$)-normal form) such that $[\![\varphi]\!]^{\mathcal{A}} = [\![\psi]\!]^{\mathcal{A}}$ for every nice τ-team algebra $\mathcal{A}$. Furthermore, ψ is logspace-computable from φ.*

Proof. Let $\mathcal{A}$ be a nice τ-team algebra, and let us write $T \vDash \varphi$ for $T \in [\![\varphi]\!]^{\mathcal{A}}$.

- We first assume that φ itself is in ($\otimes\wedge$)-normal form. Then every disjunct $\alpha \wedge \bigwedge_{j\in J} \mathsf{E}\beta_j$ is equivalent to the formula

  $$\alpha \vee \bigvee_{j\in J}((\alpha \wedge \beta_j) \wedge \textsc{ne}),$$

  which can be seen as follows. Suppose $T \vDash \alpha \wedge \bigwedge_{j\in J} \mathsf{E}\beta_j$. Then $\{s\} \vDash \alpha$ for all $s \in T$, and for every $j \in J$ there is $s_j \in T$ such that $\{s_j\} \vDash \mathsf{E}\beta_j$. But then also $\{s_j\} \vDash \alpha \wedge \beta_j \wedge \textsc{ne}$. Moreover, the union $T \cup \bigcup_{j\in J}\{s_j\}$ is a (lax) splitting of T. As a result, $T \vDash \alpha \vee \bigvee_{j\in J}(\alpha \wedge \beta_j \wedge \textsc{ne})$.

  Conversely, suppose $T \vDash \alpha \vee \bigvee_{j\in J}(\alpha \wedge \beta_j \wedge \textsc{ne})$. This is witnessed by a split $T = T' \cup \bigcup T_j$ such that $T' \vDash \alpha$ and $T_j \vDash \alpha \wedge \beta_j \wedge \textsc{ne}$ for each $j \in J$. Since flat properties are union closed, $T \vDash \alpha$. Furthermore, each T_j is non-empty and satisfies β_j, so some $s \in T_j$ exists such that $\{s_j\} \vDash \beta$. Hence $T \vDash \mathsf{E}\beta_j$, as desired.

- Next, we assume that φ is in ($\otimes\vee$)-normal form and again consider each disjunct

$\alpha \vee \bigvee_{j\in J}(\beta_j \wedge \textsc{ne})$. We show that the latter is equivalent to

$$\left(\alpha \vee \bigvee_{j\in J} \beta_j\right) \wedge \bigwedge_{j\in J} \mathrm{E}\beta_j.$$

Assume $T \vDash \alpha \vee \bigvee_{j\in J}(\beta_j \wedge \textsc{ne})$. Then in particular $T \vDash \alpha \vee \bigvee_{j\in J} \beta_j$, so the leftmost conjunct is clear. It remains to show $T \vDash \mathrm{E}\beta_j$ for every $j \in J$. Let $T = T' \cup \bigcup_{j\in J} T_j$ witness $T \vDash \alpha \vee \bigvee_{j\in J}(\beta_j \wedge \textsc{ne})$. Then $T_j \vDash \beta_j$, but since also $T_j \vDash \textsc{ne}$, T_j is non-empty. Consequently, $T_j \subseteq T$ witnesses $T \vDash \mathrm{E}\beta_j$.

Finally, for the other direction, assume $T \vDash (\alpha \vee \bigvee_{j\in J} \beta_j) \wedge \bigwedge_{j\in J} \mathrm{E}\beta_j$. By the first conjunct, T can be split into $T' \cup \bigcup_{j\in J} T_j$ such that $T' \vDash \alpha$ and $T_j \vDash \beta_j$ for all $j \in J$. By the second conjunct, for each $j \in J$ there exists $s_j \in T$ such that $\{s_j\} \vDash \beta_j$. We form the subteams $S_j := T_j \cup \{s_j\}$ of T. Then $T' \cup S_1 \cup \cdots \cup S_n = T$, and this split witnesses $T \vDash \alpha \vee \bigvee_{j\in J}(\beta_j \wedge \textsc{ne})$, since $T' \vDash \alpha$, S_j is non-empty, and $S_j \vDash \beta_j$ by union closure.

Since the above transformation can be applied to each disjunct in parallel, it is straightforward that ψ is logspace constructible. □

Next, we show how formulas can be translated into these normal forms.

Theorem 3.89. *Let* $\{\wedge, \circledvee, \sim, \neg, \top, \bot\} \subseteq \tau$. *There is a function* $f\colon \mathcal{F}_\tau \to \mathcal{F}_\tau$ *computable in time* $\exp_{\mathcal{O}(n)}(1)$ *such that for every* τ*-formula* φ*,*

- $f(\varphi)$ *is in* $(\circledvee\wedge)$*-normal form,*
- $[\![\varphi]\!]^A = [\![f(\varphi)]\!]^A$ *for every nice* τ*-team algebra* A*.*

Proof. Let $\varphi \in \mathcal{F}_\tau$ and $n := |\varphi|$, and let A be a nice τ-team algebra. Let us abbreviate $[\![\psi]\!]^A = [\![\psi']\!]^A$ as $\psi \equiv \psi'$, and refer to $(\circledvee\wedge)$-normal form simply as *normal form*.

We state an algorithm that computes $f(\varphi)$ by repeatedly applying Lemmas 3.83 and 3.84. More precisely, it constructs formulas $\varphi_0, \varphi_1, \varphi_2, \ldots, \varphi_n$ where $\varphi_0 := \varphi$ and $f(\varphi) := \varphi_n$ and for all $i \in [n]$,

- $\varphi_{i-1} \equiv \varphi_i$,
- φ_i contains at most $n - i$ subformulas that are not in normal form,
- $|\varphi_i| \leqslant \exp_{\mathcal{O}(1)}(|\varphi_{i-1}|)$.

It is clear than then $\varphi \equiv f(\varphi)$, that $f(\varphi)$ is in normal form, and that $|f(\varphi)| \leqslant \exp_{\mathcal{O}(n)}(n) = \exp_{\mathcal{O}(n)}(1)$. The formula φ_i is obtained from φ_{i-1} as follows. If φ_{i-1} is already in normal form, we do nothing and set $\varphi_i := \varphi_{i-1}$. Otherwise there is a minimal subformula ψ of φ_{i-1} that is not in normal form. Suppose we have a formula ψ^{nf} of size $|\psi^{\mathrm{nf}}| \leqslant \exp_{\mathcal{O}(1)} |\psi|$ in normal form such that $\psi^{\mathrm{nf}} \equiv \psi$ (we will prove this below). Then we set

$\varphi_i := \varphi_{i-1}[\psi/\psi^{nf}]$; and have that $\varphi_{i-1} \equiv \varphi_i$, that φ_i contains fewer subformulas not in normal form than φ_{i-1} (namely, ψ), and that

$$|\varphi_i| \leqslant |\varphi_{i-1}| + \underbrace{|\varphi_{i-1}|}_{\text{max. \# occurrences of } \psi \text{ in } \varphi_{i-1}} \cdot |\psi^{nf}| \leqslant |\varphi_{i-1}| \cdot \underbrace{\exp_{\mathcal{O}(1)} |\psi|}_{\text{size of } \psi^{nf} + 1} = \exp_{\mathcal{O}(1)} \underbrace{|\varphi_{i-1}|}_{\text{since } |\psi| \leqslant |\varphi_{i-1}|}.$$

Next, we explain how exactly the formula ψ^{nf} is constructed. As ψ is a minimal subformula not in normal form, it is of the form $\nabla(\psi_1, \ldots, \psi_r)$ (with necessarily $r > 0$) where $\psi_1, \ldots, \psi_r$ are in normal form. In particular, it cannot be an atom. By assumption of the theorem, ∇ is either $\neg$ or $\sim$ or a lax (dual) standard transversal of some operator $\triangle$. Below, we handle each case separately.

- If $\nabla = \neg$, we have $\psi = \neg \bigovee_{i \in I}(\alpha_i \wedge \bigwedge_{j \in J_i} \mathsf{E}\beta_{(i,j)})$ for suitable I, J_i and strongly flat $\alpha_i, \beta_{(i,j)}$. By definition of $\neg$, $T \vDash \psi$ iff for every $s \in T$ and every $i \in I$ it holds that $\{s\} \nvDash \alpha_i$ or there is $j \in J_i$ such that $\{s\} \nvDash \mathsf{E}\beta_{(i,j)}$. But on singletons, $\{s\} \nvDash \mathsf{E}\beta_{(i,j)}$ iff $\{s\} \nvDash \beta_{(i,j)}$. As a consequence, ψ is equivalent to

$$\psi^{nf} := \bigwedge_{i \in I} \neg \left(\alpha_i \wedge \bigwedge_{j \in J_i} \beta_{(i,j)} \right)$$

  which is strongly flat (Proposition 3.18) and hence trivially in normal form. Its length is bounded by $|\psi^{nf}| \leqslant 2|\psi|$. Note that the empty conjunction is $\top$.

- If $\nabla = \sim$, then

$$\begin{aligned} \psi &= \sim \bigovee_{i \in I} \left(\alpha_i \wedge \bigwedge_{j \in J_i} \mathsf{E}\beta_{(i,j)} \right) \\ &\equiv \bigwedge_{i \in I} \left(\mathsf{E}\neg\alpha_i \ovee \bigovee_{j \in J_i} \neg\beta_{(i,j)} \right) && \text{(De Morgan's laws)} \\ &\equiv \bigovee_{f \in \prod_{i \in I} L_i} \bigwedge_{i \in I} f(i) =: \psi^{nf} && \text{(distributive law)} \end{aligned}$$

  where $L_i := \{\mathsf{E}\neg\alpha_i, \beta_{(i,j)} \mid j \in J_i\}$. The conjunction of strongly flat formulas is strongly flat, so ψ^{nf} is in normal form. Furthermore,

$$|\psi^{nf}| \leqslant \underbrace{|\psi|^{|\psi|}}_{\# \text{ of } f} \cdot \underbrace{|\psi|}_{|I|} \cdot \underbrace{(|\psi| + 3)}_{\text{size of } \mathsf{E}\neg\alpha_i / \beta_{(i,j)}} \leqslant \exp_{\mathcal{O}(1)} |\psi|.$$

- Otherwise ∇ is the lax standard transversal or lax dual standard transversal. We proceed in two steps, as in Lemmas 3.83 and 3.84. If $\psi = \nabla(\psi_1, \ldots, \psi_r)$, and $\psi_i = \bigovee_{j \in J_i}(\alpha_{(i,j)} \wedge \bigwedge_{k \in K_{(i,j)}} \mathsf{E}\beta_{(i,j,k)})$ is in normal form for all $i \in [r]$, then $\psi \equiv \psi'$, where

$$\psi' := \bigovee_{\substack{f \in \prod J_i \\ i \in [r]}} \nabla \left(\alpha_{(1,f(1))} \wedge \bigwedge_{k \in K_{(1,f(1))}} \mathsf{E}\beta_{(1,f(1),k)}, \ldots, \alpha_{(r,f(r))} \wedge \bigwedge_{k \in K_{(r,f(r))}} \mathsf{E}\beta_{(r,f(r),k)} \right).$$

This is achieved by distributing ∇ over $\ovee$ (Proposition 3.40). Next, we proceed as in Lemmas 3.83 and 3.84 and focus on the disjuncts of ψ', which are subformulas of the form

$$\theta = \nabla \left(\alpha_1 \wedge \bigwedge_{j \in J_1} \mathsf{E}\beta_{(1,j)}, \ldots, \alpha_r \wedge \bigwedge_{j \in J_r} \mathsf{E}\beta_{(r,j)} \right).$$

We expand θ depending on whether ∇ is a lax standard transversal or dual standard transversal.

– If ∇ is a lax standard transversal, as in Lemma 3.83,

$$\theta \equiv \underbrace{\nabla(\alpha_1, \ldots, \alpha_r)}_{\text{size} \leqslant |\psi|} \quad \wedge \bigwedge_{i \in [r], j \in J_i} \underbrace{\mathsf{E}\nabla(\alpha_1, \ldots, \alpha_{i-1}, \alpha_i \wedge \beta_{(i,j)}, \alpha_{i+1}, \ldots, \alpha_r)}_{\text{size} \leqslant |\psi|}.$$

– If ∇ is a lax dual standard transversal, as in Lemma 3.84,

$$\theta \equiv \underbrace{\nabla(\alpha_1, \ldots, \alpha_r)}_{\text{size} \leqslant |\psi|} \quad \wedge \bigwedge_{i \in [r], j \in J_i} \underbrace{\mathsf{E}\neg\nabla(\bot^{i-1}, \neg(\alpha_i \wedge \beta_{(i,j)}), \bot^{r-i})}_{\text{size} \leqslant 3 + |\psi|}.$$

In both cases, the resulting formula is in normal form and of size $\leqslant |\theta| + |\theta|^3$. In total, we obtain a normal form equivalent to ψ of size at most

$$\underbrace{|\psi|^{|\psi|}}_{\#\text{ disjuncts in } \psi'} \cdot \underbrace{(|\psi| + 1 + |\psi|^2(1 + 3 + |\psi|))}_{\text{size of each disjunct } \theta}$$

which is $\leqslant \exp_2 |\psi|$ for $|\psi| \geqslant 4$, and hence in $\exp_{\mathcal{O}(1)} |\psi|$. □

By the lemma of the change of normal form, the same works for $(\ovee\vee)$-normal form:

Corollary 3.90. *Let* $\{\wedge, \ovee, \sim, \neg, \top, \bot\} \subseteq \tau$. *There is a function* $f\colon \mathcal{F}_\tau \to \mathcal{F}_\tau$ *computable in time* $\exp_{\mathcal{O}(n)}(1)$ *such that for every* τ*-formula* φ*,*

- $f(\varphi)$ *is in* $(\ovee\vee)$*-normal form,*
- $[\![\varphi]\!]^A = [\![f(\varphi)]\!]^A$ *in every nice* τ*-team algebra* A*.*

A formula in $(\ovee\wedge)$-normal form contains only finitely many E-subformulas. This means that every team satisfying it already contains a finite team doing so.

Corollary 3.91. *If* A *is a nice* τ*-team algebra,* φ *is a* τ*-formula and* $T \in [\![\varphi]\!]^A$*, then there is a finite subteam* $S \subseteq T$ *with* $|S| \leqslant |\varphi|$ *such that* $S \in [\![\varphi]\!]^A$*.*

Since the connectives $\vee, \Diamond, \Box, \exists, \forall$ from propositional, modal and first-order team logic and their atoms are lax (dual) standard transversals (Corollary 3.72 and Proposition 3.76), the results in this section apply to them:

Corollary 3.92 (Collapse theorem)**.** *For* $\mathcal{L} \in \{\mathsf{PL}, \mathsf{QPL}, \mathsf{ML}, \mathsf{FO}\}$*, and for every* $\mathcal{L}(\sim)$*-formula* φ*, there is an equivalent* $\mathcal{L}(\sim)$*-formula* ψ *in* $(\owedge\wedge)$*-normal form* (*resp.* $(\owedge\vee)$*-normal form*) *that is computable in time* $\exp_{\mathcal{O}(|\varphi|)}(1)$.

3.8 Outlook: Linear Temporal Logic

As an example of a team logic that is not based on standard transversals, we turn to linear temporal logic (LTL).

A *trace* π in a Kripke structure $\mathcal{K} = (W, R, V)$ is an infinite sequence $\pi = (s_i)_{i \in \mathbb{N}} \in W^\omega$ such that $(s_i, s_{i+1}) \in R$ for all $i \geqslant 0$. Let $\pi[j]$ denote the trace π advanced by j places, i.e., $(s_{j+i})_{i \in \mathbb{N}}$. The logic LTL consists of atomic propositions and the temporal connectives $\mathsf{X}, \mathsf{F}, \mathsf{G}, \mathsf{U}$, which are defined as follows:

$$\begin{aligned}
(\mathcal{K}, \pi) &\vDash p && \Leftrightarrow\ s_0 \in V(p), \text{ where } \pi = (s_0, s_1, \dots) \\
(\mathcal{K}, \pi) &\vDash \neg\alpha && \Leftrightarrow\ (\mathcal{K}, \pi) \nvDash \alpha \\
(\mathcal{K}, \pi) &\vDash \alpha \wedge \beta && \Leftrightarrow\ (\mathcal{K}, \pi) \vDash \alpha \text{ and } (\mathcal{K}, \pi) \vDash \beta \\
(\mathcal{K}, \pi) &\vDash \alpha \vee \beta && \Leftrightarrow\ (\mathcal{K}, \pi) \vDash \alpha \text{ or } (\mathcal{K}, \pi) \vDash \beta \\
(\mathcal{K}, \pi) &\vDash \mathsf{X}\alpha && \Leftrightarrow\ (\mathcal{K}, \pi[1]) \vDash \alpha \\
(\mathcal{K}, \pi) &\vDash \mathsf{F}\alpha && \Leftrightarrow\ \exists j \geqslant 0 : (\mathcal{K}, \pi[j]) \vDash \alpha \\
(\mathcal{K}, \pi) &\vDash \mathsf{G}\alpha && \Leftrightarrow\ \forall j \geqslant 0 : (\mathcal{K}, \pi[j]) \vDash \alpha \\
(\mathcal{K}, \pi) &\vDash \alpha \mathsf{U} \beta && \Leftrightarrow\ \exists j \geqslant 0 : (\mathcal{K}, \pi[j]) \vDash \beta \text{ and } \forall k < j : (\mathcal{K}, \pi[k]) \vDash \alpha.
\end{aligned}$$

First of all, we observe that G is the strong dual of F, which is an operator, whereas neither U nor its dual is an operator. Furthermore, X is a self-dual operator.

The team semantics of LTL works with sets of traces as teams. For a team T of traces, let $T[j] := \{\, \pi[j] \mid \pi \in T \,\}$, and for $f \colon T \to \omega$, let $T[f] := \{\, \pi[f(\pi)] \mid \pi \in T \,\}$. If $f, f' \colon T \to \omega$, then $f < f'$ means $f(\pi) < f'(\pi)$ for all $\pi \in T$.

The connectives $\wedge$, $\neg$, $\sim$ and $\vee$ are completely analogous to modal team logic. Otherwise the semantics is as follows, with all temporal connectives except for X being defined in two variants [91]. (We omit the structure $\mathcal{K}$ in the definition.)

$$T \vDash \mathsf{X}\varphi \ \Leftrightarrow\ T[1] \vDash \varphi$$

F, G and U exist as *synchronous* variants,

$$\begin{aligned} T \vDash F^s\varphi &\Leftrightarrow \exists j \geqslant 0 : T[j] \vDash \varphi \\ T \vDash G^s\varphi &\Leftrightarrow \forall j \geqslant 0 : T[j] \vDash \varphi \\ T \vDash \varphi U^s \psi &\Leftrightarrow \exists j \geqslant 0 : T[j] \vDash \psi \text{ and } \forall k < j : T[k] \vDash \varphi, \end{aligned}$$

and *asynchronous* variants,

$$\begin{aligned} T \vDash F^a\varphi &\Leftrightarrow \exists f \colon T \to \omega \ : T[f] \vDash \varphi \\ T \vDash G^a\varphi &\Leftrightarrow \forall f \colon T \to \omega \ : T[f] \vDash \varphi \\ T \vDash \varphi U^a \psi &\Leftrightarrow \exists f \colon T \to \omega \ : T[f] \vDash \psi \text{ and } \forall f' \colon T \to \omega : \text{ if } f' < f, \text{ then } T[f'] \vDash \varphi. \end{aligned}$$

Here, a curious asymmetry between synchronous and asynchronous definitions emerges: All asynchronous connectives, X and G^s are flatness preserving, whereas F^s and U^s are not [91]. In fact, X, F^a and G^a are transversals. Also, G^a (G^s) is the strong dual of F^a (F^s).

On teams, X, F^a, and F^s are still operators: The generating relations are $\mathcal{R}_X T = \{ T[1] \}$, $\mathcal{R}_{F^a} T = \{ T[f] \mid f \colon T \to \omega \}$, and $\mathcal{R}_{F^s} T = \{ T[j] \mid j \geqslant 0 \}$, while G^a and G^s are the associated strong duals. U is again neither an operator nor the dual of one. It is straightforward to see that X is a transversal, and for F^a, the underlying choice functions correspond precisely to the functions $T \to \omega$, as on singletons, $\mathcal{R}_{F^a}\{\pi\} = \big\{ \{\pi[j]\} \mid j \geqslant 0 \big\}$. In fact, we observe that F^a is the *strict standard transversal* of the classical F. The connective F^s is an operator, but no transversal, as it is not flatness preserving.

The relaxation $F^{a\cup}$ of F^a would additionally allow every trace $\pi \in T$, loosely speaking, to advance to *multiple* positions at once, that is,

$$\mathcal{R}_{F^{a\cup}}\{\pi\} = \wp^+ \{ \pi[j] \mid j \geqslant 0 \}.$$

It seems that no such operator is defined in the literature yet. Neither is there an operator G^d defined that—in the spirit of $\Box$ and $\forall$—is a teamification of G and a weak dual of F. It is the transversal

$$\mathcal{R}_{G^d} T = \big\{ \{\pi[j] \mid \pi \in T, j \geqslant 0\} \big\}.$$

In other words, with G^d, we associate with a trace π the team of all possible suffixes. Finally, we show that G^d, G^a and G^s are all not only flatness preserving, but teamifications of G:

$$\begin{aligned} T' \in \wp G(T) &\Leftrightarrow T' \subseteq G(T) \\ &\Leftrightarrow T' \subseteq \{\pi \mid \forall j \geqslant 0 : \pi[j] \in T\} \\ &\Leftrightarrow \forall \pi \in T' : \forall j \geqslant 0 : \pi[j] \in T \\ &\Leftrightarrow \big\{ \pi[j] \mid \pi \in T', j \geqslant 0 \big\} \in \wp T \\ &\Leftrightarrow \mathcal{R}_{G^d} T' \cap \wp T \neq \emptyset \end{aligned}$$

$$\Leftrightarrow T' \in G^d(\wp T).$$

For G^a, additionally

$$\begin{aligned} \ldots &\Leftrightarrow \forall \pi \in T' : \forall j \geqslant 0 : \pi[j] \in T \\ &\Leftrightarrow \forall f\colon T' \to \omega : T'[f] \in \wp T \\ &\Leftrightarrow T' \in G^a(\wp T), \end{aligned}$$

where the equivalence of the first two lines is easy to see. For G^s, the argument is identical, since G^s corresponds to G^a restricted to constant functions $f\colon T' \to \omega$.

A peculiar consequence is that, by Theorem 3.37, F^a and G^a are simultaneously strong and weak duals, and likewise for F^s and G^s. On singletons, F^s and F^a as well as G^s and G^a are equivalent, so these are flatness equivalent pairs (Definition 3.53). This is not a contradiction, but stems from the fact that, loosely speaking, starting on singleton teams, F and G advance only to other singleton teams, and on those there is no difference between $\neg$ and $\sim$. This shows that operators may have non-operators as weak duals. From the above, the only weakly dual operator of F^a is G^d; see also Figure 3.6.

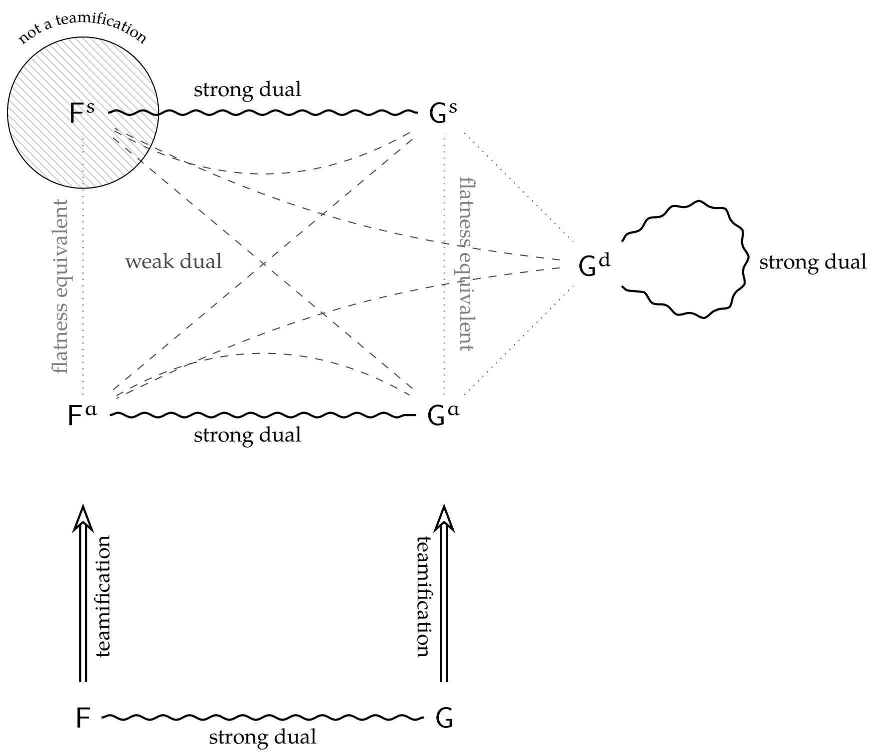


Figure 3.6: Teamification of temporal operators F and G.

3.9 Summary and outlook

3.9.1 Summary

In this chapter, we presented an abstract framework to systematically classify existing team-semantical logics. As a first step, we categorized connectives as *teamifications* of their classical counterparts in Theorem 3.33, and saw in Theorem 3.34 that teamifications are precisely the connectives that are flatness preserving (meaning that $\triangle(\mathcal{P}_1, \ldots, \mathcal{P}_r)$ is flat whenever $\mathcal{P}_1, \ldots, \mathcal{P}_r$ are flat team properties). Recall that a connective ∇ is a teamification of $\triangle$ if $\nabla(\wp T_1, \ldots, \wp T_r) = \wp\triangle(T_1, \ldots, T_r)$, and so the power set operation $\wp$ is a homomorphism between the corresponding algebras of classical and team semantics.

Next, in Theorem 3.43 and Proposition 3.44 we demonstrated that virtually all team-logical connectives, including atoms, are *operators*, that is, generated by a "successor relation" similar to classical modal logic. Furthermore, we showed in Corollary 3.51 that also the duals of operators can become operators upon teamification. For example, $\Diamond$ and $\Box$ are both operators on teams, and likewise $\wedge$, $\vee$, $\exists$ and $\forall$, while classically, of these only $\Diamond$, $\wedge$ and $\exists$ are operators. We studied the connection between such pairs, called *weak duality*, which is weak in the sense that dual connectives are in general not mutually interdefinable by means of $\neg$ in team logic. In Theorem 3.50, we gave a characterization of weak duality in terms of so-called hitting vectors.

An important subclass of operators were *transversals*. An operator is a transversal whose semantics on a team are determined by the semantics of the singletons in the team. This is a natural restriction satisfied by common team-logical connectives (Theorem 3.60). Since a classical connective can correspond to more than one transversal, we defined *standard transversals* and their *relaxations* in order to propose a canonical teamification resembling strict and lax semantics. These definitions also capture existing connectives of propositional, modal and first-order team logic (Corollary 3.72).

Finally, we proved the main result of this chapter: Given a team logic where all connectives but $\sim$ and $\neg$ are lax standard transversals, each of its formulas "collapses down" and can be written as a Boolean combination of flat formulas (Theorem 3.89). We called such formulas resp. logics *quasi-flat*. Although this result was indirectly known for, e.g., modal team logic [82, 100], we generalized the theorem to logics with arbitrarily many operators of arbitrary arity.

3.9.2 Open problems and further research directions

Atoms. One weakness of our approach is that it treats atoms as "black boxes" in order to uniformly capture the different logics. For instance, a first-order literal α simply is the unary relation of all assignments that satisfy it, oblivious of terms and variables. As a consequence, concepts such as locality cannot be studied in this framework, despite being a distinguishing feature between strict and lax semantics [37]. Moreover, our classification mostly excluded non-classical atoms, as we focused on flatness preserving connectives. In future research, the current model could be expanded to further analyze and classify those atoms.

Algebras for team logic. Boolean algebras with operators are extremely well-studied structures [138] and could yield interesting results for team logics, such as the famous Jónsson-Tarski Representation Theorem [77]. One obstacle towards a purely algebraic definition of team semantics is that we require carriers of the form $\wp\wp X$ in order to even formulate basic properties like downward closure or flatness. But even assuming this, given a Boolean algebra $\mathbb{B} = (\wp\wp X, \cup, \cdot^{\complement})$ and a pair $\{T\}, \{T'\} \in \wp\wp X$, $\mathbb{B}$ does not "know" whether $T \subseteq T'$ or not, so Boolean algebras do not suffice to talk about these closure properties. With an operation $\otimes$, where $x \otimes y$ is the set of unions of teams in x and y, $T \subseteq T'$ could be expressed as $\{T\} \otimes \{T'\} = \{T'\}$. Alternatively, we add a *flattening* operation $\downarrow x$ in analogy to Hodges [71] with the semantics that $T \vDash \downarrow\varphi$ iff $\{s\} \vDash \varphi$ for all $s \in T$. Algebraically, $\downarrow$ maps x to $\wp \bigcup x$. We can then define $T \subseteq T'$ as $(\downarrow\{T\}) \cup (\downarrow\{T'\}) = (\downarrow\{T'\})$. Then the fixpoints of $\downarrow$ form a subalgebra that is isomorphic to the underlying classical Boolean algebra on $\wp X$.

Furthermore, as Hintikka [67] noted, the dual of $\downarrow$, which we called E, is indeed an operator. ($T \vDash \mathsf{E}\varphi$ iff $\exists s \in T : \{s\} \vDash \varphi$, so the generating relation is $\{(T, \{s\}) \mid s \in T\}$.) Since $\neg\varphi \equiv {\sim}\mathsf{E}\varphi$, this allows a description of team logic in pure terms of BAOs, at least if all connectives besides $\bigwedge$, $\circledvee$, $\neg$ and $\sim$ are operators, such as for $\mathsf{PL}(\sim)$, $\mathsf{ML}(\sim)$ or $\mathsf{FO}(\sim)$.

Exotic connectives. In this chapter, we focused on teamifications as connection between classical and team-logical connectives, and special cases such as transversals. This unfortunately excludes not only a plethora of non-classical atoms, but also two other important classes of connectives: First, those not corresponding to any classical symbol, such as relevant disjunction $\overset{\bowtie}{\vee}$ [123] or forking atoms $\lessdot$ [46]. Secondly, natural operators that have classical counterparts, but are not flatness preserving, or no operators. Examples for the latter are the temporal connectives F^s and U. In further research, our framework could be extended to account for such connectives.

Finally, we could turn our attention to "intermediate" operators between the classical strict and lax ones. Take for instance the transversal $\Diamond^\star$ where $\mathcal{R}_{\Diamond^\star}\{w\}$ is any proper ideal on $\wp^+(Rw)$ (i.e., closed under subsets and finite unions, for example, $\wp^{<\omega}(Rw)$). Then $\Diamond^\star \neq \Diamond$, but one can still show that $\Diamond^\star$ preserves (quasi-)flatness. In the current form, our results do not cover such operators.

4 The complexity of modal team logic

In this chapter, we study modal team logic $\mathsf{ML}(\sim)$ and the complexity of its satisfiability and validity problem. We show that they are complete for the non-elementary class $\textsc{Tower}(\mathrm{poly})$ (Definition 2.5), and that for the fragments $\mathsf{ML}_k(\sim)$ of bounded modal depth k they are complete for $\mathrm{ATIME\text{-}ALT}(\exp_{k+1}, \mathrm{poly})$ (Definition 2.2). These results fill a long-standing gap in the study of propositional and modal team logics (see Chapter 2).

In our approach, we consider so-called *canonical models*, which are a standard tool in modal logics. In Section 4.1, we adapt this notion to modal logics with team semantics, and prove that such models exist for $\mathsf{ML}(\sim)$. For the hardness results with respect to the above complexity classes, in Sections 4.2 and 4.3 we first show that $\mathsf{ML}(\sim)$ can, in a certain sense, efficiently define bisimilarity. In the same spirit, in Section 4.4 we show that it can define the canonicity of models. Then we show in Section 4.5 that the ordering of a sufficiently large initial segment of the natural numbers can be simulated, which we then use to encode computations of non-elementary length in such large models in Section 4.6. See Figure 4.1 for an illustration of the whole reduction.

Afterwards, we adapt the lower bound to hold also in strict semantics (Section 4.7) and in restricted frame classes (Section 4.8). Finally, in Section 4.9, we identify a non-trivial fragment of $\mathsf{ML}(\sim)$ with only elementary complexity, which we prove by adapting the well-known filtration method to team semantics.

4.1 Types and canonical models

Many modal logics admit a so-called *canonical model*, which witnesses all satisfiable (sets of) formulas in some of its points. They are a standard tool for proving the completeness of modal systems [33]. Unfortunately, a canonical model for ML is necessarily infinite, and consequently impractical for complexity theoretic considerations. Instead, we use so-called (Φ, k)-*canonical models* $\mathfrak{C}_k^\Phi$ for finite $\Phi \subseteq \mathsf{Prop}$ and $k \in \mathbb{N}$. These are canonical in the above sense for the fragment ML_k^Φ, and more importantly are finite, although their size is at least the number of equivalence classes of $\rightleftharpoons_k^\Phi$ (Definition 2.24).

We will refer to these classes as (Φ, k)-*types* or just *types* and usually write them as the letter τ. A first issue arises because types are proper classes. In team semantics we have teams, so we need to speak about *sets* of types. For this reason, we begin this section by defining types on proper set-theoretic grounds, by identifying the type of a point with the set of formulas that are true in it.

Definition 4.1 (Type)**.** A set $\tau \subseteq \mathsf{ML}_k^\Phi$ of formulas is a (Φ, k)-*type* if it is (classically) satisfiable and for all $\alpha \in \mathsf{ML}_k^\Phi$ contains either α or $\neg\alpha$.

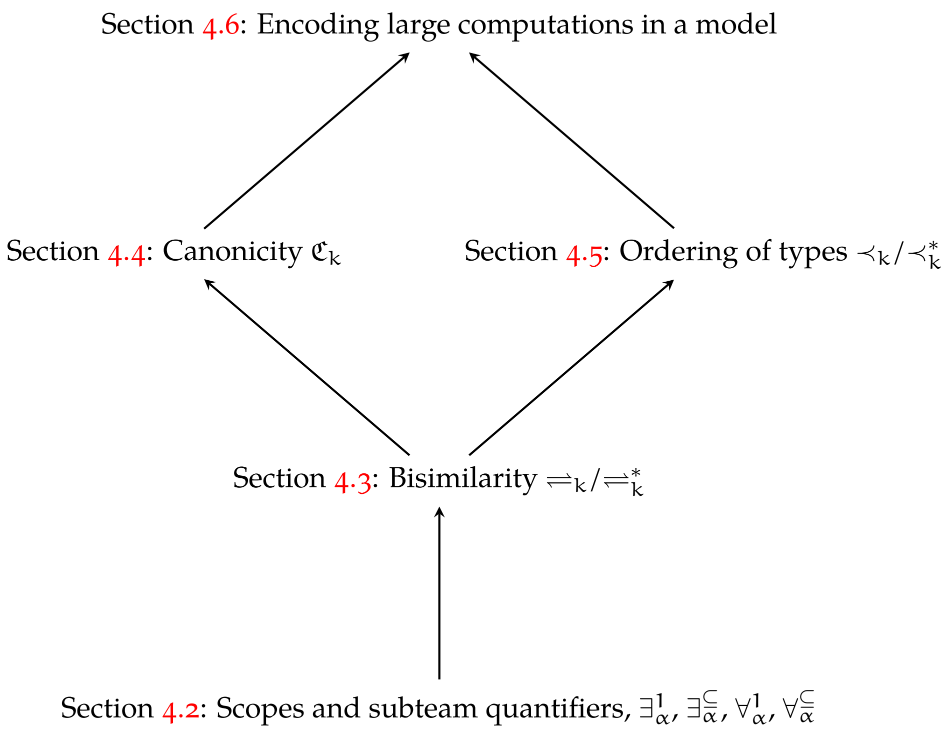


Figure 4.1: "Roadmap" for the reduction from Tower(poly) to satisfiability of $\mathsf{ML}(\sim)$.

Definition 4.2. The (Φ, k)-type of a pointed structure $(\mathcal{K}, w)$ is

$$[\![\mathcal{K}, w]\!]_k^\Phi := \left\{\alpha \in \mathsf{ML}_k^\Phi \mid (\mathcal{K}, w) \vDash \alpha\right\}.$$

Definition 4.3. The set of (Φ, k)-types of a structure with team $(\mathcal{K}, T)$ is

$$[\![\mathcal{K}, T]\!]_k^\Phi := \left\{[\![\mathcal{K}, w]\!]_k^\Phi \mid w \in T\right\}.$$

Definition 4.4. Let $\Phi \subseteq \mathsf{Prop}$ and $k \geqslant 0$. The set of all (Φ, k)-types is Δ_k^Φ.

The following assertions ensure that the above definition of types properly reflects the bisimulation relation.

Proposition 4.5. *Let $\Phi \subseteq \mathsf{Prop}$ and $k \geqslant 0$. Then*

(1) *The unique (Φ, k)-type satisfied by $(\mathcal{K}, w)$ is $[\![\mathcal{K}, w]\!]_k^\Phi$.*

(2) $(\mathcal{K}, w) \rightleftharpoons_k^\Phi (\mathcal{K}', w')$ *if and only if* $[\![\mathcal{K}, w]\!]_k^\Phi = [\![\mathcal{K}', w']\!]_k^\Phi$.

(3) $(\mathcal{K}, T) \rightleftharpoons_k^\Phi (\mathcal{K}', T')$ *if and only if* $[\![\mathcal{K}, T]\!]_k^\Phi = [\![\mathcal{K}', T']\!]_k^\Phi$.

Proof. (1) is clear: Two distinct types τ, τ' satisfied by $(\mathcal{K}, w)$ must differ in some formula $\alpha \in \mathsf{ML}_k^\Phi$, but then $(\mathcal{K}, w) \vDash \alpha, \neg\alpha$, contradiction.

(2) immediately follows from Theorem 2.25.

For (3), we begin with the direction "⇒". Due to symmetry, we only show that $(\mathcal{K}, T) \rightleftharpoons_k^\Phi (\mathcal{K}', T')$ implies $[\![\mathcal{K}, T]\!]_k^\Phi \subseteq [\![\mathcal{K}', T']\!]_k^\Phi$. Hence suppose $\tau \in [\![\mathcal{K}, T]\!]_k^\Phi$. Then there exists $w \in T$ of type $[\![\mathcal{K}, w]\!]_k^\Phi = \tau$. By Definition 2.27, there is $w' \in T'$ with $(\mathcal{K}, w) \rightleftharpoons_k^\Phi (\mathcal{K}', w')$. Then $[\![\mathcal{K}', w']\!]_k^\Phi = \tau \in [\![\mathcal{K}', T']\!]_k^\Phi$ by (2). The direction "⇐" of (3) is shown analogously. □

Unsurprisingly, the type of a point w is determined solely by the propositions in w and the types in the image Rw. In other words, all pointed structures of type τ satisfy the same propositions in their roots, viz. $\tau \cap \Phi$, and have the same types contained in their image teams. The set of k-types in the image of a (world of) $(k+1)$-type τ is defined as $\mathcal{R}\tau := \{\tau' \in \Delta_k^\Phi \mid \{\alpha \mid \Box\alpha \in \tau\} \subseteq \tau'\}$.

The following propositions show that types are indeed uniquely determined by the above constituents; their proofs can be found in the appendix.

Proposition 4.6. *Let $\Phi \subseteq \mathsf{Prop}$ be finite and $k \geqslant 0$.*

(1) $[\![w]\!]_k^\Phi \cap \Phi = V^{-1}(w) \cap \Phi$ *and* $[\![Rw]\!]_k^\Phi = \mathcal{R}[\![w]\!]_{k+1}^\Phi$, *for all pointed structures* (W, R, V, w).

(2) *The map* $h\colon \tau \mapsto \tau \cap \Phi$ *is a bijection from* Δ_0^Φ *to* $\wp\Phi$.

(3) *The map* $h\colon \tau \mapsto (\tau \cap \Phi, \mathcal{R}\tau)$ *is a bijection from* Δ_{k+1}^Φ *to* $\wp\Phi \times \wp\Delta_k^\Phi$.

Proposition 4.7. *Let (W, R, V, w) be a pointed structure, $\Phi \subseteq \mathsf{Prop}$ finite and $k \geqslant 0$.*

(1) *If* $\tau \in \Delta_0^\Phi$, *then* $[\![w]\!]_0^\Phi = \tau$ *if and only if* $V^{-1}(w) = \tau \cap \Phi$.

(2) *If* $\tau \in \Delta_{k+1}^\Phi$, *then* $[\![w]\!]_{k+1}^\Phi = \tau$ *if and only if* $V^{-1}(w) = \tau \cap \Phi$ *and* $[\![Rw]\!]_k^\Phi = \mathcal{R}\tau$.

With types, we are now ready to state the formal definition of canonicity:

Definition 4.8. A structure with team $(\mathcal{K}, T)$ is (Φ, k)-*canonical* if $[\![\mathcal{K}, T]\!]_k^\Phi = \Delta_k^\Phi$.

In the following, we often omit Φ and $\mathcal{K}$ and instead write $[\![w]\!]_k$ and $[\![T]\!]_k$, respectively, and simply say that T is (Φ, k)-canonical instead of $(\mathcal{K}, T)$ if $\mathcal{K}$ is clear.

4.1.1 Canonical models in team semantics

The usual construction of a canonical model is by taking all (infinite) maximal consistent subsets of a certain class of modal formulas as worlds (see, e.g., Fitting [33]). This indeed results in a finite number of worlds in the case of, say, ML_k^Φ (cf. [19, 20]). Truly finitary constructions of canonical models can be traced back to Fine [32], whose work has been extended towards various other modal systems (e.g., by Moss [117]). Also, Cresswell and Hughes [20] coined *mini canonical models*, models that are "canonical" only with respect to all subformulas of a fixed ML-formula, which allows them to be finite models with finite sets of formulas as worlds.

We will show that, given a (Φ, k)-canonical model $\mathfrak{C}_k^\Phi$, every satisfiable $\mathsf{ML}(\sim)_k^\Phi$-formula can be satisfied in some team of $\mathfrak{C}_k^\Phi$ as well. In other words, canonical models for $\mathsf{ML}(\sim)_k^\Phi$ and ML_k^Φ coincide.

Theorem 4.9. *Let* $(\mathcal{K}, T)$ *be* (Φ, k)*-canonical and* $\varphi \in \mathsf{ML}(\sim)^{\Phi}_{k}$. *Then* φ *is satisfiable if and only if* $(\mathcal{K}, T') \vDash \varphi$ *for some* $T' \subseteq T$.

Proof. Assume $(\mathcal{K}, T)$ and φ as above. The direction from right to left is trivial. Suppose that φ is satisfiable by a model $(\hat{\mathcal{K}}, \hat{T})$. As a team in $\mathcal{K}$ that satisfies φ, we define

$$T' := \left\{ w \in T \;\middle|\; [\![\mathcal{K}, w]\!]^{\Phi}_{k} \in [\![\hat{\mathcal{K}}, \hat{T}]\!]^{\Phi}_{k} \right\}.$$

By Proposition 2.28 and Proposition 4.5, it suffices to prove $[\![\hat{\mathcal{K}}, \hat{T}]\!]^{\Phi}_{k} = [\![\mathcal{K}, T']\!]^{\Phi}_{k}$. The direction "$\supseteq$" of the proof is clear by definition. For "$\subseteq$", as T is (Φ, k)-canonical, for every $\tau \in [\![\hat{\mathcal{K}}, \hat{T}]\!]^{\Phi}_{k}$ there exists a world $w \in T$ of type τ. Consequently, $[\![\hat{\mathcal{K}}, \hat{T}]\!]^{\Phi}_{k} \subseteq [\![\mathcal{K}, T']\!]^{\Phi}_{k}$. □

How large is a (Φ, k)-canonical model at least? To capture the number of types, we define the function $\exp^*_k$:

$$\exp^*_0(n) := n \qquad\qquad \exp^*_{k+1}(n) := n \cdot 2^{\exp^*_k(n)}$$

This function resembles $\exp_k(n)$ (p. 14) except for an additional factor of n after every exponentiation.

Proposition 4.10. *For all* $k \geqslant 0$ *and finite* $\Phi \subseteq \mathsf{Prop}$ *it holds that* $|\Delta^{\Phi}_{k}| = \exp^*_k\left(2^{|\Phi|}\right)$.

Proof. By induction on k. For the base case $k = 0$, this follows from Proposition 4.6, as there is a bijection between Δ^{Φ}_{0} and $\wp(\Phi)$ and $\exp^*_0\left(2^{|\Phi|}\right) = 2^{|\Phi|} = |\Delta^{\Phi}_{0}|$. For the inductive step, note that by induction hypothesis

$$\exp^*_{k+1}\left(2^{|\Phi|}\right) = 2^{|\Phi|} \cdot 2^{\exp^*_k(2^{|\Phi|})} = 2^{|\Phi|} \cdot 2^{|\Delta^{\Phi}_{k}|} = |\wp(\Phi) \times \wp(\Delta^{\Phi}_{k})|.$$

Since there again exists a bijection from Δ^{Φ}_{k+1} to $\wp(\Phi) \times \wp(\Delta^{\Phi}_{k})$ (Proposition 4.6), this proves the proposition. □

Next, we present an algorithm that solves the satisfiability and validity problems of $\mathsf{ML}_k(\sim)$ by computing a canonical model. Let us first explicate this in a lemma.

Lemma 4.11. *There is an algorithm that, given* $\Phi \subseteq \mathsf{Prop}$ *and* $k \geqslant 0$, *runs in time polynomial in* $|\Delta^{\Phi}_{k}|$ *and computes a* (Φ, k)*-canonical model.*

Proof. The idea is to construct sets $L_0 \cup L_1 \cup \cdots \cup L_k$ of worlds in stage-wise manner such that L_i is (Φ, i)-canonical. For L_0, we simply add a world w for each $\Phi' \subseteq \Phi$ such that $V^{-1}(w) = \Phi'$. For $i > 0$, we iterate over all $L' \in \wp(L_{i-1})$ and $\Phi' \subseteq \Phi$ and insert a new world w into L_i such that L' is the image of w and such that again $V^{-1}(w) = \Phi'$. An inductive argument based on Propositions 2.28 and 4.6 shows that L_i is (Φ, i)-canonical for all $i \in \{0, \ldots, k\}$. As $k \leqslant |\Delta^{\Phi}_{k}|$, and each L_i is constructed in time polynomial in $|\Delta^{\Phi}_{i}| \leqslant |\Delta^{\Phi}_{k}|$, the overall runtime is polynomial in $|\Delta^{\Phi}_{k}|$. □

With the help of another small lemma, we conclude the upper bound for the satisfiability and validity problem of $\mathsf{ML}(\sim)$ and its fragments.

Lemma 4.12. *For every polynomial* p *there is a polynomial* q *such that*

$$p(\exp_k^*(n)) \leqslant \exp_k(q((k+1)\cdot n))$$

for all $k \geqslant 0$ *and* $n \geqslant 1$.

Proof. See the appendix. □

Theorem 4.13. $\mathsf{SAT}(\mathsf{ML}_k(\sim))$ *and* $\mathsf{VAL}(\mathsf{ML}_k(\sim))$ *are in* $\text{ATIME-ALT}(\exp_{k+1}, \text{poly})$.

Proof. The following algorithm decides the satisfiability resp. validity problem. Let $\varphi \in \mathsf{ML}_k(\sim)$ be the input, $n := |\varphi|$, and $\Phi := \mathsf{Prop}(\varphi)$. As stated in Lemma 4.11, we deterministically construct a (Φ, k)-canonical structure $\mathcal{K} = (W, R, V)$ in time $p(|\Delta_k^\Phi|)$ for a polynomial p. This part does not depend on the input formula φ, only on $\mathsf{Prop}(\varphi)$.

By Theorem 2.32, the model checking problem of $\mathsf{ML}(\sim)$ is solvable by an alternating Turing machine that has runtime polynomial in $|\varphi| + |\mathcal{K}|$ and alternations polynomial in $|\varphi|$. We call this algorithm as a subroutine: by Theorem 4.9, φ is satisfiable (resp. valid) if and only if for at least one subteam (resp. all subteams) $T \subseteq W$ it holds that $(\mathcal{K}, T) \vDash \varphi$. Equivalently, this is the case if and only if $(\mathcal{K}, W)$ satisfies $\top \vee \varphi$ (resp. $\sim(\top \vee \sim\varphi)$), so we call the model checking routine for the formula $\top \vee \varphi$ resp. $\sim(\top \vee \sim\varphi)$.

Let us turn to the overall runtime. $\mathcal{K}$ is constructed in time polynomial in $|\Delta_k^\Phi| = \exp_k^*(2^{|\Phi|}) \leqslant \exp_{k+1}^*(|\Phi|) \leqslant \exp_{k+1}^*(n)$. The subsequent model checking runs in time polynomial in $|\mathcal{K}| + n$, and hence polynomial in $\exp_{k+1}^*(n)$ as well. By Lemma 4.12, we obtain a total runtime of $\exp_{k+1}(q((k+2)\cdot n))$ for some polynomial q. □

The upper bound for $\mathsf{ML}(\sim)$ is proved identically, as $k := \mathsf{md}(\varphi)$ is polynomial in $|\varphi|$.

Corollary 4.14. $\mathsf{SAT}(\mathsf{ML}(\sim))$ *and* $\mathsf{VAL}(\mathsf{ML}(\sim))$ *are in* $\text{TOWER}(\text{poly})$.

4.2 Scopes and subteam quantifiers

Kontinen et al. [82] proved that $\mathsf{ML}(\sim)$ is expressively complete up to bisimulation: it can define every property of Kripke structures with teams that is (Φ, k)-bisimulation invariant for some finite Φ and k, i.e., closed under $\rightleftharpoons_k^\Phi$.

Two interesting such team properties are in fact (Φ, k)-bisimilarity itself—in the sense that all worlds in a team have the same (Φ, k)-type—as well as (Φ, k)-canonicity. Consequently, these properties are definable by $\mathsf{ML}(\sim)_k^\Phi$-formulas. However, the latter might be very long, the upper bound by Kontinen et al. [82] is non-elementary.

In this section, we consider a special subclass of Kripke structures and on these define k-bisimilarity by a formula χ_k of polynomial size in Φ and k. (From now on, we always assume some finite $\Phi \subseteq \mathsf{Prop}$ and omit it in the notation, i.e., we write k-*canonicity*, k-*bisimilarity*, $\rightleftharpoons_k$, and so on.) Similarly, in Section 4.4 we devise a formula canon_k of polynomial size that expresses k-canonicity.

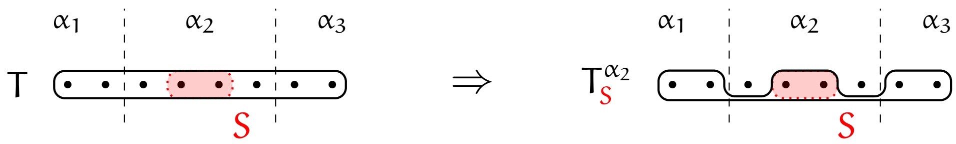


Figure 4.2: Example of subteam selection in the scope α_2

4.2.1 Scopes

It is the natural approach to implement k-bisimilarity by mutual recursion on the level of formulas. The $(k+1)$-bisimilarity of two points w, v is expressed in terms of k-team-bisimilarity of Rw and Rv, and conversely, to check whether the image teams Rw and Rv are k-team-bisimilar, we proceed analogously to the *forward* and *backward* conditions of Definition 2.24 and reduce the problem to checking k-bisimilarity of pairs of points in Rw and Rv.

There is still one issue: Formulas define team properties, which are classes of (Kripke structures with) teams, but bisimilarity is a *binary relation* between teams. For this reason, we take the "marked union" of Rw and Rv as a single team. To decompose the union again, we use the connective $\hookrightarrow$. Recall that $T \vDash \alpha \hookrightarrow \varphi$ if and only if $T_\alpha \vDash \varphi$, where $T_\alpha = \{w \in T \mid w \vDash \alpha\}$.

Now, instead of defining an n-ary relation on teams, a formula φ can define a unary relation—a team property—parameterized by formulas $\alpha_1, \ldots, \alpha_n \in \mathsf{ML}$, which function as "markers" of the respective subteams in the whole union. We emphasize this by writing $\varphi(\alpha_1, \ldots, \alpha_n)$.

It will be useful if these parameters are invariant under traversing edges in the structure. In that case, we call these formulas *scopes*:

Definition 4.15. Let $\mathcal{K} = (W, R, V)$ be a Kripke structure. A formula $\alpha \in \mathsf{ML}$ is called a *scope* (*in* $\mathcal{K}$) if $(w, v) \in R$ implies $w \vDash \alpha \Leftrightarrow v \vDash \alpha$. Two scopes α, β are called *disjoint* (*in* $\mathcal{K}$) if W_α and W_β are disjoint.

To avoid interference, we assume that scopes are formulas in $\mathsf{ML}_0^{\mathsf{Prop}\setminus\Phi}$, i.e., they are always purely propositional and do not contain propositions from Φ.

Next, we define the team that results from "cutting out" parts of a specific scope.

Definition 4.16 (Scope selection)**.** Let T be a team and $S \subseteq T$. Let $\alpha \in \mathsf{ML}$. Then $T_S^\alpha := T_{\neg\alpha} \cup (T_\alpha \cap S)$.

For singletons $\{w\}$, we simply write T_w^α instead of $T_{\{w\}}^\alpha$. Intuitively, T_S^α is obtained by "shrinking" the subteam T_α down to S while retaining $T \setminus T_\alpha$ (see Figure 4.2 for an example). The notation should remind of first-order logic: The assignment s_a^x is the same as s but maps x to a. Likewise, T_S^α is the team T but T_α is set to S.

Scopes have several desirable properties:

Proposition 4.17. *Let α, β be disjoint scopes and S, U, T teams in a Kripke structure $\mathcal{K} = (W, R, V)$. Then the following laws hold:*

(1) *Distributive laws:* $(T \cap S)_\alpha = T_\alpha \cap S = T \cap S_\alpha = T_\alpha \cap S_\alpha$ *and* $(T \cup S)_\alpha = T_\alpha \cup S_\alpha$.

(2) *Disjoint selection commutes:* $\left(T^\alpha_S\right)^\beta_U = \left(T^\beta_U\right)^\alpha_S$.

(3) *Disjoint selection is independent:* $\left((T^\alpha_S)^\beta_U\right)_\alpha = T_\alpha \cap S$.

(4) *Image and selection commute:* $(RT)_\alpha = \left(R(T_\alpha)\right)_\alpha = R(T_\alpha)$

(5) *Successor and selection commute: If* S *is a strict resp. lax successor team of* T*, then* S_α *is a strict resp. lax successor team of* T_α.

(6) *Selection propagates: If* $S \subseteq T$*, then* $R\left(T^\alpha_S\right) = (RT)^\alpha_{RS}$.

Proof. Straightforward from the definition; see the appendix. □

Accordingly, we can write $R^i T_\alpha$ instead of $(R^i T)_\alpha$ or $R^i(T_\alpha)$ and $T^{\alpha_1,\alpha_2}_{S_1,S_2}$ for $(T^{\alpha_1}_{S_1})^{\alpha_2}_{S_2}$.

4.2.2 Subteam quantifiers

In what follows, we use the material implication $\varphi \multimap \psi$ as shorthand for $\sim\varphi \varovee \psi$. Its semantics is

$$(\mathcal{K}, T) \vDash \varphi \multimap \psi \;\Leftrightarrow\; \text{if } (\mathcal{K}, T) \vDash \varphi, \text{ then } (\mathcal{K}, T) \vDash \psi.$$

We refer to the following abbreviations as *subteam quantifiers*, where $\alpha \in \mathsf{ML}$:

$$\exists^{\subseteq}_\alpha \varphi := \alpha \vee \varphi \qquad \forall^{\subseteq}_\alpha \varphi := \sim\exists^{\subseteq}_\alpha\sim\varphi$$
$$\exists^1_\alpha \varphi := \exists^{\subseteq}_\alpha \left[E\alpha \wedge \forall^{\subseteq}_\alpha(E\alpha \multimap \varphi)\right] \qquad \forall^1_\alpha \varphi := \sim\exists^1_\alpha\sim\varphi$$

Intuitively, they quantify over subteams $S \subseteq T_\alpha$ or worlds $w \in T_\alpha$ such that T^α_S resp. T^α_w satisfies φ.

Proposition 4.18. *The subteam quantifiers have the following semantics:*

$$T \vDash \exists^{\subseteq}_\alpha\varphi \Leftrightarrow \exists S \subseteq T_\alpha \colon T^\alpha_S \vDash \varphi \qquad T \vDash \exists^1_\alpha\varphi \Leftrightarrow \exists w \in T_\alpha \colon T^\alpha_w \vDash \varphi$$
$$T \vDash \forall^{\subseteq}_\alpha\varphi \Leftrightarrow \forall S \subseteq T_\alpha \colon T^\alpha_S \vDash \varphi \qquad T \vDash \forall^1_\alpha\varphi \Leftrightarrow \forall w \in T_\alpha \colon T^\alpha_w \vDash \varphi$$

Proof. We prove the existential cases, as the other ones work dually. Let us first consider the "⇒" direction for $\exists^{\subseteq}_\alpha$, so suppose $T \vDash \alpha \vee \varphi$. Then there exist $S \subseteq T$ and $U \subseteq T_\alpha$ such that $S \vDash \varphi$ and $T = S \cup U$. Since $U \cap T_{\neg\alpha} = \emptyset$, it holds $T_{\neg\alpha} \subseteq S$. For this reason, $S = (S \cap T_\alpha) \cup (S \cap T_{\neg\alpha}) = (S \cap T_\alpha) \cup T_{\neg\alpha} = T^\alpha_{S \cap T_\alpha}$. Consequently, $T^\alpha_{S\cap T_\alpha} \vDash \varphi$ for some set $S \cap T_\alpha \subseteq T_\alpha$.

For "⇐", suppose $T^\alpha_S \vDash \varphi$ for some $S \subseteq T_\alpha$. Then T^α_S and $T \setminus T^\alpha_S$ form a partition of T. Since $T \setminus T^\alpha_S = T \setminus (T_{\neg\alpha} \cup (T_\alpha \cap S)) \subseteq T \setminus T_{\neg\alpha} = T_\alpha$, it holds $T \setminus T^\alpha_S \vDash \alpha$. As a consequence, $T \vDash \alpha \vee \varphi$.

We proceed with $\exists^1_\alpha$. For "⇒", suppose that $T \vDash \exists^1_\alpha\varphi$. Then there exists $S \subseteq T_\alpha$ such that $T^\alpha_S \vDash E\alpha \wedge \forall^{\subseteq}_\alpha(E\alpha \multimap \varphi)$. Since $T^\alpha_S \vDash E\alpha$, there exists $w \in (T^\alpha_S)_\alpha$. As $\forall^{\subseteq}_\alpha$ now applies to $(T^\alpha_S)^\alpha_{\{w\}} = T^\alpha_w$ as well, it follows that $T^\alpha_w \vDash E\alpha \multimap \varphi$, and consequently $T^\alpha_w \vDash \varphi$.

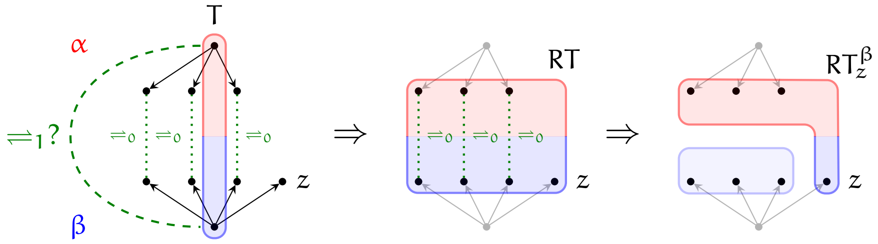


Figure 4.3: As z violates the *backward* condition, shrinking RT_β leads to a $\rightleftharpoons_0$-free subteam, falsifying $\exists^1_\alpha\exists^1_\beta\chi_0(\alpha,\beta)$.

Suppose for "⇐" that $T^\alpha_w \vDash \varphi$ for some $w \in T_\alpha$. Let $S \subseteq T_\alpha$ be arbitrary. If $w \notin S$, then $(T^\alpha_w)^\alpha_S = T^\alpha_\emptyset \nvDash E\alpha$, and if $w \in S$, then $(T^\alpha_w)^\alpha_S = T^\alpha_w \vDash \varphi$. Therefore, for any $S \subseteq T_\alpha$ it holds $(T^\alpha_w)^\alpha_S \vDash (E\alpha \multimap \varphi)$, so $T^\alpha_w \vDash \forall^{\subseteq}_\alpha(E\alpha \multimap \varphi)$. Since also $T^\alpha_w \vDash E\alpha$, it follows $T \vDash \exists^{\subseteq}_\alpha\left[E\alpha \wedge \forall^{\subseteq}_\alpha(E\alpha \multimap \varphi)\right]$. □

4.3 Implementing bisimulation

Now we have all ingredients to implement k-bisimulation as a formula. The definition is by induction on k:

$$\chi_0(\alpha,\beta) := (\alpha \vee \beta) \hookrightarrow \bigwedge_{p\in\Phi} \mathrm{dep}(p)$$

$$\chi_{k+1}(\alpha,\beta) := \chi_0(\alpha,\beta) \wedge \Box\chi^*_k(\alpha,\beta)$$

$$\chi^*_k(\alpha,\beta) := (\neg\alpha \wedge \neg\beta) \owedge \Big(E\alpha \wedge E\beta \wedge {\sim}\big[(\alpha \owedge \beta) \vee (E\alpha \wedge E\beta \wedge {\sim}\exists^1_\alpha\exists^1_\beta\chi_k(\alpha,\beta))\big]\Big)$$

The reader may wonder why this translation does not follow the forward and backward condition, which rather corresponds to $\chi^*_k(\alpha,\beta) := \forall^1_\alpha\exists^1_\beta\chi_k(\alpha,\beta) \wedge \forall^1_\beta\exists^1_\alpha\chi_k(\alpha,\beta)$. The reason is that the more complicated formula shown above avoids the exponential blowup that would come with two recursive calls.

Theorem 4.19. *Let $k \geqslant 0$. For all teams T, disjoint scopes α,β, and points $w \in T_\alpha$ and $v \in T_\beta$, the following holds:*

$$\begin{aligned} T^{\alpha,\beta}_{w,v} &\vDash \chi_k(\alpha,\beta) &&\Leftrightarrow\; w \rightleftharpoons_k v,\\ T &\vDash \chi^*_k(\alpha,\beta) &&\Leftrightarrow\; T_\alpha \rightleftharpoons_k T_\beta. \end{aligned}$$

Moreover, both $\chi_k(\alpha,\beta)$ and $\chi^*_k(\alpha,\beta)$ are $\mathrm{ML}_k({\sim})$-formulas that are constructible in space $\mathcal{O}(\log(k + |\Phi| + |\alpha| + |\beta|))$.

Proof. The formulas χ_0 and χ_{k+1} are straightforward. For $\chi^*_k(\alpha,\beta)$, let us start by first providing some intuition on how this formula expresses team bisimulation. We focus on the case where $\neg\alpha \wedge \neg\beta$ is false and $E\alpha \wedge E\beta$ is true (otherwise there is nothing to prove). The idea is to isolate a *witness point* in $z \in T_\alpha \cup T_\beta$ for $[\![T_\alpha]\!]_k \neq [\![T_\beta]\!]_k$, say, $[\![z]\!]_k \in [\![T_\beta]\!]_k \setminus [\![T_\alpha]\!]_k$. We erase $T_\beta \setminus \{z\}$ from T using the disjunction $\vee$ in χ^*_k, as $T_\beta \setminus \{z\} \vDash \alpha \oslash \beta$. The remaining team is exactly T^β_z, in which $\exists^1_\alpha \exists^1_\beta \chi_k(\alpha, \beta)$ fails (see Figure 4.3). The case $[\![z]\!]_k \in [\![T_\alpha]\!]_k \setminus [\![T_\beta]\!]_k$ is detected analogously.

We proceed with a formal correctness proof by induction on k. Let $\mathcal{K} = (W, R, V)$ be a Kripke structure. The base case $k = 0$ is straightforward, as no proposition $p \in \Phi$ occurs in α or β. The induction step is by mutual recursion: First we assume that the theorem holds for χ_k and prove it for χ^*_k, and then we show it for χ_{k+1}.

"$\chi_k \Rightarrow \chi^*_k$": Let T be a team and α, β disjoint scopes in $\mathcal{K}$. Observe that χ^*_k is always true if T_α and T_β are both empty (then $[\![T_\alpha]\!]_k = [\![T_\beta]\!]_k$), and that it is always false if exactly one of them is empty (then $[\![T_\alpha]\!]_k \neq [\![T_\beta]\!]_k$). So w.l.o.g. $T_\alpha, T_\beta \neq \emptyset$. Then $\chi^*_k(\alpha, \beta)$ boils down to

$$\psi := \sim((\alpha \oslash \beta) \vee E\alpha \wedge E\beta \wedge \sim\exists^1_\alpha \exists^1_\beta \chi_k(\alpha, \beta)),$$

for which we prove that it is equivalent to $[\![T_\alpha]\!]_k = [\![T_\beta]\!]_k$.

The first direction is proved by contradiction. Suppose $[\![T_\alpha]\!]_k = [\![T_\beta]\!]_k$ but $T \nvDash \psi$. The disjunction is then witnessed by some division $T = S \cup U$, where w.l.o.g. $S \subseteq T_\alpha$ satisfies $\alpha \oslash \beta$, (if $S \subseteq T_\beta$, the proof is symmetric), and $U \vDash E\alpha \wedge E\beta \wedge \sim\exists^1_\alpha \exists^1_\beta \chi_k(\alpha, \beta)$. Since $T_\alpha \cap T_\beta = \emptyset$, then $T_\beta \subseteq U$, and clearly $T_\beta \subseteq U_\beta$. By the formula $E\alpha$, there exists $w \in U_\alpha$. By assumption that $[\![T_\alpha]\!]_k = [\![T_\beta]\!]_k$, U_β must contain a world v of type $[\![w]\!]_k$ as well. But then $U^{\alpha,\beta}_{w,v} \vDash \chi_k(\alpha, \beta)$ by induction hypothesis, contradiction to $U \vDash \sim\exists^1_\alpha \exists^1_\beta \chi_k(\alpha, \beta)$.

For the other direction, suppose $[\![T_\alpha]\!]_k \neq [\![T_\beta]\!]_k$. W.l.o.g. there exists $w \in T_\alpha$ such that $[\![w]\!]_k \notin [\![T_\beta]\!]_k$. (For $w \in T_\beta$, the proof is again symmetric.) Consider $S := T_\alpha \setminus \{w\}$ and $U := T^\alpha_w$ as a division of T. Then $S \vDash \alpha \oslash \beta$ and $U \vDash E\alpha \wedge E\beta$. It remains to show $U \vDash \sim\exists^1_\alpha \exists^1_\beta \chi_k(\alpha, \beta)$. However, this is easy to see: $U \vDash \exists^1_\alpha \exists^1_\beta \chi_k(\alpha, \beta)$ if and only if $U \vDash \exists^1_\beta \chi_k(\alpha, \beta)$, but T_β and hence U_β contains no world of type $[\![w]\!]_k$, so by induction hypothesis U cannot satisfy $\exists^1_\beta \chi_k(\alpha, \beta)$.

"$\chi^*_k \Rightarrow \chi_{k+1}$": We follow Definition 2.24 and Proposition 2.28.

$$\begin{aligned}
& T^{\alpha,\beta}_{w,v} \vDash \chi_{k+1}(\alpha, \beta) & \\
\Leftrightarrow\ & T^{\alpha,\beta}_{w,v} \vDash \chi_0(\alpha, \beta) \wedge \Box\chi^*_k(\alpha, \beta) & \text{(def. } \chi_{k+1}) \\
\Leftrightarrow\ & w \rightleftharpoons_0 v \text{ and } T^{\alpha,\beta}_{w,v} \vDash \Box\chi^*_k(\alpha, \beta) & \text{(induction hypothesis)} \\
\Leftrightarrow\ & w \rightleftharpoons_0 v \text{ and } RT^{\alpha,\beta}_{Rw,Rv} \vDash \chi^*_k(\alpha, \beta) & \text{(Proposition 4.17)} \\
\Leftrightarrow\ & w \rightleftharpoons_0 v \text{ and } Rw \rightleftharpoons_k Rv & \text{(induction hypothesis)} \\
\Leftrightarrow\ & w \rightleftharpoons_{k+1} v. & \text{(Proposition 2.28)}
\end{aligned}$$

It is routine to check that the formulas are constructible in logarithmic space from α, β, Φ and k, and that $\mathrm{md}(\chi_k) = \mathrm{md}(\chi^*_k) = k$. ☐

Let us stress that χ_k relies on disjoint scopes to be present in the structure, and it is open whether bisimilarity is polynomially definable otherwise. A related property, namely that a point w contains exactly one type in its image (i.e., $|[\![Rw]\!]_k| \leqslant 1$), was recently studied by Hella and Vilander [62]. They proved it expressible in ML, but only by formulas of non-elementary size. By this, it seems unlikely that even in ML($\sim$) a polynomial formula is achievable without scopes.

However, Hella and Vilander [62] proved that their property is definable in exponential size in *2-dimensional modal logic* ML^2. Roughly speaking, ML^2 is evaluated by traversing over *pairs* of points independently (for a formal introduction, see Marx and Venema [112]). Pairs of points seem as a special case of teams, so it is plausible that ML($\sim$) is stronger than ML^2. But on the other hand, the modalities in ML($\sim$) do not act on the points in a team independently, as they do in ML^2, but instead always proceed to a successor team "synchronously". As a consequence, it is open whether ML($\sim$) can define any of the above properties in an elementary sized formula.

4.4 Enforcing a canonical model

In this section, we return to canonical models of ML($\sim$), but prove a lower bound, in a sense. We devise an $\mathsf{ML}_k(\sim)$-formula that is satisfiable but permits *only* k-canonical models. For $k = 0$, that is, in PL($\sim$), Hannula et al. [56] defined the formula

$$\mathsf{max}(\Phi) := \sim \bigvee_{p \in \Phi} \mathsf{dep}(p)$$

and proved that $T \vDash \mathsf{max}(\Phi)$ if and only if T is 0-canonical, i.e., contains all Boolean assignment over Φ. We generalize this for all k, i.e., construct a satisfiable formula canon_k that has only k-canonical models.

4.4.1 Staircase models

In our approach, we express k-canonicity by inductively enforcing i-canonical sets of worlds for $i = 0, \ldots, k$ located in different "height" inside the model. For this purpose, we use specific scopes $\mathfrak{s}_0, \ldots, \mathfrak{s}_k$ ("*stairs*"), and introduce a certain class of models:

Definition 4.20. Let $k, i \geqslant 0$ and let $(\mathcal{K}, T)$ be a structure with team, $\mathcal{K} = (W, R, V)$. Then T is k-*canonical with offset* i if for every $\tau \in \Delta_k$ there exists $w \in T$ with $[\![R^i w]\!]_k = \{\tau\}$. $(\mathcal{K}, T)$ is called k-*staircase* if for all $i \in \{0, \ldots, k\}$ we have that $T_{\mathfrak{s}_i}$ is i-canonical with offset $k - i$.

As an example, a 3-staircase for $\Phi = \emptyset$ is depicted in Figure 4.4. Observe that it is a *directed forest*, i.e., its induced undirected graph is acyclic and all worlds are either *roots* (i.e., without predecessor) or have exactly one predecessor. Moreover, it has bounded *height*, where the height of a directed forest is the greatest number h such that every path traverses at most h edges. It is straightforward to construct k-staircase models for arbitrary k in a way similar to Figure 4.4.

Proposition 4.21. *For each $k \geqslant 0$, there is a finite k-staircase $(\mathcal{K}, T)$ such that $\mathfrak{s}_0, \ldots, \mathfrak{s}_k$ are disjoint scopes in $\mathcal{K}$, and $\mathcal{K}$ is a directed forest of height at most k and T its set of roots.*

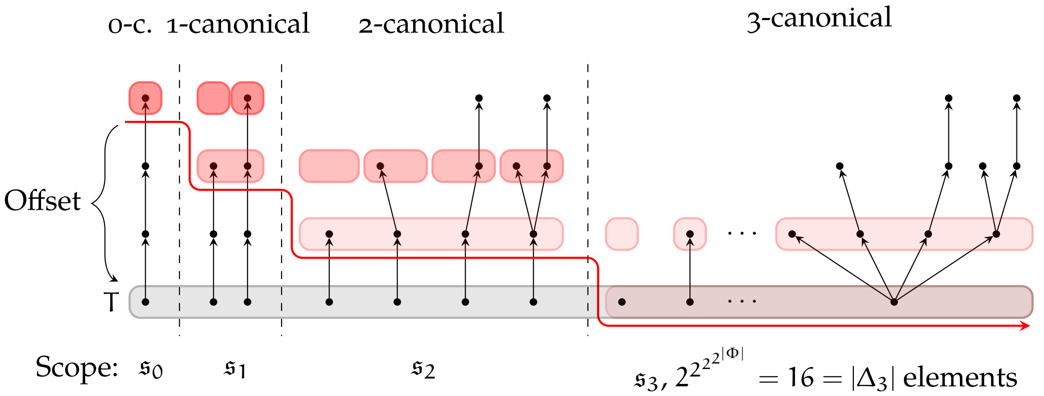


Figure 4.4: Visualization of the 3-staircase for $\Phi = \emptyset$, where the team $T_{\mathfrak{s}_i}$ is i-canonical with offset $3 - i$.

Observe that in such a model, $T_{\mathfrak{s}_k}$ is k-canonical (that is, with offset 0).

Corollary 4.22 (Finite forest model property of $\mathsf{ML}(\sim)$). *Every satisfiable* $\mathsf{ML}(\sim)$*-formula has a finite model* $(\mathcal{K}, T)$ *such that* $\mathcal{K}$ *is a directed forest of height at most* $\mathsf{md}(\varphi)$ *and its set of roots being exactly* T.

4.4.2 Enforcing canonicity

In the rest of the section, we prove that the existence of a k-staircase can be enforced in $\mathsf{ML}(\sim)$. Let us start with $k = 0$. The rather simple formula $\Box^i \mathsf{max}(\Phi)$ might come to mind, which expresses 0-canonicity of $R^i T$. But this is not the same as 0-canonicity of T with offset i, an easy counter-example is a singleton T with multiple branches leading away from it. Instead, we use the formula

$$\mathsf{max}_i := \top \lor (\Diamond^i \top \land {\sim} \bigvee_{p \in \Phi} (\Diamond^i p \otimes \Diamond^i \neg p)).$$

Lemma 4.23. $T \vDash \mathsf{max}_i$ *iff* T *is* 0*-canonical with offset* i.

Proof. By definition, a team T is 0-canonical with offset i if it has a subteam T' such that for every $w \in T'$ it holds that $[\![R^i w]\!]_0 = \{\tau\}$ for some type τ, and such that every $\tau \in \Delta_0$ occurs in this way.

First, the quantification of the subteam T' is done via $\top \lor \cdots$. Then $\Diamond^i \top$ ensures that $R^i w \neq \emptyset$ for every $w \in T$. For the remaining proof, it holds that

$$\sim \bigvee_{p \in \Phi} (\Diamond^i p \otimes \Diamond^i \neg p) \equiv \sim \bigotimes_{\Phi' \subseteq \Phi} \Big(\bigvee_{p \in \Phi'} \Diamond^i p \lor \bigvee_{p \in \Phi \setminus \Phi'} \Diamond^i \neg p \Big) \equiv \bigwedge_{\Phi' \subseteq \Phi} \mathsf{E}\Big(\bigwedge_{p \in \Phi'} \Box^i \neg p \land \bigwedge_{p \in \Phi \setminus \Phi'} \Box^i p \Big).$$

This follows by the distributive law $\varphi \lor (\psi_1 \otimes \psi_2) \equiv (\varphi \lor \psi_1) \otimes (\varphi \lor \psi_2)$, the duality ${\sim}(\psi_1 \otimes \psi_2) \equiv {\sim}\psi_1 \land {\sim}\psi_2$, and the definition $\mathsf{E}\psi = {\sim}\neg\psi$. The rightmost formula now states that for all types $\tau \in \Delta_0$ (each represented by a subset of Φ, cf. Proposition 4.6), there exists a world $w \in T$ such that $[\![R^i w]\!]_0^\Phi \subseteq \{\tau\}$. □

Next, we proceed with the inductive step, which is obtaining $(k+1)$-canonicity from k-canonicity. First, we provide some intuition in the simple case $\Phi = \emptyset$. For this, we can consider the formula $\forall^{\subseteq}_{\alpha} \exists^{1}_{\beta} \Box\chi^{*}_{k}(\alpha,\beta)$. It states that for every *subteam* $T' \subseteq T_\alpha$ there exists a *point* $w \in T_\beta$ such that $[\![RT']\!]_k = [\![Rw]\!]_k$. Intuitively, every possible set of types is captured as the image of some point in T_β. As a consequence, if T_α is k-canonical with offset 1, then T_β will be $(k+1)$-canonical.

Combining the above two ideas, k-canonicity with offset i is now recursively defined as ρ^i_k:

$$\begin{aligned} \rho^i_0(\beta) &:= \beta \hookrightarrow \mathsf{max}_i \\ \rho^i_{k+1}(\alpha,\beta) &:= \forall^{\subseteq}_{\alpha} \exists^{\subseteq}_{\beta} \left(\rho^i_0(\beta) \wedge \Box^i \forall^1_\beta \Box\chi^*_k(\alpha,\beta)\right) \\ \mathsf{canon}_k &:= \rho^k_0(\mathfrak{s}_0) \wedge \bigwedge_{m=1}^{k} \rho^{k-m}_m(\mathfrak{s}_{m-1},\mathfrak{s}_m) \end{aligned}$$

Theorem 4.24. *Let $k \geqslant 0$ and $\mathcal{K}$ be a structure with disjoint scopes $\mathfrak{s}_0,\ldots,\mathfrak{s}_k$. Then $(\mathcal{K},T) \vDash \mathsf{canon}_k$ if and only if $(\mathcal{K},T)$ is a k-staircase. Moreover, canon_k is an $\mathsf{ML}_k(\sim)$-formula constructible in space $\mathcal{O}(\log(|\Phi|+k))$.*

Proof. Similar to Theorem 4.19, the construction of the above formula in logspace is straightforward. We proceed with the correctness. Suppose that $\mathfrak{s}_0,\ldots,\mathfrak{s}_k$ are disjoint scopes in $\mathcal{K}$. We show the following by induction on $0 \leqslant i \leqslant k$: Assuming that T_α is k-canonical with offset $i+1$, it holds that T_β is $(k+1)$-canonical with offset i if and only if $T \vDash \rho^i_{k+1}(\alpha,\beta)$. With the induction basis done in Lemma 4.23, the inductive step is proved by the following chain of equivalences:

$$\begin{aligned} & T_\beta \text{ is } (k+1)\text{-canonical with offset } i \\ \Leftrightarrow\ & \forall \tau \in \Delta_{k+1} \colon \exists w \in T_\beta \colon [\![R^i w]\!]_{k+1} = \{\tau\} \end{aligned}$$

Using the bijection $h\colon \tau \mapsto (\tau \cap \Phi, \mathcal{R}\tau)$ from Proposition 4.6, we can equivalently quantify over $\wp\Delta_k$ and $\wp\Phi$:

$$\begin{aligned} \Leftrightarrow\ & \forall \Delta' \subseteq \Delta_k \colon \forall \Phi' \subseteq \Phi \colon \exists w \in T_\beta \colon [\![R^i w]\!]_{k+1} = \{h^{-1}(\Phi',\Delta')\} \\ \Leftrightarrow\ & \forall \Delta' \subseteq \Delta_k \colon \forall \Phi' \subseteq \Phi \colon \exists w \in T_\beta \colon R^i w \neq \emptyset \text{ and } \forall v \in R^i w \colon [\![v]\!]_{k+1} = h^{-1}(\Phi',\Delta') \end{aligned}$$

By Proposition 4.7, $V^{-1}(v) = \Phi'$ and $[\![Rv]\!]_k = \Delta'$ is equivalent to $[\![v]\!]_{k+1} = h^{-1}(\Phi',\Delta')$:

$$\begin{aligned} \Leftrightarrow\ & \forall \Delta' \subseteq \Delta_k \colon \forall \Phi' \subseteq \Phi \colon \exists w \in T_\beta \colon R^i w \neq \emptyset \\ & \qquad \text{and } \forall v \in R^i w \colon V^{-1}(v) = \Phi' \text{ and } [\![Rv]\!]_k = \Delta' \end{aligned}$$

Again by Proposition 4.6, $h\colon \tau \mapsto \tau \cap \Phi$ is a bijection from Δ_0 to $\wp(\Phi)$:

$$\begin{aligned} \Leftrightarrow\ & \forall \Delta' \subseteq \Delta_k \colon \forall \tau_0 \in \Delta_0 \colon \exists w \in T_\beta \colon R^i w \neq \emptyset \\ & \qquad \text{and } \forall v \in R^i w \colon V^{-1}(v) = \tau_0 \cap \Phi \text{ and } [\![Rv]\!]_k = \Delta' \end{aligned}$$

Once more by Proposition 4.7:

$$\begin{aligned}
&\Leftrightarrow \forall\Delta' \subseteq \Delta_k\colon \forall\tau_0 \in \Delta_0\colon \exists w \in T_\beta\colon R^i w \neq \emptyset \\
&\qquad \text{and } \forall v \in R^i w\colon [\![v]\!]_0 = \tau_0 \text{ and } [\![Rv]\!]_k = \Delta' \\
&\Leftrightarrow \forall\Delta' \subseteq \Delta_k\colon \forall\tau_0 \in \Delta_0\colon \exists w \in T_\beta\colon [\![R^i w]\!]_0 = \{\tau_0\} \text{ and } \forall v \in R^i w\colon [\![Rv]\!]_k = \Delta'
\end{aligned}$$

Since T_α is assumed k-canonical with offset $i+1$, for every $\tau' \in \Delta_k$ there exists $u \in T_\alpha$ such that $[\![R^{i+1}u]\!]_k = \{\tau'\}$. Accordingly, for every *set* $\Delta' \subseteq \Delta_k$ there exists $S \subseteq T_\alpha$ such that $[\![R^{i+1}S]\!]_k = \Delta'$:

$$\Leftrightarrow \forall S \subseteq T_\alpha\colon \forall\tau_0 \in \Delta_0\colon \exists w \in T_\beta\colon [\![R^i w]\!]_0 = \{\tau_0\} \text{ and } \forall v \in R^i w\colon [\![Rv]\!]_k = [\![R^{i+1}S]\!]_k$$

For each S, gather the respective w in a team $U \subseteq T_\beta$:

$$\begin{aligned}
&\Leftrightarrow \forall S \subseteq T_\alpha\colon \exists U \subseteq T_\beta\colon \big(\forall\tau_0 \in \Delta_0\colon \exists w \in U\colon [\![R^i w]\!]_0 = \{\tau_0\}\big) \\
&\qquad\qquad \text{and } \forall v \in R^i U\colon [\![Rv]\!]_k = [\![R^{i+1}S]\!]_k \\
&\Leftrightarrow \forall S \subseteq T_\alpha : \exists U \subseteq T_\beta : U \text{ is } 0\text{-canonical with offset } i \\
&\qquad\qquad \text{and } \forall v \in R^i U\colon [\![Rv]\!]_k = [\![R^{i+1}S]\!]_k
\end{aligned}$$

By the base case $k = 0$, and since $U = (T^{\alpha,\beta}_{S,U})_\beta$:

$$\Leftrightarrow \forall S \subseteq T_\alpha\colon \exists U \subseteq T_\beta\colon T^{\alpha,\beta}_{S,U} \vDash \rho^i_0(\beta) \text{ and } \forall v \in R^i U\colon [\![Rv]\!]_k = [\![R^{i+1}S]\!]_k$$

By Theorem 4.19:

$$\Leftrightarrow \forall S \subseteq T_\alpha\colon \exists U \subseteq T_\beta\colon T^{\alpha,\beta}_{S,U} \vDash \rho^i_0(\beta) \text{ and } \forall v \in R^i U\colon (R^{i+1}T)^{\alpha,\beta}_{R^{i+1}S,Rv} \vDash \chi^*_k(\alpha,\beta)$$

By Proposition 4.17 (6):

$$\Leftrightarrow \forall S \subseteq T_\alpha\colon \exists U \subseteq T_\beta\colon T^{\alpha,\beta}_{S,U} \vDash \rho^i_0(\beta) \text{ and } \forall v \in R^i U\colon (R^iT)^{\alpha,\beta}_{R^iS,v} \vDash \Box\chi^*_k(\alpha,\beta)$$

By Proposition 4.18 applied to $(R^iT)^{\alpha,\beta}_{R^iS,R^iU}$:

$$\Leftrightarrow \forall S \subseteq T_\alpha\colon \exists U \subseteq T_\beta\colon T^{\alpha,\beta}_{S,U} \vDash \rho^i_0(\beta) \text{ and } (R^iT)^{\alpha,\beta}_{R^iS,R^iU} \vDash \forall^1_\beta\Box\chi^*_k(\alpha,\beta)$$

Again by Proposition 4.17 (6) and Proposition 4.18:

$$\begin{aligned}
&\Leftrightarrow \forall S \subseteq T_\alpha\colon \exists U \subseteq T_\beta\colon T^{\alpha,\beta}_{S,U} \vDash \rho^i_0(\beta) \text{ and } R^i\left(T^{\alpha,\beta}_{S,U}\right) \vDash \forall^1_\beta\Box\chi^*_k(\alpha,\beta) \\
&\Leftrightarrow \forall S \subseteq T_\alpha\colon \exists U \subseteq T_\beta\colon T^{\alpha,\beta}_{S,U} \vDash \rho^i_0(\beta) \land \Box^i\forall^1_\beta\Box\chi^*_k(\alpha,\beta) \\
&\Leftrightarrow T \vDash \forall^{\subseteq}_\alpha \exists^{\subseteq}_\beta \left(\rho^i_0(\beta) \land \Box^i\forall^1_\beta\Box\chi^*_k(\alpha,\beta)\right) \\
&\Leftrightarrow T \vDash \rho^i_{k+1}(\alpha,\beta).
\end{aligned}$$

□

4.4.3 Enforcing scopes

As the next step, we lift the restriction of the $\mathfrak{s}_i$ being scopes *a priori*, because, in a sense, they are definable in $\mathsf{ML}(\sim)$ as well. For this, let $\psi \subseteq \mathsf{Prop}$ be finite and disjoint from Φ. Then the formula below ensures that ψ is a set of disjoint scopes "up to height k".

$$\mathsf{scopes}_k(\Psi) := \bigwedge_{\substack{p,q\in\Psi\\ p\neq q}} \neg(p \wedge q) \wedge \bigwedge_{p\in\Psi} \bigwedge_{i=1}^{k} \Big((p \wedge \Box^i p) \vee (\neg p \wedge \Box^i \neg p)\Big).$$

The definition up to height k is sufficient for our purposes, which follows from the next lemma.

Lemma 4.25. *If $\varphi \in \mathsf{ML}_k(\sim)$, then φ is satisfiable if and only if $\varphi \wedge \Box^{k+1}\bot$ is satisfiable.*

Proof. As the direction from right to left is trivial, suppose φ is satisfiable. By Corollary 4.22, it then has a model $(\mathcal{K}, T)$ that is a directed forest of height at most k. But then $(\mathcal{K}, T) \vDash \Box^{k+1}\bot$, since $R^{k+1}T = \emptyset$ and $(\mathcal{K}, \emptyset)$ satisfies all ML-formulas, including $\bot$. □

Theorem 4.26. $\mathsf{canon}_k \wedge \mathsf{scopes}_k(\{\mathfrak{s}_0, \ldots, \mathfrak{s}_k\}) \wedge \Box^{k+1}\bot$ *is satisfiable, but has only k-staircases as models.*

Proof. By combining Proposition 4.21, Theorem 4.24 and Lemma 4.25, the formula is satisfiable. Since in every model $(\mathcal{K}, T)$ the propositions $\mathfrak{s}_0, \ldots, \mathfrak{s}_k$ must be disjoint scopes due to $\Box^{k+1}\bot$ and scopes_k, we can apply Theorem 4.24. □

Like for bisimilarity, it is open whether (Φ, k)-canonicity can be defined in $\mathsf{ML}(\sim)^{\Phi}_{k}$ *efficiently* without restricting the models to those with scopes. Note that the model size lower bounds of this section do not imply that the brute force algorithm given in Theorem 4.13 is optimal from a complexity theoretic perspective, as there could be a satisfiability test that does not construct or a model. For a proper complexity theoretic hardness result, we need to encode computations in such models, to which we will proceed in the next sections.

4.5 Defining an order on types

In the previous section, we enforced k-canonicity with a formula, i.e., such that $|\Delta_k|$ different types are contained in the team. In order to encode computations of length $|\Delta_k|$, we additionally need to be able to talk about coordinates in time and space, and hence need an ordering of Δ_k.

Let us call any finite strict linear ordering simply an *order*. We specify an order $\prec_k$ on Δ_k, and an order $\prec_k^*$ on $\wp\Delta_k$. To begin with, let us first agree on some arbitrary order $<$ on Φ, say, $p_1 < p_2 < \cdots < p_{|\Phi|}$. Furthermore, if $\sqsubset$ is some order on some set X, then the *lexicographic order* $\sqsubset^*$ on $\wp X$ is defined by

$$X_1 \sqsubset^* X_2 \text{ iff } \exists x \in X_2 \setminus X_1 \text{ such that } \forall x' \in X\colon (x \sqsubset x') \Rightarrow (x' \in X_1 \Leftrightarrow x' \in X_2).$$

For example, let $X = \{0, 1\}$ and $0 \sqsubset 1$. Then $\emptyset \sqsubset^* \{0\} \sqsubset^* \{1\} \sqsubset^* \{0, 1\}$.

The order $\prec_k$ depends on the propositions true in a world, and otherwise recursively on the lexicographic order of the image team:

$$\tau \prec_0 \tau' \;\Leftrightarrow\; \tau \cap \Phi <^* \tau' \cap \Phi,$$
$$\tau \prec_{k+1} \tau' \;\Leftrightarrow\; \tau \cap \Phi <^* \tau' \cap \Phi \text{ or } (\tau \cap \Phi = \tau' \cap \Phi \text{ and } \mathcal{R}\tau \prec_k^* \mathcal{R}\tau').$$

It is easy to verify by induction that $\prec_k$ and $\prec_k^*$ are orders on Δ_k and $\wp\Delta_k$, respectively. The next step is to prove that $\prec_k$ and $\prec_k^*$ are (efficiently) definable in $\mathsf{ML}_k(\sim)$. For this, we pursue the same approach as for χ_k and χ_k^* in Section 4.2, and show that $\prec_k$ and $\prec_k^*$ are definable in formulas ζ_k and ζ_k^* in a mutually recursive fashion. Since order is a binary relation, the formulas below are once more parameterized by two scopes.

$$\zeta_0(\alpha, \beta) := \bigvee_{p \in \Phi} \Big[(\alpha \hookrightarrow \neg p) \wedge (\beta \hookrightarrow p) \wedge \bigwedge_{\substack{q \in \Phi \\ q < p}} (\alpha \vee \beta) \hookrightarrow \mathsf{dep}(q) \Big]$$

$$\zeta_{k+1}(\alpha, \beta) := \zeta_0(\alpha, \beta) \;\oslash\; \chi_0(\alpha, \beta) \wedge \Box \zeta_k^*(\alpha, \beta)$$

$$\zeta_k^*(\alpha, \beta) := \exists^1_{\mathfrak{s}_k} \left(\exists^1_\beta \chi_k(\mathfrak{s}_k, \beta)\right) \wedge \left({\sim}\exists^1_\alpha \chi_k(\mathfrak{s}_k, \alpha)\right)$$
$$\wedge \Big(\left(\chi_k^*(\alpha, \beta) \wedge (\alpha \vee \beta)\right) \vee \left(\forall^1_{\alpha \vee \beta} {\sim} \zeta_k(\mathfrak{s}_k, \alpha \vee \beta)\right) \Big)$$

Note that we make use of the scopes $\mathfrak{s}_0, \ldots, \mathfrak{s}_k$ in the formula, and in the following we restrict ourselves to k-staircase models. Moreover, in the subformula $\zeta_k(\mathfrak{s}_k, \alpha \vee \beta)$, we use the fact that $\alpha \vee \beta$ is a scope whenever α, β are scopes.

We require the next lemma for the correctness of ζ_k and ζ_k^*. Intuitively, it states that $\mathsf{ML}_k(\sim)$ is invariant under substitution of "locally equivalent" ML-formulas.

Lemma 4.27. *Let $\alpha, \beta \in \mathsf{ML}$ and $\varphi \in \mathsf{ML}_k(\sim)$. Let T be a team such that $R^i T \vDash \alpha \leftrightarrow \beta$ for all $i \in \{0, \ldots, k\}$. Then $T \vDash \varphi$ if and only if $T \vDash \varphi[\alpha/\beta]$.*

Proof. By straightforward induction; see the appendix. □

The following theorem states that in the class of k-staircase models (see the previous section) ζ_k and ζ_k^* define the required orders.

Theorem 4.28. *Let $k \geqslant 0$, and let $(\mathcal{K}, T)$ be a k-staircase with disjoint scopes $\mathfrak{s}_0, \ldots, \mathfrak{s}_k, \alpha, \beta$. If $w \in T_\alpha$ and $v \in T_\beta$, then*

$$T^{\alpha,\beta}_{w,v} \vDash \zeta_k(\alpha, \beta) \text{ if and only if } [\![w]\!]_k \prec_k [\![v]\!]_k,$$
$$T \vDash \zeta_k^*(\alpha, \beta) \text{ if and only if } [\![T_\alpha]\!]_k \prec_k^* [\![T_\beta]\!]_k.$$

Furthermore, both $\zeta_k(\alpha, \beta)$ and $\zeta_k^*(\alpha, \beta)$ are $\mathsf{ML}_k(\sim)$-formulas that are constructible in space $\mathcal{O}(\log(k + |\Phi| + |\alpha| + |\beta|))$.

We first give a rough idea of the proof, and after a series of lemmas fully prove the theorem. The definition of ζ_{k+1} simply follows the definition of $\prec_{k+1}$. Furthermore, the

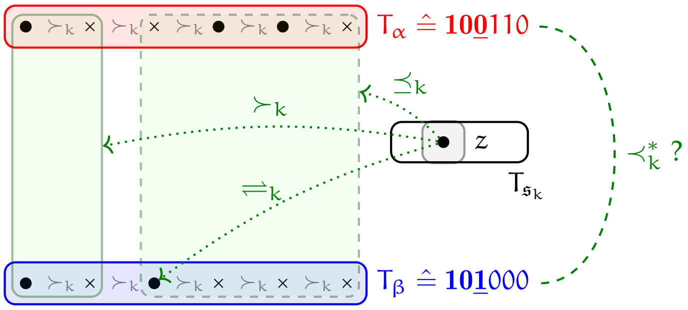


Figure 4.5: The pivot $z \in T_{\mathfrak{s}_k}$ **determines that** $[\![T_\alpha]\!]_k \prec_k^* [\![T_\beta]\!]_k$**. The subteam of** $T_{\alpha\vee\beta}$ **of worlds** $\prec_k$**-greater than** z **must satisfy** $\chi_k^*(\alpha,\beta)$**.**

formula ζ_k^* implements the lexicographic order $\prec_k^*$ as follows. As shown in Figure 4.5, some $z \in T_{\mathfrak{s}_k}$ acts as an *pivot* that witnesses $[\![T_\alpha]\!]_k \prec_k^* [\![T_\beta]\!]$, in the sense that it is the $\prec_k$-maximal type in which T_α and T_β differ.[1] The first line of ζ_k^* indeed expresses that $[\![z]\!]_k \in [\![T_\beta]\!]_k \setminus [\![T_\alpha]\!]_k$.

The disjunction in the second line intuitively states that we then can "split off" the subteam of $T_\alpha \cup T_\beta$ consisting of the elements $\prec_k$-greater than z (the solid green area in Figure 4.5), while χ_k^* ensures that they agree on the contained types (this reflects the part after the quantifier in the definition of $\sqsubset^*$). To achieve this, the subformula $\forall^1_{\alpha\vee\beta}{\sim}\zeta_k(\mathfrak{s}_k, \alpha\vee\beta)$ stipulates that any "remaining" elements from $T_\alpha \cup T_\beta$ possess only types not $\prec_k$-greater than $[\![z]\!]_k$ (the dashed green area in the figure).

Here, Lemma 4.27 is applied, as it ensures that after processing $\forall^1_{\alpha\vee\beta}$ the formula $\zeta_k(\mathfrak{s}_k, \alpha\vee\beta)$ in fact is either equivalent to $\zeta_k(\mathfrak{s}_k, \alpha)$ or to $\zeta_k(\mathfrak{s}_k, \beta)$; and hence behaves correctly by induction hypothesis.

Next, we come to the formal proof, which requires a series of lemmas and the following definition.

Definition 4.29. Let $k \geqslant 0$. Let α, β be disjoint scopes and T a team in a Kripke structure. Then α and β are called $\prec_k$-*comparable in* T if for all $w \in T_\alpha, v \in T_\beta$

$$T^{\alpha,\beta}_{w,v} \vDash \zeta_k(\alpha,\beta) \text{ iff } [\![w]\!]_k \prec_k [\![v]\!]_k \text{ and}$$
$$T^{\alpha,\beta}_{w,v} \vDash \zeta_k(\beta,\alpha) \text{ iff } [\![v]\!]_k \prec_k [\![w]\!]_k.$$

Likewise, α and β are $\prec_k^*$-*comparable in* T if

$$T \vDash \zeta_k^*(\alpha,\beta) \text{ iff } [\![T_\alpha]\!]_k \prec_k^* [\![T_\beta]\!]_k \text{ and}$$
$$T \vDash \zeta_k^*(\beta,\alpha) \text{ iff } [\![T_\beta]\!]_k \prec_k^* [\![T_\alpha]\!]_k.$$

The next lemma shows that the correctness of $\prec_k^*$ follows from that of $\prec_k$.

[1]Since the pivot is selected from $T_{\mathfrak{s}_k}$, at this point it is crucial that the underlying structure is a k-staircase, so that any k-type can be potentially picked.

Lemma 4.30. *Suppose that* $(\mathcal{K}, T)$ *is a* k*-staircase with disjoint scopes* $\alpha, \beta, \mathfrak{s}_0, \dots, \mathfrak{s}_k$*. If both* α *and* β *are* $\prec_k$*-comparable to* $\mathfrak{s}_k$ *in all subteams* S *of the form* $T_{\mathfrak{s}_0} \cup \dots \cup T_{\mathfrak{s}_{k-1}} \subseteq S \subseteq T$*, then* α *and* β *are* $\prec_k^*$*-comparable in* T*.*

Proof. Assuming $\mathcal{K}, T, \alpha, \beta, \mathfrak{s}_0, \dots, \mathfrak{s}_k$ as above, the proof is split into the following claims.

Claim (a). *The disjoint scopes* $\alpha \vee \beta$ *and* $\mathfrak{s}_k$ *are* $\prec_k$*-comparable in any team* S *that satisfies* $T_{\mathfrak{s}_0} \cup \dots \cup T_{\mathfrak{s}_{k-1}} \subseteq S \subseteq T$*.*

Proof of claim. Let $w \in S_{\alpha\vee\beta}$ and $v \in S_{\mathfrak{s}_k}$. W.l.o.g. $w \in S_\alpha$ (the case $w \in S_\beta$ works analogously). Then

$$
\begin{aligned}
& S^{\alpha\vee\beta,\mathfrak{s}_k}_{w,v} \vDash \zeta_k(\alpha \vee \beta, \mathfrak{s}_k) \\
\Leftrightarrow\ & S^{\alpha,\beta,\mathfrak{s}_k}_{w,\emptyset,v} \vDash \zeta_k(\alpha \vee \beta, \mathfrak{s}_k) && \text{(since } S^{\alpha\vee\beta,\mathfrak{s}_k}_{w,v} = S^{\alpha,\beta,\mathfrak{s}_k}_{w,\emptyset,v}\text{)} \\
\Leftrightarrow\ & S^{\alpha,\beta,\mathfrak{s}_k}_{w,\emptyset,v} \vDash \zeta_k(\alpha, \mathfrak{s}_k) && \text{(by Lemma 4.27, as } \textstyle\bigcup_{i=0}^k R^i S^{\alpha,\beta,\mathfrak{s}_k}_{w,\emptyset,v} \vDash \alpha \leftrightarrow (\alpha \vee \beta)\text{)} \\
\Leftrightarrow\ & [\![w]\!]_k \prec_k [\![v]\!]_k. && \text{(by assumption of the lemma)}
\end{aligned}
$$

The case $\zeta_k(\mathfrak{s}_k, \alpha \vee \beta)$ is symmetric. ◁

For the remaining proof, we omit the subscript k when referring to types and $\prec$. Furthermore, for all $\tau \in \Delta_k$, let $[\![T]\!]^\tau$ denote the restriction of $[\![T]\!]$ to types $\tau' \succ \tau$. Intuitively, these types are the "more significant positions" for the lexicographic ordering. In the next claim, we essentially show that the second line in the definition of $\zeta_k^*(\alpha, \beta)$ can be expressed as a statement of the form $[\![T_\alpha]\!]^\tau = [\![T_\beta]\!]^\tau$.

Claim (b). *Let* T *be a team and* $\tau \in \Delta_k$*. Then* $[\![T_\alpha]\!]^\tau = [\![T_\beta]\!]^\tau$ *if and only if there exists* $S \subseteq T_{\alpha\vee\beta}$ *such that* $[\![S_\alpha]\!] = [\![S_\beta]\!]$ *and* $[\![w]\!] \not\succ \tau$ *for all* $w \in T_{\alpha\vee\beta} \setminus S$*.*

Proof of claim. "$\Rightarrow$": Simply define $S := \{v \in T_{\alpha\vee\beta} \mid [\![v]\!] \succ \tau\}$. Then $[\![S_\alpha]\!] = [\![T_\alpha]\!]^\tau = [\![T_\beta]\!]^\tau = [\![S_\beta]\!]$. Moreover, for every $w \in T_{\alpha\vee\beta} \setminus S$ clearly $[\![w]\!] \not\succ \tau$ holds.

"$\Leftarrow$": Assume that S exists as stated in the claim. By symmetry, we only prove $[\![T_\alpha]\!]^\tau \subseteq [\![T_\beta]\!]^\tau$. Consequently, let $w \in T_\alpha$ such that $[\![w]\!] \in [\![T_\alpha]\!]^\tau$. Then $[\![w]\!] \succ \tau$ by definition. But then $w \notin T_{\alpha\vee\beta} \setminus S$. However, we have $w \in T_\alpha$, hence $w \in T_{\alpha\vee\beta}$, which only leaves the possibility $w \in S$. Combining $w \in S$ and $w \in T_\alpha$ yields $w \in S_\alpha$, which by assumption also implies $[\![w]\!] \in [\![S_\beta]\!]$. As $[\![S_\beta]\!] \subseteq [\![T_\beta]\!]$ and $[\![w]\!] \succ \tau$, the membership $[\![w]\!] \in [\![T_\beta]\!]^\tau$ follows. ◁

We finish the proof of Lemma 4.30 in the final claim below.

Claim (c). α *and* β *are* $\prec_k^*$*-comparable in* T*.*

Proof of claim. Due to symmetry, we prove only that $T \vDash \zeta_k^*(\alpha, \beta)$ iff $[\![T_\alpha]\!]_k \prec_k^* [\![T_\beta]\!]_k$.

$$
\begin{aligned}
& [\![T_\alpha]\!] \prec^* [\![T_\beta]\!] \\
\Leftrightarrow\ & \exists \tau \in [\![T_\beta]\!] \setminus [\![T_\alpha]\!] \colon \forall \tau' \in \Delta, \tau \prec \tau' \colon \tau' \in [\![T_\alpha]\!] \Leftrightarrow \tau' \in [\![T_\beta]\!] && \text{(def. } \prec_k^*\text{)} \\
\Leftrightarrow\ & \exists \tau \in [\![T_\beta]\!] \setminus [\![T_\alpha]\!] \colon [\![T_\alpha]\!]^\tau = [\![T_\beta]\!]^\tau && \text{(def. } [\![\cdot]\!]^\tau\text{)}
\end{aligned}
$$

Since $T_{\mathfrak{s}_k}$ is k-canonical, for every $\tau \in \Delta_k$ there exists $z \in T_{\mathfrak{s}_k}$ of type τ:

$$\Leftrightarrow \exists z \in T_{\mathfrak{s}_k} \colon [\![T_\alpha]\!]^{[\![z]\!]} = [\![T_\beta]\!]^{[\![z]\!]} \text{ and } [\![z]\!] \in [\![T_\beta]\!] \setminus [\![T_\alpha]\!]$$
$$\Leftrightarrow \exists z \in T_{\mathfrak{s}_k} \colon [\![T_\alpha]\!]^{[\![z]\!]} = [\![T_\beta]\!]^{[\![z]\!]} \text{ and } \exists x \in T_\beta \colon [\![z]\!] = [\![x]\!] \text{ and } \nexists y \in T_\alpha \colon [\![z]\!] = [\![y]\!]$$

As α, β and $\mathfrak{s}_k$ are disjoint, we have $T_\alpha = O_\alpha$ and $T_\beta = O_\beta$ for $O := T_z^{\mathfrak{s}_k}$:

$$\Leftrightarrow \exists z \in T_{\mathfrak{s}_k} \colon [\![O_\alpha]\!]^{[\![z]\!]} = [\![O_\beta]\!]^{[\![z]\!]} \text{ and } \exists x \in O_\beta \colon [\![z]\!] = [\![x]\!] \text{ and } \nexists y \in O_\alpha \colon [\![z]\!] = [\![y]\!]$$
$$\Leftrightarrow \exists z \in T_{\mathfrak{s}_k} \colon \exists x \in O_\beta \colon [\![z]\!] = [\![x]\!] \text{ and } \nexists y \in O_\alpha \colon [\![z]\!] = [\![y]\!]$$
$$\text{and } \exists S \subseteq O_{\alpha\vee\beta} \colon [\![S_\alpha]\!] = [\![S_\beta]\!] \text{ and } \forall w \in O_{\alpha\vee\beta} \setminus S \colon [\![z]\!] \not\prec [\![w]\!] \qquad \text{(Claim (b))}$$

Clearly, S is a subteam of $O_{\alpha\vee\beta}$ if and only if it is a subteam of O and satisfies $\alpha \vee \beta$:

$$\Leftrightarrow \exists z \in T_{\mathfrak{s}_k} \colon \exists x \in O_\beta \colon [\![z]\!] = [\![x]\!] \text{ and } \nexists y \in O_\alpha \colon [\![z]\!] = [\![y]\!]$$
$$\text{and } \exists S \subseteq O \colon [\![S_\alpha]\!] = [\![S_\beta]\!] \text{ and } S \vDash \alpha \vee \beta \text{ and } \forall w \in O_{\alpha\vee\beta} \setminus S \colon [\![z]\!] \not\prec [\![w]\!]$$

Observe that the property $\forall w \in U \colon [\![z]\!] \not\prec [\![w]\!]$ is downward closed in U and hence holds for $U = O \setminus S$ iff it holds for any $U \supseteq O \setminus S$:

$$\Leftrightarrow \exists z \in T_{\mathfrak{s}_k} \colon \exists x \in O_\beta \colon [\![z]\!] = [\![x]\!] \text{ and } \nexists y \in O_\alpha \colon [\![z]\!] = [\![y]\!]$$
$$\text{and } \exists S \subseteq O \colon [\![S_\alpha]\!] = [\![S_\beta]\!] \text{ and } S \vDash \alpha \vee \beta$$
$$\text{and } \exists U \subseteq O \colon U \supseteq O \setminus S \text{ and } \forall w \in U_{\alpha\vee\beta} \colon [\![z]\!] \not\prec [\![w]\!]$$

By Theorem 4.19:

$$\Leftrightarrow \exists z \in T_{\mathfrak{s}_k} \colon O \vDash (\exists^1_\beta \chi_k(\mathfrak{s}, \beta)) \wedge ({\sim}\exists^1_\alpha \chi_k(\mathfrak{s}, \alpha)) \text{ and } \exists S \subseteq O \colon$$
$$S \vDash (\alpha \vee \beta) \wedge \chi^*_k(\alpha, \beta) \text{ and } \exists U \subseteq O \colon U \supseteq O \setminus S \text{ and } \forall w \in U_{\alpha\vee\beta} \colon [\![z]\!] \not\prec [\![w]\!]$$

Note that $T_{\mathfrak{s}_0}, \ldots, T_{\mathfrak{s}_{k-1}}$ are retained in O. Moreover, $S \subseteq O_{\alpha\vee\beta}$, which implies that they are still subteams of $O \setminus S$ and hence of U. But by Claim (a), $\alpha \vee \beta$ and $\mathfrak{s}_k$ are then $\prec_k$-comparable scopes in U and we can replace $[\![z]\!] \not\prec [\![w]\!]$:

$$\Leftrightarrow \exists z \in T_{\mathfrak{s}_k} \colon O \vDash (\exists^1_\beta \chi_k(\mathfrak{s}, \beta)) \wedge ({\sim}\exists^1_\alpha \chi_k(\mathfrak{s}, \alpha)) \text{ and } \exists S \subseteq O \colon S \vDash (\alpha \vee \beta) \wedge \chi^*_k(\alpha, \beta)$$
$$\text{and } \exists U \subseteq O \colon U \supseteq O \setminus S \text{ and } \forall w \in U_{\alpha\vee\beta} \colon U_w^{\alpha\vee\beta} \vDash {\sim}\zeta_k(\mathfrak{s}_k, \alpha \vee \beta)$$

Recalling that $O = T_z^{\mathfrak{s}_k}$, and by Proposition 4.18, we obtain:

$$\Leftrightarrow \exists z \in T_{\mathfrak{s}_k} \colon T_z^{\mathfrak{s}_k} \vDash (\exists^1_\beta \chi_k(\mathfrak{s}, \beta)) \wedge ({\sim}\exists^1_\alpha \chi_k(\mathfrak{s}, \alpha)) \text{ and } \exists S \subseteq T_z^{\mathfrak{s}_k} \colon S \vDash (\alpha \vee \beta) \wedge \chi^*_k(\alpha, \beta)$$
$$\text{and } \exists U \subseteq T_z^{\mathfrak{s}_k} \colon U \supseteq T_z^{\mathfrak{s}_k} \setminus S \text{ and } U \vDash \forall^1_{\alpha\vee\beta} {\sim}\zeta_k(\mathfrak{s}_k, \alpha \vee \beta)$$
$$\Leftrightarrow T \vDash \exists^1_{\mathfrak{s}_k} (\exists^1_\beta \chi_k(\mathfrak{s}, \beta)) \wedge ({\sim}\exists^1_\alpha \chi_k(\mathfrak{s}, \alpha))$$
$$\wedge \big((\alpha \vee \beta) \wedge \chi^*_k(\alpha, \beta)\big) \vee \big(\forall^1_{\alpha\vee\beta} {\sim}\zeta_k(\mathfrak{s}_k, \alpha \vee \beta)\big)$$
$$\Leftrightarrow T \vDash \zeta^*(\alpha, \beta). \qquad \triangleleft \quad \square$$

In the next lemma, we prove the converse direction of Lemma 4.30.

Lemma 4.31. *Let $k > 0$, and let $(\mathcal{K}, T)$ be a k-staircase with disjoint scopes $\alpha, \beta, \mathfrak{s}_0, \ldots, \mathfrak{s}_{k-1}$. Then α and β are $\prec_k$-comparable in every subteam S of T that contains $T_{\mathfrak{s}_0} \cup \cdots \cup T_{\mathfrak{s}_{k-1}}$.*

Proof. The proof is by induction on k. Disjoint scopes α and β are always $\prec_0$-comparable, which can be easily seen in ζ_0. For the inductive step to $k+1$, assume $(\mathcal{K}, T)$ and S as above, and let $\mathcal{K} = (W, R, V)$. Let $O := S^{\alpha,\beta}_{w,v}$ with $w \in S_\alpha, v \in S_\beta$ arbitrary.

Claim (d). *α and β are $\prec^*_k$-comparable in RO.*

Proof of claim. In the inductive step, now $\mathfrak{s}_0, \ldots, \mathfrak{s}_k, \alpha, \beta$ are disjoint scopes. Additionally, $(\mathcal{K}, RT)$ is a k-staircase. In particular, in the induction step, α and β are disjoint from $\mathfrak{s}_k$. For this reason, $(\mathcal{K}, RO)$ is a k-staircase as well, as $(RO)_{\mathfrak{s}_0 \vee \cdots \vee \mathfrak{s}_k} = (RT)_{\mathfrak{s}_0 \vee \cdots \vee \mathfrak{s}_k}$.

Hence, by induction hypothesis, for every team U such that $RO_{\mathfrak{s}_0} \cup \cdots \cup RO_{\mathfrak{s}_{k-1}} \subseteq U \subseteq RO$, we obtain that $\mathfrak{s}_k$ and α are $\prec_k$-comparable in U, as well as $\mathfrak{s}_k$ and β. Consequently, we can apply Lemma 4.30, which proves the claim. $\lhd$

We proceed with the induction step. Again by symmetry, we only show that $O \vDash \zeta_{k+1}(\alpha, \beta)$ iff $[\![w]\!]_{k+1} \prec_{k+1} [\![v]\!]_{k+1}$. We distinguish three cases w. r. t. $\prec_0$:

- If $[\![w]\!]_0 \prec_0 [\![v]\!]_0$, then $O \vDash \zeta_0(\alpha, \beta)$ by the induction basis. As the former implies $[\![w]\!]_{k+1} \prec_{k+1} [\![v]\!]_{k+1}$ and the latter $O \vDash \zeta_{k+1}(\alpha, \beta)$, the equivalence holds.

- If $[\![w]\!]_0 \succ_0 [\![v]\!]_0$, then $[\![w]\!]_{k+1} \not\prec_{k+1} [\![v]\!]_{k+1}$. Moreover, $O \nvDash \zeta_0(\alpha, \beta)$ by induction basis. Additionally, $O \nvDash \chi_0(\alpha, \beta)$ by Theorem 4.19. Consequently, both sides of the equivalence are false.

- If $[\![w]\!]_0 = [\![v]\!]_0$, then $O \vDash \chi_0(\alpha, \beta)$ by Theorem 4.19, but $O \nvDash \zeta_0(\alpha, \beta)$ by induction basis. Consequently, $O \vDash \zeta_{k+1}(\alpha, \beta)$ iff $O \vDash \Box\zeta^*_k(\alpha, \beta)$. Also, $[\![w]\!]_{k+1} \prec_{k+1} [\![v]\!]_{k+1}$ iff $\mathcal{R}[\![w]\!]_{k+1} \prec^*_k \mathcal{R}[\![v]\!]_{k+1}$. The following equivalence concludes the proof:

$$\begin{aligned} & \mathcal{R}[\![w]\!]_{k+1} \prec^*_k \mathcal{R}[\![v]\!]_{k+1} \\ \Leftrightarrow\ & [\![Rw]\!]_k \prec^*_k [\![Rv]\!]_k && \text{(Proposition 4.6)} \\ \Leftrightarrow\ & RO \vDash \zeta^*_k(\alpha, \beta) && \text{(Claim (d))} \\ \Leftrightarrow\ & O \vDash \Box\zeta^*_k(\alpha, \beta). && \Box \end{aligned}$$

With the above lemmas, we are now in the position to prove Theorem 4.28:

Proof of Theorem 4.28. First, it is straightforward to construct ζ_k and ζ^*_k in logarithmic space. For the correctness, let $(\mathcal{K}, T)$ be a model with disjoint scopes $\alpha, \beta, \mathfrak{s}_0, \ldots, \mathfrak{s}_k$ as in the theorem. By Lemma 4.31 it immediately follows that α and β are $\prec_k$-comparable in T. The second part, that α and β are $\prec^*_k$-comparable in T, follows from the combination of Lemma 4.30 and 4.31. $\Box$

4.6 Encoding non-elementary computations

We combine all the previous sections of this chapter, and complement Theorem 4.13 and Corollary 4.14 with their matching lower bounds:

Theorem 4.32.

- $\mathrm{SAT}(\mathsf{ML}(\sim))$ *and* $\mathrm{VAL}(\mathsf{ML}(\sim))$ *are complete for* $\textsc{Tower}(\mathrm{poly})$.
- *If* $k \geqslant 0$*, then* $\mathrm{SAT}(\mathsf{ML}_k(\sim))$ *and* $\mathrm{VAL}(\mathsf{ML}_k(\sim))$ *are complete for* $\mathrm{AT}\textsc{ime}\text{-}\textsc{Alt}(\exp_{k+1}, \mathrm{poly})$.

The above complexity classes are complement-closed, and additionally $\mathsf{ML}(\sim)$ and $\mathsf{ML}_k(\sim)$ are syntactically closed under negation. For this reason, it suffices to prove the hardness of $\mathrm{SAT}(\mathsf{ML}(\sim))$ and $\mathrm{SAT}(\mathsf{ML}_k(\sim))$, respectively. Moreover, the case $k = 0$ is equivalent to $\mathrm{SAT}(\mathsf{PL}(\sim))$ being $\mathrm{AT}\textsc{ime}\text{-}\textsc{Alt}(\exp, \mathrm{poly})$-hard, which was proven by Hannula et al. [56]. Their reduction also works in logarithmic space. Consequently, the result comes down to the following lemma:

Lemma 4.33.

- *If* $A \in \textsc{Tower}(\mathrm{poly})$*, then* $A \leqslant_m^{\log} \mathrm{SAT}(\mathsf{ML}(\sim))$.
- *If* $k \geqslant 1$ *and* $A \in \mathrm{AT}\textsc{ime}\text{-}\textsc{Alt}(\exp_{k+1}, \mathrm{poly})$*, then* $A \leqslant_m^{\log} \mathrm{SAT}(\mathsf{ML}_k(\sim))$.

We devise for each A a reduction $x \mapsto \varphi_x$ such that φ_x is a formula that is satisfiable if and only if $x \in A$. By assumption, there exists a single-tape alternating Turing machine M that decides A in suitable runtime. For the case of $\textsc{Tower}(\mathrm{poly})$, clearly a deterministic machine is a special case of an alternating one.

Let Q be the set of states of M, and the disjoint union of $Q_\exists$ (*existential states*), $Q_\forall$ (*universal states*), Q_{acc} (*accepting states*) and Q_{rej} (*rejecting states*). Also, let Q contain the initial state q_0. Let M have a finite tape alphabet Γ with blank symbol $\flat \in \Gamma$, and a transition relation δ.

We design φ_x in a fashion that forces its models $(\mathcal{K}, T)$ to encode an accepting computation of M on x. Let us call any legal sequence of configurations of M (not necessarily starting with the initial configuration) a *run*. Then, similarly to Cook's theorem [18], we encode runs as tableaus, which can be depicted as square grids with a vertical time coordinate and a horizontal space coordinate, i.e., each row of the grid represents a configuration of M. In what follows, let $x = x_1 \cdots x_n$ be some input, i.e., $|x| = n$.

W.l.o.g. M never leaves the input to the left, and there exists N that is an upper bound on both the length of a configuration and the runtime of M. Formally, a run of M is then a function $C\colon \{1, \ldots, N\}^2 \to \Gamma \cup (Q \times \Gamma)$, Here, $C(i, j) = c$ for $c \in \Gamma$ means that the i-th configuration (i.e., after M performed $i - 1$ transitions) contains the symbol c in its j-th cell. The same holds if $C(i, j) = (q, c)$ for $(q, c) \in Q \times \Gamma$, but then additionally the machine is in the state q with its head visiting the j-th cell in the i-th configuration. As an example, if C starts from M's initial configuration then we have $C(1, 1) = (q_0, x_1)$, $C(1, i) = x_i$ for $2 \leqslant i \leqslant n$, and $C(1, i) = \flat$ for $n < i \leqslant N$.

Due to the semantics of $\mathsf{ML}(\sim)$, such a run must be encoded in $(\mathcal{K}, T)$ very carefully. We force a team T to contain N^2 worlds $w_{i,j}$ in which the respective value of $C(i,j)$ is encoded as a propositional assignment. However, we cannot enforce an actual $N \times N$-grid in the frame of $\mathcal{K}$, as by Corollary 4.22, we cannot force the model to even contain R-paths longer than $\mathsf{md}(\varphi_x)$. Instead, we define grid neighborship indirectly. The idea is to encode i and j in $w_{i,j}$ by its *type*. More precisely, we use the linear order $\prec_k$ on Δ_k we defined with the $\mathsf{ML}_k(\sim)$-formula ζ_k in the previous section. Then, instead of using $\Box$ and $\Diamond$, we examine the grid by letting ζ_k judge whether a given pair of worlds is deemed (horizontally or vertically) adjacent.

4.6.1 Encoding runs in a team

Next, we discuss how runs $C\colon \{1,\ldots,N\}^2 \to \Gamma \cup (Q \times \Gamma)$ are encoded in T. Given a world $w \in T$, we partition the image Rw with two fresh propositions $\mathfrak{t}$ ("timestep") and $\mathfrak{p}$ ("position"). Then we assign to w its *location* $\ell(w)$, which is the pair (i,j) such that $[\![(Rw)_{\mathfrak{t}}]\!]_{k-1}$ is the i-th element, and $[\![(Rw)_{\mathfrak{p}}]\!]_{k-1}$ is the j-th element in the order $\prec^*_{k-1}$.

The space of such representable coordinates is $N := |\wp(\Delta^{\Phi}_{k-1})|$. For the reduction from the class ATIME-ALT$(\exp_{k+1}, \text{poly})$, M has runtime bounded by $\exp_{k+1}(g(n))$ for a polynomial g. Then taking $\Phi := \{p_1, \ldots, p_{g(n)}\}$ yields sufficiently large coordinates, as

$$\exp_{k+1}(g(n)) = \exp_{k+1}(|\Phi|) = 2^{\exp_{k-1}(2^{|\Phi|})} \leqslant 2^{\exp^*_{k-1}(2^{|\Phi|})} = 2^{|\Delta^{\Phi}_{k-1}|} = N$$

by Proposition 4.10. Likewise for the class TOWER(poly) where M has runtime $\exp_{g(n)}(1)$ for a polynomial g: We define $\Phi := \emptyset$ and compute $k := g(|x|) + 1$ in the reduction, but otherwise proceed identically.

Next, let Ξ be a set of propositions disjoint from Φ that encodes $\Gamma \cup (Q \times \Gamma)$, formally there is a bijection $c\colon \Xi \to \Gamma \cup (Q \times \Gamma)$. If a world w satisfies exactly one proposition p of those in Ξ, then by slight abuse of notation we write $c(w)$ instead of $c(p)$. Intuitively, $c(w) \in \Gamma \cup (Q \times \Gamma)$ is the content of the grid cell at location $\ell(w)$.

Using ℓ and c, the function C can be encoded into a team T as follows. First, a team T is called *grid* if every point in T satisfies exactly one proposition in Ξ, and every pair $(i,j) \in \{1,\ldots,N\}^2$ occurs as the location $\ell(w)$ of some point $w \in T$. Moreover, a grid T is called *pre-tableau* if for every pair (i,j) *and* every element $p \in \Xi$ there is some world $w \in T$ such that $\ell(w) = (i,j)$ and $w \vDash p$. Finally, a grid T is a *tableau* if any two elements $w, w' \in T$ with $\ell(w) = \ell(w')$ also agree on Ξ, i.e., $c(w) = c(w')$.

Let us motivate the above definitions. Clearly, the definition of a *grid* T means that T captures the whole domain of C, and that c is well-defined on the level of *points*. If T is additionally a *tableau*, then c is also well-defined on the level of *locations*. In other words, a tableau T induces a function $C_T\colon \{1,\ldots,N\}^2 \to \Gamma \cup (Q \times \Gamma)$ via $C_\alpha(i,j) := c(w)$, where $w \in T$ is arbitrary such that $\ell(w) = (i,j)$.

A *pre-tableau* can be seen as the union of all possible tableaus. In particular, given any pre-tableau, with this definition we ensure that arbitrary tableaus can be obtained from it by the means of subteam quantification (cf. p. 84).

A tableau T is *legal* if C_T is a run of M, i.e., if every row is a configuration of M, and if every pair of two successive rows represents a valid δ-transition.

The idea of the reduction is now to capture the alternating computation of M by nesting polynomially many quantifications (via $\exists^{\subseteq}$ and $\forall^{\subseteq}$) of legal tableaus, of which each one continues the computation of the previous one.

4.6.2 Accessing two components of locations

As discussed earlier, we choose to represent a location (i, j) in a point w as a pair (Δ', Δ'') by stipulating that $\Delta' = [\![(Rw)_{\mathfrak{t}}]\!]_{k-1}$ and $\Delta'' = [\![(Rw)_{\mathfrak{p}}]\!]_{k-1}$. To access the two components of an encoded location independently, we introduce the shorthand

$$|^{\alpha}_{\mathfrak{q}}\,\psi := (\alpha \wedge \neg \mathfrak{q}) \vee ((\alpha \hookrightarrow \mathfrak{q}) \wedge \psi),$$

where $\mathfrak{q} \in \{\mathfrak{t}, \mathfrak{p}\}$ and $\alpha \in \mathsf{ML}$. It is easy to check that $T \vDash |^{\alpha}_{\mathfrak{q}}\,\psi$ iff $T^{\alpha}_{T_{\mathfrak{q}}} \vDash \psi$.

In order to *compare* the locations of grid cells, for each component $\mathfrak{q} \in \{\mathfrak{t}, \mathfrak{p}\}$ we define the following formulas. Assuming singleton teams T_α and T_β, $\psi^{\mathfrak{q}}_{\prec}(\alpha, \beta)$ tests whether the location in T_α is less than that in T_β w. r. t. its $\mathfrak{q}$-component. Analogously, $\psi^{\mathfrak{q}}_{\equiv}(\alpha, \beta)$ checks for equality of the respective components:

$$\psi^{\mathfrak{q}}_{\prec}(\alpha, \beta) := \Box\, |^{\alpha}_{\mathfrak{q}} |^{\beta}_{\mathfrak{q}}\, \zeta^{*}_{k-1}(\alpha, \beta)$$
$$\psi^{\mathfrak{q}}_{\equiv}(\alpha, \beta) := \Box\, |^{\alpha}_{\mathfrak{q}} |^{\beta}_{\mathfrak{q}}\, \chi^{*}_{k-1}(\alpha, \beta)$$

For this purpose, $\psi^{\mathfrak{q}}_{\prec}$ is built upon the formula ζ^{*}_{k-1} from Theorem 4.28, while $\psi^{\mathfrak{q}}_{\equiv}$ checks for equality using χ^{*}_{k-1} from Theorem 4.19.

Claim (e). *Let $\mathcal{K}$ be a structure with a team T and disjoint scopes α and β. Suppose $w \in T_\alpha$ and $v \in T_\beta$, where $\ell(w) = (i_w, j_w)$ and $\ell(v) = (i_v, j_v)$. Then:*

$$T^{\alpha,\beta}_{w,v} \vDash \psi^{\mathfrak{t}}_{\equiv}(\alpha, \beta) \;\Leftrightarrow\; i_w = i_v$$
$$T^{\alpha,\beta}_{w,v} \vDash \psi^{\mathfrak{p}}_{\equiv}(\alpha, \beta) \;\Leftrightarrow\; j_w = j_v$$

Moreover, if $\alpha, \beta, \mathfrak{s}_0, \ldots, \mathfrak{s}_k$ are disjoint scopes in $\mathcal{K}$ and $(\mathcal{K}, T)$ is a k*-staircase, then:*

$$T^{\alpha,\beta}_{w,v} \vDash \psi^{\mathfrak{t}}_{\prec}(\alpha, \beta) \;\Leftrightarrow\; i_w < i_v$$
$$T^{\alpha,\beta}_{w,v} \vDash \psi^{\mathfrak{p}}_{\prec}(\alpha, \beta) \;\Leftrightarrow\; j_w < j_v$$

Proof of claim. Let us begin with $\psi^{\mathfrak{t}}_{\equiv}$ (then $\psi^{\mathfrak{p}}_{\equiv}$ works identically):

$$\begin{aligned} i_w = i_v \;&\Leftrightarrow [\![(Rw)_{\mathfrak{t}}]\!]_{k-1} = [\![(Rv)_{\mathfrak{t}}]\!]_{k-1} && \text{(def.)} \\ &\Leftrightarrow RT^{\alpha,\beta}_{(Rw)_{\mathfrak{t}},(Rv)_{\mathfrak{t}}} \vDash \chi^{*}_{k-1}(\alpha, \beta) && \text{(Theorem 4.19)} \\ &\Leftrightarrow \left(RT^{\alpha,\beta}_{Rw,Rv}\right)^{\alpha,\beta}_{RT_{\mathfrak{t}},RT_{\mathfrak{t}}} \vDash \chi^{*}_{k-1}(\alpha, \beta) \\ &\Leftrightarrow RT^{\alpha,\beta}_{Rw,Rv} \vDash |^{\alpha}_{\mathfrak{t}} |^{\beta}_{\mathfrak{t}}\, \chi^{*}_{k-1}(\alpha, \beta) \end{aligned}$$

$$\Leftrightarrow T^{\alpha,\beta}_{w,v} \vDash \Box |^{\alpha}_{t}|^{\beta}_{t} \chi^{*}_{k-1}(\alpha,\beta) \qquad \text{(Proposition 4.17)}$$

Similarly for $\psi^{t}_{\prec}$ (then $\psi^{p}_{\prec}$ again works identically):

$$\begin{aligned} i_w < i_v &\Leftrightarrow [\![(Rw)_t]\!]_{k-1} \prec^{*}_{k-1} [\![(Rv)_t]\!]_{k-1} && \text{(def.)} \\ &\Leftrightarrow RT^{\alpha,\beta}_{(Rw)_t,(Rv)_t} \vDash \zeta^{*}_{k-1}(\alpha,\beta) && \text{(Theorem 4.28)} \\ &\Leftrightarrow \left(RT^{\alpha,\beta}_{Rw,Rv}\right)^{\alpha,\beta}_{T_t,T_t} \vDash \zeta^{*}_{k-1}(\alpha,\beta) \\ &\Leftrightarrow RT^{\alpha,\beta}_{Rw,Rv} \vDash |^{\alpha}_{t}|^{\beta}_{t} \zeta^{*}_{k-1}(\alpha,\beta) \\ &\Leftrightarrow T^{\alpha,\beta}_{w,v} \vDash \Box |^{\alpha}_{t}|^{\beta}_{t} \zeta^{*}_{k-1}(\alpha,\beta) && \text{(Proposition 4.17)} \ \lhd \end{aligned}$$

4.6.3 Defining grids, pre-tableaus, and tableaus

Next, we aim at constructing formulas that check whether a given team is a grid, pre-tableau, or a tableau, respectively.

First, to check that every location $(i,j) \in \{1,\ldots,N\}^2$ of the grid occurs as $\ell(w)$ of some $w \in T$, we quantify over all corresponding pairs $(\Delta',\Delta'') \in \wp(\Delta_{k-1})^2$. To cover all these sets of types we can quantify, for instance, over the images of all points of $T_{\mathfrak{s}_k}$. However, as subteam quantifiers $\exists^{\subseteq}, \exists^{1}, \forall^{\subseteq}, \forall^{1}$ cannot pick *two* subteams of the pair from the same scope, we enforce a k-canonical copy $\mathfrak{s}'_k$ of $\mathfrak{s}_k$ in the spirit of Theorem 4.24:

$$\mathsf{canon}' := \rho^{k}_{0}(\mathfrak{s}_0) \wedge \bigwedge_{m=1}^{k} \rho^{k-m}_{m}(\mathfrak{s}_{m-1},\mathfrak{s}_m) \wedge \rho^{0}_{k}(\mathfrak{s}_{k-1},\mathfrak{s}'_k)$$

Then $\mathfrak{s}_k$ is used for the first component and $\mathfrak{s}'_k$ is used for the second.

Claim (f). *If $\mathfrak{s}_0,\ldots,\mathfrak{s}_k,\mathfrak{s}'_k$ are disjoint scopes in $\mathcal{K}$, then $(\mathcal{K},T) \vDash \mathsf{canon}'$ if and only if $(\mathcal{K},T)$ is a k-staircase and $T_{\mathfrak{s}'_k}$ is k-canonical. Moreover, $\mathsf{canon}' \wedge \mathsf{scopes}_k(\{\mathfrak{s}_0,\ldots,\mathfrak{s}_k,\mathfrak{s}'_k\}) \wedge \Box^{k+1}\bot$ is satisfiable, but is only satisfied by k-staircases $(\mathcal{K},T)$ in which both $T_{\mathfrak{s}_k}$ and $T_{\mathfrak{s}'_k}$ are k-canonical. Furthermore, both formulas are constructible in space $\mathcal{O}(\log(|\Phi|+k))$.*

Proof of claim. Proven similarly to Theorem 4.24 and 4.26. ◁

The next formula checks whether a given team is a grid. More precisely, the subformula ψ_{pair} compares the t-component of the selected location in T_α to the image of the world quantified in $\mathfrak{s}_k$, and its p-component to $\mathfrak{s}'_k$, respectively. That every world satisfies exactly one element of Ξ is guaranteed by ψ_{grid} as well.

$$\psi_{\text{grid}}(\alpha) := \Big(\alpha \hookrightarrow \bigvee_{e\in\Xi} e \wedge \bigwedge_{\substack{e'\in\Xi\\ e'\neq e}} \neg e'\Big) \wedge \forall^{1}_{\mathfrak{s}_k} \forall^{1}_{\mathfrak{s}'_k} \exists^{1}_{\alpha} \psi_{\text{pair}}(\alpha)$$

$$\psi_{\text{pair}}(\alpha) := \Box \left[\left(|^{\alpha}_{t}\, \chi^{*}_{k-1}(\mathfrak{s}_k,\alpha)\right) \wedge \left(|^{\alpha}_{p}\, \chi^{*}_{k-1}(\mathfrak{s}'_k,\alpha)\right)\right]$$

In the following and all subsequent claims, we always assume that T is a team in a Kripke structure $\mathcal{K}$ such that $(\mathcal{K},T)$ satisfies $\mathsf{canon}' \wedge \Box^{k+1}\bot$. Moreover, all stated scopes are always assumed pairwise disjoint in $\mathcal{K}$ (as we can enforce this later in the reduction using the formula scopes_k).

Claim (g). $T \vDash \psi_{grid}(\alpha)$ *if and only if* T_α *is a grid.*

Proof of claim. Clearly $T \vDash \alpha \hookrightarrow \bigvee_{e \in \Xi} e \wedge \bigwedge_{e' \in \Xi, e' \neq e} \neg e'$ if and only if every world $w \in T_\alpha$ satisfies exactly one element of Ξ. For the proof that all locations appear in T_α we use the following chain of equivalences:

$$\forall (i,j) \in \{1, \ldots, N\}^2 \colon \exists w \in T_\alpha \colon \ell(w) = (i,j)$$
$$\Leftrightarrow \forall \Delta', \Delta'' \subseteq \Delta_{k-1} \colon \exists w \in T_\alpha : [\![(Rw)_t]\!]_{k-1} = \Delta' \text{ and } [\![(Rw)_p]\!]_{k-1} = \Delta'' \qquad \text{(def.)}$$

By k-canonicity of $\mathfrak{s}_k, \mathfrak{s}'_k$ due to Claim (f):

$$\Leftrightarrow \forall v \in T_{\mathfrak{s}_k}, v' \in T_{\mathfrak{s}'_k} \colon \exists w \in T_\alpha \colon [\![(Rw)_t]\!]_{k-1} = [\![Rv]\!]_{k-1} \text{ and } [\![(Rw)_p]\!]_{k-1} = [\![Rv']\!]_{k-1}$$

By Theorem 4.19:

$$\Leftrightarrow \forall v \in T_{\mathfrak{s}_k}, v' \in T_{\mathfrak{s}'_k} \colon \exists w \in T_\alpha \colon RT^{\alpha,\mathfrak{s}_k,\mathfrak{s}'_k}_{(Rw)_t,Rv,Rv'} \vDash \chi^*_{k-1}(\mathfrak{s}_k,\alpha)$$
$$\text{and } RT^{\alpha,\mathfrak{s}_k,\mathfrak{s}'_k}_{(Rw)_p,Rv,Rv'} \vDash \chi^*_{k-1}(\mathfrak{s}'_k,\alpha)$$
$$\Leftrightarrow \forall v \in T_{\mathfrak{s}_k}, v' \in T_{\mathfrak{s}'_k} \colon \exists w \in T_\alpha : \left(RT^{\alpha,\mathfrak{s}_k,\mathfrak{s}'_k}_{Rw,Rv,Rv'}\right)^\alpha_{RT_t} \vDash \chi^*_{k-1}(\mathfrak{s}_k,\alpha)$$
$$\text{and } \left(RT^{\alpha,\mathfrak{s}_k,\mathfrak{s}'_k}_{Rw,Rv,Rv'}\right)^\alpha_{RT_p} \vDash \chi^*_{k-1}(\mathfrak{s}'_k,\alpha)$$
$$\Leftrightarrow \forall v \in T_{\mathfrak{s}_k}, v' \in T_{\mathfrak{s}'_k} \colon \exists w \in T_\alpha \colon RT^{\alpha,\mathfrak{s}_k,\mathfrak{s}'_k}_{Rw,Rv,Rv'} \vDash |^\alpha_t \chi^*_{k-1}(\mathfrak{s}_k,\alpha) \wedge |^\alpha_p \chi^*_{k-1}(\mathfrak{s}'_k,\alpha)$$

By Proposition 4.17 (6):

$$\Leftrightarrow \forall v \in T_{\mathfrak{s}_k}, v' \in T_{\mathfrak{s}'_k} \colon \exists w \in T_\alpha \colon T^{\alpha,\mathfrak{s}_k,\mathfrak{s}'_k}_{w,v,v'} \vDash \Box\left(|^\alpha_t \chi^*_{k-1}(\mathfrak{s}_k,\alpha) \wedge |^\alpha_p \chi^*_{k-1}(\mathfrak{s}'_k,\alpha)\right)$$

By Proposition 4.18:

$$\Leftrightarrow T \vDash \forall^1_{\mathfrak{s}_k} \forall^1_{\mathfrak{s}'_k} \exists^1_\alpha \Box(|^\alpha_t \chi^*_{k-1}(\mathfrak{s}_k,\alpha) \wedge |^\alpha_p \chi^*_{k-1}(\mathfrak{s}'_k,\alpha))$$
$$\Leftrightarrow T \vDash \forall^1_{\mathfrak{s}_k} \forall^1_{\mathfrak{s}'_k} \exists^1_\alpha \psi_{\mathrm{pair}}(\alpha) \qquad \triangleleft$$

With slight modifications it is straightforward to define pre-tableaus:

$$\psi_{\text{pre-tableau}}(\alpha) := \psi_{\text{grid}}(\alpha) \wedge \forall^1_{\mathfrak{s}_k} \forall^1_{\mathfrak{s}'_k} \bigwedge_{e \in \Xi} \exists^1_\alpha \left(\psi_{\text{pair}}(\alpha) \wedge (\alpha \hookrightarrow e)\right)$$

Claim (h). $T \vDash \psi_{\textit{pre-tableau}}(\alpha)$ *if and only if* T_α *is a pre-tableau.*

Proof of claim. Proven similarly to Claim (g). $\triangleleft$

The other special case of a grid, that is, a *tableau*, requires a more elaborate approach to define in $\mathsf{ML}(\sim)$. The difference to a grid or pre-tableau is that we have to quantify over all *pairs* (w, w') of points in T, and check that they agree on Ξ whenever $\ell(w) = \ell(w')$. However, as mentioned before, while $\forall^1$ can quantify over all points in a team, it cannot quantify over pairs.

As a workaround, we consider not only a tableau T_α, but also a *second* tableau that acts as a copy of T_α. Formally, for grids T_α, T_β, let $T_\alpha \approx T_\beta$ mean that for all pairs $(w, w') \in T_\alpha \times T_\beta$ it holds that $\ell(w) = \ell(w')$ implies $c(w) = c(w')$. As $\approx$ is symmetric and transitive, $T_\alpha \approx T_\beta$ in fact implies $T_\alpha \approx T_\alpha$ and $T_\beta \approx T_\beta$, and hence that T_α and T_β are both tableaus and in fact $C_{T_\alpha} = C_{T_\beta}$. The following formula defines this, where γ_0 is a new scope that contains the "copy" of T_α.

$$\psi_{\text{tableau}}(\alpha) := \psi_{\text{grid}}(\alpha) \land \exists^{\subseteq}_{\gamma_0} \psi_{\text{grid}}(\gamma_0) \land \psi_{\approx}(\alpha, \gamma_0)$$

$$\psi_{\approx}(\alpha, \beta) := \forall^1_\alpha \forall^1_\beta \left((\psi^{t}_{\equiv}(\alpha, \beta) \land \psi^{p}_{\equiv}(\alpha, \beta)) \rightharpoonup \bigwedge_{e \in \Xi} ((\alpha \hookrightarrow e) \leftrightarrow (\beta \hookrightarrow e) \right)$$

In the above formula, we use the scopes $\gamma_0, \gamma_1, \gamma_2, \ldots$ as "auxiliary pre-tableaus". Later, we will also use them as domains to quantify extra locations or rows from. (The index of γ_i is incremented whenever necessary to avoid quantifying from the same scope twice.) For this reason, from now on we always assume, for sufficiently large i, that T_{γ_i} is a pre-tableau. This can be later enforced in the reduction with $\psi_{\text{pre-tableau}}(\gamma_i)$.

Claim (i). (1) $T \vDash \psi_{\textit{tableau}}(\alpha)$ *if and only if* T_α *is a tableau.*

(2) *For grids* T_α, T_β, *it holds* $T \vDash \psi_{\approx}(\alpha, \beta)$ *if and only if* $T_\alpha \approx T_\beta$.

Proof of claim. (2) follows straightforwardly from Claim (e), so let us consider (1). As ψ_{tableau} entails ψ_{grid}, we can assume that T_α is a grid.

Suppose that the formula is true. Then there exists $S \subseteq T_{\gamma_0}$ such that $T^{\alpha}_{S} \vDash \psi_{\text{grid}}(\gamma_0)$. By Claim (g), then S is a grid as well. Moreover, $T_\alpha \approx S$ by (2). As argued above, this implies that T_α (and S) is a tableau.

For the other direction, suppose that T_α is a tableau. Then it defines a function C_{T_α}. Since T_{γ_0} is a pre-tableau, we can pick a subteam S of it that contains for each $(i, j) \in \{1, \ldots, N\}^2$ exactly those worlds w with $\ell(w) = (i, j)$ such that $c(w) = C_{T_\alpha}(i, j)$. Then $T_\alpha \approx S$, and ψ_{tableau} is true, with the quantifier $\exists^{\subseteq}_{\gamma_0}$ witnessed by S. ◁

4.6.4 From tableaus to runs

To ascertain that a tableau contains a run of M, we have to check whether each row indeed is a configuration of M—in other words, exactly one cell of each row contains a pair $(q, a) \in Q \times \Gamma$—and whether consecutive configurations obey the transition relation δ of M.

For this, in the spirit of Cook's theorem [18] it suffices to consider all *legal windows* in the grid, i.e., cells that are adjacent as follows, where $e_1, \ldots, e_6 \in \Gamma \cup (Q \times \Gamma)$:

e_1	e_2	e_3
e_4	e_5	e_6

If, say, $(q, a, q', a', R) \in \delta$—$M$ switches to state q' from q, replacing a on the tape by a', and moves to the right—then for all $b, b' \in \Gamma$, the windows obtained by setting $e_1 = e_4 = b$, $e_2 = (q, a)$, $e_5 = a'$, $e_3 = b'$, $e_6 = (q', b')$ are legal. Using this scheme, δ is completely represented by a constant finite set $\mathsf{win} \subseteq \Xi^6$ of tuples $(e_1, \ldots, e_6)$ that represent the allowed windows in a run of M.

Let us next explain how adjacency of cells is expressed. Suppose that two points $w \in T_\alpha$ and $v \in T_\beta$ are given. That v is the immediate ($\mathfrak{t}$- or $\mathfrak{p}$-)successor of w then means that no element of the order exists between them. Simultaneously, w and v have to agree on the other component of their location, which is expressed by the first conjunct below. If $\mathfrak{q} \in \{\mathfrak{t}, \mathfrak{p}\}$ and $\overline{\mathfrak{q}} \in \{\mathfrak{t}, \mathfrak{p}\} \setminus \{\mathfrak{q}\}$, we define:

$$\psi^{\mathfrak{q}}_{\mathrm{succ}}(\alpha, \beta) := \psi^{\overline{\mathfrak{q}}}_{\equiv}(\alpha, \beta) \land \psi^{\mathfrak{q}}_{\prec}(\alpha, \beta) \land {\sim}\exists^{1}_{\gamma_0} \left(\psi^{\mathfrak{q}}_{\prec}(\alpha, \gamma_0) \land \psi^{\mathfrak{q}}_{\prec}(\gamma_0, \beta)\right)$$

Claim (j). *If $w \in T_\alpha$ and $v \in T_\beta$, then:*

$$T^{\alpha,\beta}_{w,v} \vDash \psi^{\mathfrak{t}}_{\mathrm{succ}}(\alpha, \beta) \Leftrightarrow \exists i, j \in \{1, \ldots, N\}\colon \ell(w) = (i, j) \textit{ and } \ell(v) = (i+1, j)$$
$$T^{\alpha,\beta}_{w,v} \vDash \psi^{\mathfrak{p}}_{\mathrm{succ}}(\alpha, \beta) \Leftrightarrow \exists i, j \in \{1, \ldots, N\}\colon \ell(w) = (i, j) \textit{ and } \ell(v) = (i, j+1)$$

Proof of claim. Let us consider only $\mathfrak{q} = \mathfrak{t}$, as the case $\mathfrak{q} = \mathfrak{p}$ is proven analogously. Assume that the formula $\psi^{\mathfrak{t}}_{\mathrm{succ}}(\alpha, \beta)$ is true in $T^{\alpha,\beta}_{w,v}$. By Claim (e), $\psi^{\mathfrak{p}}_{\equiv}$ holds if and only if there is a unique j such that $\ell(w) = (i, j)$ and $\ell(v) = (i', j)$, for some i, i'; in other words, if w and v agree on their $\mathfrak{p}$-component.

Next, consider the sets $\Delta_w := [\![(Rw)_{\mathfrak{t}}]\!]_{k-1}$ and $\Delta_v := [\![(Rv)_{\mathfrak{t}}]\!]_{k-1}$ which correspond to the $\mathfrak{t}$-components of $\ell(w)$ and $\ell(v)$. Suppose that Δ_w is the i-th element of $\prec^*_{k-1}$. By $\psi^{\mathfrak{t}}_{\prec}$ and Claim (e), then clearly Δ_v is the i'-th element for some $i' > i$.

Suppose for the sake of contradiction that also $i' > i+1$, and let then instead $\Delta' \subseteq \Delta_{k-1}$ be the $(i+1)$-th element of $\prec^*_{k-1}$. As T_{γ_0} is a pre-tableau, it contains a world z such that $\ell(z) = (i+1, j)$. But then $\psi^{\mathfrak{q}}_{\prec}(\alpha, \gamma_0) \land \psi^{\mathfrak{q}}_{\prec}(\gamma_0, \beta)$ is true in $T^{\alpha,\beta,\gamma_0}_{w,v,z}$, contradiction to $\psi^{\mathfrak{t}}_{\mathrm{succ}}$. Consequently, $i' = i + 1$. The direction from right to left is shown similarly. ◁

To check all windows in the tableau T_α, we need to simultaneously quantify elements from *six* tableaus $T_{\gamma_1}, \ldots, T_{\gamma_6}$ that are copies of T_α. For this purpose, we define the following abbreviation:

$$\exists^{\approx\alpha}_{\gamma_i} \varphi := \exists^{\subseteq}_{\gamma_i} (\psi_{\mathrm{grid}}(\gamma_i) \land \psi_{\approx}(\alpha, \gamma_i) \land \varphi)$$

Intuitively, under the premise that T_{γ_i} is a pre-tableau and T_α is a tableau, it "copies" the tableau T_α into T_{γ_i} by shrinking T_{γ_i} accordingly. This is proven analogously to Claim (i). The next formula states that the picked points are arranged as in the picture below:

$$\psi_{\mathrm{window}}(\gamma_1, \ldots, \gamma_6) := \psi^{\mathfrak{t}}_{\mathrm{succ}}(\gamma_1, \gamma_4) \land \psi^{\mathfrak{t}}_{\mathrm{succ}}(\gamma_2, \gamma_5) \land \psi^{\mathfrak{t}}_{\mathrm{succ}}(\gamma_3, \gamma_6) \land \psi^{\mathfrak{p}}_{\mathrm{succ}}(\gamma_1, \gamma_2) \land \psi^{\mathfrak{p}}_{\mathrm{succ}}(\gamma_2, \gamma_3)$$

T_{γ_1}	T_{γ_2}	T_{γ_3}
T_{γ_4}	T_{γ_5}	T_{γ_6}

The formula defining legal tableaus follows, for which we define subformulas $\theta_1, \theta_2, \theta_3$.

$$\psi_{\text{legal}}(\alpha) := \psi_{\text{tableau}}(\alpha) \land \exists^{\approx\alpha}_{\gamma_1} \cdots \exists^{\approx\alpha}_{\gamma_6} \, (\theta_1 \land \theta_2 \land \theta_3)$$

With θ_1, we check that the head of M is over at most one cell:

$$\theta_1 := \forall^1_{\gamma_1} \forall^1_{\gamma_2} \Big(\psi^t_{\equiv}(\gamma_1, \gamma_2) \land \psi^p_{\prec}(\gamma_1, \gamma_2)\Big) \twoheadrightarrow \bigwedge_{(q_1,a_1),(q_2,a_2) \in Q \times \Gamma} {\sim}\big((\gamma_1 \hookrightarrow c^{-1}(q_1, a_1)) \land (\gamma_2 \hookrightarrow c^{-1}(q_2, a_2))\big)\Big)$$

With θ_2, we check that the head is on at least one cell. For this, $\forall^1_{\gamma_1}$ fixes a row and $\exists^1_{\gamma_2} \psi^t_{\equiv}(\gamma_1, \gamma_2)$ searches that particular row for a state of M:

$$\theta_2 := \forall^1_{\gamma_1} \exists^1_{\gamma_2} \; \psi^t_{\equiv}(\gamma_1, \gamma_2) \land \bigovee_{(q,a) \in Q \times \Gamma} (\gamma_2 \hookrightarrow c^{-1}(q, a))$$

Finally, θ_3 states that every window must obey the transition relation:

$$\theta_3 := \forall^1_{\gamma_1} \cdots \forall^1_{\gamma_6} \Big(\psi_{\text{window}}(\gamma_1, \ldots, \gamma_6) \twoheadrightarrow \bigovee_{(e_1,\ldots,e_6) \in \text{win}} \bigwedge_{i=1}^{6} (\gamma_i \hookrightarrow e_i) \Big)$$

Claim (k). $T \vDash \psi_{legal}(\alpha)$ *iff* T_α *is a legal tableau, i.e., iff* C_{T_α} *exists and is a run of* M.

Proof of claim. Suppose that the formula holds. We show that T_α is a legal tableau; the other direction is proven similarly.

Due to Claim (i), there are tableaus $S_1 \subseteq T_{\gamma_1}$, ..., $S_6 \subseteq T_{\gamma_6}$ that are copies of T_α such that $\theta_1 \land \theta_2 \land \theta_3$ holds in $T^{\gamma_1,\ldots,\gamma_6}_{S_1,\ldots,S_6}$.

Due to Claim (e), the subformula θ_1 ensures the following: For all $w \in S_1, w' \in S_2$, $\ell(w) = (i, j)$, $\ell(w') = (i', j')$, if $i = i'$ and $j < j'$ hold, then it is not the case that both $c(w) = (q, a)$ and $c(w') = (q', a')$ for any state symbols $q, q' \in Q$. Since $C_{S_1} = C_{S_2} = C_{T_\alpha}$, this is precisely the case if each row of C_{T_α} contains at most one state symbol.

Conversely, again by Claim (e), the subformula θ_2 states that for every cell $w \in S_1$ there is some cell $w' \in S_2$ in the same row (possibly w itself) that carries a state symbol: in other words, every row of C_{T_α} contains at least one state symbol.

Finally, θ_3 relies on Claim (j) and states for every choice of singletons $w_1, \ldots, w_6$ in $S_1, \ldots, S_6$, assuming that they are arranged as a window, that there exists a tuple $(e_1, \ldots, e_6) \in \text{win}$ such that $w_i \in S_i$ satisfies $c(w_i) = e_i$. As we showed that C_{T_α} contains in each row a configuration of M, this implies that C_{T_α} exists and is a run of M. $\lhd$

4.6.5 From runs to a computation

To encode the initial configuration on input $x = x_1 \cdots x_n$ in a tableau, we access the first n cells of the first row and assign the respective letter of x, also we assign the initial state to the first cell. Finally, we assign $\flat$ to all other cells in that row. To find the first cell, for each $q \in \{t, p\}$, we check whether the q-component of a point in T_α is minimal:

$$\psi^q_{\min}(\alpha) := {\sim}\exists^1_{\gamma_0} \psi^q_{\prec}(\gamma_0, \alpha)$$

This enables us to fix the first row of the configuration:

$$\begin{aligned}\psi_{\text{input}}(\alpha) := \exists^{\approx\alpha}_{\gamma_1} \cdots \exists^{\approx\alpha}_{\gamma_{n+1}} \exists^1_{\gamma_1} \cdots \exists^1_{\gamma_n} \psi^t_{\min}(\gamma_1) \wedge \psi^p_{\min}(\gamma_1) \wedge \left(\gamma_1 \hookrightarrow c^{-1}(q_0, x_1)\right) \\ \bigwedge_{i=2}^{n} \psi^p_{\text{succ}}(\gamma_{i-1}, \gamma_i) \wedge \left(\gamma_i \hookrightarrow c^{-1}(x_i)\right) \\ \wedge \forall^1_{\gamma_{n+1}} \Big(\left(\psi^t_{\equiv}(\gamma_n, \gamma_{n+1})\right) \wedge \psi^p_{\prec}(\gamma_n, \gamma_{n+1})\big) \twoheadrightarrow \left(\gamma_{n+1} \hookrightarrow c^{-1}(\flat)\right)\Big)\end{aligned}$$

Claim (1). *Let T_α be a tableau. Then $T \vDash \psi_{input}(\alpha)$ if and only if*

(1) $C_{T_\alpha}(1, 1) = (q_0, x_1)$,

(2) $C_{T_\alpha}(1, i) = x_i$ *for* $2 \leqslant i \leqslant n$,

(3) $C_{T_\alpha}(1, i) = \flat$ *for* $n < i \leqslant N$.

Proof of claim. Suppose that the formula holds. After processing $\exists^{\approx\alpha}_{\gamma_1} \cdots \exists^{\approx\alpha}_{\gamma_{n+1}}$, for all $m \in \{1, \ldots, n+1\}$ the team T_{γ_m} is a tableau such that $C_{T_{\gamma_m}} = C_{T_\alpha}$. (Obviously this requires these teams to be pre-tableaus beforehand.) For this reason, we can freely replace $C_{T_\alpha}(i, j)$ with $C_{T_{\gamma_m}}(i, j)$ when proving the properties (1)–(3).

In the second line of the formula, we make sure that $c(w) = (q_0, x_1)$ holds for least one point $w \in C_{T_{\gamma_1}}$ of location $\ell(w) = (1, 1)$. That $\ell(w) = (1, 1)$ holds follows from Claim (e), $\psi^q_{\min}$, and the assumption that T_{γ_0} is a pre-tableau (which it still is after processing the quantifiers $\exists^{\approx\alpha}_{\gamma_1} \cdots \exists^{\approx\alpha}_{\gamma_{n+1}}$). In particular, $C_{T_{\gamma_1}}(1, 1) = (q_0, x_1)$.

The third line works similarly: for $2 \leqslant i \leqslant n$, it assigns x_i to $C_{T_{\gamma_i}}(1, i)$ and hence to $C_{T_\alpha}(1, i)$. Finally, the last two lines state that every other location $(1, j')$ with $j' > n$ contains $\flat$. The other direction is again similar. ◁

Until now, we ignored the fact that M (polynomially often) alternates. To simulate this, we alternatingly quantify polynomially many tableaus, each containing a part of the computation of M. Each of these tableaus possesses a *tail configuration*, which is the configuration where M either accepts, rejects, or alternates. Formally, a number $i \in \{1, \ldots, N\}$ is a *tail index* of a run C if, for some j,

(1) $C(i, j)$ has an accepting or rejecting state,

(2) or $C(i, j)$ has an existential state and there are $i' < i$ and j' with a universal state in $C(i', j')$,

(3) or $C(i, j)$ has a universal state and there are $i' < i$ and j' with an existential state in $C(i', j')$.

The least such i is called *first tail index*, and the corresponding configuration is the *first tail configuration*. The idea is that we can split the computation of M into multiple tableaus if any tableau (except the initial one) contains a run that continues from the previous tableau's first tail configuration.

We formalize the above as follows. Assume that T_α is a tableau, and that T_β marks a single row i by being a singleton $\{w\}$ with $\ell(w) = (i, j)$ for some j. Then the formula $\psi_{\text{tail}}(\alpha, \beta)$ below will be true if and only if the i-th row of C_{T_α} is a tail configuration. With the subformula

$$Q'\text{-state}(\beta) := \bigovee_{(q,a)\in Q'\times\Gamma} (\beta \hookrightarrow c^{-1}(q, a)),$$

we check if a given singleton $T_\beta = \{w\}$ encodes an accepting, rejecting, existential, universal, or arbitrary state by setting Q' to Q_{acc}, Q_{rej}, $Q_\exists$, $Q_\forall$ or Q, respectively.

$$\begin{aligned}\psi_{\text{tail}}(\alpha, \beta) \;:=\; & \exists^{\approx\alpha}_{\gamma_0}\, \exists^1_\alpha \Big[\psi^t_{\equiv}(\alpha, \beta) \wedge Q\text{-state}(\alpha) \wedge \Big[Q_{\text{acc}}\text{-state}(\alpha) \ovee Q_{\text{rej}}\text{-state}(\alpha) \\ & \ovee \exists^1_{\gamma_0}\Big(\psi^t_{\prec}(\gamma_0, \alpha) \wedge \big(Q_\exists\text{-state}(\alpha) \wedge Q_\forall\text{-state}(\gamma_0)\big) \\ & \ovee \big(Q_\forall\text{-state}(\alpha) \wedge Q_\exists\text{-state}(\gamma_0)\big)\Big)\Big]\Big] \\ \psi_{\text{first-tail}}(\alpha, \beta) \;:=\; & \psi_{\text{tail}}(\alpha, \beta) \wedge {\sim}\exists^1_{\gamma_1}\Big(\psi^t_{\prec}(\gamma_1, \beta) \wedge \psi_{\text{tail}}(\alpha, \gamma_1)\Big)\end{aligned}$$

Claim (m). *Suppose that T_α is a tableau, $T_\beta = \{w\}$, and $\ell(w) = (i, j)$. Then $T \vDash \psi_{tail}(\alpha, \beta)$ if and only if i is a tail index of C_{T_α}. Moreover, $T \vDash \psi_{first\text{-}tail}(\alpha, \beta)$ if and only if i is the first tail index of C_{T_α}.*

Proof of claim. Since T_{γ_1} is a pre-tableau and hence contains all locations in rows $i' < i$, it is easy to see that the proof for $\psi_{\text{first-tail}}$ boils down to that of ψ_{tail}. Consequently, let us consider ψ_{tail}. First, due to $\exists^{\approx\alpha}_{\gamma_0}$, we can assume that T_{γ_0} is a tableau that is a copy of T_α, i.e., $C_{T_\alpha} = C_{T_{\gamma_0}}$. Here, it is required for the inner quantification in the definition of a tail index.

The first line of the formula reduces T_α to a singleton that is (due to $\psi^t_{\equiv}$) in row i. Furthermore, it carries a state q of M due to Q-state(α). The further examination of this state will determine if i is a tail index. Now, q is exactly one of accepting, rejecting, existential, or universal. If $q \in Q_{\text{acc}} \cup Q_{\text{rej}}$, then i is a tail index by definition.

Otherwise we quantify over the states q' of all (copies of) earlier rows in T_α, using $\exists^1_{\gamma_0}\psi^t_{\prec}(\gamma_0, \alpha)$, and search for a universal state if q is existential and vice versa, which as well, if it exists, proves by definition that i is a tail index. ◁

Given a run C of M that has a tail configuration, we say that C *accepts* if the state q in its first tail configuration is in Q_{acc}, C *rejects* if that q is in Q_{rej}, and C *alternates* otherwise. That a run of the form C_{T_α} accepts or rejects is expressed by

$$\psi_{\text{acc}}(\alpha) := \exists^{\approx\alpha}_{\gamma_2}\, \exists^1_{\gamma_2}\, (Q_{\text{acc}}\text{-state}(\gamma_2) \wedge \psi_{\text{first-tail}}(\alpha, \gamma_2)),$$

$$\psi_{\mathrm{rej}}(\alpha) := \exists^{\approx\alpha}_{\gamma_2}\, \exists^{1}_{\gamma_2}\, (Q_{\mathrm{rej}}\text{-}\mathsf{state}(\gamma_2) \wedge \psi_{\text{first-tail}}(\alpha, \gamma_2)).$$

In this formula, first the tableau T_α is copied to T_{γ_2} to extract with $\exists^1_{\gamma_2}$ the world carrying an accepting/rejecting state, while $\psi_{\text{first-tail}}(\alpha, \gamma_2)$ ensures that no alternation or rejecting/accepting state occurs at some earlier point in C_{T_α}.

If the first tail configuration of the run contains an alternation, and if the run was existentially quantified, then it should be continued in a universally quantified tableau, and vice versa. The following formula expresses, given two tableaus T_α, T_β, that C_{T_β} is a *continuation* of C_{T_α}, i.e., that the first configuration of C_{T_β} equals the first tail configuration of C_{T_α}. In other words, if i is the first tail index of C_{T_α}, then $C_{T_\alpha}(i, j) = C_{T_\beta}(1, j)$ for all $j \in \{1, \ldots, N\}$.

$$\psi_{\mathrm{cont}}(\alpha, \beta) := \exists^1_{\gamma_2}\Bigg[\psi_{\text{first-tail}}(\alpha, \gamma_2) \ \wedge\ \forall^1_\alpha \forall^1_\beta \\ \Big[\Big(\psi^{\mathfrak{t}}_{\min}(\beta) \wedge \psi^{\mathfrak{t}}_{\equiv}(\alpha, \gamma_2) \wedge \psi^{\mathfrak{p}}_{\equiv}(\alpha, \beta)\Big) \multimap \Big(\bigovoid_{e \in \Xi} (\alpha \vee \beta) \hookrightarrow e\Big)\Big]\Bigg]$$

The above formula first obtains the first tail index i of C_{T_α} and stores it in a singleton $y \in T_{\gamma_2}$. Then for all worlds $w \in T_\alpha$ and $v \in T_\beta$, where v is $\mathfrak{t}$-minimal (i.e., in the first row) and w is in the same row as y, and which additionally agree on their $\mathfrak{p}$-component, the third line states that w and v agree on Ξ. Altogether, the i-th row of C_{T_α} and the first row of C_{T_β} then have to coincide.

M performs at most $r(n) - 1$ alternations for some polynomial r. Then we require $r = r(n)$ tableaus, which we call $\alpha_1, \ldots, \alpha_r$. In the following, the formula $\psi_{\mathrm{run},i}$ describes the behaviour of the i-th run, i.e., the part of the computation after $i - 1$ alternations. W.l.o.g. r is even and $q_0 \in Q_\exists$. We may then define the final run by

$$\psi_{\mathrm{run},r} := \forall^{\subseteq}_{\alpha_r}\Big[\big(\psi_{\mathrm{legal}}(\alpha_r) \wedge \psi_{\mathrm{cont}}(\alpha_{r-1}, \alpha_r)\big) \multimap \big({\sim}\psi_{\mathrm{rej}}(\alpha_r) \wedge \psi_{\mathrm{acc}}(\alpha_r)\big)\Big].$$

For $1 < i < r$ and i even, let

$$\psi_{\mathrm{run},i} := \forall^{\subseteq}_{\alpha_i}\Big[\big(\psi_{\mathrm{legal}}(\alpha_i) \wedge \psi_{\mathrm{cont}}(\alpha_{i-1}, \alpha_i)\big) \multimap \big({\sim}\psi_{\mathrm{rej}}(\alpha_i) \wedge \big(\psi_{\mathrm{acc}}(\alpha_i) \ovoid \psi_{\mathrm{run},i+1}\big)\big)\Big]$$

and for $1 < i < r$ and i odd,

$$\psi_{\mathrm{run},i} := \exists^{\subseteq}_{\alpha_i}\Big[\psi_{\mathrm{legal}}(\alpha_i) \wedge \psi_{\mathrm{cont}}(\alpha_{i-1}, \alpha_i) \wedge {\sim}\psi_{\mathrm{rej}}(\alpha_i) \wedge \big(\psi_{\mathrm{acc}}(\alpha_i) \ovoid \psi_{\mathrm{run},i+1}\big)\Big].$$

Finally, the initial run is described by

$$\psi_{\mathrm{run},1} := \exists^{\subseteq}_{\alpha_1}\Big(\psi_{\mathrm{legal}}(\alpha_1) \wedge \psi_{\mathrm{input}}(\alpha_1) \wedge {\sim}\psi_{\mathrm{rej}}(\alpha_1) \wedge \big(\psi_{\mathrm{acc}}(\alpha_1) \ovoid \psi_{\mathrm{run},2}\big)\Big).$$

We are now in the position to state the full reduction. Let us gather all relevant scopes in the set $\Psi \subseteq \mathsf{Prop}$:

$$\Psi := \{\mathfrak{s}_i \mid 0 \leqslant i \leqslant k\} \cup \{\mathfrak{s}'_k\} \cup \{\gamma_i \mid 0 \leqslant i \leqslant n+1\} \cup \{\alpha_i \mid 1 \leqslant i \leqslant r\}$$

The scopes that host pre-tableaus are

$$\Psi' := \{\gamma_i \mid 0 \leqslant i \leqslant n+1\} \cup \{\alpha_i \mid 1 \leqslant i \leqslant r\}.$$

W.l.o.g. $n \geqslant 5$, as $\gamma_1, \ldots, \gamma_6$ are always required in the construction. The reduction now maps x to the $\mathsf{ML}_k(\sim)$-formula

$$\varphi_x := \mathsf{canon}' \wedge \mathsf{scopes}_k(\Psi) \wedge \bigwedge_{p \in \Psi'} \psi_{\text{pre-tableau}}(p) \wedge \psi_{\text{run},1}.$$

It is straightforward to check that all of the above steps are computable in logspace from x and k, where k itself is either constant or a polynomial in $|x|$ and hence logspace-computable. By Lemma 4.25, φ_x is satisfiable if and only if $\varphi_x \wedge \Box^{k+1}\bot$ is satisfiable. For this reason, we conclude the reduction with the following proof.

Proof of Lemma 4.33. It remains to argue that $\varphi_x \wedge \Box^{k+1}\bot$ is satisfiable if and only if M accepts x. For the sake of simplicity, assume $r = 2$. The cases $r > 2$ are proven analogously.

"$\Rightarrow$": Suppose $(\mathcal{K}, T) \vDash \varphi_x \wedge \Box^{k+1}\bot$. Similarly as in Theorem 4.26, the $p \in \Psi$ are disjoint scopes due to $\mathsf{scopes}_k(\Psi)$. Moreover, by canon' and Claim (f), $(\mathcal{K}, T)$ is then a k-staircase in which $T_{\mathfrak{s}_k}$ and $T_{\mathfrak{s}'_k}$ both are k-canonical teams. Due to Claim (h) and the large conjunction in φ_x, $T_{\alpha_1}, T_{\alpha_2}, T_{\gamma_1}, \ldots, T_{\gamma_{n+1}}$ are then pre-tableaus.

As the formula $\psi_{\text{run},1}$ holds, by Claim (k) and Claim (l), T_{α_1} has a subteam S_1 that is a legal tableau and starts with M's initial configuration on x. In particular, C_{S_1} exists. Moreover, either ψ_{acc} holds (i.e., C_{S_1} and hence M is accepting) or $\psi_{\text{run},2}$ holds (i.e., if C_{S_1} alternates). Consider the latter case. Then for all legal tableaus $S_2 \subseteq T_{\alpha_2}$ such that C_{S_2} is a continuation of C_{S_1} it holds that C_{S_2} is accepting. However, as T_{α_2} is a pre-tableau, every run is of the form C_{S_2} for some $S_2 \subseteq T_{\alpha_2}$. Consequently, M accepts x.

"$\Leftarrow$": Suppose M accepts x. First of all, due to Claim (f), the formula $\mathsf{canon}' \wedge \mathsf{scopes}_k(\{\mathfrak{s}_0, \ldots, \mathfrak{s}_k, \mathfrak{s}'_k\}) \wedge \Box^{k+1}\bot$ has a model $(\mathcal{K}, T)$. Moreover, we can freely add a pre-tableau T_p for each $p \in \psi$ to satisfy the large conjunction in φ_x. By labeling the propositions in ψ correctly (as disjoint scopes), we ensure that $\mathsf{scopes}_k(\psi)$ holds as well.

It remains to demonstrate $T \vDash \psi_{\text{run},1}$. As M accepts x, there exists a run C_1 starting from M's initial configuration such that either C_1 accepts, or, for all runs C_2 continuing C_1, C_2 accepts.

Since T_{α_1} is a pre-tableau, it also contains a subteam S_1 such that S_1 is a legal tableau and $C_{S_1} = C_1$. We choose S_1 as witness for $\exists^{\subseteq}_{\alpha_1}$. If C_1 itself accepts, then $\psi_{\text{acc}}(\alpha_1)$ and hence $\psi_{\text{run},1}$ is satisfied. Otherwise we consider $\psi_{\text{run},2}$. Suppose that $S_2 \subseteq T_{\alpha_2}$ is picked as a subteam by $\forall^{\subseteq}_{\alpha_2}$. If it forms a legal tableau and C_{S_2} is a continuation of C_1, then C_2 must be accepting since M accepts x. But this implies that $\psi_{\text{acc}}(\alpha_2)$ is true for any such S_2. Consequently, $\psi_{\text{run},2}$ and hence $\psi_{\text{run},1}$ is true. $\square$

4.7 Hardness under strict semantics

Next, we further generalize the hardness results of the previous section and show that they also hold under strict semantics. Recall the strict connectives $\dot{\vee}$ and $\dot{\Diamond}$:

$$\begin{aligned}(\mathcal{K}, T) \vDash \psi \dot{\vee} \theta &\Leftrightarrow \exists S, U \subseteq T \text{ such that } T = S \cup U, S \cap U = \emptyset,\\ &\qquad (\mathcal{K}, S) \vDash \psi \text{ and } (\mathcal{K}, U) \vDash \theta,\\ (\mathcal{K}, T) \vDash \dot{\Diamond}\psi &\Leftrightarrow (\mathcal{K}, S) \vDash \psi \text{ for some strict successor team } S \text{ of } T,\end{aligned}$$

where a *strict successor team* of T is a successor team $S \subseteq RT$ for which there exists a surjective $f\colon T \to S$ satisfying $f(w) \in Rw$ for all $w \in T$.

In the lax disjunction the teams of the splitting may overlap, while in the strict disjunction they are disjoint. Likewise, a lax successor team may contain multiple successor of any $w \in T$, while in a strict successor team we pick exactly one successor for each $w \in T$.

We prove that our hardness results also hold in strict semantics. Let the logics $\mathsf{ML}(\sim, \dot{\vee}, \Box)$ and $\mathsf{ML}_k(\sim, \dot{\vee}, \Box)$ be defined like $\mathsf{ML}(\sim)$ and $\mathsf{ML}_k(\sim)$, but with $\dot{\vee}$ instead of $\vee$ and without $\Diamond$ and $\dot{\Diamond}$ (i.e., only using the modality $\Box$). Hardness for this fragment clearly implies hardness for strict semantics.

Theorem 4.34. *The satisfiability and validity problem of* $\mathsf{ML}(\sim, \dot{\vee}, \Box)$ *is* $\textsc{Tower}(\mathrm{poly})$*-hard. For* $k \geqslant 0$*, the satisfiability and validity problem of* $\mathsf{ML}_k(\sim, \dot{\vee}, \Box)$ *is* $\textsc{ATime-Alt}(\exp_{k+1}, \mathrm{poly})$*-hard.*

Proof. An analysis of the proof of Lemma 4.33 yields that the $\mathsf{ML}(\sim)$-formula φ_x produced in the reduction can be easily adapted to strict semantics. First, observe that $\Diamond$ occurs only in the subformula max_i, which is by Proposition 2.31 equivalent to

$$\top \dot{\vee} \left(\neg\Box^i \bot \wedge {\sim} \dot{\bigvee}_{p \in \Phi} (\neg\Box^i p \owedge \neg\Box^i \neg p) \right),$$

since $\Diamond\alpha \equiv \neg\Box\neg\alpha$, and since $\neg\Box^i p \owedge \neg\Box^i \neg p$ is a downward closed formula. A quick check reveals that Proposition 2.31 applies to all other instances of $\vee$ in φ_x as well, except of the occurrence in the second line of ζ_k^*. Here, the critical part of the correctness proof is the choice of the subteam U in Claim (c) of Lemma 4.30. In strict semantics, the only possibility becomes $U = O \setminus S$, for which the proof works identically. Finally, for the case $k = 0$, a similar check of the proof for $\mathsf{PL}(\sim)$ [56, Theorem 4.9] reveals that again every $\vee$ can be replaced by $\dot{\vee}$ due to Proposition 2.31. ☐

Note that the corresponding upper bound via the construction of a canonical model (Theorem 4.9) does not apply to strict semantics. As already mentioned in Chapter 2, the reason for this is the failure of Proposition 2.30, which roots in the fact that in strict semantics $\mathsf{ML}_k(\sim)$-formulas are not invariant under k-team-bisimulation in general.

4.8 Hardness on restricted frame classes

A *frame* is a pair $F = (W, R)$, where W is a non-empty set and $R \subseteq W \times W$. A natural restriction in the context of modal logic is to focus on Kripke structures over a specific subclass of frames, which is useful for instance for modeling belief or temporal systems. For an introduction to frame classes, consider, e.g., Fitting [33]. Prominent frame classes include

K: the class of all frames,

D: serial frames ($w \in W \Rightarrow Rw \neq \emptyset$),

T: reflexive frames ($w \in W \Rightarrow w \in Rw$),

K4: transitive frames ($u \in Rv \land v \in Rw \Rightarrow u \in Rw$),

D4: serial and transitive frames,

S4: reflexive and transitive frames.

In this section, we consider these classes from a complexity theoretic perspective, and show that the lower bounds of $\mathsf{ML}(\sim)$ are preserved when restricted to these classes. Given a frame class $\mathcal{F}$ and a fragment $\mathcal{L}$ of $\mathsf{ML}(\sim)$, let $\mathrm{SAT}(\mathcal{L}, \mathcal{F})$ denote the set of all $\mathcal{L}$-formulas that are satisfied in a model $(\mathcal{K}, T)$ where $\mathcal{K} = (F, V)$ and $F \in \mathcal{F}$. Define $\mathrm{VAL}(\mathcal{L}, \mathcal{F})$ analogously.

For example, $E\Box\bot \in \mathrm{SAT}(\mathsf{ML}(\sim), \mathsf{K})$, but $E\Box\bot \notin \mathrm{SAT}(\mathsf{ML}(\sim), \mathsf{D4})$.

Ladner's theorem [94, Theorem 3.1] implies that classical modal satisfiability and validity are PSpace-hard for any frame class between S4 and K, i.e., the complexity does not change for restricted frame classes. This includes all the above frame classes. We show an analog to Ladner's theorem for team semantics, in the sense that the complexity of $\mathsf{ML}(\sim)$ does not decrease for any frame class between S4 and K.

Theorem 4.35. *Let $\mathcal{F}$ be a frame class such that $\mathsf{S4} \subseteq \mathcal{F} \subseteq \mathsf{K}$. Then $\mathrm{SAT}(\mathsf{ML}(\sim), \mathcal{F})$ and $\mathrm{VAL}(\mathsf{ML}(\sim), \mathcal{F})$ are hard for* Tower(poly)*, and $\mathrm{SAT}(\mathsf{ML}_k(\sim), \mathcal{F})$ and $\mathrm{VAL}(\mathsf{ML}_k(\sim), \mathcal{F})$ are hard for* ATime-Alt$(\exp_{k+1}, \mathrm{poly})$, *for $k \geqslant 0$.*

The hardness is shown by a reduction from the case of unrestricted frames, i.e., $\mathrm{SAT}(\mathsf{ML}_k(\sim), \mathsf{K}) \leqslant_m^{\log} \mathrm{SAT}(\mathsf{ML}_k(\sim), \mathcal{F})$. The proof for $\mathsf{ML}(\sim)$ (i.e., unbounded modal depth) is similar. Also, the reduction for VAL is clear.

As a part of the reduction, we use a class of models that are "stratified" in the following sense. Given a formula $\varphi \in \mathsf{ML}_k^{\Phi}(\sim)$, we introduce new propositions $\ell_0, \ldots, \ell_k \notin \Phi$ that mark the layers of different height in a structure. For this to be a consistent labeling, we use propositional formulas $\lambda_i := \ell_i \land \bigwedge_{j \neq i} \neg \ell_j$ that exclude all but one ℓ_i from being true in each world. Given a $\Phi \cup \{\ell_0, \ldots, \ell_k\}$-structure $\mathcal{K} = (W, R, V)$, let $\mathcal{K}^\circ := (W, R^\circ, V)$ be the structure where only "good" edges are retained, i.e., between layers with numbers i

and $i+1$, respectively:

$$R^\circ = R \cap \bigcup_{i=0}^{k-1} (V(\ell_i) \times V(\ell_{i+1})).$$

On the formula side, we mirror this by changing the modalities such that all "bad" edges are ignored. For this, we inductively define the formula φ^i from φ as follows. The propositions and Boolean connectives are unchanged, i.e., $p^i := p$ for $p \in \Phi$, $(\psi \wedge \theta)^i := \psi^i \wedge \theta^i$, $({\sim}\psi)^i := {\sim}\psi^i$, $(\psi \vee \theta)^i := \psi^i \vee \theta^i$. For the modalities, let $(\Diamond\psi)^i := \Diamond(\lambda_{i+1} \wedge \psi^{i+1})$ and $(\Box\psi^i) := \Box(\lambda_{i+1} \hookrightarrow \psi^{i+1})$.

The next lemma now states that we can arbitrarily add "bad" edges as long as each world has a well-defined "layer".

Lemma 4.36. *Let $k \geqslant 0$. Let $\mathcal{K} = (W, R, V)$ be a structure such that $V(\ell_i) \cap V(\ell_j) = \emptyset$ for all $0 \leqslant i < j \leqslant k$.*

Then, for all $i \in \{0, \ldots, k\}$, teams $T \subseteq V(\ell_i)$, and formulas $\psi \in \mathrm{ML}_{k-i}({\sim})$, it holds that $(\mathcal{K}, T) \vDash \psi^i$ if and only if $(\mathcal{K}^\circ, T) \vDash \psi$.

Proof. By induction on ψ. Let $\mathcal{K}$, i, T be as above.

- Atomic propositions are clear, and the Boolean connectives and splitting follow easily from the induction hypothesis.

- Let $\psi = \Diamond\theta \in \mathrm{ML}_{k-i}({\sim})$. In this case, $k - i \geqslant 1$, so $k \geqslant i + 1$. Now suppose $(\mathcal{K}, T) \vDash \psi^i$, i.e., $(\mathcal{K}, S) \vDash \lambda_{i+1} \wedge \theta^{i+1}$ for some R-successor team S of T. Then observe that $S \subseteq V(\ell_{i+1})$, and $\theta \in \mathrm{ML}_{k-(i+1)}({\sim})$. Consequently, by the induction hypothesis, $(\mathcal{K}^\circ, S) \vDash \theta$. S is also an R°-successor team of T, since $(w, v) \in R$ is retained in R° for every $(w, v) \in V(\ell_i) \times V(\ell_{i+1})$, and we have $T \subseteq V(\ell_i)$ and $S \subseteq V(\ell_{i+1})$. This proves $(\mathcal{K}^\circ, T) \vDash \psi$.

  Conversely, if $(\mathcal{K}^\circ, T) \vDash \psi$, then $(\mathcal{K}^\circ, S) \vDash \theta$ for some R°-successor team S of T. However, any R°-successor team of T is a subset of $V(\ell_{i+1})$. This is because $i + 1 \leqslant k$, so $T \subseteq V(\ell_i)$ implies $T \cap V(\ell_j) = \emptyset$ for all $j \neq i$, which means that any outgoing R°-edge must lead into $V(\ell_{i+1})$. Consequently, we obtain both $(\mathcal{K}, S) \vDash \ell_{i+1}$ and by induction hypothesis $(\mathcal{K}, S) \vDash \theta^{i+1}$. This ultimately yields $(\mathcal{K}, T) \vDash \psi^i = \Diamond(\ell_{i+1} \wedge \theta^{i+1})$, since the R°-successor team S is trivially a R-successor team of T.

- Let $\psi = \Box\theta$. Then:

$$\begin{aligned}
(\mathcal{K}, T) \vDash \psi^i &\Leftrightarrow (\mathcal{K}, RT) \vDash (\ell_{i+1} \hookrightarrow \theta^{i+1}) && \text{(def. } \psi^i = (\Box\theta)^i\text{)}\\
&\Leftrightarrow (\mathcal{K}, RT \cap V(\ell_{i+1})) \vDash \theta^{i+1} && \text{(def. } \hookrightarrow\text{)}\\
&\Leftrightarrow (\mathcal{K}^\circ, RT \cap V(\ell_{i+1}) \vDash \theta && \text{(induction hypothesis)}\\
(\star) &\Leftrightarrow (\mathcal{K}^\circ, R^\circ T \vDash \theta\\
&\Leftrightarrow (\mathcal{K}^\circ, T) \vDash \psi
\end{aligned}$$

For $(\star)$, we show that $R^\circ T = RT \cap V(\ell_{i+1})$. Clearly $R^\circ T \subseteq RT$ and by a similar argument as before we have $R^\circ T \subseteq V(\ell_{i+1})$, which proves "$\subseteq$". Conversely, if $w \in RT \cap V(\ell_{i+1})$, then $(v, w) \in R$ for some $v \in T$. As $(v, w) \in V(\ell_i) \times V(\ell_{i+1})$, then $(v, w) \in R^\circ$, hence $w \in R^\circ T$ This shows "$\supseteq$". ☐

We proceed with the proof of the full theorem.

Proof of Theorem 4.35. The reduction is now

$$\varphi \mapsto \varphi' := \ell_0 \wedge \varphi^0 \wedge \bigwedge_{i=0}^{k} \Box^i \bigvee_{j=0}^{k} \lambda_j$$

We proceed with proving the correctness of the reduction, i.e., φ is satisfied in an arbitrary model if and only if φ' is satisfied in a reflexive and transitive model.

"$\Rightarrow$": First, assume that $\varphi \in \mathrm{ML}_k(\sim)$ is satisfiable by some model $(\mathcal{K}, T)$. We show that φ' has a reflexive and transitive model.

By Corollary 4.22, w.l.o.g. $(\mathcal{K}, T)$ is a forest of height k with the set of roots being T. W.l.o.g. the propositions $\ell_0, \ldots, \ell_k$ (which we can assume do not occur in φ) appear in $\mathcal{K}$ as follows: $V(\ell_i) = R^i T$, that is, $V(\ell_0) = T$, $V(\ell_1) = RT$ and so on. The sets $V(\ell_0), V(\ell_1), \ldots, V(\ell_k)$ are then pairwise disjoint since $\mathcal{K}$ is a forest.

This enables us to apply the lemma. Let R^* be the reflexive transitive closure of R and $\mathcal{K}^* := (W, R^*, V)$. We will show below $(R^*)^\circ = R$. Then by the lemma we obtain $(\mathcal{K}^*, T) \vDash \varphi^0$, and since we ensured $(\mathcal{K}^*, T) \vDash \ell_0 \wedge \bigwedge_{i=0}^{k} \Box^i \bigvee_{j=0}^{k} \lambda_j$ from the beginning, $(\mathcal{K}^*, T)$ is then a reflexive transitive model of φ'.

It remains to prove $(R^*)^\circ = R$. It is easy to see that $R = R^\circ \subseteq (R^*)^\circ$. For the other direction, suppose $(w, v) \in (R^*)^\circ$. By definition of $(R^*)^\circ$, there is i such that $w \in V(\ell_i)$, $v \in V(\ell_{i+1})$, and v is reachable from w by some R-path $(u_0, \ldots, u_n)$ where $w = u_0$ and $v = u_n$ (and possibly $n = 0$). But since $u_0 \in R^i T$, for all m it holds $u_m \in R^{i+m} T = V(\ell_{i+m})$. As $V(\ell_{i+n}) \cap V(\ell_{i+1}) = \emptyset$ for $n \neq 1$, we conclude $n = 1$, so $(w, v) \in R$.

"$\Leftarrow$": Suppose $\varphi' = \ell_0 \wedge \varphi^0 \wedge \bigwedge_{i=0}^{k} \Box^i \bigvee_{j=0}^{k} \lambda_j$ is satisfiable by some model $(\mathcal{K}, T)$. W.l.o.g. $(\mathcal{K}, T)$ again is a directed forest of height k. The large conjunction then has the effect that every world in $\mathcal{K}$ satisfies precisely one ℓ_i. Moreover, $(\mathcal{K}, T) \vDash \ell_0$. For these reasons, the lemma again applies, and from $(\mathcal{K}, T) \vDash \varphi^0$ we conclude $(\mathcal{K}^\circ, T) \vDash \varphi$, so φ is satisfiable. ☐

Originally, Ladner [94] proved the hardness of the satisfiability problem of modal logic by a direct reduction from the satisfiability problem of QPL. The proof worked, like our result, for any frame class between S4 and K. The new part of our result here is that such a flexible reduction is not only possible from, say, QPL, but even between different frame classes of modal logic. This way, it is also applies to modal team logic and possibly other logics as well.

4.9 Filtration in team semantics

Next, we study a model-theoretic concept called the *filtration* technique. Filtration turned out to be a powerful tool to prove that a logic has the *small model property*. This property states that, if a formula φ has a model, then φ has a model that obeys a specific size bound. If this is for instance an exponential size in $|\varphi|$, then the logic has the *exponential model property*. Prominent examples of logics with this property are modal logics [9], dynamic and temporal logics [30], and even fragments of first-order logic [48]. The application of filtration to modal logic seems to go back to Segerberg [127] as well as Gabbay [35].

In this section, we adapt the filtration technique to team semantics, and prove the exponential model property for a non-trivial fragment of $\mathsf{ML}(\sim)$ we call $\mathsf{ML}(\mathsf{mon})$.

4.9.1 Morphisms and filtrations

Definition 4.37 (Modal homomorphism)**.** Let $\mathcal{K} = (W, R, V)$ and $\mathcal{K}' = (W', R', V')$ be Kripke structures over $\Phi \subseteq \mathsf{Prop}$. A mapping $h\colon W \to W'$ is a *homomorphism*, in symbols $h\colon \mathcal{K} \to \mathcal{K}'$, if

(1) for all $p \in \Phi$, if $w \in V(p)$, then $h(w) \in V'(p)$

(2) for all $w, v \in W$, if Rwv, then $R'f(w)f(v)$.

If h is additionally surjective, then $\mathcal{K}'$ is called *morphic image* of $\mathcal{K}$ and written $h(\mathcal{K})$.

If $\approx$ is an equivalence relation on a set S, then $[s]_\approx := \{ s' \in S \mid s' \approx s \}$ denotes the *equivalence class* of $s \in S$. The set of all equivalence classes in S is the *quotient* $S/_\approx := \{ [s]_\approx \mid s \in S \}$, and the *index* of $\approx$ is the cardinality $|S/_\approx|$. For a subset $U \subseteq S$, let $[U]_\approx := \{ [s]_\approx \mid s \in U \}$. We often will drop the subscript and write $[s]$ and $[U]$. If $\approx$ and $\approx'$ are equivalence relations on S such that $s \approx' s'$ implies $s \approx s'$, then $\approx'$ is a *refinement* of $\approx$. Given two equivalence relations $\approx_1, \approx_2$ on S, their intersection $\approx_1 \cap \approx_2$ is again an equivalence relation on S and a refinement of both $\approx_1$ and $\approx_2$, and $|S/_{\approx_1 \cap \approx_2}| \leqslant |S/_{\approx_1}| \cdot |S/_{\approx_2}|$.

Definition 4.38 (Filtration)**.** Let $\mathcal{K} = (W, R, V)$ be a Kripke structure. Let $\approx$ be an equivalence relation on W. Then the Kripke structure (W', R', V') defined by

$$\begin{aligned} W' &:= W/_\approx, \\ R'[w][v] &\Leftrightarrow \exists w' \in [w], \exists v' \in [v] \text{ such that } Rwv, \\ [w] \in V'(p) &\Leftrightarrow [w] \cap V(p) \neq \emptyset, \end{aligned}$$

is the *filtration of $\mathcal{K}$ through* $\approx$, denoted $\mathcal{K}/_\approx$.[1]

Every filtration of a structure is also a morphic image of it, via the mapping $w \mapsto [w]$.

[1]The definition used here is also known as the *minimal filtration*.

Standard modal logic has, for any given formula $\alpha \in \mathsf{ML}$ and model $\mathcal{K}$, a filtration down to a model of α of size exponential in $|\alpha|$. The approach is the following: For a fixed Kripke structure $\mathcal{K} = (W, R, V)$ and a subset $\Gamma \subseteq \mathsf{ML}$, we define an equivalence relation $\approx_\Gamma$ on W such that $w \approx_\Gamma w'$ if and only if $\forall \alpha \in \Gamma : (\mathcal{K}, w) \vDash \alpha \Leftrightarrow (\mathcal{K}, w') \vDash \alpha$. The result is the next theorem, which is standard (see, e.g., Blackburn and van Benthem [9]):

Theorem 4.39. *Let $\mathcal{K} = (W, R, V)$ be a Kripke structure, and let $\Gamma \subseteq \mathsf{ML}$ be closed under taking subformulas, i.e., $\mathsf{sub}(\Gamma) = \Gamma$. Let $\approx'$ be any refinement of $\approx_\Gamma$. Then $(\mathcal{K}, w) \vDash \alpha$ iff $(\mathcal{K}/_{\approx'}, [w]_{\approx'}) \vDash \alpha$, for all formulas $\alpha \in \Gamma$ and worlds $w \in W$.*

Corollary 4.40 (Small model property of modal logic)**.** *Every satisfiable formula $\alpha \in \mathsf{ML}$ has a model of size at most $2^{|\alpha|}$.*

4.9.2 Strongly invariant formulas

An obvious generalization of Theorem 4.39 is to replace the universal quantification of worlds by that of teams. It is plausible that a similar inductive proof works for teams. The next definition yields some filtration results for team semantics, but its limits are quickly reached, as we see later in this section.

Definition 4.41. If $\mathcal{K} = (W, R, V)$ is a Kripke structure, $T \subseteq W$ is a team, $\approx$ is an equivalence relation on W, and $\Phi \subseteq \mathsf{ML}(\sim)$, then $\approx$ is

- *Φ-invariant on* $(\mathcal{K}, T)$ if $\forall \varphi \in \Phi : (\mathcal{K}, T) \vDash \varphi \;\Leftrightarrow\; (\mathcal{K}/_{\approx}, [T]_{\approx}) \vDash \varphi$,
- *Φ-invariant on* $\mathcal{K}$ if it is $(\mathcal{K}, T)$-invariant for all $T \subseteq W$,
- *strongly Φ-invariant* on $\mathcal{K}$ (resp. $(\mathcal{K}, T)$) if every refinement $\approx'$ of $\approx$ is Φ-invariant on $\mathcal{K}$ (resp. $(\mathcal{K}, T)$).

Recall that $\mathsf{sub}(\Gamma)$ is the closure of Γ under taking subformulas.

Proposition 4.42. *Let $\Gamma \subseteq \mathsf{ML}$. Then on any structure $\mathcal{K}$, the corresponding equivalence relation $\approx_{\mathsf{sub}(\Gamma)}$ is strongly Γ-invariant on $\mathcal{K}$.*

Proof. Let $\mathcal{K} = (W, V, R)$, and let $\approx'$ be a refinement of $\approx_{\mathsf{sub}(\Gamma)}$. We have to show that $\approx'$ is α-invariant on $\mathcal{K}$ for all $\alpha \in \Gamma$. By Theorem 4.39, it holds $w \vDash \alpha \Leftrightarrow [w]_{\approx'} \vDash \alpha$ for all $w \in W$. The statement is then proven, since for all $T \subseteq W$,

$$\begin{aligned}
& T \vDash \alpha \\
\Leftrightarrow\; & \forall w \in T : w \vDash \alpha && \text{(flatness)} \\
\Leftrightarrow\; & \forall w \in T : [w]_{\approx'} \vDash \alpha && \text{(by assumption)} \\
\Leftrightarrow\; & \forall [w]_{\approx'} \in [T]_{\approx'} : [w]_{\approx'} \vDash \alpha && \text{(def. } [T]_{\approx'}) \\
\Leftrightarrow\; & [T]_{\approx'} \vDash \alpha && \text{(flatness)}
\end{aligned}$$ □

The above result demonstrates that team semantics supports filtration when restricted to flat formulas. Next, we proceed with non-flat fragments of $\mathsf{ML}(\sim)$ and show that they still admit filtration.

Definition 4.43 ($\mathcal{B}$- and $\mathcal{S}$-closures)**.** If $\Phi \subseteq \mathsf{ML}(\sim)$, then $\mathcal{B}(\Phi)$ is the closure of Φ under $\sim$ and $\wedge$, and $\mathcal{S}(\Phi)$ is the closure of Φ under $\sim, \wedge$ and $\vee$.

Clearly $\Phi \subseteq \mathcal{B}(\Phi) \subseteq \mathcal{S}(\Phi) \subseteq \mathsf{ML}(\sim)$.

Lemma 4.44. *Let $\mathcal{K} = (W, R, V)$ be a structure and $\Phi \subseteq \mathsf{ML}(\sim)$. If $\approx$ is (strongly) Φ-invariant on $\mathcal{K}$, then $\approx$ is also (strongly) $\mathcal{S}(\Phi)$-invariant on $\mathcal{K}$.*

Proof. The proof is by induction on φ, where $\varphi \in \mathcal{S}(\Phi)$. The inductive step is clear for the truth-functional connectives on the level of teams, i.e., $\sim$ and $\wedge$.

Next, suppose $\varphi = \psi_1 \vee \psi_2$. Let $(\mathcal{K}, T) \vDash \varphi$ via $S \cup U = T$, $S \vDash \psi_1$ and $U \vDash \psi_2$. Then clearly $[T]_\approx = [S]_\approx \cup [U]_\approx$. By induction hypothesis, $[T]_\approx \vDash \psi_1 \vee \psi_2$.

Conversely, let $[T]_\approx \vDash \varphi$ via teams $\tilde{S}, \tilde{U} \subseteq W/_\approx$ such that $\tilde{S} \cup \tilde{U} = [T]_\approx$, $\tilde{S} \vDash \psi_1$ and $\tilde{U} \vDash \psi_2$. There is not necessarily a unique choice of $S, U \subseteq T$ such that $[S]_\approx = \tilde{S}$ and $[U]_\approx = \tilde{U}$, so we choose corresponding S and U as large as possible to ensure T is covered by $S \cup U$. Namely, define $S := \{ w \in T \mid [w]_\approx \in \tilde{S} \}$ and $U := \{ w \in T \mid [w]_\approx \in \tilde{U} \}$. If now $w \in T$, then $[w]_\approx \in [T]_\approx$, so $[w]_\approx \in \tilde{S}$ or $[w]_\approx \in \tilde{U}$, and consequently $w \in S$ or $w \in U$. Therefore $T \subseteq S \cup U$. By definition $S, U \subseteq T$, so $T = S \cup U$.

To show that $T \vDash \psi_1 \vee \psi_2$ follows from the induction hypothesis, it remains to show that actually $[S]_\approx = \tilde{S}$ resp. $[U]_\approx = \tilde{U}$ holds. Suppose $[w]_\approx \in [S]_\approx$. (The proof is analogous for U). Then by definition of $[\cdot]_\approx$ there exists $\hat{w} \in S$ such that $\hat{w} \approx w$, again implying by definition of S that $[\hat{w}]_\approx = [w]_\approx \in \tilde{S}$. Hence $[S]_\approx \subseteq \tilde{S}$.

Let conversely $[w]_\approx \in \tilde{S}$. Then $[w]_\approx \cap T$ is non-empty, since otherwise $[w]_\approx \notin [T]_\approx$ by definition of $[\cdot]_\approx$, contradicting $[w]_\approx \in \tilde{S} \subseteq [T]_\approx$. Hence there exists some $\hat{w} \in T$ such that $\hat{w} \approx w$. Since $\hat{w} \in T$ and $[\hat{w}]_\approx = [w]_\approx \in \tilde{S}$, it follows $\hat{w} \in S$ by the definition of S, and hence $[\hat{w}]_\approx = [w]_\approx \in [S]_\approx$ as desired. □

Theorem 4.45. *For every Kripke structure $\mathcal{K}$ and every finite $\Phi \subseteq \mathcal{S}(\mathsf{ML})$ there is an equivalence relation of index at most $\prod_{\varphi \in \Phi} 2^{|\varphi|}$ that is strongly Φ-invariant on $\mathcal{K}$.*

Proof. Let $\Gamma := \mathsf{sub}(\Phi) \cap \mathsf{ML}$. By Proposition 4.42, $\approx_\Gamma$ is Γ-invariant. $|\Gamma| \leqslant \sum_{\varphi \in \Phi} |\varphi|$, so $\approx_\Gamma$ has index at most $\prod_{\varphi \in \Phi} 2^{|\varphi|}$. Since $\Phi \subseteq \mathcal{S}(\Gamma)$, the theorem follows from Lemma 4.44. □

Corollary 4.46. *Every satisfiable $\varphi \in \mathcal{S}(\mathsf{ML})$ has a model of size at most $2^{|\varphi|}$.*

Based on these results, it appears that (strong) invariance is a natural property of filtrations. It seems like a straightforward tool to generalize the usual filtration technique to team semantics. However, it is inadequate when team-wide modalities come into play, as the following counter-example shows. Recall the $\mathsf{PL}(\sim)$-formula from p. 87,

$$\mathsf{max}(\Phi) := \sim \bigvee_{p \in \Phi} \mathsf{dep}(p),$$

due to Hannula et al. [56], where $\Phi \subseteq \mathsf{Prop}$ is a finite set of propositions, and $\mathsf{dep}(p) := p \varovee \neg p$. Clearly, $\mathsf{max}(\Phi)$ has length $\mathcal{O}(|\Phi|)$. It is true in a team T if and only if all Boolean assignments to variables in Φ appear in T. The following counter-example is based on it. It even applies to a more general notion of homomorphism than filtration.

Definition 4.47. Let $\mathrm{h}\colon \mathcal{K} \to \mathcal{K}'$ be a homomorphism between Kripke structures and $\Phi \subseteq \mathsf{ML}(\sim)$. If T is a team in $\mathcal{K}$, then $h(T) := \{ h(w) \mid w \in T \}$.

- h is Φ-*invariant on* $(\mathcal{K}, T)$ if $(\mathcal{K}, T) \vDash \varphi \Leftrightarrow (\mathcal{K}', h(T)) \vDash \varphi$ for all $\varphi \in \Phi$,
- h is Φ-*invariant on* $\mathcal{K}$ if it is Φ-invariant on $(\mathcal{K}, T)$ for all teams T in $\mathcal{K}$.

We simply write φ-invariant instead of $\{\varphi\}$-invariant.

Theorem 4.48. *Let* $\Phi \subseteq \mathsf{Prop}$ *and* $\varphi = \Box\mathsf{max}(\Phi)$. *Then there is a structure* $\mathcal{K}$ *such that the image of any homomorphism that is* φ*-invariant on* $\mathcal{K}$ *has size at least* $2^{2^{|\Phi|}}$.

Proof. Let $\Phi := \{p_1, \ldots, p_n\}$. Define $\mathcal{K} = (W, R, V)$ as follows. Let W be the set of all propositional teams over Φ, that is, $W := \wp(\Phi \to \{0, 1\})$. Consequently, $|W| = 2^{2^n}$.

For any singleton $X = \{s\} \in W$ of exactly one assignment s, let $\{s\} \in V(p) \Leftrightarrow s(p) = 1$, that is, worlds in the Kripke structure that are singletons mimic the propositional labeling represented by their unique member. In all other worlds, all propositions are true, i.e., if $|X| \neq 1$, then $X \in V(p)$ for all $p \in \Phi$. Finally, for all $X \in W$ and $s \in X$, add the edge from X to $\{s\}$.

Let now $h(\mathcal{K})$ be a morphic image of $\mathcal{K}$ with h being φ-invariant. As a first step, we show that h is also p-invariant for all $p \in \Phi$, which is not clear just from the fact that is φ-invariant. By definition of homomorphism, $X \vDash p$ implies $h(X) \vDash p$. For the converse direction, suppose $X \nvDash p$ but $h(X) \vDash p$ for some $X \in W$ and $p \in \Phi$. Then by construction of V, X must be one of the singletons $\{s\} \in W$ for $s\colon \Phi \to \{0, 1\}$. It is not hard to see that of the $2^n - 1$ assignments with at least one zero, only $2^n - 2$ of them occur as the valuation of a world in $h(\mathcal{K})$. As a consequence, no team in $h(\mathcal{K})$ can satisfy $\Box\mathsf{max}(\Phi)$ anymore, and in particular not $h(T)$. So $X \nvDash p$ must imply $h(X) \nvDash p$.

Next, we come to the actual proof. Suppose that the image $h(\mathcal{K})$ has *less* than 2^{2^n} worlds. Then h is not injective, i.e., $h(X) = h(Y)$ for some distinct $X, Y \in W$. W.l.o.g. there is an assignment $s \in X \setminus Y$. Let $Z := (\Phi \to \{0, 1\}) \setminus \{s\}$ and consider now the team $T := \{ Y, Z \}$. Neither of its element has $\{s\}$ as successor, and no edges lead to non-singleton elements of W, and so there is no edge from T to any world with the same propositions as in the assignment s. Hence $T \nvDash \Box\mathsf{max}(\Phi)$. But below we show that $h(T) \vDash \Box\mathsf{max}(\Phi)$, which contradicts the fact that h is φ-invariant, so $h(\mathcal{K})$ must have 2^{2^n} or more worlds.

By definition, Z has $\{s'\}$ as successor for any assignment $s' \neq s$. Recall that $h(X) = h(Y)$, so $h(T) = \big\{h(Y), h(Z)\big\} = \big\{h(X), h(Z)\big\} = T'$. But the team T' is a team in $h(\mathcal{K})$ that satisfies $\Box\mathsf{max}(\Phi)$, since h preserves edges, and so $h(X)$ has successor $h(\{s\})$, and $h(Z)$ has successor $h(\{s'\})$ for all $s' \neq s$. ∎

Corollary 4.49. *There are families* $(\varphi_n)_{n \geqslant 0}$ *of* $\mathsf{PL}(\sim)$*-formulas of size* $|\varphi_n| \in \mathcal{O}(n)$ *and structures* $(\mathcal{K}_n)_{n \geqslant 0}$ *such that every* $\Box\varphi_n$*-invariant equivalence relation on* $\mathcal{K}_n$ *has index at least* 2^{2^n}.

Recall that by Theorem 4.45, $\mathcal{S}(\mathsf{ML})$ has filtration down to an exponential model. Hence there is a succinctness gap between $\mathcal{S}(\mathsf{ML})$ and $\mathcal{S}(\mathsf{ML})$ preceded by modal operators.

Corollary 4.50. *There is a family* $(\varphi_n)_{n\geqslant 0}$ *of* $\mathcal{S}(\mathsf{ML})$*-formulas of size* $\mathcal{O}(n)$ *such that every* $\mathcal{S}(\mathsf{ML})$*-formula equivalent to* $\Box\varphi_n$ *has length* $\geqslant 2^n$*.*

Also, any homomorphism invariant for $\Diamond\mathsf{max}(\Phi)$ is also invariant for $\top \vee \Diamond\mathsf{max}(\Phi)$, which is shown similarly to Lemma 4.44. But $\top \vee \Diamond\mathsf{max}(\Phi) \equiv \Box\mathsf{max}(\Phi)$, so as before, no $\Diamond\mathsf{max}(\Phi)$-invariant homomorphism exists for $\mathcal{K}$ with an image of size $< 2^{2^n}$.

Corollary 4.51. *There is a family* $(\varphi_n)_{n\geqslant 0}$ *of* $\mathcal{S}(\mathsf{ML})$*-formulas of size* $\mathcal{O}(n)$ *such that every* $\mathcal{S}(\mathsf{ML})$*-formula equivalent to* $\Diamond\varphi_n$ *has length* $\geqslant 2^n$*.*

However, if $\varphi \in \mathcal{B}(\mathsf{ML})$, then $\Box\varphi$ has an equivalent $\mathcal{B}(\mathsf{ML})$-formula of length only $\leqslant 2|\varphi|$. The translation is performed by $\Box{\sim}\psi \equiv {\sim}\Box\psi$ and $\Box(\psi \wedge \psi') \equiv \Box\psi \wedge \Box\psi'$.

Corollary 4.52. *There is a family* $(\varphi_n)_{n\geqslant 0}$ *of* $\mathcal{S}(\mathsf{ML})$*-formulas of size* $|\varphi_n| \in \mathcal{O}(n)$ *such that every* $\mathcal{B}(\mathsf{ML})$*-formula equivalent to* φ_n *has length* $\geqslant 2^n$*.*

In other words, $\Box$ is easy to distribute over $\wedge$ and $\sim$, but hard to distribute over $\vee$, and $\mathcal{S}(\mathsf{ML})$ is exponentially more succinct than $\mathcal{B}(\mathsf{ML})$.

4.9.3 A weak filtration for monotone modal team logic

An exponential model property can be obtained for larger fragments of $\mathsf{ML}(\sim)$, provided the requirements of filtration are weakened properly. An obvious candidate is the *invariance* property. To find a small model of φ starting from a given model $(\mathcal{K}, T)$, it is unnecessary to have ${\sim}\varphi$ preserved as well; hence we replace invariance by an asymmetric feature called *preservation*.

Moreover, a filtration $\approx$ does not need to preserve a formula φ in *all* teams of a model $(\mathcal{K}, T)$ — having φ true in $[T]_\approx$ would be completely sufficient. For this reason, we do not define preservation on the whole structure $\mathcal{K}$, but only *locally*:

Definition 4.53. If $\approx$ is an equivalence relation on a Kripke structure $\mathcal{K} = (W, R, V)$, $T \subseteq W$ is a team, and $\Phi \subseteq \mathsf{ML}(\sim)$, then $\approx$ is

- Φ*-preserving on* $(\mathcal{K}, T)$ if $\forall\varphi \in \Phi : (\mathcal{K}, T) \vDash \varphi \;\Rightarrow\; (\mathcal{K}/_\approx, [T]_\approx) \vDash \varphi$,
- *strongly* Φ*-preserving* if every refinement $\approx'$ of $\approx$ is Φ-preserving.

The above property is clearly not closed under negation, but still closed under the modal connectives. In this context, we consider the $\mathsf{ML}(\sim)$ operators $\wedge, \vee, \Diamond$ and $\Box$ as *monotone*, and also add the Boolean disjunction $\varovee$ as a primitive connective. Based on these, we define the following fragment.

Definition 4.54. The *monotone fragment* $\mathsf{ML}(\mathsf{mon})$ of $\mathsf{ML}(\sim)$ is defined as the closure of $\mathcal{S}(\mathsf{ML})$ under $\wedge, \varovee, \vee, \Box$ and $\Diamond$.

In a series of lemmas, we will prove the following upper bound for this restricted fragment:

Theorem 4.55. *For every finite* $\Phi \subseteq \mathsf{ML}(\mathsf{mon})$*, every structure* $\mathcal{K} = (W, R, V)$*, and every team* $T \subseteq W$*, there is an equivalence relation of index at most* $\prod_{\varphi \in \Phi} 2^{|\varphi|}$ *that is strongly* Φ*-preserving on* $(\mathcal{K}, T)$.

Note that we still quantify over all teams T, but are allowed to choose a different filtration for each team. The order of these quantifications makes a crucial difference here, in particular it eliminates the vulnerability against the method of Theorem 4.48 for filtration lower bounds.

In the following lemmas, let $\mathcal{K} = (W, V, R)$ be a Kripke structure and $\approx$ an equivalence relation on W, and accordingly $\mathcal{K}/_{\approx} = (W/_{\approx}, R', V')$ as in Definition 4.38.

First we prove that subformulas starting with $\Diamond$ are preserved.

Lemma 4.56. *If* S *is a successor team of* T*, then* $[S]_{\approx}$ *is a successor team of* $[T]_{\approx}$.

Proof. Suppose that S is a successor team of T. We have to show that every $[w] \in [T]$ has an R'-successor in $[S]$ and that every $[v] \in [S]$ has an R'-predecessor in $[T]$. Let $[w] \in [T]$, then $w' \in T$ for some $w' \approx w$. w' has an R-successor $v \in S$, so $[v] \in [S]$. But $Rw'v$ implies $R'[w'][v]$, so $[w'] = [w]$ has an R'-successor in $[S]$.

Conversely, if $[v] \in [S]$, then $v' \in S$ for some $v' \approx v$. v' has an R-predecessor $w \in T$. However, Rwv' again implies $R'[w][v']$, so $R'[w][v]$ for some $[w] \in [T]$. ☐

Formulas starting with $\Box$ are similarly preserved in teams T, at least as long as the filtration does not cross the boundaries of the preimage team T:

Lemma 4.57. *If* S *is the image of* T*, and* T *is closed under* $\approx$ *(i.e.,* $w \approx w'$ *and* $w \in T$ *implies* $w' \in T$*), then* $[S]$ *is the image of* $[T]$.

Proof. As in the previous lemma, every $[v] \in [S]$ has an R'-predecessor in $[T]$. It remains to prove that $[S]$ contains all R'-successors $[v]$ of all $[w] \in [T]$. Let $R'[w][v]$ for $[w] \in [T]$. There exist $w' \approx w$ and $v' \approx v$ such that $Rw'v'$. By assumption of the lemma, $w' \in T$, so its R-successor v' must be in S, and $[v'] = [v] \in [S]$. ☐

It is easy to verify that the converse of the above two lemmas is false, or in other words, that the negations of modal operators are not preserved in filtrations.

We are now ready to prove the theorem.

Proof of Theorem 4.55. Let $\mathcal{K} = (W, R, V)$, $T \subseteq W$ and $\Phi \subseteq \mathsf{ML}(\mathsf{mon})$ be as in Theorem 4.55 such that $(\mathcal{K}, T) \vDash \Phi$. Below, we define an equivalence relation $\approx$ and show that $\varphi := \bigwedge_{\psi \in \Phi} \psi$ is strongly preserved.

By definition of $\mathsf{ML}(\mathsf{mon})$, φ is a monotone combination (i.e., using only operators $\wedge, \otimes, \vee, \Diamond, \Box$) of $\mathcal{S}(\mathsf{ML})$ formulas. We exploit the monotonicity and define a *witness team* $T^\star(\psi)$ for subformulas ψ of φ. W.l.o.g. every subformula of φ occurs only once in φ. The idea is that it suffices to preserve these subformulas in their corresponding witness teams instead of the whole structure.

The team $T^\star(\psi) \subseteq W$ is defined top-down for each $\psi \notin \mathcal{S}(\mathsf{ML})$ such that $(\mathcal{K}, T^\star(\psi)) \vDash \psi$. Accordingly, we start by setting $T^\star(\varphi) := T$. Whenever $T^\star(\Box\psi)$ is defined for $\Box\psi \in \mathsf{sub}(\varphi)$, set $T^\star(\psi) := RT^\star(\Box\psi)$. Intuitively, if the team $T' := T^\star(\Box\psi)$ witnesses $\Box\psi$, then the team RT' must witness ψ.

Similarly, $T^\star(\Diamond\psi)$ must have a successor team S that satisfies ψ, so set $T^\star(\psi) := S$. Any team $T^\star(\psi \vee \psi')$, for $\psi \vee \psi' \in \mathsf{sub}(\varphi)$, likewise can be split into $S \vDash \psi$ and $U \vDash \psi'$, consequently then $T^\star(\psi) := S$ and $T^\star(\psi') := U$. If $\psi = \psi' \wedge \psi''$ or $\psi = \psi' \otimes \psi''$ is in $\mathsf{sub}(\varphi)$, then $T^\star(\psi')$ and/or $T^\star(\psi'')$ simply equals $T^\star(\psi)$.

The equivalence relation $\approx$ is now constructed as follows. Similarly as in Theorem 4.45, let $\Gamma := \mathsf{sub}(\varphi) \cap \mathsf{ML}$. Define $\approx'$ as the coarsest refinement of $\approx_\Gamma$ that does not cross the boundaries of witness teams $T^\star(\psi)$, i.e., $w \approx' w'$ if and only if $w \approx_\Gamma w'$ and, for all $\psi \in \mathsf{sub}(\varphi)$, $w \in T^\star(\psi) \Leftrightarrow w' \in T^\star(\psi)$.

Lemmas 4.56 and 4.57 now allow to prove $(\mathcal{K}/_{\approx''}, [T^\star(\psi)]_{\approx''}) \vDash \psi$ for any refinement $\approx''$ of $\approx'$, and any $\psi \in \mathsf{sub}(\varphi)$, by induction on ψ. The splitting case is handled as in Lemma 4.44. As $\approx'$ has index at most $2^{|\mathsf{sub}(\varphi)\cap\mathsf{ML}|} \cdot 2^{|\mathsf{sub}(\varphi)\setminus\mathcal{S}(\mathsf{ML})|} \leqslant 2^{|\varphi|}$, this proves the theorem. ☐

Corollary 4.58. *Every satisfiable formula* $\varphi \in \mathsf{ML}(\mathsf{mon})$ *has a model of size at most* $2^{|\varphi|}$.

Proposition 4.59. *The satisfiability problem of* $\mathsf{ML}(\mathsf{mon})$ *is* ATIME-ALT(exp, poly)*-complete.*

Proof. The hardness already holds for $\mathsf{PL}(\sim)$ [56], which coincides with the modality-free fragment of $\mathsf{ML}(\mathsf{mon})$, so we only need to prove the upper bound. By Theorem 2.32, the model checking problem for $\mathsf{ML}(\sim)$ is decidable by an alternating Turing machine that with runtime polynomial in $|\mathcal{K}| + |\varphi|$ and alternations polynomial in $|\varphi|$. This allows to decide the satisfiability problem of $\mathsf{ML}(\mathsf{mon})$ as follows. Given a formula φ, guess a Kripke structure $\mathcal{K}$ of size up to $2^{|\varphi|}$ and a team T in $\mathcal{K}$. Then execute the above model checking algorithm on $(\mathcal{K}, T, \varphi)$. By the preceding corollary, the algorithm decides $\mathsf{ML}(\mathsf{mon})$ in exponential runtime and with polynomially many alternations. ☐

4.10 Summary and outlook

4.10.1 Summary

Theorem 4.32 settles the complexity of $\mathsf{ML}(\sim)$ and proves that its satisfiability and validity problems are complete for the non-elementary complexity class TOWER(poly). Moreover, the fragments $\mathsf{ML}_k(\sim)$ are proved complete for ATIME-ALT$(\exp_{k+1}, \mathrm{poly})$, the levels of the elementary hierarchy with polynomially many alternations.

In our approach, we developed a notion of (k-)canonical models for modal logics with team semantics. We demonstrated that such models exist for $\mathsf{ML}(\sim)$ and $\mathsf{ML}_k(\sim)$. Moreover, we also proved a matching lower bound for this (Theorem 4.24) in the sense that small $\mathsf{ML}_k(\sim)$-formulas exist that are satisfiable, but only have k-canonical models.

Afterwards, we considered variants of the satisfiability problem for $\mathsf{ML}(\sim)$. We showed that it is as hard as the original problem when $\mathsf{ML}(\sim)$ is interpreted in strict semantics, and in fact already for $\Diamond$-free formulas with $\Box$ and either $\vee$ or $\dot\vee$. Also, any restriction of the satisfiability problem to a frame class that includes at least the reflexive transitive frames is at least as hard.

Using the filtration method, we found the fragment $\mathsf{ML}(\mathsf{mon})$ of $\mathsf{ML}(\sim)$ with its satisfiability problem in ATIME-ALT(exp, poly). It seems that there is a tipping point with respect to the complexity of the satisfiability problem: ML-formulas, and their closure under $\wedge, \vee$ and $\sim$ are easy to solve. Furthermore, adding a layer of $\Diamond$, $\Box$, $\vee$, $\oslash$ and $\wedge$ around these (resulting in $\mathsf{ML}(\mathsf{mon})$) does still not increase the complexity. The jump in computational hardness happens as soon as $\vee$ as well as $\Box$ and/or $\Diamond$ are allowed to occur negatively. A close inspection of the proof of Theorem 4.19 and the following theorems show that *universal splitting quantification*, and thus negatively occurring $\vee$, is indispensable in order to define subteam quantifiers, which we utilized to define bisimilarity and canonicity. For this reason, the reduction cannot work in $\mathsf{ML}(\mathsf{mon})$, which explains why the latter has only elementary complexity.

4.10.2 Open problems and further research directions

Canonical models. In future research, it could be useful to further generalize the concept of canonical models to other logics with team semantics. As a first idea, in their proof that the satisfiability problem of FO^2 is in NEXPTIME, Grädel et al. [48] defined *types* as maximal consistent set of literals quite similar to our definition of (Φ, k)-types in Definition 4.1. Can this approach be adapted to a notion of canonical models for $\mathsf{FO}^2(\sim)$? In this vein, can a small model property can be achieved, perhaps again by filtration? This would also be interesting because for the fragments $\mathsf{ML}_k(\sim)$ of bounded modal depth k we obtained completeness results for the levels of the elementary hierarchy. The corresponding fragments of first-order logic would be $\mathsf{FO}^2_k(\sim)$, the two-variable fragments with additionally bounded quantifier depth k. Since the model checking problem of $\mathsf{FO}^2_k(\sim)$ also is in AP (see the next chapter), this would imply that its satisfiability problem is complete for ATIME-ALT($\exp_{k+1}$, poly) as well.

In principle we could solve $\mathsf{FO}^2_k(\sim)$ by expanding its formulas as in the result Corollary 3.92, but that translation is non-elementary in the nesting depth of $\sim$, even if k is constant. Hence, for the fragment $\mathsf{FO}^2_k(\sim)$, it seems that a detour via a small model property is necessary, similar to $\mathsf{ML}_k(\sim)$.

Strict semantics and frame classes. Two other open problems are the matching upper bounds for Theorems 4.34 and 4.35. Does the complexity coincide with the case of lax semantics and arbitrary frames? To solve these issues, it seems that the model theory of modal team logic has to be refined. For example, what is the analog of Proposition 2.30 for strict semantics? More specifically, what is a suitable notion of bisimulation $\rightleftharpoons^{\Phi}_{k}$ such that $(\mathcal{K}_1, T_1) \rightleftharpoons^{\Phi}_{k} (\mathcal{K}_2, T_2)$ if and only if $(\mathcal{K}_1, T_1)$ and $(\mathcal{K}_2, T_2)$ satisfy the same $\mathsf{ML}_k(\sim)$-formulas, but with strict semantics?

It seems that any sensible notion of bisimilarity, canonical model and types for strict semantics has to account for *multiplicity* of types, so an approach similar to, e.g., the *multiteam semantics* by Durand et al. [26] seems to be a good starting point.

Parameterized complexity. In *parameterized complexity theory*, introduced by Downey and Fellows [25], decision problems are studied with respect to a so-called *parameter* κ, which is a function that maps every input to a natural number. A problem is called *fixed-parameter tractable* (*fpt*) if it is decidable in time $f(\kappa(x)) \cdot |x|^{\mathcal{O}(1)}$ for some computable function f. The idea is that the runtime of the algorithm, even if it may depend on $\kappa(x)$, is reasonably short if we assume κ to take only small values on "practical" input instances, and indeed many NP-complete problems turn out as *fpt*. In the area of team logic, recently Meier and Reinbold [113] studied the enumeration complexity of a fragment of propositional dependence logic in terms of parameterized complexity. In the same vein, Mahmood and Meier [110] considered the model checking problem of $\mathsf{PL}(\mathsf{dep})$.

In modal team logics, several measures suggest themselves as a parameter. Examples include the modal depth of formulas, the nesting depth of $\sim$, the number of occurring propositions, the size of the team, or the treewidth of the underlying syntax dag of the input formula. Parameters such as the modal depth, number of propositions, or treewidth have already been studied in the setting of temporal logic (cf. Lück and Meier [106] and Lück et al. [107]) and modal logic (cf. Achilleos et al. [3]). Investigating those parameters potentially leads to new progress on the complexity of (modal) team logic.

5 First-order team logic

In the past decades, the work of logicians has unearthed a plethora of decidable fragments of first-order logic FO. Many of these decidability results are rooted in a finite model property: if there exists a (computable) upper bound on the size of minimal models with respect to a class of formulas, and the logic admits effective model checking, then the question of satisfiability can be settled by exhaustively searching all structures of suitable size. Prominent examples are logics with restricted quantifier prefixes, such as the BSR-fragment which contains only $\exists^*\forall^*$-sentences [122]. Others include the monadic class [98], the guarded fragment GF [4, 45], the recently introduced separated fragment [133, 140], or the two-variable fragment FO^2 [48, 116, 126], which all are decidable. The above fragments all have been subject to intensive study with the purpose of further pushing the boundary of decidability.

In this chapter, we continue this line of research in the setting of team semantics. We study the logic $\mathsf{FO}(\sim)$, that is, first-order team logic with negation but no atoms of dependence. In the following sections, we show that $\mathsf{FO}(\sim)$ shares many nice properties with FO, including several decidable fragments. First, in Section 5.1, we revisit the idea of *normal forms* which we treated abstractly in Section 3.7, and now give some concrete applications for the case of first-order logic. One is the result that $\mathsf{FO}(\sim)$ is recursively enumerable, and hence of the same complexity as FO, that is, its satisfiability problem is Π^0_1-complete. Moreover, in the same spirit, we consider an analog to two-variable logic FO^2 and the guarded fragment GF in team semantics, which we call $\mathsf{FO}^2(\sim)$ and $\mathsf{GF}(\sim)$, and prove upper bounds for their complexity.

Next, in Section 5.2, we show that the classical standard translation from ML to FO^2 can be generalized to team semantics. With minor adaptations, we show that there is a similar translation from $\mathsf{ML}(\sim)$ to $\mathsf{FO}^2(\sim)$. This also carries the lower bounds from the previous chapter into the first-order setting.

Finally, in Section 5.3, we study $\mathsf{FO}(\sim)$ from the perspective of model theory and prove a variant of Łoś's ultraproduct theorem. Roughly speaking, it states that a structure and its ultrapower satisfy the same first-order formulas. We adapt this to team logic, which includes a novel definition of the ultraproduct of teams, and prove an analogous result for $\mathsf{FO}(\sim)$-formulas. This also entails, for instance, a variant of the compactness theorem for $\mathsf{FO}(\sim)$.

5.1 Upper bounds for satisfiability and validity

Several results on dependence logic and its variants stem from the well-known translation to second-order logic due to Väänänen [135] and Kontinen and Nurmi [84].

Distributive laws for $\varovee$ over $\wedge, \vee, \exists, \forall$:	
$(\theta_1 \varovee \theta_2) \wedge \theta_3 \equiv (\theta_1 \wedge \theta_3) \varovee (\theta_2 \wedge \theta_3)$	$\theta_1 \wedge (\theta_2 \varovee \theta_3) \equiv (\theta_1 \wedge \theta_2) \varovee (\theta_1 \wedge \theta_3)$
$(\theta_1 \varovee \theta_2) \vee \theta_3 \equiv (\theta_1 \vee \theta_3) \varovee (\theta_2 \vee \theta_3)$	$\theta_1 \vee (\theta_2 \varovee \theta_3) \equiv (\theta_1 \vee \theta_2) \varovee (\theta_1 \vee \theta_3)$
$\exists x(\theta_1 \varovee \theta_2) \equiv (\exists x\theta_1) \varovee (\exists x\theta_2)$	$\forall x(\theta_1 \varovee \theta_2) \equiv (\forall x\theta_1) \varovee (\forall x\theta_2)$
Distributive laws for $\wedge$ over $\vee, \exists, \forall$:	
$(\alpha \wedge \bigwedge_{i\in I} \mathsf{E}\beta_i) \vee (\gamma \wedge \bigwedge_{j\in J} \mathsf{E}\delta_j) \equiv (\alpha \vee \gamma) \wedge \bigwedge_{i\in I} \mathsf{E}(\alpha \wedge \beta_i) \wedge \bigwedge_{j\in J} \mathsf{E}(\gamma \wedge \delta_j)$	
$\exists x(\alpha \wedge \bigwedge_{i\in I} \mathsf{E}\beta_i) \equiv (\exists x\alpha) \wedge \bigwedge_{i\in I} \mathsf{E}\exists x(\alpha \wedge \beta_i)$	
$\forall x(\alpha \wedge \bigwedge_{i\in I} \mathsf{E}\beta_i) \equiv (\forall x\alpha) \wedge \bigwedge_{i\in I} \mathsf{E}\exists x(\alpha \wedge \beta_i)$	

Table 5.1: Distributive laws for $\mathsf{FO}(\sim)$

Dependence logic $\mathsf{FO}(\mathsf{dep})$ and team logic $\mathsf{FO}(\mathsf{dep}, \sim)$ are equivalent to existential and full second-order logic, respectively, so it appears there is not much left to say from a model-theoretic or complexity theoretic perspective.

The logic $\mathsf{FO}(\sim)$ on the other hand has no such known characterization. For sentences, it collapses to FO [38], and otherwise it can only express Boolean combinations of flat formulas (recall that $\mathsf{E}\gamma = {\sim}\neg\gamma$):

$$\varphi \equiv \bigovee_{i\in I} \left(\alpha_i \wedge \bigwedge_{j\in J_i} \mathsf{E}\beta_{i,j} \right)$$

This was first shown in [100, Thm. 7.5]. (We called this *quasi-flat* in Corollary 3.92, and the above form in particular $(\varovee\wedge)$-*normal form*.)

As a consequence, in the spirit of Kontinen and Nurmi [84] we can translate $\mathsf{FO}(\sim)$ without using *any* second-order quantifiers. The result is a Boolean combination of formulas of the form $\forall\vec{x}(R(\vec{x}) \to \alpha(\vec{x}))$ (where R is a new relation variable not appearing in α that represents the team), but this is not a particularly natural fragment of first-order logic.

The above normal form can be achieved by repeated application of the logical laws depicted in Table 5.1. For the distributive laws for $\varovee$, see Galliani [38, Prop. 5] or Proposition 3.40. The other laws were proven in general form in Lemmas 3.83 and 3.84.

A similar but independent result by Galliani [38], which was discovered earlier already by Yang [142] in the propositional setting, is that every $\mathsf{FO}(\sim)$-formula can be written as:[1]

$$\varphi \equiv \bigovee_{i\in I} \left(\alpha_i \vee \bigvee_{j\in J_i} (\beta_{i,j} \wedge \textsc{ne}) \right)$$

[1] In the case of downward closed team logic, we can omit the NE atom. For dependence logic, the corresponding normal form appeared already in Abramsky and Väänänen [2, p. 304].

We called this form $(\otimes\vee)$-*normal form* in Subsection 3.7.1. We also showed that these normal forms are easily mutually derivable, since

$$\alpha_i \wedge \bigwedge_{j \in J_i} E\beta_{i,j} \equiv \alpha_i \vee \bigvee_{j \in J_i} ((\alpha_i \wedge \beta_{i,j}) \wedge \text{NE}),$$

and conversely

$$\alpha_i \vee \bigvee_{j \in J_i} (\beta_{i,j} \wedge \text{NE}) \equiv (\alpha \vee \bigvee_{j \in J_i} \beta_{i,j}) \wedge \bigwedge_{j \in J_i} E\beta_{i,j}.$$

The above normal forms offer a point of attack for complexity theoretic considerations. In particular, by such a translation we establish a new decidability result for two-variable team logic $\mathsf{FO}^2(\sim)$. More precisely, we show that satisfiability and validity of $\mathsf{FO}^2(\sim)$ are TOWER(poly)-complete. This is an interesting result when compared with two-variable dependence logic, in our notation called $\mathsf{FO}^2(\mathsf{dep})$. These logics are incomparable in terms of expressive power, since $\mathsf{FO}^2(\mathsf{dep})$ is downward closed and $\mathsf{FO}^2(\sim)$ is not, but $\mathsf{dep}(x)$ is expressible in $\mathsf{FO}^2(\mathsf{dep})$ and not in $\mathsf{FO}^2(\sim)$ [38]. (The same holds without the restriction to two variables).

From the viewpoint of computational complexity, we now obtain a similar result. Specifically, $\mathsf{FO}^2(\mathsf{dep})$ has an undecidable validity problem [81], but only a NEXPTIME-complete satisfiability problem [80]. By contrast, satisfiability and validity of $\mathsf{FO}^2(\sim)$ are complete for TOWER(poly). That means that $\mathsf{FO}^2(\mathsf{dep})$ and $\mathsf{FO}(\sim)$ are also incomparable in terms of their computational complexity.

5.1.1 Computing the normal form

For the sake of self-containedness, we repeat some terminology from Chapter 3 and concretize it to first-order logic.

We say that a σ-$\mathsf{FO}(\sim)$-formula is in $(\otimes\wedge)$-normal form if it is of the form

$$\psi = \bigotimes_{i \in I} \left(\alpha_i \wedge \bigwedge_{j \in J_i} E\beta_{(i,j)} \right)$$

for finite sets I, J_i and σ-FO-formulas $\alpha_i, \beta_{(i,j)}$.

A formula is *quasi-flat* if it is equivalent to a formula in $(\otimes\wedge)$-normal form. In this chapter, we do not use the $(\otimes\vee)$-normal form, so let us just refer to the $(\otimes\wedge)$-form as *normal form*.

Theorem 5.1. *For every* $\varphi \in \sigma$-$\mathsf{FO}(\sim)$, *there exists an equivalent formula* ψ *in normal form computable from* φ *in time* $\exp_{\mathcal{O}(|\varphi|)}(1)$.

The first part, i.e., that there exists such a translation, was already shown in Chapter 3, in particular in Corollary 3.92. However, the abstract proof can be greatly simplified if we restrict ourselves to first-order logic. Let us sketch an outline of the proof for this special case.

Proof. Applied to σ-FO$(\sim)$, the idea of the proof of Theorem 3.89 boils down to the distributive laws depicted in Table 5.1. The algorithm that computes ψ from φ proceeds as follows. As long as there exists a subformula θ of φ that is not in normal form, pick one of the shortest such subformulas and replace it by a normal form in a case by case distinction regarding the outermost operator, as shown below. Suppose $\theta' = \bigovee_{i\in I}(\alpha_i \wedge \bigwedge_{j\in J_i} E\beta_{i,j})$ and $\theta'' = \bigovee_{i\in I'}(\gamma_i \wedge \bigwedge_{j\in J'_i} E\delta_{i,j})$.

- $\neg\theta' \equiv \bigwedge_{i\in I} \neg(\alpha_i \wedge \bigwedge_{j\in J_i} \beta_{i,j})$, which is flat.
- $\sim\theta' \equiv \bigwedge_{i\in I}(E\neg\alpha_i \wedge \bigovee_{j\in J_i} \neg\beta_{i,j})$, which can be expanded into normal form by standard propositional laws for $\ovee$ and $\wedge$.
- $\exists x\theta' \equiv \bigovee_{i\in I}[(\exists x\alpha_i) \wedge \bigwedge_{j\in J_i} E\exists x(\alpha_i \wedge \beta_{i,j})]$.
- $\forall x\theta' \equiv \bigovee_{i\in I}[(\forall x\alpha_i) \wedge \bigwedge_{j\in J_i} E\exists x(\alpha_i \wedge \beta_{i,j})]$.
- $\theta' \vee \theta'' \equiv \bigovee_{(i,i')\in I\times I'} \big((\alpha_i \wedge \bigwedge_{j\in J_i} E\beta_{i,j}) \vee (\gamma_{i'} \wedge \bigwedge_{j'\in J'_i} E\delta_{i',j'})\big)$
  $\equiv \bigovee_{(i,i')\in I\times I'} \big((\alpha_i \vee \gamma_{i'}) \wedge \bigwedge_{j\in J_i} E(\alpha_i \wedge \beta_{i,j}) \wedge \bigwedge_{j'\in J'_{i'}} E(\gamma_{i'} \wedge \delta_{i',j'})\big)$.

Each of the above equivalences was shown in the proof of Theorem 3.89 and/or follows from Table 5.1.

Since the number of subformulas that are not in normal form strictly decreases with each step, at most $|\varphi|$ iterations are needed. Moreover, as the $\sim$-case leads at most to doubly exponential blow-up, and the other cases to at most polynomial blow-up, the overall runtime is $\exp_{\mathcal{O}(|\varphi|)}(1)$. ☐

Clearly, the number of required elements in the team can be bounded from above by the number of E-subformulas in the above normal form.

Corollary 5.2. *If $\varphi \in$ FO$(\sim)$ and $(\mathcal{A}, T) \vDash \varphi$, then there is a finite subteam $T' \subseteq T$ of size $\exp_{\mathcal{O}(|\varphi|)}(1)$ such that $(\mathcal{A}, T') \vDash \varphi$.*

The above corollary is related to the notion of k-coherence. A formula φ is k-*coherent* if a structure with team $(\mathcal{A}, T)$ satisfies φ precisely if $(\mathcal{A}, T') \vDash \varphi$ for every $T' \subseteq T$ with $|T'| = k$. However, the notions of k-coherence (for $k \geqslant 2$) and quasi-flatness are orthogonal, as we show next.

Proposition 5.3. *There are FO(dep, $\sim$)-formulas $\psi, \varphi_1, \varphi_2, \ldots$ such that ψ is 2-coherent but not quasi-flat, and for each k, φ_k is quasi-flat but not k-coherent.*

Proof. The constancy atom $\psi := \mathrm{dep}(x)$ is 2-coherent but undefinable in FO$(\sim)$ [38], which includes all formulas in normal form, hence $\mathrm{dep}(x)$ is not quasi-flat. For φ_k, assume pairwise distinct constants $c_1, \ldots, c_{k+1}$ and let

$$\varphi_k := \bigovee_{\substack{I\subseteq[k+1]\\|I|=k}} \bigvee_{i\in I} (x = c_i).$$

The formula φ_k is not k-coherent, since in the team $T = \{s_1, \ldots, s_{k+1}\}$, $s_i(x) = c_i$, it is false, but true in all size k subteams of T. It is however quasi-flat since each $\bigvee_{i \in I}(x = c_i)$ is flat for all I. □

5.1.2 An upper bound for FO(∼)

Next, we show that the satisfiability problem of FO(∼) can be reduced to that of mere FO-sentences. Intuitively, this is because an existential literal of the form $E\beta$, which states that the team contains some assignment satisfying β, can be simulated by existential quantifiers $\exists x_1 \cdots \exists x_n \beta$. This leads to the next translation.

Theorem 5.4. *For every σ-FO(∼)-formula φ in normal form there is a polynomial time computable σ-FO-sentence ψ such that $\mathcal{A} \vDash \psi$ if and only if there is a team T such that $(\mathcal{A}, T) \vDash \varphi$. Moreover, T can be assumed to have size $|T| \leqslant |\varphi|$.*

Proof. Suppose $\varphi \in \sigma\text{-FO}(\sim)$ is of the form

$$\varphi = \bigovee_{i \in I} \left(\alpha_i \wedge \bigwedge_{j \in J_i} E\beta_{i,j} \right)$$

for $\alpha_i, \beta_{i,j} \in \text{FO}$. Let $\text{Fr}(\varphi) = \{x_1, \ldots, x_n\}$. Then we define the sentence

$$\psi := \bigvee_{i \in I} \bigwedge_{j \in J_i} \exists x_1 \cdots \exists x_n (\alpha_i \wedge \beta_{i,j})$$

which is obviously polynomial time computable.

Now we prove the correctness as stated in the theorem. Suppose that there is $i \in I$ such that $\mathcal{A} \vDash \exists x_1 \cdots \exists x_n (\alpha_i \wedge \beta_{i,j})$ for all $j \in J_i$, and for each j let $s_j : \{x_1, \ldots, x_n\} \to \mathcal{A}$ be the assignment witnessing the existential quantifiers. Then $(\mathcal{A}, \{s_j\}) \vDash \alpha_i \wedge \beta_{i,j}$ in team semantics. Now by union closure (Proposition 2.11), $(\mathcal{A}, \{s_j \mid j \in J_i\}) \vDash \alpha_i$. Also, by upward closure of E-formulas (as their negations are downward closed (Proposition 2.11)), $(\mathcal{A}, \{s_j \mid j \in J_i\}) \vDash E\beta_{i,j}$. Consequently, with the team $T = \{s_j \mid j \in J_i\}$ we prove the first direction.

Next, assume some team T such that $(\mathcal{A}, T) \vDash \varphi$ via $i \in I$, where $(\mathcal{A}, T) \vDash \alpha_i$ and $(\mathcal{A}, T) \vDash E\beta_{i,j}$ for all $j \in J_i$. The latter is witnessed by a family $(s_j)_{j \in J_i}$ of assignments $s_j \in T$. Then for each j we have $(\mathcal{A}, \{s_j\}) \vDash \beta_{i,j}$ and additionally $(\mathcal{A}, \{s_j\}) \vDash \alpha_i$ by downward closure. Hence the assignment s_j witnesses $\mathcal{A} \vDash \exists x_1 \cdots x_n (\alpha_i \wedge \beta_{i,j})$. □

Corollary 5.5. *For every σ-FO(∼)-formula φ in normal form there is a polynomial time computable σ-FO-sentence ψ such that φ is satisfiable in team semantics iff ψ is classically satisfiable.*

We are now in the position to conclude the overall complexity of FO(∼).

Theorem 5.6. *The problem SAT(FO(∼)) is Π^0_1-complete. The problem VAL(FO(∼)) is Σ^0_1-complete.*

Proof. The lower bounds already hold for FO, so we argue for the upper bounds. For those, recall that classical satisfiability resp. validity of FO are complete for Π^0_1 resp. Σ^0_1.

Given $\varphi \in \mathsf{FO}(\sim)$, we compute a normal form ψ of φ (Theorem 5.1). By the previous corollary, we can then compute a formula ψ' that is classically satisfiable iff ψ is satisfiable in team semantics. Hence φ is satisfiable in team semantics iff ψ' is classically satisfiable, which reduces the problem $\mathrm{SAT}(\mathsf{FO}(\sim))$ to $\mathrm{SAT}(\mathsf{FO})$. As a consequence, $\mathrm{SAT}(\mathsf{FO}(\sim))$ is in Π^0_1. For VAL, φ is valid iff $\sim\varphi$ is unsatisfiable, which is the complement of the previous problem, so $\mathrm{VAL}(\mathsf{FO}(\sim))$ is in Σ^0_1. □

The approach of a translation to normal form is flexible enough to even yield decidability results for fragments of $\mathsf{FO}(\sim)$, as is shown in the next subsections.

5.1.3 Two-variable logic

The logic FO^n, called *n-variable logic*, is the fragment of FO that contains the formulas α such that $|\mathsf{Var}(\alpha)| \leqslant n$, i.e., α contains at most n distinct variables (although it can contain any variable arbitrarily often). By a famous theorem by Mortimer [116], FO^2 is decidable. The result was subsequently improved to a sharp complexity bound:

Theorem 5.7 (Grädel et al. [48])**.** *If σ is a relational vocabulary, then $\mathrm{SAT}(\sigma\text{-}\mathsf{FO}^2)$ is complete for* NExpTime*.*

For three or more variables, the problem again becomes undecidable [12].

We consider a similar fragment in team semantics. We define *n-variable team logic*, $\sigma\text{-}\mathsf{FO}^n(\sim)$, as the fragment of $\sigma\text{-}\mathsf{FO}(\sim)$ that contains all formulas φ such that $|\mathsf{Var}(\varphi)| \leqslant n$. We notice that $\sigma\text{-}\mathsf{FO}^n(\sim)$ admits the same normal forms as $\sigma\text{-}\mathsf{FO}(\sim)$:

Theorem 5.8. *For every $\varphi \in \sigma\text{-}\mathsf{FO}^n(\sim)$, there exists an equivalent formula $\psi \in \sigma\text{-}\mathsf{FO}^n(\sim)$ in normal form that is computable from φ in time $\exp_{\mathcal{O}(|\varphi|)}(1)$.*

Proof. The proof of Theorem 5.1 does not introduce any new variables. □

Theorem 5.9. *For every $\sigma\text{-}\mathsf{FO}^n(\sim)$-formula φ in normal form there is a polynomial time computable $\sigma\text{-}\mathsf{FO}^n$-sentence ψ such that φ is satisfiable in team semantics iff ψ is classically satisfiable.*

Proof. If φ is in $\sigma\text{-}\mathsf{FO}^n(\sim)$, then the formula ψ constructed in the proof of Theorem 5.4 itself also is in $\sigma\text{-}\mathsf{FO}^n$, since again no new variables are added. □

Theorem 5.10. *If σ is a relational vocabulary, then $\mathrm{SAT}(\sigma\text{-}\mathsf{FO}^2(\sim))$ and $\mathrm{VAL}(\sigma\text{-}\mathsf{FO}^2(\sim))$ are in* Tower(poly).

Proof. Completely analogous to Theorem 5.6. For the sake of self-containedness, we sketch the algorithm for SAT.

Given $\varphi \in \sigma\text{-}\mathsf{FO}^2(\sim)$, first translate φ to an equivalent normal form φ' as in Corollary 5.5. This takes time $\exp_{\mathcal{O}(|\varphi|)}(1)$. Next, translate φ' to a formula $\varphi'' \in \sigma\text{-}\mathsf{FO}^2$ that is classically satisfiable iff φ' is satisfiable in team semantics. This takes time polynomial in $|\varphi'|$, so φ'' itself is computable again in time $\exp_{\mathcal{O}(|\varphi|)}(1)$.

Finally, the problem $\mathrm{SAT}(\sigma\text{-}\mathsf{FO}^2)$ is in NExpTime [48] and consequently in 2ExpTime. So to decide whether φ is satisfiable, we call the decision algorithm for FO^2 as a subroutine on φ''. The overall runtime is still $\exp_{\mathcal{O}(|\varphi|)}(1)$. As $\mathsf{FO}^2(\sim)$ is closed under negation, the algorithm for VAL is analogous. □

The classical two-variable fragment, FO^2, is decidable in NExpTime, and hence elementary. However, this complexity is dwarfed by the cost of the translation into normal form. So one could say that the complexity of the underlying classical logic vanishes inside the Tower(poly) upper bound.

5.1.4 The guarded fragment

Another well-behaved fragment of first-order logic is the *guarded fragment* by Andréka et al. [4]. It is called guarded because quantification can only be performed relative to some atomic formula, called the *guard* of the quantifier. Guarded σ-FO-formulas are inductively defined:

- Any atomic σ-FO-formula is guarded.
- If α, β are guarded σ-FO-formulas, then so are $\alpha \wedge \beta$, $\alpha \vee \beta$, and $\neg\alpha$.
- If α is a guarded σ-FO-formula and γ is an atomic σ-FO-formula, then $\exists x(\gamma \wedge \alpha)$ and $\forall x(\gamma \rightarrow \alpha)$ are guarded σ-FO-formulas, provided that $\mathsf{Fr}(\alpha) \subseteq \mathsf{Var}(\gamma)$, i.e., every variable free in α occurs in the atomic formula γ.

The logic σ-GF is now the fragment of all guarded σ-FO-formulas.

Theorem 5.11 (Grädel [45])**.** *Let σ be a relational vocabulary. Then* $\mathrm{SAT}(\sigma\text{-}\mathsf{GF})$ *is complete for* 2ExpTime, *and for every* $n \geqslant 2$, *the problem* $\mathrm{SAT}(\sigma\text{-}\mathsf{GF}^n)$ *is complete for* ExpTime.

Next, we propose a definition of a guarded fragment in team logic, which we call $\mathsf{GF}(\sim)$. Compared to the classical logic GF, there are only minor changes in the definition:

- Any atomic σ-FO(∼)-formula is guarded.
- If φ, ψ are guarded σ-FO(∼)-formulas, then so are $\varphi \wedge \psi$, $\varphi \vee \psi$, $\sim\varphi$ and $\neg\varphi$.
- If φ is a guarded σ-FO(∼)-formula and γ is an atomic σ-FO-formula, then $\exists x(\gamma \wedge \varphi)$ and $\forall x(\gamma \hookrightarrow \varphi)$ are guarded σ-FO(∼)-formulas, provided that $\mathsf{Fr}(\varphi) \subseteq \mathsf{Var}(\gamma)$, i.e., every variable free in φ occurs in the atomic formula γ.

Recall that $\alpha \hookrightarrow \varphi$ is defined as $\neg\alpha \vee (\alpha \wedge \varphi)$. There are technical reasons why this change in the definition of guardedness is necessary, which will become clear later.

Definition 5.12. The logic $\sigma\text{-}\mathsf{GF}(\sim)$ is the fragment of all guarded σ-FO(∼)-formulas. The logic $\sigma\text{-}\mathsf{GF}^n(\sim)$ is the fragment of all formulas in $\sigma\text{-}\mathsf{GF}(\sim) \cap \sigma\text{-}\mathsf{FO}^n(\sim)$.

Distributive laws for $\varovee$ over $\exists, \forall$:	
$\exists x[\gamma](\theta_1 \varovee \theta_2) \equiv (\exists x[\gamma]\theta_1) \varovee (\exists x[\gamma]\theta_2)$	$\forall x[\gamma](\theta_1 \varovee \theta_2) \equiv (\forall x[\gamma]\theta_1) \varovee (\forall x[\gamma]\theta_2)$
Distributive laws for $\wedge$ over $\exists, \forall$:	
$\exists x[\gamma](\alpha \wedge \bigwedge_{i \in I} \mathsf{E}\beta_i) \equiv (\exists x[\gamma]\alpha) \wedge \bigwedge_{i \in I} \mathsf{E}\exists x[\gamma](\alpha \wedge \beta_i)$	
$\forall x[\gamma](\alpha \wedge \bigwedge_{i \in I} \mathsf{E}\beta_i) \equiv (\forall x[\gamma]\alpha) \wedge \bigwedge_{i \in I} \mathsf{E}\exists x[\gamma](\alpha \wedge \beta_i)$	

Table 5.2: Distributive laws for $\mathsf{GF}(\sim)$

As before, we often omit σ.

Next, we present a result on the complexity of satisfiability of the guarded fragment. It behaves like two-variable logic in that the complexity becomes non-elementary in team semantics due to the translation to normal form.

In the following, we also use the shorthands $\exists x[\gamma]\varphi$ and $\forall x[\gamma]\varphi$ for $\exists x(\gamma \wedge \varphi)$ and $\forall x(\gamma \hookrightarrow \varphi)$, where γ is the guard.

Theorem 5.13. *For every $\varphi \in \sigma\text{-}\mathsf{GF}(\sim)$ there is an equivalent $\sigma\text{-}\mathsf{GF}(\sim)$-formula ψ in normal form that is computable in time $\exp_{\mathcal{O}(|\varphi|)}(1)$.*

Proof. The proof is similar to the non-guarded case (Theorem 5.1). For this, we prove variants of the laws in Table 5.1 for guarded formulas. For laws that swap only $\varovee$ and $\wedge/\vee$, this is clear, as these preserve guardedness. Let us focus on the laws that concern quantifiers. Those are depicted in Table 5.2, and we prove them as follows.

- Distribute $\exists x[\gamma]$ over $\varovee$:

$$
\begin{aligned}
& \exists x[\gamma](\theta_1 \varovee \theta_2) \\
\equiv\ & \exists x(\gamma \wedge (\theta_1 \varovee \theta_2)) && (\text{def. } \exists x[\gamma]) \\
\equiv\ & \exists x((\gamma \wedge \theta_1) \varovee (\gamma \wedge \theta_2)) && (\wedge \text{ distributes over } \varovee) \\
\equiv\ & \exists x(\gamma \wedge \theta_1) \varovee \exists x(\gamma \wedge \theta_2) && (\exists \text{ distributes over } \varovee) \\
\equiv\ & \exists x[\gamma]\theta_1 \varovee \exists x[\gamma]\theta_2 && (\text{def. } \exists x[\gamma])
\end{aligned}
$$

  Clearly $\mathsf{Fr}(\theta_1) \cup \mathsf{Fr}(\theta_2) \subseteq \mathsf{Var}(\gamma)$ if and only if $\mathsf{Fr}(\theta_1 \varovee \theta_2) \subseteq \mathsf{Var}(\gamma)$. The cases below behave analogously.

- Distribute $\forall x[\gamma]$ over $\varovee$:

$$
\begin{aligned}
& \forall x[\gamma](\theta_1 \varovee \theta_2) \\
\equiv\ & \forall x(\gamma \hookrightarrow (\theta_1 \varovee \theta_2)) && (\text{def. } \forall x[\gamma]) \\
\equiv\ & \forall x(\neg\gamma \vee (\gamma \wedge (\theta_1 \varovee \theta_2))) && (\text{def. } \hookrightarrow) \\
\equiv\ & \forall x(\neg\gamma \vee ((\gamma \wedge \theta_1) \varovee (\gamma \wedge \theta_2))) && (\wedge \text{ distributes over } \varovee) \\
\equiv\ & \forall x((\neg\gamma \vee (\gamma \wedge \theta_1)) \varovee (\neg\gamma \vee (\gamma \wedge \theta_2))) && (\vee \text{ distributes over } \varovee) \\
\equiv\ & \forall x((\gamma \hookrightarrow \theta_1) \varovee (\gamma \hookrightarrow \theta_2)) && (\text{def. } \hookrightarrow)
\end{aligned}
$$

$$\equiv \forall x(\gamma \hookrightarrow \theta_1) \otimes \forall x(\gamma \hookrightarrow \theta_2) \qquad (\forall \text{ distributes over } \otimes)$$
$$\equiv \forall x[\gamma]\theta_1 \otimes \forall x[\gamma]\theta_2 \qquad (\text{def. } \forall x[\gamma])$$

- Distribute $\exists x[\gamma]$ over $\wedge$:

$$\begin{aligned}
&\exists x[\gamma](\alpha \wedge \bigwedge_{i \in I} E\beta_i) \\
&\equiv \exists x(\gamma \wedge (\alpha \wedge \bigwedge_{i \in I} E\beta_i)) && (\text{def. } \exists x[\gamma]) \\
&\equiv \exists x((\gamma \wedge \alpha) \wedge \bigwedge_{i \in I} E\beta_i) && (\vee \text{ is associative}) \\
&\equiv \exists x(\gamma \wedge \alpha) \wedge \bigwedge_{i \in I} E\exists x((\gamma \wedge \alpha) \wedge \beta_i) && (\text{by Table 5.1}) \\
&\equiv \exists x[\gamma]\alpha \wedge \bigwedge_{i \in I} E\exists x[\gamma](\alpha \wedge \beta_i) && (\text{def. } \exists x[\gamma])
\end{aligned}$$

- Distribute $\forall x[\gamma]$ over $\wedge$:

$$\begin{aligned}
&\forall x[\gamma](\alpha \wedge \bigwedge_{i \in I} E\beta_i) \\
&\equiv \forall x(\gamma \hookrightarrow (\alpha \wedge \bigwedge_{i \in I} E\beta_i)) && (\text{def. } \forall x[\gamma]) \\
&\equiv \forall x((\gamma \hookrightarrow \alpha) \wedge \bigwedge_{i \in I} (\gamma \hookrightarrow E\beta_i)) && (\text{as } \gamma \hookrightarrow (\theta_1 \wedge \theta_2) \equiv (\gamma \hookrightarrow \theta_1) \wedge (\gamma \hookrightarrow \theta_2)) \\
&\equiv \forall x((\gamma \hookrightarrow \alpha) \wedge \bigwedge_{i \in I} E(\gamma \wedge \beta_i)) && (\text{as } \gamma \hookrightarrow E\beta \equiv E(\gamma \wedge \beta)) \\
&\equiv \forall x(\gamma \hookrightarrow \alpha) \wedge \bigwedge_{i \in I} E\exists x((\gamma \hookrightarrow \alpha) \wedge (\gamma \wedge \beta_i)) && (\text{by Table 5.1}) \\
&\equiv \forall x(\gamma \hookrightarrow \alpha) \wedge \bigwedge_{i \in I} E\exists x(\gamma \wedge (\alpha \wedge \beta_i)) && (\text{as } (\gamma \hookrightarrow \alpha) \wedge \gamma \text{ implies } \alpha) \\
&\equiv \forall x[\gamma]\alpha \wedge \bigwedge_{i \in I} E\exists x[\gamma](\alpha \wedge \beta_i) && (\text{def. } \forall x[\gamma]) \quad \square
\end{aligned}$$

In the above proof of the final distributive law for $\forall$ and $\wedge$, it also becomes clear why the guard of $\forall$ needs to be connected via $\hookrightarrow$ and not $\rightarrow$. Suppose we define $\forall x[\gamma]\varphi$ as $\forall x(\gamma \rightarrow \varphi) \equiv \forall x(\neg\gamma \vee \varphi)$ instead. Then we have the equivalences

$$\begin{aligned}
&\forall x[\gamma](\alpha \wedge \bigwedge_{i \in I} E\beta_i) \\
&\equiv \forall x(\neg\gamma \vee (\alpha \wedge \bigwedge_{i \in I} E\beta_i))
\end{aligned}$$

$$\equiv \forall x((\neg\gamma \vee \alpha) \wedge \bigwedge_{i \in I} E(\alpha \wedge \beta_i))$$
$$\equiv \forall x[\gamma]\alpha \wedge \bigwedge_{i \in I} \forall x E(\alpha \wedge \beta_i)$$
$$\equiv \forall x[\gamma]\alpha \wedge \bigwedge_{i \in I} E\exists x(\alpha \wedge \beta_i)$$

but here the final formula may be not guarded, since we only know that γ, but not α, guards β_i.

Next, we come to the step analogous to Theorem 5.4 where a formula in normal form can be simulated by classical sentences. The approach we used previously for $\mathsf{FO}(\sim)$ and $\mathsf{FO}^2(\sim)$ does not simply go through for the guarded fragment, because the resulting formula is not guarded due to the additionally introduced quantifiers. To circumvent this, we tweak the proof a bit as follows.

Theorem 5.14. *Let σ be a relational vocabulary. For every σ-$\mathsf{GF}(\sim)$-formula φ in normal form there is a relational vocabulary $\sigma' \supseteq \sigma$ and a polynomial time computable σ'-GF-sentence ψ such that φ is satisfiable in team semantics iff ψ is classically satisfiable.*

Proof. We change the sentence ψ from the proof of Theorem 5.4 from

$$\psi := \bigvee_{i \in I} \bigwedge_{j \in J_i} \exists x_1 \cdots \exists x_n (\alpha_i \wedge \beta_{i,j}).$$

to

$$\psi' := \bigvee_{i \in I} \bigwedge_{j \in J_i} \exists x_1 [R_1 x_1]\, \exists x_2 [R_2 x_1 x_2] \cdots \exists x_n [R_n x_1 \cdots x_n]\, (\alpha_i \wedge \beta_{i,j}),$$

where each R_i is a fresh i-ary predicate not in σ. The formulas ψ and ψ' are clearly satisfiability equivalent, as a model of ψ' can easily be obtained from one of ψ by interpreting each R_i as the full i-ary relation. Moreover, as we assumed that φ is guarded, so must be all α_i, $\beta_{i,j}$, and hence ψ' is guarded as well. □

Theorem 5.15. *Let σ be a relational vocabulary. Then* $\mathrm{SAT}(\sigma\text{-}\mathsf{GF}(\sim))$ *and* $\mathrm{VAL}(\sigma\text{-}\mathsf{GF}(\sim))$ *are in* $\mathrm{TOWER}(\mathrm{poly})$.

Proof. Completely analogous to the approach for $\mathsf{FO}^2(\sim)$ (Theorem 5.10), using the corresponding translation to GF (Theorem 5.14 and Theorem 5.13) instead of FO^2. □

Before we come to the lower bounds for the above logics, we will consider the model checking problem in the next section.

5.1.5 Model checking

Next, we consider the model checking problem, and present an algorithm.

Algorithm: $\mathrm{MC}(\mathcal{A}, \mathsf{T}, \varphi)$ for $\varphi \in \mathsf{FO}(\sim)$, a σ-structure $\mathcal{A}$, and a team T in $\mathcal{A}$.

1 $\mathsf{T} \leftarrow \mathsf{T}\upharpoonright\mathsf{Fr}(\varphi)$;
2 **if** $\varphi = \alpha$ *is a first-order formula* **then**
3 | universally guess $s \in \mathsf{T}$; **if** $(\mathcal{A}, s) \vDash \alpha$ **then return** `true` **else return** `false`;
4 **end**
5 **else if** $\varphi = \sim\alpha$ *for a first-order formula* α **then**
6 | existentially guess $s \in \mathsf{T}$; **if** $(\mathcal{A}, s) \nvDash \alpha$ **then return** `true` **else return** `false`;
7 **end**
8 **else if** $\varphi = \sim\sim\psi$ **then return** $\mathrm{MC}(\mathcal{A}, \mathsf{T}, \psi)$;
9 **else if** $\varphi = \psi_1 \wedge \psi_2$ **then** universally guess $i \in \{1, 2\}$ and **return** $\mathrm{MC}(\mathcal{A}, \mathsf{T}, \psi_i)$;
10 **else if** $\varphi = \sim(\psi_1 \wedge \psi_2)$ **then**
11 | existentially guess $i \in \{1, 2\}$ and **return** $\mathrm{MC}(\mathcal{A}, \mathsf{T}, \sim\psi_i)$;
12 **end**
13 **else if** $\varphi = \psi_1 \vee \psi_2$ **then**
14 | existentially guess $S_1, S_2 \subseteq \mathsf{T}$ such that $S_1 \cup S_2 = \mathsf{T}$
15 | universally guess $i \in \{1, 2\}$ and **return** $\mathrm{MC}(\mathcal{A}, S_i, \psi_i)$;
16 **end**
17 **else if** $\varphi = \sim(\psi_1 \vee \psi_2)$ **then**
18 | universally guess $S_1, S_2 \subseteq \mathsf{T}$ such that $S_1 \cup S_2 = \mathsf{T}$
19 | existentially guess $i \in \{1, 2\}$ and **return** $\mathrm{MC}(\mathcal{A}, S_i, \sim\psi_i)$;
20 **end**
21 **else if** $\varphi = \forall x\psi$ **then return** $\mathrm{MC}(\mathcal{A}, \mathsf{T}^x_{\mathcal{A}}, \psi)$;
22 **else if** $\varphi = \sim\forall x\psi$ **then return** $\mathrm{MC}(\mathcal{A}, \mathsf{T}^x_{\mathcal{A}}, \sim\psi)$;
23 **else if** $\varphi = \exists x\psi$ **then** existentially guess $f \colon \mathsf{T} \to \wp^+(\mathcal{A})$; **return** $\mathrm{MC}(\mathcal{A}, \mathsf{T}^x_f, \psi)$;
24 **else if** $\varphi = \sim\exists x\psi$ **then** universally guess $f \colon \mathsf{T} \to \wp^+(\mathcal{A})$; **return** $\mathrm{MC}(\mathcal{A}, \mathsf{T}^x_f, \sim\psi)$;

Algorithm 1: Algorithm for $\mathrm{MC}(\mathsf{FO}(\sim))$

Proposition 5.16. *$\mathrm{MC}(\mathsf{FO}(\sim))$ is decidable in time $2^{n^{\mathcal{O}(1)}}$ and with $\mathcal{O}(n)$ alternations.*

In other words, $\mathrm{MC}(\mathsf{FO}(\sim)) \in \mathrm{ATIME\text{-}ALT}(\exp, \mathrm{poly})$.

Proof. Algorithm 1 decides $\mathrm{MC}(\mathsf{FO}(\sim))$. Every run of the algorithm has at most $|\varphi|$ recursive calls, since the formula passed to the **return**-statements always is strictly shorter than the respective input formula. Each call has at most two alternations, yielding at most $2|\varphi|$ alternations in total. Lines 3 and 6 run in time polynomial in $\log|\mathsf{T}| + 2^{|\mathcal{A}||\alpha|}$, since first-order model checking is in PSPACE (cf. [47]). Lines 14 and 18 are also polynomial in $|\mathsf{T}|$. Finally, the lines 23 and 24 run in time polynomial in $|\mathsf{T}||\mathcal{A}|$, which is the maximal size of a supplementing function f.

The structure $\mathcal{A}$ is clearly unchanged in every recursive call, and $|\varphi|$ strictly decreases. As the runtime in every call depends polynomially on T, for an overall exponential runtime it suffices to show that the team T can only grow exponentially with respect to the original input, call it $(\mathcal{A}, \mathsf{T}_0, \varphi_0)$. This is ensured by line 1, which exploits that φ

is local (Proposition 2.14). Hence T is a team with domain $\operatorname{dom} T \subseteq \mathsf{Var}(\varphi_0)$ and so is bounded by $\mathcal{O}\left(|\mathcal{A}|^{|\varphi_0|}\right) = \mathcal{O}\left(2^{\log|\mathcal{A}|\cdot|\varphi_0|}\right)$. □

Corollary 5.17. $\mathrm{MC}(\mathsf{FO}^n(\sim))$ *is in* PSPACE *for all finite* n.

Proof. Since PSPACE = AP [13], we modify the previous proof and show that the runtime of each recursive call is polynomial. If the number of variables in the formula bounded a priori, then lines 3 and 6 run in deterministic time polynomial in $\log|T| + |\mathcal{A}| + |\alpha|$, as model checking for FO^n is in P for every n [137]. Moreover, lines 14 and 18 resp. 23 and 24 run in polynomial time since the size of T (and hence of S_1, S_2 and f) is always polynomial in the original input, as $|T| \in \mathcal{O}\left(|\mathcal{A}|^{\mathsf{Var}(\varphi_0)}\right) = |\mathcal{A}|^{\mathcal{O}(1)}$. □

5.2 A standard translation for team semantics

The well-known *standard translation* embeds modal logic ML into FO^2 with the relational vocabulary $\sigma_M = (R, Q_{p\in\mathsf{Prop}})$, where the binary relation R represents the edges in a Kripke structure, and the Q_p are unary relations containing worlds that satisfy the variable $p \in \mathsf{Prop}$. The standard translation of an ML-formula α is denoted by $\mathsf{st}_x(\alpha)$ or $\mathsf{st}_y(\alpha)$ depending on its free variable, which is either x or y. It is defined by mutual recursion between x and y:

$$\begin{aligned} \mathsf{st}_x(p) &:= Q_p x \text{ for } p \in \mathsf{Prop} & \mathsf{st}_x(\neg\alpha) &:= \neg\mathsf{st}_x(\alpha) \\ \mathsf{st}_x(\Box\alpha) &:= \forall y\,(Rxy \to \mathsf{st}_y(\alpha)) & \mathsf{st}_x(\alpha\land\beta) &:= \mathsf{st}_x(\alpha)\land\mathsf{st}_x(\beta) \\ \mathsf{st}_x(\Diamond\alpha) &:= \exists y\,(Rxy \land \mathsf{st}_y(\alpha)) & \mathsf{st}_x(\alpha\lor\beta) &:= \mathsf{st}_x(\alpha)\lor\mathsf{st}_x(\beta), \end{aligned}$$

with $\mathsf{st}_y(\alpha)$ defined in a symmetric fashion. The corresponding *first-order interpretation* of a Kripke structure $\mathcal{K} = (W, R', V)$ is the σ_M-structure $\ulcorner\mathcal{K}\urcorner$ with domain W and interpretations $R^{\ulcorner\mathcal{K}\urcorner} = R'$ and $Q_p^{\ulcorner\mathcal{K}\urcorner} = V(p)$ of σ_M. For a world $w \in W$ and $x \in \mathsf{Var}$, define the corresponding first-order assignment $w^x \colon \{x\} \to W$ by $w^x(x) = w$.

The next theorem is standard:

Theorem 5.18. *Let* $(\mathcal{K}, w)$ *be a pointed Kripke structure and* $\alpha \in \mathsf{ML}$*. Then* $(\mathcal{K}, w) \vDash \alpha$ *if and only if* $(\ulcorner\mathcal{K}\urcorner, w^x) \vDash \mathsf{st}_x(\alpha)$.

Proof. By straightforward induction (cf. Blackburn and van Benthem [9]). □

In this section, we lift this result to team semantics and translate $\mathsf{ML}(\sim)$ to $\mathsf{FO}^2(\sim)$. On the model side, the first-order interpretation of a team T in a Kripke structure becomes $T^x := \{\, w^x \mid w \in T \,\}$. In other words, the team of worlds becomes a team of assignments of x to these worlds.

The standard translation for $\mathsf{ML}(\sim)$ now extends that of ML by the final $\sim$-case. For similar reasons as for the guarded fragment, we also change the clause for $\Box$ to utilize $\hookrightarrow$ instead of $\to$. Recall that $\alpha \hookrightarrow \varphi \equiv \neg\alpha \lor (\alpha \land \varphi)$.

$$\mathsf{st}_x(p) := Q_p x \text{ for } p \in \mathsf{Prop} \qquad\qquad \mathsf{st}_x(\neg\varphi) := \neg\mathsf{st}_x(\varphi)$$

$$\mathrm{st}_x(\Box\varphi) := \forall y\,(Rxy \hookrightarrow \mathrm{st}_y(\varphi)) \qquad \mathrm{st}_x(\varphi \wedge \psi) := \mathrm{st}_x(\varphi) \wedge \mathrm{st}_x(\psi)$$
$$\mathrm{st}_x(\Diamond\varphi) := \exists y\,(Rxy \wedge \mathrm{st}_y(\varphi)) \qquad \mathrm{st}_x(\varphi \vee \psi) := \mathrm{st}_x(\varphi) \vee \mathrm{st}_x(\psi)$$
$$\mathrm{st}_x({\sim}\varphi) := {\sim}\mathrm{st}_x(\varphi),$$

with $\mathrm{st}_y(\varphi)$ again defined symmetrically. Here, the naive translation of $\Box\varphi$ to

$$\forall y\,(Rxy \to \mathrm{st}_y(\varphi)) \equiv \forall y\,(\neg Rxy \vee \mathrm{st}_y(\varphi))$$

would be unsound under team semantics, so using $\hookrightarrow$ instead of $\to$ is crucial. (The similar translation by Yang [141] for modal dependence logic did not require this tweak since this logic has downward closure.)

Theorem 5.19. *For all Kripke structures with team* $(\mathcal{K}, T)$, *all* $\varphi \in \mathrm{ML}(\sim)$ *and* $x \in \mathrm{Var}$ *it holds* $(\mathcal{K}, T) \vDash \varphi$ *if and only if* $(\ulcorner\mathcal{K}\urcorner, T^x) \vDash \mathrm{st}_x(\varphi)$.

Proof. Proof by induction on φ. Let $\mathcal{K} = (W, R, V)$. We omit $\mathcal{K}$ and $\ulcorner\mathcal{K}\urcorner$ to the left of $\vDash$.

- $\varphi \in \mathrm{ML}$: We have $T \vDash \varphi$ iff $\forall w \in T\colon w \vDash \varphi$ due to flatness, which by Theorem 5.18 is equivalent to $\forall w^x \in T^x\colon w^x \vDash \mathrm{st}_x(\varphi)$. However, as $\mathrm{st}_x(\varphi) \in \mathrm{FO}$, the latter is equivalent to $T^x \vDash \mathrm{st}_x(\varphi)$ again by flatness.

- $\varphi = \psi \wedge \theta$, $\varphi = \neg\psi$ and $\varphi = {\sim}\psi$ are clear by induction hypothesis.

- $\varphi = \psi \vee \theta$: Suppose $T \vDash \psi \vee \theta$. Then $T = S \cup U$ such that $S \vDash \psi$ and $U \vDash \theta$. By induction hypothesis, $S^x \vDash \mathrm{st}_x(\psi)$ and $U^x \vDash \mathrm{st}_x(\theta)$. As $S \cup U = T$, clearly $S^x \cup U^x = T^x$. As a consequence, $T^x \vDash \mathrm{st}_x(\psi) \vee \mathrm{st}_x(\theta) = \mathrm{st}_x(\psi \vee \theta)$.

  For the other direction, suppose $T^x \vDash \mathrm{st}_x(\psi \vee \theta) = \mathrm{st}_x(\psi) \vee \mathrm{st}_x(\theta)$ by the means of some subteams $S' \cup U' = T^x$ such that $S' \vDash \mathrm{st}_x(\psi)$ and $U' \vDash \mathrm{st}_x(\theta)$. Then $S' = S^x$ and $U' = U^x$ for some suitably chosen $S, U \subseteq T$. By induction hypothesis, $S \vDash \psi$ and $U \vDash \theta$. In order to prove $T \vDash \psi \vee \theta$, it remains to show $T \subseteq S \cup U$. For this assume $w \in T$. As then $w^x \in T^x$, as least one of $w^x \in S'$ or $w^x \in U'$ holds. But then $w \in S$ or $w \in U$.

- $\varphi = \Box\psi$: Consider the duplicating team $(T^x)^y_W$ of T^x. We define subteams S and U of $(T^x)^y_W$ as follows: S contains all "outgoing edges": $S := \{s \in (T^x)^y_W \mid s(y) \in Rs(x)\}$. By contrast, U contains all "non-edges": $U := \{s \in (T^x)^y_W \mid s(y) \notin Rs(x)\}$. Then clearly $(T^x)^y_W = S \cup U$, $S \vDash Rxy$ and $U \vDash \neg Rxy$. Moreover, the above division of $(T^x)^y_W$ into S and U is the *only* possible splitting of $(T^x)^y_W$ such that $S \vDash Rxy$ and $U \vDash \neg Rxy$. (This is the step that requires $\hookrightarrow$, not $\to$, in the standard translation.)

  By the induction hypothesis, $T \vDash \Box\psi$ iff $RT \vDash \psi$ iff $(RT)^y \vDash \mathrm{st}_y(\psi)$. Moreover, by the above argument, $T^x \vDash \mathrm{st}_x(\Box\psi)$ iff $T^x \vDash Rxy \hookrightarrow \mathrm{st}_y(\psi)$ iff $S \vDash \mathrm{st}_y(\psi)$. Consequently, it suffices to show that $(RT)^y$ and S agree on $\mathrm{st}_y(\psi)$. This follows from locality (Proposition 2.14), since

  $$(RT)^y = \{v^y \mid v \in Rw, w \in T\}$$

$$= \{s \mid s\colon \{y\} \to W, s(y) \in Rw, w \in T\} \qquad \text{(def. } \nu^y\text{)}$$
$$= \{s\restriction\{y\} \mid s\colon \{x,y\} \to W, s(y) \in Rs(x), s(x) \in T\}$$
$$= \{s\restriction\{y\} \mid s \in (T^x)^y_W, s(y) \in Rs(x)\}$$
$$= S\restriction\{y\}. \qquad \text{(def. } S\text{)}$$

- $\varphi = \Diamond\psi$: For the first direction, suppose $T \vDash \Diamond\psi$, i.e., $S \vDash \psi$ for some successor team S of T. We have to prove $T^x \vDash \exists y\,(Rxy \wedge st_y(\psi))$, that is, find a supplementing function $f\colon T^x \to \wp^+(W)$ such that $(T^x)^y_f \vDash Rxy \wedge st_y(\psi)$.

  By induction hypothesis, $S^y \vDash st_y(\psi)$. We define f on T^x as follows. Given $w^x \in T^x$, let $f(w^x) := Rw \cap S$. Then $f(w^x)$ is non-empty for each w, as S is a successor team, so $f\colon T^x \to \wp^+(W)$ as required. Moreover, $(T^x)^y_f \vDash Rxy$. To prove that $(T^x)^y_f \vDash st_y(\psi)$, we again apply locality (Proposition 2.14) and show that $(T^x)^y_f\restriction\{y\} = S^y$:

  $$(T^x)^y_f\restriction\{y\} = \{s^y_v \mid s \in T^x, v \in f(s)\}\restriction\{y\} \qquad \text{(def. supplementing team)}$$
  $$= \{s^y_v\restriction\{y\} \mid s \in T^x, v \in f(s)\} \qquad \text{(def. } \restriction\text{)}$$
  $$= \{s\colon \{y\} \to \{v\} \mid \exists s' \in T^x, v \in f(s')\}$$
  $$= \{s\colon \{y\} \to \{v\} \mid \exists w \in T, v \in Rw \cap S\} \qquad \text{(def. } f\text{)}$$
  $$= \{s\colon \{y\} \to \{v\} \mid v \in S\} \qquad \text{(since } S = \textstyle\bigcup_{w\in T}(Rw \cap S)\text{)}$$
  $$= S^y$$

  For the other direction, suppose $T^x \vDash \exists y\,(Rxy \wedge st_y(\psi))$ by means of a supplementing function $f : T^x \to \wp^+(W)$ such that $(T^x)^y_f \vDash Rxy \wedge st_y(\psi)$.

  We define the team $S := \bigcup_{w\in T} f(w^x)$, first prove that it is a successor team of T, and then show that it satisfies ψ. For the first part, let $v \in S$. Then there exists $w \in T$ such that $v \in f(w^x)$. As a consequence, the assignment s given by $s(x) = w$ and $s(y) = v$ is in $(T^x)^y_f$, and hence satisfies Rxy. In other words, v has a predecessor in T, namely w. Conversely, if $w \in T$, then $f(w^x)$ is non-empty, i.e., contains an element v. Again, v is a successor of w. Since $v \in f(w^x)$, $v \in S$, so w has a successor in S. For the second part, we again use locality and the fact that $S^y = (T^x)^y_f\restriction\{y\}$ as before. □

It is easy to see that the standard translation of an $\mathsf{ML}(\sim)$-formula is in fact an $\mathsf{GF}^2(\sim)$-formula. From this, in the next section we obtain several lower bounds for fragments of $\mathsf{FO}(\sim)$.

5.2.1 Lower bounds

Next, we prove that the standard translation carries several complexity lower bounds into the first-order setting.

In fact, the lower bounds already hold with equality *and* without equality if another predicate is present. For this reason, in what follows we explicitly consider equality as a predicate and then write $= \in \sigma$.

Lemma 5.20. *$\mathrm{MC}(\sigma\text{-}\mathsf{GF}^1(\sim))$ is* PSPACE*-hard if σ contains infinitely many predicates.*

Proof. We reduce from $\mathrm{MC}(\mathsf{PL}(\sim))$ which is PSPACE-complete (see [119] or [56, Thm. 3.3]). The lower bound by Hannula et al. [56] is easily seen to hold under logspace reductions. The reduction is now simply the standard translation, i.e., it maps $(\mathcal{K}, T, \varphi)$ to $(\ulcorner\mathcal{K}\urcorner, T^x, \mathsf{st}_x(\varphi))$. (W.l.o.g. σ contains predicates Q_p for every $p \in \mathsf{Prop}$; otherwise they are easily simulated by predicates of higher arity). It is easy to see that $\mathsf{st}_x(\varphi)$ is quantifier-free, hence guarded, and contains only the variable x. Moreover, by Theorem 5.19, $(\mathcal{K}, T) \vDash \varphi \Leftrightarrow (\ulcorner\mathcal{K}\urcorner, T^x) \vDash \mathsf{st}_x(\varphi)$. □

Lemma 5.21. *$\mathrm{MC}(\sigma\text{-}\mathsf{FO}(\sim))$ is* ATIME-ALT$(\exp, \mathrm{poly})$*-hard if σ contains at least one predicate or equality. This even holds when restricted to sentences and for a fixed σ-structure $\mathcal{A}$ with domain $\{0, 1\}$ and team $\{\emptyset\}$.*

Proof. We reduce from $\mathrm{SAT}(\mathsf{PL}(\sim))$, which is ATIME-ALT$(\exp, \mathrm{poly})$-hard (see Theorem 4.32 or [56, Thm. 4.9], the hardness result by Hannula et al. [56] again also holds for logspace reductions).

The reduction from $\mathrm{SAT}(\mathsf{PL}(\sim))$ consists of several steps. Given $\varphi \in \mathsf{PL}(\sim)$, suppose that $\mathsf{Prop}(\varphi) = \{p_1, \ldots, p_n\}$. First, observe that φ is satisfiable if and only if $\varphi' := \top \vee \varphi$ is true in the full propositional team $T^{\max}$ over $p_1, \ldots, p_n$. So it suffices to reduce from the problem of deciding truth of φ' in the full team w. r. t. φ'.

Next, we translate from propositional to first-order logic. Fix a σ-structure $\mathcal{A}$ with domain $\{0, 1\}$. Recall that σ contains either equality or some other predicate. We start with the first case. The idea is that the Boolean assignments are simulated by first-order assignments $s \colon X \to \mathcal{A}$, where $X = \{z, x_1, \ldots, x_n\}$. Here, x_i simulates p_i, and the auxiliary variable z plays the role of the domain element 1.

Hence we map φ' to the formula φ^* which is obtained from φ' by replacing every occurrence of a proposition p_i by the atomic formula $x_i = z$. We define the team accordingly; for $b \in \{0, 1\}$, let $V_b := \{s \colon X \to \{0, 1\} \mid s(z) = b\}$, i.e., V_b is the full team w. r. t. $\{x_1, \ldots, x_n\}$, but z is constantly b. By a straightforward induction, now $V_1 \vDash \varphi^*$ iff $T^{\max} \vDash \varphi'$.

The team V_1 is exponentially large, so we cannot compute it as part of the reduction. But we can compute an equivalent instance with the team $U_1 := \{z \mapsto 1\}$, and the formula $\forall x_1 \cdots \forall x_n \varphi^*$, as V_1 is the n-fold duplicating team of U_1. In fact, we can output the team $U := \{\emptyset\}$ and formula $\psi := \exists z \forall x_1 \cdots \forall x_n \varphi^*$, as $T^{\max} \vDash \varphi'$ implies $U \vDash \psi$. The other direction holds as well, since w.l.o.g. z is chosen as 1. (Otherwise flip all bits in all assignments, this does not change the truth of $x_i = z$ for any i, and hence not of φ^*.) Hence we map φ to the team $\{\emptyset\}$, structure $\mathcal{A}$ and formula $\exists z \forall x_1 \cdots \forall x_n \varphi^*$.

The case remains where σ does not contain $=$, but contains some predicate R. Then we define the team as above, but let $R^{\mathcal{A}} := \{(1, \ldots, 1)\}$, and output the formula $\forall x_1 \cdots \forall x_n \varphi^*$, where φ^* is defined as the formula obtained from φ by replacing p_i with $R(x_i, \ldots, x_i)$. □

Clearly, the standard translation of satisfiable formulas is itself satisfiable. But the converse is also true: In a sense, from a given first-order structure (and team) that satisfies $\mathsf{st}_x(\varphi)$ we can reconstruct a Kripke model (and team) for φ.

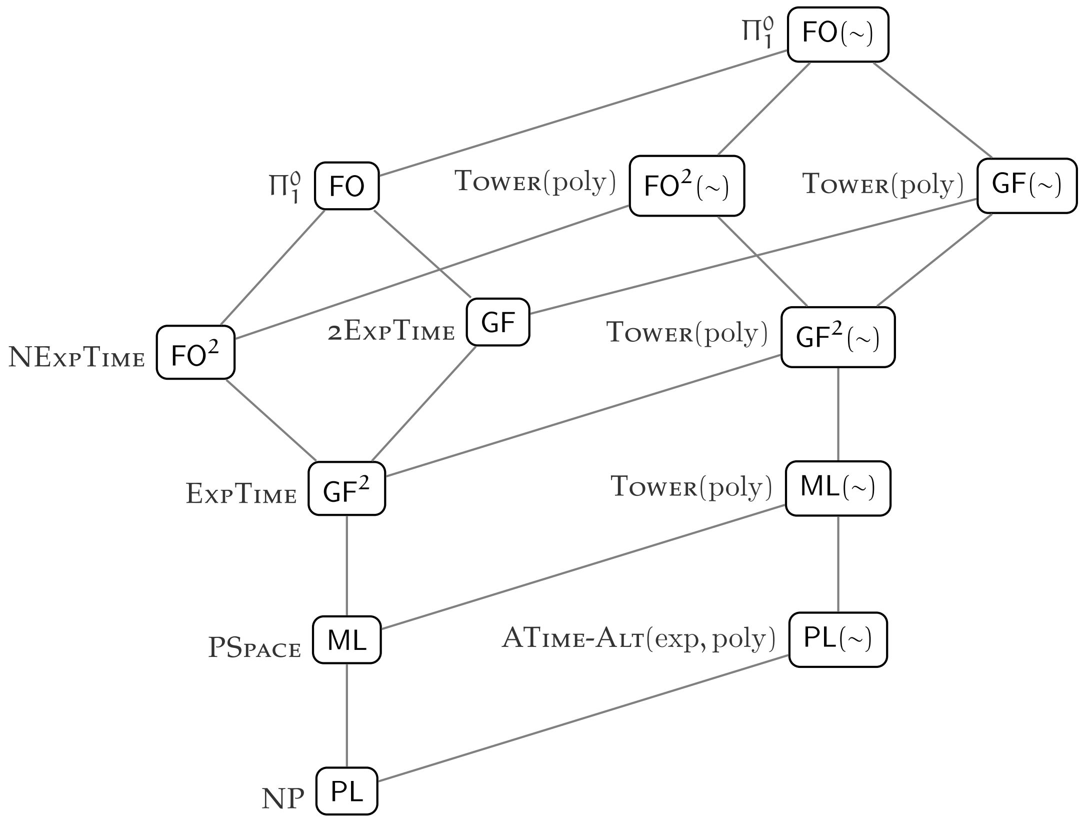


Figure 5.3: Inclusion diagram of fragments of first-order team logic.

	Complexity of ...			
$\mathcal{L}$	SAT($\mathcal{L}$)		SAT($\mathcal{L}(\sim)$)	
PL	NP	[18]	ATime-Alt(exp, poly)	[56]
ML	PSpace	[94]	Tower(poly)	Thm. 4.32
GF^2	ExpTime	[45]	Tower(poly)	Thm. 5.26
FO^2	NExpTime	[48]	Tower(poly)	Thm. 5.26
GF	2ExpTime	[45]	Tower(poly)	Thm. 5.26
FO	Π_1^0	[12]	Π_1^0	Thm. 5.6

Table 5.4: The complexity of fragments of first-order team logic. All entries are completeness results.

Lemma 5.22. *Let* $\varphi \in \mathsf{ML}(\sim)$*. Then* φ *is satisfiable if and only if* $\mathsf{st}_x(\varphi)$ *is satisfiable.*

Proof. As Theorem 5.19 proves "⇒", we only consider "⇐". Suppose $(\mathcal{B}, S) \vDash \mathsf{st}_x(\varphi)$, where $\mathcal{B}$ is some first-order structure and S is a team in $\mathcal{B}$. By locality (Proposition 2.14), w.l.o.g. S has domain $\{x\}$. Define now the Kripke structure $\mathcal{K} = (|\mathcal{B}|, R^{\mathcal{B}}, V)$ such that $V(p) := Q_p^{\mathcal{B}}$. Then clearly $\ulcorner\mathcal{K}\urcorner = \mathcal{B}$. By Theorem 5.19, $(\mathcal{K}, x\langle S\rangle) \vDash \varphi$, so φ is satisfiable. ☐

Theorem 5.23. *Let* σ *be a vocabulary that contains at least one predicate or equality.*

- $\mathrm{MC}(\sigma\text{-}\mathsf{FO}(\sim))$ *is* ATIME-ALT(exp, poly)*-complete, with hardness also on sentences and for a fixed* σ*-structure* $\mathcal{A}$ *with domain* $\{0, 1\}$ *and a fixed team* $\{\emptyset\}$*.*
- *If* σ *contains infinitely many predicates and* $0 < n < \omega$*, then* $\mathrm{MC}(\sigma\text{-}\mathsf{FO}^n(\sim))$ *and* $\mathrm{MC}(\sigma\text{-}\mathsf{GF}^n(\sim))$ *are* PSPACE*-complete.*

Proof. The upper bounds are due to Proposition 5.16 and Corollary 5.17. The lower bounds are due to Lemmas 5.20 and 5.21. ☐

Let $\mathcal{D}$ be a set of non-classical atoms. We call $\mathcal{D}$ *polynomial time computable* if

$$\left\{(\mathcal{A}, T, \delta) \mid \delta \in \mathcal{D} \text{ and } (\mathcal{A}, T) \vDash \delta\right\} \in \mathrm{P}$$

where $\mathcal{A}$ denotes a first-order structure and T a team in $\mathcal{A}$. Let $\mathsf{FO}(\mathcal{D}, \sim)$ be the logic $\mathsf{FO}(\sim)$ extended by atoms in $\mathcal{D}$. Define $\mathsf{FO}^n(\mathcal{D}, \sim)$ analogously.

Now, Algorithm 1 is easily adapted to have another clause for atoms in $\mathcal{D}$. The resulting runtime is clearly still exponential, or polynomial in the case of a bounded number of variables, respectively.

Corollary 5.24. *Let* $\mathcal{D}$ *be a polynomial time computable set of non-classical atoms. Then* $\mathrm{MC}(\mathsf{FO}(\mathcal{D}, \sim))$ *is* ATIME-ALT(exp, poly)*-complete, and* $\mathrm{MC}(\mathsf{FO}^n(\mathcal{D}, \sim))$ *is* PSPACE*-complete if* $0 < n < \omega$*.*

Corollary 5.25. $\mathrm{MC}(\mathsf{FO}(\mathsf{dep}, \sim))$ *is* ATIME-ALT(exp, poly)*-complete.*

Likewise, the model checking problems of independence logic, inclusion logic and exclusion logic with Boolean negation are ATIME-ALT(exp, poly)-complete, and their two-variable fragments are PSPACE-complete.

The next theorem gathers the results of this chapter regarding the complexity of satisfiability (see also Figure 5.3 and Table 5.4).

Theorem 5.26. *Let* σ *be an infinite relational vocabulary with at least one binary predicate. Let* $\mathcal{L}$ *be any fragment such that* $\sigma\text{-}\mathsf{GF}^2(\sim) \subseteq \mathcal{L} \subseteq \sigma\text{-}\mathsf{FO}^2(\sim) \cup \sigma\text{-}\mathsf{GF}(\sim)$*. Then* $\mathrm{SAT}(\mathcal{L})$ *and* $\mathrm{VAL}(\mathcal{L})$ *are* TOWER(poly)*-complete.*

Proof. The upper bounds for $\mathsf{FO}^2(\sim)$ and $\mathsf{GF}(\sim)$ are by Theorems 5.10 and 5.15, so an algorithm can choose the appropriate decision procedure as a subroutine depending on the input formula. For the lower bounds, the mapping $\varphi \mapsto \mathsf{st}_x(\varphi)$ constitutes a reduction from $\mathrm{SAT}(\mathsf{ML}(\sim))$ to $\mathrm{SAT}(\mathsf{GF}^2(\sim))$ (see Theorem 4.32 and Lemma 5.22). Finally, the validity cases easily follow since the logics $\sigma\text{-}\mathsf{GF}^2(\sim)$ and $\sigma\text{-}\mathsf{FO}^2(\sim) \cup \sigma\text{-}\mathsf{GF}(\sim)$ and the class TOWER(poly) are closed under negation. ☐

5.3 Łoś's theorem for team semantics

Model theory offers manifold notions of morphisms and constructions, of which many preserve the first-order theory of structures. One such construction is the *product* of structures, and in particular the *ultraproduct* of structures (also called *ultrapower* if all factors are identical). A famous theorem by Jerzy Łoś states that a structure and its ultrapower have the same first-order theory. For this section we mostly follow Chang and Keisler [14] (see also van Dalen [21]); the fluent reader is referred to the original result by Łoś [97] and to Skolem [131] for an early appearance of an ultraproduct construction of $\mathbb{N}$.

In this section, we generalize the construction to team semantics. Currently, for example for dependence logic $\mathsf{FO}(\mathsf{dep})$, most model-theoretic results stem from its equivalence to existential second-order logic $\mathsf{SO}(\exists)$. With added negation however, $\mathsf{FO}(\mathsf{dep}, \sim)$ is equivalent to full SO and loses many nice model-theoretic properties, such as compactness [135]. Because of this, for investigating the model theory of $\mathsf{FO}(\sim)$ or other team logics with negation, their second-order characterization may not be a good starting point. Instead, a direct approach that does not hinge on translation to higher order logic may be more favorable.

In this section, we instead study the model-theoretic properties of $\mathsf{FO}(\sim)$ directly. As a new result in this area, we present a team-semantical analog to Łoś's theorem. A corollary of it is the compactness theorem for $\mathsf{FO}(\sim)$.[1] Previously it was only known that FO and $\mathsf{FO}(\mathsf{dep})$ are compact (since $\mathsf{SO}(\exists)$ is compact) and that $\mathsf{FO}(\mathsf{dep}, \sim)$ is not. With a compactness theorem for $\mathsf{FO}(\sim)$, this now implies that it cannot define, e.g., the infinity of structures (even if this unsurprising since it can also not define the infinity of teams, cf. Corollary 5.2).

We begin with some required notation from basic order theory. The reader is also referred to the very good introduction by Davey and Priestley [22]. A *partially ordered set* (*poset*) is a pair $(X, \leqslant)$ where X is a set and $\leqslant$ is a reflexive, transitive and anti-symmetric (if $x \leqslant y$ and $y \leqslant x$ then $x = y$) binary relation. A *filter* on a poset $(X, \leqslant)$ is a family $\mathcal{F} \subseteq \wp X$ of subsets of X that is upward closed ($Y \in \mathcal{F}$ and $Y \subseteq Z \subseteq X$ implies $Z \in \mathcal{F}$), and that is closed under finite intersection ($Y, Z \in \mathcal{F}$ implies $Y \cap Z \in \mathcal{F}$). For example, the *Fréchet filter* is the filter of all subsets of X with a finite complement.

An *ultrafilter* $\mathcal{F}$ on a poset $(X, \leqslant)$ is a filter such that additionally $Y \in \mathcal{F} \Leftrightarrow X \setminus Y \notin \mathcal{F}$ for all $Y \subseteq X$. In what follows, ultrafilters will denoted by $\mathcal{U}$. The Fréchet filter on infinite X is never an ultrafilter, since there are sets Y such that neither Y nor $X \setminus Y$ is finite. For any fixed $x \in X$ however, for example $\mathcal{U} := \{Y \subseteq X \mid x \in Y\}$ is an ultrafilter. Any ultrafilter is non-empty, as it contains X. The existence of ultrafilters is implied, for example, by the so-called finite intersection property (see, e.g., [14, Proposition 4.1.3]):

Proposition 5.27. *If X is a set, $\mathcal{F} \subseteq \wp X$, and the intersection of finitely many elements of $\mathcal{F}$ is always non-empty, then there exists an ultrafilter $\mathcal{U} \subseteq \wp X$ such that $\mathcal{U} \supseteq \mathcal{F}$.*

[1] Of course the compactness theorem can also be obtained by other means, such as a sound and complete proof system for $\mathsf{FO}(\sim)$ (see Chapter 6), or by a translation to FO sentences (see Theorem 5.4).

We write (ultra-)products of sets, structures etc. in German letters. In what follows, we always assume that I is a set and $\mathcal{U}$ is an ultrafilter on I.

Two families $\mathfrak{a} = (a_i)_{i\in I}$ and $\mathfrak{b} = (b_i)_{i\in I}$ are $\mathcal{U}$-*equivalent*, in symbols $\mathfrak{a} \approx_{\mathcal{U}} \mathfrak{b}$, if $\{i \mid a_i = b_i\} \in \mathcal{U}$. Observe that this is an equivalence relation. The intuition is that an ultrafilter $\mathcal{U}$ contains all the "large" subsets, and if the set $\{i \mid a_i = b_i\}$ is "large", then $\mathfrak{a}$ and $\mathfrak{b}$ "agree almost everywhere". The $\approx_{\mathcal{U}}$-equivalence class of $\mathfrak{a}$ is $[\mathfrak{a}]$.

Ultraproducts. The *set ultraproduct* of sets $(X_i)_{i\in I}$ is the quotient w. r. t. $\approx_{\mathcal{U}}$ of their Cartesian product:

$$\prod_{\mathcal{U}} X_i := \left\{ [\mathfrak{a}] \;\middle|\; \mathfrak{a} \in \prod_{i\in I} X_i \right\}$$

Next, for each $i \in I$, let $\mathcal{A}_i$ be a σ-structure with domain A_i.

Definition 5.28 (Structure ultraproduct)**.** The *structure ultraproduct* $\mathfrak{A} := \prod_{\mathcal{U}} \mathcal{A}_i$ is the σ-structure with domain $\prod_{\mathcal{U}} A_i$ and the following interpretation of σ.

- For an r-ary relation symbol $R \in \sigma$ and $\mathfrak{a}_1, \ldots, \mathfrak{a}_r \in \prod_{i\in I} A_i$, let

$$([\mathfrak{a}_1], \ldots, [\mathfrak{a}_r]) \in R^{\mathfrak{A}} \;\Leftrightarrow\; \{i \in I \mid (\mathfrak{a}_1(i), \ldots, \mathfrak{a}_r(i)) \in R^{\mathcal{A}_i}\} \in \mathcal{U}$$

- For an r-ary function symbol $f \in \sigma$, and $\mathfrak{a}_1, \ldots, \mathfrak{a}_r \in \prod_{i\in I} A_i$, let

$$f^{\mathfrak{A}}([\mathfrak{a}_1], \ldots, [\mathfrak{a}_r]) \;:=\; \left[(f^{\mathcal{A}_i}(\mathfrak{a}_1(i), \ldots, \mathfrak{a}_n(i)))_{i\in I}\right]$$

The above interpretations of relations and functions are well-defined [14, Proposition 4.1.7], since they depend only on the respective equivalence class.

For formulas with free variables we have to construct an assignment in the same fashion. Let $s_i \colon \mathsf{Var} \to A_i$ be an assignment for each $i \in I$. We cannot simply define $\mathfrak{s} = (s_i)_{i\in I}$; observe that $\mathfrak{s} \in \prod_{i\in I}(\mathsf{Var} \to A_i)$, so $\mathfrak{s}$ maps I to functions of Var instead of the other way around. We have to transpose the arguments to obtain an assignment.

Definition 5.29 (Assignment ultraproduct)**.** Let $(s_i)_{i\in I} \in \prod_{i\in I}(\mathsf{Var} \to A_i)$. Then the *assignment ultraproduct* $\prod_{\mathcal{U}} s_i \colon \mathsf{Var} \to \prod_{\mathcal{U}} A_i$ is defined by

$$\prod_{\mathcal{U}} s_i \;:\; x \mapsto \left[\big(s_i(x)\big)_{i\in I}\right]$$

for all $x \in \mathsf{Var}$.

Intuitively, $\mathfrak{s}(x) := \left(\prod_{\mathcal{U}} s_i\right)(x)$ is the "consensus" of the evaluation of all the s_i at x. Evaluating a term in the ultraproduct $\mathfrak{A}$ behaves as expected:

Lemma 5.30. *Let* t *be a term. Then* $t\langle \mathfrak{A}, \mathfrak{s}\rangle = [(t\langle \mathcal{A}_i, s_i\rangle)_{i\in I}]$.

Proof. Proved in Chang and Keisler [14, Theorem 4.1.9]. Essentially, the proof is by induction on t and application of Definitions 5.28 and 5.29. □

From the above lemma, Łoś's theorem again follows easily by induction.

Theorem 5.31 (Łoś). *Let* $\mathfrak{A} = \prod_{\mathcal{U}} \mathcal{A}_i$ *and* $\mathfrak{s} = \prod_{\mathcal{U}} s_i$*. Then, for every formula* $\alpha \in \sigma\text{-}\mathsf{FO}$*, it holds that* $(\mathfrak{A}, \mathfrak{s}) \vDash \alpha$ *if and only if* $\{\, i \mid (\mathcal{A}_i, s_i) \vDash \alpha \,\} \in \mathcal{U}$.

Proof. See Chang and Keisler [14, Theorem 4.1.9]. □

Here, the intuition is again that the ultraproduct $\mathfrak{A}$ satisfies the formulas that are true in "almost all" $\mathcal{A}_i$. Having stated the classical definitions and result, we now switch to the team-semantical setting and introduce basic definitions that are required to generalize the theorem.

5.3.1 Definition for team semantics and main result

First of all, we propose a suitable notion of ultraproducts of teams.

Definition 5.32 (Team ultraproduct). Let $(T_i)_{i \in I}$ be a family of teams in the respective structures $\mathcal{A}_i$, that is, $T_i \subseteq \mathsf{Var} \to \mathcal{A}_i$ for all i. Then the *team ultraproduct* $\prod_{\mathcal{U}} T_i$ is defined as the team

$$\prod_{\mathcal{U}} T_i := \left\{ \prod_{\mathcal{U}} s_i \;\middle|\; (s_i)_{i \in I} \in \prod_{i \in I} (\mathsf{Var} \to \mathcal{A}_i) \text{ and } \{i \mid s_i \in T_i\} \in \mathcal{U} \right\}.$$

Let us lose some words on this definition. First, by definition of assignment ultraproducts $\prod_{\mathcal{U}} s_i$ (Definition 5.29) it is easy to see that $\prod_{\mathcal{U}} T_i \subseteq \mathsf{Var} \to \prod_{\mathcal{U}} \mathcal{A}_i$, that is, this is a team in $\mathfrak{A} := \prod_{\mathcal{U}} \mathcal{A}_i$. The intuition is now that this team contains precisely those assignments that are a member of "almost all" teams T_i, which is a definition that fits with the idea of set, structure and assignment ultraproducts. This is expressed by the condition $\{i \mid s_i \in T_i\} \in \mathcal{U}$.

With the above definition, we are now ready to prove the analog to Łoś's theorem. Let $\mathfrak{A} := \prod_{\mathcal{U}} \mathcal{A}_i$ and $\mathfrak{T} := \prod_{\mathcal{U}} T_i$ be the ultraproduct of structures $(\mathcal{A}_i)_{i \in I}$ with domains $(A_i)_{i \in I}$ and teams $(T_i)_{i \in I}$, respectively, where $T_i \subseteq \mathsf{Var} \to \mathcal{A}_i$.

In what follows, we say that a formula φ *is preserved in ultraproducts* if

$$(\mathfrak{A}, \mathfrak{T}) \vDash \varphi \;\Leftrightarrow\; \{\, i \mid (\mathcal{A}_i, T_i) \vDash \varphi \,\} \in \mathcal{U}$$

for all I, $\mathcal{U}$, $\mathfrak{A}$, $\mathfrak{T}$, $\mathcal{A}_i$, and T_i as above.

Lemma 5.33. *Every flat formula is preserved in ultraproducts.*

Proof. We have to show $(\mathfrak{A}, \mathfrak{T}) \vDash \alpha \Leftrightarrow \{\, i \mid (\mathcal{A}_i, T_i) \vDash \alpha \,\} \in \mathcal{U}$, where α is flat. The proof is by contraposition.

"$\Rightarrow$": Suppose $\{\, i \mid (\mathcal{A}_i, T_i) \vDash \alpha \,\} \notin \mathcal{U}$. As $\mathcal{U}$ is an ultrafilter, $J := \{\, i \mid (\mathcal{A}_i, T_i) \nvDash \alpha \,\} \in \mathcal{U}$. By flatness of α, for each $i \in J$ the team T_i contains an assignment s_i falsifying α. By the axiom of choice, extend this to a family $(s_i)_{i \in I}$ of assignments where s_i is arbitrary for $i \notin J$, but such that $s_i \in T_i$ and $(\mathcal{A}_i, s_i) \nvDash \alpha$ if $i \in J$. Let $\mathfrak{s} := \prod_{\mathcal{U}} s_i$. Our goal is to show that $\mathfrak{s}$ is a witness of the flat formula α being false in $\mathfrak{T}$. First observe that $J \subseteq \{i \mid s_i \in T_i\}$ by choice of the s_i, and by upwards closure of ultrafilters, the latter set is in $\mathcal{U}$. But then $\mathfrak{s} \in \mathfrak{T}$ by Definition 5.32. Moreover, $J \subseteq \{ i \mid (\mathcal{A}_i, s_i) \nvDash \alpha \}$, so the latter set is in $\mathcal{U}$ as well, and by Łoś's theorem, $(\mathfrak{A}, \mathfrak{s}) \nvDash \alpha$. Consequently, $(\mathfrak{A}, \mathfrak{T}) \nvDash \alpha$ by flatness of α.

"$\Leftarrow$": Suppose $(\mathfrak{A}, \mathfrak{T}) \nvDash \alpha$. Then $(\mathfrak{A}, \mathfrak{s}) \nvDash \alpha$ for some family $(s_i)_{i \in I}$ with $\mathfrak{s} = \prod_{\mathcal{U}} s_i$ and $\mathfrak{s} \in \mathfrak{T}$. By Łoś's theorem, $\{ i \mid (\mathcal{A}_i, s_i) \vDash \alpha \} \notin \mathcal{U}$, so for the complement $J := \{ i \mid (\mathcal{A}_i, s_i) \nvDash \alpha \}$ we have $J \in \mathcal{U}$. If now each of these s_i is in T_i we would be done, but this is not necessarily the case. However, recall that $\{ i \mid s_i \in T_i \} \in \mathcal{U}$ by definition of $\mathfrak{T}$. $\mathcal{U}$ is closed under finite intersection, so $\{ i \mid s_i \in T_i \text{ and } (\mathcal{A}_i, s_i) \nvDash \alpha \} \in \mathcal{U}$ and as a consequence, $\{ i \mid (\mathcal{A}_i, T_i) \nvDash \alpha \} \in \mathcal{U}$ by flatness. □

Theorem 5.34. *Every* FO(∼)*-formula is preserved in ultraproducts.*

Proof. Let $\varphi \in \mathrm{FO}(\sim)$. The proof is now by induction on φ. By Theorem 5.1, φ is semantically equivalent to a Boolean combination of FO-formulas, so it suffices to consider only classical formulas, negation $\sim$, and conjunction $\wedge$.

- The case $\varphi \in \mathrm{FO}$ is proved by the previous lemma.
- If $\varphi = \psi \wedge \theta$, then $(\mathfrak{A}, \mathfrak{T}) \vDash \psi, \theta$ iff $\{ i \mid (\mathcal{A}_i, T_i) \vDash \psi \} \in \mathcal{U}$ and $\{ i \mid (\mathcal{A}_i, T_i) \vDash \theta \} \in \mathcal{U}$. But this is equivalent to $\{ i \mid (\mathcal{A}_i, T_i) \vDash \psi \wedge \theta \} \in \mathcal{U}$ due to closure under finite intersection and upward closure of ultrafilters.
- If $\varphi = \sim\psi$, then $(\mathfrak{A}, \mathfrak{T}) \nvDash \psi$ is equivalent to $\{ i \mid (\mathcal{A}_i, T_i) \vDash \psi \} \notin \mathcal{U}$, which is equivalent to $\{ i \mid (\mathcal{A}_i, T_i) \nvDash \psi \} \in \mathcal{U}$ by definition of an ultrafilter. □

5.3.2 Łoś's theorem for non-classical atoms

Next, we show that Łoś's theorem for teams does not hold only for FO(∼), but also for the non-classical atoms such as dependence or independence.

In what follows, fix a vocabulary σ and an n-ary relation symbol $R \notin \sigma$. If $\mathcal{R}$ is an n-ary relation, then $(\mathcal{A}, \mathcal{R})$ means the $\sigma \cup \{R\}$-structure that expands $\mathcal{A}$ by interpreting R as $\mathcal{R}$.

Definition 5.35. Every first-order $\sigma \cup \{R\}$-formula δ defines an n-ary atom $\mathbf{D}$ as follows: For all σ-structures $\mathcal{A}$, teams T in $\mathcal{A}$ and σ-terms $\vec{t} = t_1, \ldots, t_n$ it holds that

$$(\mathcal{A}, T) \vDash \mathbf{D}\vec{t} \iff (\mathcal{A}, \vec{t}\langle \mathcal{A}, T \rangle) \vDash \delta(R).$$

An atom $\mathbf{D}$ is *first-order definable* if it is defined by some formula.

Theorem 5.36. *Every first-order definable atom is preserved in ultraproducts.*

Proof. Let $\mathbf{D}$ be an atom defined by the first-order formula $\delta(R)$. To reduce clutter, we assume that $\mathbf{D}$ is unary. The case of higher arities is handled analogously. We also omit $\mathcal{A}$ in the notation $t\langle \mathcal{A}, \cdot \rangle$ if it is clear.

Let now $\mathfrak{A} = \prod_{\mathcal{U}} \mathcal{A}$, $\mathfrak{T} = \prod_{\mathcal{U}} T_i$, and let t be a term. The key is that

$$[\mathfrak{a}] \in t\langle\mathfrak{T}\rangle \text{ if and only if } \{\, i \mid a_i \in t\langle T_i\rangle \,\} \in \mathcal{U}. \qquad (\star)$$

for all families $\mathfrak{a} = (a_i)_{i\in I} \in \prod_{i\in I} \mathcal{A}_i$, which mirrors the definition of the interpretation of a relation symbol in an ultraproduct (Definition 5.28). Consequently, $(\mathfrak{A}, \vec{t}\langle\mathfrak{T}\rangle) = \prod_{\mathcal{U}}(\mathcal{A}_i, \vec{t}\langle T_i\rangle)$. Hence by Łoś's theorem, $(\mathfrak{A}, \vec{t}\langle\mathfrak{T}\rangle) \vDash \delta$ if and only if $\{i \mid (\mathcal{A}_i, \vec{t}\langle T_i\rangle) \vDash \delta\} \in \mathcal{U}$, which proves the theorem by definition of **D**.

It remains to prove $(\star)$.

"$\Rightarrow$": Assume $\mathfrak{a} = (a_i)_{i\in I}$ as above and let $[\mathfrak{a}] \in t\langle\mathfrak{T}\rangle$. This means there is some assignment $\mathfrak{s} \in \mathfrak{T}$ such that $t\langle\mathfrak{s}\rangle = [\mathfrak{a}]$. Now $\mathfrak{s}$ is of the form $\prod_{\mathcal{U}} s_i$ for some $(s_i)_{i\in I}$. By Lemma 5.30, $t\langle\mathfrak{s}\rangle = [\mathfrak{a}] = [(t\langle s_i\rangle)_{i\in I}]$. So by definition of $[\cdot]$, $\{i \mid a_i = t\langle s_i\rangle\} \in \mathcal{U}$. Moreover, by definition of $\mathfrak{T}$, the set $\{i \mid s_i \in T_i\}$ is in $\mathcal{U}$. By intersection and upward closure,

$$\{i \mid s_i \in T_i \text{ and } a_i = t\langle s_i\rangle\} \subseteq \{i \mid a_i \in t\langle T_i\rangle\} \in \mathcal{U}$$

which finishes the first direction.

"$\Leftarrow$": By assumption, the set $J := \{\, i \mid a_i \in t\langle T_i\rangle \,\}$ is in $\mathcal{U}$. This means that for every $i \in J$ there exists an assignment $s_i \in T_i$ such that $a_i = t\langle s_i\rangle$. By the axiom of choice, extend this to a family $(s_i)_{i\in I}$ of assignments such that $s_i \in T_i$ and $a_i = t\langle s_i\rangle$ for $i \in J$, and s_i is arbitrary for $i \notin J$. Define the assignment $\mathfrak{s} := \prod_{\mathcal{U}} s_i$. Firstly, $\mathfrak{s} \in \mathfrak{T}$ since $J \subseteq \{i \mid s_i \in T_i\} \in \mathcal{U}$. Secondly, $t\langle\mathfrak{s}\rangle = [t\langle s_i\rangle_{i\in I}]$ by Lemma 5.30, which equals $[\mathfrak{a}]$ by definition of $[\cdot]$, as $J \subseteq \{i \mid a_i = t\langle s_i\rangle\} \in \mathcal{U}$. It follows that $[\mathfrak{a}] \in t\langle\mathfrak{T}\rangle$. ☐

Corollary 5.37. *The atoms of dependence, independence, inclusion, and exclusion are preserved in ultraproducts.*

By the inductive proof for $\wedge$ and $\sim$ in Theorem 5.34, we obtain:

Corollary 5.38. *Every Boolean combination of* $\mathsf{FO}(\sim)$*-formulas and/or first-order definable atoms is preserved in ultraproducts.*

5.3.3 Łoś's theorem for non-classical atoms: A direct proof

The above proof that Łoś's theorem extends to all first-order definable generalized atoms is powerful but rather opaque. In what follows, we give an exemplary direct proof for the independence atom in order to demonstrate how such a result can be proved without assuming that an atom is first-order definable. This can easily be adapted to the other non-classical atoms.

Theorem 5.39. *The independence atom is preserved in ultraproducts.*

Alternative proof. Again, we omit the structure $\mathcal{A}$ in the notation $t\langle\mathcal{A}, \cdot\rangle$ if it is clear.

Let $\vec{p}, \vec{q}, \vec{r}$ be sequences of terms and let $\varphi := \vec{p} \perp_{\vec{r}} \vec{q}$. We have to show that $(\mathfrak{A}, \mathfrak{T}) \vDash \varphi$ if and only if $\{\, i \mid (\mathcal{A}_i, T_i) \vDash \varphi \,\} \in \mathcal{U}$.

"$\Rightarrow$": Proof by contraposition. Let $\{\, i \mid (\mathcal{A}_i, T_i) \vDash \varphi \,\} \notin \mathcal{U}$. Then $J := \{\, i \mid (\mathcal{A}_i, T_i) \nvDash \varphi \,\} \in \mathcal{U}$. By definition of the independence atom, for each $i \in J$ we can pick $s_i, s_i' \in T_i$ that violate φ in the sense that $\vec{r}\langle s_i \rangle = \vec{r}\langle s_i' \rangle$, but there exists no $s_i'' \in T_i$ that satisfies both $\vec{p}\vec{r}\langle s_i \rangle = \vec{p}\vec{r}\langle s_i'' \rangle$ and $\vec{q}\langle s_i' \rangle = \vec{q}\langle s_i'' \rangle$. From this, we define assignments $\mathfrak{s}, \mathfrak{s}'$ as follows. By the above argument, there are families $(s_i)_{i \in I}$, $(s_i')_{i \in I}$ such that s_i and s_i' are arbitrary for $i \notin J$, but if $i \in J$ then $s_i, s_i' \in T_i$ and no s_i'' as stated before exists in T_i. Next, let $\mathfrak{s} := \prod_{\mathcal{U}} s_i$ and $\mathfrak{s}' := \prod_{\mathcal{U}} s_i'$. Then clearly $\mathfrak{s}, \mathfrak{s}' \in \mathfrak{T}$, since $J \subseteq \{i \mid s_i \in T_i\}$ and $J \subseteq \{i \mid s_i' \in T_i\}$.

If now $(\mathfrak{A}, \mathfrak{T}) \vDash \varphi$, then there must exist an assignment $\mathfrak{s}'' \in \mathfrak{T}$ such that $\vec{p}\vec{r}\langle \mathfrak{s} \rangle = \vec{p}\vec{r}\langle \mathfrak{s}'' \rangle$ and $\vec{q}\langle \mathfrak{s}' \rangle = \vec{q}\langle \mathfrak{s}'' \rangle$. There is a family $(s_i'')_{i \in I}$ such that $\mathfrak{s}'' = \prod_{\mathcal{U}} s_i''$. By Lemma 5.30 and Definition 5.32, the sets $\{i \mid \vec{p}\vec{r}\langle s_i \rangle = \vec{p}\vec{r}\langle s_i'' \rangle\}$, $\{i \mid \vec{q}\langle s_i' \rangle = \vec{q}\langle s_i'' \rangle\}$ and $\{i \mid s_i'' \in T_i\}$ are all in $\mathcal{U}$, and so is their intersection

$$J' := \left\{ i \;\middle|\; s_i'' \in T_i \text{ and } \vec{p}\vec{r}\langle s_i \rangle = \vec{p}\vec{r}\langle s_i'' \rangle \text{ and } \vec{q}\langle s_i' \rangle = \vec{q}\langle s_i'' \rangle \right\}.$$

But then the sets J' and

$$J'' := \left\{ i \;\middle|\; \nexists s'' \in T_i : \vec{p}\vec{r}\langle s_i \rangle = \vec{p}\vec{r}\langle s'' \rangle \text{ and } \vec{q}\langle s_i' \rangle = \vec{q}\langle s'' \rangle \right\} \supseteq J$$

are both in $\mathcal{U}$, which is impossible since they have an empty intersection.

"$\Leftarrow$": Suppose that $\{ i \mid (\mathcal{A}_i, T_i) \vDash \varphi \} \in \mathcal{U}$. Let $\mathfrak{s} = \prod_{\mathcal{U}} s_i$ and $\mathfrak{s}' = \prod_{\mathcal{U}} s_i'$ be arbitrary assignments in $\mathfrak{T}$ such that $\vec{r}\langle \mathfrak{s} \rangle = \vec{r}\langle \mathfrak{s}' \rangle$. First observe that

$$J := \left\{ i \;\middle|\; \begin{array}{l} s_i \notin T_i \text{ or } s_i' \notin T_i \text{ or } \vec{r}\langle s_i \rangle \neq \vec{r}\langle s_i' \rangle \\ \qquad \text{or } \exists s_i'' \in T_i : (\vec{p}\vec{r}\langle s_i \rangle = \vec{p}\vec{r}\langle s_i'' \rangle \text{ and } \vec{q}\langle s_i' \rangle = \vec{q}\langle s_i'' \rangle) \end{array} \right\}$$

is in $\mathcal{U}$ due to $J \supseteq \{ i \mid (\mathcal{A}_i, T_i) \vDash \varphi \}$. Intuitively, in J the independence only needs to hold for the specific pair s_i, s_i' and not for all pairs. Next, from $\vec{r}\langle \mathfrak{s} \rangle = \vec{r}\langle \mathfrak{s}' \rangle$ and Lemma 5.30, it follows that

$$J_0 := \left\{ i \;\middle|\; \vec{r}\langle s_i \rangle = \vec{r}\langle s_i' \rangle \right\} \in \mathcal{U}.$$

Also, as $\mathfrak{s}, \mathfrak{s}' \in \mathfrak{T}$, the sets $J_1 := \{ i \mid s_i \in T_i \}$ and $J_2 := \{ i \mid s_i' \in T_i \}$ are in $\mathcal{U}$. Hence

$$J' := J_0 \cap J_1 \cap J_2 = \left\{ i \;\middle|\; \exists s_i'' \in T_i : (\vec{p}\vec{r}\langle s_i \rangle = \vec{p}\vec{r}\langle s_i'' \rangle \text{ and } \vec{q}\langle s_i' \rangle = \vec{q}\langle s_i'' \rangle) \right\} \in \mathcal{U}.$$

Given the witnesses s_i'' quantified in this set, we extend these to a family $\mathfrak{s}'' = (s_i'')_{i \in I}$ such that s_i'' is arbitrary for $i \notin J'$, but for $i \in J'$ it holds that $s_i'' \in T_i$, $\vec{p}\vec{r}\langle s_i \rangle = \vec{p}\vec{r}\langle s_i'' \rangle$, and finally $\vec{q}\langle s_i' \rangle = \vec{q}\langle s_i'' \rangle$. It is easy to see that $\mathfrak{s}'' \in \mathfrak{T}$, $\vec{p}\vec{r}\langle \mathfrak{s} \rangle = \vec{p}\vec{r}\langle \mathfrak{s}'' \rangle$ and $\vec{q}\langle \mathfrak{s}' \rangle = \vec{q}\langle \mathfrak{s}'' \rangle$. As $\mathfrak{s}, \mathfrak{s}'$ were arbitrary, $(\mathfrak{A}, \mathfrak{T}) \vDash \varphi$. □

5.3.4 Application: The compactness theorem

We present an application of the ultraproduct theorem, namely a compactness theorem for team logic that does not rely on translation to SO(∃) or a similar logic. The proof is standard for first-order logic (e.g., Chang and Keisler [14, Cor. 4.1.11]). Below, we adapt it to team semantics.

Theorem 5.40 (Compactness theorem)**.** *Let Φ be a set of formulas that are preserved under ultraproducts. Then Φ is satisfiable if every finite subset of Φ is satisfiable.*

Proof. Let $I := \wp^{<\omega}\Phi$, i.e., I is the set of finite subsets of Φ. Assume that every finite subset of Φ is satisfiable. Then there exists a family $(\mathcal{A}_i, T_i)_{i \in I}$ of models such that $(\mathcal{A}_i, T_i) \vDash i$ for all $i \in I$. If we now find an ultrafilter $\mathcal{U}$ on I that for all $\varphi \in \Phi$ contains the subset $\{ i \mid (\mathcal{A}_i, T_i) \vDash \varphi \}$, then by the assumption that all $\varphi \in \Phi$ are preserved in ultraproducts, $(\mathfrak{A}, \mathfrak{T}) \vDash \Phi$.

We obtain $\mathcal{U}$ as follows. For each $i \in I$, let $i^{\uparrow} := \{ i' \in I \mid i' \supseteq i \}$. We apply Proposition 5.27 to the set $\mathcal{F} := \left\{ i^{\uparrow} \;\middle|\; i \in I \right\} \subseteq \wp I$, for which we have to show the finite intersection property. But any finite intersection $i_1^{\uparrow} \cap \cdots \cap i_n^{\uparrow}$ is non-empty, since $i := i_1 \cup \cdots \cup i_n \in I$, and so $i \in i^{\uparrow} = (i_1 \cup \cdots \cup i_n)^{\uparrow} = i_1^{\uparrow} \cap \cdots \cap i_n^{\uparrow}$.

Next, we ensure that $\{ i \mid (\mathcal{A}_i, T_i) \vDash \varphi \}$ is in $\mathcal{U}$ for all $\varphi \in \Phi$. For this, it suffices to show that $\{\varphi\}^{\uparrow} \subseteq \{ i \mid (\mathcal{A}_i, T_i) \vDash \varphi \}$, since $\{\varphi\}^{\uparrow} \in \mathcal{F} \subseteq \mathcal{U}$, and $\mathcal{U}$ is upward closed. For showing that $\{\varphi\}^{\uparrow} \subseteq \{ i \mid (\mathcal{A}_i, T_i) \vDash \varphi \}$, suppose $i \in \{\varphi\}^{\uparrow}$. Then $\{\varphi\} \subseteq i$. But we assumed at the beginning that $(\mathcal{A}_i, T_i)$ is a model of i. Consequently, also $(\mathcal{A}_i, T_i) \vDash \varphi$. □

Corollary 5.41. *$\mathsf{FO}(\sim)$ satisfies the compactness theorem, i.e., if $\Phi \subseteq \mathsf{FO}(\sim)$ is unsatisfiable, then already some finite $\Phi' \subseteq \Phi$ is unsatisfiable.*

As an example for a dependence logic formula that is beyond the power of first-order logic, Väänänen [135] gave the $\mathsf{FO}(\mathsf{dep})$-sentence

$$\varphi_\infty = \exists c \forall x \exists y \forall z \exists w (\mathsf{dep}(z; w) \wedge c \neq y \wedge (x = z \leftrightarrow y = w))$$

which states that a structure is (Dedekind-)infinite, i.e., its universe is in bijection with a proper subset. The set $\{\textsc{ne}, {\sim}\varphi_\infty\} \cup \{\psi_n \mid n \geqslant 1\}$, where ψ_n states that the structure has at least n elements, is unsatisfiable, but each finite subset is satisfiable. Hence it is not compact. With Corollary 5.38, this proves that a Boolean combination of FO-formulas and non-classical atoms cannot define φ_∞, we *must* nest them inside $\vee$ or quantifiers in order to define φ_∞, as demonstrated above.

5.4 Summary and outlook

5.4.1 Summary

In this chapter, we settled the computational complexity of $\mathsf{FO}(\sim)$ and identified decidable fragments (see Figure 5.3 and Table 5.4). In terms of decidability, team logic in a sense mirrors classical logic. We showed that $\mathsf{FO}(\sim)$ is recursively enumerable, just like FO, and that both its two-variable fragment $\mathsf{FO}^2(\sim)$ and its guarded fragment $\mathsf{GF}(\sim)$ are decidable, similarly to FO^2 and GF. Our method of proof was the translation into an equivalent—albeit non-elementarily longer—form called ($\otimes\wedge$)-normal form. Moreover, $\mathsf{FO}(\sim)$ has compactness, just like FO, and unlike $\mathsf{FO}(\mathsf{dep}, \sim)$.

All in all, one could argue that $\mathsf{FO}(\sim)$ is just a non-elementarily more succinct encoding of first-order logic, and that it is perhaps closer related with FO than with $\mathsf{FO}(\mathsf{dep}, \sim)$.

This is supported by the result of Galliani [38] that the expressive power of $\mathsf{FO}(\sim)$ and FO coincides on sentences. From this point of view, however, we have at least precisely quantified the difference in succinctness.

In Table 5.4, it is worth noting that all decidable first-order logics are complete for the class Tower(poly). Subtle differences in the classical realm, such as GF^2 being complete for ExpTime, FO^2 for NExpTime, and GF for 2ExpTime, vanish due to the vastly larger succinctness of team logic. In order to prove the matching lower bounds, we generalized the well-known standard translation to modal team logic, and thus utilized the lower bounds we proved in Chapter 4.

Finally, we transferred the celebrated theorem by Łoś, i.e., that first-order theories of models are preserved in ultrapowers, to team logic. By this, we also gave an alternative proof that $\mathsf{FO}(\sim)$ satisfies the compactness theorem, even if non-classical atoms are added, as long as they are only inside Boolean connectives.

5.4.2 Open problems and further research directions

Expressiveness. We showed that the logics $\mathsf{ML}(\sim)$, $\mathsf{GF}^2(\sim)$, $\mathsf{FO}^2(\sim)$ and $\mathsf{GF}(\sim)$ all have the same complexity, that is, they are Tower(poly)-complete. However, a separation of these logics in terms of expressiveness would be desirable. Can we separate $\mathsf{ML}(\sim)$, $\mathsf{GF}(\sim)$ and $\mathsf{FO}(\sim)$ in the same way as ML, GF and FO, by a notion similar to (guarded) bisimulation as defined by Andréka et al. [4]? Specifically, can we lift these bisimulation relations to teams analogously to modal team-bisimulation (Definition 2.27)?

Standard translations and loosely guarded formulas. Analogously to our standard translation from $\mathsf{ML}(\sim)$ to $\mathsf{GF}^2(\sim)$, it would be interesting to embed other logics, such as team-logical linear temporal logic (LTL) [91] or computation tree logic (CTL) [90], into fragments of $\mathsf{FO}(\sim)$.

Classically, the translation of LTL into FO is similar to the standard translation of ML: Assuming a linear order $\leqslant$, for example the translation of $\mathsf{F}\varphi$ is $\exists y\,(x \leqslant y \wedge \mathsf{st}_y(\varphi))$. Likewise, $\varphi \mathsf{U} \psi$ is translated to $\exists y\,(x \leqslant y \wedge \mathsf{st}_y(\psi) \wedge \forall z((x \leqslant z \wedge z < y) \to \mathsf{st}_z(\varphi)))$. However, the latter is not guarded since $(x \leqslant z \wedge z < y)$ is not atomic. Van Benthem [8] defined the *loosely guarded fragment* LGF where certain conjunctions of atomic formulas are allowed as guards. LGF contains LTL, and Grädel [45] proved that this fragment has an 2ExpTime-complete satisfiability problem, just like GF.

In future research, we could define a team analog $\mathsf{LGF}(\sim)$, which presumably is complete for Tower(poly) as well. This could yield results also for $\mathsf{LTL}(\sim)$, that is, LTL with team semantics and added negation, of which the complexity of both the model checking problem and the satisfiability problem are open under asynchronous semantics [91]. Finally, the result on $\mathsf{GF}(\sim)$ could serve as an upper bound for a *polyadic* modal team logic which does not translate into FO^2 but into GF (see, e.g., Goranko and Otto [43]).

Ultraproducts and model theory. Besides our ultraproduct construction, which results from model theory else carry over to the team setting? For example, it is known that $\mathsf{FO}(\mathsf{dep})$ is compact [135], since existential second-order logic is. However, it cannot be preserved in ultraproducts: If a formula is preserved, then so is its negation, but the negated $\mathsf{FO}(\mathsf{dep})$-formula

$$\sim\exists c \forall x \exists y \forall z \exists w (\mathsf{dep}(z; w) \land c \neq y \land (x = z \leftrightarrow y = w))$$

is not preserved, since it expresses that the universe is finite (whereas the ultraproduct of infinitely many finite but unbounded structures is infinite).

That being said, we showed that Boolean combinations of $\mathsf{FO}(\sim)$-formulas and first-order definable atoms of dependency are preserved in ultraproducts, so the crucial difference is the power of nesting the atoms inside quantifiers and splitting disjunctions. It would be interesting to find the exact boundary of where Łoś's theorem fails.

Also, are there other approaches for proving, say, the compactness of $\mathsf{FO}(\mathsf{dep})$ and related logics, without relying on translations to second-order logic?

6 An axiomatization of team logic

In this chapter, we turn to the question of *axiomatization* of team logic, that is, finding a set of rules from which we can derive all true formulas. In practice, one wants this rule set to be small, natural, obvious and simple. In this thesis, we focus on *Hilbert-style proof systems*. For an introduction to this area, we refer the reader to van Dalen [21].

This chapter is organized as follows. In Section 6.1, we start with an introduction to proof systems and remind the reader of the different existing system for classical propositional, modal and first-order logic. Afterwards, in Section 6.2, we switch to team semantics and first consider the Boolean combinations (via $\wedge$ and $\sim$) of classical formulas. We call this fragment $\mathcal{B}(\mathcal{F})$, for a classical logic $\mathcal{F}$. It was already investigated in Sections 3.7, 4.9 and 5.1; we return to it now and show how a proof system for $\mathcal{F}$ can be adapted to one for $\mathcal{B}(\mathcal{F})$.

The full logics $\mathsf{PL}(\sim)$, $\mathsf{ML}(\sim)$ and $\mathsf{FO}(\sim)$ are handled in the subsequent Section 6.3. Essentially, the idea is that the connectives $\vee$, $\Box$, $\Diamond$, $\forall$, and $\exists$ can be *eliminated* from formulas, which again leads to the tractable fragment $\mathcal{B}(\mathcal{F})$. We proved similar results already in Corollary 3.92 and Theorem 5.1. Here, we extend this result by showing that the elimination can be carried out *inside* our proof system. As then every formula can be translated to one in $\mathcal{B}(\mathcal{F})$ (with $\mathcal{F} \in \{\mathsf{PL}, \mathsf{ML}, \mathsf{FO}\}$ accordingly), the completeness of a proof system for a team logic boils down to that of $\mathcal{B}(\mathcal{F})$, which ultimately relies on a system for $\mathcal{F}$. Figure 6.1 visualizes this approach.

6.1 Introduction

A *logic* $\mathcal{L}$ is a triple $(\Phi_{\mathcal{L}}, \mathfrak{A}_{\mathcal{L}}, \vDash_{\mathcal{L}})$, where $\Phi_{\mathcal{L}}$ is a set called *formulas* of $\mathcal{L}$, $\mathfrak{A}_{\mathcal{L}}$ is a class called *valuations*, and $\vDash_{\mathcal{L}}$ is the *satisfaction relation* between $\mathfrak{A}_{\mathcal{L}}$ and $\Phi_{\mathcal{L}}$.

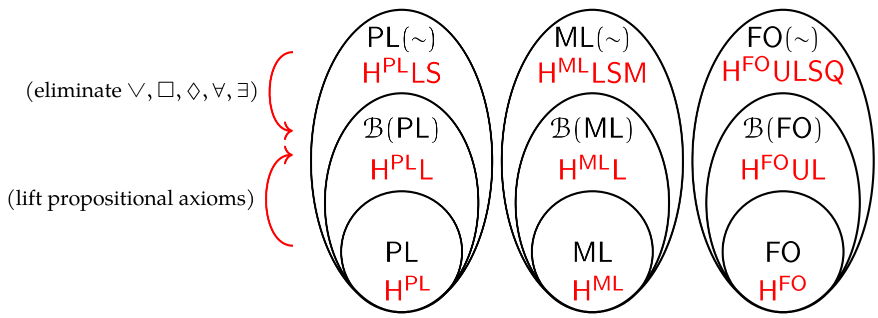


Figure 6.1: Axiomatization of $\mathsf{PL}(\sim)$, $\mathsf{ML}(\sim)$ and $\mathsf{FO}(\sim)$.

For example, in modal team logic, $\Phi_{\mathcal{L}}$ is the set of $\mathsf{ML}(\sim)$-formulas, $\mathfrak{A}_{\mathcal{L}}$ is the class of pairs $(\mathcal{K}, T)$ where $\mathcal{K}$ is a Kripke structure and T a team in $\mathcal{K}$, and $\vDash_{\mathcal{L}}$ is defined as in Chapter 2. In what follows, we often omit $\mathcal{L}$ as a subscript. Sometimes we write $\varphi \in \mathcal{L}$ and $\Phi' \subseteq \mathcal{L}$ to mean $\varphi \in \Phi_{\mathcal{L}}$ and $\Phi' \subseteq \Phi_{\mathcal{L}}$.

For sets $\Phi', \Phi'' \subseteq \mathcal{L}$, the notation $\Phi' \vDash \Phi''$ means that $\mathcal{A} \vDash \Phi'$ implies $\mathcal{A} \vDash \Phi''$ for all $\mathcal{A} \in \mathfrak{A}$. Likewise, $\Phi' \equiv \Phi''$ means $\Phi' \vDash \Phi''$ and $\Phi'' \vDash \Phi'$. If Φ' and/or Φ'' is a single formula, we omit the braces and write, e.g., $\Phi' \vDash \varphi$ instead of $\Phi' \vDash \{\varphi\}$.

Definition 6.1. A *proof system* is a triple $\Omega = (\Xi, \Psi, \mathfrak{I})$ where Ξ is a set of *judgments* (usually formulas), $\Psi \subseteq \Xi$ is a set of *axioms*, and $\mathfrak{I} \subseteq \wp^{<\omega}(\Xi) \times \Xi$ is a set of *inference rules*.

We often depict an inference rule $(\Phi', \varphi) \in \mathfrak{I}$ as a bar with the premises $\varphi_1, \ldots, \varphi_n \in \Phi'$ on top of it and the conclusion φ below it. An example for an inference rule is the well-known *modus ponens*:

$$\frac{\alpha \qquad \alpha \to \beta}{\beta}$$

In this chapter, Ξ, Ψ and $\mathfrak{I}$ are all assumed countable and decidable in polynomial time. The component-wise union of two proof systems Ω, Ω' is written $\Omega\Omega'$.

Definition 6.2. Let $\Omega = (\Xi, \Psi, \mathfrak{I})$ be a proof system and $\Phi \subseteq \Xi$. An *Ω-proof $\mathcal{P}$ from* Φ is a finite sequence $\mathcal{P} = (P_0, \ldots, P_n)$ of finite sets $P_i \subseteq \Xi$ such that $\xi \in P_i$ implies that either $\xi \in P_{i-1} \cup \Psi \cup \Phi$, or $(P, \xi) \in \mathfrak{I}$ for some $P \subseteq P_{i-1}$.

In other words, every formula in $P_i \setminus P_{i-1}$ is either an axiom, an premise from Φ, or is derived by some rule in $\mathfrak{I}$ from P_{i-1}.

We say that the proof $\mathcal{P} = (P_1, \ldots, P_n)$ *proves* or *derives* a formula φ from Φ if $\varphi \in P_n$ and $\mathcal{P}$ is an Ω-proof from Φ. We write $\Phi \vdash_\Omega \varphi$ if there is some Ω-proof of φ from Φ. We omit Ω if it is understood.

Definition 6.3 (Theorem)**.** If $\Omega = (\Xi, \Psi, \mathfrak{I})$ is a proof system, $\varphi \in \Xi$ and $\emptyset \vdash \varphi$, then φ is called *theorem* of Ω.

Instead of $\emptyset \vdash \varphi$, we also write $\vdash \varphi$.

If two formulas φ and φ' prove each other, i.e., $\{\varphi\} \vdash \varphi'$ and $\{\varphi'\} \vdash \varphi$, then we write $\varphi \dashv\vdash \varphi'$ and say that φ and φ' are *provably equivalent*.

Definition 6.4. Let Ω be a proof system and $\mathcal{L}$ a logic. Then Ω is *sound* for $\mathcal{L}$ if for all $\Phi \subseteq \mathcal{L}$ and $\varphi \in \mathcal{L}$ it holds that $\Phi \vdash_\Omega \varphi$ implies $\Phi \vDash_{\mathcal{L}} \varphi$. Moreover, Ω is *complete* for $\mathcal{L}$ if for all such Φ and φ it holds that conversely $\Phi \vDash_{\mathcal{L}} \varphi$ implies $\Phi \vdash_\Omega \varphi$.

A sound and complete proof system for $\mathcal{L}$ is an *axiomatization* of $\mathcal{L}$.

We use the classical proof systems depicted in Table 6.2. This variant of the propositional calculus ((H1)–(H9) and (E→)) goes back to Łukasiewicz [109]. We refer to it as H^{PL}. The modal logic part ((H1)–(H9), (K1)–(K2), (E→), (Nec)) is standard, see, e.g., Fitting [33]. We call this system H^{ML}. The notation "(α theorem)" means that the rule

(H1)	$\alpha \to (\beta \to \alpha)$	
(H2)	$(\alpha \to (\beta \to \gamma)) \to (\alpha \to \beta) \to (\alpha \to \gamma)$	
(H3)	$(\neg\alpha \to \neg\beta) \to (\beta \to \alpha)$	
(H4)	$\alpha \to (\beta \to (\alpha \wedge \beta))$	
(H5)	$(\alpha \wedge \beta) \to \alpha$	
(H6)	$(\alpha \wedge \beta) \to \beta$	
(H7)	$\alpha \to (\alpha \vee \beta)$	
(H8)	$\beta \to (\alpha \vee \beta)$	
(H9)	$(\alpha \to \gamma) \to (\beta \to \gamma) \to ((\alpha \vee \beta) \to \gamma)$	
(E→)	$\dfrac{\alpha \qquad \alpha \to \beta}{\beta}$	
(K1)	$\Diamond\alpha \leftrightarrow \neg\Box\neg\alpha$	
(K2)	$\Box(\alpha \to \beta) \to (\Box\alpha \to \Box\beta)$	
(Nec)	$\dfrac{\alpha}{\Box\alpha}$	(α theorem)
(H10)	$\alpha^x_t \to \exists x\alpha$	(t term)
(H11)	$\forall x\alpha \to \alpha^x_t$	(t term)
(H12)	$x = x$	
(H13)	$x = y \to (\alpha \to \alpha^x_y)$	
(G∀)	$\dfrac{\alpha \to \beta}{\alpha \to \forall x\, \beta}$	($x \notin \mathsf{Fr}(\alpha)$)
(G∃)	$\dfrac{\alpha \to \beta}{\exists x\, \alpha \to \beta}$	($x \notin \mathsf{Fr}(\beta)$)

Table 6.2: Hilbert-style axiomatizations of PL, ML and FO.

may only be applied to α that are derived without any assumptions, i.e., are theorems. It is well-known that $\Box\alpha$ is valid (i.e., true in all models) if α is valid, but $\Box\alpha$ is not necessarily true if α is true. We will come back to this complication in Subsection 6.2.1. For now, observe that it can easily be encoded whether a formula is a theorem, e.g., by choosing $\{0, 1\} \times \mathsf{ML}$ as the set of judgments, and flipping $(0, \varphi)$ to $(1, \varphi)$ whenever a non-axiom premise is used in a proof, i.e., the first component is a "dirty bit".

For the first-order axioms ((H1)–(H13), (E→), (G∀), (G∃)), we follow Hodges [69], and call this system H^{FO}.[1]

In each of these logics, we can assume that $\top$, $\bot$, $\to$ and $\leftrightarrow$ are the usual abbreviations via $\wedge$, $\vee$ and $\neg$.

Proposition 6.5. *In classical semantics,* H^{PL} *is sound and complete for* PL*,* H^{ML} *is sound and complete for* ML*, and* H^{FO} *is sound and complete for* FO*.*

[1]The notation α^x_t in (H10)–(H11) means that all free occurrences of x are replaced by the term t, but in such a way that all variables introduced via t are still free. This is achieved by renaming all bound variables into whose scope t falls.

(L1)	$\varphi \multimap (\psi \multimap \varphi)$
(L2)	$(\varphi \multimap (\psi \multimap \theta)) \multimap (\varphi \multimap \psi) \multimap (\varphi \multimap \theta)$
(L3)	$(\sim\varphi \multimap \sim\psi) \multimap (\psi \multimap \varphi)$
(L4)	$(\varphi \wedge \psi) \multimap \varphi$
(L5)	$(\varphi \wedge \psi) \multimap \psi$
(L6)	$\varphi \multimap (\psi \multimap (\varphi \wedge \psi))$
(L7)	$(\alpha \to \beta) \multimap (\alpha \multimap \beta)$
(E$\multimap$)	$\dfrac{\varphi \qquad \varphi \multimap \psi}{\psi}$

Table 6.3: The system L of lifted propositional axioms for $\mathcal{B}(\mathcal{F})$; α and β denote classical $\mathcal{F}$-formulas.

Recall that the logics PL, ML and FO all enjoy the *flatness* property (Propositions 2.10 and 2.23). Flatness has one particularly useful consequence regarding proof systems:

Proposition 6.6. *Let $\mathcal{F} \in \{\mathsf{PL}, \mathsf{ML}, \mathsf{FO}\}$, $\Gamma \subseteq \mathcal{F}$, $\alpha \in \mathcal{F}$. Then $\Gamma \vDash \alpha$ holds in classical semantics if and only if it holds in team semantics.*

Proof. We prove only the FO case, as the others are similar. For "$\Rightarrow$", let $\Gamma \vDash \alpha$ in classical semantics. Let $(\mathcal{A}, T)$ be arbitrary such that $(\mathcal{A}, T)$ satisfies Γ. Then $(\mathcal{A}, s) \vDash \Gamma$ for all $s \in T$ by flatness. By assumption, $(\mathcal{A}, s) \vDash \alpha$ in for all $s \in T$. Consequently, $(\mathcal{A}, T) \vDash \alpha$ again by flatness.

Next, we prove "$\Leftarrow$" by contraposition. If $\Gamma \nvDash \alpha$ in classical semantics, then there is a valuation $(\mathcal{A}, s)$ such that $(\mathcal{A}, s) \vDash \Gamma$ and $(\mathcal{A}, s) \nvDash \alpha$. But then also $(\mathcal{A}, \{s\}) \vDash \Gamma$ and $(\mathcal{A}, \{s\}) \nvDash \alpha$ by flatness. Consequently, $\Gamma \nvDash \alpha$ in team semantics. ☐

Corollary 6.7. *The systems H^{PL}, H^{ML} and H^{FO} are sound and complete for* PL, ML *and* FO *in team semantics, respectively.*

6.2 Axioms of the Boolean connectives

The second step towards an axiomatization of team logic is to investigate the Boolean operators $\wedge$ and $\sim$, which are added on top of a given classical logic $\mathcal{F}$. The other Boolean connectives are defined as abbreviations, besides $\varphi \varovee \psi := \sim(\sim\varphi \wedge \sim\psi)$ these are the material implication $\varphi \multimap \psi := \sim\varphi \varovee \psi$, the equivalence $\varphi \multimapboth \psi := (\varphi \multimap \psi) \wedge (\psi \multimap \varphi)$ and the strict falsum $\bot\!\!\!\bot := \sim\top$.

We again consider the Boolean closure $\mathcal{B}(\mathcal{F})$, and generalize the definition from that in Section 4.9.

Definition 6.8. Let $\mathcal{F}$ be a logic. Then $\mathcal{B}(\mathcal{F})$ is the logic called the *Boolean closure* of $\mathcal{F}$, with its formulas given by the grammar $\varphi ::= \alpha \mid \sim\varphi \mid \varphi \wedge \varphi$, for $\alpha \in \mathcal{F}$, with the same

A	$\xi \multimap \alpha$	
B	$\xi \multimap (\alpha \to \beta)$	
1	$(\alpha \to \beta) \multimap (\alpha \multimap \beta)$	(L7)
2	$\xi \multimap ((\alpha \to \beta) \multimap (\alpha \multimap \beta))$	(L1), 1
3	$\xi \multimap (\alpha \multimap \beta)$	(L2), B, 2
$\triangleright$	$\xi \multimap \beta$	(L2), A, 3

Figure 6.4: Example derivation in L

valuations as $\mathcal{F}$, and with the semantics

$$A \vDash_{\mathcal{B}(\mathcal{F})} \varphi \;\Leftrightarrow\; \begin{cases} A \vDash_{\mathcal{F}} \varphi & \text{if } \varphi \in \mathcal{F}, \\ A \nvDash_{\mathcal{B}(\mathcal{F})} \psi & \text{if } \varphi \notin \mathcal{F} \text{ and } \varphi = {\sim}\psi, \\ A \vDash_{\mathcal{B}(\mathcal{F})} \psi_1 \text{ and } A \vDash_{\mathcal{B}(\mathcal{F})} \psi_2 & \text{if } \varphi \notin \mathcal{F} \text{ and } \varphi = \psi_1 \wedge \psi_2. \end{cases}$$

On the axiom side, the proof system L shown in Table 6.3 mainly consists of propositional axioms "L"ifted to team logic. The classical axioms (L1) to (L3) and (E$\multimap$) describe the meaning of $\multimap$ and $\sim$. The axioms (L4) to (L6) define $\wedge$. The only "non-classical" axiom is (L7), which is necessary to relate the implication on the level of singletons ($\to$) with that on the level of teams ($\multimap$).

Derivations are written down as in the example below (Figure 6.4). The premises have the special line numbers A, B, ..., whereas $\triangleright$ marks the conclusion. The right column of each proof shows the applied rules with the line numbers of the arguments. The format is

$$(\mathsf{rule}_1), \ldots, (\mathsf{rule}_n),\ \mathsf{argument}_1, \ldots, \mathsf{argument}_n$$

where omitted line numbers of the arguments means that the preceding lines are used. For brevity, we omit applications of (E$\multimap$) in L that are clear.

In the next several subsections, we prove that L, when combined with a proof system for $\mathcal{F}$, completely axiomatizes $\mathcal{B}(\mathcal{F})$. First, we show that L also preserves soundness.

Lemma 6.9. *Let $\Omega = (\Xi, \Psi, \mathfrak{I})$ be a proof system such that every rule and axiom of Ω contains only $\mathcal{F}$-formulas, that is, $\Xi \subseteq \mathcal{F}$. If (E$\to$) $\in \mathfrak{I}$ and Ω is sound for $\mathcal{F}$, then ΩL is sound for $\mathcal{B}(\mathcal{F})$.*

Proof. We show that all axioms and inference rules of ΩL are sound. Then the soundness of ΩL is easily shown by induction on the length of proofs.

The axioms and rules of Ω apply only to $\mathcal{F}$, and for this reason are sound by assumption. As (E$\to$) is also sound, $\{\alpha, \alpha \to \beta\} \vDash \beta$ for all $\alpha, \beta \in \mathcal{F}$. For this reason, $\alpha \to \beta \vDash \alpha \multimap \beta$, so (L7) is sound. For the other axioms and rules of L this is straightforward by the semantics of $\sim$, $\wedge$ and $\multimap$. □

6.2.1 The deduction theorem for team logics

Similar to propositional logic, the first step in the completeness proof is the deduction theorem, i.e., that $\Phi \vdash (\varphi \multimap \psi)$ if and only if $\Phi \cup \{\varphi\} \vdash \psi$. Unfortunately, for logics beyond the propositional connectives, the status of the deduction theorem is rather unclear. The prime example is modal logic, for which Hakli and Negri [50] claim:

> *For modal logic, however, there seems to be lack of agreement about the validity of the deduction theorem. The answer to the question whether the deduction theorem fails for modal logic is far from unanimous. Some sources in the literature claim that the deduction theorem holds, whereas others claim that it fails, some give conditions and restrictions for the theorem to hold or argue for the failure of the deduction theorem as a consequence of a certain formulation of the rule of necessitation.*

Classical modal logic consists of the propositional calculus together with the distribution axiom $\Box(\varphi \to \psi) \to (\Box\varphi \to \Box\psi)$ and the Gödel rule (necessitation): $\varphi \vdash \Box\varphi$. But clearly $\nvdash \varphi \to \Box\varphi$, which is the said *failure of the deduction theorem*. Basically, the deduction theorem is incompatible with the necessitation rule. On this account, some authors claim only *weak* soundness and completeness of an axiom system for modal logic, meaning $\vdash \varphi \Leftrightarrow \vDash \varphi$ (cf. Sider [129]).

Instead, we pursue the approach of Hakli and Negri [50], which goes back to Fitting (cf., e.g., [33]). Essentially, they restrict the application of $\varphi \vdash \Box\varphi$ to those cases where φ is a theorem (cf. p. 150). This approach leaves the calculus complete and preserves the deduction theorem with only minor changes to the classical proof systems.

Since team logic contains several non-Boolean connectives, we generalize the proof of Hakli and Negri [50, Thm. 2]. As the crucial property of a rule to be compatible with the deduction theorem, we identify the following.

Definition 6.10. Let $\Omega = (\Xi, \psi, \mathfrak{I})$ be a proof system. A rule $(\{\xi_1, \ldots, \xi_k\}, \psi) \in \mathfrak{I}$ *has conditioning* if $\{\, \varphi \multimap \xi_1, \ldots, \varphi \multimap \xi_k \,\} \vdash (\varphi \multimap \psi)$ for all $\varphi \in \Xi$.

In other words, the rule can be applied relative to an arbitrary premise φ. It is needless to say that we eventually will prove the above property for all rules of the corresponding team-logical connectives.

We say that a system Ω has conditioning if all inference rules have it.

Theorem 6.11 (Deduction theorem)**.** *If Ω is a proof system and $\Omega\mathsf{L}$ has conditioning, then $\Phi \vdash_{\Omega\mathsf{L}} (\varphi \multimap \psi)$ if and only if $\Phi \cup \{\varphi\} \vdash_{\Omega\mathsf{L}} \psi$.*

Proof. "$\Rightarrow$" is clear, as L has ($\mathsf{E}{\multimap}$). We prove "$\Leftarrow$" by induction on the length n of a shortest proof of ψ. If $\psi \in \Phi$, $\psi = \varphi$, or if ψ is an axiom, then $\Phi \vdash (\varphi \multimap \psi)$ by (L1) and ($\mathsf{E}{\multimap}$). For $n = 1$ these are the only cases. If $n > 1$, then ψ could also be obtained by application of some inference rule $(\{\xi_1, \ldots, \xi_k\}, \psi)$. But then $\xi_1, \ldots, \xi_k$ each have a proof of length $\leqslant n - 1$ from $\Phi \cup \{\varphi\}$, so by induction hypothesis, $\Phi \vdash \varphi \multimap \xi_i$ for $1 \leqslant i \leqslant k$. As $\Omega\mathsf{L}$ has conditioning by assumption, $\Phi \vdash \varphi \multimap \psi$ as desired. $\Box$

This for example applies to the rule $\varphi \vdash \Box\varphi$ in its weakened form (requiring φ to be a theorem), as we will show next.

Definition 6.12. Let Ω and Ω' be proof systems. Ω' is a *conservative extension of* Ω, in symbols $\Omega' \succeq \Omega$, if Ω' contains all judgments, rules, and axioms of Ω, but all rules of Ω' that are not in Ω apply only to theorems.

Theorem 6.13. *Every conservative extension of* L *or* $\mathsf{H}^{\mathsf{PL}}\mathsf{L}$ *has the deduction theorem.*

Proof. Let Ω be a proof system as above. It suffices to show that all rules of Ω have conditioning.

There are three cases to distinguish: ($\mathsf{E}\!\rightarrow$), ($\mathsf{E}\!\multimap$), and rules that apply only to theorems. The latter case is clear: If a rule (Φ', φ) applies only to theorems, then the produced formula φ itself is a theorem. Then by (L1) and ($\mathsf{E}\!\multimap$) we can prove $\xi \multimap \varphi$ for arbitrary ξ, so the rule certainly has conditioning.

Next, consider the rule ($\mathsf{E}\!\multimap$), i.e., $(\{\varphi, \varphi \multimap \psi\}, \psi)$. To demonstrate that it has conditioning, we assume the premises $\xi \multimap (\varphi \multimap \psi)$ and $\xi \multimap \varphi$, where ξ is arbitrary. By (L2) and ($\mathsf{E}\!\multimap$), it is straightforward to derive $\xi \multimap \psi$. Finally, for ($\mathsf{E}\!\rightarrow$), conditioning is proved as in Figure 6.4. $\Box$

For team logic, we will later introduce rules like $\varphi \vdash \Box\varphi$, $\varphi \vdash \forall x\varphi$ etc. that act only on theorems, and thereby fulfill the above requirement.

6.2.2 Completeness of L

We follow the standard completeness proof for propositional logic, which relies on Lindenbaum's lemma to construct a so-called maximal consistent set. Let us first introduce an analogous notion of consistency. In what follows, let $\Omega = (\Xi, \psi, \mathfrak{I})$ be a proof system.

Definition 6.14. A set $\Phi \subseteq \Xi$ is Ω-*inconsistent* if $\Phi \vdash \Xi$, i.e., if everything can be derived. Φ is Ω-*consistent* if it is not Ω-inconsistent. Moreover, $\Phi \subseteq \Xi$ is *maximal* Ω-*consistent* if it is Ω-consistent and contains ξ or $\sim\xi$ for every $\xi \in \Xi$.

As before, we usually omit Ω. The following lemmas are standard, with their proofs also found in the appendix.

Lemma 6.15. *Let* $\Omega \succeq \mathsf{L}$. *The following statements are equivalent:*

(1) $\Phi \vdash_\Omega \varphi$ *and* $\Phi \vdash_\Omega \sim\varphi$ *for some* φ,

(2) Φ *is* Ω*-inconsistent,*

(3) $\Phi \vdash_\Omega \perp\!\!\!\perp$.

Lemma 6.16. *Let* $\Omega \succeq \mathsf{L}$ *and let* Φ *be consistent. Then* $\Phi \nvdash_\Omega \varphi$ *implies that* $\Phi \cup \{\sim\varphi\}$ *is* Ω*-consistent, and* $\Phi \vdash_\Omega \varphi$ *implies that* $\Phi \cup \{\varphi\}$ *is* Ω*-consistent.*

Lemma 6.17 (Lindenbaum's lemma)**.** *If* $\Omega \succeq \mathsf{L}$, *then every* Ω*-consistent set has a maximal* Ω*-consistent superset.*

The next step in standard completeness proofs is to construct a model for any maximal consistent set. The application of Lindenbaum's lemma is usually as follows: if Φ is maximal consistent, then there is a model M fulfilling all its atomic formulas. By the maximality of Φ, then also all Boolean combinations of atomic formulas, if they are in Φ, are true in M.

The part where we have to deviate from the standard proof is in fact the induction basis, since at the bottom we do not have atomic propositions, but $\mathcal{F}$-formulas as "atoms". For this reason, a bit more work will be required.

Let $\sim\mathcal{F}$ denote the fragment of $\mathcal{B}(\mathcal{F})$ that is restricted to the formulas in $\{ \sim\alpha \mid \alpha \in \mathcal{F} \}$. Likewise, $\mathcal{F} \cup \sim\mathcal{F}$ denotes the fragment restricted to formulas in $\{ \alpha, \sim\alpha \mid \alpha \in \mathcal{F} \}$. Intuitively, $\mathcal{F} \cup \sim\mathcal{F}$ is the set of "literals."

Definition 6.18. A proof system is *refutation complete* for $\mathcal{L}$ if every unsatisfiable $\Phi \subseteq \mathcal{L}$ is inconsistent.

With an additional assumption, the standard proof goes through:

Theorem 6.19 (Completeness of L)**.** *If $\Omega \succeq \mathsf{L}$ is refutation complete for $\mathcal{F} \cup \sim\mathcal{F}$, then it is complete for $\mathcal{B}(\mathcal{F})$.*

Proof. Let $\Phi \subseteq \mathcal{B}(\mathcal{F})$ and $\varphi \in \mathcal{B}(\mathcal{F})$. For completeness, we have to show that $\Phi \nvdash \varphi$ implies $\Phi \nvDash \varphi$. If $\Phi \nvdash \varphi$, then by Lemma 6.16, $\Phi \cup \{\sim\varphi\}$ is consistent. Then $\Phi \cup \{\sim\varphi\}$ has a maximal consistent superset Φ^* by Lemma 6.17. Clearly, $\Phi^* \cap (\mathcal{F} \cup \sim\mathcal{F})$ is then consistent as well. By refutation completeness of Ω for $\mathcal{F} \cup \sim\mathcal{F}$, it has a model A. We show that $\psi \in \Phi^* \Leftrightarrow A \vDash \psi$ for all $\psi \in \mathcal{B}(\mathcal{F})$. In particular, $\Phi \cup \{\sim\varphi\}$ is then satisfiable, which proves $\Phi \nvDash \varphi$. That $\psi \in \Phi^* \Leftrightarrow A \vDash \psi$ holds for $\psi \notin (\mathcal{F} \cup \sim\mathcal{F})$ can be proven by induction on the length of ψ (see the appendix). $\square$

Why is the refutation completeness of literals an issue in team semantics? Let us consider propositional logic PL as an example. Classically, it is the Boolean closure of Prop, but the set $\{p, \neg p \mid p \in \mathsf{Prop}\}$ of literals is *trivially* refutation complete: Any subset Φ is inconsistent only if it contains $p, \neg p$ for some proposition p. Otherwise it is satisfiable simply due to the assignment s with $s(p) = 0$ iff $\neg p \in \Phi$. But full team logic now constitutes another layer on top of classical logic, in the sense that $\mathcal{B}(\mathcal{F})$ is the Boolean closure of *$\mathcal{F}$-formulas*. That means that "atoms" of team logic are not propositions, but formulas of the underlying classical logic. For this reason, refutation completeness on the level of literals becomes a non-trivial issue, whereas in classical logic it is not.

However, the case of having only propositions as atoms gives us a useful result. Let $\varphi \in \mathcal{B}(\mathsf{Prop})$. A formula φ' is a *substitution instance* of φ if there are $n \in \mathbb{N}$, propositions $p_1, \ldots, p_n$ and formulas $\psi_1, \ldots, \psi_n$ such that $\varphi' = \varphi[p_1/\psi_1] \cdots [p_n/\psi_n]$.

Theorem 6.20. *If $\vDash_{\mathcal{B}(\mathsf{Prop})} \varphi$, then $\vdash_{\mathsf{L}} \varphi'$ for any substitution instance φ' of φ.*

Example 6.21. The distributive law $a \wedge (b \otimes c) \Leftrightarrow (a \wedge b) \otimes (a \wedge c)$ is *semantically* valid. Therefore all its instances $\varphi \wedge (\psi \otimes \theta) \Leftrightarrow (\varphi \wedge \psi) \otimes (\varphi \wedge \psi)$ are *provable*.

Proof of Theorem 6.20. Let $\varphi \in \mathcal{B}(\mathsf{Prop})$ such that $\vDash_{\mathcal{B}(\mathsf{Prop})} \varphi$. First, we show that L is complete for $\mathcal{B}(\mathsf{Prop})$. For this, we apply Theorem 6.19 and show that L is trivially refutation complete for $\mathsf{Prop} \cup {\sim}\mathsf{Prop}$. The argument is similar to the one above: Any set $\Phi \subseteq \mathsf{Prop} \cup {\sim}\mathsf{Prop}$ either contains $p, {\sim}p$ for some p, or it is satisfiable due to the team $T = \{s\}$ with $s(p) = 0 \Leftrightarrow {\sim}p \in \Phi$ for all $p \in \mathsf{Prop}$.

Hence $\vdash_L \varphi$. Now, suppose that $\varphi^\star$ is a substitution instance of φ, i.e., there are n, p_i and ψ_i such that $\varphi^\star = \varphi[p_1/\psi_1]\cdots[p_n/\psi_n]$. Let $\theta^\star$ denote the same substitution applied to θ, for arbitrary formulas θ, i.e., $\theta^\star := \theta[p_1/\psi_1]\cdots[p_n/\psi_n]$.

We proceed with showing $\vdash_L \varphi^\star$ by induction on the length of a shortest proof of φ in L. If φ is an instance of (L1) to (L6), then the same is the case for $\varphi^\star$. (Being a $\mathcal{B}(\mathsf{Prop})$ formula, φ cannot be an instance of (L7).)

If φ was derived from $\psi \multimap \varphi$ and ψ via (E$\multimap$), then $\vdash_L (\psi \multimap \varphi)^\star$ and $\vdash_L \psi^\star$ by induction hypothesis. As $(\psi \multimap \varphi)^\star = \psi^\star \multimap \varphi^\star$, we can apply (E$\multimap$) to obtain $\varphi^\star$. $\square$

Corollary 6.22. *The standard propositional laws such as De Morgan's laws, distributive laws, commutative laws, etc. (over $\wedge$ and $\sim$) are all provable in* L.

6.2.3 Refutation completeness on literals

For the completeness of $\mathcal{B}(\mathcal{F})$, we required that our proof system is at least refutation complete for literals, i.e., $\mathcal{F} \cup {\sim}\mathcal{F}$. Next, several ways to establish this are presented.

Counter-model merging

Definition 6.23. A logic $\mathcal{F}$ admits *counter-model merging* if, for arbitrary sets $\Gamma, \Delta \subseteq \mathcal{F}$ the following holds: If for every $\delta \in \Delta$ there is a model M such that $M \vDash \Gamma$ and $M \nvDash \delta$, then there is a model M such that $M \vDash \Gamma$ and $M \nvDash \delta$ for all $\delta \in \Delta$.

In other words, if every $\delta \in \Delta$ is falsified by a model of Γ, then Γ also has a model that falsifies all formulas in Δ simultaneously. A similar property, the $\varovee$*-disjunction property*, was used by Virtema [139] and Yang and Väänänen [142, 143, 144]. It says that $\vDash \varphi \varovee \psi$ implies that either $\vDash \varphi$ or $\vDash \psi$. For this reason, the Boolean disjunction $\varovee$ is sometimes also known as *intuitionistic disjunction* in team logic.

Our definition speaks about entailment instead of only validity, but the proof is essentially the same as for the disjunction property in the literature. For the sake of self-containedness, we include it below.

Proposition 6.24. PL *and* ML *admit counter-model merging.*

Proof. We prove the more general case, ML. Let $\Gamma, \Delta \subseteq \mathsf{ML}$, and for each $\delta \in \Delta$, let $(\mathcal{K}_\delta, T_\delta)$ be a model of $\Gamma \cup \{{\sim}\delta\}$, where $\mathcal{K}_\delta$ is a Kripke structure and T_δ is a team in $\mathcal{K}_\delta$. W.l.o.g., the Kripke structures $\mathcal{K}_\delta$ are pairwise disjoint. Let $\mathcal{K}^*$ be the union of the $\mathcal{K}_\delta$. The truth of ML-formulas is invariant under disjoint union of structures [43]; hence $(\mathcal{K}^*, w) \vDash \alpha$ if and only if $(\mathcal{K}, w) \vDash \alpha$, for all formulas $\alpha \in \mathsf{ML}$ and $w \in T_\delta$. By flatness of ML it follows that $(\mathcal{K}^*, T_\delta) \vDash \Gamma$ and $(\mathcal{K}^*, T_\delta) \nvDash \delta$ for all $\delta \in \Delta$. Finally, consider the team

$T^* := \bigcup_{\delta \in \Delta} T_\delta$. As ML is union closed, $(\mathcal{K}^*, T^*)$ satisfies Γ, and as it is downward closed, $(\mathcal{K}^*, T^*)$ falsifies each $\delta \in \Delta$. □

Lemma 6.25. *If $\mathcal{F}$ admits counter-model merging and Ω is complete for $\mathcal{F}$, then $\Omega\mathsf{L}$ is refutation complete for $\mathcal{F} \cup {\sim}\mathcal{F}$.*

Proof. Let $\Phi \subseteq \mathcal{F} \cup {\sim}\mathcal{F}$ be unsatisfiable. Let $\Gamma := \Phi \cap \mathcal{F}$ and $\Delta := \Phi \cap {\sim}\mathcal{F}$. There exists ${\sim}\delta \in \Delta$ such that $\Gamma \cup \{{\sim}\delta\}$ is unsatisfiable, since otherwise Φ would be satisfiable by counter-model merging. But then $\Gamma \vDash \delta$, which implies $\Gamma \vdash \delta$ by completeness of Ω for $\mathcal{F}$. Consequently, $\Phi \vdash \{\delta, {\sim}\delta\}$. By Lemma 6.15, Φ is inconsistent. □

Theorem 6.26. $\mathsf{H}^{\mathsf{PL}}\mathsf{L}$ *axiomatizes* $\mathcal{B}(\mathsf{PL})$. $\mathsf{H}^{\mathsf{ML}}\mathsf{L}$ *axiomatizes* $\mathcal{B}(\mathsf{ML})$.

Proof. The soundness follows from Corollary 6.7 and Lemma 6.9. The completeness follows from Proposition 6.24, Lemma 6.25 and Theorem 6.19. □

Closure under quantification: First-order logic

First-order logic FO does not enjoy the counter-model merging property. Consider, for instance, the sentences $R(c)$ and $\neg R(c)$, where c is a constant. Clearly, either of them can be falsified by an appropriate interpretation in team semantics, but to falsify both in the same structure is impossible regardless of the assigned teams. The crucial point is that $R(c)$ and $\neg R(c)$ are contradicting *sentences*.

In this section, we show that sentences are in fact the *only* obstacle for axiomatizing $\mathcal{B}(\mathsf{FO})$. The problem can be remedied by introducing an additional axiom, the *unanimity axiom*:

(U)	${\sim}\alpha \multimap \neg\alpha$	(α sentence)

We will refer to the above system simply as U.

Similar to classical first-order logic, the truth of a sentence depends only on the underlying structure itself and not on the assignments in a given team:

Lemma 6.27. *For any sentence $\alpha \in \mathsf{FO}$ and structure $\mathcal{A}$, the following are equivalent:*

(1) $(\mathcal{A}, T) \vDash \alpha$ *for some non-empty team* T.

(2) $(\mathcal{A}, T) \vDash \alpha$ *for all teams* T.

(3) $(\mathcal{A}, s) \vDash \alpha$ *for some* $s : \mathsf{Var} \to \mathcal{A}$.

(4) $(\mathcal{A}, s) \vDash \alpha$ *for all* $s : \mathsf{Var} \to \mathcal{A}$.

Proof. Straightforward by the flatness property. □

Corollary 6.28. *The system* U *is sound for* $\mathsf{FO}({\sim})$.

We proceed by investigating the fragment $\sim\mathsf{FO} := \{\sim\alpha \mid \alpha \in \mathsf{FO}\}$. The next proposition and the subsequent lemma show that the system $\mathsf{H}^{\mathsf{FO}}\mathsf{U}$ is not only sound, but also "complete" for FO-entailments from sets of $\sim$FO-formulas:

Proposition 6.29. *Let $\Delta \subseteq \sim\mathsf{FO}$ be non-empty, and suppose that $\Delta \vDash \alpha$ for some $\alpha \in \mathsf{FO}$. Then there is an* FO*-sentence ε such that $\Delta \vDash \sim\varepsilon \vDash \neg\varepsilon \vDash \alpha$.*

Proof. Define $\varepsilon := \exists x_1 \cdots \exists x_n \neg\alpha$, where $\{x_1, \ldots, x_n\} = \mathsf{Fr}(\alpha)$. Clearly, $\neg\varepsilon \equiv \forall x_1 \cdots \forall x_n \alpha$. In particular, $\neg\varepsilon \vDash \alpha$. Moreover, $\sim\varepsilon \vDash \neg\varepsilon$ by the previous lemma.

It remains to prove $\Delta \vDash \sim\varepsilon$. Suppose $(\mathcal{A}, T) \vDash \Delta$ for some team T and first-order structure $\mathcal{A}$. Let $V = \{ s \mid s : \mathsf{Var} \to \mathcal{A} \}$ be the team of *all* assignments. Then $T \subseteq V$, and $(\mathcal{A}, V) \vDash \Delta$ by upwards closure of $\sim$FO. By assumption of the proposition, $(\mathcal{A}, V) \vDash \alpha$.

The next step is to show that $\mathcal{A} \vDash \neg\varepsilon$. Since V contains all assignments, it also contains a team of the form $U^{x_1}_{\mathcal{A}} \ldots ^{x_n}_{\mathcal{A}}$ for non-empty U, which then satisfies α by downward closure. By definition of $\forall$ in team semantics, $(\mathcal{A}, U) \vDash \forall x_1 \cdots \forall x_n\, \alpha \equiv \neg\varepsilon$. Note that $T \neq \emptyset$, as T satisfies at least one $\sim$FO-formula. By Lemma 6.27, $(\mathcal{A}, T) \vDash \sim\varepsilon$. □

The above proposition exhibits an important property of $\sim$FO: If a subset $\Delta \subseteq \sim\mathsf{FO}$ is not satisfiable, then it already entails contradicting *sentences*. This fact is exploited in the next lemma. It is the first step toward refutation completeness of the fragment $\mathsf{FO} \cup \sim\mathsf{FO}$, which is required in order to utilize Theorem 6.19 for completeness of $\mathcal{B}(\mathsf{FO})$.

Lemma 6.30. *The system $\mathsf{H}^{\mathsf{FO}}\mathsf{UL}$ is refutation complete for $\sim$FO.*

Proof. Let $\Delta \subseteq \sim\mathsf{FO}$ be unsatisfiable. Note that $\sim\delta \vdash_{\mathsf{H}^{\mathsf{FO}}\mathsf{L}} \sim\bot$ for all $\delta \in \mathsf{FO}$, as $\bot \vdash_{\mathsf{H}^{\mathsf{FO}}} \delta$. As Δ necessarily contains at least one formula, which is of the form $\sim\delta$, we have $\Delta \vdash \sim\bot$. So we only need to demonstrate $\Delta \vdash \bot$ to show it is inconsistent.

For the rest of the proof, we write $\delta(x_1, \ldots, x_n)$ to indicate that δ has the free variables $x_1, \ldots, x_n$. Then we define a set $\Gamma \subseteq \mathsf{FO}$ of sentences by

$$\Gamma := \{ \exists x_1 \cdots \exists x_n \neg\delta(x_1, \ldots, x_n) \mid \sim\delta(x_1, \ldots, x_n) \in \Delta \}.$$

The remaining proof of $\Delta \vdash \bot$ is split into showing $\Delta \vdash_{\mathsf{H}^{\mathsf{FO}}\mathsf{UL}} \Gamma$ and $\Gamma \vdash_{\mathsf{H}^{\mathsf{FO}}} \bot$. For the first part, note that $\forall x_1 \ldots \forall x_n\, \delta(x_1, \ldots, x_n) \vdash_{\mathsf{H}^{\mathsf{FO}}} \delta(x_1, \ldots, x_n)$ as H^{FO} is complete for FO. Consequently, for all $\exists x_1 \cdots \exists x_n \neg\delta(x_1, \ldots, x_n) \in \Gamma$,

$$\begin{aligned} \Delta \vdash\; & \sim\delta(x_1, \ldots, x_n) \\ \vdash_{\mathsf{H}^{\mathsf{FO}}\mathsf{L}}\; & \sim\forall x_1 \cdots \forall x_n\, \delta(x_1, \ldots, x_n) \\ \vdash_{\mathsf{U}}\; & \neg\forall x_1 \cdots \forall x_n\, \delta(x_1, \ldots, x_n) \\ \vdash_{\mathsf{H}^{\mathsf{FO}}}\; & \exists x_1 \cdots \exists x_n \neg\delta(x_1, \ldots, x_n). \end{aligned}$$

So $\Delta \vdash \Gamma$. It remains to prove $\Gamma \vdash \bot$. By classical completeness, it suffices to show that Γ is classically unsatisfiable. Hence, for the sake of contradiction, assume that Γ has a model $(\mathcal{A}, s)$. With $\mathcal{A}$ fixed, now for every formula $\exists x_1 \cdots \exists x_n \neg\delta(x_1, \ldots, x_n) \in \Gamma$ the set $S_\delta := \{ s \colon \mathsf{Var} \to \mathcal{A} \mid (\mathcal{A}, s) \vDash \neg\delta(x_1, \ldots, x_n) \}$ must then be non-empty. But then $(\mathcal{A}, S_\delta) \vDash \sim\delta(x_1, \ldots, x_n)$. By downward closure of FO, and hence upward closure of $\sim$FO, we obtain $(\mathcal{A}, \bigcup_\delta S_\delta) \vDash \Delta$, contradiction to the assumption of the lemma. □

Lemma 6.31. *The system* $\mathsf{H}^{\mathsf{FO}}\mathsf{UL}$ *is refutation complete for* $\mathsf{FO} \cup {\sim}\mathsf{FO}$.

Proof. We have to show that any unsatisfiable $\Phi \subseteq \mathsf{FO} \cup {\sim}\mathsf{FO}$ is inconsistent, so suppose Φ is unsatisfiable. We showed that $\mathsf{FO}(\sim)$ satisfies the compactness theorem (see Chapter 5, Corollary 5.41), so w.l.o.g. Φ is finite. Let $\Gamma := \Phi \cap \mathsf{FO}$ and $\Delta := \Phi \cap {\sim}\mathsf{FO}$. As Γ is finite, and by completeness of H^{FO}, w.l.o.g. $\Gamma = \{\gamma\}$ for some $\gamma \in \mathsf{FO}$.

We construct the following set $\Delta^\gamma \subseteq {\sim}\mathsf{FO}$, which "adjoins" γ to all formulas in Δ:

$$\Delta^\gamma := \{ {\sim}(\neg\gamma \vee \delta) \mid {\sim}\delta \in \Delta \} \equiv \{ \mathsf{E}(\gamma \wedge \neg\delta) \mid {\sim}\delta \in \Delta \}$$

The remainder of the proof shows that $\{\gamma\} \cup \Delta \vdash \Delta^\gamma$ and that Δ^γ is unsatisfiable. As $\mathsf{H}^{\mathsf{FO}}\mathsf{UL}$ is refutation complete for ${\sim}\mathsf{FO}$ by Lemma 6.30, then Δ^γ and consequently Φ is inconsistent. As $\{\gamma, \neg\gamma \vee \delta\} \vdash_{\mathsf{H}^{\mathsf{FO}}} \delta$, we have $\Phi \vdash \{\gamma, {\sim}\delta\} \vdash_{\mathsf{H}^{\mathsf{FO}}\mathsf{L}} {\sim}(\neg\gamma \vee \delta)$ for all ${\sim}(\delta \vee \neg\gamma) \in \Delta^\gamma$. Hence $\Phi \vdash \Delta^\gamma$.

Next, assume for the sake of contradiction that Δ^γ is satisfiable, say, in $(\mathcal{A}, T)$ for a first-order structure $\mathcal{A}$ and team T. For each ${\sim}\delta \in \Delta$, there is $s \in T$ such that $(\mathcal{A}, s) \vDash \gamma \wedge \neg\delta$. However, if $T' := \{ s \in T \mid (\mathcal{A}, s) \vDash \gamma \}$, then $(\mathcal{A}, T') \vDash \gamma$ by flatness and $(\mathcal{A}, T') \vDash \Delta$ by downward closure. Contradiction to the assumption that $\{\gamma\} \cup \Delta$ is unsatisfiable. □

Theorem 6.32. $\mathsf{H}^{\mathsf{FO}}\mathsf{UL}$ *is sound and complete for* $\mathcal{B}(\mathsf{FO})$.

Proof. The system H^{FO} is sound by Proposition 6.5, and L by Lemma 6.9. U is sound by Corollary 6.28. By Theorem 6.19 and the above lemma, $\mathsf{H}^{\mathsf{FO}}\mathsf{UL}$ is complete. □

6.3 Operator elimination

In this section, we build on top of the system L and axiomatize the other connectives, $\vee, \Diamond, \Box, \exists$ and $\forall$, in multiple steps. This yields a proof system for the respective logics $\mathcal{F}(\sim)$, where $\mathcal{F} \in \{\mathsf{PL}, \mathsf{ML}, \mathsf{FO}\}$. We prove completeness by presenting a translation of $\mathcal{F}(\sim)$ to the fragment $\mathcal{B}(\mathcal{F})$ that can be carried out in the proof system. A similar approach was pursued by Yang [141, 142] for propositional and modal dependence logic, although she translated formulas into a normal form based on $\varovee$ and $\vee$ instead of $\varovee$ and $\wedge$ (cf. Subsection 3.7.1).The following lemma formalizes our approach.

Lemma 6.33. *Let* $\mathcal{L}, \mathcal{L}'$ *be logics such that* $\mathcal{L}' \subseteq \mathcal{L}$. *Let* Ω *be a proof system that is sound for* $\mathcal{L}$ *and complete for* $\mathcal{L}'$, *and such that every* $\mathcal{L}$*-formula is provably equivalent to an* $\mathcal{L}'$*-formula in* Ω. *Then* Ω *is also complete for* $\mathcal{L}$.

Proof. Assume $\Phi \subseteq \mathcal{L}$ and $\varphi \in \mathcal{L}$. We have to show that $\Phi \vDash \varphi$ implies $\Phi \vdash \varphi$. By assumption, every $\mathcal{L}$-formula is provably equivalent to an $\mathcal{L}'$-formula, hence $\Phi \dashv\vdash \Phi'$ for some set $\Phi' \subseteq \mathcal{L}'$. Likewise, $\varphi \dashv\vdash \varphi'$ for some $\varphi' \in \mathcal{L}'$. Since these equivalences are proved between (sets of) $\mathcal{L}$-formulas, soundness for $\mathcal{L}$ implies $\Phi \equiv \Phi'$ and $\varphi \equiv \varphi'$. Consequently, $\Phi' \vDash \varphi'$. By completeness of Ω for $\mathcal{L}'$, we obtain $\Phi' \vdash \varphi'$. Altogether, then $\Phi \vdash \Phi' \vdash \varphi' \vdash \varphi$. As $\vdash$ is transitive, the lemma follows. □

(Dual$\multimap$)	$(\varphi \vee \psi) \leftrightarrow \sim(\varphi \multimap \sim\psi)$	Definition of $\multimap$
(Sub$\vee$)	$\alpha \rightarrow (\varphi \multimap \alpha)$	Downwards closure
(Lax$\vee$)	$\varphi \rightarrow (\theta \vee \psi) \rightarrow (\varphi \vee \psi)$	Lax semantics
(Ass$\vee$)	$(\varphi \vee (\psi \vee \theta)) \rightarrow ((\varphi \vee \psi) \vee \theta)$	Associative law
(Com$\vee$)	$(\varphi \vee \psi) \rightarrow (\psi \vee \varphi)$	Commutative law
(Dis$\multimap$)	$(\varphi \multimap (\psi \rightarrow \theta)) \rightarrow (\varphi \multimap \psi) \rightarrow (\varphi \multimap \theta)$	Distribution axiom
(Nec$\multimap$)	$\dfrac{\varphi}{\psi \multimap \varphi}$ (φ theorem)	Necessitation

Table 6.5: The system S, splitting axioms

To translate a formula to $\mathcal{B}(\mathcal{F})$ in our proof system, we use the following definition of the *elimination* of connectives.

Definition 6.34. Let $\mathcal{L}$ be a logic and Ω a proof system. Let $\mathfrak{f}$ be an n-ary connective. $\mathcal{L}$ has $\mathfrak{f}$-*elimination in* Ω if for all formulas $\xi_1, \ldots, \xi_n \in \mathcal{L}$ there exists some $\varphi \in \mathcal{L}$ such that $\mathfrak{f}(\xi_1, \ldots, \xi_n) \dashv\vdash_\Omega \varphi$.

In other words, if $\xi_1, \ldots, \xi_n$ are $\mathcal{L}$-formulas, then $\mathfrak{f}(\xi_1, \ldots, \xi_n)$ is provably equivalent to an $\mathcal{L}$-formula as well. As we let the elimination start at the innermost subformulas, we additionally require the next definition, which is a syntactic counterpart to the full abstraction principle (Proposition 2.8):

Definition 6.35. Let $\mathfrak{g}$ be an n-ary connective. A proof system Ω has *substitution in* $\mathfrak{g}$ if for all $\varphi_1, \psi_1, \ldots, \varphi_n, \psi_n$ it holds that $\varphi_1 \dashv\vdash \psi_1, \ldots, \varphi_n \dashv\vdash \psi_n$ implies $\mathfrak{g}(\varphi_1, \ldots, \varphi_n) \dashv\vdash \mathfrak{g}(\psi_1, \ldots, \psi_n)$.

6.3.1 Splitting elimination

The splitting disjunction is axiomatized by the rules listed in Table 6.5. We also include a universal version of the splitting operator, which is denoted by $\multimap$. It has the semantics

$$T \vDash \varphi \multimap \psi \;\Leftrightarrow\; \forall S, U : \text{if } T = S \cup U \text{ and } S \vDash \varphi, \text{then } U \vDash \psi.$$

In this thesis, we consider $\varphi \multimap \psi$ as an abbreviation for $\sim(\varphi \vee \sim\psi)$. The connective $\multimap$ is useful in describing properties such as downward closure, which are formulated by universal quantification over all subteams.

Let us briefly explain the role of each axiom and rule of S in the next lemma, and prove that they are sound. We do not give a proof that this system is minimal, but the informal explanation below hopefully convinces the reader that it cannot be condensed much further.

Lemma 6.36. *The proof system* $\mathsf{H}^{\mathsf{PL}}\mathsf{LS}$ *is sound for* $\mathsf{PL}(\sim)$.

Proof. The soundness follows by induction on the length of proofs, for which we show that all axioms are valid and rules preserve truth. For H^0 and L, this is by Proposition 6.5 and Lemma 6.9, respectively. Finally, for S, we consider each axiom and rule separately.

- (Dual$\multimap$) states that $\varovee$ can be defined in terms of $\multimap$. It is clearly sound, since $\sim(\varphi \multimap \sim\psi)$ is $\sim\sim(\varphi \varovee \sim\sim\psi) \equiv \varphi \varovee \psi$ by definition.
- (Sub$\varovee$) states that $\varovee$ always produces *subteams* of T (and not, say, successor teams like $\Diamond$). This is expressed in terms of downward closure of classical formulas α; if α holds, then α holds in every split (hence we use $\multimap$ instead of $\varovee$). Formally, let $T \vDash \alpha$, and let $S \cup U$ be an arbitrary split of T such that $S \vDash \varphi$. Then $U \vDash \alpha$ by downward closure, as $U \subseteq T$.
- (Lax$\varovee$) describes that $\varovee$ is *lax*. If a team T satisfies φ, and there is an arbitrary split into $S \cup U$ such that $S \vDash \theta$ and $U \vDash \psi$, then $T \cup U$ is also a split, witnessing $\varphi \varovee \psi$. In strict semantics, this law is not true, hence it is not provable from the other axioms (which are sound for strict semantics).
- (Ass$\varovee$) and (Com$\varovee$) are the associative and commutative law of $\varovee$, and clearly are true since $\cup$ is associative and commutative.
- (Dis$\multimap$) and (Nec$\multimap$) are used to introduce $\varovee$ in the spirit of the *necessitation* (*Gödel rule*) and the *axiom* K (*distribution law*) of modal logic. In analogy to ML, where necessitation introduces the universal $\Box$ and not the existential $\Diamond$, here we introduce the universal $\multimap$. We prove that they are sound. For (Dis$\multimap$), suppose that for every split $S \cup U$ of T such that $S \vDash \varphi$ it holds that $U \vDash \psi \rightarrow \theta$. If then additionally for every split $S \cup U$ of T it holds that $U \vDash \psi$, then also $U \vDash \theta$ for all such U. For (Nec$\multimap$), let φ be a theorem. Then by induction hypothesis, it is valid, i.e., true in all teams. But then $\psi \multimap \varphi$ is valid, since $U \vDash \varphi$ for every split $S \cup U$ of T. □

Example 6.37. The dependency atom $\mathsf{dep}(\alpha; \beta)$ can be defined as $\top \multimap (\mathsf{dep}(\alpha) \rightarrow \mathsf{dep}(\beta))$, where $\mathsf{dep}(\gamma) := \gamma \owedge \neg\gamma$. Figure 6.6 depicts a proof of one of Armstrong's axioms of dependence [6] in our system, namely the axiom of transitivity. It states that from $\mathsf{dep}(\alpha; \beta)$ and $\mathsf{dep}(\beta; \gamma)$ we can infer $\mathsf{dep}(\alpha; \gamma)$.

We proceed with showing that $\varovee$ can be eliminated by means of the system S. For that matter, the following lemma considerably simplifies the required proof.

Lemma 6.38. *Let $\Omega \succeq$ LS. Then Ω has substitution in $\sim$, $\wedge$ and $\varovee$. Furthermore, Ω admits the following meta-rules:*

- *Reductio ad absurdum* (RAA)*: If $\Phi \cup \{\varphi\} \vdash \{\psi, \sim\psi\}$, then $\Phi \vdash \sim\varphi$. If $\Phi \cup \{\sim\varphi\} \vdash \{\psi, \sim\psi\}$, then $\Phi \vdash \varphi$.*
- *Modus ponens in $\multimap$* (MP$\multimap$)*: If $\vdash \varphi \rightarrow \psi$ and $\Phi \vdash \theta \multimap \varphi$, then $\Phi \vdash \theta \multimap \psi$.*
- *Modus ponens in $\varovee$* (MP$\varovee$)*: If $\vdash \varphi \rightarrow \psi$ and $\Phi \vdash \theta \varovee \varphi$, then $\Phi \vdash \theta \varovee \psi$.*

Proof. First, we derive the meta-rules in Ω. For (RAA), the standard proof is as follows. Suppose $\Phi \cup \{\varphi\} \vdash \{\psi, \sim\psi\}$. By the deduction theorem (Theorem 6.13), $\Phi \vdash \{\varphi \rightarrow \psi, \varphi \rightarrow$

A	$\mathsf{dep}(\alpha;\beta)$	
B	$\mathsf{dep}(\beta;\gamma)$	
1	$\top \multimap (\mathsf{dep}(\alpha) \rightarrowtriangle \mathsf{dep}(\beta))$	def., A
2	$\top \multimap (\mathsf{dep}(\beta) \rightarrowtriangle \mathsf{dep}(\gamma))$	def., B
3	$(\mathsf{dep}(\alpha) \rightarrowtriangle \mathsf{dep}(\beta)) \rightarrowtriangle ((\mathsf{dep}(\beta) \rightarrowtriangle \mathsf{dep}(\gamma)) \rightarrowtriangle (\mathsf{dep}(\alpha) \rightarrowtriangle \mathsf{dep}(\gamma)))$	L
4	$\top \multimap \big(((\mathsf{dep}(\alpha) \rightarrowtriangle \mathsf{dep}(\beta)) \rightarrowtriangle ((\mathsf{dep}(\beta) \rightarrowtriangle \mathsf{dep}(\gamma)) \rightarrowtriangle (\mathsf{dep}(\alpha) \rightarrowtriangle \mathsf{dep}(\gamma)))\big)$	(Nec$\multimap$)
5	$\big(\top \multimap (\mathsf{dep}(\alpha) \rightarrowtriangle \mathsf{dep}(\beta))\big) \rightarrowtriangle \big(\top \multimap ((\mathsf{dep}(\beta) \rightarrowtriangle \mathsf{dep}(\gamma)) \rightarrowtriangle (\mathsf{dep}(\alpha) \rightarrowtriangle \mathsf{dep}(\gamma)))\big)$	(Dis$\multimap$)
6	$\top \multimap \big((\mathsf{dep}(\beta) \rightarrowtriangle \mathsf{dep}(\gamma)) \rightarrowtriangle (\mathsf{dep}(\alpha) \rightarrowtriangle \mathsf{dep}(\gamma))\big)$	(E$\rightarrowtriangle$), 1, 5
7	$\big(\top \multimap (\mathsf{dep}(\beta) \rightarrowtriangle \mathsf{dep}(\gamma))\big) \rightarrowtriangle \big(\top \multimap (\mathsf{dep}(\alpha) \rightarrowtriangle \mathsf{dep}(\gamma))\big)$	(Dis$\multimap$)
8	$\top \multimap (\mathsf{dep}(\alpha) \rightarrowtriangle \mathsf{dep}(\gamma))$	(E$\rightarrowtriangle$), 2, 7
$\triangleright$	$\mathsf{dep}(\alpha;\gamma)$	def.

Figure 6.6: Example derivation: Transitivity of dependence

$\sim\psi\}$. Moreover, the propositional law $(\varphi \rightarrowtriangle \sim\psi) \rightarrowtriangle (\psi \rightarrowtriangle \sim\varphi)$ is derivable in L due to Theorem 6.20. Consequently, $\Phi \vdash \{\varphi \rightarrowtriangle \psi, \psi \rightarrowtriangle \sim\varphi\}$, and together $\Phi \vdash \{\varphi \rightarrowtriangle \sim\varphi\}$. But $(\varphi \rightarrowtriangle \sim\varphi) \rightarrowtriangle \sim\varphi$ is again a theorem, so $\Phi \vdash \sim\varphi$ as required. The other case is proved analogously.

The rule (MP$\multimap$) is straightforward by application of (Nec$\multimap$), (Dis$\multimap$) and (E$\rightarrowtriangle$): Given $\vdash \varphi \rightarrowtriangle \psi$, by (Nec$\multimap$) we have $\vdash \theta \multimap (\varphi \rightarrowtriangle \psi)$, so by (Dis$\multimap$) then $\vdash (\theta \multimap \varphi) \rightarrowtriangle (\theta \multimap \psi)$, and finally with (E$\rightarrowtriangle$) we obtain $\Phi \vdash \theta \multimap \psi$ from $\Phi \vdash \theta \multimap \varphi$.

Finally, (MP$\vee$) is derived as follows, where "(thm)" marks a formula as a theorem:

A	$\varphi \rightarrowtriangle \psi$	(thm)
B	$\theta \vee \varphi$	
1	$\sim\psi \rightarrowtriangle \sim\varphi$	(thm), L, A
2	$\theta \multimap (\sim\psi \rightarrowtriangle \sim\varphi)$	(thm), (Nec$\multimap$)
3	$(\theta \multimap \sim\psi) \rightarrowtriangle (\theta \multimap \sim\varphi)$	(thm), (Dis$\multimap$)
4	$\sim(\theta \multimap \sim\varphi) \rightarrowtriangle \sim(\theta \multimap \sim\psi)$	(thm), L
5	$\sim(\theta \multimap \sim\varphi)$	(Dual$\multimap$), B
6	$\sim(\theta \multimap \sim\psi)$	(E$\rightarrowtriangle$)
$\triangleright$	$\theta \vee \psi$	(Dual$\multimap$)

Next, we prove substitution in $\sim$, $\wedge$ and $\vee$. For $\sim$, suppose $\varphi = \sim\xi$ and $\xi \dashv\vdash \psi$. Obviously, $\{\varphi, \psi\} \vdash \xi, \sim\xi$. By (RAA), $\varphi \vdash \sim\psi$. For $\wedge$, suppose $\varphi = \xi_1 \wedge \xi_2$, $\xi_1 \dashv\vdash \psi_1$ and $\xi_2 \dashv\vdash \psi_2$. Then in L, immediately $\varphi = \xi_1 \wedge \xi_2 \vdash \{\xi_1, \xi_2\} \vdash \{\psi_1, \psi_2\} \vdash \psi_1 \wedge \psi_2$. Finally, substitution is $\vee$ is obtained by two applications of (Com$\vee$) and (MP$\vee$) and the deduction theorem. $\square$

Example 6.39. For $\alpha, \beta \in \mathrm{PL}$, the formula $(\alpha \multimap \beta) \rightarrowtriangle \beta$ is valid: α is satisfied by the

(D∧∨)	$(\alpha \wedge (\varphi \vee \psi)) \leftrightarrow ((\alpha \wedge \varphi) \vee (\alpha \wedge \psi))$	Distr. ∧ over ∨
(D∨⊘)	$(\varphi \vee (\psi \oslash \theta)) \leftrightarrow ((\varphi \vee \psi) \oslash (\varphi \vee \theta))$	Distr. ∨ over ⊘
(D∨∧)	$(\varphi \vee (\alpha \wedge \mathsf{E}\beta)) \leftrightarrow ((\varphi \vee \alpha) \wedge \mathsf{E}(\alpha \wedge \beta))$	Distr. ∨ over ∧
(AbsE∨)	$(\mathsf{E}\alpha \vee \varphi) \rightarrow \mathsf{E}\alpha$	Absorption of ∨ in E
(AbsE∧)	$(\alpha \wedge \mathsf{E}\beta) \rightarrow \mathsf{E}(\alpha \wedge \beta)$	Absorption of ∧ in E

Table 6.7: Useful theorems of $\mathsf{H}^{\mathsf{PL}}\mathsf{LS}$

empty team, and as every team T has the trivial division into $\emptyset$ and T, having $T \vDash \alpha \multimap \beta$ implies $T \vDash \beta$. We sketch a proof of it in the system $\mathsf{H}^{\mathsf{PL}}\mathsf{LS}$. For a proof by (RAA), we start with the assumptions $(\alpha \multimap \beta)$ and $\sim\beta$ and derive a contradiction. First, observe that $\top \vee \bot$ is a classical tautology and hence provable in H^{PL}. By (Lax∨), we obtain $\sim\beta \vee \bot$, and since classically $\bot \vdash \alpha$, we derive $\sim\beta \vee \alpha$ by (MP∨) and $\alpha \vee \sim\beta$ by (Com∨). This yields the desired contradiction, as $\alpha \multimap \beta$ is short for $\sim(\alpha \vee \sim\beta)$.

Moreover, the axioms S allow to derive auxiliary laws regarding ∨ and ⊸:

Lemma 6.40. *Let $\Omega \succeq \mathsf{H}^{\mathsf{PL}}\mathsf{LS}$. Then all instances of the laws in Table 6.7 are provable in Ω.*

Proof. Proven in the appendix. □

The actual proof that $\mathsf{H}^{\mathsf{PL}}\mathsf{LS}$ has ∨-elimination spans several further lemmas. We implicitly apply Lemma 6.38 when using substitution in $\wedge, \sim$ and $\vee$ and make use of the laws in Lemmas 6.38 and 6.40. The first step is the *and/or lemma*.

Lemma 6.41 (And/Or lemma). *If $\Omega \succeq \mathsf{H}^{\mathsf{PL}}\mathsf{LS}$, then*

$$\bigwedge_{i=1}^{n} \mathsf{E}\beta_i \dashv\vdash \bigvee_{i=1}^{n} \mathsf{E}\beta_i$$

in Ω for all $\beta_1, \dots, \beta_n \in \mathcal{F}$.

Proof. We begin with the direction "⊢", and proceed by induction on n. The case $n = 1$ is trivial. For $n > 1$, due to the induction hypothesis and by substitution in $\wedge$ it suffices to prove $(\bigvee_{i=1}^{n-1} \mathsf{E}\beta_i) \wedge \mathsf{E}\beta_n \vdash \bigvee_{i=1}^{n} \mathsf{E}\beta_i$.

In L, we can separate the two conjuncts and obtain $\mathsf{E}\beta_n$ and $\bigwedge_{i=1}^{n-1} \mathsf{E}\beta_i$. Now, $\top \vee \top$ is classically provable in H^{PL}, and yields $\mathsf{E}\beta_n \vee \top$ by one application of (Lax∨), $\top \vee \mathsf{E}\beta_n$ by (Com∨), and $\bigvee_{i=1}^{n-1} \mathsf{E}\beta_i \vee \mathsf{E}\beta_n$ by another (Lax∨), hence $\bigvee_{i=1}^{n} \mathsf{E}\beta_i$.

The other direction "⊣" is shown by a separate derivation of each conjunct with (AbsE∨), (Ass∨) and (Com∨), which in L then yields the whole conjunction. □

Lemma 6.42 (Change of normal form). *If $\Omega \succeq \mathsf{H}^{\mathsf{PL}}\mathsf{LS}$, then*

$$\alpha \wedge \bigwedge_{i=1}^{n} \mathsf{E}\beta_i \dashv\vdash \bigvee_{i=1}^{n} (\alpha \wedge \mathsf{E}\beta_i) \tag{1}$$

for all $n \in \mathbb{N}$ *and* $\alpha, \beta_1, \ldots, \beta_n \in \mathcal{F}$, *and*

$$\bigvee_{i=1}^{n} (\alpha_i \wedge \mathsf{E}\beta_i) \dashv\vdash \left(\bigvee_{i=1}^{n} \alpha_i\right) \wedge \bigwedge_{i=1}^{n} \mathsf{E}(\alpha_i \wedge \beta_i) \qquad (2)$$

for all $n \in \mathbb{N}$ *and* $\alpha_1, \ldots, \beta_n, \beta_1, \ldots, \beta_n \in \mathcal{F}$.

Proof. Let us start with (1). Here, we first apply the and/or lemma to replace the large conjunction by $\bigvee_{i=1}^{n} \mathsf{E}\beta_i$. Then we simply distribute α with repeated application of (D$\wedge\vee$), (Ass$\vee$) and (Com$\vee$). Both steps are provable equivalences.

For (2), we consider both directions separately. For "$\vdash$", we obtain $\bigvee_{i=1}^{n} \alpha_i$ from $\bigvee_{i=1}^{n} (\alpha_i \wedge \mathsf{E}\beta_i)$ by the application of (Ass$\vee$), (Com$\vee$) and (MP$\vee$), as $(\alpha_i \wedge \mathsf{E}\beta_i) \vdash_{\mathsf{L}} \alpha_i$ for all i. Next, we apply (AbsE$\wedge$) to similarly derive $\bigvee_{i=1}^{n} \mathsf{E}(\alpha_i \wedge \beta_i)$, which by Lemma 6.41 yields $\bigwedge_{i=1}^{n} \mathsf{E}(\alpha_i \wedge \beta_i)$. In L, we form the conjunction of both.

For "$\dashv$", we repeatedly apply the theorem (D$\vee\wedge$) of Lemma 6.40, that is, $(\varphi \vee \alpha) \wedge \mathsf{E}(\alpha \wedge \beta) \vdash \varphi \vee (\alpha \wedge \mathsf{E}\beta)$. We proceed as follows. Assume that the formula has the following form after k applications:

$$\left(\bigvee_{i=1}^{k} (\alpha_i \wedge \mathsf{E}\beta_i) \vee \bigvee_{i=k+1}^{n} \alpha_i\right) \wedge \bigwedge_{i=k+1}^{n} \mathsf{E}(\alpha_i \wedge \beta_i).$$

For $k = 0$, this is just the right hand side of (2). We isolate a single subformula on each side with the commutative and associative laws:

$$\left[\left(\bigvee_{i=1}^{k} (\alpha_i \wedge \mathsf{E}\beta_i) \vee \bigvee_{i=k+2}^{n} \alpha_i\right) \vee \alpha_{k+1}\right] \wedge \mathsf{E}(\alpha_{k+1} \wedge \beta_{k+1}) \wedge \bigwedge_{i=k+2}^{n} \mathsf{E}(\alpha_i \wedge \beta_i)$$

Then we apply (D$\vee\wedge$) (from right to left), resulting in

$$\left[\left(\bigvee_{i=1}^{k} (\alpha_i \wedge \mathsf{E}\beta_i) \vee \bigvee_{i=k+2}^{n} \alpha_i\right) \vee (\alpha_{k+1} \wedge \mathsf{E}\beta_{k+1})\right] \wedge \bigwedge_{i=k+2}^{n} \mathsf{E}(\alpha_i \wedge \beta_i),$$

and again with commutative and associative laws in

$$\left(\bigvee_{i=1}^{k+1} (\alpha_i \wedge \mathsf{E}\beta_i) \vee \bigvee_{i=k+2}^{n} \alpha_i\right) \wedge \bigwedge_{i=k+2}^{n} \mathsf{E}(\alpha_i \wedge \beta_i),$$

where we can repeat the above steps until $k = n$. □

With the above lemma, we are ready to prove $\vee$-elimination.

Lemma 6.43 ($\vee$-elimination). *Let* $\mathcal{F}$ *be a logic closed under* $\neg, \vee, \wedge, \top, \bot$. *Let* $\Omega \succeq \mathsf{H}^{\mathsf{PL}}\mathsf{LS}$. *Then* $\mathcal{B}(\mathcal{F})$ *has* $\vee$*-elimination in* Ω.

Proof. Suppose that $\varphi = \psi \vee \theta$ where $\psi, \theta \in \mathcal{B}(\mathcal{F})$. For $\vee$-elimination, we have to show that φ is provably equivalent to a $\mathcal{B}(\mathcal{F})$-formula. By Theorem 6.20, all propositional laws are available, and we have substitution in $\vee$ (Lemma 6.38), so w.l.o.g. ψ, θ are in $(\otimes\wedge)$-normal form (cf. Section 3.7.1), that is, in the form

$$\bigotimes_{i=1}^{n} \left(\alpha_i \wedge \bigwedge_{j=1}^{m_i} \mathsf{E}\beta_{i,j} \right)$$

for $n, m_i \in \mathbb{N}$ and $\alpha_i, \beta_{i,j} \in \mathcal{F}$. Then, we have the following provable equivalences in Ω:

$$\begin{aligned}
\varphi &\dashv\vdash \left[\bigotimes_{i=1}^{n} \left(\alpha_i \wedge \bigwedge_{j=1}^{m_i} \mathsf{E}\beta_{i,j} \right) \vee \bigotimes_{i=1}^{n'} \left(\alpha'_i \wedge \bigwedge_{j=1}^{m'_i} \mathsf{E}\beta'_{i,j} \right) \right] \\
&\dashv\vdash \bigotimes_{\substack{1 \leqslant i \leqslant n \\ 1 \leqslant i' \leqslant n'}} \left[\left(\alpha_i \wedge \bigwedge_{j=1}^{m_i} \mathsf{E}\beta_{i,j} \right) \vee \left(\alpha'_{i'} \wedge \bigwedge_{j=1}^{m'_{i'}} \mathsf{E}\beta'_{i',j} \right) \right] && (\mathrm{D}\vee\otimes) \\
&\dashv\vdash \bigotimes_{\substack{1 \leqslant i \leqslant n \\ 1 \leqslant i' \leqslant n'}} \left(\bigvee_{j=1}^{m_i} \left(\alpha_i \wedge \mathsf{E}\beta_{i,j} \right) \vee \bigvee_{j=1}^{m'_{i'}} \left(\alpha'_{i'} \wedge \mathsf{E}\beta'_{i',j} \right) \right) && (\text{Lemma 6.42, (1)}) \\
&\dashv\vdash \bigotimes_{i=1}^{\ell} \bigvee_{j=1}^{k_i} \left(\gamma_{i,j} \wedge \mathsf{E}\delta_{i,j} \right) && (\text{renaming, } \gamma_{i,j}, \delta_{i,j} \in \mathcal{F}) \\
&\dashv\vdash \bigotimes_{i=1}^{\ell} \left(\bigvee_{j=1}^{k_i} \gamma_{i,j} \wedge \bigwedge_{j=1}^{k_i} \mathsf{E} \left(\gamma_{i,j} \wedge \delta_{i,j} \right) \right) \in \mathcal{B}(\mathcal{F}). && (\text{Lemma 6.42, (2)})
\end{aligned}$$

□

Theorem 6.44. *The system* $\mathsf{H}^{\mathsf{PL}}\mathsf{LS}$ *axiomatizes* $\mathsf{PL}(\sim)$.

Proof. Using substitution in $\wedge, \sim$ and $\vee$ (Lemma 6.38), any $\mathsf{PL}(\sim)$-formula can be transformed into a $\mathcal{B}(\mathsf{PL})$-formula by means of $\vee$-elimination (Lemma 6.43). Hence every $\mathsf{PL}(\sim)$ is provably equivalent to a $\mathcal{B}(\mathsf{PL})$-formula in the system $\mathsf{H}^{\mathsf{PL}}\mathsf{LS}$. The completeness consequently follows from the combination of Lemma 6.33 and Theorem 6.26. The soundness was shown in Lemma 6.36. □

6.3.2 Modality elimination

Next, we extend the proof system to cover the modal operators. This is achieved with the proof system M depicted in Table 6.8. As for $\mathsf{PL}(\sim)$, we introduce the universal dual of $\Diamond$, which we write $\triangle$. It has the semantics

$$T \vDash \triangle\varphi \;\Leftrightarrow\; \forall S : \text{if } S \text{ is a successor team of } T, \text{then } S \vDash \varphi$$

and is syntactically defined as $\triangle\varphi := \sim\Diamond\sim\varphi$. Recall that the operator $\Box$ is self-dual, so we do not require an additional connective here.

Lemma 6.45. *The proof system* $\mathsf{H}^{\mathsf{ML}}\mathsf{LSM}$ *is sound for* $\mathsf{ML}(\sim)$.

(Dual$\Diamond$)	$\Diamond\varphi \leftrightarrow \sim\triangle\sim\varphi$	Duality of $\triangle$ and $\Diamond$
(Lin$\Box$)	$\Box\sim\varphi \leftrightarrow \sim\Box\varphi$	Self-duality of $\Box$
(Dis$\Diamond\vee$)	$\Diamond(\varphi \vee \psi) \leftrightarrow (\Diamond\varphi \vee \Diamond\psi)$	Distr. $\Diamond$ over $\vee$
(E$\Box$)	$\Box\alpha \rightarrow \triangle\alpha$	Successors are subteams of image
(I$\Box$)	$\Diamond\varphi \rightarrow (\triangle\psi \rightarrow \Box\psi)$	Image is a successor (if one exists)
(Dis$\Box$)	$\Box(\varphi \rightarrow \psi) \rightarrow (\Box\varphi \rightarrow \Box\psi)$	Distribution axiom of $\Box$
(Dis$\triangle$)	$\triangle(\varphi \rightarrow \psi) \rightarrow (\triangle\varphi \rightarrow \triangle\psi)$	Distribution axiom of $\triangle$
(Nec$\Box$)	$\dfrac{\varphi}{\Box\varphi}$ (φ theorem)	Necessitation of $\Box$
(Nec$\triangle$)	$\dfrac{\varphi}{\triangle\varphi}$ (φ theorem)	Necessitation of $\triangle$

Table 6.8: The system M, modal axioms

Proof. The soundness of H^{ML}, L and S is shown analogously to Lemma 6.36. Below, we consider every rule and axiom of M.

- (Dual$\Diamond$) states that $\Diamond$ can be defined in terms of $\triangle$, and is clearly sound, as $\Diamond\varphi \equiv \sim\sim\Diamond\sim\sim\varphi$, which is $\sim\triangle\sim\varphi$ by definition.
- (Lin$\Box$) states that $\Box$ is self-dual, or equivalently, that the image team RT of a team T is unique. Hence $T \vDash \Box\sim\varphi \Leftrightarrow RT \nvDash \varphi \Leftrightarrow T \nvDash \Box\varphi \Leftrightarrow T \vDash \sim\Box\varphi$.
- (Dis$\Diamond\vee$) states that $\Diamond$ is lax. This is formalized by saying that it distributes over the lax disjunction (whereas its strict counterpart $\dot\Diamond$ distributes only over strict disjunction $\dot\vee$). A formal proof follows. Let $\mathcal{K} = (W, R, V)$ be a Kripke structure and $T \subseteq W$ a team.

"$\rightarrow$": Suppose $(\mathcal{K}, T) \vDash \Diamond(\varphi \vee \psi)$. Then T has a successor team T' such that there are S' and U' with $T' = S' \cup U'$, $(\mathcal{K}, S') \vDash \varphi$ and $(\mathcal{K}, U') \vDash \psi$. We define subteams S and U of T such that $T = S \cup U$, $(\mathcal{K}, S) \vDash \Diamond\varphi$ and $(\mathcal{K}, U) \vDash \Diamond\psi$:

$$S := \{\, v \in T \mid \exists v' \in S' : (v, v') \in R \,\},$$
$$U := \{\, v \in T \mid \exists v' \in U' : (v, v') \in R \,\}.$$

Every world $v \in T$ has at least one successor $v' \in T'$. Since $S' \cup U' = T'$, either $v' \in S'$, or $v' \in U'$, or both. By definition, v is in then in S or U. Consequently, $T = S \cup U$.

To prove $(\mathcal{K}, S) \vDash \Diamond\varphi$, we demonstrate that S' is a successor team of S. $(\mathcal{K}, U) \vDash \Diamond\psi$ is then shown analogously. First, by definition of S, every $v \in S$ has at least one successor in S'. Likewise, every $v' \in S'$ has at least one predecessor in S: Since $S' \subseteq T'$ and T' is a successor team of T, v' has some predecessor v in T. By definition of S, $v \in S$. It follows that S' is a successor team of S.

"$\leftarrow$": Suppose $(\mathcal{K}, T) \vDash \Diamond\varphi \vee \Diamond\psi$ due to subteams S and U of T such that $T = S \cup U$, $(\mathcal{K}, S) \vDash \Diamond\varphi$ and $(\mathcal{K}, U) \vDash \Diamond\psi$. Then there is a successor team S' of S satisfying

φ, and a successor team U' of U satisfying ψ.

We show that $T' := S' \cup U'$, which satisfies $\varphi \vee \psi$, itself is a successor team of T. If $v \in T$, then $v \in S$ or $v \in U$, and v has a successor in S' or U', and consequently in T'. On the other hand, $v' \in T'$ implies $v' \in S'$ or $v' \in U'$. But then v' has a predecessor in S or U, and hence in T.

- (E$\Box$) states that the image team RT of T contains all successor teams of T as a subteam. Like (Sub$\vee$), it is formalized in terms of the downward closure of classical formulas.
- (I$\Box$) states that the image team RT of T is itself a successor team of T, provided that there is some successor team at all. If some $w \in T$ has no successor, then $\triangle\psi$ is trivially true as no successor team exists, but $\Box\psi$ is not necessarily true. If the premise $\Diamond\varphi$ holds for some φ, then the implication is true.
- (Dis$\Box$), (Nec$\Box$), (Dis$\triangle$) and (Nec$\triangle$) are the distribution and necessitation of $\Box$ and $\triangle$, respectively, and are sound by the same argument as (Dis$\multimap$) and (Nec$\multimap$). ☐

Lemma 6.46. *Let $\Omega \succeq \mathsf{LSM}$ be a proof system. Then Ω has substitution in $\sim, \wedge, \vee, \Box$ and $\Diamond$. Furthermore, Ω admits the following meta-rules:*

- *Modus ponens in $\Box$ (MP$\Box$): If $\vdash \varphi \rightarrowtriangle \psi$ and $\Phi \vdash \Box\varphi$, then $\Phi \vdash \Box\psi$.*
- *Modus ponens in $\triangle$ (MP$\triangle$): If $\vdash \varphi \rightarrowtriangle \psi$ and $\Phi \vdash \triangle\varphi$, then $\Phi \vdash \triangle\psi$.*
- *Modus ponens in $\Diamond$ (MP$\Diamond$): If $\vdash \varphi \rightarrowtriangle \psi$ and $\Phi \vdash \Diamond\varphi$, then $\Phi \vdash \Diamond\psi$.*

Proof. It is straightforward to prove (MP$\Box$) and (MP$\triangle$) from (Nec$\Box$) and (Dis$\Box$) resp. (Nec$\triangle$) and (Dis$\triangle$) (see also the proof for (MP$\multimap$) in Lemma 6.38). As $\Omega \succeq \mathsf{LS}$, (RAA) is available by Lemma 6.38, so (MP$\Diamond$) can be derived as follows.

A $\varphi \rightarrowtriangle \psi$	(thm)
B $\Diamond\varphi$	
1 $\sim\psi \rightarrowtriangle \sim\varphi$	(thm), L, A
2 $\triangle\sim\psi$	
3 $\triangle\sim\varphi$	(MP$\triangle$), 1, 2
4 $\sim\triangle\sim\varphi$	(Dual$\Diamond$), A
5 $\sim\triangle\sim\psi$	(RAA), 3, 4
▷ $\Diamond\psi$	(Dual$\Diamond$)

It remains to prove that Ω admits substitution. The cases $\sim$, $\wedge$ and $\vee$ follow from Lemma 6.38, as $\Omega \succeq \mathsf{LS}$. Finally, the cases $\Box$ and $\Diamond$ immediately follow from (MP$\Box$) and (MP$\Diamond$). ☐

Lemma 6.47. *Let $\Omega \succeq \mathsf{H}^{\mathsf{ML}}\mathsf{LSM}$. Then all instances of the laws in Table 6.9 are provable in Ω.*

Proof. Proven in the appendix. ☐

(Dis$\Box\wedge$)	$\Box(\varphi \wedge \psi) \leftrightarrow (\Box\varphi \wedge \Box\psi)$	Distr. $\Box$ over $\wedge$
(Dis$\Diamond\varovee$)	$\Diamond(\varphi \varovee \psi) \leftrightarrow (\Diamond\varphi \varovee \Diamond\psi)$	Distr. $\Diamond$ over $\varovee$
(Dis$\Diamond\wedge$)	$\Diamond(\alpha \wedge \mathsf{E}\beta) \leftrightarrow \Diamond\alpha \wedge \mathsf{E}\Diamond(\alpha \wedge \beta)$	Distr. $\Diamond$ over $\wedge$

Table 6.9: Useful theorems of $\mathsf{H}^{\mathsf{ML}}\mathsf{LSM}$

Lemma 6.48. *Let $\Omega \succeq \mathsf{LSM}$. Then $\mathcal{B}(\mathsf{ML})$ has $\Box$-elimination in Ω.*

Proof. Suppose $\varphi \in \mathcal{B}(\mathsf{ML})$. To prove the lemma, we have to show that $\Box\varphi \dashv\vdash \psi$ for some $\psi \in \mathcal{B}(\mathsf{ML})$. We repeatedly apply (Dis$\Box\wedge$) and (Lin$\Box$) in order to push $\Box$ inside any $\wedge$ and $\sim$. Since afterwards $\Box$ only occurs in classical subformulas, and since the used laws are symmetric, we conclude that $\Box\varphi$ is provably equivalent to a $\mathcal{B}(\mathsf{ML})$-formula. $\Box$

Lemma 6.49. *Let $\Omega \succeq \mathsf{H}^{\mathsf{ML}}\mathsf{LSM}$. Then $\mathcal{B}(\mathsf{ML})$ has $\Diamond$-elimination in Ω.*

Proof. Suppose $\varphi \in \mathcal{B}(\mathsf{ML})$. We prove that $\Diamond\varphi \dashv\vdash \psi$ for some $\psi \in \mathcal{B}(\mathsf{ML})$. Analogously to the proof of Lemma 6.43, we can assume that φ is in normal form.

$$\begin{aligned}
\Diamond\varphi &\dashv\vdash \Diamond \bigovee_{i=1}^{n} \left(\alpha_i \wedge \bigwedge_{j=1}^{k_i} \mathsf{E}\beta_{i,j}\right) && \\
&\dashv\vdash \Diamond \bigovee_{i=1}^{n} \bigvee_{j=1}^{k_i} \left(\alpha_i \wedge \mathsf{E}\beta_{i,j}\right) && \text{(Lemma 6.42 (1))} \\
&\dashv\vdash \bigovee_{i=1}^{n} \Diamond \bigvee_{j=1}^{k_i} (\alpha_i \wedge \mathsf{E}\beta_{i,j}) && (\mathsf{Dis}\Diamond\varovee) \\
&\dashv\vdash \bigovee_{i=1}^{n} \bigvee_{j=1}^{k_i} \Diamond (\alpha_i \wedge \mathsf{E}\beta_{i,j}) && (\mathsf{Dis}\Diamond\vee)) \\
&\dashv\vdash \bigovee_{i=1}^{n} \bigvee_{j=1}^{k_i} (\Diamond\alpha_i \wedge \mathsf{E}\Diamond(\alpha_i \wedge \beta_{i,j})) && (\mathsf{Dis}\Diamond\wedge)) \\
&\dashv\vdash \bigovee_{i=1}^{n} \bigvee_{j=1}^{k_i} (\mu_i \wedge \mathsf{E}\nu_{i,j}) && \text{(renaming, } \mu_i, \nu_{i,j} \in \mathsf{ML}) \\
&\dashv\vdash \bigovee_{i=1}^{n} \left(\bigvee_{j=1}^{k_i} \mu_i \wedge \bigwedge_{j=1}^{k_i} \mathsf{E}\left(\mu_i \wedge \nu_{i,j}\right)\right) \in \mathcal{B}(\mathsf{ML}). && \text{(Lemma 6.42 (2))} \quad \Box
\end{aligned}$$

Theorem 6.50. *The system $\mathsf{H}^{\mathsf{ML}}\mathsf{LSM}$ axiomatizes $\mathsf{ML}(\sim)$.*

Proof. Similar to Theorem 6.44. With $\Box$-elimination (Lemma 6.48) and $\Diamond$-elimination (Lemma 6.49), every $\mathsf{ML}(\sim)$-formula is provably equivalent to a $\mathcal{B}(\mathsf{ML})$-formula, so we can apply Lemma 6.33 and Theorem 6.26. Soundness was proved in Lemma 6.45. $\Box$

6.3.3 Quantifier elimination

For first-order team logic, we use the system Q depicted in Table 6.10. The universal dual of $\exists x$ is denoted by $!x$ and is defined as $!x\,\varphi := {\sim}\exists x{\sim}\varphi$. The axioms almost resemble those of the system M. There are two differences: First, there is no necessitation rule for $\Box$, as it is derivable from (Nec!) and (I$\forall$). Second, (I$\forall$) lacks the additional premise compared to (I$\Box$), because there always is a supplementing team.

Lemma 6.51. *The proof system* $\mathsf{H}^{\mathsf{FO}}\mathsf{ULSQ}$ *is sound for* $\mathsf{FO}(\sim)$.

Proof. The soundness of H^{FO}, L and S is again shown analogously to Lemma 6.36. U was shown sound in Corollary 6.28. It remains to consider the rules and axioms of Q.

- (Dual$\exists$) defines ! in terms of $\exists$, analogously to $\triangle/\Diamond$ and $\multimap/\varovee$.
- (Lin$\forall$) states that $\forall x$ is self-dual, as the duplicating team $T^x_{\mathcal{A}}$ is unique.
- (Dis$\exists\varovee$) states that $\exists x$ is lax, i.e., distributes over $\varovee$. For a proof, let $\mathcal{A}$ be a first-order structure, $x \in \mathsf{Var}$ and T a team in $\mathcal{A}$.

"$\rightarrowtriangle$": Suppose $(\mathcal{A}, T) \vDash \exists x(\varphi \varovee \psi)$. There is a supplementing function $f \colon T \to \wp^+(\mathcal{A})$ such that T^x_f can be split into $T^x_f = S' \cup U'$ with $(\mathcal{A}, S') \vDash \varphi$ and $(\mathcal{A}, U') \vDash \psi$. We define subteams S and U of T such that $T = S \cup U$, $(\mathcal{A}, S) \vDash \exists x\varphi$ and $(\mathcal{A}, U) \vDash \exists x\psi$:

$$S := \left\{\, s \in T \;\middle|\; \exists s' \in S', a \in \mathcal{A} : s' = s^x_a \,\right\},$$
$$U := \left\{\, s \in T \;\middle|\; \exists s' \in U', a \in \mathcal{A} : s' = s^x_a \,\right\}.$$

Let $s \in T$. As $f(s) \neq \emptyset$, there is at least one $a \in \mathcal{A}$ such that $s^x_a \in T^x_f$, and hence $s^x_a \in S' \cup U'$. Consequently, $s \in S$ or $s \in U$. As s was arbitrary, $T = S \cup U$. Next, we will prove that S is a supplementing team of S' (the proof for U is analogous). As then $(\mathcal{A}, S') \vDash \exists x\,\varphi$ and $(\mathcal{A}, U') \vDash \exists x\,\psi$, $(\mathcal{A}, T) \vDash (\exists x\,\varphi) \varovee (\exists x\,\psi)$ follows.

We show that $S = (S')^x_g$ for $g(s) := \{\, a \in \mathcal{A} \mid s^x_a \in S \,\}$. $g(s)$ is always non-empty, since $s \in S'$ implies $s^x_a \in S$ for some a by definition of S'. So g is a

(Dual$\exists$)	$\exists x\,\varphi \leftrightarrowtriangle {\sim}\,!x\,{\sim}\varphi$	Duality of $\exists$ and !
(Lin$\forall$)	$\forall x\,{\sim}\varphi \leftrightarrowtriangle {\sim}\forall x\,\varphi$	Self-duality of $\forall$
(Dis$\exists\varovee$)	$\exists x(\varphi \varovee \psi) \leftrightarrowtriangle \exists x\,\varphi \varovee \exists x\,\psi$	Distr. $\exists$ over $\varovee$
(E$\forall$)	$\forall x\,\alpha \rightarrowtriangle\, !x\,\alpha$	Suppl. teams are subteams of dupl. team
(I$\forall$)	$!x\,\psi \rightarrowtriangle \forall x\,\psi$	Dupl. team is a suppl. team
(Dis$\Box$)	$\forall x(\varphi \rightarrowtriangle \psi) \rightarrowtriangle (\forall x\varphi \rightarrowtriangle \forall x\,\psi)$	Distribution axiom
(Dis!)	$!x(\varphi \rightarrowtriangle \psi) \rightarrowtriangle (!x\varphi \rightarrowtriangle\, !x\,\psi)$	Distribution axiom
(Nec!)	$\dfrac{\varphi}{!x\varphi}$ (φ theorem)	Necessitation

Table 6.10: The system Q, quantifier axioms

supplementing function. In order to prove $S \subseteq (S')^x_g$, suppose $s' \in S$. As $S \subseteq T^x_f$, then $s' = s^x_a$ for some $a \in f(s)$ and $s \in T$. By definition of S', then $s \in S'$, and since $a \in g(s)$, we have $s^x_a \in (S')^x_g$. For $(S')^x_g \subseteq S$, let $s' \in (S')^x_g$. Then $s' = s^x_a$ for some $s \in S'$ and $a \in g(s)$. By definition of g, then $s' = s^x_a \in S$.

"$\Leftarrow$": Suppose $(\mathcal{A}, T) \vDash (\exists x\, \varphi) \vee (\exists x\, \psi)$, i.e., that $(\mathcal{A}, S) \vDash \exists x\, \varphi$ and $(\mathcal{A}, U) \vDash \exists x\, \psi$ for $T = S \cup U$. Let S^x_f and U^x_g be supplementing teams of S and U such that $(\mathcal{A}, S^x_f) \vDash \varphi$ and $(\mathcal{A}, U^x_g) \vDash \psi$. We prove that $S^x_f \cup U^x_g$ is a supplementing team of T, which implies $(\mathcal{A}, T) \vDash \exists x\, (\varphi \vee \psi)$. Consider the function h on $T = S \cup U$ given by

$$h(s) := \begin{cases} f(s) & \text{if } s \in S \setminus U, \\ g(s) & \text{if } s \in U \setminus S, \\ f(s) \cup g(s) & \text{if } s \in S \cap U. \end{cases}$$

Clearly $h : T \to \wp^+(\mathcal{A})$. We demonstrate $S^x_f \cup U^x_g = T^x_h$. For $S^x_f \subseteq T^x_h$ (U^x_g is analogous), suppose $s' \in S^x_f$. Then $s' = s^x_a$ for some $s \in S \subseteq T$ and $a \in f(s) \subseteq h(s)$. Consequently, $s' \in T^x_h$.

Conversely, for $T^x_h \subseteq S^x_f \cup U^x_g$, let $s' \in T^x_h$, i.e., $s' = s^x_a$ for some $s \in T$ and $a \in h(s)$. If $s \in S \setminus U$, then necessarily $a \in f(s)$, and $s^x_a \in S^x_f$. Likewise, if $s \in U \setminus S$, then $a \in g(s)$ and $s^x_a \in U^x_g$. Finally, if $s \in S \cap U$, then $a \in f(s) \cup g(s)$, so s^x_a is either in S^x_f or in U^x_g.

- (E$\forall$) states that the duplicating team $T^x_{\mathcal{A}}$ of T contains all supplementing teams of T as a subteam.
- (I$\forall$) states that the duplicating team $T^x_{\mathcal{A}}$ of T is a supplementing team, namely by the full supplementing function $f\colon T \to \wp^+(\mathcal{A})$ with $f(s) = \mathcal{A}$.
- (Dis!), (Dis$\Box$) and (Nec!) work as their modal counterparts in Table 6.8. ☐

Lemma 6.52. *For each $x \in \mathsf{Var}$, the logic $\mathcal{B}(\mathsf{FO})$ has $\exists x$-elimination and $\forall x$-elimination in* $\mathsf{H^{FO}LSQ}$.

Proof. Shown identically as for the modal operators $\Diamond$ and $\Box$. All necessary proofs in Subsection 6.3.2 and in the appendix are valid proofs in $\mathsf{H^{FO}LSQ}$ when each $\Diamond$ is replaced by $\exists x$, $\Box$ by $\forall x$, and $\triangle$ by $!x$. ☐

Theorem 6.53. *The system* $\mathsf{H^{FO}ULSQ}$ *axiomatizes* $\mathsf{FO}(\sim)$.

Proof. The completeness is analogous to that of $\mathsf{ML}(\sim)$ (Theorem 6.50), since by the above lemma we have elimination of $\exists x$ and $\forall x$ for every variable $x \in \mathsf{Var}$. The soundness follows from Lemma 6.51. Hence, we can again apply Lemma 6.33 together with Theorem 6.32. ☐

As a consequence of completeness of $\mathsf{FO}(\sim)$, we also obtain compactness. Note that we needed compactness in Lemma 6.31, but in fact only that of $\mathsf{FO} \cup {\sim}\mathsf{FO}$, which can be proven by other means (cf. [100]).

Corollary 6.54 (cf. Corollary 5.41). *$\mathsf{FO}(\sim)$ satisfies the compactness theorem.*

Corollary 6.55 (cf. Theorem 5.6). *$\mathrm{SAT}(\mathsf{FO}(\sim))$ is complete for Π^0_1, and $\mathrm{VAL}(\mathsf{FO}(\sim)$ is complete for Σ^0_1.*

6.4 A remark on the empty team

Most existing literature deals with team logics that have the empty team property. For this reason, the empty team often is excluded from questions of complexity and definability (cf. e.g., [59, 128, 135]). This goes back to Hodges [70], who defined *trumps* (his term for teams), as non-empty sets of assignments that satisfy a formula.

The sense and meaning of the empty team $\emptyset$, or of the contradictory formula $\bot$, is worth discussing. There are some arguments to consider it purely as an artifact, or as a technical detail, that $\emptyset \vDash \bot$ holds. At least it is unintuitive and often requires special treatment.[1] Rönnholm [123] argued that the empty team naturally corresponds to the absence of data, but lacks more useful interpretations. For example, in the epistemic interpretation, or in the related inquisitive setting [16], a team represents a set of possible states, but then should contain at least the *actual* state.

In this section, we discuss this matter from a proof-theoretic perspective, and also show that the notions of consistency and satisfiability are subtle and deserve some attention in team logic. A set Φ is called *absolutely inconsistent* if $\Phi \vdash \varphi$ for all formulas φ, and it is *$\bot$-inconsistent* if $\Phi \vdash \bot$. Moreover, it is *Aristotle inconsistent* if $\Phi \vdash \varphi, \neg\varphi$ for some formula φ. In classical logic, all these conditions coincide, and by the completeness theorem are equivalent to unsatisfiability.

Proposition 6.56. *Let $\mathcal{F} \in \{\mathsf{PL}, \mathsf{ML}, \mathsf{FO}\}$ and $\Gamma \subseteq \mathcal{F}$. The following are equivalent:*

- $\Gamma \vdash \mathcal{F}$
- $\Gamma \vdash \bot$
- $\Gamma \vdash \alpha, \neg\alpha$ *for some* $\alpha \in \mathcal{F}$.
- Γ *is unsatisfiable is classical semantics.*

A logical constant $\bot$ is called *proof-theoretic falsum* if $\bot \vdash \varphi$ for every φ, and a connective $\neg$ is a *proof-theoretic negation* if $\neg\varphi$ is derivable whenever $\varphi \vdash \bot$ [118]. A *semantic falsum* is a formula that is never true, and a *semantic negation* is a unary connective that inverts the truth of its argument. In classical logic, these of course coincide with proof-theoretic falsum and negation.

In team semantics of classical logics, however, the above notions of inconsistency still coincide, but every formula is true in the empty team. Consequently, proof-theoretic

[1] The paper *On Definability in Dependence Logic* [88] characterized the properties definable in dependence logic by means of $\mathsf{SO}(\exists)$. Later, an erratum [87] appeared solely to address the issue that this characterization does not extend to the empty team.

falsum and negation exist, but semantical falsum and negation do not. Also, a set Φ of formulas can be satisfiable but inconsistent. Does this mean that the completeness theorem fails for team logic? Obviously, the matter is more complicated, since we have completeness by Corollary 6.7. This apparent contradiction is of course due to the lack of semantic negation and falsum: If the semantic falsum $\perp\!\!\!\perp$ is available, then inconsistency of φ would mean that it is derivable from φ, so φ cannot be satisfiable. However, in classical logics with team semantics, $\perp\!\!\!\perp$ is simply not a formula.

One could exclude the empty team to obtain a more "natural" behaviour. Let $\mathcal{F}^+$ denote the restriction of the classical logic $\mathcal{F}$ (with team semantics) to only valuations with non-empty teams. Then the consistent sets again become exactly the satisfiable sets:

Proposition 6.57. *Let $\mathcal{F} \in \{\mathsf{PL}, \mathsf{ML}, \mathsf{FO}\}$ and $\Gamma \subseteq \mathcal{F}$. The following are equivalent:*

- $\Gamma \vdash \mathcal{F}$
- Γ *is unsatisfiable is classical semantics.*
- Γ *is unsatisfiable by non-empty teams.*

In fact, $\mathcal{F}^+$ is axiomatizable: Clearly $\Gamma \vDash_{\mathcal{F}} \alpha$ implies $\Gamma \vDash_{\mathcal{F}^+} \alpha$. Since valuations with the empty team satisfy every α, the converse is also true. So $\vDash_{\mathcal{F}} = \vDash_{\mathcal{F}^+}$, and since $\mathcal{F}$ is axiomatizable, so is $\mathcal{F}^+$.

Proposition 6.58. *A proof system is sound resp. complete for $\mathcal{F} \in \{\mathsf{PL}, \mathsf{ML}, \mathsf{FO}\}$ if and only if it is sound resp. complete for $\mathcal{F}^+$.*

How useful is the logic $\mathcal{F}^+$? Now, $\perp$ is a semantical falsum when excluding the empty team, yet $\neg$ is still no semantical negation. In particular, the law of excluded middle still fails, i.e., there are formulas α and valuations satisfying neither α nor $\neg\alpha$ under team semantics (for instance p in the team $T = \{p \mapsto 0, p \mapsto 1\}$). Furthermore, excluding the empty team has unintuitive side effects, for instance, the formula $\perp \vee \top$ would be unsatisfiable instead of valid and hence contradict the flatness property.

In $\mathcal{B}(\mathcal{F})$, the picture changes, and much of the classical behaviour is restored. The operators $\sim$ and $\perp\!\!\!\perp$ are both the semantic and proof-theoretic negation and falsum. In order to express non-emptiness, one can simply use the formula $\textsc{ne} = {\sim}\perp$.

Proposition 6.59. *Let $\mathcal{F} \in \{\mathsf{PL}, \mathsf{ML}, \mathsf{FO}\}$ and $\Phi \subseteq \mathcal{B}(\mathcal{F})$. The following are equivalent:*

- $\Phi \vdash \mathcal{B}(\mathcal{F})$
- $\Phi \vdash \perp\!\!\!\perp$
- $\Phi \vdash \varphi, {\sim}\varphi$ *for some* $\varphi \in \mathcal{B}(\mathcal{F})$
- Φ *is unsatisfiable under team semantics.*

But now we have that $\{\alpha, \neg\alpha\} \vdash \mathcal{F} \neq \mathcal{B}(\mathcal{F})$, so the set $\{\alpha, \neg\alpha\}$ is *consistent* relative to $\mathcal{B}(\mathcal{F})$. Mossakowski and Schröder [118] call a logic *paraconsistent* that has a negation $\neg$ for which Aristotle inconsistency does not imply absolute inconsistency. But it is hardly justifiable to call team logic a paraconsistent logic, as this behaviour once more rather is an artifact due to the empty team being allowed, and due to the fact that team logic consists of two "layers" of logic stacked on top of each other. Forbidding the empty team again, we would obtain $\bot \vDash \mathcal{B}(\mathcal{F})$ and avoid paraconsistency, but it seems preferable to allow it in general and instead require the team to be non-empty where it is necessary.

6.5 Summary and outlook

6.5.1 Summary

We axiomatized the logics $\mathsf{PL}(\sim)$, $\mathsf{ML}(\sim)$ and $\mathsf{FO}(\sim)$, and for this proceeded in several steps. First, the Boolean connectives $\wedge$ and $\sim$ have been captured by the system L, then the disjunction $\vee$ by the system S, and finally the modalities and quantifiers by M and Q, respectively. For first-order logic, we additionally required the axiom U, $\sim\alpha \multimap \neg\alpha$ for sentences α, to achieve refutation completeness on the level of literals. Propositional and modal team logic do not require this step as they do not have sentences. Together with existing complete proof systems for the underlying classical logics PL, ML and FO, this yields the full axiomatizations. Below, in Table 6.11, we present an overview of the introduced rules and axioms. Since the (truth-functional) non-classical atoms of dependence, independence, inclusion and exclusion can be efficiently defined in the above logics with only polynomial blow-up [108], adding these translations as axioms almost trivially leads to sound and complete proof systems for propositional and modal logics of dependence, independence, and so on.[1]

Comparison to existing results. We proceed with comparing our results with the existing approaches in literature. Most notably, we presented a *Hilbert-style* proof system (i.e., mostly axioms and only a few rules), whereas most other authors proposed Gentzen-style proof systems of *sequent calculi* and *natural deduction* (i.e., only rules and no axioms) for various team logics. This includes Yang and Väänänen [143, 144] and Yang [141] in the propositional and modal setting, and Kontinen and Väänänen [86] and Galliani [36] for fragments of first-order team logic. Hilbert-style systems, and a tableau calculus, were presented by Sano and Virtema [124] as well as Yang and Väänänen [144] for modal dependence logic.

This bias is unsurprising, since Hilbert-style systems are infamous for being rather opaque and requiring lengthy proofs even for simple theorems. In particular, for practical purposes they lack the *subformula property* that many Gentzen-style systems enjoy, which means that it suffices to have subformulas of the premises and/or conclusion occurring in the proofs [21]. Nevertheless, it turned out as a fruitful approach to con-

[1] In [100], which appeared before [108], these axioms have been stated explicitly.

sider not only $\Diamond$, $\Box$, but also the connectives $\exists$, $\forall$ and in particular $\vee$ as *modalities*, and to include Hilbert-style necessitation rules (cf. Tables 6.5 and 6.8).

Although the axiomatization of modal logic-like operators is also possible by means of natural deduction, this creates new technical pitfalls. For instance, one has to distinguish between "local" and "global" subproofs, since the necessitation rule has no simple correspondent in natural deduction [33].

Yang [141] proposed a Hilbert-style calculus that works in a similar fashion as ours, and is sound and complete for the downward closed fragment $\mathsf{ML}(\mathsf{dep}, \varovee)$ obtained from adding $\varovee$ and $\mathsf{dep}(\cdot;\cdot)$ to ML. By downward closure of the logic, her normal form consisted of Boolean disjunctions of flat formulas instead of full $\mathcal{B}(\mathsf{ML})$, but essentially the idea is the same. As these results occurred independently [99, 141], the method of operator elimination seems to be a viable option for the axiomatization of team logic.

The other major difference to existing work is that we include the contradictory negation $\sim$. Previous authors have avoided $\sim$ as part of the logic because their approaches rely on downward closure (in particular, Yang and Väänänen [143, Thm. 4.7], Sano and Virtema [124, Lem. 21], and Kontinen and Väänänen [86, Lem. 8]). It is noteworthy that even Yang and Väänänen [144] considered not the full logic $\mathsf{PL}(\sim)$, but only an expressively equivalent fragment of it where $\sim$ is rewritten in terms of other connectives.

Here, we instead embrace $\sim$ as a primitive connective and hence can employ propositional calculus to lift much of the heavy work. This not only generalizes the existing axiomatizability results towards logics closed under Boolean connectives, but also permits a notably simpler set of rules. For example, Yang and Väänänen [144] included

$$\frac{\left(\bigvee_{s\in X}(p_{i_1}^{s(i_1)} \wedge \cdots \wedge p_{i_n}^{s(i_n)} \wedge \textsc{ne})\right) \wedge \left(\bigvee_{s\in Y}(p_{i_1}^{s(i_1)} \wedge \cdots \wedge p_{i_n}^{s(i_n)} \wedge \textsc{ne})\right)}{\bot \wedge \textsc{ne}}$$

as the rule "Strong contradiction introduction" for all distinct propositional teams X, Y with domain $i_1, \ldots, i_n$. Intuitively, it says that if a team T equals X and Y at the same time then T cannot exist, i.e., $T \vDash \perp\!\!\!\perp \equiv \bot \wedge \textsc{ne}$. Admittedly, our completeness proof required complicated distributive laws such as $\Diamond(\alpha \wedge \mathsf{E}\beta) \equiv \Diamond\alpha \wedge \mathsf{E}\Diamond(\alpha \wedge \beta)$ instead, but nonetheless we showed that they are provable from simpler axioms.

6.5.2 Open problems and further research directions

First of all, it is desirable to find a natural and simple system of sequent calculus or natural deduction for the full logics $\mathsf{PL}(\sim)$, $\mathsf{ML}(\sim)$ and $\mathsf{FO}(\sim)$ that also accounts for $\sim$. Also, it would be interesting to see whether our approach extends to other team logics, say, in the first-order setting. Just like dependence logic $\mathsf{FO}(\mathsf{dep})$, team logic $\mathsf{FO}(\mathsf{dep}, \sim)$ is not axiomatizable [135]. But is there some hope to find a new partial proof system? For example, can we axiomatize all $\mathsf{FO}(\sim)$-consequences of $\mathsf{FO}(\mathsf{dep}, \sim)$-formulas in the spirit of Kontinen and Väänänen [86], who axiomatized all FO-consequences of $\mathsf{FO}(\mathsf{dep})$-formulas?

Also, small modifications like adding the *universal modality* $\boxdot$ (cf. Example 3.63) to modal team logic leads to issues for the axiomatization. The reason is that the logic then in a sense has "sentences", and the technique of counter-model merging (Proposition 6.24) fails, as modal logic with $\boxed{u}$ has no longer the property that truth is preserved under disjoint union of Kripke structures. It would be interesting to see whether an axiom similar to U could help here.

Finally, one can combine the axiomatization with the ideas of Chapter 3. We showed in Chapter 3 in an abstract way that certain team logics permit a translation into a normal form based on Boolean combinations of flat formulas. In this chapter, we used a similar result, but additionally had to carry out this translation in our proof system. Is it possible to combine these approaches, and to find an axiomatization for arbitrary team logics that are quasi-flat in the sense of Chapter 3?

The necessitation and distribution axioms are easy to adapt to arbitrary arities:

$$\text{(Nec)}: \quad \frac{\varphi}{\triangle(\perp\!\!\!\perp, \cdots, \perp\!\!\!\perp, \varphi, \perp\!\!\!\perp, \cdots, \perp\!\!\!\perp)}$$

$$\text{(Dis)}: \ \triangle(\perp\!\!\!\perp, \cdots, \perp\!\!\!\perp, \varphi_i \multimap \psi, \perp\!\!\!\perp, \cdots, \perp\!\!\!\perp) \multimap \triangle(\varphi_1, \dots, \varphi_r) \multimap \triangle(\varphi_1, \dots, \psi, \dots, \varphi_r)$$

The other axioms are more difficult. For instance, for a generalized diamond $\triangle$, (E$\Box$) becomes

$$\neg\triangle\neg(\bot, \dots, \alpha, \dots, \bot) \multimap {\sim}\triangle{\sim}(\perp\!\!\!\perp, \dots, \alpha, \dots, \perp\!\!\!\perp)$$

For axioms such as (Lax$\vee$), it is open how they can be generalized.

L	(L1)	$\varphi \rightarrowtriangle (\psi \rightarrowtriangle \varphi)$	
	(L2)	$(\varphi \rightarrowtriangle (\psi \rightarrowtriangle \theta)) \rightarrowtriangle (\varphi \rightarrowtriangle \psi) \rightarrowtriangle (\varphi \rightarrowtriangle \theta)$	
	(L3)	$({\sim}\varphi \rightarrowtriangle {\sim}\psi) \rightarrowtriangle (\psi \rightarrowtriangle \varphi)$	
	(L4)	$(\varphi \wedge \psi) \rightarrowtriangle \varphi$	
	(L5)	$(\varphi \wedge \psi) \rightarrowtriangle \psi$	
	(L6)	$\varphi \rightarrowtriangle (\psi \rightarrowtriangle (\varphi \wedge \psi))$	
	(L7)	$(\alpha \rightarrow \beta) \rightarrowtriangle (\alpha \rightarrowtriangle \beta)$	
	(E$\rightarrowtriangle$)	$\dfrac{\varphi \qquad \varphi \rightarrowtriangle \psi}{\psi}$	
U	(U)	${\sim}\alpha \rightarrowtriangle \neg\alpha$	(α sentence)
S	(Dual$\multimap$)	$(\varphi \vee \psi) \leftrightarrowtriangle {\sim}(\varphi \multimap {\sim}\psi)$	
	(Sub$\vee$)	$\alpha \rightarrowtriangle (\varphi \multimap \alpha)$	
	(Lax$\vee$)	$\varphi \rightarrowtriangle (\theta \vee \psi) \rightarrowtriangle (\varphi \vee \psi)$	
	(Ass$\vee$)	$(\varphi \vee (\psi \vee \theta)) \rightarrowtriangle ((\varphi \vee \psi) \vee \theta)$	
	(Com$\vee$)	$(\varphi \vee \psi) \rightarrowtriangle (\psi \vee \varphi)$	
	(Dis$\multimap$)	$(\varphi \multimap (\psi \rightarrowtriangle \theta)) \rightarrowtriangle (\varphi \multimap \psi) \rightarrowtriangle (\varphi \multimap \theta)$	
	(Nec$\multimap$)	$\dfrac{\varphi}{\psi \multimap \varphi}$	(φ theorem)
M	(Dual$\Diamond$)	$\Diamond\varphi \leftrightarrowtriangle {\sim}\triangle{\sim}\varphi$	
	(Lin$\Box$)	$\Box{\sim}\varphi \leftrightarrowtriangle {\sim}\Box\varphi$	
	(Dis$\Diamond\vee$)	$\Diamond(\varphi \vee \psi) \leftrightarrowtriangle (\Diamond\varphi \vee \Diamond\psi)$	
	(E$\Box$)	$\Box\alpha \rightarrowtriangle \triangle\alpha$	
	(I$\Box$)	$\Diamond\varphi \rightarrowtriangle (\triangle\psi \rightarrowtriangle \Box\psi)$	
	(Dis$\Box$)	$\Box(\varphi \rightarrowtriangle \psi) \rightarrowtriangle (\Box\varphi \rightarrowtriangle \Box\psi)$	
	(Dis$\triangle$)	$\triangle(\varphi \rightarrowtriangle \psi) \rightarrowtriangle (\triangle\varphi \rightarrowtriangle \triangle\psi)$	
	(Nec$\Box$)	$\dfrac{\varphi}{\Box\varphi}$	(φ theorem)
	(Nec$\triangle$)	$\dfrac{\varphi}{\triangle\varphi}$	(φ theorem)
Q	(Dual$\exists$)	$\exists x\, \varphi \leftrightarrowtriangle {\sim}\, !x\, {\sim}\varphi$	
	(Lin$\forall$)	$\forall x\, {\sim}\varphi \leftrightarrowtriangle {\sim}\forall x\, \varphi$	
	(Dis$\exists\vee$)	$\exists x(\varphi \vee \psi) \leftrightarrowtriangle \exists x\, \varphi \vee \exists x\, \psi$	
	(E$\forall$)	$\forall x\, \alpha \rightarrowtriangle\, !x\, \alpha$	
	(I$\forall$)	$!x\, \psi \rightarrowtriangle \forall x\, \psi$	
	(Dis$\Box$)	$\forall x(\varphi \rightarrowtriangle \psi) \rightarrowtriangle (\forall x \varphi \rightarrowtriangle \forall x\, \psi)$	
	(Dis!)	$!x(\varphi \rightarrowtriangle \psi) \rightarrowtriangle (!x \varphi \rightarrowtriangle\, !x\, \psi)$	
	(Nec!)	$\dfrac{\varphi}{!x\varphi}$	(φ theorem)

Table 6.11: Overview of the systems L, U, S, M, **and** Q

7 Conclusion

Summary

The thesis can be divided into two parts: One on abstract team logic and one on concrete team logics.

In the first part of the thesis, Chapter 3, we studied team logic in general. We identified *teamification* as a pattern which ensures that a team logic is a well-behaved and faithful extension of the classical logic it is based on. Next, we showed that most common team-logical connectives are *operators* in the sense of Boolean algebras with operators (BAOS). As mentioned in Section 3.9, a fully algebraic and abstract description of team logic in terms of BAOS is a promising goal for future research. We also identified *transversals*—operators whose action on a team is determined by those on its elements—as a natural class of operators that preserve flatness. This series of restrictions on the operators culminated in *lax standard transversals*, and in the result that team logics based on those, e.g., $\mathsf{PL}(\sim)$, $\mathsf{ML}(\sim)$ and $\mathsf{FO}(\sim)$, collapse to the Boolean closure of classical formulas.

This result was also crucial for the complexity of these logics in Chapters 4 and 5 and their axiomatizations in Chapter 6, which form the second part of the thesis. In Chapter 5, we proved that $\mathsf{FO}(\sim)$ has the same complexity as FO, and that it admits a compactness theorem, which we proved with an adaptation of Łoś's ultraproduct theorem to team logic. Moreover, we showed that it mirrors classical logic also in the sense that it encompasses $\mathsf{ML}(\sim)$, $\mathsf{GF}(\sim)$ and $\mathsf{FO}^2(\sim)$ as decidable fragments, namely having a satisfiability problem that is complete for the non-elementary class TOWER(poly). The lower bound for this complexity was shown for $\mathsf{ML}(\sim)$ in Chapter 4 by succinctly enforcing canonical models. In Chapter 6, we cast the transformation into said Boolean normal form into a proof system, and by this axiomatized the above logics.

Discussion

Let us draw some final conclusions and discuss the results of this thesis. The main insight from the first part is that existing team semantics seems carefully designed to ensure that the respective logics are well-behaved and follow certain patterns. By studying the *teamification* pattern in particular, we formally approached the question whether a team logic is necessarily based on a classical logic. We showed that a team logic is a teamification of some classical logic precisely if all its connectives preserve flatness. This again seems to be an indispensable feature of any well-behaved team logic (leaving aside the negation $\sim$ and non-classical atoms, as these are not corresponding to any classical connectives).

One could even argue that, in a sense, we studied only *one* team logic in this thesis that manifests in the different formalisms (propositional, modal, first-order), but is uniformly defined. The similarities between the team-logical operators are striking. Table 3.1 illustrates them as *standard transversals*, displaying a row $\wedge, \Diamond, \dot{\Diamond}, \exists, \dot{\exists}$ of "diamonds" and a row $\vee, \dot{\vee}, \Box, \forall$ of "boxes".

We consider two possible interpretations of this fact. One is that propositional and modal team logic have originally been meant to mirror the semantics of first-order team logic as close as possible, and that alternative definitions besides strict and lax semantics have not been considered much. The other is that the constraint of having flatness preserving connectives in a team logic—and hence being limited to teamifications—is so restrictive that no other sensible choice exists but (strict and lax) transversals.

A strong argument in favor of the first alternative is the fact that, for example, temporal logic after all noticeably deviates from this pattern. Of its connectives, only $\mathsf{F}^{\mathfrak{a}}$ and X are standard transversals. By contrast, $\mathsf{G}^{\mathfrak{a}}$, $\mathsf{G}^{\mathfrak{s}}$ and $\mathsf{U}^{\mathfrak{a}}$ are flatness preserving but no operators, and $\mathsf{F}^{\mathfrak{s}}$ is an operator, but not flatness preserving, and consequently none of them is a transversal. Still, these definitions seem to be natural generalizations of classical LTL. This suggests the conclusion that the framework presented in this thesis has much potential to be further generalized and relaxed, in hope to classify more types of natural team-logical connectives.

The main technical result of the first part is the collapse theorem, i.e., that every formula using only lax standard transversals and negation can be written as a Boolean combination of flat formulas. This result seems to severely limit the expressiveness of, e.g., $\mathsf{PL}(\sim)$, $\mathsf{ML}(\sim)$ and $\mathsf{FO}(\sim)$. On the other hand, it is strongly hinged on lax semantics. While the latter also implies other natural properties such as locality [37], in this light, all complexity upper bounds obtained in this thesis seem rather fragile. From this perspective, finding similar normal forms for non-lax semantics is definitely worth pursuing.

In Chapters 4 and 5, we demonstrated that the satisfiability problems of $\mathsf{ML}(\sim)$, $\mathsf{GF}(\sim)$ and $\mathsf{FO}^2(\sim)$ are all complete for the class $\textsc{Tower}(\text{poly})$. This proves that the Boolean negation vastly increases the complexity of these problems. The surprisingly uniform complexity of the different logics ultimately reflects the non-elementary succinctness gap between the mentioned Boolean combinations of flat formulas and the respective full logic, in which the original complexity-wise differences between ML, GF and FO^2 vanish. In a sense, these completeness results refine the expressiveness result of Chapter 3 in a *quantitative* sense.

In a nutshell, the Boolean negation seems to drastically increase the *succinctness*, but only marginally the *expressiveness* of team logic, which comes down to Boolean combinations of flat formulas. By contrast, non-classical atoms such as the dependence atom tremendously increase the *expressiveness*—in a sense, from first-order to second-order logic—but the non-classical atoms yield no additional *succinctness*.

Team logic was designed as a compositional semantics for logic of imperfect information, and as a framework for dependency notions in logic. Upon a closer look, however, its rich and fascinating structure surpasses these purposes by far.

Bibliography

[1] Samson Abramsky, Juha Kontinen, Jouko Väänänen and Heribert Vollmer. *Dependence Logic: Theory and Applications*. Cham: Birkhäuser / Springer, 2016.

[2] Samson Abramsky and Jouko Väänänen. *From IF to BI*. Synthese **167**(2), 2009, pp. 207–230.

[3] Antonis Achilleos, Michael Lampis and Valia Mitsou. *Parameterized Modal Satisfiability*. Algorithmica **64**(1), 2012, pp. 38–55.

[4] Hajnal Andréka, István Németi and Johan van Benthem. *Modal Languages and Bounded Fragments of Predicate Logic*. Journal of Philosophical Logic **27**(3), 1998, pp. 217–274.

[5] Hajnal Andréka, István Németi and Ildikó Sain. *Algebraic Logic*. In: Handbook of Philosophical Logic. Ed.: Dov M. Gabbay and F. Guenthner. Dordrecht: Springer Netherlands, 2001, pp. 133–247.

[6] William W. Armstrong. *Dependency Structures of Data Base Relationships*. Information Processing, Proc. of the 6th IFIP Congress 1974, Stockholm, Sweden, August 5-10, 1974. Ed.: J. Rosenfeld. North Holland / Elsevier, 1974, pp. 580–583.

[7] Jon Barwise. *On branching quantifiers in English*. Journal of Philosophical Logic **8**(1), 1979, pp. 47–80.

[8] Johan van Benthem. *Dynamic bits and pieces*. Institute for Logic, Language and Computation (ILLC), University of Amsterdam, 1997.

[9] Patrick Blackburn and Johan van Benthem. *1 Modal logic: A semantic perspective*. In: Handbook of Modal Logic. Ed.: P. Blackburn, J. van Benthem and F. Wolter. **3**. Studies in Logic and Practical Reasoning. Elsevier Science, 2007, pp. 1–84.

[10] Patrick Blackburn, Johan van Benthem and Frank Wolter. *Handbook of Modal Logic*. Vol. 3. Studies in Logic and Practical Reasoning. Elsevier Science, 2007.

[11] George Boole. *An investigation of the laws of thought, on which are founded the mathematical theories of logic and probabilities*. London: Walton & Maberly, 1854.

[12] Egon Börger, Erich Grädel and Yuri Gurevich. *The Classical Decision Problem*. Universitext. Berlin, Heidelberg: Springer, 1997.

[13] Ashok K. Chandra, Dexter C. Kozen and Larry J. Stockmeyer. *Alternation*. Journal of the ACM **28**(1), Jan. 1981, pp. 114–133.

[14] Chen C. Chang and Howard J. Keisler. *Model Theory*. 3rd ed. Dover Books on Mathematics. Dover Publications, 2012.

[15] Alonzo Church. *An unsolvable problem of elementary number theory*. American Journal of Mathematics **58**(2), 1936, pp. 345–363.

[16] Ivano Ciardelli. *Dependency as Question Entailment*. In: Dependence Logic: Theory and Applications. Ed.: S. Abramsky, J. Kontinen, J. Väänänen and H. Vollmer. Cham: Birkhäuser / Springer, 2016, pp. 129–181.

[17] Kevin J. Compton and C. Ward Henson. *A uniform method for proving lower bounds on the computational complexity of logical theories*. Annals of Pure and Applied Logic **48**(1), 1990, pp. 1–79.

[18] Stephen A. Cook. *The Complexity of Theorem-Proving Procedures*. Proc. of the 3rd Annual ACM Symp. on Theory of Computing. Ed.: M. Harrison, R. Banerji and J. Ullman. ACM, 1971, pp. 151–158.

[19] Max J. Cresswell. *KM and the finite model property*. Notre Dame Journal of Formal Logic **24**(3), 1983, pp. 323–327.

[20] Max J. Cresswell and George E. Hughes. *A New Introduction to Modal Logic*. London: Routledge, 1996.

[21] Dirk van Dalen. *Logic and Structure*. 5th ed. Universitext. London: Springer, 2013.

[22] Brian A. Davey and Hilary A. Priestley. *Introduction to Lattices and Order*. 2nd ed. Cambridge University Press, 2002.

[23] Stéphane Demri and Raul Fervari. *On the Complexity of Modal Separation Logics*. Advances in Modal Logic 12, Proc. of the 12th Conf. on "Advances in Modal Logic," held in Bern, Switzerland, August 27-31, 2018. Ed.: G. Bezhanishvili, G. D'Agostino, G. Metcalfe and T. Studer. College Publications, 2018, pp. 179–198.

[24] Klaus Denecke and Shelly L. Wismath. *Universal algebra and coalgebra*. Singapore, Hackensack, NJ: World Scientific, 2009.

[25] R. G. Downey and M. R. Fellows. *Parameterized Complexity*. Springer New York, 1999.

[26] Arnaud Durand, Miika Hannula, Juha Kontinen, Arne Meier and Jonni Virtema. *Approximation and dependence via multiteam semantics*. Annals of Mathematics and Artificial Intelligence **83**(3-4), 2018, pp. 297–320.

[27] Arnaud Durand, Miika Hannula, Juha Kontinen, Arne Meier and Jonni Virtema. *Probabilistic Team Semantics*. Foundations of Information and Knowledge Systems - 10th Int. Symp., FoIKS 2018, Budapest, Hungary, May 14-18, 2018, Proc. Ed.: F. Ferrarotti and S. Woltran. **10833**. Lecture Notes in Computer Science. Cham: Springer, 2018, pp. 186–206.

[28] Arnaud Durand, Juha Kontinen and Heribert Vollmer. *Expressivity and Complexity of Dependence Logic*. In: Dependence Logic: Theory and Applications. Ed.: S. Abramsky, J. Kontinen, J. Väänänen and H. Vollmer. Cham: Birkhäuser / Springer, 2016, pp. 5–32.

[29] Johannes Ebbing and Peter Lohmann. *Complexity of Model Checking for Modal Dependence Logic*. SOFSEM 2012: Theory and Practice of Computer Science. **7147**. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2012, pp. 226–237.

[30] Ernest Allen Emerson. *Temporal and Modal Logic*. In: Handbook of Theoretical Computer Science, Volume B: Formal Models and Sematics. MIT Press, 1990, pp. 995–1072.

[31] Herbert B. Enderton. *A Mathematical Introduction to Logic*. 2nd ed. Elsevier Science, 2001.

[32] Kit Fine. *Normal forms in modal logic*. Notre Dame Journal of Formal Logic **16**(2), 1975, pp. 229–237.

[33] Melvin Fitting. *2 Modal proof theory*. In: Handbook of Modal Logic. Ed.: P. Blackburn, J. van Benthem and F. Wolter. **3**. Studies in Logic and Practical Reasoning. Elsevier Science, 2007, pp. 85–138.

[34] Gottlob Frege. *Logische Untersuchungen III: Gedankengefüge, 1923*. In: Kleine Schriften. Ed.: I. Angelelli. Hildesheim: Georg Olms, 1967.

[35] Dov M. Gabbay. *Investigations in Modal and Tense Logics with Applications to Problems in Philosophy and Linguistics*. Dordrecht: Springer Netherlands, 1976.

[36] Pietro Galliani. *General Models and Entailment Semantics for Independence Logic*. Notre Dame Journal of Formal Logic **54**(2), 2013, pp. 253–275.

[37] Pietro Galliani. *Inclusion and exclusion dependencies in team semantics—on some logics of imperfect information*. Annals of Pure and Applied Logic **163**(1), 2012, pp. 68–84.

[38] Pietro Galliani. *On Strongly First-Order Dependencies*. In: Dependence Logic: Theory and Applications. Ed.: S. Abramsky, J. Kontinen, J. Väänänen and H. Vollmer. Cham: Birkhäuser / Springer, 2016, pp. 53–71.

[39] Pietro Galliani. *Team Semantics for Spatial Reasoning: Locality and Separability*. Joint Proc. of the Workshops C3GI: The 7th Int. Workshop on Computational Creativity, Concept Invention, and General Intelligence ISD4: The 4th Image Schema Day, and SCORE: From Image Schemas to Cognitive Robotics, Bozen-Bolzano, Italy, December 13-15, 2018. Ed.: O. Kutz, M. Hedblom, T. Besold, R. Confalonieri, C. León, T. Veale and J. Bateman. **2347**. CEUR Workshop Proc. CEUR-WS.org, 2018.

[40] Pietro Galliani. *The Doxastic Interpretation of Team Semantics*. In: Logic Without Borders - Essays on Set Theory, Model Theory, Philosophical Logic and Philosophy of Mathematics. Ed.: Å. Hirvonen, J. Kontinen, R. Kossak and A. Villaveces. **5**. Ontos Mathematical Logic. De Gruyter, 2015, pp. 167–192.

[41] Pietro Galliani. *Upwards closed dependencies in team semantics*. Information and Computation **245**, 2015, pp. 124–135.

[42] Steven Givant. *Duality Theories for Boolean Algebras with Operators*. Springer Monographs in Mathematics. Cham: Springer, 2014.

[43] Valentin Goranko and Martin Otto. *5 Model theory of modal logic*. In: Handbook of Modal Logic. Ed.: P. Blackburn, J. van Benthem and F. Wolter. **3**. Studies in Logic and Practical Reasoning. Elsevier Science, 2007, pp. 249–329.

[44] Erich Grädel. *Model-checking games for logics of imperfect information*. Theoretical Computer Science **493**, 2013, pp. 2–14.

[45] Erich Grädel. *On The Restraining Power of Guards*. The Journal of Symbolic Logic **64**(4), 1999, pp. 1719–1742.

[46] Erich Grädel and Stefan Hegselmann. *Counting in Team Semantics*. 25th EACSL Annual Conference on Computer Science Logic, CSL 2016, August 29 - September 1, 2016, Marseille, France. Ed.: J. Talbot and L. Regnier. **62**. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016, 35:1–35:18.

[47] Erich Grädel, Phokion G. Kolaitis, Leonid Libkin, Maarten Marx, Joel Spencer, Moshe Y. Vardi, Yde Venema and Scott Weinstein. *Finite Model Theory and Its Applications*. Texts in Theoretical Computer Science. An EATCS Series. Berlin, Heidelberg: Springer, 2007.

[48] Erich Grädel, Phokion G. Kolaitis and Moshe Y. Vardi. *On the Decision Problem for Two-Variable First-Order Logic*. Bulletin of Symbolic Logic **3**(1), 1997, pp. 53–69.

[49] Erich Grädel and Jouko Väänänen. *Dependence and Independence*. Studia Logica **101**(2), 2013, pp. 399–410.

[50] Raul Hakli and Sara Negri. *Does the deduction theorem fail for modal logic?* Synthese **187**(3), 2012, pp. 849–867.

[51] Miika Hannula. *Validity and Entailment in Modal and Propositional Dependence Logics*. Logical Methods in Computer Science **15**(2), 2019.

[52] Miika Hannula and Juha Kontinen. *A finite axiomatization of conditional independence and inclusion dependencies*. Information and Computation **249**, 2016, pp. 121–137.

[53] Miika Hannula, Juha Kontinen, Martin Lück and Jonni Virtema. *On Quantified Propositional Logics and the Exponential Time Hierarchy*. Proc. of the Seventh Int. Symp. on Games, Automata, Logics and Formal Verification, GandALF 2016, Catania, Italy, 14-16 September 2016. Ed.: D. Cantone and G. Delzanno. **226**. EPTCS. 2016, pp. 198–212.

[54] Miika Hannula, Juha Kontinen and Jonni Virtema. *Polyteam Semantics*. Logical Foundations of Computer Science - Int. Symp., LFCS 2018, Deerfield Beach, FL, USA, January 8-11, 2018, Proc. Ed.: S. Artëmov and A. Nerode. **10703**. Lecture Notes in Computer Science. Springer, 2018, pp. 190–210.

[55] Miika Hannula, Juha Kontinen, Jonni Virtema and Heribert Vollmer. *Complexity of Propositional Independence and Inclusion Logic*. Mathematical Foundations of Computer Science 2015 - 40th Int. Symp., MFCS 2015, Milan, Italy, August 24-28, 2015, Proc., Part I. Ed.: G. Italiano, G. Pighizzini and D. Sannella. **9234**. Lecture Notes in Computer Science. Springer, 2015, pp. 269–280.

[56] Miika Hannula, Juha Kontinen, Jonni Virtema and Heribert Vollmer. *Complexity of Propositional Logics in Team Semantic*. ACM Transactions on Computational Logic **19**(1), 2018, 2:1–2:14.

[57] Miika Hannula and Sebastian Link. *On the Interaction of Functional and Inclusion Dependencies with Independence Atoms*. Database Systems for Advanced Applications - 23rd Int. Conf., DASFAA 2018, Gold Coast, QLD, Australia, May 21-24, 2018, Proc., Part II. Ed.: J. Pei, Y. Manolopoulos, S. Sadiq and J. Li. **10828**. Lecture Notes in Computer Science. Springer, 2018, pp. 353–369.

[58] Lauri Hella, Antti Kuusisto, Arne Meier and Jonni Virtema. *Model Checking and Validity in Propositional and Modal Inclusion Logics*. 42nd Int. Symp. on Mathematical Foundations of Computer Science, MFCS 2017, August 21-25, 2017 - Aalborg, Denmark. Ed.: K. Larsen, H. Bodlaender and J. Raskin. **83**. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017, 32:1–32:14.

[59] Lauri Hella, Antti Kuusisto, Arne Meier and Heribert Vollmer. *Satisfiability of Modal Inclusion Logic: Lax and Strict Semantics*. ACM Trans. Comput. Log. **21**(1), 2020, 7:1–7:18.

[60] Lauri Hella, Kerkko Luosto, Katsuhiko Sano and Jonni Virtema. *The Expressive Power of Modal Dependence Logic*. Advances in Modal Logic. College Publications, 2014, pp. 294–312.

[61] Lauri Hella and Johanna Stumpf. *The expressive power of modal logic with inclusion atoms*. Proc. Sixth Int. Symp. on Games, Automata, Logics and Formal Verification, GandALF 2015, Genoa, Italy, 21-22nd September 2015. Ed.: J. Esparza and E. Tronci. **193**. EPTCS. 2015, pp. 129–143.

[62] Lauri Hella and Miikka Vilander. *The succinctness of first-order logic over modal logic via a formula size game*. Advances in Modal Logic 11, proc. of the 11th conference on "Advances in Modal Logic," held in Budapest, Hungary, August 30 - September 2, 2016. Ed.: L. Beklemishev, S. Demri and A. Maté. College Publications, 2016, pp. 401–419.

[63] Leon Henkin. *Some Remarks on Infinitely Long Formulas*. In: Journal of Symbolic Logic. **30**. Pergamon Press, 1961, pp. 167–183.

[64] Jaakko Hintikka. *Defining Truth, the Whole Truth and Nothing but the Truth*. In: Lingua Universalis vs. Calculus Ratiocinator: An Ultimate Presupposition of Twentieth-Century Philosophy. Dordrecht: Springer Netherlands, 1997, pp. 46–103.

[65] Jaakko Hintikka. *Quantifiers vs. quantification theory*. Linguistic inquiry **5**(2), 1974, pp. 153–177.

[66] Jaakko Hintikka. *The Principles of Mathematics Revisited*. Cambridge University Press, 1996.

[67] Jaakko Hintikka. *What is the True Algebra of Logic?* In: First-Order Logic Revisited. Ed.: V. Hendricks. Logos, 2004, pp. 117–128.

[68] Jaakko Hintikka and Gabriel Sandu. *Informational Independence as a Semantical Phenomenon*. In: Logic, Methodology and Philosophy of Science VIII. Ed.: J. Fenstad, I. Frolov and R. Hilpinen. **126**. Studies in Logic and the Foundations of Mathematics. Elsevier, 1989, pp. 571–589.

[69] Wilfrid Hodges. *Classical Logic I: First-order Logic*. In: The Blackwell Guide to Philosophical Logic. John Wiley & Sons, Ltd, 2017. Chap. 1, pp. 9–32.

[70] Wilfrid Hodges. *Compositional Semantics for a Language of Imperfect Information*. Logic Journal of the IGPL **5**(4), 1997, pp. 539–563.

[71] Wilfrid Hodges. *Some Strange Quantifiers*. Structures in Logic and Computer Science, A Selection of Essays in Honor of Andrzej Ehrenfeucht. Ed.: J. Mycielski, G. Rozenberg and A. Salomaa. **1261**. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 1997, pp. 51–65.

[72] Tapani Hyttinen, Gianluca Paolini and Jouko Väänänen. *A logic for arguing about probabilities in measure teams*. Archive for Mathematical Logic **56**(5-6), 2017, pp. 475–489.

[73] Tapani Hyttinen, Gianluca Paolini and Jouko Väänänen. *Quantum Team Logic and Bell's inequalities*. The Review of Symbolic Logic **8**(4), 2015, pp. 722–742.

[74] Rosalie Iemhoff and Fan Yang. *Structural completeness in propositional logics of dependence*. Archive for Mathematical Logic **55**(7-8), 2016, pp. 955–975.

[75] Theo M. V. Janssen. *Frege, Contextuality and Compositionality*. Journal of Logic, Language and Information **10**(1), 2001, pp. 115–136.

[76] Theo Janssen and Barbara Partee. *Compositionality*. In: Handbook of logic and language. North Holland / Elsevier, 1997, pp. 417–473.

[77] Bjarni Jónsson and Alfred Tarski. *Boolean algebras with operators. Part I*. American Journal of Mathematics **73**(4), 1951, pp. 891–939.

[78] Juha Kontinen. *On Natural Deduction in Dependence Logic*. In: Logic Without Borders - Essays on Set Theory, Model Theory, Philosophical Logic and Philosophy of Mathematics. Ed.: Å. Hirvonen, J. Kontinen, R. Kossak and A. Villaveces. **5**. Ontos Mathematical Logic. De Gruyter, 2015, pp. 297–304.

[79] Juha Kontinen, Antti Kuusisto, Peter Lohmann and Jonni Virtema. *Complexity of Two-Variable Dependence Logic and IF-Logic*. Proc. of the 26th Annual IEEE Symp. on Logic in Computer Science, LICS 2011, June 21-24, 2011, Toronto, Ontario, Canada. IEEE Computer Society, 2011, pp. 289–298.

[80] Juha Kontinen, Antti Kuusisto, Peter Lohmann and Jonni Virtema. *Complexity of two-variable dependence logic and IF-logic*. Information and Computation **239**, 2014, pp. 237–253.

[81] Juha Kontinen, Antti Kuusisto and Jonni Virtema. *Decidability of Predicate Logics with Team Semantics*. 41st Int. Symp. on Mathematical Foundations of Computer Science, MFCS 2016, August 22-26, 2016 - Kraków, Poland. Ed.: P. Faliszewski, A. Muscholl and R. Niedermeier. **58**. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016, 60:1–60:14.

[82] Juha Kontinen, Julian-Steffen Müller, Henning Schnoor and Heribert Vollmer. *A Van Benthem Theorem for Modal Team Semantics*. 24th EACSL Annual Conference on Computer Science Logic, CSL 2015, September 7-10, 2015, Berlin, Germany. Ed.: S. Kreutzer. **41**. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2015, pp. 277–291.

[83] Juha Kontinen, Julian-Steffen Müller, Henning Schnoor and Heribert Vollmer. *Modal independence logic*. Journal of Logic and Computation **27**(5), 2016, pp. 1333–1352.

[84] Juha Kontinen and Ville Nurmi. *Team Logic and Second-Order Logic*. Fundamenta Informaticae **106**(2-4), 2011, pp. 259–272.

[85] Juha Kontinen and Jouko Väänänen. *A Remark on Negation in Dependence Logic*. Notre Dame Journal of Formal Logic **52**(1), 2011, pp. 55–65.

[86] Juha Kontinen and Jouko Väänänen. *Axiomatizing first-order consequences in dependence logic*. Annals of Pure and Applied logic **164**(11), 2013, pp. 1101–1117.

[87] Juha Kontinen and Jouko Väänänen. *Erratum to: On Definability in Dependence Logic*. Journal of Logic, Language and Information **20**(1), 2011, pp. 133–134.

[88] Juha Kontinen and Jouko Väänänen. *On Definability in Dependence Logic*. Journal of Logic, Language and Information **18**(3), 2009, pp. 317–332.

[89] Juha Kontinen and Fan Yang. *Logics for First-Order Team Properties*. Logic, Language, Information, and Computation - 26th Int. Workshop, WoLLIC 2019, Utrecht, The Netherlands, July 2-5, 2019, Proc. Ed.: R. Iemhoff, M. Moortgat and R. de Queiroz. **11541**. Lecture Notes in Computer Science. Springer, 2019, pp. 392–414.

[90] Andreas Krebs, Arne Meier and Jonni Virtema. *A Team Based Variant of CTL*. 22nd Int. Symp. on Temporal Representation and Reasoning, TIME 2015, Kassel, Germany, September 23-25, 2015. Ed.: F. Grandi, M. Lange and A. Lomuscio. IEEE Computer Society, 2015, pp. 140–149.

[91] Andreas Krebs, Arne Meier, Jonni Virtema and Martin Zimmermann. *Team Semantics for the Specification and Verification of Hyperproperties*. 43rd Int. Symp. on Mathematical Foundations of Computer Science, MFCS 2018, August 27-31, 2018, Liverpool, UK. Ed.: I. Potapov, P. Spirakis and J. Worrell. **117**. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018, 10:1–10:16.

[92] Michał Krynicki and Marcin Mostowski. *Henkin Quantifiers*. In: Quantifiers: Logics, Models and Computation: Volume One: Surveys. Ed.: M. Krynicki, M. Mostowski and L. Szczerba. Dordrecht: Springer Netherlands, 1995, pp. 193–262.

[93] Antti Kuusisto. *A Double Team Semantics for Generalized Quantifiers*. Journal of Logic, Language and Information **24**(2), 2015, pp. 149–191.

[94] Richard E. Ladner. *The Computational Complexity of Provability in Systems of Modal Propositional Logic*. SIAM Journal on Computing **6**(3), 1977, pp. 467–480.

[95] Azriel Lévy. *A hierarchy of formulas in set theory*. 57. American Mathematical Soc., 1965.

[96] Peter Lohmann and Heribert Vollmer. *Complexity Results for Modal Dependence Logic*. Studia Logica **101**(2), 2013, pp. 343–366.

[97] Jerzy Łoś. *Quelques Remarques, Théorèmes Et Problèmes Sur Les Classes Définissables D'algèbres*. In: Mathematical Interpretation of Formal Systems. Ed.: Th. Skolem, G. Hasenjaeger, G. Kreisel, A. Robinson, Hao Wang, L. Henkin and J. Łoś. **16**. Studies in Logic and the Foundations of Mathematics. Elsevier, 1955, pp. 98–113.

[98] Leopold Löwenheim. *Über Möglichkeiten im Relativkalkül*. Mathematische Annalen **76**, 1915, pp. 447–470.

[99] Martin Lück. *Axiomatizations for Propositional and Modal Team Logic*. 25th EACSL Annual Conf. on Computer Science Logic, CSL 2016, August 29 - September 1, 2016, Marseille, France. Ed.: J. Talbot and L. Regnier. **62**. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016, 33:1–33:18.

[100] Martin Lück. *Axiomatizations of team logics*. Annals of Pure and Applied Logic **169**(9), 2018, pp. 928–969.

[101] Martin Lück. *Canonical Models and the Complexity of Modal Team Logic*. 27th EACSL Annual Conf. on Computer Science Logic, CSL 2018, September 4-7, 2018, Birmingham, UK. Ed.: D. Ghica and A. Jung. **119**. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018, 30:1–30:23.

[102] Martin Lück. *Canonical Models and the Complexity of Modal Team Logic*. Logical Methods in Computer Science **15**(2), 2019, 2:1–2:45.

[103] Martin Lück. *On the Complexity of Linear Temporal Logic with Team Semantics*. Submitted.

[104] Martin Lück. *On the Complexity of Team Logic and Its Two-Variable Fragment*. 43rd Int. Symp. on Mathematical Foundations of Computer Science, MFCS 2018, August 27-31, 2018, Liverpool, UK. Ed.: I. Potapov, P. Spirakis and J. Worrell. **117**. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018, 27:1–27:22.

[105] Martin Lück. *The Power of the Filtration Technique for Modal Logics with Team Semantics*. 26th EACSL Annual Conf. on Computer Science Logic, CSL 2017, August 20-24, 2017, Stockholm, Sweden. Ed.: V. Goranko and M. Dam. **82**. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017, 31:1–31:20.

[106] Martin Lück and Arne Meier. *LTL Fragments are Hard for Standard Parameterisations*. 22nd Int. Symp. on Temporal Representation and Reasoning, TIME 2015. Ed.: F. Grandi, M. Lange and A. Lomuscio. IEEE Computer Society, 2015, pp. 59–68.

[107] Martin Lück, Arne Meier and Irena Schindler. *Parametrised Complexity of Satisfiability in Temporal Logic*. ACM Transactions on Computational Logic **18**(1), 2017, 1:1–1:32.

[108] Martin Lück and Miikka Vilander. *On the Succinctness of Atoms of Dependency*. Logical Methods in Computer Science **15**(3), 2019, 17:1–17:28.

[109] Jan Łukasiewicz. *Elements of Mathematical Logic*. 1958. Engl. trans. by O. Wojtasiewicz. Pergamon Press, 1963.

[110] Yasir Mahmood and Arne Meier. *Parametrised Complexity of Model Checking and Satisfiability in Propositional Dependence Logic*. CoRR **abs/1904.06107**, 2019.

[111] Allen L. Mann. *Independence-friendly cylindric set algebras*. Logic Journal of the IGPL **17**(6), 2009, pp. 719–754.

[112] Maarten Marx and Yde Venema. *Multi-dimensional modal logic*. Vol. 4. Applied logic series. Dordrecht: Springer, 1997.

[113] Arne Meier and Christian Reinbold. *Enumeration Complexity of Poor Man's Propositional Dependence Logic*. Foundations of Information and Knowledge Systems - 10th Int. Symp., FoIKS 2018, Budapest, Hungary, May 14-18, 2018, Proc. Ed.: F. Ferrarotti and S. Woltran. **10833**. Lecture Notes in Computer Science. Springer, 2018, pp. 303–321.

[114] Albert R. Meyer. *The inherent computational complexity of theories of ordered sets*. Proc. Int. Congress of Mathematicians. **19722**. 1974, p. 481.

[115] Albert R. Meyer. *Weak monadic second order theory of succesor is not elementary-recursive*. Logic Colloquium. Ed.: R. Parikh. Berlin, Heidelberg: Springer, 1975, pp. 132–154.

[116] Michael Mortimer. *On languages with two variables*. Mathematical Logic Quarterly **21**(1), 1975, pp. 135–140.

[117] Lawrence S. Moss. *Finite Models Constructed from Canonical Formulars*. Journal of Philosophical Logic **36**(6), 2007, pp. 605–640.

[118] Till Mossakowski and Lutz Schröder. *On Inconsistency and Unsatisfiability*. Int. Journal of Software and Informatics **9**(2), 2015, pp. 141–152.

[119] Julian-Steffen Müller. *Satisfiability and model checking in team based logics*. Dissertation. Leibniz Universität Hannover. Göttingen: Cuvillier, 2014.

[120] Francis Jeffry Pelletier. *Did Frege Believe Frege's Principle?* Journal of Logic, Language and Information **10**(1), 2001, pp. 87–114.

[121] Michael O Rabin. *Decidability of second-order theories and automata on infinite trees*. Transactions of the American Mathematical Society **141**, 1969, pp. 1–35.

[122] Frank P. Ramsey. *On a Problem of Formal Logic*. In: Classic Papers in Combinatorics. Ed.: I. Gessel and G. Rota. Boston, MA: Birkhäuser, 1987, pp. 1–24.

[123] Raine Rönnholm. *Arity Fragments of Logics with Team Semantics*. Dissertation. Tampere University Press, 2018.

[124] Katsuhiko Sano and Jonni Virtema. *Axiomatizing Propositional Dependence Logics*. 24th EACSL Annual Conference on Computer Science Logic, CSL 2015, September 7-10, 2015, Berlin, Germany. Ed.: S. Kreutzer. **41**. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2015, pp. 292–307.

[125] Sylvain Schmitz. *Complexity Hierarchies beyond Elementary*. ACM Transactions on Computation Theory **8**(1), 2016, 3:1–3:36.

[126] Dana Scott. *A decision method for validity of sentences in two variables*. Journal of Symbolic Logic **27**(4), 1962, p. 477.

[127] Krister Segerberg. *Decidability of S4.1*. Theoria **34**(1), 1968, pp. 7–20.

[128] Merlijn Sevenster. *Model-theoretic and Computational Properties of Modal Dependence Logic*. Journal of Logic and Computation **19**(6), 2009, pp. 1157–1173.

[129] Theodore Sider. *Logic for philosophy*. Vol. 304. Oxford University Press, 2010.

[130] Michael Sipser. *Introduction to the Theory of Computation*. 3rd ed. Cengage learning, 2012.

[131] Thoralf Skolem. *Über die Nicht-charakterisierbarkeit der Zahlenreihe mittels endlich oder abzählbar unendlich vieler Aussagen mit ausschliesslich Zahlenvariablen*. Fundamenta mathematicae **23**(1), 1934, pp. 150–161.

[132] Larry J. Stockmeyer and Albert R. Meyer. *Word Problems Requiring Exponential Time: Preliminary Report*. Proc. of the 5th Annual ACM Symp. on Theory of Computing. 1973, pp. 1–9.

[133] Thomas Sturm, Marco Voigt and Christoph Weidenbach. *Deciding First-Order Satisfiability when Universal and Existential Variables are Separated*. Proc. of the 31st Annual ACM/IEEE Symp. on Logic in Computer Science, LICS '16, New York, NY, USA, July 5-8, 2016. Ed.: M. Grohe, E. Koskinen and N. Shankar. ACM, 2016, pp. 86–95.

[134] Alan M. Turing. *On Computable Numbers, with an Application to the Entscheidungsproblem*. Proc. of the London Mathematical Society **s2-42**(1), Jan. 1937, pp. 230–265.

[135] Jouko Väänänen. *Dependence logic: A New Approach to Independence Friendly Logic*. London Mathematical Society student texts 70. Cambridge University Press, 2007.

[136] Jouko Väänänen. *Modal dependence logic*. New perspectives on games and interaction **4**, 2008, pp. 237–254.

[137] Moshe Y. Vardi. *On the Complexity of Bounded-Variable Queries*. PODS. ACM Press, 1995, pp. 266–276.

[138] Yde Venema. *6 Algebras and coalgebras*. In: Handbook of Modal Logic. Ed.: P. Blackburn, J. van Benthem and F. Wolter. **3**. Studies in Logic and Practical Reasoning. Elsevier Science, 2007, pp. 331–426.

[139] Jonni Virtema. *Complexity of validity for propositional dependence logics*. Information and Computation **253**, 2017, pp. 224–236.

[140] Marco Voigt. *A fine-grained hierarchy of hard problems in the separated fragment*. 32nd Annual ACM/IEEE Symp. on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017. IEEE Computer Society, 2017, pp. 1–12.

[141] Fan Yang. *Modal dependence logics: Axiomatizations and model-theoretic properties*. Logic Journal of the IGPL **25**(5), 2017, pp. 773–805.

[142] Fan Yang. *On extensions and variants of dependence logic*. Dissertation. University of Helsinki, 2014.

[143] Fan Yang and Jouko Väänänen. *Propositional logics of dependence*. Annals of Pure and Applied Logic **167**(7), 2016, pp. 557–589.

[144] Fan Yang and Jouko Väänänen. *Propositional team logics*. Annals of Pure and Applied Logic **168**(7), 2017, pp. 1406–1441.

Appendix

The appendix contains several technical or standard proofs omitted from the previous chapters of this thesis.

A Proof details for Chapter 4

Proposition 4.6. *Let* $\Phi \subseteq \mathsf{Prop}$ *be finite and* $k \geqslant 0$*.*

(1) $[\![w]\!]^\Phi_k \cap \Phi = V^{-1}(w) \cap \Phi$ *and* $[\![Rw]\!]^\Phi_k = \mathcal{R}[\![w]\!]^\Phi_{k+1}$*, for all pointed structures* (W, R, V, w)*.*

(2) *The mapping* $h\colon \tau \mapsto \tau \cap \Phi$ *is a bijection from* Δ^Φ_0 *to* $\wp\Phi$*.*

(3) *The mapping* $h\colon \tau \mapsto (\tau \cap \Phi, \mathcal{R}\tau)$ *is a bijection from* Δ^Φ_{k+1} *to* $\wp\Phi \times \wp\Delta^\Phi_k$*.*

- *Proof of (1).* Assume (W, R, V, v), $\Phi \subseteq \mathsf{Prop}$ and $k \geqslant 0$ as above. For all $p \in \Phi$, clearly $p \in [\![w]\!]^\Phi_k$ iff $w \vDash p$ iff $p \in V^{-1}(w)$. Next, we show that $[\![Rw]\!]^\Phi_k = \mathcal{R}[\![w]\!]^\Phi_{k+1}$. Let $\tau = [\![w]\!]^\Phi_{k+1}$, and recall that $\mathcal{R}\tau = \{\tau' \in \Delta^\Phi_k \mid \{\alpha \mid \Box\alpha \in \tau\} \subseteq \tau'\}$. To prove $[\![Rw]\!]^\Phi_k \subseteq \mathcal{R}\tau$, let $\tau' \in [\![Rw]\!]^\Phi_k$ be arbitrary. Then $[\![v]\!]^\Phi_k = \tau'$ for some $v \in Rw$. Now, for all $\alpha \in \mathsf{ML}^\Phi_k$, $\Box\alpha \in \tau$ implies $w \vDash \Box\alpha$. In particular, $v \vDash \alpha$, i.e., $\alpha \in \tau'$. Hence, $\{\alpha \mid \Box\alpha \in \tau\} \subseteq \tau'$, which implies $\tau' \in \mathcal{R}\tau$.

  For the converse direction, $\mathcal{R}\tau \subseteq [\![Rw]\!]^\Phi_k$, let $\tau' \in \mathcal{R}\tau$ be arbitrary. By definition, $\{\alpha \mid \Box\alpha \in \tau\} \subseteq \tau'$. Since τ' is a k-type, it has a model $(\mathcal{K}', v')$, and due to Proposition 4.5, $[\![\mathcal{K}', v']\!]^\Phi_k = \tau'$. By Proposition 2.26, there is a formula $\zeta \in \mathsf{ML}^\Phi_k$ such that $(\mathcal{K}'', v'') \vDash \zeta$ if and only if $(\mathcal{K}', v') \rightleftharpoons^\Phi_k (\mathcal{K}'', v'')$. As τ is a $(k+1)$-type, either $\Diamond\zeta \in \tau$ or $\neg\Diamond\zeta \in \tau$.

  First, suppose $\neg\Diamond\zeta \in \tau$. Then $\Box\neg\zeta \in \tau$, hence $\neg\zeta \in \tau'$ by definition of τ'. But as $(\mathcal{K}', v') \vDash \tau'$, then both $(\mathcal{K}', v') \nvDash \zeta$ and $(\mathcal{K}', v') \vDash \zeta$, as $(\mathcal{K}', v') \rightleftharpoons^\Phi_k (\mathcal{K}', v')$. Contradiction, therefore $\Diamond\zeta \in \tau$. Consequently, w has an R-successor v such that $v \vDash \zeta$, i.e., $\tau' = [\![v]\!]^\Phi_k \in [\![Rw]\!]^\Phi_k$.

- *Proof that* h *in (2) and (3) is injective.* Let $\tau, \tau' \in \Delta^\Phi_k$ be arbitrary. Let $(\mathcal{K}, w) = (W, R, V, w)$ be of type τ, and $(\mathcal{K}', w') = (W', R', V', w')$ of type τ'. We first consider (2) and demonstrate that $h\colon \tau \mapsto \tau \cap \Phi$ injective. This follows from (1), as $\tau \cap \Phi = \tau' \cap \Phi$ implies $V^{-1}(w) = \tau \cap \Phi = \tau' \cap \Phi = V'^{-1}(w')$, i.e., $(\mathcal{K}, w) \rightleftharpoons^\Phi_0 (\mathcal{K}', w')$. By Proposition 4.5, then $\tau = \tau'$.

  For (3), let $k > 0$, and additionally suppose $\mathcal{R}\tau = \mathcal{R}\tau'$. Again by (1), we have $[\![\mathcal{K}, Rw]\!]^\Phi_{k-1} = \mathcal{R}\tau = \mathcal{R}\tau' = [\![\mathcal{K}', R'w']\!]^\Phi_{k-1}$. By Proposition 4.5, $(\mathcal{K}, Rw) \rightleftharpoons^\Phi_{k-1}$

$(\mathcal{K}', R'w')$ follows. Since $(\mathcal{K}, w) \rightleftharpoons_0^\Phi (\mathcal{K}', w')$ holds as before, $(\mathcal{K}, w) \rightleftharpoons_k^\Phi (\mathcal{K}', w')$ by Proposition 2.28. By Proposition 4.5, $\tau = [\![\mathcal{K}, w]\!]_k^\Phi = [\![\mathcal{K}', w']\!]_k^\Phi = \tau'$.

- *Proof that* h *in* (2) *and* (3) *is surjective.* First, consider (2). We have to show that, for all $\Phi' \subseteq \Phi$, there exists a type $\tau \in \Delta_0^\Phi$ such that $\tau \cap \Phi = \Phi'$. Likewise, for (3) we have to show that for all $k \geqslant 0$, $\Phi' \subseteq \Phi$ and $\Delta' \subseteq \Delta_k^\Phi$, there exists a type $\tau \in \Delta_{k+1}^\Phi$ such that $\tau \cap \Phi = \Phi'$ and $\mathcal{R}\tau = \Delta'$. We show the second statement, as the first one is then analogous. The following model $(\mathcal{K}, w) = (W, R, V, w)$ witnesses that there exists $\tau \in \Delta_{k+1}$ such that $\tau \cap \Phi = \Phi'$ and $\mathcal{R}\tau = \Delta'$. First, recall that each $\tau' \in \Delta'$ has a model $(\mathcal{N}_{\tau'}, \nu_{\tau'})$ such that, by Proposition 4.5, $[\![\mathcal{N}_{\tau'}, \nu_{\tau'}]\!]_k^\Phi = \tau'$. W.l.o.g. all $\mathcal{N}_{\tau'}$ are pairwise disjoint. Define $\mathcal{K}$ as the disjoint union of the models $\mathcal{N}_\tau$ and of a distinct point w, and let $V^{-1}(w) = \Phi'$. By (1), then $[\![w]\!]_{k+1}^\Phi \cap \Phi = \Phi'$. Moreover, let $Rw = \{\nu_{\tau'} \mid \tau' \in \Delta'\}$. Again due to (1), $\mathcal{R}[\![w]\!]_{k+1}^\Phi = [\![Rw]\!]_k^\Phi$. By definition, $[\![Rw]\!]_k^\Phi = [\![\{\nu_{\tau'} \mid \tau' \in \Delta'\}]\!]_k^\Phi = \{[\![\nu_{\tau'}]\!]_k^\Phi \mid \tau' \in \Delta'\} = \Delta'$. □

Proposition 4.7. *Let* (W, R, V, w) *be a pointed structure,* $\Phi \subseteq \mathsf{Prop}$ *finite and* $k \geqslant 0$.

(1) *If* $\tau \in \Delta_0^\Phi$, *then* $[\![w]\!]_0^\Phi = \tau$ *if and only if* $V^{-1}(w) = \tau \cap \Phi$.

(2) *If* $\tau \in \Delta_{k+1}^\Phi$, *then* $[\![w]\!]_{k+1}^\Phi = \tau$ *if and only if* $V^{-1}(w) = \tau \cap \Phi$ *and* $[\![Rw]\!]_k^\Phi = \mathcal{R}\tau$.

Proof. The direction "$\Rightarrow$" of both (1) and (2) follows directly from Proposition 4.6. Moreover, we prove "$\Leftarrow$" only for statement (2), as the proof is analogous for (1).

Suppose that there are $\tau, \tau' \in \Delta_{k+1}^\Phi$ such that $V^{-1}(w) = \tau \cap \Phi$ and $[\![Rw]\!]_k^\Phi = \mathcal{R}\tau$, but $[\![w]\!]_{k+1}^\Phi = \tau'$. Then, by "$\Rightarrow$", we have $V^{-1}(w) = \tau' \cap \Phi$ and $[\![Rw]\!]_k^\Phi = \mathcal{R}\tau'$ as well. In other words, $\tau \cap \Phi = \tau' \cap \Phi$ and $\mathcal{R}\tau = \mathcal{R}\tau'$. However, since the mapping $h\colon \tau \mapsto (\tau \cap \Phi, \mathcal{R}\tau)$ is bijective according to Proposition 4.6, we have $\tau = \tau' = [\![w]\!]_{k+1}^\Phi$. □

Lemma 4.12. *For every polynomial* p *there is a polynomial* q *such that*

$$p(\exp_k^*(n)) \leqslant \exp_k(q((k+1)\cdot n))$$

for all $k \geqslant 0$ *and* $n \geqslant 1$.

We require the following inequalities.

Lemma A.1. *Let* $n, k, c \geqslant 0$. *Then* $c + \exp_k(n) \leqslant \exp_k(c+n)$. *If also* $n \geqslant 1$, *then* $c \cdot \exp_k(n) \leqslant \exp_k(cn)$.

Proof. Induction on k, where $k = 0$ is trivial. For $k \geqslant 1$,

$$\begin{aligned} c + \exp_{k+1}(n) &= c + 2^{\exp_k(n)} \leqslant 2^c \cdot 2^{\exp_k(n)} && \text{(as } c + a \leqslant 2^c \cdot a \text{ for } c \geqslant 0, a \geqslant 1\text{)} \\ &= 2^{c+\exp_k(n)} \leqslant 2^{\exp_k(c+n)} && \text{(induction hypothesis)} \\ &= \exp_{k+1}(c+n). \end{aligned}$$

For the product, the cases $c = 0, 1$ are trivial. For $c \geqslant 2$,

$$\begin{aligned} c \cdot \exp_{k+1}(n) &\leqslant 2^{c-1} \cdot 2^{\exp_k(n)} && \text{(since } c \geqslant 2 \text{ implies } c \leqslant 2^{c-1}) \\ &= 2^{c-1+\exp_k(n)} \leqslant 2^{\exp_k(c-1+n)} && \text{(by + case)} \\ &\leqslant 2^{\exp_k(cn)} = \exp_{k+1}(cn). && \text{(as } (c-1)+n \leqslant cn \text{ for } c, n \geqslant 1) \quad \square \end{aligned}$$

Recall that $\exp^*_0(n) := n$ and $\exp^*_{k+1}(n) := n \cdot 2^{\exp^*_k(n)}$.

Lemma A.2. *Let* $n, k \geqslant 0$. *Then* $\exp^*_k(n) \leqslant \exp_k((k+1) \cdot n)$.

Proof. Induction on k. For $k = 0$, $\exp^*_0(n) = n = \exp_0((0+1) \cdot n)$. For the inductive step,

$$\begin{aligned} \exp^*_{k+1}(n) &= n \cdot 2^{\exp^*_k(n)} \leqslant 2^n \cdot 2^{\exp^*_k(n)} = 2^{n+\exp^*_k(n)} \\ &\leqslant 2^{n+\exp_k((k+1)n)} && \text{(induction hypothesis)} \\ &\leqslant 2^{\exp_k(n+(k+1)n)} = \exp_{k+1}((k+2)n) && \text{(Lemma A.1)} \quad \square \end{aligned}$$

The next inequality states that a polynomial can be "pulled inside" $\exp_k$:

Lemma A.3. *For every polynomial* p *there is a polynomial* q *such that*

$$p(\exp_k(n)) \leqslant \exp_k(q(n)))$$

for all $k \geqslant 0, n \geqslant 1$.

Proof. For every polynomial p there are integers $c, d \geqslant 1$ such that $p(n) \leqslant cn^d$ for all $n \geqslant 1$. Let $q(n) := cdn^d + c$. Then the case $k = 0$ is clear. For $k \geqslant 1$ and $n \geqslant 1$,

$$\begin{aligned} p(\exp_k(n)) &\leqslant c \cdot \exp_k(n)^d \leqslant 2^c \cdot (2^{\exp_{k-1}(n)})^d = 2^{c+d\cdot\exp_{k-1}(n)} \\ &\leqslant 2^{q(\exp_{k-1}(n))} && \text{(as } q(n) \geqslant c + dn) \\ &\leqslant 2^{\exp_{k-1}(q(n))} = \exp_k(q(n)). && \text{(Lemma A.1)} \quad \square \end{aligned}$$

Finally, we combine both lemmas:

Proof of Lemma 4.12. Let p be a polynomial as above. W.l.o.g. p is non-decreasing. Then by Lemma A.2, $p(\exp^*_k(n)) \leqslant p\big(\exp_k((k+1) \cdot n)\big)$. Moreover, due to Lemma A.3, there is a polynomial q such that $p\big(\exp_k((k+1) \cdot n)\big) \leqslant \exp_k\big(q((k+1) \cdot n)\big)$. □

Proposition 4.17. *Let* α, β *be disjoint scopes and* S, U, T *teams in a Kripke structure* $\mathcal{K} = (W, R, V)$. *Then the following laws hold:*

(1) *Distributive laws:* $(T \cap S)_\alpha = T_\alpha \cap S = T \cap S_\alpha = T_\alpha \cap S_\alpha$ *and* $(T \cup S)_\alpha = T_\alpha \cup S_\alpha$.

(2) *Disjoint selection commutes:* $\big(T^\alpha_S\big)^\beta_U = \big(T^\beta_U\big)^\alpha_S$.

(3) *Disjoint selection is independent:* $\big((T^\alpha_S)^\beta_U\big)_\alpha = T_\alpha \cap S$.

(4) *Image and selection commute:* $(RT)_\alpha = \big(R(T_\alpha)\big)_\alpha = R(T_\alpha)$

(5) *Successor and selection commute: If* S *is a strict resp. lax successor team of* T*, then* S_α *is a strict resp. lax successor team of* T_α*.*

(6) *Selection propagates: If* $S \subseteq T$*, then* $R(T^\alpha_S) = (RT)^\alpha_{RS}$.

Proof. (1) Observe that $X_\alpha = X \cap W_\alpha$. Hence, for the union $(T \cup S)_\alpha = (T \cup S) \cap W_\alpha = (T \cap W_\alpha) \cup (S \cap W_\alpha) = T_\alpha \cup S_\alpha$ holds. For the intersection, likewise $(T \cap S) \cap W_\alpha = (T \cap W_\alpha) \cap S = T \cap (W_\alpha \cap S) = (T \cap W_\alpha) \cap (S \cap W_\alpha)$.

(2) Proved in the following equation. We use the fact that $X_{\gamma \wedge \gamma'} = (X_\gamma)_{\gamma'} = (X_{\gamma'})_\gamma = X_{\gamma' \wedge \gamma}$ for all teams X and scopes γ, γ.

$$\begin{aligned} &\left(T^\alpha_S\right)^\beta_U \\ &= \left(T_{\neg\alpha} \cup (T_\alpha \cap S)\right)_{\neg\beta} \cup \Big(\left(T_{\neg\alpha} \cup (T_\alpha \cap S)\right)_\beta \cap U\Big) \end{aligned}$$

Distributing all scopes according to (1):

$$= T_{\neg\alpha\wedge\neg\beta} \cup \left(T_{\alpha\wedge\neg\beta} \cap S_{\neg\beta}\right) \cup \left(T_{\neg\alpha\wedge\beta} \cap U\right) \cup \left(T_{\alpha\wedge\beta} \cap S_\beta \cap U\right)$$

Replace U by $U_{\neg\alpha}/U_\alpha$ due to the intersection law of (1):

$$= T_{\neg\alpha\wedge\neg\beta} \cup \left(T_{\alpha\wedge\neg\beta} \cap S_{\neg\beta}\right) \cup \left(T_{\neg\alpha\wedge\beta} \cap U_{\neg\alpha}\right) \cup \left(T_{\alpha\wedge\beta} \cap S_\beta \cap U_\alpha\right)$$

Likewise, replace $S_{\neg\beta}/S_\beta$ by S:

$$\begin{aligned} &= T_{\neg\alpha\wedge\neg\beta} \cup \left(T_{\alpha\wedge\neg\beta} \cap S\right) \cup \left(T_{\neg\alpha\wedge\beta} \cap U_{\neg\alpha}\right) \cup \left(T_{\alpha\wedge\beta} \cap S \cap U_\alpha\right) \\ &= T_{\neg\beta\wedge\neg\alpha} \cup \left(T_{\beta\wedge\neg\alpha} \cap U_{\neg\alpha}\right) \cup \left(T_{\neg\beta\wedge\alpha} \cap S\right) \cup \left(T_{\beta\wedge\alpha} \cap U_\alpha \cap S\right) \end{aligned}$$

Reverse distribution of scopes:

$$\begin{aligned} &= \left(T_{\neg\beta} \cup (T_\beta \cap U)\right)_{\neg\alpha} \cup \Big(\left(T_{\neg\beta} \cup (T_\beta \cap U)\right)_\alpha \cap S\Big) \\ &= \left(T^\beta_U\right)^\alpha_S. \end{aligned}$$

(3) By definition and application of (2), $\left((T^\alpha_S)^\beta_U\right)_\alpha$ equals

$$\begin{aligned} &\Big[\left(T_{\neg\beta} \cup (T_\beta \cap U)\right)_{\neg\alpha} \cup \Big(\left(T_{\neg\beta} \cup (T_\beta \cap U)\right)_\alpha \cap S\Big)\Big]_\alpha \\ &= \left(T_{\neg\beta} \cup (T_\beta \cap U)\right)_{\neg\alpha\wedge\alpha} \cup \Big(\left(T_{\neg\beta} \cup (T_\beta \cap U)\right)_\alpha \cap S\Big)_\alpha \\ &= \emptyset \cup \Big(\left(T_{\neg\beta} \cup (T_\beta \cap U)\right)_\alpha \cap S_\alpha\Big) \\ &= \left(T_{\neg\beta\wedge\alpha} \cap S_\alpha\right) \cup \left(T_{\beta\wedge\alpha} \cap U_\alpha \cap S_\alpha\right) \end{aligned}$$

Since α and β are disjoint:

$$= \left(T_\alpha \cap S_\alpha\right) \cup \left(\emptyset \cap U_\alpha \cap S_\alpha\right) = T_\alpha \cap S.$$

(4) $(RT)_\alpha \subseteq \big(R(T_\alpha)\big)_\alpha$: Suppose $v \in (RT)_\alpha$. Then $v \in Rw$ for some $w \in T$. Moreover, $w \in T_\alpha$, since α is a scope. Hence $v \in R(T_\alpha)$. As $v \vDash \alpha$, $v \in \big(R(T_\alpha)\big)_\alpha$ follows.

$\big(R(T_\alpha)\big)_\alpha \subseteq R(T_\alpha)$: Obvious.

$R(T_\alpha) \subseteq (RT)_\alpha$: Again, let $v \in R(T_\alpha)$ be arbitrary. Then $v \in Rw$ for some $w \in T_\alpha$. Hence $v \in RT$. Since $v \vDash \alpha$ follows from $w \vDash \alpha$, we conclude $v \in (RT)_\alpha$.

(5) If $w \in T_\alpha$ and S is a lax successor team of T, then $Rw \cap S \neq \emptyset$. Scopes are closed under R, so $Rw = Rw_\alpha$. Since $Rw_\alpha \cap S = (Rw \cap S)_\alpha \subseteq Rw \cap S_\alpha$ by (1), w has a successor in S_α. Conversely, since $S \subseteq RT$ we have $S_\alpha \subseteq (RT)_\alpha = R(T_\alpha)$ by (4), so every $v \in S_\alpha$ has a predecessor in T_α. Hence S_α is a lax successor team of T_α.

If S is a strict successor team of T, then $S = \{f(w) \mid w \in T\}$ for some $f \in \prod_{w \in T} Rw$. Consider the function $f' := f{\upharpoonright}T_\alpha \in \prod_{w \in T_\alpha} Rw$. As for all $w \in T_\alpha$ we have that $Rw = (Rw)_\alpha$ and $f(w) \in S$, it holds that $\{f'(w) \mid w \in T_\alpha\} \subseteq S_\alpha$. Likewise, if $v \in S_\alpha$, then $v = f(w)$ for some $w \in T$. But as α is a scope, then $w \in T_\alpha$, and so $v = f'(w)$. This shows $\{f'(w) \mid w \in T_\alpha\} = S_\alpha$. Consequently, S_α is a strict successor team of T_α.

(6) For "$\subseteq$", suppose $v \in R(T^\alpha_S)$, i.e., $v \in Rw$ for some $w \in T^\alpha_S$. In particular, $v \in RT$. If $w \nvDash \alpha$, then $v \in RT_{\neg\alpha}$ and trivially $v \in (RT)^\alpha_{RS}$. If $w \vDash \alpha$, then necessarily $w \in S$. Moreover, $v \vDash \alpha$. Consequently, $v \in RS_\alpha \cap RT_\alpha$, hence $v \in (RT)^\alpha_{RS}$.

For "$\supseteq$", suppose $v \in (RT)^\alpha_{RS} = RT_{\neg\alpha} \cup (RT_\alpha \cap RS)$.

If $v \in RT_{\neg\alpha}$, then by (4) $v \in Rw$ for some $w \in T_{\neg\alpha}$. In particular, $w \in T^\alpha_S$, hence $v \in R\big(T^\alpha_S\big)$.

If $v \in RT_\alpha \cap RS$, then by (1) $v \in RS_\alpha$. By (4) $v \in R(S_\alpha)$, in other words, $v \in Rw$ for some $w \in S_\alpha$. As $S \subseteq T$, then $w \in S_\alpha \cap T$, and in fact $w \in T_\alpha \cap S$ due to (1). Consequently, $w \in T^\alpha_S$ and $v \in R(T^\alpha_S)$. ☐

Lemma 4.27. *Let* $\alpha, \beta \in \mathsf{ML}$ *and* $\varphi \in \mathsf{ML}_k(\sim)$. *Let* T *be a team such that* $R^i T \vDash \alpha \leftrightarrow \beta$ *for all* $i \in \{0, \ldots, k\}$. *Then* $T \vDash \varphi$ *if and only if* $T \vDash \varphi[\alpha/\beta]$.

Proof. Proof by induction on k and the syntax on φ. W.l.o.g. α occurs in φ. If $\varphi = \alpha$, then $\varphi[\alpha/\beta] = \beta$, in which case the proof comes down to showing $T \vDash \alpha \Leftrightarrow T \vDash \beta$. However, this easily follows from $T \vDash \alpha \leftrightarrow \beta$ by the semantics for classical ML-formulas.

Otherwise, α is a proper subformula of φ. We distinguish the following cases.

- $\varphi = \neg\gamma$: Then $(\neg\gamma)[\alpha/\beta] = \neg\gamma[\alpha/\beta]$, and

$$\begin{aligned}
& T \vDash \varphi[\alpha/\beta] \\
\Leftrightarrow\ & T \vDash \neg\gamma[\alpha/\beta] \\
\Leftrightarrow\ & \forall w \in T\colon \{w\} \vDash \neg\gamma[\alpha/\beta] \\
\Leftrightarrow\ & \forall w \in T\colon \{w\} \vDash \neg\gamma \qquad \text{(induction hypothesis, as } \{w\}, Rw, \ldots \vDash \alpha \leftrightarrow \beta) \\
\Leftrightarrow\ & T \vDash \neg\gamma \\
\Leftrightarrow\ & T \vDash \varphi
\end{aligned}$$

- $\varphi = {\sim}\psi$ resp. $\varphi = \psi \wedge \theta$: Obvious by induction hypothesis.
- $\varphi = \psi \vee \theta$: First note that $(\psi \vee \theta)[\alpha/\beta] = \psi[\alpha/\beta] \vee \theta[\alpha/\beta]$. Then:

$$\begin{aligned} & T \vDash \varphi[\alpha/\beta] \\ \Leftrightarrow\ & T \vDash \psi[\alpha/\beta] \vee \theta[\alpha/\beta] \\ \Leftrightarrow\ & \exists S, U \colon T = S \cup U, S \vDash \psi[\alpha/\beta], U \vDash \theta[\alpha/\beta] \end{aligned}$$

  By downward closure, $S, U, RS, RU, \ldots \vDash \alpha \leftrightarrow \beta$, so by induction hypothesis:

$$\begin{aligned} \Leftrightarrow\ & \exists S, U \colon T = S \cup U, S \vDash \psi, U \vDash \theta \\ \Leftrightarrow\ & T \vDash \varphi \end{aligned}$$

- $\varphi = \Box\psi$: We have $(\Box\psi)[\alpha/\beta] = \Box\psi[\alpha/\beta]$, hence

$$\begin{aligned} & T \vDash \varphi[\alpha/\beta] \\ \Leftrightarrow\ & T \vDash \Box\psi[\alpha/\beta] \\ \Leftrightarrow\ & RT \vDash \psi[\alpha/\beta]. \end{aligned}$$

  However, since $\psi \in \mathsf{ML}_{k-1}(\sim)$ and $RT, \ldots, R^{k-1}RT \vDash \alpha \leftrightarrow \beta$ holds by assumption, we obtain by induction hypothesis:

$$\begin{aligned} \Leftrightarrow\ & RT \vDash \psi \\ \Leftrightarrow\ & T \vDash \varphi \end{aligned}$$

- $\varphi = \Diamond\psi$: As before, $(\Diamond\psi)[\alpha/\beta] = \Diamond\psi[\alpha/\beta]$. Then:

$$\begin{aligned} & T \vDash \varphi[\alpha/\beta] \\ \Leftrightarrow\ & T \vDash \Diamond\psi[\alpha/\beta] \\ \Leftrightarrow\ & \exists S \subseteq RT, T \subseteq R^{-1}S \colon S \vDash \psi[\alpha/\beta] \end{aligned}$$

  Note that $S, RS, \ldots, R^{k-1}S$ are subteams of $RT, \ldots, R^kT$, respectively. For this reason, the teams $S, RS, \ldots, R^{k-1}S$ satisfy $\alpha \leftrightarrow \beta$ as well. As also $\psi \in \mathsf{ML}_{k-1}(\sim)$ holds, we obtain by induction hypothesis:

$$\begin{aligned} \Leftrightarrow\ & \exists S \subseteq RT, T \subseteq R^{-1}S \colon S \vDash \psi \\ \Leftrightarrow\ & T \vDash \varphi \end{aligned}$$

□

B Proof details for Chapter 6

Recall that a set Φ is Ω-*inconsistent* if from it we can derive all formulas in Ω.

Lemma 6.15. *Let $\Omega \succeq \mathsf{L}$. The following statements are equivalent:*

(1) $\Phi \vdash_\Omega \varphi$ *and* $\Phi \vdash_\Omega {\sim}\varphi$ *for some* φ,

(2) Φ *is* Ω*-inconsistent,*

(3) $\Phi \vdash_\Omega \perp\!\!\!\perp$.

Proof. For (1) to (2), we have to show $\Phi \vdash \xi$ for arbitrary ξ. First, $\Phi \vdash ({\sim}\xi \rightharpoonup {\sim}\varphi)$ follows from $\Phi \vdash {\sim}\varphi$ by (L1) and (E$\rightharpoonup$). Next, $\Phi \vdash (\varphi \rightharpoonup \xi)$ follows by (L3) and (E$\rightharpoonup$), and finally $\Phi \vdash \xi$ by (E$\rightharpoonup$) again. (2) to (3) is obvious. For (3) to (1), we take any standard proof of propositional logic that derives $\top$, as $\perp\!\!\!\perp = {\sim}\top$. □

Lemma 6.16. *Let* $\Omega \succeq \mathsf{L}$ *and let* Φ *be consistent. Then* $\Phi \nvdash_\Omega \varphi$ *implies that* $\Phi \cup \{{\sim}\varphi\}$ *is* Ω*-consistent, and* $\Phi \vdash_\Omega \varphi$ *implies that* $\Phi \cup \{\varphi\}$ *is* Ω*-consistent.*

Proof. For the first part, suppose for the sake of contradiction that $\Phi \nvdash \varphi$, but $\Phi \cup \{{\sim}\varphi\}$ is inconsistent. Then $\Phi \cup \{{\sim}\varphi\} \vdash {\sim}\psi$ for any axiom ψ. As conservative extensions have the deduction theorem (Theorem 6.13), $\Phi \vdash ({\sim}\varphi \rightharpoonup {\sim}\psi)$. But by (L3), $\Phi \vdash \psi \rightharpoonup \varphi$, and ultimately $\Phi \vdash \varphi$, since ψ is an axiom. This contradicts $\Phi \nvdash \varphi$, so instead $\Phi \cup \{{\sim}\varphi\}$ must be consistent.

The second statement is proven similarly: Suppose that $\Phi \vdash \varphi$, but $\Phi \cup \{\varphi\}$ is inconsistent. Then $\Phi \cup \{\varphi\} \vdash \perp\!\!\!\perp$ by Lemma 6.15, and again by the deduction theorem, $\Phi \vdash \varphi \rightharpoonup \perp\!\!\!\perp$. As a result, $\Phi \vdash \perp\!\!\!\perp$, contradicting Lemma 6.15, since Φ is consistent. □

Recall that *maximal* means that either $\psi \in \Phi$ or ${\sim}\psi \in \Phi$ for every formula ψ.

Lemma 6.17 (Lindenbaum's lemma)**.** *If* $\Omega \succeq \mathsf{L}$*, then every* Ω*-consistent set has a maximal* Ω*-consistent superset.*

Proof. Let Φ be Ω-consistent, $\Omega = (\Xi, \psi, I)$, and $\Xi = \{\xi_1, \xi_2, \ldots\}$. Define $\Phi_0 := \Phi$, and for each $i \geqslant 1$,

$$\Phi_i := \begin{cases} \Phi_{i-1} \cup \{\xi_i\} & \text{if } \Phi_{i-1} \vdash \xi_i, \\ \Phi_{i-1} \cup \{{\sim}\xi_i\} & \text{otherwise.} \end{cases}$$

By Lemma 6.16, the Ω-consistency of Φ_{i-1} implies that of Φ_i. Consequently, Φ_i is Ω-consistent for all i, and hence $\Phi^* := \bigcup_{n \geqslant 0} \Phi_n$ is Ω-consistent as well. By construction, Φ^* is maximal Ω-consistent. □

Recall that refutation completeness means that every unsatisfiable set is also inconsistent, which is a strictly weaker property than completeness.

Theorem 6.19. *If* $\Omega \succeq \mathsf{L}$ *is refutation complete for* $\mathcal{F} \cup {\sim}\mathcal{F}$*, then it is complete for* $\mathcal{B}(\mathcal{F})$*.*

Proof. Let $\Phi \subseteq \mathcal{B}(\mathcal{F})$ and $\varphi \in \mathcal{B}(\mathcal{F})$. For completeness, we have to show that $\Phi \nvdash \varphi$ implies $\Phi \nvDash \varphi$. If $\Phi \nvdash \varphi$, then by Lemma 6.16, $\Phi \cup \{{\sim}\varphi\}$ is consistent. Then $\Phi \cup \{{\sim}\varphi\}$ has a maximal consistent superset Φ^* by Lindenbaum's lemma. Clearly, $\Phi^* \cap (\mathcal{F} \cup {\sim}\mathcal{F})$ is then consistent as well. By refutation completeness of Ω for $\mathcal{F} \cup {\sim}\mathcal{F}$, it has a model $\mathcal{A}$. We show that $\psi \in \Phi^* \Leftrightarrow A \vDash \psi$ for all $\psi \in \mathcal{B}(\mathcal{F})$. In particular, $\Phi \cup \{\sim\varphi\}$ is then satisfiable, which proves $\Phi \nvDash \varphi$.

The proof is by induction on ψ. In the simplest case, ψ is just an $\mathcal{F}$-formula. Then either $\psi \in \Phi^*$, and $A \vDash \psi$ by definition of A, or $\psi \notin \Phi^*$, but then $\sim\psi \in \Phi^*$ by maximality of Φ^*, and $A \nvDash \psi$ by definition of A.

For the induction step, let $\psi \notin \mathcal{F}$. The case $\psi = \sim\theta$ is clear as Φ^* is maximal consistent, which implies that $\sim\varphi \in \Phi^*$ iff $\varphi \notin \Phi^*$.

Next, suppose $\psi = \psi_1 \wedge \psi_2$. If $\psi \in \Phi^*$, then both $\sim\psi_1 \notin \Phi^*$ and $\sim\psi_2 \notin \Phi^*$, as otherwise Φ^* would be inconsistent by (L4) and (L5). Hence $\{\psi_1, \psi_2\} \subseteq \Phi^*$ by maximality of Φ^*, so $A \vDash \psi_1, \psi_2$ by induction hypothesis, and $A \vDash \psi$.

Conversely, if $\psi \notin \Phi^*$, then $\sim\psi \in \Phi^*$. By consistency, $\Phi^* \nvdash \psi$. For the sake of contradiction, suppose that $A \vDash \psi$, i.e., $A \vDash \psi_1, \psi_2$. By induction hypothesis, $\{\psi_1, \psi_2\} \subseteq \Phi^*$. But then $\Phi^* \vdash \psi_1 \rightarrowtriangle (\psi_2 \rightarrowtriangle \psi)$ via (L6), so $\Phi^* \vdash \psi$ by two applications of (E$\rightarrowtriangle$). But we showed $\Phi^* \nvdash \psi$, so $A \nvDash \psi$ must hold. □

C Proof details for Lemma 6.40 (Table 6.7)

In the proofs below, we sometimes implicitly apply (E$\rightarrowtriangle$), (MP$\vee$) and (MP$\multimap$) to replace subformulas of $\vee$ and $\multimap$ without stating the rule in the right column. We also tacitly apply (Com$\vee$) if we replace the first argument of $\vee$ instead of the second.

(Aug$\vee$)

A	$\varphi \vee \psi$	
B	$\varphi \multimap \theta$	
1	$\varphi \multimap \sim(\psi \wedge \theta)$	
2	$\varphi \multimap (\theta \rightarrowtriangle \sim\psi)$	L
3	$(\varphi \multimap \theta) \rightarrowtriangle (\varphi \multimap \sim\psi)$	(Dis$\multimap$)
4	$\varphi \multimap \sim\psi$	(E$\rightarrowtriangle$), B, 3
5	$\sim(\varphi \vee \sim\sim\psi)$	def.
6	$\varphi \vee \sim\sim\psi$	L, A
7	$\sim(\varphi \multimap \sim(\psi \wedge \theta))$	(RAA), 5, 6
▷	$\varphi \vee (\psi \wedge \theta)$	(Dual$\multimap$)

(I$\vee$):

A	$E\alpha$	
1	$\alpha \Leftrightarrow \neg\neg\alpha$	(thm), H^{PL}, (L7)
2	$\neg\alpha \multimap \sim\sim(\alpha \rightarrowtriangle \neg\alpha)$	
3	$\neg\alpha \multimap (\alpha \rightarrowtriangle \neg\alpha)$	L
4	$\neg\alpha \vee \alpha$	(thm), H^{PL}
5	$\neg\alpha \vee (\alpha \wedge (\alpha \rightarrowtriangle \neg\alpha))$	(Aug$\vee$)
6	$\neg\alpha \vee \neg\alpha$	L
7	$\neg\alpha$	H^{PL}
8	$\sim\neg\alpha$	def., A
9	$\sim(\neg\alpha \multimap \sim\sim(\alpha \rightarrowtriangle \neg\alpha))$	(RAA)
10	$\neg\alpha \vee \sim(\alpha \rightarrowtriangle \neg\alpha)$	(Dual$\multimap$)
11	$\top \vee (\alpha \wedge \sim\neg\alpha)$	H^{PL}, L
▷	$\top \vee (\alpha \wedge E\alpha)$	def.

(AbsE$\vee$):

A	$E\alpha \vee \varphi$	
1	$\neg\alpha$	
2	$\varphi \multimap \neg\alpha$	(Sub$\vee$)
3	$\sim\sim(\varphi \multimap \sim\sim\neg\alpha)$	L
4	$\sim(\varphi \vee \sim\neg\alpha)$	(Dual$\multimap$), L
5	$\sim\neg\alpha \vee \varphi$	def., A
6	$\varphi \vee \sim\neg\alpha$	(Com$\vee$)
7	$\sim\neg\alpha$	(RAA), 4, 6
▷	$E\alpha$	def.

(AbsE$\wedge$)

A	$\alpha \wedge E\beta$	
1	$\neg(\alpha \wedge \beta) \rightarrow (\alpha \rightarrow \neg\beta)$	H^{PL}
2	$\neg(\alpha \wedge \beta) \rightarrowtriangle (\alpha \rightarrow \neg\beta)$	(L7)
3	α	L, A
4	$\alpha \rightarrow \neg\beta$	
5	$\neg\beta$	H^{PL}, 3, 4
6	$E\beta$	L, A
7	$\sim\neg\beta$	def., 6
8	$\sim(\alpha \rightarrow \neg\beta)$	(RAA), 5, 7
9	$\sim\neg(\alpha \wedge \beta)$	L, 2, 8
▷	$E(\alpha \wedge \beta)$	def.

$(\mathrm{D}\wedge\vee)^1$:

A	$\alpha \wedge (\varphi \vee \psi)$	
1	α	L, A
2	$\varphi \multimap \alpha$	(Sub∨)
3	$\varphi \vee \psi$	L, A
4	$\varphi \vee (\psi \wedge \alpha)$	(Aug∨)
5	$(\psi \wedge \alpha) \multimap \alpha$	(Sub∨), 1
6	$(\psi \wedge \alpha) \vee \varphi$	(Com∨), 4
7	$(\psi \wedge \alpha) \vee (\varphi \wedge \alpha)$	(Aug∨)
▷	$(\alpha \wedge \varphi) \vee (\alpha \wedge \psi)$	L

$(\mathrm{D}\wedge\vee)^2$:

A	$(\alpha \wedge \varphi) \vee (\alpha \wedge \psi)$	
1	$(\alpha \wedge \varphi) \vee \alpha$	L
2	$\alpha \vee (\alpha \wedge \varphi)$	(Com∨)
3	$\alpha \vee \alpha$	L
4	α	H^{PL}
5	$(\alpha \wedge \varphi) \vee \psi$	L, A
6	$\psi \vee (\alpha \wedge \varphi)$	(Com∨)
7	$\psi \vee \varphi$	L
8	$\varphi \vee \psi$	(Com∨)
▷	$\alpha \wedge (\varphi \vee \psi)$	L, 4, 8

$(\mathrm{D}\vee\ovee)^1$:

A	$\varphi \vee (\psi \ovee \theta)$	
1 (hyp.)	$\sim((\varphi \vee \psi) \ovee (\varphi \vee \theta))$	
2	$\sim(\varphi \vee \psi)$	L, 1
3	$\sim(\varphi \vee \theta)$	L, 1
4	$\varphi \multimap \sim\psi$	(Dual⊸), L, 2
5	$\varphi \multimap \sim\theta$	(Dual⊸), L, 3
6	$\varphi \multimap \sim(\psi \ovee \theta)$	L, 4, 5
7	$\sim(\varphi \vee (\psi \ovee \theta))$	(Dual⊸), L
▷	$(\varphi \vee \psi) \ovee (\varphi \vee \theta)$	(RAA), A, 7

$(\mathrm{D}\vee\ovee)^2$:

A	$(\varphi \vee \psi) \ovee (\varphi \vee \theta)$	
1 (hyp.)	$\varphi \multimap \sim(\psi \ovee \theta)$	
2	$\varphi \multimap \sim\psi$	L, (MP⊸)
3	$\sim\sim(\varphi \multimap \sim\psi)$	L
4	$\sim(\varphi \vee \psi)$	(Dual⊸), L
5	$\varphi \vee \theta$	L, A, 4
6	$\varphi \multimap \sim\theta$	L, 1
7	$\sim(\varphi \vee \theta)$	(Dual⊸), L
8	$\sim(\varphi \multimap \sim(\psi \ovee \theta))$	(RAA), 5, 7
▷	$\varphi \vee (\psi \ovee \theta)$	(Dual⊸)

$(\mathrm{D}\vee\wedge)^1$:

A	$\varphi \vee (\alpha \wedge \mathrm{E}\beta)$	
1	$\varphi \vee \mathrm{E}(\alpha \wedge \beta)$	(AbsE∧)
2	$\mathrm{E}(\alpha \wedge \beta)$	(AbsE∨)
3	$\varphi \vee \alpha$	L, A
▷	$(\varphi \vee \alpha) \wedge \mathrm{E}(\alpha \wedge \beta)$	L

$(\mathrm{D}\vee\wedge)^2$:

A	$(\varphi \vee \alpha) \wedge \mathrm{E}(\alpha \wedge \beta)$	
1	$\varphi \vee \alpha$	L
2	$\mathrm{E}(\alpha \wedge \beta)$	L, A
3	$\top \vee ((\alpha \wedge \beta) \wedge \mathrm{E}(\alpha \wedge \beta))$	(I∨)
4	$\top \vee (\alpha \wedge \mathrm{E}\beta)$	H^{PL}, L
5	$(\varphi \vee \alpha) \vee (\alpha \wedge \mathrm{E}\beta)$	(Lax∨), 1, 4
8	$\varphi \vee (\alpha \vee (\alpha \wedge \mathrm{E}\beta))$	(Ass∨)
9	$\varphi \vee ((\alpha \wedge \alpha) \vee (\alpha \wedge \mathrm{E}\beta))$	H^{PL}
10	$\varphi \vee (\alpha \wedge (\alpha \vee \mathrm{E}\beta))$	(D∧∨)
▷	$\varphi \vee (\alpha \wedge \mathrm{E}\beta)$	(AbsE∨)

D Proof details for Lemma 6.47 (Table 6.9)

As for (MP∨) and (MP⊸), we mostly omit applications of (MP◊), (MP△) and (MP□) in the derivations.

$(\mathrm{Dis}\Box\wedge)^1$:

A	$\Box(\varphi \wedge \psi)$	
1	$\Box\varphi$	L, A
2	$\Box\psi$	L, A
▷	$\Box\varphi \wedge \Box\psi$	L

$(\mathrm{Dis}\Box\wedge)^2$:

A	$\Box\varphi \wedge \Box\psi$	
1 (hyp.)	$\sim\Box(\varphi \wedge \psi)$	
2	$\Box\sim(\varphi \wedge \psi)$	(Lin□)
3	$\Box(\varphi \rightarrowtriangle \sim\psi)$	L
4	$\Box\varphi \rightarrowtriangle \Box\sim\psi$	(Dis□)
5	$\Box\varphi$	L, A
6	$\Box\sim\psi$	(E⇾)
7	$\sim\Box\psi$	(Lin□)
8	$\Box\psi$	L, A
▷	$\Box(\varphi \wedge \psi)$	(RAA), 7, 8

(Dis$\Diamond\circledvee$)[1]:

A	$\Diamond(\varphi \circledvee \psi)$	
1	$\sim(\Diamond\varphi \circledvee \Diamond\psi)$	
2	$\sim(\sim\triangle\sim\varphi \circledvee \sim\triangle\sim\psi)$	(Dual$\Diamond$)
3	$\triangle\sim\varphi$	L, 2
4	$\triangle\sim\psi$	L, 2
5	$\sim\varphi \rightarrowtriangle (\sim\psi \rightarrowtriangle \sim(\varphi \circledvee \psi))$	(thm), L
6	$\triangle\sim(\varphi \circledvee \psi)$	(MP$\triangle$), 3, 4, 5
7	$\sim\sim\triangle\sim(\varphi \circledvee \psi)$	L
8	$\sim\Diamond(\varphi \circledvee \psi)$	(Dual$\Diamond$)
$\triangleright$	$\Diamond\varphi \circledvee \Diamond\psi$	(RAA), A, 8

(Dis$\Diamond\circledvee$)[2]:

A	$\Diamond\varphi \circledvee \Diamond\psi$	
1	$\sim\Diamond(\varphi \circledvee \psi)$	
2	$\sim\sim\triangle\sim(\varphi \circledvee \psi)$	(Dual$\Diamond$)
3	$\triangle\sim\varphi$	L, 2
4	$\triangle\sim\psi$	L, 2
5	$(\sim\sim\triangle\sim\varphi) \wedge (\sim\sim\triangle\sim\psi)$	L
6	$(\sim\Diamond\varphi) \wedge (\sim\Diamond\psi)$	(Dual$\Diamond$), L
7	$\sim(\Diamond\varphi \circledvee \Diamond\psi)$	L
$\triangleright$	$\Diamond(\varphi \circledvee \psi)$	(RAA), A, 7

(Com$\Diamond$E):

A	$\Diamond E\alpha$	
1	$\Diamond\sim\neg\alpha$	def.
2	$\sim E\Diamond\alpha$	
3	$\sim\sim\neg\Diamond\alpha$	def.
4	$\neg\Diamond\alpha$	L
5	$\Box\neg\alpha$	H^{ML}
6	$\triangle\neg\alpha$	(E$\Box$)
7	$\sim\Diamond\sim\neg\alpha$	(Dual$\Diamond$)
$\triangleright$	$E\Diamond\alpha$	(RAA), 1, 7

(ComE$\Diamond$):

A	$\Diamond\varphi$	
B	$E\Diamond\alpha$	
1	$\triangle\sim\sim\neg\alpha$	
2	$\triangle\neg\alpha$	L
3	$\Box\neg\alpha$	(I$\Box$), A, 2
4	$\neg\Diamond\alpha$	H^{ML}
5	$\sim\neg\Diamond\alpha$	def., B
6	$\sim\triangle\sim\sim\neg\alpha$	(RAA), 4, 5
$\triangleright$	$\Diamond E\alpha$	(Dual$\Diamond$), def.

(Aug$\Diamond$)

A	$\Diamond\varphi$	
B	$\triangle\psi$	
1	$\sim\Diamond(\varphi \wedge \psi)$	
2	$\triangle\sim(\varphi \wedge \psi)$	(Dual$\Diamond$), L
3	$\triangle(\psi \rightarrowtriangle \sim\varphi)$	L
4	$\triangle\psi \rightarrowtriangle \triangle\sim\varphi$	(Dis$\triangle$)
5	$\triangle\sim\varphi$	L, B, 4
6	$\sim\Diamond\varphi$	(Dual$\Diamond$), L
$\triangleright$	$\Diamond(\varphi \wedge \psi)$	(RAA), A, 6

(Join$\Diamond$)

A	$\Diamond\alpha$	
B	$E\Diamond\alpha$	
1	$\triangle\sim(\alpha \wedge E\alpha)$	
2	$\triangle\sim(\alpha \wedge \sim\neg\alpha)$	def.
3	$\triangle(\alpha \rightarrowtriangle \neg\alpha)$	L
4	$\Diamond(\alpha \wedge (\alpha \rightarrowtriangle \neg\alpha))$	(Aug$\Diamond$), A, 4
5	$\Diamond(\alpha \wedge \neg\alpha)$	L
6	$\bot$	H^{ML}
7	$\neg\Diamond\alpha$	H^{ML}
8	$\sim\neg\Diamond\alpha$	def., B
9	$\sim\triangle\sim(\alpha \wedge E\alpha)$	(RAA)
$\triangleright$	$\Diamond(\alpha \wedge E\alpha)$	(Dual$\Diamond$)

(Dis$\Diamond\wedge$)[1]:

A	$\Diamond(\alpha \wedge E\beta)$	
1	$\Diamond\alpha$	L
2	$\Diamond E(\alpha \wedge \beta)$	(AbsE$\wedge$), A
3	$E\Diamond(\alpha \wedge \beta)$	(Com$\Diamond$E)
$\triangleright$	$\Diamond\alpha \wedge E\Diamond(\alpha \wedge \beta)$	L, 1, 3

(Dis$\Diamond\wedge$)[2]:

A	$\Diamond\alpha \wedge E\Diamond(\alpha \wedge \beta)$	
1	$E\Diamond(\alpha \wedge \beta)$	L
2	$\top \vee (\Diamond(\alpha \wedge \beta) \wedge E\Diamond(\alpha \wedge \beta))$	(I$\vee$)
3	$\top \vee \Diamond((\alpha \wedge \beta) \wedge E(\alpha \wedge \beta))$	(Join$\Diamond$)
4	$\top \vee \Diamond(\alpha \wedge E\beta)$	L, H^{ML}
5	$\Diamond\alpha$	L, A
6	$\Diamond(\alpha \wedge \alpha)$	H^{ML}
7	$\Diamond(\alpha \wedge \alpha) \vee \Diamond(\alpha \wedge E\beta)$	(Lax$\vee$), 4, 6
8	$\Diamond((\alpha \wedge \alpha) \vee (\alpha \wedge E\beta))$	(Dis$\Diamond\vee$)
9	$\Diamond(\alpha \wedge (\alpha \vee E\beta))$	(D$\wedge\vee$)
$\triangleright$	$\Diamond(\alpha \wedge E\beta)$	(AbsE$\vee$), (Com$\vee$)

Index

Curriculum Vitae

Persönliches

Name	Martin Lück
Geburtsdatum und -ort	15. Dezember 1988, Güstrow

Ausbildung

1995–1999	Astrid-Lindgren-Grundschule Elze
06/2009	Allgemeine Hochschulreife, Gymnasium CJD Elze, Note: 1,9 Prüfungsfächer: Physik, Mathematik, Englisch, Religion, Informatik
10/2010–09/2013	Studium Bachelor Informatik, Nebenfach Physik Leibniz Universität Hannover, Note: 1,1 Abschlussthema: *Logikprogrammierung mit Abhängigkeiten*
10/2013–06/2015	Studium Master Informatik, Nebenfach Physik Leibniz Universität Hannover, Note: 1,0 Abschlussthema: *Parameterized Complexity of Temporal Logic*

Bisherige Tätigkeiten

11/2008–08/2010	Geringfügige Beschäftigung bei der Franz + Partner Steuerberatungsgesellschaft, Hannover.
07/2009–07/2010	Zivildienst und geringfügige Beschäftigung bei der Ambulante Krankenpflege Berezow GmbH, Hildesheim.
07/2011	Studentische Hilfskraft am Institut für Analysis, Leibniz Universität Hannover.
10/2011–06/2015	Studentische Hilfskraft an den Instituten für Praktische Informatik, Mikroelektronische Systeme, Systems Engineering und Theoretische Informatik, Leibniz Universität Hannover.
04/2015–06/2015	Studentische Hilfskraft am Institut für Raumfahrtsysteme, DLR Bremen.
seit 07/2015	Wissenschaftlicher Mitarbeiter am Institut für Theoretische Informatik, Leibniz Universität Hannover.

List of publications

[1] Martin Lück. *Canonical Models and the Complexity of Modal Team Logic*. Logical Methods in Computer Science **15**(2), 2019, 2:1–2:45.

[2] Martin Lück and Miikka Vilander. *On the Succinctness of Atoms of Dependency*. Logical Methods in Computer Science **15**(3), 2019, 17:1–17:28.

[3] Martin Lück. *Axiomatizations of team logics*. Annals of Pure and Applied Logic **169**(9), 2018, pp. 928–969.

[4] Martin Lück. *Canonical Models and the Complexity of Modal Team Logic*. 27th EACSL Annual Conf. on Computer Science Logic, CSL 2018, September 4-7, 2018, Birmingham, UK. Ed.: D. Ghica and A. Jung. **119**. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018, 30:1–30:23.

[5] Martin Lück. *On the Complexity of Team Logic and Its Two-Variable Fragment*. 43rd Int. Symp. on Mathematical Foundations of Computer Science, MFCS 2018, August 27-31, 2018, Liverpool, UK. Ed.: I. Potapov, P. Spirakis and J. Worrell. **117**. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018, 27:1–27:22.

[6] Martin Lück. *Quirky Quantifiers: Optimal Models and Complexity of Computation Tree Logic*. Int. J. of Foundations of Computer Science **29**(01), 2018, pp. 17–61.

[7] Martin Lück. *The Power of the Filtration Technique for Modal Logics with Team Semantics*. 26th EACSL Annual Conf. on Computer Science Logic, CSL 2017, August 20-24, 2017, Stockholm, Sweden. Ed.: V. Goranko and M. Dam. **82**. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017, 31:1–31:20.

[8] Martin Lück, Arne Meier and Irena Schindler. *Parametrised Complexity of Satisfiability in Temporal Logic*. ACM Transactions on Computational Logic **18**(1), 2017, 1:1–1:32.

[9] Miika Hannula, Juha Kontinen, Martin Lück and Jonni Virtema. *On Quantified Propositional Logics and the Exponential Time Hierarchy*. Proc. of the Seventh Int. Symp. on Games, Automata, Logics and Formal Verification, GandALF 2016, Catania, Italy, 14-16 September 2016. Ed.: D. Cantone and G. Delzanno. **226**. EPTCS. 2016, pp. 198–212.

[10] Martin Lück. *Axiomatizations for Propositional and Modal Team Logic*. 25th EACSL Annual Conf. on Computer Science Logic, CSL 2016, August 29 - September 1, 2016, Marseille, France. Ed.: J. Talbot and L. Regnier. **62**. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016, 33:1–33:18.

[11] Martin Lück and Arne Meier. *LTL Fragments are Hard for Standard Parameterisations*. 22nd Int. Symp. on Temporal Representation and Reasoning, TIME 2015. Ed.: F. Grandi, M. Lange and A. Lomuscio. IEEE Computer Society, 2015, pp. 59–68.

[12] Martin Lück, Arne Meier and Irina Schindler. *Parameterized Complexity of CTL - A Generalization of Courcelle's Theorem*. Language and Automata Theory and Applications - 9th Int. Conf., LATA 2015, Nice, France, March 2-6, 2015, Proc. Ed.: A. Dediu, E. Formenti, C. Martín-Vide and B. Truthe. **8977**. Lecture Notes in Computer Science. Springer, 2015, pp. 549–560.