BDD-based automated reasoning in propositional non-classical logics: progress report

Rajeev Goré and Jimmy Thomson

Logic and Computation Group
Research School of Computer Science
The Australian National University
Canberra, ACT 0200, Australia

Abstract

Recent work has shown that a technique using Binary Decision Diagrams (BDDs) to decide **CTL** and **Int** gives promising results. Based on this we explore how the method can be extended to other non-classical logics. In particular, we describe a putative method for deciding the modal μ-calculus using BDDs.

1 Introduction

For many logics, we can decide the validity of a given formula φ_0 by constructing the set of all subsets of some closure $cl(\varphi_0)$, and checking whether these subsets can support a (counter) model that makes φ_0 false. If no such model exists, then we can safely declare φ_0 to be valid. Typically, we proceed by first building a finite pseudo-model where each "world" is a member of $2^{cl(\varphi)}$, and then showing that the pseudo-model can be "unfolded" into a model.

At first sight, this "finite pseudo-model (fpm) method" seems impractical since the first step requires us to "construct" the set of all (exponentially many) subsets of $cl(\varphi_0)$, thus giving a procedure whose worst case and best case complexity is always of order $O(2^{|cl(\varphi_0)|})$. However, for **K** and **CTL**, Pan et al. [5] and Marrero [4] have shown that Binary Decision Diagrams (BDDs) can be used to represent the required subsets efficiently, without actually "constructing" them explicitly. We have recently shown how to extend this method to handle modal, tense and bi-extensions of intuitionistic logic **Int** [3]. In particular, for **CTL** and **Int** the resulting reasoners were highly competitive with the current state of the art [2, 3].

In light of this, we are exploring whether such BDD-based implementations can be extended to handle a number of other non-classical logics, and if so, to see whether the practicality remains. Here we concentrate on extending the method to many different classical modal logics, and in particular, the modal mu-calculus. Practicality remains to be seen since we are still implementing the various classical modal logics described here. We do have an initial unoptimised implementation of the putative mu-calculus BDD-method which minimal testing has shown to give the correct answers so far. We have not made it available since it is quite possible that our soundness and completeness proofs for the mu-calculus may not pan out.

Since the focus of PAAR is on practicality, we have deliberately given our descriptions at a lower level than for **Int** [3]. Thus while previously we elided explicit BDD aspects, here they are included, so it may be beneficial to read the other paper first.

We assume that the reader is familiar with non-classical logics in general, in particular with the notion of Kripke semantics and the fpm method for deciding satisfiability. Before discussing extensions to our implementation of the fpm method, we begin by presenting the ideas behind the fpm method in general as a guide for following its actual BDD-based implementation.

1.1 An Abstract View of the fpm Method

Given a semantically formulated logic **L**, a naive way to determine satisfiability and validity is to consider the set of all models for said logic and literally determine whether some world in some model exists that makes a formula φ true, or if all worlds in all models make φ true. In theory, the set of all models is infinite, and in some cases the set of worlds in one model may also be infinite, so we need a way to explore this possibility in a finitary way.

Given a formula φ_0, the intent of the fpm method is to find a finite filtration of the set of all worlds, such that each member in the filtration represents an equivalence class in the original worlds, and the denotation of the formula φ_0 in the original model depends only on the truth value of this formula at some representative of this equivalence class. This allows us to examine all worlds in a finitary way. We will in general be referring to members of the filtration as worlds or potential worlds, effectively taking any representative of the equivalence class.

The finite filtration itself is constructed by identifying a finite set of formulae $cl(\varphi_0)$, usually called the "closure of φ_0", and transforming the given, possibly infinite, model $\mathcal{M}$ into a finite pseudo-model $\mathcal{M}_{cl(\varphi_0)}$ such that the truth value of members of $cl(\varphi_0)$ is preserved.

1.2 An operational view of the fpm method

With the space of all potential worlds restricted to a finite space $2^{cl(\varphi_0)}$ (initially), it remains to identify which of these potential worlds correspond to worlds in actual models. The pseudo-model method constructs a finite pseudo-model (W_f, R^f) which is canonical in the following sense: if φ_0 is satisfiable (falsifiable) then some world of W_f satisfies (falsifies) φ_0. We find these worlds W_f by defining a monotonic function on sets of potential worlds that removes worlds from the argument set if they contradict the semantics of the logic. For example, a potential world claiming to satisfy both $\Box p$ and $\neg\Box p$ goes against the semantics of most logics, and thus must be removed if present. Thus we construct a chain $W_0, W_1, \cdots W_f$ of refinements on the set of an initial set W_0, until W_f is immune to our monotonic function (a fixpoint).

Any (non-empty) fixpoint of this appropriately-constructed function corresponds to a set of worlds which all agree with the semantics of the logic. The "completeness" of this approach requires us to show that every world in any model must have a representative in W_f. Because the function we construct satisfies the conditions of the Knaster-Tarski theorem [7] it has a greatest fixpoint, and moreover the greatest fixpoint is a superset of all fixpoints. Thus the greatest fixpoint contains representatives for all worlds in all models. For this reason we start with $W_0 = 2^{cl(\varphi_0)}$, as repeated iteration from the top element will compute the greatest fixpoint.

The "soundness" of this approach requires showing that each world remaining in W_f can be extended into a model using only other worlds in the fixpoint. In some logics this is immediate. In others, like **CTL**, the set must be "unwound" or otherwise manipulated to construct a model.

Thus generating the set of all worlds modulo the closure $cl(\varphi_0)$ of some formula φ_0 and computing the greatest fixpoint of a sound and complete semantics-inspired function gives a decision procedure where satisfiability and validity are determined by checking whether the intersection of those worlds claiming to satisfy or falsify φ_0 with the set of all worlds is empty.

We describe specifics of how BDDs are used and how the fixpoint construction works, using **K** as an example, first described by Pan et al. [5].

2 Implementation in BDDs

We now describe the BDD implementation at a high level.

Constructing a Finite Set of Finite Worlds. As we have seen, given some finite closure $cl(\varphi_0)$, the naive way to construct the finite set of all finite worlds is simply to use the set of all subsets of $cl(\varphi_0)$. We instead use only the "sensible subsets" following Pan et al. and Marrero. Specifically, we construct $Atoms(\varphi_0)$ as the base set of atoms whose truth values guarantee that we can distinguish worlds. Typically this is the non-classical subset of the closure, from which the truth values of the rest of the closure can be computed using classical conjunction, disjunction and negation. We then define $\mathcal{W} = 2^{Atoms(\varphi_0)}$ to be the set of all subsets of these atoms. Any potential world will either satisfy or falsify each of these atoms, so we can associate a world w with exactly the set of atoms that it satisfies, and hence view w as a simple bi-valent valuation on $Atoms(\varphi_0)$. The set $\mathcal{W}$ is smaller than $2^{cl(\varphi_0)}$, and does not contains worlds which behave inappropriately with respect to classical conjunction and disjunction.

For the logic **K**, an acceptable closure $cl(\varphi_0)$ is the set of subformulae of φ_0 and their negations. The set of atoms however is defined as the smaller set:

$$Atoms(\varphi_0) = \{\Box\psi \mid \Box\psi \in cl(\varphi_0)\} \cup (Prop \cap cl(\varphi_0))$$

For example, $Atoms(\Box(p \Rightarrow q) \Rightarrow \Box p \Rightarrow \Box q) = \{p, q, \Box p, \Box q, \Box(p \Rightarrow q)\}$ and the set $\{p, \Box(p \Rightarrow q)\}$ corresponds to a world that makes p and $\Box(p \Rightarrow q)$ true, and makes q, $\Box p$ and $\Box q$ false.

BDDs as set of worlds. We need an efficient way to represent potential worlds and (denotations of formulae as) sets of potential worlds.

A BDD over a set $V = \{v_1, \cdots, v_k\}$ of Boolean-valued variables represents a function mapping each Boolean valuation on these variables to one of $\{t, f\}$. If we associate each atom $a \in Atoms(\varphi_0)$ with a unique BDD variable v_a, then a BDD over these variables is a function mapping each valuation on $Atoms(\varphi_0)$ to one of $\{t, f\}$. If we view the valuations which the BDD maps to t as being "selected", then a BDD represents a set of valuations, or a set of potential worlds. Thus a BDD is a function $f : 2^V \mapsto \{t, f\}$ that selects a subset from the powerset 2^V of V.

For example, in **K** with atoms as above, the set $\{p, \Box(p \Rightarrow q)\}$ corresponds to a valuation under which the BDD variables v_p and $v_{\Box(p \Rightarrow q)}$ are true, while $v_q, v_{\Box p}$ and $v_{\Box q}$ are all false. The BDD which returns t whenever v_p, v_q, and $v_{\Box p}$ are true corresponds to the set of worlds $\{\{p, q, \Box p\}, \{p, q, \Box p, \Box q\}, \{p, q, \Box p, \Box(p \Rightarrow q)\}, \{p, q, \Box p, \Box q, \Box(p \Rightarrow q)\}\}$.

In particular, the BDD $\top$, which returns t for every valuation, represents the set $\mathcal{W}$ of *all* worlds/subsets over $Atoms(\varphi_0)$ in constant space and time!

The fpm method is usually considered to be naive because it must *"first construct the set of all subsets of $cl(\varphi_0)$, whose cardinality is exponential in the size of $cl(\varphi_0)$"*. The main reason why the fpm method can be implemented efficiently using BDDs is that they turn this "wisdom" on its head. Specifically, by using reduced ordered BDDs the BDD only branches on variables that would cause two valuations to give different results.

Defining denotations. For each $a \in Atoms(\varphi_0)$ we use $[\![a]\!]$ to refer to the BDD which is true exactly when the variable corresponding to a is true. Equivalently, $[\![a]\!]$ is the set of worlds that make a true. The denotations of non-atomic formulae in the closure $cl(\varphi_0)$ are computed inductively, usually in an obvious way. For example for **K**, $[\![\psi \wedge \phi]\!] = [\![\psi]\!] \wedge [\![\phi]\!]$, similarly for disjunction and negation, and $[\![\Diamond\psi]\!] = \neg[\![\Box\neg\psi]\!]$.

Representing relations. All the logics we consider have relational Kripke semantics so we must be able to represent and reason about these relations.

A BDD $f : 2^V \mapsto \{t, f\}$ over a finite set of variables V corresponds to some subset (of worlds) of 2^V. Consider a BDD $g(V \cup V')$, corresponding to some subset S of $2^{V \cup V'}$. Any member of S, such as $\{v_1, \ldots, v_k\} \cup \{v'_1, \ldots, v'_k\}$, corresponds to a particular valuation on $V \cup V'$. If we conceptually split the valuation into its two components over V and over V', as above, then we can view the valuation as an ordered pair of sub-valuations. This allows us to think of $g(V \cup V')$ as a subset of $2^V \times 2^{V'}$ since there is a bijection from $2^{V \cup V'}$ onto $2^V \times 2^{V'}$ whenever V and V' are disjoint. If $Atoms(\varphi_0)$, V and V' have the same cardinality then $g(V \cup V')$ can be viewed as a subset of $\mathcal{W} \times \mathcal{W}$ using any bijection of $Atoms(\varphi_0)$ onto V and V'.

When discussing such BDDs representing pairs, we can think of $g(V \cup V')$ as $g(V, V')$. We will often construct such BDDs by combining BDDs using variables in V or V'. When we write $[\![\psi]\!]$, it is constructed from variables in V and represents the first world of a pair, while when we write $[\![\psi]\!]'$ it is constructed from variables in V' and represents the second world in a pair. There is some subtlety here: writing $[\![\psi']\!]$ does not make any sense since ψ is from $cl(\varphi_0)$. Thus $[\![\psi]\!]'$ is a BDD defined over V', which is obtained by making a "photocopy" of the BDD over V for $[\![\psi]\!]$ and replacing each $v_i \in V$ with its clone $v'_i \in V'$.

The case of K. Constraints on the specific relation vary by logic, but we present the reasoning for **K** here. The semantics of **K** refer to a Kripke relation R. The relation itself is unrestricted, but its interactions with the modal formulae provide constraints such as the following:

$$\forall w.\mathcal{M}, w \Vdash \Box\psi \Rightarrow \forall v.R(w, v) \Rightarrow \mathcal{M}, v \Vdash \psi \tag{1}$$

Dropping quantifiers, we can rearrange this formula to state a restriction on R:

$$R(w, v) \Rightarrow \mathcal{M}, w \Vdash \Box\psi \Rightarrow \mathcal{M}, v \Vdash \psi \tag{2}$$

We treat this formula as an upper bound on R, and take the intersection of all the right hand sides given by all $\Box$-formulae in the closure $Atoms(\varphi_0)$ as the definition of a maximal R, where maximal means that it links any two worlds that are "allowed to be linked":

$$R(w, v) = \bigwedge_{\Box\psi \in Atoms(\varphi_0)} \mathcal{M}, w \Vdash \Box\psi \Rightarrow \mathcal{M}, v \Vdash \psi \tag{3}$$

The semantics of $\Box$-formulae are captured by this maximal R: if two worlds can be related by this R, and the first world w claims to satisfy a $\Box\psi$, then the second world v must satisfy ψ.

The specific BDD representation of this constraint is as follows, where we use $R(V, V')$ to represent a BDD parametrised by sets of variables V and V':

$$R(V, V') = \bigwedge_{\Box\psi \in Atoms(\varphi_0)} [\![\Box\psi]\!] \Rightarrow [\![\psi]\!]' \tag{4}$$

We have now represented both $\mathcal{W}$ and R using BDDs over $Atoms(\varphi_0)$ and their copies. Recall that the general procedure requires us to refine $\mathcal{W}$ to exclude those worlds that do not obey the semantics of **K**. The remaining task is to construct and solve a fixpoint formula corresponding to the remaining semantics of the logic. We will construct a greatest-fixpoint formula which is monotonic decreasing, so by the Knaster-Tarski theorem we can repeatedly iterate the formula starting with the top element $\mathcal{W}_0 = \mathcal{W} = \top$ to compute the greatest fixpoint $\mathcal{W}_f$. Note that the set $\mathcal{W}$, despite representing $2^{|Atoms(\varphi_0)|}$ worlds, is represented very succinctly by the BDD $\top$, and in general the size of a BDD (and time taken to perform BDD operations) is not proportional to the size of the set it represents but instead depends upon the dependencies between the variables in the characteristic function of the set.

For **K**, the choice of atoms and definition of $[\![\cdot]\!]$ address the classical semantics of $\wedge, \vee$ and $\neg$, and the construction of R enforces the semantics of the $\Box$-formulae. The only semantic conditions left to address are for $\Diamond$-formulae:

$$\forall w.\mathcal{M}, w \Vdash \Diamond\psi \Rightarrow \exists v.R(w,v) \wedge \mathcal{M}, v \Vdash \psi \tag{5}$$

This equation can be used almost exactly to enforce the constraint. One thing to note is that v must be in $\mathcal{M}$, that is it must be in the set of "good" worlds being considered.

$$good(S) = S \wedge \bigwedge_{\Diamond\psi \in cl(\varphi_0)} [\![\Diamond\psi]\!] \Rightarrow \exists V'.R(V,V') \wedge S(V') \wedge [\![\psi]\!]' \tag{6}$$

By $S(V')$ here, we mean a "photocopy" of S, where each variable in V is replaced by the variable in V' that corresponds to the same atom.

Given a set of potential good worlds S, this function retains the worlds that have candidate R-successors in S to witness each diamond they claim to satisfy. That is, it removes from S all potential worlds which cannot "satisfy" their diamonds in the set S.

The existential appearing in this formula is QBF-style quantification over a set of variables. Intuitively, this is logically equivalent to the disjunction of all assignments to those variables. In practice, the BDD package we used provides such a function. We have not looked into better ways of doing it ourselves since this is beyond the scope of our research.

3 Potential Extensions to other Non-Classical Logics

We now show how the method for **K** [5] can be extended in various directions. Note that all logics considered in this section are known to be decidable (via the "fpm method"), so the main question is really just whether we can find an easy way to capture the method using BDDs.

3.1 Multimodal K, extra frame conditions and interacting relations

The huge diversity of propositional modal logics arises from the ability to modally characterise numerous first-order frame conditions on the underlying binary Kripke relation (s).

Multimodal K. Pan et al. do not need to consider extra frame conditions on the reachability relation since the modal logic **K** allows arbitrary frames. Extending from **K** to multimodal **K** (aka $\mathcal{ALC}$) is simple as the semantics of the modalities are independent of each other. Instead of constructing a single R relation, there is now an R_π relation for each action π. In the greatest fixpoint computation, instead of referring to the relation R, the appropriate R_π is used for the $\langle\pi\rangle\psi$ formula at hand. Otherwise, everything follows as for **K** [5].

Extra frame conditions. Adding extra frame conditions is not quite so trivial. Marrero handles seriality in **CTL** [4], while our work on **Int** [3] shows how to handle reflexivity and transitivity. We revisit these conditions, and show how to handle euclideanness and symmetry.

Having computed a maximal base relation R^0, how can we enforce reflexivity? A naive way is to just take the reflexive closure of R^0. However, the R^0 we compute is maximally permissive, so if $(w,w) \notin R^0$ then this indicates that w cannot be part of a reflexive model. Thus it is not sound to just add (w,w) back, instead we must remove w from the set of potential worlds by considering only the reflexive worlds from the start:

$$W_0 = \mathcal{W}_{refl} = R^0(V, V) \tag{7}$$

In a similar way, seriality can't be enforced by modifying a base maximal relation R^0. In addition, as the fixpoint procedure refines the set of worlds and thus the domain of R, the restricted relation may become non-serial, so seriality must be addressed as part of the fixpoint function. Marrero enforces seriality by modifying the *good* function as follows:

$$good(S) = (\exists V'.R(V, V') \wedge S(V')) \wedge \ldots \tag{8}$$

Both seriality and reflexivity are modular in that adding these constraints work for any context without knowledge of the maximal relation R^0, while transitivity requires extra knowledge of R^0. The important thing about transitivity is the concept of preserving constraints forwards: in the simple case of **K4** this is equivalent to "boxes persist", but with a more complicated relation or logic this may need to be re-evaluated:

$$R(V, V') = R^0(V, V') \wedge \bigwedge_{\Box\psi \in Atoms(\varphi_0)} [\![\Box\psi]\!] \Rightarrow [\![\Box\psi]\!]' \tag{9}$$

Euclideanness can be treated in a very similar way to transitivity. Instead of constraints persisting forwards, constraints must persist backwards: if some successor of w has a constraint ($\Box$-formula) then w itself must have that constraint. For **K5** this is encapsulated as follows:

$$R(V, V') = R^0(V, V') \wedge \bigwedge_{\Box\psi \in Atoms(\varphi_0)} [\![\Box\psi]\!]' \Rightarrow [\![\Box\psi]\!] \tag{10}$$

Symmetry can be handled in a modular way given a maximal relation R^0 by restricting it to the maximal symmetric sub-relation as follows:

$$R(w, v) = R^0(w, v) \wedge R^0(v, w) \qquad \equiv \qquad R(V, V') = R^0(V, V') \wedge R^0(V', V) \tag{11}$$

We can thus handle the basic modal logics **KT**, **KD**, **K4**, **K5** and **KB**. The modularity of most of these extensions, and the simplicity of transitivity and euclideanness means that we can also handle combinations of these, allowing us to deal with the 15 basic normal modal logics.

Interacting relations. Another direction to consider is interactions between relations. We showed that this approach extends to **BiKt** [3] which has two interacting modal relations. In that case, we were able to sidestep the complications by showing that we could work in a different frame without interaction conditions, and get equivalent answers.

Some interaction conditions are plausibly able to be handled directly however. Statements such as one relation R_1 contains R_2 result in constraints like so:

$$R_2^1(V, V') = R_2^0(V, V') \wedge R_1^0(V, V') \tag{12}$$

Thus, if wR_2^1v, then wR_1^0v. If there are multiple conditions, then these restrictions may have to be chained together. Also, such restrictions do not preserve transitivity. If R_2 is transitive and R_1 is not, then R_2^1 may not be transitive after this restriction even if R_2^0 is transitive.

Finally, one of the strengths of this method is its versatility. For example, there are two ways to obtain tense logic **Kt**. The first is to start with two modal relations $R_\Box$ and $R_\blacksquare$, and enforce $R_\Box = R_\blacksquare^{-1}$ by requiring that each relation is a subset of the other. The other is to use a single relation R which is defined from semantics referring to both $\Box$-formulae and $\blacksquare$-formulae.

3.2 Description Logic

Lite description logics are deliberately weakened fragments of multimodal logics, which can of course be solved using the approaches described here. However, constructing formulae from BDDs is potentially an exponential operation in the number of atoms, and the fixpoints potentially take an exponential number of iterations to compute, so the decision procedure would be inherently exponential, not benefiting from the low computational complexity of Lite logics.

Additionally, a common use-case of description logics is in situations where constraints are relatively simple, but the number of concepts and individuals becomes very large. In these situations this method may not work well, as the number of atoms becomes large.

However, there are ways in which this approach may benefit the types of reasoning done in description logics. Specifically, classifying a TBox reduces to calculating the greatest fixpoint using the denotation of the TBox as an initial value instead of $\mathcal{W}$, then multiple simple queries can be made of the final set to determine whether $C \sqsubseteq D$ for each C and D.

Not all the features of the more expressive description logics are feasible either, specifically it is not obvious to us how to handle cardinality constraints.

Functional properties may be possible by treating both $\exists R.C$ and $\forall R.C$ in the same way, as they must both refer to the single successor that an individual must have. The constructed R relation may not itself be functional, but by choosing exactly one of the options at each world, a model where it is functional can be generated.

3.3 Hybrid logic

A fixed finite set of nominals is plausible, but binder causes problems both because the logic becomes undecidable, and it is not obvious how to allow arbitrary worlds to be named.

K-with-nominals can be represented by requiring that that if a nominal i is true at some world, any other world in the same model claiming to be i must be equivalent to that world.

To represent this, the set of atoms is not just those of **K** with additional atoms for each nominal i^k, but an additional $m \times |Atoms(\varphi_0)|$ new atoms for m nominals i^k. For each of the "base" atoms a, the "additional" atom $@_i a$ is read as "In this model, the world i makes a true". These new atoms must be invariant over the modal relation as shown below at the left, and must interact with the base nominal atoms as shown below at the right:

$$R(V, V') \Rightarrow [\![@_i a]\!] \Leftrightarrow [\![@_i a]\!]' \qquad\qquad [\![i]\!] \Rightarrow [\![@_i a]\!] \Leftrightarrow [\![a]\!]$$

Now if there is a path along R and its converse between two worlds (that is, they appear in the same model) that both claim to make i true, they must be represented by the same set of atoms. Without complications such as irreflexivity or antisymmetry, we are able to treat the equivalence classes as worlds themselves, and thus we can construct a model where the nominal is true at exactly one world.

This choice of atoms works with non-atomic $@_i \psi$ as well by deconstructing ψ into atoms, using $[\![@_i(\psi \wedge \varphi)]\!] = [\![@_i \psi]\!] \wedge [\![@_i \varphi]\!]$, similarly for $\vee$ and $\neg$, and $[\![@_i @_j \psi]\!] = [\![@_j \psi]\!]$.

Although the number of atoms is significantly larger than for **K** and thus performance may well suffer, the procedure remains EXPTIME . But because the i_a^k atoms essentially partition the set of worlds into non-interacting components, the impact on performance may be reduced.

4 Using BDDs to decide the μ-calculus

The μ-calculus is infamously tricky to work with, both understanding what a particular formula "means" and deciding whether or not a given formula is satisfiable. The primary difficulty arises from the almost arbitrary fixpoint computations that can be expressed, and the complex interactions between nested fixpoints. Since Marrero [4] described a decision procedure making use of BDDs which involved explicitly calculating least-fixpoints for the temporal "until" eventualities, we consider whether this can be extended to the μ-calculus. We present a procedure which we believe decides the μ-calculus in EXPTIME.

One point to note in particular from Marrero is the way that $\mathbf{A}(\varphi \mathbf{U} \psi)$ was treated: not only did each **EX**-formula have to have a successor (like diamonds in **K**), but they had to have a successor which was consistent with the $\Box$-like nature of the least fixpoint being computed. This concept of considering otherwise-unrelated formulae together is also required for the μ-calculus.

Another thing to note is that the traditional fpm-method does not work for the μ-calculus, and instead automata are traditionally used. Unlike the other logics considered here, we attempt to give a more rigorous explanation, and also give the proofs we currently have. Specifically we believe that we have termination and soundness, but not yet completeness.

4.1 Syntax and semantics of the μ-calculus

Formulae of the μ-calculus are built from mutually disjoint sets of atomic formulae $Prop$, atomic actions Act and atomic variables Var, where $p \in Prop$, $X \in Var$ and $\pi \in Act$ via:

$$\varphi ::= p \mid \neg p \mid X \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \mu X.\varphi \mid \nu X.\varphi \mid [\pi]\varphi \mid \langle \pi \rangle \varphi$$

Models of μ-calculus formulae are structures $\mathcal{M} = (W, \{R_i\}, \rho)$. Given a valuation $\vartheta : Var \to 2^W$ on variables, denotations with respect to a model $(W, \{R_i\}, \rho)$ are defined via [1]:

$$[\![p]\!]_\vartheta = \rho(p) \qquad [\![\neg p]\!]_\vartheta = W \setminus \rho(p) \qquad [\![X]\!]_\vartheta = \vartheta(X)$$

$$[\![\varphi \wedge \psi]\!]_\vartheta = [\![\varphi]\!]_\vartheta \cap [\![\psi]\!]_\vartheta \qquad [\![\varphi \vee \psi]\!]_\vartheta = [\![\varphi]\!]_\vartheta \cup [\![\psi]\!]_\vartheta$$

$$[\![\mu X.\varphi]\!]_\vartheta = \bigcap \{S \subseteq W \mid S \supseteq [\![\varphi]\!]_{\vartheta[X:=S]}\} \qquad [\![\nu X.\varphi]\!]_\vartheta = \bigcup \{S \subseteq W \mid S \subseteq [\![\varphi]\!]_{\vartheta[X:=S]}\}$$

$$[\![[\pi]\varphi]\!]_\vartheta = \{w \in W \mid \forall v.wR_\pi v \Rightarrow v \in [\![\varphi]\!]_\vartheta\} \qquad [\![\langle \pi \rangle \varphi]\!]_\vartheta = \{w \in W \mid \exists v.wR_\pi v \wedge v \in [\![\varphi]\!]_\vartheta\}$$

Note that $[\![\mu X.\varphi]\!]_\vartheta$ ($[\![\nu X.\varphi]\!]_\vartheta$) can be expressed as least (greatest) fixpoints of $\lambda A.[\![\varphi]\!]_{\vartheta[X:=A]}$.

We will work with closed formulae, and variables are required to be uniquely bound, so for $X \in Var \cap cl(\varphi_0)$ there is exactly one $\xi X.\psi \in cl(\varphi_0)$ where $\xi \in \{\mu, \nu\}$. Thus a variable is uniquely associated with a single fixpoint expression.

The questions we seek to answer are whether there exists a model $\mathcal{M}$ with a world w such that $w \in [\![\varphi]\!]_\emptyset$ (satisfiability), and whether there exists a model and world such that $w \notin [\![\varphi]\!]_\emptyset$ (falsifiability/validity). We solve both questions simultaneously by determining the set of all worlds "relevant to φ_0" in any model $\mathcal{M}$.

4.2 Defining denotations

In addition to the atoms used in multimodal **K**, we create atoms for fixpoints/variables of the μ-calculus. That is, given the set $cl(\varphi_0)$ of all subformulae of φ_0 and their negations (ensuring to rename variables as necessary to maintain unique bindings), we define:

$$Atoms(\varphi_0) = \{[\pi]\psi \mid [\pi]\psi \in cl(\varphi_0)\} \cup (Prop \cap cl(\varphi_0)) \cup \{\mu X.\psi \mid \mu X.\psi \in cl(\varphi_0)\}$$

Because least and greatest fixpoints are negation duals, we only add one to the set of atoms similarly to how only $\Box$-formulae are made atoms and $\Diamond$-formulae are computed. Because each variable is uniquely bound by exactly one fixpoint formula, we consider the worlds where X holds to be equivalent to the worlds where $\xi X.\psi$ holds. If computing $[\![\varphi_0]\!]_{\vartheta_0}$ eventually requires computing $[\![X]\!]_{\vartheta_n}$, then at some intermediate point it must be computing $[\![\xi X.\psi]\!]_{\vartheta_i}$. The value of $[\![\xi X.\psi]\!]_{\vartheta_i}$ is a fixed point Z such that $Z = [\![\psi]\!]_{\vartheta_i[X:=Z]}$. Thus $[\![X]\!]_{\vartheta_n} = Z$, so X and $\xi X.\psi$ have the same denotation, and we refer to the atom as X or $\xi X.\psi$ interchangeably.

Thus, we define the following:

$$\begin{aligned}
[\![a]\!] &= \{w \in \mathcal{W} \mid w \in [\![a]\!]_\vartheta\} && \text{where } a \in Atoms(\varphi_0) \text{ and } \vartheta(X) = [\![\xi X.\psi]\!]_\vartheta \\
[\![X]\!] &= [\![\xi X.\psi]\!] && \text{where } X \text{ is uniquely bound by } \xi X.\psi \\
[\![\nu X.\psi]\!] &= \neg[\![\mu Y.\phi]\!] && \text{where } \mu Y.\phi \text{ is the negation dual} \\
[\![\neg p]\!] &= \neg[\![p]\!] \\
[\![\langle\pi\rangle\psi]\!] &= \neg[\![[\pi]\neg\psi]\!] \\
[\![\psi_1 \wedge \psi_2]\!] &= [\![\psi_1]\!] \wedge [\![\psi_2]\!] \\
[\![\psi_1 \vee \psi_2]\!] &= [\![\psi_1]\!] \vee [\![\psi_2]\!]
\end{aligned}$$

It is important to note that $[\![\psi]\!]$ and $[\![\psi]\!]_\vartheta$ have different meanings: $[\![\psi]\!]$ is something that we construct, and we eventually want it to correspond to the semantic notion of $[\![\psi]\!]_\vartheta$, but this is not the case yet.

The R_π relations are constructed in the same manner as for multimodal **K**, but it is much more important to note that R_π is an over-approximation here. Because we now have variables, the denotation for $[\pi]X$ is not fixed, so while the R_π we construct here will be useful, it does not capture the entire semantics of $\Box$-formulae now.

4.3 Enforcing semantics

As with the other logics, we now want to construct a fixpoint formula that enforces the model-theoretic semantics. The component of the fixpoint formula dealing with $\langle\pi\rangle\psi$ formula is the same as for multimodal **K**, so the remaining consideration is the fixpoints.

Instead of a shallow "local" evaluation, such as used for the limited eventualities in **CTL**, because the fixpoint formulae expressible in the μ-calculus are almost arbitrary, we inspect the formula deeply to compute the appropriate denotation.

For each least (greatest) fixpoint $\xi X.\psi$ in the closure we use the fixpoint semantics of the logic, rather than the infinite intersection / union, by calculating $\lambda A.[\![\psi]\!]_{[X:=A]}$: the denotation of ψ given that the denotation of X is A. This involves computing nested fixpoints and dealing with modalities as well. Diamond-formulae are simple, as the pre-image of the denotation of the successor world can be computed.

However, box-formulae in the fixpoint are not as simple as negating and treating as diamonds. A world w satisfying $\langle\pi\rangle\psi$ at some intermediate fixpoint valuation is interpreted as "w can have a successor satisfying ψ", which means that the negation or complement of this set is interpreted as "w cannot have a successor satisfying ψ". At intermediate stages this

interpretation can be overly restrictive, and we should instead consider "it is possible for w to have 0 or more successors, all of which falsify ψ".

For example, consider the fixpoint $\mu X.[\pi]X$ in a closure $\{X, [\pi]X, \langle\pi\rangle q, q\}$. At some stage when computing worlds where this least fixpoint X holds, we might consider worlds $w = \{X, [\pi]X, \langle\pi\rangle q\}$, $u = \{X, [\pi]X, [\pi]\neg q, q\}$, $v = \{X, [\pi]X, \langle\pi\rangle q, q\}$. According to R_π, we have $R_\pi(w,u), R_\pi(w,v), R_\pi(w,w), R_\pi(u,w), R_\pi(v,w), R_\pi(v,u)$ and $R_\pi(v,v)$. Suppose that at some iteration of the least fixpoint, u is found to be in the fixpoint, but w and v are not. If we compute $\neg\exists V'.R_\pi(V,V') \wedge S(V') \wedge \neg X(V')$, as $[\pi]X \equiv \neg\langle\pi\rangle\neg X$, then w and v will be excluded, because $R_\pi(w,v)$ and $R_\pi(v,w)$. However, it is possible to construct a model with a world w and without the edge $R_\pi(w,v)$, so this result is incorrect.

In order to correct this, we use something like the following formula:

$$[\![[\pi]\psi]\!] \wedge \bigwedge_{\langle\pi\rangle\chi \in cl(\varphi_0)} [\![\langle\pi\rangle\chi]\!] \Rightarrow \exists V'.R_\pi(V,V') \wedge \mathcal{V}(\chi)' \wedge G([\pi]\psi)$$

Here $\mathcal{V}(\chi)$ deeply expands χ according to the intuitions here and above, and $G([\pi]\psi)$ is a term to account for boxes being true simultaneously.

The intuition behind this formula is that if $[\pi]\psi$ holds at a world, for that world to be acceptable then all its existentials must be satisfiable in a way that is consistent with the box: If $\langle\pi\rangle\chi$ is true, then there is an R_π successor where χ is true (the $\Diamond$-formula is satisfied) and this is consistent with the boxes that are true (the G term).

Before going into specifics of what G is, note that as-is the formula can have an infinite loop: when considering the formula $\langle\pi\rangle[\pi]X$, the $\Box$-formula will recurse on the $\Diamond$-formula which will refer once again to the $\Box$-formula. We resolve this by introducing another fixpoint formula, such that any fixpoint of the formula gives a consistent denotation for $[\pi]\psi$. Then the greatest fixpoint of this formula contains all fixpoints, and thus the greatest fixpoint of the formula gives a maximal denotation for $[\pi]\psi$. This requires some changes elsewhere, which we address after presenting the expansion as a whole.

The G term in the formula is intended to capture the restriction of boxes, in much the same way as the constructed R_π relation. The difference is that it once again deeply expands the formulae and considers the current set of assumed denotations, both for fixpoint variables and additionally for $\Box$-formulae.

To bring this all together, we define a function $\mathcal{V}(\psi, S, \sigma_{var}, \sigma_\Box)$ which performs the deep-analysis of ψ given that all worlds must be in S, some variables have denotations given by σ_{var}, and some $\Box$-formulae have denotations given by $\sigma_\Box$, shown in Figure 1.

Note that when a new fixpoint is encountered, the assignments to $\Box$-formulae are forgotten during that calculation, since the assignments to variables changing can potentially change the denotation of a $\Box$. For example $[\pi]X$ may have some current denotation including worlds with successors satisfying X, but then X is assigned the empty denotation, meaning that $[\pi]X$ can only be true at worlds with no π-successors.

Finally, we bring this all together for the greatest fixpoint formula as follows:

$$\begin{aligned} good(S) = S \wedge & \bigwedge_{\langle\pi\rangle\psi \in cl(\varphi_0)} [\![\langle\pi\rangle\psi]\!] \Rightarrow \mathcal{V}(\langle\pi\rangle\psi, S, \emptyset, \emptyset) \\ \wedge & \bigwedge_{\mu Z.\psi \in cl(\varphi_0)} [\![\mu Z.\psi]\!] \Rightarrow \mathcal{V}(\mu Z.\psi, S, \emptyset, \emptyset) \\ \wedge & \bigwedge_{\nu Z.\psi \in cl(\varphi_0)} [\![\nu Z.\psi]\!] \Rightarrow \mathcal{V}(\nu Z.\psi, S, \emptyset, \emptyset) \end{aligned}$$

$$
\begin{aligned}
\mathcal{V}(p, S, \sigma_{var}, \sigma_{\Box}) &= [\![p]\!] \wedge S \\
\mathcal{V}(\neg p, S, \sigma_{var}, \sigma_{\Box}) &= \neg[\![p]\!] \wedge S \\
\mathcal{V}(X, S, \sigma_{var}, \sigma_{\Box}) &= \begin{cases} \sigma_{var}(X) \wedge S & \text{if } X \in \sigma_{var} \\ [\![X]\!] \wedge S & \text{otherwise} \end{cases} \\
\mathcal{V}(\psi_1 \wedge \psi_2, S, \sigma_{var}, \sigma_{\Box}) &= \mathcal{V}(\psi_1, S, \sigma_{var}, \sigma_{\Box}) \wedge \mathcal{V}(\psi_2, S, \sigma_{var}, \sigma_{\Box}) \\
\mathcal{V}(\psi_1 \vee \psi_2, S, \sigma_{var}, \sigma_{\Box}) &= \mathcal{V}(\psi_1, S, \sigma_{var}, \sigma_{\Box}) \vee \mathcal{V}(\psi_2, S, \sigma_{var}, \sigma_{\Box}) \\
\mathcal{V}(\mu X.\psi, S, \sigma_{var}, \sigma_{\Box}) &= \begin{cases} \sigma_{var}(X) \wedge S & \text{if } X \in \sigma_{var} \\ LFP(\lambda A.\mathcal{V}(\psi, S, \sigma_{var}[X := A], \emptyset)) & \text{otherwise} \end{cases} \\
\mathcal{V}(\nu X.\psi, S, \sigma_{var}, \sigma_{\Box}) &= \begin{cases} \sigma_{var}(X) \wedge S & \text{if } X \in \sigma_{var} \\ GFP(\lambda A.\mathcal{V}(\psi, S, \sigma_{var}[X := A], \emptyset)) & \text{otherwise} \end{cases} \\
\mathcal{V}(\langle\pi\rangle\psi, S, \sigma_{var}, \sigma_{\Box}) &= S \wedge \exists V'.R_\pi(V, V') \wedge \mathcal{V}(\psi, S, \sigma_{var}, \sigma_{\Box})' \\
\mathcal{V}([\pi]\psi, S, \sigma_{var}, \sigma_{\Box}) &= \begin{cases} \sigma_{\Box}([\pi]\psi) \wedge S & \text{if } [\pi]\psi \in \sigma_{\Box} \\ GFP(\lambda A.S \wedge [\![[\pi]\psi]\!] \wedge \bigwedge_{\langle\pi\rangle\chi \in cl(\varphi_0)} & \\ \qquad [\![\langle\pi\rangle\chi]\!] \Rightarrow \exists V'.R_\pi(V, V') \wedge G(A) & \text{otherwise} \\ \qquad\qquad \wedge \mathcal{V}(\chi, S, \sigma_{var}, \sigma_{\Box}[[\pi]\psi := A])) & \end{cases}
\end{aligned}
$$

where

$$
G(A) = \bigwedge_{[\pi]\phi \in Atoms(\varphi_0)} [\![[\pi]\phi]\!] \Rightarrow \mathcal{V}(\phi, S, \sigma_{var}, \sigma_{\Box}[[\pi]\psi := A])'
$$

Figure 1: The function to compute the denotation of a μ-calculus formula by deep-inspection.

In fact, the component dealing with $\Diamond$-formulae can also be written in the same manner as for **K**, but this more general statement is easier on the proofs.

4.4 Proofs

First we note that all fixpoints can be computed accurately by repeated iteration. This is a consequence of all the fixpoint formulae being monotone, and the Knaster-Tarski theorem.

We present a proof that the procedure described above is sound: If a formula is falsifiable, then the procedure will find a witness.

We aim to prove that given a subset S of the filtration, any model $\mathcal{M} = (W, \{R_i\}, \rho)$ such that the filtration of W is a subset of S, and a world $w \in W$, if $w \in [\![\psi]\!]_\emptyset$ then the representative of w in the filtration is in $\mathcal{V}(\psi, S, \emptyset, \emptyset)$.

We do this by proving a stronger theorem:

Theorem 1. *Given a model $\mathcal{M} = (W, \{R_i\}, \rho)$, a subset S of $\mathcal{W}$, a partial map σ_{var} from fixpoint formulae to denotations, and a partial map $\sigma_{\Box}$ from $\Box$-formulae to denotations, if*

1. *the worlds of W are all represented in S; and*
2. *for each fixpoint variable $Z \in dom(\sigma_{var})$, $[\![\xi Z.\varphi]\!]_{\sigma_{var}} = \sigma_{var}(Z)$;*

3. *for each fixpoint variable* $Z \notin dom(\sigma_{var})$*, when* σ_{var} *is used as a valuation then* $[\![Z]\!]_{\sigma_{var}} = \sigma_{var}(Z) = [\![\xi Z.\varphi]\!]_{\sigma_{var}}$*; and*

4. *for each formula* $[\pi]\varphi \in dom(\sigma_\Box)$*,* $[\![[\pi]\varphi]\!]_{\sigma_{var}} = \sigma_\Box([\pi]\varphi)$

then for all $w \in W$*, if* $w \in [\![\psi]\!]_{\sigma_{var}}$ *then* $w \in \mathcal{V}(\psi, S, \sigma_{var}, \sigma_\Box)$.

We define an ordering on $(\psi, \sigma_{var}, \sigma_\Box) \in cl(\varphi_0) \times ((Var \cap cl(\varphi_0)) \times \mathcal{W}) \times (\{[\pi]\psi \mid cl(\varphi_0)\} \times \mathcal{W})$ as follows:

Definition 2. $(\psi^1, \sigma^1_{var}, \sigma^1_\Box) < (\psi^2, \sigma^2_{var}, \sigma^2_\Box)$ iff $\sigma^1_{var} \supset \sigma^2_{var}$ or $(\sigma^1_{var} = \sigma^2_{var}$ and $(\sigma^1_\Box \supset \sigma^2_\Box$ or $(\sigma^1_\Box = \sigma^2_\Box$ and ψ^1 is a strict subformula of ψ^2.)))

This ordering is well-founded because $\subset$ and strict subformulae are well-founded. The ordering also corresponds to the definition of the $\mathcal{V}$ function.

Proof. We make use of well-founded induction over this ordering

- $\mathcal{V}(p, S, \sigma_{var}, \sigma_\Box)$. From the definition of $[\![\cdot]\!]$ for atoms, if $w \in [\![p]\!]_{\sigma_{var}}$ then $w \in [\![p]\!]$. Also by assumption $w \in S$, so $w \in [\![p]\!] \cap S$.

- $\mathcal{V}(\neg p, S, \sigma_{var}, \sigma_\Box)$. From the semantics, if $w \in [\![\neg p]\!]_{\sigma_{var}}$ then $w \in \mathcal{M} \setminus [\![p]\!]_{\sigma_{var}} \subseteq \neg[\![p]\!]$. Also by assumption $w \in S$, so $w \in \neg[\![p]\!] \cap S$.

- $\mathcal{V}(Z, S, \sigma_{var}[Z := X], \sigma_\Box)$. By definition, $[\![Z]\!]_{\sigma_{var}[Z:=X]}$ must be X. Thus $w \in X$, and $w \in S$ by assumption, therefore $w \in X \cap S$.

- $\mathcal{V}([\pi]\psi, S, \sigma_{var}, \sigma_\Box[[\pi]\psi := X])$. By assumption 4, $w \in X$, and by assumption 1, $w \in S$. Thus $w \in X \cap S$.

- $\mathcal{V}(\mu Z.\psi, S, \sigma_{var}[Z := X], \sigma_\Box)$. By assumption 2, $w \in X$, and by assumption 1, $w \in S$. Thus $w \in X \cap S$

- $\mathcal{V}(\nu Z.\psi, S, \sigma_{var}[Z := X], \sigma_\Box)$. As above.

- $\mathcal{V}(X, S, \sigma_{var}, \sigma_\Box)$ when $X \notin dom(\sigma_{var})$. By assumption 3, since $w \in [\![X]\!]_{\sigma_{var}}$ we have that $w \in [\![\xi X.\psi]\!]_{\sigma_{var}}$. By the definition of $[\![\cdot]\!]$, this means that $w \in [\![\xi X.\psi]\!]$ or equivalently $w \in [\![X]\!]$. Since $w \in S$ by assumption, we therefore have $w \in [\![X]\!] \wedge S$ as required.

- $\mathcal{V}(\psi_1 \wedge \psi_2, S, \sigma_{var}, \sigma_\Box)$. If $w \in [\![\psi_1 \wedge \psi_2]\!]_{\sigma_{var}}$ then $w \in [\![\psi_1]\!]_{\sigma_{var}}$ and $w \in [\![\psi_2]\!]_{\sigma_{var}}$. By induction we therefore have $w \in \mathcal{V}(\psi_1, S, \sigma_{var}, \sigma_\Box)$ and $w \in \mathcal{V}(\psi_2, S, \sigma_{var}, \sigma_\Box)$, and thus w is in the intersection as required.

- $\mathcal{V}(\psi_1 \vee \psi_2, S, \sigma_{var}, \sigma_\Box)$. As for $\psi_1 \wedge \psi_2$.

- $\mathcal{V}(\langle\pi\rangle\psi, S, \sigma_{var}, \sigma_\Box)$. If $w \in [\![\langle\pi\rangle\psi]\!]_{\sigma_{var}}$ then there exists a $v \in \mathcal{M}$ such that $wR_\pi v$, and $v \in [\![\psi]\!]_{\sigma_{var}}$. By induction, such a v must be in $\mathcal{V}(\psi, S, \sigma_{var}, \sigma_\Box)$.

  Consider one component of the constructed R_π, say $[\![[\pi]\varphi]\!] \Rightarrow [\![\varphi]\!]'$. If $w \in [\![[\pi]\varphi]\!]_{\sigma_{var}}$ then $w \in [\![[\pi]\varphi]\!]$ by the definition of $[\![\cdot]\!]$. Additionally $v \in [\![\varphi]\!]_{\sigma_{var}}$ due to the semantics of $\Box$-formulae, so $v \in [\![\varphi]\!]$. Thus (w, v) is in that component. If $w \notin [\![[\pi]\varphi]\!]_\vartheta$ then we have $w \in \neg[\![[\pi]\varphi]\!]$ and thus (w, v) is in the component. Thus (w, v) is in each component of R_π, so it is in their intersection and $(w, v) \in R_\pi$.

  Together with the assumed $w \in S$, this means that
  $w \in S \wedge \exists V'.R_\pi(V, V') \wedge \mathcal{V}(\psi, S, \sigma_{var}, \sigma_\Box)'$ as required.

- $\mathcal{V}(\nu X.\psi, S, \sigma_{var}, \sigma_{\Box})$. Given that $w \in [\![\nu X.\psi]\!]_{\vartheta}$, the semantics require that $w \in [\![\psi]\!]_{\vartheta[X:=A]}$ for some $A \subseteq [\![\psi]\!]_{\vartheta[X:=A]}$. For such A, each $v \in A$, must of course satisfy $v \in [\![\psi]\!]_{\vartheta[X:=A]}$. Since formulae are restricted to being monotone with respect to variable assignments, A is a subset of some fixpoint, and the greatest fixpoint Z contains all fixpoints. Thus $w \in [\![\psi_{\vartheta[X:=Z]}]\!]$.

  By induction, for each $v \in Z$ we therefore have $v \in \mathcal{V}(\psi, S, \sigma_{var}[X := Z], \emptyset)$, and so Z is a fixed point of $\lambda A.\mathcal{V}(\psi, S, \sigma_{var}[X := A], \emptyset)$. Since the greatest fixpoints contains all fixpoints, w is in the greatest fixpoint, and thus $w \in \mathcal{V}(\nu X.\psi, S, \sigma_{var}, \sigma_{\Box})$.

- $\mathcal{V}(\mu X.\psi, S, \sigma_{var}, \sigma_{\Box})$. Given that $w \in [\![\mu X.\psi]\!]_{\sigma_{var}}$, the semantics require that $w \in \bigcap\{A \subseteq \mathcal{M} \mid [\![\psi]\!]_{\sigma_{var}[X:=A]} \subseteq A\}$.

  Take $Z = LFP(\lambda A.A \vee \mathcal{V}(\psi, S, \sigma_{var}[X := A], \emptyset))$. Because Z is a fixed point, $Z = \mathcal{V}(\psi, S, \sigma_{var}[X := Z], \emptyset)$. Consider any $v \in [\![\psi]\!]_{\sigma_{var}[X:=Z]}$. By induction,
  $v \in \mathcal{V}(\psi, S, \sigma_{var}[X := Z], \emptyset)$. This means that $[\![\psi]\!]_{\sigma_{var}[X:=Z]} \subseteq Z$ holds, and thus $w \in Z$, and thus $w \in \mathcal{V}(\mu X.\psi, S, \sigma_{var}, \sigma_{\Box})$ as required.

- $\mathcal{V}([\pi]\psi, S, \sigma_{var}, \sigma_{\Box})$. when $[\pi]\psi \notin dom(\sigma_{\Box})$.

  We first show that

  $$w \in ([\![\langle\pi\rangle\chi]\!] \Rightarrow \exists V'.R_{\pi}(V, V') \wedge G([\![[\pi]\psi]\!]_{\sigma_{var}}) \wedge \mathcal{V}(\chi, S, \sigma_{var}, \sigma_{\Box}[[\pi]\psi := [\![[\pi]\psi]\!]_{\sigma_{var}}]))$$

  for each $\langle\pi\rangle\chi$. If $w \notin [\![\langle\pi\rangle\chi]\!]_{\sigma_{var}}$ then $w \in [\![[\pi]\neg\psi_1]\!]_{\sigma_{var}}$ and so $w \in [\![[\pi]\neg\psi_1]\!] = \neg[\![\langle\pi\rangle\psi_1]\!]$.

  Otherwise, there must be some $v \in \mathcal{M}$ such that $wR_{\pi}v$ and $v \in [\![\chi]\!]_{\sigma_{var}}$. By induction, $v \in \mathcal{V})(\chi, S, \sigma_{var}, \sigma_{\Box}[[\pi]\psi := [\![[\pi]\psi]\!]_{\sigma_{var}}])$ since condition 4 holds by definition. Using the same method as we did for $\langle\pi\rangle\psi$ we have that $(w, v) \in R_{\pi}$.

  We must show that (w, v) satisfies each of the conjuncts of $G([\![[\pi]\psi]\!]_{\sigma_{var}})$. If $w \notin [\![[\pi]\phi]\!]_{\sigma_{var}}$ then $w \in \neg[\![[\pi]\phi]\!]$, and thus the pair satisfies the conjunct. Otherwise $v \in [\![\phi]\!]_{\sigma_{var}}$, so by induction $v \in \mathcal{V}(\phi, S, \sigma_{var}, \sigma_{\Box}[[\pi]\psi := [\![[\pi]\psi]\!]_{\sigma_{var}}])$, so (w, v) satisfies the conjunct.

  Thus we have shown that (w, v) must satisfy the existentially quantified formula, and thus w satisfies the existential quantification for each $\langle\pi\rangle\chi$ in the closure.

  Because $w \in [\![[\pi]\psi]\!]_{\sigma_{var}}$ we have $w \in [\![[\pi]\psi]\!]$, and by assumption we have $w \in S$. By generalising, we have $[\![[\pi]\psi]\!]_{\sigma_{var}} \subseteq f([\![[\pi]\psi]\!]_{\sigma_{var}})$ for the fixpoint expression we define, and thus it is a subset of the greatest fixpoint. Thus $w \in \mathcal{V}([\pi]\psi, S, \sigma_{var}, \sigma_{\Box})$ as required.

⊣

We can then apply this theorem to show that for any world w of any model $\mathcal{M}$ using only worlds in S, if $w \in [\![\psi]\!]_{\emptyset}$, then $w \in \mathcal{V}(\psi, S, \emptyset, \emptyset)$. Condition 1 of Theorem 1 is explicitly enforced, and conditions 2 and 4 hold vacuously. Condition 3 potentially restricts the valuations we consider, but this has no impact on closed formulae.

To show that this method is complete, an additional step is required: If $w \in S$, then $w \in good(S)$ for any world w in any model $\mathcal{M}$.

Proof. Given that $w \in S$, the first conjunct is trivially satisfied. For the remaining conjuncts, Suppose that $w \in [\![\psi]\!]_{\emptyset}$ for $\psi \in \{\langle\pi\rangle\psi_1, \mu X.\psi_1, \nu X.\psi_1\}$. By Theorem 1 we have $w \in \mathcal{V}(\psi, S, \emptyset, \emptyset)$ as required. Thus w is in each of the conjuncts, and so $w \in good(S)$ as required. ⊣

We now show that this procedure remains in EXPTIME.

Theorem 3. *The procedure described above can be computed in $O(2^{O(n)})$ time.*

Proof. Computing a BDD over a set of variables V takes time proportional to $2^{|V|}$. Since we have two variables per atom, and we have $O(N)$ atoms, each BDD formula can be computed in $O(2^{O(N)})$ time. Each fixpoint is computed by repeated iteration until the answer is unchanged. The result space of each fixpoint is $\mathcal{W}$ of size $2^{|Atoms(\varphi_0)|}$, so each fixpoint is computed in $O(2^{O(n)})$ iterations.

Consider the $\mathcal{V}$ function. Since S is fixed, we can associate each instance of the function with a tuple of formula, fixpoint denotations, and $\Box$-formula denotations. The number of formulae is polynomial in the size of the initial formula, and there are $O(N2^{|Atoms(\varphi_0)|})$ possible ways of assigning fixpoint denotations and $\Box$-formula denotations. This means that there are $O(2^{O(N)})$ different calls to $\mathcal{V}$ given S. There is the possibility that $\mathcal{V}$ may be called with the same arguments multiple times. However, since it is pure functional, if this is the case then the results can be cached and the exponential bound retained.

To complete the proof, observe that each call to $\mathcal{V}$ does at most an exponential amount of work, computing a BDD or calling $\mathcal{V}$ at most exponentially many times, and the outermost greatest fixpoint formula calls interpret a linear number of times each iteration, thus the procedure takes at most $O(2^N)$ time. ⊣

5 Further work

μ-calculus. We have yet to prove that the method we describe for the μ-calculus is sound: in theory the fixpoint computed may contain representatives for potential worlds which do not appear in any model, and thus may find false counterexamples. One solution is to construct a model for each representative in the final fixpoint, and we are currently working on this.

Methodology for semantic constraints. Another area worth considering is whether it is possible to algorithmically construct BDDs to represent certain classes of first-order-definable conditions. Currently we have tried to explain our insights for particular frame conditions of interest (Section 3), but we do not have a mechanical translation from semantics to BDD conditions. This kind of construction is possible in tableau methods [6], so similar methods may allow for less human intuition in these BDD-based methods as well.

Substructural logics. When moving to substructural logics, instead of a binary Kripke relation, there is often a ternary relation of some sort. In much the same way as we represented a binary relation by having a single copy a' of each atom a, we can represent a ternary relation by having 2 copies a' and a'' of each atom a. The question then is whether we construct maximal ternary relations and use them in the same way that we used the maximal binary relations.

Many substructural logics are undecidable, and thus we won't be able to make a decision procedure. Nonetheless we do believe that some decidable substructural are amenable to this approach. We have started looking at a decidable fragment of separation logic, and believe it to be feasible, but do not have any results yet.

First order logic. One area that we have yet to consider is extending to first order logics. Once again many such logics are undecidable, but perhaps we can construct a semidecison procedure, or perhaps this approach could work for a decidable fragment of first order logic.

One of the first issues to consider is what set of atoms should be used. As soon as function symbols are introduced, there are potentially an infinite number of distinct objects, distinguishable by how many times the function symbol is applied. If we cannot set a fixed finite space to care about in the first place, then significant changes must follow. So far we have yet to find a way of handling this without essentially using a different automated reasoning technique.

BernaysSchönfinkel class. This class of first order logic requires that in prenex normal form, all existential quantifiers occur before any universal quantifier, and there are no function symbols. Equivalently, the skolem form of the formulae contains only nullary functors/constants. This restricted setting is known to be decidable.

We considered treating the constants as nominals, and predicates as relations or propositions true of a world. However what should the closure be? If the closure includes the negation of the input formula, then the closure includes formulae which are not in the Bernays-Schönfinkel class, and will in general include existential statements.

BDDs for other methods. Given that the BDD-based decision procedures for **CTL** and **Int** were competitive, we are also considering whether other automated reasoning methods could benefit from using BDDs. In particular, we are implementing a tableau procedure using BDDs. Potential benefits include fast equality checks; simple unsat caching by constructing a BDD of known-bad formulae and restricting the tableau nodes considered to the complement of that; and fast saturation phases.

References

[1] Julian Bradfield and Colin Stirling. Modal mu-calculi. In *The Handbook of Modal Logic*, pages 721–756. Elsevier, 2006.

[2] R. Goré, J. Thomson, and F. Widmann. An experimental comparison of theorem provers for CTL. In *Temporal Representation and Reasoning (TIME), 2011 Eighteenth International Symposium on*, pages 49 –56, sept. 2011. doi: 10.1109/TIME.2011.16.

[3] Rajeev Goré and Jimmy Thomson. BDD-based automated reasoning for propositional bi-intuitionistic tense logics. In *IJCAR*, 2012, to appear.

[4] Will Marrero. Using BDDs to decide CTL. *Lecture Notes in Computer Science*, 3440/2005: 222–236, 2005.

[5] Guoqiang Pan, Ulrike Sattler, and Moshe Y. Vardi. BDD-based decision procedures for the modal logic K. *Journal of Applied Non-classical Logics*, 49, 2005.

[6] R. A. Schmidt and D. Tishkovsky. Automated synthesis of tableau calculi. *Logical Methods in Computer Science*, 7(2):1–32, 2011. doi: http://dx.doi.org/10.2168/LMCS-7(2:6)2011.

[7] Alfred Tarski. A lattice-theoretical fixpoint theorem and its applications. *Pacific journal of Mathematics*, 5(4):285–309, 1955.