Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Combined Proof Methods for Multimodal Logic

Daniella Angelos

Dissertação apresentada como requisito parcial para
qualificação do Mestrado em Informática

Orientadora
Prof.a Dr.a Cláudia Nalon

Brasília
2017

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Combined Proof Methods for Multimodal Logic

Daniella Angelos

Dissertação apresentada como requisito parcial para
qualificação do Mestrado em Informática

Prof.a Dr.a Cláudia Nalon (Orientadora)
CIC/UnB

Prof. Dr. Dr.

Prof. Dr. Bruno Luiggi Macchiavello Espinoza
Coordenador do Programa de Pós-graduação em Informática

Brasília, de de 2017

Abstract

Keywords: modal logics, resolution, sat-solvers

Contents

List of Figures

Chapter 1

Introduction

Chapter 2

Modal Logics

This chapter introduces K_n, a *propositional modal logic language*, semantically determined by an account of necessity and possibility.

A propositional modal language is the well known propositional language augmented by a collection of *modal operators*. In classical logic, propositions or sentences are either evaluated to true or false, in any model. Propositional logic and predicate logic, for instance, do not allow for any further possibilities. However, in natural language, we often distinguish between various modalities of truth, such as *necessarily* true, *known to be* true, *believed to be* true or yet true *in some future*, for example. Therefore, one may think that classical logics lacks expressivity in this sense.

Modal logic adds operators to express one or more of these different modes of truth. Different modalities define different languages. The key concept behind these operators is that they allow us to reason over relations among contexts or interpretations, an abstraction that here we think as *possible worlds*. The purpose of the modal operators is to permit the information that holds at other worlds to be examined — but, crucially, only at worlds visible from the current one via an accessibility relation [1]. Then, evaluation of a modal formula depends on a set of possible worlds and the accessibility relations defined for these worlds. It is possible to define several accessibility relations between worlds, and different modal logics are defined by different relations.

The modal language which is the focus of this work is the extension of the classical propositional logic that adds the unary operators: $\boxed{a}$ and $\langle a \rangle$, whose reading are "is necessary by the agent a" and "is possible by the agent a", respectively. This language, known as K_n, is characterized by the schema $\boxed{a}(\varphi \Rightarrow \psi) \Rightarrow (\boxed{a}\varphi \Rightarrow \boxed{a}\psi)$ (axiom K), where $a \in \mathcal{A} = \{1, \ldots, n\}$ and φ, ψ are well-formed formulae. The addition of other axioms defines different systems of modal logics and it imposes restrictions on the class of models where formulae are valid [3].

Worlds and their accessibility relations define a structure known as *Kripke model*.

The satisfiability and validity of a formula depend on this structure. For example, given a Kripke model that contains a set of possible worlds, a binary relation of accessibility between worlds and a valuation function that maps in which worlds a proposition symbol holds, we say that a formula $\Box p$ is satisfiable at some world w of this model, if the valuation function establishes that p is true at all worlds accessible from w.

It is now time to formally define the modal language we will be working with. The syntax and semantics of K_n are showed in Sections 2.1 and 2.2, respectively, and the definitions presented in these two sections are adaptations from [8].

2.1 Syntax

The language of K_n is equivalent to its set of *well-formed formulae*, denoted by $\mathrm{WFF}_{\mathsf{K}_n}$, which is constructed from a denumerable set of *propositional symbols* $\mathcal{P} = \{p, q, r, \ldots\}$, the negation symbol $\neg$, the disjunction symbol $\vee$ and the modal connectives $\boxed{a}$, that express the notion of necessity, for each index a in a finite, non-empty fixed set of labels $\mathcal{A} = \{1, \ldots, n\}, n \in \mathbb{N}$.

Definition 1 The set of well-formed formulae, $\mathrm{WFF}_{\mathsf{K}_n}$, is the least set such that:

1. $p \in \mathrm{WFF}_{\mathsf{K}_n}$, for all $p \in \mathcal{P}$
2. if $\varphi, \psi \in \mathrm{WFF}_{\mathsf{K}_n}$, then so are $\neg\varphi$, $(\varphi \vee \psi)$ and $\boxed{a}\varphi$, for each $a \in \mathcal{A}$

Just as the familiar first-order existential and universal quantifiers are duals to each other, that is, $\forall x\ \varphi \Leftrightarrow \neg\exists x\ \neg\varphi$, we have the dual connectives $\Diamond_a$ for necessity, which express possibility, and they are defined by $\Diamond_a\varphi \stackrel{\text{def}}{=} \neg\boxed{a}\neg\varphi$, for each $a \in \mathcal{A}$. Other logic operators may be used as abbreviations. In this work, we consider the usual ones:

- $\varphi \wedge \psi \stackrel{\text{def}}{=} \neg(\neg\varphi \vee \neg\psi)$ (conjuction)
- $\varphi \Rightarrow \psi \stackrel{\text{def}}{=} \neg\varphi \vee \psi$ (implication)
- $\varphi \Leftrightarrow \psi \stackrel{\text{def}}{=} (\varphi \Rightarrow \psi) \wedge (\psi \Rightarrow \varphi)$ (equivalence)
- $\mathbf{false} \stackrel{\text{def}}{=} \varphi \wedge \neg\varphi$ (*falsum*)
- $\mathbf{true} \stackrel{\text{def}}{=} \neg\mathbf{false}$ (*verum*)

Parentheses may be omitted if the reading is not ambiguous. When $n = 1$, we often omit the index in the modal operators, i.e., we just write $\Box\varphi$ (or 'box' φ) and $\Diamond\varphi$ (or 'diamond' φ), for a well-formed formula φ.

We define as *literal* a propositional symbol $p \in \mathcal{P}$ or its negation $\neg p$, and denote by $\mathcal{L}$ the set of all literals. A *modal literal* is a formula of the form $\boxed{a} l$ or $\diamondsuit_a l$, with $l \in \mathcal{L}$ and $a \in \mathcal{A}$.

The following function definitions based on formulae' syntax will be helpful when we introduce the proof system we make use in this work:

Definition 2 The *modal depth* of a formula, $mdepth : \mathrm{WFF}_{\mathsf{K}_n} \longrightarrow \mathbb{N}$, represents the maximal number of nesting modal operators in this formula. Inductively, we have:

1. $mdepth(p) = 0$
2. $mdepth(\neg\varphi) = mdepth(\varphi)$
3. $mdepth(\varphi \vee \psi) = \max\{mdepth(\varphi), mdepth(\psi)\}$
4. $mdepth(\boxed{a}\varphi) = mdepth(\varphi) + 1$

With $p \in \mathcal{P}$ and $\varphi, \psi \in \mathrm{WFF}_{\mathsf{K}_n}$.

Definition 3 The *modal level* function, $ml : \mathrm{WFF}_{\mathsf{K}_n} \longrightarrow \mathbb{N}$, represents the maximal number of modal operators in which scope the formula occurs.

For instance, in $\boxed{a}\diamondsuit_a p$, $mdepth(p) = 0$ and $ml(p) = 2$.

2.2 Semantics

The semantics of K_n is presented in terms of Kripke structures.

Definition 4 A Kripke model for $\mathcal{P}$ and $\mathcal{A} = \{1, \ldots, n\}$ is given by the tuple

$$\mathcal{M} = (W, w_0, R_1, \ldots, R_n, \pi) \tag{2.1}$$

where W is a non-empty set of possible worlds with a distinguinshed world w_0, the root of $\mathcal{M}$; each R_a, $a \in \mathcal{A}$, is a binary relation on W, that is, $R_a \subseteq W \times W$, and $\pi : W \times \mathcal{P} \longrightarrow \{false, true\}$ is the valuation function that associates to each world $w \in W$ a truth-assignment to propositional symbols.

Satisfiability and *validity* of a formula is defined in terms of the *satisfiability relation*.

Definition 5 Let $\mathcal{M} = (W, w_0, R_1, \ldots, R_n, \pi)$ be a Kripke model, $w \in W$ and $\varphi, \psi \in \mathrm{WFF}_{\mathsf{K}_n}$. The *satisfiability relation*, denoted by $\langle \mathcal{M}, w \rangle \models \varphi$, between a world w and a formula φ, is inductively defined by:

1. $\langle \mathcal{M}, w \rangle \models p$ if, and only if, $\pi(w, p) = \mathbf{true}$, for all $p \in \mathcal{P}$;
2. $\langle \mathcal{M}, w \rangle \models \neg\varphi$ if, and only if, $\langle \mathcal{M}, w \rangle \not\models \varphi$;
3. $\langle \mathcal{M}, w \rangle \models \varphi \vee \psi$ if, and only if, $\langle \mathcal{M}, w \rangle \models \varphi$ or $\langle \mathcal{M}, w \rangle \models \psi$
4. $\langle \mathcal{M}, w \rangle \models \boxed{a}\varphi$ if, and only if, for all $t \in W$, $(w, t) \in R_a$ implies $\langle \mathcal{M}, t \rangle \models \varphi$

Satisfiability is defined with respect to the root of a model. A formula $\varphi \in \mathrm{WFF}_{\mathsf{K}_n}$ is said to be *satisfiable* if there exists a Kripke model $\mathcal{M} = (W, w_0, R_1, \ldots, R_n, \pi)$ such that $\langle \mathcal{M}, w_0 \rangle \models \varphi$. A formula is said to be *valid* if it is satisfiable in all models.

The satisfiability problem for K_n corresponds to determining the existence of a model in which a formula is satisfied. This problem is proven to be PSPACE-complete [10].

Example 1 In Figure 2.1.

2.3 Normal Form

Normal forms can provide elegant and constructive proofs of many standard results. They can also provide proofs of results that are not readily proved by standard means [6].

Formulae in K_n can be transformed into a normal form called *Separated Normal Form for Normal Logics* [8].

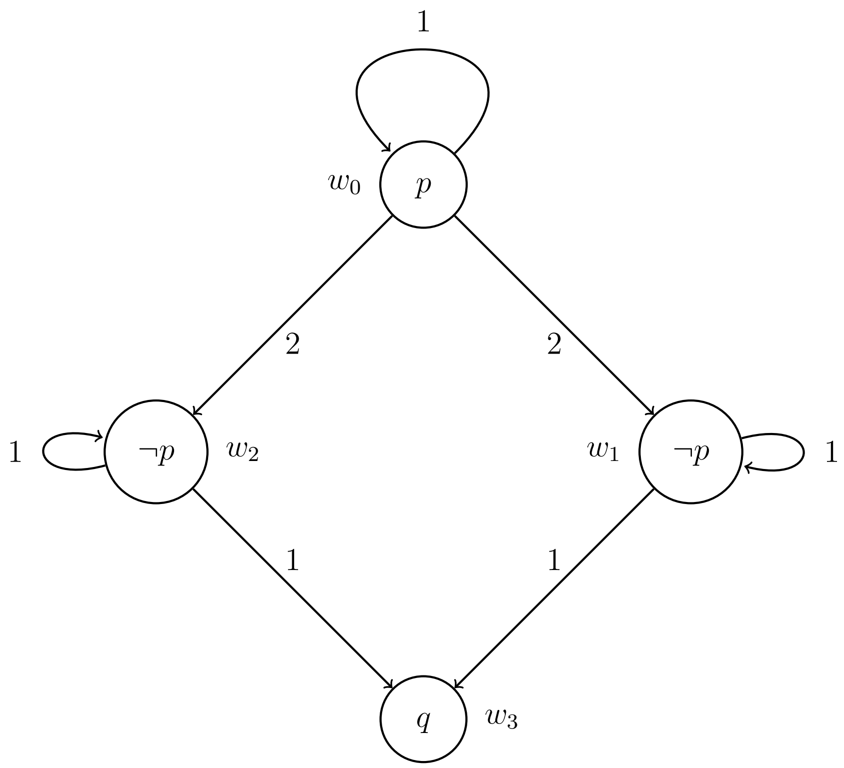

Figure 2.1: Example of a Kripke model for K_n

Chapter 3

Resolution

Resolution appeared in the early 1960s through investigations on performance improvements of refutational procedures based on *Herbrand Theorem*. In particular, Prawitz' studies on such procedures brought back the idea of unification. J. A. Robinson incorporated the concept of unification on a refutation method, creating what was later known as resolution [2].

The standard rules for resolution systems take two or more premises with literals or modal literals that are contradictory, and generate a resolvent. Most of these systems work exclusively with clauses in a specific normal form. Resolution systems are, in general, refutational systems, that is, to show that a formula φ is valid, $\neg\varphi$ is translated into a normal form. The inference rules are applied until either no new resolvents can be generated or a contradiction is obtained. The contradiction implies that $\neg\varphi$ is unsatisfiable and hence, that φ is valid.

3.1 Clausal Resolution

Clausal resolution was proposed as a proof method for classical logic by Robinson in 1965 [9], and was claimed to be suitable to be performed by computer, as it has only one inference rule that is applied many times. Nonclausal proof methods, in general, require a larger number of rules, making implementation more difficult. Clausal resolution is a simple and adaptable proof method for classical logics and, since it was proposed, a bank of research into heuristics and strategies has been growing.

3.1.1 Substitution

3.1.2 Unification

3.1.3 The resolution principle

Chapter 4

Satisfiability Solvers

The problem of determining whether a boolean formula is satisfiable has the historical honor of being the first problem ever shown to be NP-Complete [4]. Great theoretical and practical efforts have been directed in improving the efficiency of solvers for this problem, known as *Boolean Satisfiability Solvers*, or just *SAT solvers*. Despite the worst-case exponential run time of all the algorithms known, satisfiability solvers are increasingly leaving their mark as a general purpose tool in the most diverse areas [7]. In essence, SAT solvers provide a generic combinatorial reasoning and search platform.

In the context of SAT solvers, the underlying representational formalism is propositional logic. We are interested in propositional formulae in *Conjunctive Normal Form* (CNF): F is in CNF if it is a conjunction of *clauses*, where each clause is a disjunction of *literals*. For example, $F = (p \vee \neg q) \wedge (\neg p \vee r \vee s) \wedge (q \vee r)$ is a CNF formula with four variables and three clauses.

Therefore, the *Boolean Satisfiability Problem* (SAT) can be expressed as: Given a CNF formula F, does F have a satisfying assignment? One can be interested not only in the answer of this decision problem, but also in finding an actual satisfying assignment when it there exists one. All practical SAT solvers do produce such assignment [5].

4.1 The DPLL Procedure

4.2 Conflict Driven Clause Learning

4.3 MiniSat

Algorithm 1: DPLL-recursive(F, ρ)

Input : A CNF formula F and an initially empty partial assignment ρ
Output: UNSAT, or an assignment satisfying F

$(F, \rho) \leftarrow$ UnitPropagate(F, ρ)
if F *contains the empty clause* **then**
    return UNSAT
end
if F *has no clauses left* **then**
    Output ρ
    return *SAT*
end
$l \leftarrow$ a literal not assigned by ρ
if *DPLL-recursive*($F|_l, \rho \cup \{l\}$) $=$ *SAT* **then**
    return *SAT*
end
return *DPLL-recursive*($F|_{\neg l}, \rho \cup \{\neg l\}$)

Algorithm 2: CDCL(F, ν)

Input :
Output:

if *UnitPropagate*(F, ν) $==$ *CONFLICT* **then**
    return *UNSAT*
end
$dl \leftarrow 0$
while ***not*** *AllVariablesAssigned*(F, ν) **do**
    $(x, \nu) \leftarrow$ PickBranchingVariable(F, ν)
end
return *SAT*

Bibliography

[1] P. Blackburn, M. De Rijke, and Y. Venema. *Modal Logic: Graph. Darst*, volume 53. Cambridge University Press, 2002. 2

[2] M. A. Casanova. *Programação em lógica e a linguagem Prolog.* E. Blucher, 1987. 7

[3] B. F. Chellas. *Modal logic — an introduction.* Press Syndicate of the University of Cambridge, London, 1980. 2

[4] S. A. Cook. The complexity of theorem proving procedures. In *stoc71*, pages 151–158, 1971. 9

[5] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms, 3rd Edition.* MIT Press, 2009. 9

[6] K. Fine. Normal forms in modal logic. *Notre Dame J. Formal Logic*, 16(2):229–237, 04 1975. 5

[7] C. P. Gomes, H. Kautz, A. Sabharwal, and B. Selman. Satisfiability solvers. *Foundations of Artificial Intelligence*, 3:89–134, 2008. 9

[8] C. Nalon and C. Dixon. Clausal resolution for normal modal logics. *J. Algorithms*, 62(3-4):117–134, 2007. 3, 5

[9] J. A. Robinson. A machine-oriented logic based on the resolution principle. *Journal of the ACM*, 12(1):23–41, Jan. 1965. 7

[10] E. Spaan. *Complexity of Modal Logics.* PhD thesis, University of Amsterdam, 1993. 5