BDD-Based Automated Reasoning for Propositional Bi-Intuitionistic Tense Logics

Rajeev Goré and Jimmy Thomson

Logic and Computation Group,
Research School of Computer Science, The Australian National University

Abstract. We give Binary Decision Diagram (BDD) based methods for deciding validity and satisfiability of propositional Intuitionistic Logic **Int** and Bi-intuitionistic Tense Logic **BiKt**. We handle intuitionistic implication and bi-intuitionistic exclusion by treating them as modalities, but the move to an intuitionistic basis requires careful analysis for handling the reflexivity, transitivity and antisymmetry of the underlying Kripke relation. **BiKt** requires a further extension to handle the interactions between the intuitionistic and modal binary relations, and their converses. We explain our methodology for using the Kripke semantics of these logics to constrain the underlying least and greatest fixpoint approaches of the finite model construction. With some optimisations this technique is competitive with the state of the art theorem provers for Intuitionistic Logic using the ILTP benchmark and randomly generated formulae.

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 using this finite model property (fmp).

At first sight, this "fmp 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, Pan et al. [12] and Marrero [9] have shown that Binary Decision Diagrams (BDDs) can be used to represent the required subsets efficiently, without actually "constructing" them explicitly for **K** and **CTL**.

We investigate the potential of this BDD-based method for Intuitionistic Propositional Logic (**Int**) and its extensions Bi-Intuitionistic Logic (**BiInt**) and Bi-Intuitionistic Tense Logic (**BiKt**). These logics introduce various complications over **K** and **CTL**: the logic **Int** has an intuitionistic rather than a classical basis; the logic **BiInt** has an operator whose semantics uses the converse of the Kripke binary relation; the logic **BiKt** has two binary relations $R_\Box$ and $R_\Diamond$ so that $\Box$ and $\Diamond$ are not De Morgan duals, has their converses to handle $\blacklozenge$ and $\blacksquare$ and has two further interaction conditions. A priori, it is not obvious how to

B. Gramlich, D. Miller, and U. Sattler (Eds.): IJCAR 2012, LNAI 7364, pp. 301–315, 2012.

$w \not\Vdash \bot$		$w \Vdash p$	iff $\rho(w,p) = t$
$w \Vdash \varphi \wedge \psi$	iff $w \Vdash \varphi$ and $w \Vdash \psi$	$w \Vdash \varphi \vee \psi$	iff $w \Vdash \varphi$ or $w \Vdash \psi$
$w \Vdash \varphi \to \psi$	iff $\forall v \sqsupseteq w.v \not\Vdash \varphi$ or $v \Vdash \psi$	$w \Vdash \varphi \prec \psi$	iff $\exists v \sqsubseteq w.v \Vdash \varphi$ and $v \not\Vdash \psi$
$w \Vdash \Diamond\varphi$	iff $\exists v.wR_\Diamond v$ and $v \Vdash \varphi$	$w \Vdash \blacklozenge\varphi$	iff $\exists v.wR_\Box^{-1}v$ and $v \Vdash \varphi$
$w \Vdash \Box\varphi$	iff $\forall z\forall v.w \sqsubseteq zR_\Box v \Rightarrow v \Vdash \varphi$	$w \Vdash \blacksquare\varphi$	iff $\forall z\forall v.w \sqsubseteq zR_\Diamond^{-1}v \Rightarrow v \Vdash \varphi$

Fig. 1. Kripke Semantics for **BiKt** in model $\mathcal{M} = (W, \sqsubseteq, R_\Box, R_\Diamond, \rho)$ and $w \in W$

handle all of these complications using the BDD method, and indeed, we find that the least fixpoint approach for BDDs does not work for all of our logics.

We show how to adapt the BDD-method to **Int**, extend it to **BiInt** and **BiKt**, and describe some useful optimisations. We also compare our implementation with the state of the art theorem provers for **Int** (`PITP` [1] and `Imogen` [11]), and `DBiKt` [15], the only theorem prover for **BiKt** that we are aware of.

Our results show that with the help of some optimisations, this method is competitive with state-of-the-art theorem provers for **Int**, and still works well for some of its tense extensions. Its biggest advantage is its versatility.

1.1 Related Work

Current state of the art theorem provers for **Int** are based on an optimised tableau method [1] or a heuristically guided, focused, polarised, inverse method [11]. Pointers to other theorem provers for **Int** can be found on the ILTP Benchmark website [16]: most of them are based upon tableaux or sequent calculi.

Various sequent calculi for **BiInt** exist [4, 6, 13, 14, 15]. Some of them allow backward proof-search, and some have been extended to handle **BiKt** [7]. However, we know of only one implementation for both of these logics [15].

Pan et al. [12] give a BDD-based algorithm for deciding **K**, the simplest propositional classical normal modal logic. They show how to handle a single binary relation using BDDs, but do not need to consider multiple interacting "converse" relations, nor further frame conditions like reflexivity, transitivity and anti-symmetry, as we do. They also experiment with some potential optimisations, some of which are not limited to **K**. We make use of some of these optimisations, as well as describing some new optimisations appropriate for **Int**.

Marrero [9] gives a BDD-based algorithm for deciding computation tree logic **CTL**, a propositional modal temporal logic with fixpoints. He provides a way of handling the transitive closure of a discrete and serial relation by explicitly calculating a least fixpoint, which he uses to deal with eventualities. For our logics, the relation itself is required to be transitive, so we use a different method.

2 Syntax and Semantics of Bi-Intuitionistic Tense Logics

Formulae of **BiKt** [7] are defined from a set *Prp* of primitive propositions as:

$$\varphi ::= p \in Prp \mid \bot \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \varphi \to \varphi \mid \varphi \prec \varphi \mid \Diamond\varphi \mid \Box\varphi \mid \blacklozenge\varphi \mid \blacksquare\varphi$$

Models of **BiKt** are structures $\mathcal{M} = (W, \sqsubseteq, R_\Box, R_\Diamond, \rho)$ where W is a non-empty set of worlds; $\sqsubseteq$ is a reflexive, transitive and antisymmetric binary relation on W; both $R_\Box$ and $R_\Diamond$ are binary relations on W satisfying the "zig-zag" frame conditions $(F1)$ and $(F2)$ below; and $\rho : W \times Prp \mapsto \{t, f\}$ is a valuation which obeys the persistence property:

$(F1)$:	If $x \sqsubseteq y$ and $xR_\Diamond z$ then $\exists w.yR_\Diamond w$ and $z \sqsubseteq w$
$(F2)$:	If $xR_\Box y$ and $y \sqsubseteq z$ then $\exists w.x \sqsubseteq w$ and $wR_\Box z$
Persistence:	If $\rho(w, p) = t$ and $w \sqsubseteq v$ then $\rho(v, p) = t$.

Given $\mathcal{M} = (W, \sqsubseteq, R_\Box, R_\Diamond, \rho)$ and $w \in W$, the semantics of **BiKt** are given in Figure 1. We use $\to$ for intuitionistic implication while we use $\Rightarrow$ for classical implication in the meta-logic. We define intuitionistic negation $\neg\varphi$ as $\varphi \to \bot$. Note that $\Box$ and $\Diamond$, and $\blacksquare$ and $\blacklozenge$, are not de Morgan duals via negation.

BiInt [18] is the $\{\wedge, \vee, \to, -\!\!<, \bot\}$-fragment of **BiKt** and models of **BiInt** thus do not need the $R_\Box$ and $R_\Diamond$ relations. **Int** is **BiInt** without $-\!\!<$ -formulae.

A formula φ is **L**-valid if for all **L**-models $\mathcal{M}$, and for all worlds $w \in \mathcal{M}$ we have $\mathcal{M}, w \Vdash \varphi$. Dually, φ is **L**-satisfiable if there is some **L**-model $\mathcal{M}$ with some world $w \in \mathcal{M}$ such that $\mathcal{M}, w \Vdash \varphi$. A formula is **L**-falsifiable iff it is not **L**-valid. We define global logical consequence for **BiKt** and fragments as follows where $\mathcal{M} = (W, \sqsubseteq, R_\Box, R_\Diamond, \rho)$ and Γ is a finite set of "global assumptions":

$$\Gamma \models \varphi \text{ iff } \forall \mathcal{M}.\ (\forall w \in W. \mathcal{M}, w \Vdash \Gamma) \Rightarrow \forall w \in W.\ \mathcal{M}, w \Vdash \varphi.$$

3 A BDD Perspective of the Finite Model Method

For each of our logics, our goal is to construct a finite model $\mathcal{M} = (W_f, \preceq_f, R_\Box, R_\Diamond, \rho)$, as appropriate, similar to Pan et al., by constructing a sequence of frames $(W_0, \preceq_0, R^0_\Box, R^0_\Diamond)$, $(W_1, \preceq_1, R^1_\Box, R^1_\Diamond)$, ..., $(W_f, \preceq_f, R^f_\Box, R^f_\Diamond)$ such that the final frame gives a model which is "canonical" in two senses: if φ_0 is satisfiable (falsifiable) then some world of W_f satisfies (falsifies) φ_0. Given such a finite "canonical" model, we can decide whether a given φ_0 is satisfiable or valid by checking whether such worlds exist.

For a given formula φ_0, and a closure $cl(\varphi_0)$ we first define a set of atoms $Atm \subseteq cl(\varphi_0)$, as appropriate for the logic. Each subset of Atm is a classical (bi-valent) valuation on these atoms, where membership means truth-hood. The set of potentially good worlds $\mathcal{W} = 2^{Atm}$ is thus an upper bound on each W_i above, and the binary relation $\mathcal{W} \times \mathcal{W}$ is an upper bound on each $\preceq_i$.

We next use the Kripke semantics to extract necessary constraints to construct a relation $\preceq_{\max} \subseteq \mathcal{W} \times \mathcal{W}$ that is maximal in that it throws out only the edges which break these constraints. We then monotonically refine an initial approximation W_0 towards W_f, using the constructed $\preceq_{\max}$ relation to enforce the correct modal interpretation of the elements of $cl(\varphi_0)$ in all the worlds. Once W_f has been computed, the final step is to determine which, if any, worlds in W_f satisfy and falsify φ_0, giving the satisfiability and validity of φ_0.

3.1 A Better Basis for W_f

For our logics, $cl(\varphi_0) = sub(\varphi_0)$, the set of all subformulae of φ_0 including φ_0. The naive way to construct W_f is simply to use the set of all subsets of $cl(\varphi_0)$. We instead use only the "sensible subsets" following Pan et al.'s "lean" representation and Marrero's choice of BDD variables. We represent each primitive proposition and implication from $cl(\varphi_0)$ as an explicit BDD-variable, and compute the "denotation" of an arbitrary formula from $cl(\varphi_0)$ as follows:

$$Atm = (Prp \cap sub(\varphi_0)) \cup \{\varphi \to \psi \mid \varphi \to \psi \in sub(\varphi_0)\}$$
$$\mathcal{W} = 2^{Atm} \quad [\![\bot]\!] = \emptyset \qquad\qquad [\![a]\!] = \{w \in \mathcal{W} \mid a \in w\}$$
$$[\![\phi \land \psi]\!] = [\![\phi]\!] \cap [\![\psi]\!] \qquad\qquad [\![\phi \lor \psi]\!] = [\![\phi]\!] \cup [\![\psi]\!]$$

Thus $\mathcal{W}$ is finite, $w \in \mathcal{W}$ corresponds to a classical binary valuation on our BDD-variables, and for every $\psi \in cl(\varphi_0)$, the world w claims to satisfy ψ if $w \in [\![\psi]\!]$, and claims to falsify ψ if $w \in \overline{[\![\psi]\!]}$, where $\overline{[\![\psi]\!]} = \mathcal{W} \setminus [\![\psi]\!]$.

The set $\mathcal{W}$ is smaller than $2^{cl(\varphi_0)}$, and does not contains worlds which behave inappropriately with respect to conjunction and disjunction. We are thus left with worlds containing primitive propositions and implications. The semantics of an intuitionistic implication refers to $\sqsubseteq$. We therefore use an explicit representation $\preceq$ of the $\sqsubseteq$ relation as a finite set of ordered pairs from $\mathcal{W} \times \mathcal{W}$.

3.2 Constructing the Maximal $\preceq$ Relation

Our eventual goal is to construct a $W_f \subseteq \mathcal{W}$ and a binary relation $\preceq_f$ over W_f which obeys all of the semantic restrictions of intuitionistic models. We now show how to construct an over-approximation $\preceq_{\max}$ over $\mathcal{W}$ which is persistent, transitive and anti-symmetric, and which also obeys one half of the semantics of implication. These restrictions on the binary relation are not required for **K** or **CTL**, and so are not considered by Pan et al. [12] and Marrero [9].

Persistence. For any particular primitive proposition $p \in Atm$, the persistence condition can be expressed in terms of denotations as below:

$$\forall w, v \in \mathcal{W}.\ \ w \in [\![p]\!]\ \&\ w \preceq v \Rightarrow v \in [\![p]\!] \tag{1}$$

Alternatively, dropping universal quantifiers, we can write it as either of:

$$w \preceq v \Rightarrow w \in \overline{[\![p]\!]} \text{ or } v \in [\![p]\!] \qquad\qquad w \in [\![p]\!]\ \&\ v \in \overline{[\![p]\!]} \Rightarrow\ w \not\preceq v \tag{2}$$

The constraint obtained from (2) is expressed in terms of set notation as:

$$\preceq\ \subseteq (\overline{[\![p]\!]} \times \mathcal{W}) \cup (\mathcal{W} \times [\![p]\!]) \qquad\qquad ([\![p]\!] \times \mathcal{W}) \cap (\mathcal{W} \times \overline{[\![p]\!]}) \subseteq\ \not\preceq \tag{3}$$

That is, an upper bound on $\preceq$ is the set of ordered pairs from $\mathcal{W} \times \mathcal{W}$ where the first world is not in the denotation of p or the second is in the denotation of p. Alternately, a pair of worlds from $\mathcal{W} \times \mathcal{W}$ is forbidden from being in $\preceq$ if the first is in the denotation of p and the second is not. Taking the conjunction over all $p \in cl(\varphi_0)$ gives our final over-approximation from persistence:

$$\preceq \; \subseteq \bigcap_{p \in Prp \cap Atm} (\overline{[\![p]\!]} \times \mathcal{W}) \cup (\mathcal{W} \times [\![p]\!]) \tag{4}$$

Semantics of implication. Since $\sqsubseteq$ is transitive, if $w \sqsubseteq v$ then all successors of v are successors of w as well: thus if $\mathcal{M}, w \Vdash \phi \to \psi$, then $\mathcal{M}, v \Vdash \phi \to \psi$ and implications persist across $\sqsubseteq$. We mimic this by extending (4) from just the primitive propositions to all atoms ψ in Atm:

$$\preceq \; \subseteq \bigcap_{\psi \in Atm} (\overline{[\![\psi]\!]} \times \mathcal{W}) \cup (\mathcal{W} \times [\![\psi]\!]) \tag{5}$$

For any particular implication $\phi \to \psi$, the "only if" part of the semantics of implication can be expressed using denotations by dropping quantifiers as either:

$$w \in [\![\phi \to \psi]\!] \;\&\; w \preceq v \;\Rightarrow\; v \in \overline{[\![\phi]\!]} \cup [\![\psi]\!] \tag{6}$$

$$w \preceq v \;\Rightarrow\; w \in \overline{[\![\phi \to \psi]\!]} \text{ or } v \in \overline{[\![\phi]\!]} \cup [\![\psi]\!] \tag{7}$$

Just as (1) became (4), constraint (7) becomes the following in terms of sets:

$$\preceq \; \subseteq \bigcap_{\phi \to \psi \in Atm} (\overline{[\![\phi \to \psi]\!]} \times \mathcal{W}) \cup (\mathcal{W} \times (\overline{[\![\phi]\!]} \cup [\![\psi]\!])) \tag{8}$$

The conjunction of (5) and (8) gives $\preceq_{\max}$, an upper bound on $\preceq$, as:

$$\preceq_{\max} = RHS(5) \cap RHS(8) \tag{9}$$

Transitivity and Antisymmetry

Lemma 1 *The relation* $\preceq_{\max}$ *is transitive:* $(\preceq_{\max} \circ \preceq_{\max}) \subseteq \preceq_{\max}$.

Proof. For a contradiction, pick any $(x, y) \in \preceq_{\max} \circ \preceq_{\max}$ and suppose (x, y) fails the persistence condition (5): thus for some $\psi \in Atm$, we have $x \in [\![\psi]\!]$ and $y \in \overline{[\![\psi]\!]}$. By the definition of $\circ$ there must be some "midpoint" z such that $(x, z) \in \preceq_{\max}$ and $(z, y) \in \preceq_{\max}$. Since $x \preceq_{\max} z$ and $x \in [\![\psi]\!]$ we must have $z \in [\![\psi]\!]$ by persistence of $\preceq_{\max}$. Then $z \preceq_{\max} y$ gives $y \in [\![\psi]\!]$: contradiction.

Suppose then that (x, y) fails condition (8): thus $x \in [\![\phi \to \psi]\!]$ and $y \in [\![\phi]\!]$ and $y \in \overline{[\![\psi]\!]}$. As before, the midpoint $z \in [\![\phi \to \psi]\!]$. By (8), if $(z, y) \in \preceq_{\max}$ then $y \in \overline{[\![\phi]\!]} \cup [\![\psi]\!]$, but this again contradicts our earlier assumption. Thus any pair in $\preceq_{\max} \circ \preceq_{\max}$ must obey (9). So $\preceq_{\max}$ is transitively closed.

Lemma 2 *If* $w \preceq_{\max} v$ *and* $v \preceq_{\max} w$ *then* $w = v$*: thus* $\preceq_{\max}$ *is antisymmetric.*

Proof. Let $x \preceq_{\max} y$ and $y \preceq_{\max} x$. Suppose they differ on some atom a. If $a \in x$, then $a \in y$ by persistence, and vice-versa. Thus x and y cannot be distinct.

The relation $\preceq_{\max}$ may not be reflexive since $\mathcal{W}$ may contain a $w \in [\![\phi \to \psi]\!] \cap [\![\phi]\!] \cap \overline{[\![\psi]\!]}$, meaning that $(w, w) \notin \preceq_{\max}$.

3.3 Using $\preceq_{\max}$ to Construct W_f

We now have a set of "sensible" worlds $\mathcal{W}$ and an over-approximation $\preceq_{\max}$ of $\sqsubseteq$ that is persistent, transitive and anti-symmetric (but not necessarily reflexive). The structure $(\mathcal{W}, \preceq_{\max})$ may still contain "bad" worlds that do not obey the semantics: for example, a world $w \in \mathcal{W}$ with $w \in \overline{[\![\phi \to \psi]\!]}$ which lacks a $v \in [\![\phi]\!] \cap \overline{[\![\psi]\!]}$ with $w \preceq_{\max} v$. We can refine this structure into a model in two ways: by starting with $W_0 = \mathcal{W}$ as the set of all "potentially good" worlds and removing only "bad" worlds, or by starting with $W_0 = \emptyset$ as the set of all "known good" worlds and adding only "good" worlds. The greatest fixpoint of the first way, and the least fixpoint of the second way gives the W_f we seek.

At each stage, $\preceq_i$ is just the restriction of $\preceq_{\max}$ to W_i. These restrictions maintain persistence and antisymmetry of $\preceq_i$ because no new edges are added. We maintain transitivity ($x \preceq_i y$ & $y \preceq_i z \Rightarrow x \preceq_i z$) because each restriction removes worlds rather than edges, thus the only way we can lose an existing edge $x \preceq_i z$, is by removing x or z, whence we also lose $x \preceq_i y$ or $y \preceq_i z$.

Regaining Reflexivity. The restriction of $\preceq_{\max}$ to W is reflexive if the formula below left holds, so the maximal subset of $\mathcal{W}$ on which $\preceq_{\max}$ is reflexive is *Refl*:

$$\forall w \in W.\ w \preceq_{\max} w \qquad\qquad \mathit{Refl} = \{w \in \mathcal{W} \mid (w, w) \in \preceq_{\max}\}$$

Any $w \in \mathcal{W}$ such that $w \notin \mathit{Refl}$ is not permitted to be reflexive by our constraints on $\preceq_{\max}$, and thus must not appear in any model and so $W_f \subseteq \mathit{Refl}$.

Enforcing the Semantics of Implications. The remaining aspect of the semantics to consider is the (contra-position of the) "if" component of implication:

$$w \in \overline{[\![\phi \to \psi]\!]} \Rightarrow \exists v \in W_i.\ w \preceq v \ \&\ v \in [\![\phi]\!] \cap \overline{[\![\psi]\!]} \tag{10}$$

Given the current "potentially good" or "known good" worlds W_i, the potential witnesses $V_i^{\phi\to\psi}$ that falsify a particular $\phi \to \psi \in \mathit{Atm}$ are found by:

$$V_i^{\phi\to\psi} = W_i \cap [\![\phi]\!] \cap \overline{[\![\psi]\!]}.$$

From this, we can identify the worlds which can reach such a witness using the $\preceq_{\max}$ pre-image of $V_i^{\phi\to\psi}$, and complete the representation:

$$W_i^{\phi\to\psi} = [\![\phi \to \psi]\!] \cup \{x \mid \exists y \in V_i^{\phi\to\psi}\ .\ (x, y) \in \preceq_{\max}\} \tag{11}$$

The pre-image is found using existential quantification of BDDs, as described by Pan et al. and Marrero. We now show how to find the set W_f using fixpoints.

The GFP Method. We start with $W_0 = \mathcal{W}$, as the set of all "potentially good" worlds and prune bad worlds by computing the greatest fixpoint of:

$$W_{i+1} = W_i \cap \mathit{Refl}\ \cap \bigcap_{\phi\to\psi\in \mathit{Atm}} W_i^{\phi\to\psi} \tag{12}$$

Since *Refl* does not depend on W_i, we can instead just start with $W_0 = \mathit{Refl}$. Formula (12) is monotonic decreasing, and since $\mathcal{W}$ is finite, finding a fixpoint by repeated iteration is guaranteed to terminate in exponential time.

The LFP Method. We start with $W_0 = \emptyset$ and add only "good" worlds. Unlike **K**, where the least- and greatest-fixpoint iterated formulae are essentially the same, we must account for reflexivity at each iteration to handle the case where $x = y$ in (11). A solution to this is to explicitly allow for reflexivity:

$$W^+ = \overline{[\![\phi]\!]} \cup [\![\psi]\!] \qquad W_i^- = ([\![\phi]\!] \cap \overline{[\![\psi]\!]}) \cup \{x \mid \exists y \in V_i^{\phi\to\psi} \, . \, (x,y) \in \preceq_{\max}\}$$

$$W_{i+1} = W_i \cup \bigcap_{\phi\to\psi\in Atm} (\overline{[\![\phi \to \psi]\!]} \cap W_i^-]) \cup ([\![\phi \to \psi]\!] \cap W^+) \tag{13}$$

Here the formula for W^+ captures the worlds that satisfy $\phi \to \psi$ locally. The formula for W_i^- captures the worlds that falsify $\phi \to \psi$ locally, or by having some other successor which falsifies $\phi \to \psi$.

In the first iteration, $W_0 = \emptyset$, so $V_0^{\phi\to\psi} = \emptyset$ and $W_i^- = [\![\phi]\!] \cap \overline{[\![\psi]\!]}$. Thus

$$W_1 = \bigcap_{\phi\to\psi\in Atm} \left(\overline{[\![\phi \to \psi]\!]} \cap [\![\phi]\!] \cap \overline{[\![\psi]\!]}\right) \cup \left([\![\phi \to \psi]\!] \cap (\overline{[\![\phi]\!]} \cup [\![\psi]\!])\right)$$

That is, W_1 contains all worlds that satisfy/falsify all their implications locally. Then, W_2 will be the worlds which satisfy/falsify all their implications either locally or in the worlds of W_1, and so on. Since equation (13) is monotonically increasing, and $\mathcal{W}$ is finite, this least fixpoint computation terminates.

Deciding Satisfiability, Falsifiability and Validity. Once W_f is constructed, the model is $\mathcal{M}_f = (W_f, \preceq_f, \rho)$ where, for all $w \in W_f$, we put $\rho(w,p) = t$ iff $p \in w$. We can lift this valuation to $cl(\varphi_0)$ by showing that $\mathcal{M}_f, w \Vdash \psi$ iff $w \in [\![\psi]\!]$ for all $\psi \in cl(\varphi_0)$, giving us soundness.

For completeness, we have to show that the witness w_0 which satisfies or falsifies φ_0 in some model $\mathcal{M}$ is also represented in W_f. Since the fmp guarantees that only members of $cl(\varphi_0)$ are relevant, w_0 is represented by $w_0' \in \mathcal{W}$ as the subset $w_0' = \{\psi \in Atm \mid \mathcal{M}, w_0 \Vdash \psi\}$. For the greatest fixpoint method, $w_0' \in W_0 = \mathcal{W}$ and we prove that after all refinements, it is in W_f. For the least fixpoint method, $W_0 = \emptyset$ so we prove that w_0' is added to W_i, for some $i > 0$.

Theorem 3 *φ_0 is satisfiable iff $[\![\varphi_0]\!] \cap W_f \neq \emptyset$ and φ_0 is valid iff $\overline{[\![\varphi_0]\!]} \cap W_f = \emptyset$.*

Since we construct a representation of a model, we can relatively easily create a concrete example model illustrating satisfiability or falsifiability. But we deduce unsatisfiability and validity by the absence of certain worlds, so a convincing proof or "reason" for unsatisfiability or validity is more difficult to produce.

Global Assumptions. The greatest fixpoint method can easily handle global assumptions to decide whether $\Gamma \models \varphi_0$ by using $W_0 = [\![\Gamma]\!]$ instead of $W_0 = \mathcal{W}$, thus immediately considering only those worlds that satisfy Γ. For the least fixpoint method, we must assert the global assumptions Γ at each iteration.

3.4 Extension to Bi-Intuitionistic Logic BiInt

We now show that the greatest fixpoint BDD-method extends easily to handle converse, but the least fixpoint one does not. Our outline follows the methodology we set out for **Int**, but we no longer explicitly distinguish between $\preceq$, $\preceq_i$ and $\preceq_{\max}$. Strictly speaking, the same distinctions as for **Int** apply.

The denotation of $-\!\!<$ -formulae uses the (converse of the) semantic binary relation $\sqsubseteq$, so as for $\rightarrow$-formulae, we add all $-\!\!<$ -formulae from $cl(\varphi_0)$ to Atm.

The semantics of $-\!\!<$ are handled similarly to $\rightarrow$. Transitivity of the underlying relation $\sqsubseteq$ means that if $\mathcal{M}, w \Vdash \phi -\!\!< \psi$ and $w \sqsubseteq v$ then $\mathcal{M}, v \Vdash \phi -\!\!< \psi$, so exclusions persist. Thus we demand that all atoms still persist across $\preceq$.

The (contra-position of the) "if" component of the semantics of $\phi -\!\!< \psi$ is:

$$w \in \overline{[\![\phi -\!\!< \psi]\!]} \;\;\&\;\; w \preceq^{-1} v \quad\Rightarrow\quad v \in \overline{[\![\phi]\!]} \cup [\![\psi]\!] \tag{14}$$

Equation (14) transforms to a constraint on $\preceq^{-1}$ and hence $\preceq$:

$$\preceq^{-1} \;\subseteq\; ([\![\phi -\!\!< \psi]\!] \times \mathcal{W}) \cup (\mathcal{W} \times (\overline{[\![\phi]\!]} \cup [\![\psi]\!])) \tag{15}$$

$$\preceq \;\subseteq\; (\mathcal{W} \times [\![\phi -\!\!< \psi]\!]) \cup ((\overline{[\![\phi]\!]} \cup [\![\psi]\!]) \times \mathcal{W}) \tag{16}$$

The "only if" component of the semantics of exclusion is:

$$w \in [\![\phi -\!\!< \psi]\!] \Rightarrow \exists v.\; w \preceq^{-1} v \;\&\; v \in [\![\phi]\!] \cap \overline{[\![\psi]\!]} \tag{17}$$

We now have to modify the fixpoint formula. For greatest fixpoints, we first calculate the witnesses $V_i^{\phi -\!\!< \psi}$ to the existential of (17), as for implication earlier, and then determine the worlds $W_i^{\phi -\!\!< \psi}$ which reach the witness via $\preceq^{-1}$:

$$V_i^{\phi -\!\!< \psi} = W_i \cap [\![\phi]\!] \cap \overline{[\![\psi]\!]} \quad W_i^{\phi -\!\!< \psi} = \overline{[\![\phi -\!\!< \psi]\!]} \cup \{y \mid \exists x \in V_i^{\phi -\!\!< \psi}.\; (x, y) \in \preceq\}$$

The greatest fixpoint simply extends the one for **Int** with this new constraint:

$$W_{i+1} = W_i \;\cap\; \mathit{Refl} \;\cap \bigcap_{\phi\rightarrow\psi \in Atm} W_i^{\phi\rightarrow\psi} \;\cap \bigcap_{\phi -\!\!< \psi \in Atm} W_i^{\phi -\!\!< \psi}$$

On the other hand, it is not clear that there can be a least fixpoint approach for **BiInt**. For example, the formula $p \wedge (((p \rightarrow \bot) \rightarrow \bot) -\!\!< p)$ is satisfiable, but only in models containing a group of worlds which simultaneously require the existence of each other. Such worlds lead to a non-well-founded ordering on the inclusion of worlds in the least fixpoint, meaning that $W_0 = \emptyset$ does not suffice.

3.5 Extension to Bi-Intuitionistic Tense Logic BiKt

Moving from **BiInt** to **BiKt** presents more of a challenge. In addition to the 4 new connectives $\Box, \Diamond, \blacksquare$ and $\blacklozenge$, we must handle the two frame conditions $(F1)$ and $(F2)$. These conditions are difficult to capture directly as they refer to both the intuitionistic and modal relations and are existential in nature. However, their purpose is to ensure that truth persists over $\sqsubseteq$, and this is easier to use.

Theorem 4 (Persistence of BiKt) *For all* **BiKt** *models* $\mathcal{M}$*, if* $\mathcal{M}, w \Vdash \varphi$ *and* $w \sqsubseteq v$ *then* $\mathcal{M}, v \Vdash \varphi$*.*

The proof proceeds by induction on the size of φ and relies on $(F1)$ and $(F2)$ for the persistence of $\Diamond$- and $\blacklozenge$-formulae. Thus $(F1)$ and $(F2)$ cause persistence.

Suppose now that we have a structure which fails $(F1)$ or $(F2)$, but in which all formulae persist across $\sqsubseteq$. We can soundly add the missing $R_\Box$ or $R_\Diamond$ edges, without changing the satisfaction relation, to obtain a **BiKt**-model (which obeys $(F1)$ and $(F2)$) as encapsulated in the next theorem.

Theorem 5 *By adding* $R_\Diamond$ *and* $R_\Box$ *edges, a structure* $\mathcal{M}_1 = (W, \sqsubseteq, R^1_\Diamond, R^1_\Box, \rho)$ *which is persistent can be converted to a structure* $\mathcal{M}_n = (W, \sqsubseteq, R^n_\Diamond, R^n_\Box, \rho)$ *which satisfies* $(F1)$ *and* $(F2)$*, and such that* $\mathcal{M}_1, w \Vdash \varphi$ *iff* $\mathcal{M}_n, w \Vdash \varphi$*, for all* $w \in W$*.*

Thus, considering all persistent structures is sufficient. We must first extend Atm by adding all formulae with a main connective from $\{\Box, \Diamond, \blacklozenge, \blacksquare\}$ from $cl(\varphi_0)$. We can then enforce persistence as for **Int** and **BiInt** via (5).

Having handled the frame conditions, we handle the semantics for $\Diamond$ and $\blacklozenge$ using Pan et al.'s methods for **K**. The $\Diamond$-formulae impose a restriction on $R_\Diamond$, while the $\blacklozenge$-formulae impose a similar restriction on $R_\Box^{-1}$, and hence upon $R_\Box$:

$$R_\Diamond \subseteq (\llbracket \Diamond\psi \rrbracket \times \mathcal{W}) \cup (\mathcal{W} \times \overline{\llbracket \psi \rrbracket}) \tag{18}$$
$$R_\Box^{-1} \subseteq (\llbracket \blacklozenge\psi \rrbracket \times \mathcal{W}) \cup (\mathcal{W} \times \overline{\llbracket \psi \rrbracket}) \tag{19}$$

For the greatest fixpoint method, we also need:

$$W_i^{\Diamond\psi} = \overline{\llbracket \Diamond\psi \rrbracket} \cup \{x \mid \exists y \in (W_i \cap \llbracket \psi \rrbracket) \,.\, (x, y) \in R_\Diamond\}$$

The $\Box$- and $\blacksquare$-formulae are more complicated to represent since $R_\Box$ and $R_\Diamond$ interact with $\preceq$. The contra-positive of the “if” part of the semantics for $\Box$ is:

$$w \in \overline{\llbracket \Box\psi \rrbracket} \;\;\Rightarrow\;\; \exists z.\, w \preceq z \;\&\; \exists v.\, zR_\Box v \;\&\; v \in \overline{\llbracket \psi \rrbracket}$$

This has two existentials, which can be handled by computing two pre-images as follows. Let $Z_i^{\Box\psi} = \{z \mid \exists y \in (W_i \cap \overline{\llbracket \psi \rrbracket}) \,.\, (z, y) \in R_\Box\}$ and let

$$W_i^{\Box\psi} = \llbracket \Box\psi \rrbracket \cup \{x \mid \exists z \in (W_i \cap Z_i^{\Box\psi}) \,.\, (x, z) \in \preceq_{\max}\}$$

For the “only if” component, the interactions of the relations are more troublesome. But since $\preceq$ is required to be reflexive, the following are essential:

$$wR_\Box v \Rightarrow w \in \overline{\llbracket \Box\psi \rrbracket} \vee v \in \llbracket \psi \rrbracket \tag{20}$$
$$vR_\Diamond w \Rightarrow w \in \overline{\llbracket \blacksquare\psi \rrbracket} \vee v \in \llbracket \psi \rrbracket \tag{21}$$

Additionally, because (5) enforces persistence, if $u \in \llbracket \Box\psi \rrbracket$, then any w such that $u \preceq w$ must also satisfy $w \in \llbracket \Box\psi \rrbracket$. By induction this will force all $\preceq$-successors w of u to satisfy (20), and thus to satisfy the original semantics.

The constraints on $R_\Box$ are thus (19) and (20):

$$R_\Box \subseteq (\overline{\llbracket \Box\psi \rrbracket} \times \mathcal{W}) \cup (\mathcal{W} \times \llbracket \psi \rrbracket) \qquad (\mathcal{M}, w \Vdash \Box\psi \Rightarrow \mathcal{M}, v \Vdash \psi)$$
$$R_\Box \subseteq (\mathcal{W} \times \llbracket \blacklozenge\psi \rrbracket) \cup (\overline{\llbracket \psi \rrbracket} \times \mathcal{W}) \qquad (\mathcal{M}, w \Vdash \psi \Rightarrow \mathcal{M}, v \Vdash \blacklozenge\psi)$$

Similarly, the constraints on $R_\Diamond$ are (18) and (21):

$$R_\Diamond \subseteq (\llbracket \Diamond\psi \rrbracket \times \mathcal{W}) \cup (\mathcal{W} \times \overline{\llbracket \psi \rrbracket}) \qquad (\mathcal{M}, v \Vdash \psi \Rightarrow \mathcal{M}, w \Vdash \Diamond\psi)$$
$$R_\Diamond \subseteq (\mathcal{W} \times \overline{\llbracket \blacksquare\psi \rrbracket}) \cup (\llbracket \psi \rrbracket \times \mathcal{W}) \qquad (\mathcal{M}, v \Vdash \blacksquare\psi \Rightarrow \mathcal{M}, w \Vdash \psi)$$

To complete the decision procedure, the greatest fixpoint calculation is:

$$W_{i+1} = W_i \cap \mathit{Refl} \cap W_i^{\phi \to \psi} \cap W_i^{\phi \prec \psi} \cap W_i^{\Box\psi} \cap W_i^{\Diamond\psi} \cap W_i^{\blacklozenge\psi} \cap W_i^{\blacksquare\psi}$$

3.6 Optimisations

Variable ordering. The choice of variable ordering is critical when using ordered BDDs. We chose the following ordering after minimal experimentation since its preliminary results are encouraging. For each member of Atm whose main connective is non-classical (i.e. implication, exclusion, diamond or box), we do a pre-order traversal of the formation tree stopping at other members of Atm. The first time any member of Atm is encountered, it is appended to the current ordering. Pre-image computations require copying Atm, so the copy a' of a appears immediately after a in the ordering. Using an ordering which puts all copies at the end of the ordering is particularly bad.

Since determining the best ordering is a difficult problem in itself, BDD packages allow us to dynamically reorder the BDD variables. There is a trade-off between the quality of a reordering and the time taken to perform the reordering, so the main question is when to use this feature. For the greatest-fixpoint method, we provide an option which uses this feature once only to find a good ordering after computing W_0 using *Refl* and any global assumptions.

Normalisation. Another component of complexity is the number of BDD variables. We use techniques such as constant propagation ($\top \wedge \varphi = \varphi$, $\bot \to \varphi = \top$, etc.) to reduce formula size, and possibly reduce the number of atoms. We use syntactic equality to check whether two formulae are equivalent when determining the set of atoms. Normalising wrt an arbitrary fixed ordering $<$ on formulae improves the efficiency of this equality check. For example, putting $p < q$ collapses $\{(p \wedge q) \to \bot, (q \wedge p) \to \bot\}$ to $\{(p \wedge q) \to \bot\}$, requiring fewer atoms.

Early Termination. If we only want to check satisfiability or validity, then early termination is possible. In the greatest fixpoint approach, the sets W_i are strictly decreasing. If any $W_i \subseteq \llbracket \varphi_0 \rrbracket$ then $W_f \subseteq \llbracket \varphi_0 \rrbracket$, so φ_0 is valid, and if any $W_i \cap \llbracket \varphi_0 \rrbracket = \emptyset$ then $W_f \cap \llbracket \varphi_0 \rrbracket = \emptyset$, so φ_0 is unsatisfiable. In the least fixpoint approach, W_i is strictly increasing. If any $W_i \cap \llbracket \varphi_0 \rrbracket \neq \emptyset$ then $W_f \cap \llbracket \varphi_0 \rrbracket \neq \emptyset$, so φ_0 is satisfiable, and if any $W_i \cap \overline{\llbracket \varphi_0 \rrbracket} \neq \emptyset$ then $W_f \cap \overline{\llbracket \varphi_0 \rrbracket} \neq \emptyset$, so φ_0 is not valid.

Explicit representation of $\leftrightarrow$*.* Expanding bi-implications $\leftrightarrow$ into two implications can lead to an exponential blowup in the size of the formula. We therefore gave a direct semantics for $\leftrightarrow$ and added it to the set of atoms.

Explicit global assumptions. When determining **Int**-validity of a formula $\varphi_0 = (\gamma \rightarrow \varphi)$, any counterexample must make γ true in all states reachable from the root. The formula is valid iff all models where γ is true globally must make φ true. Thus, in the greatest fixpoint approach, we can start with $W_0 = [\![\gamma]\!]$, rather than $W_0 = \mathcal{W}$. By translating top-level implications to global assumptions in this manner, there are fewer atoms to consider, and the global assumptions may restrict the search space resulting in fewer iterations before reaching the fixpoint. This optimisation cannot be used for **BiInt** because $-\!\!<$ allows us to look "backwards" along $\sqsubseteq$, so γ cannot be turned into a global assumption.

4 Experimental Results

We used the ILTP propositional benchmarks [17, 16] and randomly generated formulae. All tests were performed on 32bit Ubuntu with a Core 2 Duo 3.0GHz processor, 3 GB RAM and a timeout of 600 seconds for each problem instance.

Benchmarks. The ILTP benchmarks consists of several categories of structured intuitionistic formulae. Some are "uninteresting" since they are easy for all provers. The remaining "interesting" benchmarks are split into 12 problem sets with 20 instances each, parametrised by a size n, consisting of zero or more axioms $\{\gamma_0, \cdots, \gamma_k\}$ and a single conjecture C giving $(\gamma_0 \wedge \cdots \wedge \gamma_k) \rightarrow C$.

The random benchmarks are generated to have a fixed number of symbols (treating $\neg\varphi = \varphi \rightarrow \bot$ as only one additional symbol) and a maximum ratio of distinct propositions to formula size. Formulae for **Int** use connectives $\wedge, \vee, \neg, \rightarrow$ and $\leftrightarrow$ while formulae for **BiKt** add in connectives $-\!\!<$, $\Box$, $\blacklozenge$, $\Diamond$ and $\blacksquare$. We used 1000 instances of each size from 10 through to 90 in steps of 5, which are available here: `http://users.cecs.anu.edu.au/~rpg/BDDBiKtProver/`

Theorem provers. According to the ILTP benchmark [16], the two best provers for propositional intuitionistic logic are `PITP`/`PITPINV`, and `Imogen`.

`PITP` and `PITPINV` [1] implement a signed tableau calculus to determine **Int** validity. The tableau rules are divided into 6 categories based on the branching factor and whether or not they are invertible. `PITPINV` attempts a non-invertible branch of one category before the invertible branch, while `PITP` attempts the invertible branch first. `PITP` and `PITPINV` are written in C++, and make use of optimisations such as dynamic formula simplification.

`Imogen` [11] uses a focused polarised inverse method to determine **Int** validity. Given a formula, `Imogen` performs a pre-processing step to assign polarities to each subformula. It then makes use of focusing based on the polarities to generate inference rules, and these rules are used (and extended) by the inverse method in a saturation phase to attempt to construct a sequent proving the original formula. Imogen is written in ML, and uses heuristics when assigning polarities to try to minimise the search space. When given 600 seconds, it tries one heuristic for 2 seconds and then, if needed, tries an alternate heuristic for 598 seconds.

DBiKt [15] is the only theorem prover we are aware of for **BiKt**. It uses a deep-inference nested sequent calculus for **BiKt** and is implemented in Java. It has not been heavily optimised, but is intended as a proof of concept.

BDDBiKt is our Ocaml theorem prover using the Buddy [3] BDD library. It is available here: http://users.cecs.anu.edu.au/~rpg/BDDBiKtProver/

Results. The numbers reported here for PITP differ from those on the ILTP website for two reasons. The first is that we use different hardware. The second is that formula SYN007+1.0014 expands to 4GB when converted to the input format of PITP, which did not allow it to be converted in-memory in the initial comparison. We instead write the formula to disk during the conversion, which allows the conversion to finish and hence allows PITP to solve it.

Our numbers for PITPINV differ from the ILTP website because we discovered a bug during the comparison on randomly generated formulae. The authors of PITPINV corrected the bug, and this has impacted its performance.

We analysed the impact of the implemented optimisations by testing the following versions of our own implementation:

- GFP: Greatest-fixpoint with early termination and explicit handling of $\leftrightarrow$
- Ga: GFP, with explicit global assumptions
- Gn: GFP, with normalisation.
- Gna: GFP with both explicit global assumptions and normalisation
- Gnar: Gna with dynamic variable reordering.
- LFP: Least-fixpoint with the same optimisations as GFP
- Ln: LFP with normalisation.

In Figure 2, "sum" is the sum out of the 240 "interesting" problems shown individually, while "total" is out of the whole 274 instance benchmark.

With all optimisations enabled, our BDD-method (Gnar) solved the second highest number of instances on the ILTP benchmark. Unlike the other theorem provers, when the BDD-method fails, it usually runs out of memory rather than time. Experiments on the same hardware with a 64bit OS and 8GB RAM showed that no instance caused Gnar to run out of memory and BDD times were improved, but only one additional problem was solved by Gnar, while Imogen performed notably slower. We now discuss the effects of each optimisation.

Explicit Assumptions. Converting top-level implications to explicit global assumptions has the largest impact. All of the benchmark formulae with axioms were helped by this optimisation, and some were trivialised by it.

Normalisation. Normalising the input formula was not as beneficial as explicit assumptions. In some cases it helps significantly: for example it rewrites SYJ206 to $\top$. In others it is detrimental because changing the formula structure changes our heuristically chosen BDD order into a worse one.

	GFP	Gn	Ga	Gna	Gnar	LFP	Ln	Imogen	PITP	PITPINV	Out of
SYJ201	6	6	20	20	20	6	6	20	20	20	20
SYJ202	12	10	12	10	8	12	10	8	10	10	20
SYJ203	17	18	20	20	20	18	19	20	20	20	20
SYJ204	17	19	20	20	20	18	20	20	20	20	20
SYJ205	14	14	14	19	19	12	12	20	20	11	20
SYJ206	15	20	15	20	20	19	20	20	20	20	20
SYJ207	6	6	20	20	20	7	7	20	7	8	20
SYJ208	7	7	9	10	8	9	9	19	20	20	20
SYJ209	17	18	20	20	20	18	19	20	10	10	20
SYJ210	17	18	20	20	20	19	20	20	20	20	20
SYJ211	6	6	9	8	15	9	9	20	20	9	20
SYJ212	13	20	13	20	20	18	20	20	20	20	20
sum of above	147	162	192	207	210	165	171	227	206	187	240
total over all	181	196	226	241	244	199	205	261	240	221	274

Fig. 2. Number of instances solved in the ILTP benchmark

Assumptions + Normalisation. Combining both optimisations works well on the whole. For class SYJ205, the formula is a conjunction of two semantically equivalent implications which are syntactically reordered. Normalisation combines the two implications into one, which is then converted into a global assumption. No assumptions can be made explicit without normalisation, and normalisation alone only removes one implication from the closure.

Dynamic Variable Reordering. Adding dynamic variable ordering is a mixed bag. In most cases its overhead is significant, while the benefits are small. For SYJ211 this is reversed, with reordering taking little time but giving significant improvement. We speculate that our relatively naive ordering performs reasonably well on most of the benchmarks, possibly because of the prevalence of lexicographically ordered sequences of propositional variables, so in general the dynamic ordering does not give a big benefit. However when the initial ordering is bad, the dynamic ordering can assist.

GFP vs LFP. LFP performs similarly to GFP in many cases, although it is not compatible with some of the helpful optimisations. The small differences between LFP and GFP arise from their different fixpoint formulae. LFP generally has fewer iterations, but each iteration is a more complex formula than the one used by GFP and thus takes longer to compute.

The results of random **Int** tell a different story. Now `Imogen` performs considerably worse than all other provers, failing to solve many cases. GFP scales reasonably, but is still significantly worse than `PITP`. Gnar is consistently slower than Gna, however at size 90 the non-reordering version runs out of memory on 7 formulae, while the reordering version times out on only 5. It seems that reordering may not help very often, but it makes the method more robust.

LFP does quite well, though not as well as `PITP`. The majority of the randomly generated formulae are invalid, so LFP can terminate early. Since each iteration is quite expensive, performing fewer iterations on the invalid formulae here gives a large benefit. On the valid formulae, it performs worse than GFP.

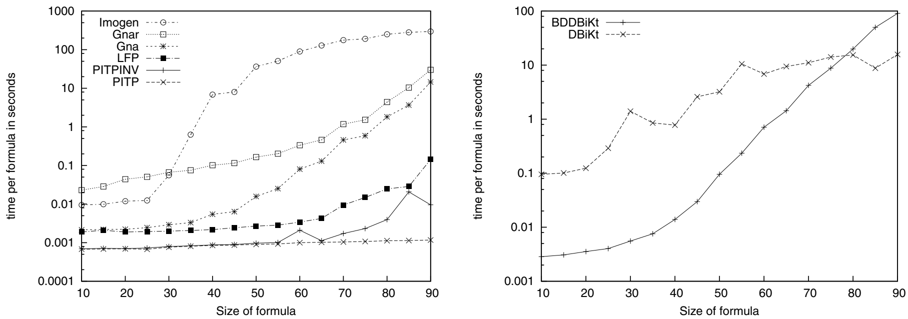

Fig. 3. Average time taken per random **Int** and **BiKt** instance with a timeout of 600s

The comparison with `DBiKt` for **BiKt** shows that each theorem prover can handle some formulae that the other could not. For sizes up to 75, GFP solved all instances, and did it faster than `DBiKt`. Past that point, `DBiKt` solved an increasing number of problems that GFP could not solve due to time or memory limits, and the time taken by GFP increased significantly above that of `DBiKt`. In general, all but 5 or so of the randomly generated formulae were invalid, but some of the few valid instances proved difficult for `DBiKt` and not GFP.

5 Conclusion and Further Work

Our optimised BDD-method `Gnar` for **Int** is competitive with the state-of-the-art provers `Imogen` and `PITP` in the following sense: on the ILTP benchmarks, it solves more problems than `PITP` but less problems than `Imogen`, and on randomly generated formulae, it performs better than `Imogen` but worse than `PITP`.

Unlike the other methods, BDD-methods are "memory hungry" so adding memory is likely to improve their relative performance. Indeed, moving from a 32bit OS to a 64bit OS gave a small improvement, but not as much as we hoped since the bottleneck just moved from memory to time.

To some extent, our implementation is naive, and further optimisations from the model checking community need to be investigated. In particular, we need to ascertain whether the BDD method is relatively brittle to variable ordering heuristics, or robust over many potential choices.

We are currently extending this method to handle all 15 basic modal logics obtained by combinations of reflexivity, transitivity, seriality, euclideaness, and symmetry, as well as to the modal mu-calculus. We are also extending the implementation to generate explicit (counter) models. A characterisation of when and how the method works would also be nice. Finally, can the BDD method be extended to predicate logics, possibly using instantiation-based methods?

The biggest advantage of the BDD-method is the ease with which it extends from **Int** to **BiInt** to **BiKt** compared to tableaux and inverse methods. For example, handling a "converse" operator to give **BiInt** using tableaux requires significant methodological extensions [2, 8]. Similarly, the inverse method has not been extended to handle "converse" as far as we know. McLaughlin and Pfenning [10] have implemented an inverse method for intuitionistic modal logics which do not require the complications of converse. We can handle these intuitionistic modal logics using our BDD-method for **BiKt** by just dropping $-\!\!<$, ■ and ⧫, and replacing $R_\Box$ and $R_\Diamond$ with a single modal relation R.

References

[1] Avellone, A., Fiorino, G., Moscato, U.: Optimization techniques for propositional intuitionistic logic and their implementation. Theoretical Computer Science 409(1), 41–58 (2008)
[2] Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds.): The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press (2003)
[3] Buddy (2011), `http://sourceforge.net/projects/buddy/`
[4] Buisman, L., Goré, R.: A Cut-Free Sequent Calculus for Bi-intuitionistic Logic. In: Olivetti, N. (ed.) TABLEAUX 2007. LNCS (LNAI), vol. 4548, pp. 90–106. Springer, Heidelberg (2007)
[5] Goré, R., Thomson, J., Widmann, F.: An experimental comparison of theorem provers for CTL. In: TIME 2011: Eighteenth International Symposium on Temporal Representation and Reasoning, pp. 49–56 (September 2011)
[6] Goré, R., Postniece, L., Tiu, A.: Cut-elimination and proof-search for bi-intuitionistic logic using nested sequents. In: AiML 2008, pp. 43–66 (2008)
[7] Goré, R., Postniece, L., Tiu, A.: Cut-elimination and proof search for bi-intuitionistic tense logic. In: Advances in Modal Logic, pp. 156–177 (2010)
[8] Goré, R., Widmann, F.: Sound Global State Caching for *ALC* with Inverse Roles. In: Giese, M., Waaler, A. (eds.) TABLEAUX 2009. LNCS, vol. 5607, pp. 205–219. Springer, Heidelberg (2009)
[9] Marrero, W.: Using BDDs to Decide CTL. In: Halbwachs, N., Zuck, L.D. (eds.) TACAS 2005. LNCS, vol. 3440, pp. 222–236. Springer, Heidelberg (2005)
[10] McLaughlin, S., Pfenning, F.: The focused constraint inverse method for intuitionistic modal logics (2010), (unpublished manuscript) (accessed January 31, 2012)
[11] McLaughlin, S., Pfenning, F.: Imogen: Focusing the Polarized Inverse Method for Intuitionistic Propositional Logic. In: Cervesato, I., Veith, H., Voronkov, A. (eds.) LPAR 2008. LNCS (LNAI), vol. 5330, pp. 174–181. Springer, Heidelberg (2008)
[12] Pan, G., Sattler, U., Vardi, M.Y.: BDD-based decision procedures for the modal logic K. Journal of Applied Non-classical Logics 49 (2005)
[13] Pinto, L., Uustalu, T.: Proof Search and Counter-Model Construction for Bi-intuitionistic Propositional Logic with Labelled Sequents. In: Giese, M., Waaler, A. (eds.) TABLEAUX 2009. LNCS, vol. 5607, pp. 295–309. Springer, Heidelberg (2009)
[14] Postniece, L.: Deep Inference in Bi-intuitionistic Logic. In: Ono, H., Kanazawa, M., de Queiroz, R. (eds.) WoLLIC 2009. LNCS, vol. 5514, pp. 320–334. Springer, Heidelberg (2009)
[15] Postniece, L.: Proof Theory and Proof Search of Bi-Intuitionistic and Tense Logic. Ph.D. thesis, Australian National University (2011)
[16] Raths, T., Otten, J.: The ILTP library (2007), `http://www.cs.uni-potsdam.de/ti/iltp/` (accessed January 2012)
[17] Raths, T., Otten, J., Kreitz, C.: The ILTP problem library for intuitionistic logic, release v1.1. Journal of Automated Reasoning (2006)
[18] Rauszer, C.: Applications of Kripke models to Heyting-Brouwer logic. Studia Logica 36, 61–71 (1977)