Department of Mathematics and Computer Science
Institute of Computer Science

Bachelor thesis

Systematic Verification of the Intuitionistic Modal Logic Cube in Isabelle/HOL

Hanna Lachnitt
hanna.lachnitt@fu-berlin.de

Submission date: 31.01.2017
Examiner: PD Dr. Christoph Benzmüller
Second examiner: Prof. Dr. Raúl Rojas
Supervisor: Maximilian Claus

Eidesstattliche Erklärung

Ich versichere hiermit an Eides Statt, dass diese Arbeit von niemand anderem als meiner Person verfasst worden ist. Alle verwendeten Hilfsmittel wie Berichte, Bücher, Internetseiten oder ähnliches sind im Literaturverzeichnis angegeben, Zitate aus fremden Arbeiten sind als solche kenntlich gemacht. Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungskommission vorgelegt und auch nicht veröffentlicht.

Berlin, den ______________________ ______________________

Hanna Lachnitt

Abstract

Due to its computational interpretation, there has been a lot of interest in intuitionistic logic in computer science. Adding combinations of the intuitionistic modal axioms to intuitionistic modal logic IK results in different systems. Together they consitute the intuitionistic modal logic cube. We use an embedding of intuitionistic modal logic in higher order logic to verify this cube. Automatic reasoning tools, such as Sledgehammer and Nitpick, were used to prove the inclusion relations between the cube's logics and the equality of some logics.

Contents

1 Introduction

On Intuitionistic Modal Logic Intuitionistic modal logics (IML) have been studied by various researchers [10, 19]. Mostly, they apply in theoretical computer science. In his dissertation Alex Simpson [19] names several application fields, for example staged computation, computational effects, security, distributed computation, and typed lambda calculi. Furthermore, he states that IML is considered to be a better language for describing security policies than classical logic by some researchers.

In classical logic a formula is always associated to one of the values $\bot$ or $\top$. In contrast to that in intuitionistic logic a formula is only true iff a proof exists for it. This does not mean that there is a third value but that the truth value is unknown until a proof or counterproof has been found. As one may have noticed this sounds very philosophical.

And in fact, the founder of intuitionism Luitzen Brouwer was convinced that mathematics is a creation of the mind. He reasoned that the law of the excluded middle ($A \lor \neg A$) should no longer be valid. Based on Brouwers idea, Arend Heyting developed the first model theory for intuitionistic logic. Later, Saul Kripke followed with the so called Kripke semantics. This is the semantics used in this paper and explained more thoroughly in section 3.

Basically, intuitionistic modal logics are intuitionistic propositional logics but they are extended with the $\Diamond$ and $\Box$ operators. Moreover, the following axioms which are called k-axioms are added:

k1: $\Box(A \to B) \to (\Box A \to \Box B)$
k2: $\Box(A \to B) \to (\Diamond A \to \Diamond B)$
k3: $\Diamond(A \lor B) \to (\Diamond A \lor \Diamond B)$
k4: $(\Diamond A \to \Box B) \to \Box(A \to B)$
k5: $\neg\Diamond\bot$

The resulting logic is called intuitionistic IK. However, there is another proposal to define intuitionistic modal logic. That variant is called constructive IK and does only include k1 and k2. Furthermore, its semantics is different from those of intuitionistic IK. In this thesis we will assume intuitionistic IK when we talk about intuitionistic modal logic.

Aims of this thesis By adding combinations of the intuitionistic modal axioms T,D,B,4, and 5 to IK 15 different logics are generated, see Figure 1. The scope of this paper is to analyse the relations between these logics. All logics were proved complete and sound in [19].

In the following, proofs are given to show the equivalences of different axiomatisations and the inclusion relations shown in the cube. We chose to use an embedding of IML in higher-order logic for this purpose. Embeddings of other logics in higher-order logic have already been realised by Christoph Benzmüller in [2], Christoph Benzmüller, Maximilian Claus and Nik Sultana in [4], and Christoph Benzmüller and Bruno Woltzenlogel Paleo in [6].

We were able to show that the axioms T,D,B,4, and 5 are equal to some frame correspondences. However, using these correspondences we will verify that some combinations of the intuitionistic modal axioms are indeed equivalent. Furthermore, proofs are given to demonstrate that some logics are stronger than others. In summary, we will verify the whole intuitionistic modal logic cube, except from some trivial statements.

Another goal of this thesis is to show advantages and limits of automatic theorem proving. In [2] Benzmüller published the time the provers needed for verifying the modal logic cube. It only took 40 seconds to verify all relations. As one can see in the following, in this case it was a bit more complicated, as some proofs could not be reconstructed in Isabelle.

The biggest influence of this thesis is given by two noticeable papers [2, 4] in which the authors elegantly verified the modal logic cube. This work is strongly oriented towards them and their methodology has been adapted. Furthermore, we make use of the definition of intuitionistic modal logic as given by Gordon Plotkin and Colin Stirling in [18].

The PDF presentation of this paper is automatically generated from the Isabelle/HOL [16] source code by using the Isabelle's document preparation tool. We also made use of the reasoning tools provided by Isabelle. Especially Sledgehammer [17] turned out to be very useful. This tool applies automatic theorem provers and satisfiability-modulo theories solvers on a given goal. Of particular note are the external higher-order theorem provers LEO-II [5] and Satallax [8] and the build-in prover Metis [12]. We also used Nitpick [7], a counterexample generator for Isabelle, especially to prove the inclusion relationships.

Outline The thesis is structured as follows: Section 2 gives an introduction in higher-order logic. Section 3 defines syntax and semantics of IML, section 4 presents the IML cube and gives information about frame correspondences in IML. Then section 5 defines an encoding of IML in higher-order logic. The premises assumed in previous sections are proven in section 6. After that, section 7 gives evidence for the correspondences named in section 4. Section 8 shows that some different axiomatisations are equivalent whereas in section 9 cases are shown in which one logic is stronger than the other. Finally, section 10 compares LEO-II and Satallax, and section 11 concludes this thesis.

2 Higher-order Logic

Higher-order logic (HOL) was formalised by Bertrand Russel and Alfred Whitehead. The Alonzo Church's formulation, based on his simple type theory, was published in 1940 and is the canonical choice [9]. HOL was disregarded for many years. But because it can be used for mechanised reasoning and linguistics it was invoked again. It was in the late 1960's when HOL was combined with modal operators [15]. A description of higher-order modal logic can be found in [3, 6].

To understand HOL, it is necessary to understand the simply typed λ calculus. Let T be a set of types. In T there are the basic types o, which stands for booleans and μ, which denotes individuals. Whenever α, $\beta \in$ T then $(\alpha \rightarrow \beta) \in$ T. T is freely generated from the set of basic types $\{o, \mu\}$. This means that $(\alpha_1 \rightarrow \beta_1) \equiv (\alpha_2 \rightarrow \beta_2)$ implies $\alpha_1 \equiv \alpha_2$ and $\beta_1 \equiv \beta_2$.

All in all, a type T is generated according to the following grammar:

$$\mathrm{T} ::= \mu \mid \mathrm{o} \mid (\mathrm{T} \rightarrow \mathrm{T})$$

In the following, parentheses are avoided, function types associate to the right.

A formula in HOL is given by:

$$\mathrm{A,B} ::= \mathrm{p}_\alpha \mid \mathrm{X}_\alpha \mid (\lambda \mathrm{X}_\alpha.\ \mathrm{A}_\beta)_{\alpha \rightarrow \beta} \mid (\mathrm{A}_{\alpha \rightarrow \beta}\ \mathrm{B}_\alpha)_\beta \mid (\neg_{\mathrm{o} \rightarrow \mathrm{o}} \mathrm{A}_\mathrm{o})_\mathrm{o} \mid ((\vee_{\mathrm{o} \rightarrow \mathrm{o} \rightarrow \mathrm{o}} \mathrm{A}_\mathrm{o}) \mathrm{B}_\mathrm{o})_\mathrm{o}$$
$$\mid (\forall_{(\alpha \rightarrow \mathrm{o}) \rightarrow \mathrm{o}} (\lambda \mathrm{X}_\alpha.\ \mathrm{A}_\mathrm{o}))_\mathrm{o} \mid (\Box_{\mathrm{o} \rightarrow \mathrm{o}} \mathrm{A}_\mathrm{o})_\mathrm{o}.$$

,where α, $\beta \in$ T. p_α is a typed constant, X_α denotes typed variables. Terms of type o are called formulas.

$(\lambda \mathrm{X}_\alpha.\ \mathrm{A}_\beta)_{\alpha \rightarrow \beta}$ is a term of type $\alpha \rightarrow \beta$. This yields to be a lambda abstraction. When applying a variable X of type α to the term of type $\alpha \rightarrow \beta$ a term of type β is formed. To emphasise this type concept the grammar above was given in prefix notation, but in the following we will switch to infix notation. Additionally to the connectives above, $\bot$, $\top$, $\wedge$, $\rightarrow$, $\equiv$, and $\exists$ can be defined canonically.

β-reduction is realised as follows. [A/X]B is defined as the substitution of a term A_α in a term B_β with a variable X_α. When a variable A is applied to a lambda abstraction (λX. B) it β-reduces to [A/X]B. Implicitly, this substitution rule also defines α reduction, but of course it is a bit more complicated if B contains bound variables which are replaced by other bound variables. η-reduction on the other hand is simple, a term (λX. B X) where X is not free in B reduces to B.

A model for HOL is a tuple M = <D,I> where D is a frame and I a set of interpretation functions. A frame is a collection $\{\mathrm{D}_\alpha\}_{\alpha \in \beta}$ of nonempty sets D_α and D_o is the set of basic types {T,F}. $\mathrm{D}_{\alpha \rightarrow \beta}$ contains functions mapping D_α to D_β.

In the following we use Henkin semantics, which have been proven to be complete [11]. This is important because standard semantics does not allow a complete mechanisation [3].

An interpretation function maps constant symbols q_α to elements of D_α. For example, a constant symbol p_o is mapped to True or False. The valuation function $||A_\beta||$ determines the value $d \in D_\alpha$ of a HOL term A of type β on a model M = <D,I>. g denotes the used assignment.

$$
\begin{aligned}
||q_\alpha||^{M,g} &= I(q_\alpha) \\
||X_\alpha||^{M,g} &= g(X_\alpha) \\
||(A_{\alpha \to \beta}\ B_\alpha)_\beta||^{M,g} &= ||A_{\alpha \to \beta}||^{M,g}||B_\alpha||^{M,g} \\
||(\lambda X_\alpha.\ A_\beta)_{\alpha \to \beta}||^{M,g} &= \text{the function f from } D_\alpha \text{ to } D_\beta \text{ such that} \\
&\quad\ f(d) = ||A_\beta||^{M,g[d/X_\alpha]} \text{ for all } d \in D_\alpha \\
||(\neg_{o \to o})\ A_o)_o||^{M,g} &= T \text{ iff } ||A_o||^{M,g} = F \\
||((\vee_{o \to o \to o}\ A_o)\ B_o)_o||^{M,g} &= T \text{ iff } ||A_o||^{M,g} = T \text{ or } ||B_o||^{M,\,g} = T \\
||(\forall_{(\alpha \to o) \to o}(\lambda X_\alpha.\ A_o))_o||^{\ M,g} &= T \text{ iff for all } d \in D_\alpha \text{ we have } ||A_o||^{M,g[d/X_\alpha]} = T
\end{aligned}
$$

3 Intuitionistic Modal Logic

In this section an introduction to intuitionistic modal logic (IML) is presented. As mentioned before there is no canonical choice which axioms should be considered.

There are two different proposals that prevail in literature: intuitionistic modal logic and constructive modal logic [1]. This thesis focuses on intuitionistic logic as proposed by Plotkin et al. in [18]. Minimal changes are made in accordance with Lutz Straßburger [20] towards a simplified notation. As there are a lot of different proposals for syntax and semantics regarding intuitionistic modal logic, it is most important to define every construct with highest precision.

The sentential modal language L consists of a set of formulas. A formula A is generated by:

$$A ::= A \wedge A \mid A \vee A \mid A \to A \mid \Diamond A \mid \Box A \mid \bot \mid q$$

where $q \in Q$ and $Q = \{a,b,c,...\}$ is a set of atomic sentences. $\neg A$ is defined as $A \to \bot$

The $\Box$ operator may be verbalised with the expression "it is necessary that", while the $\Diamond$ symbol means "it is possible that". In natural language we often use the signal word "must" for the $\Box$ and "may" for the $\Diamond$ operator. Both connectives are strongly connected with the concept of possible worlds explained in the next paragraph.

In propositional logic, an atomic sentence valuates to True or False. The truth value can be computed easily by using a truth table. By adding the concept of modality the truth value is dependent on a set W of worlds. For example, the statement "Batman exists" may be false in the real world but it is true in the DC Universe. However, two worlds could also be different temporal states of a system (e.g. a computer before a program was executed and after that). To show a connection between two worlds, we use an accessibility relation

R. If w R v applies for two worlds w and v, it means that from w's point of view it is possible that w is true [19].

In classical logic a model for a formula is a variable assignment whose interpretation function evaluates the formula to true. In IML a model has more components. We will use the semantics proposed by Plotkin et al. which are based on Kripke semantics [14, 18].

An intuitionistic modal model for the language L is tuple $<W, \leq, R, V>$. The valuation V is a function $W \rightarrow P(Q)$ which maps a world to the set of atomic sentences which are true in this world. P(Q) is the power set of Q. The other three components are called a modal frame.

The most minimal version of a Kripkean intuitionistic modal frame is $<W, \leq, R>$, W is a set of worlds and R the modal accessibility relation. The other relation $\leq$ is called the intuitionistic information relation and is partially ordered. That means that it is reflexive, transitive, and antisymmetric. This third property of antisymmetry was left out in [20], where $\leq$ just has to be pre-ordered. In the following the notation $\geq$ is used as it is more convenient in some cases: w $\leq$ v iff v $\geq$ w.

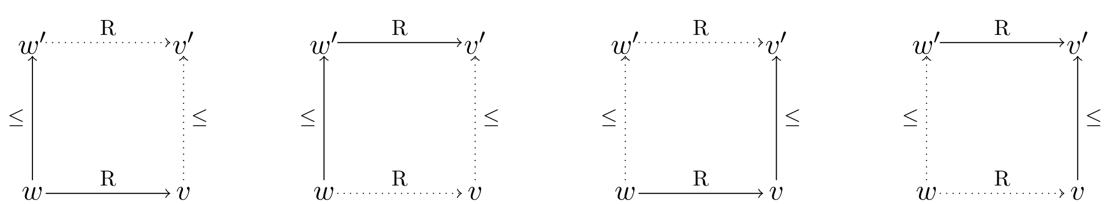
The valuation V function is connected with the $\leq$ relation in the following manner:

if w $\leq$ w' then V(w) $\subset$ V(w') (VMon)

In [18] four different ways are proposed in which the $\leq$ relation and the R relation may be connected:

1. if w $\leq$ w' and w R v then $\exists$v'. w' R v' and v $\leq$ v'
2. if w $\leq$ w' and w' R v' then $\exists$v. w R v and v $\leq$ v'
3. if v $\leq$ v' and w R v then $\exists$w'. w' R v' and w $\leq$ w'
4. if v $\leq$ v' and w' R v' then $\exists$w. w R v and w $\leq$ w'

For a better understanding of these conditions, one may visualise them as diagrams:



condition 1 condition 2 condition 3 condition 4

We will use frame conditions 1 and 3 as suggested in [18] and adopted in [20]. Plotkin et al. emphasise, that the choice of frame conditions is connected strongly with the semantic clauses for the modal operators.

Moreover, they suggest these two statements:

1. $\Diamond A \rightarrow \neg\Box\neg A$
2. $\neg\Diamond A \rightarrow \Box\neg A$

Number 1 follows from frame condition 1, number 2 from frame condition 3. See section 6.1 for an automated proof of these statements.

To evaluate a formula A in the context of a world w we define the evaluation relation $\models$.

$$
\begin{array}{lll}
w \models q & \text{iff} & q \in V(w) \\
w \models A \wedge B & \text{iff} & w \models A \text{ and } w \models B \\
w \models A \vee B & \text{iff} & w \models A \text{ or } w \models B \\
w \models A \rightarrow B & \text{iff} & \forall w' \geq w \text{ if } w' \models A \text{ then } w' \models B \\
w \models \Diamond A & \text{iff} & \exists u.\ w\ R\ u \text{ and } u \models A \\
w \models \Box A & \text{iff} & \forall w' \geq w\ \forall u.\ \text{if } w'\ R\ u \text{ then } u \models A
\end{array}
$$

Furthermore, there are different understandings of validity. A formel A is:

1. valid in a model $M = <W, \leq, R, V>$ iff $\forall w \in W$: $w \models A$.
2. valid in a frame $<W, \leq, R>$ iff $\forall V$: $<W, \leq, R, V> \models A$.
3. valid iff $\forall <W, \leq, R>$: $<W, \leq, R> \models A$.

The following lemma is important as it is impossible to use the relationship between $\leq$ and V (VMon) directly in the encoding.

$$\text{if } w \leq w' \text{ and } w \models A \text{ then } w' \models A \qquad \text{(Monotonicity)}$$

This property is used a lot in the verification process discussed in the rest of this thesis.

However, it is not necessary to declare the k-axioms mentioned in section 1. As shown in section 6.2 they can be proven just by using that the relation $\leq$ is partially ordered. Any theorem that is derivable from these k-axioms or from the axioms of intuitionistic propositional logic by using the modus ponens or necessitation rule is a theorem of intuitionistic modal logic IK. In fact, if a theorem is derivable under those circumstances, it is exactly one of the theorems of IK.

In the following we want to explain the relationship between classical modal logic and intuitionistic modal logic. It is obvious that all theorems of IML are valid in classical logic but not all classically valid formulae are valid in IML.

One example is the double negation. In IML $p \to \neg\neg p$ is a theorem, but $\neg\neg p \to p$ is not, whereas in classical logic both statements are tautologies. To achieve classical logic one of these axioms is added:

$A \vee \neg A$	(Law of excluded middle)
$\neg\neg A \to A$	(Double negation)
$(\neg A \to \neg B) \to (B \to A)$	(Law of contraposition)
$((A \to B) \to A) \to A$	(Peirce's law)

For further information on intuitionistic modal logic there can be found a lot of literature [15, 19].

4 Intuitionistic Modal Logic Cube and Frame Correspondences

The IML cube resembles the one of classical modal logic [4]. There are 15 different logics generated by adding the intuitionistic modal axioms D, T, B, 4, and 5 to the logic IK. One may argue that by combining these five axioms $\binom{5}{0} + \binom{5}{1} + \binom{5}{2} + \binom{5}{3} + \binom{5}{4} + \binom{5}{5} = 32$ logics are obtained but as we will prove exemplary in section 8 some of these logic are in fact equivalent. The whole cube is presented in Figure 1.

The intuitionistic modal axioms are not the same as the modal axioms in classical modal logic, because there is no duality between $\Box$ and $\Diamond$ anymore. For example axiom T consists of $A \to \Diamond A$ and $\Box A \to A$. In classical logic these statements are equivalent, in IML they are not. Thus the T axiom in IML is expressed by the conjunction of both axioms.

The only exception is axiom D: $\Box A \to \Diamond A$, here the second part is the same as the first. To distinguish between the two parts of an intuitionistic modal axiom, we will denote one part with adding a $\Diamond$ and one with adding a $\Box$ to their name.

At this point, we want to extend the definition of validity. We say that for $X \subseteq \{D,T,B,4,5\}$ a frame is called an X-frame if the relations R and $\leq$ obey the frame conditions given in Table 1. A formula A is derivable from IK+X iff A is valid in all X-frames. A formula is X-valid iff it is valid in all X-frames.

Now we can define different logics more clearly. For example IK+{D,5} is the logic which evolves from IK by adding the axioms D and 5. The logic is called ID5, the K in the name is left out. Hereafter, this article will make use of names like ID5 instead of writing IK+{D,5}. But it is important to bear in mind how a logic is generated from IK.

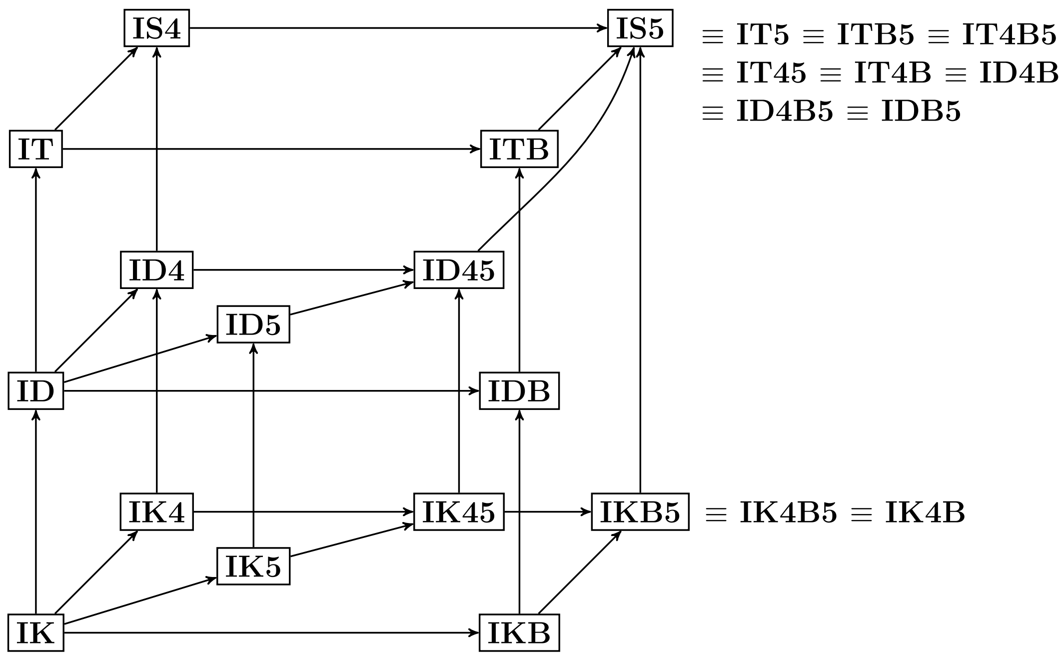


Figure 1: IML Cube

At this point, we could use the intuitionistic modal axioms for the verification of the IML cube. But it is more efficient to use conditions regarding the frame, as done by Benzmüller et al. in [4] for classical modal logic.

The first challenge was to find correspondences which are valid in the same semantical setting as the one we used. In [13] the author uses different semantics but in one chapter he discussed the semantics used here. His assumption was that all intuitionistic modal axioms except from D do not correspond to the classical conditions.

The second challenge was that nobody seems to have given a proof for the inequality of the intuitionistic modal axioms to the classical frame correspondences. Most authors referred to Plotkin et al. [18], who give two reasons why it would be unlikely that they are equal.

Their first reason is that because of the breakdown, mentioned above, it would be unlikely if both parts of an intuitionistic modal axiom correspond to the same frame condition.

Their second reason is that a correspondence theorem should not only make restraints on R but also on the relationship between R and $\leq$. In their argumentation Plotkin et al. do not include thoughts about the influence of the properties the $\leq$ relation has, nor about the interconnection defined with frame condition 1 and 3.

When we tried to find a counterexample for equality of axiom T and reflexivity we were surprised. Instead of refuting the equality we actually were able to prove it. The same happened for all other intuitionistic modal axioms.

Even after examining all our assumptions most thoroughly, we could not find any divergence to the axiomatisation of Plotkin et al. To fully understand the reasons for this behaviour further work is needed. Possibly we need to reexamine the long-established position that the classical correspondences are not valid in the setting of Plotkin et al.

Nevertheless we will work with slightly different frame conditions, which make statements about the connection of the $\leq$ and R relations. It is helpful, that each correspondence axiom is a instance of the intuitionistic version of the $G^{k,l,m,n}$ schema [18], also called Lemmon-Scott schema [13].

$G^{k,l,m,n}$ is the schema:

$$\Diamond^k \Box^l A \rightarrow \Box^m \Diamond^n A, \text{ for k,l,m,n} \geq 0.$$

Where R^n for n $\geq$ 0 is defined as:

$w\ R^0\ v$ iff $w = v$
$w\ R^{n+1}\ v$ iff $\exists u.\ w\ R\ u$ and $u\ R^n\ v$

The following theorem specifies a very useful connection:

A modal frame <W, $\leq$, R> validates $G^{k,l,m,n}$ iff:
if $w\ R^k\ u$ and $w\ R^m\ v$ then $\exists u' \geq u\ \exists x.\ (u'\ R^l\ x \wedge v\ R^n\ x)$

Take for example the T axiom mentioned above. It consists of two parts: $A \rightarrow \Diamond A$ and $\Box A \rightarrow A$. As one can see both are instances of the $G^{k,l,m,n}$ schema. For example the first part resolves like this:

$G^{k,l,m,n}$ with k=l=m=0 and n = 1
$\equiv \Diamond^0 \Box^0 A \rightarrow \Box^1 \Diamond^0 A$
$\equiv \forall w\ u\ v.\ (w\ R^0\ u \wedge w\ R^0\ v) \rightarrow \exists u' \geq u.\ \exists x.\ (u'\ R^0\ x \wedge v\ R^l\ x)$
$\equiv \forall w\ u\ v.\ (w = u \wedge w = v) \rightarrow \exists u' \geq u.\ \exists x.\ (u' = x \wedge (\exists a.\ v\ R\ a \wedge a\ R^0\ x))$
$\equiv \forall w.\ \exists u' \geq w.\ (\exists a.\ w\ R\ a \wedge a = u')$
$\equiv \forall w.\ \exists u' \geq w.\ w\ R\ u'$

By applying the schema to all IML axioms we achieved the results indicated in Table 1.

The correspondences have been proven in section 7. The classical correspondences are illustrated in the same table and also have been proven in section 7.

Table 1: Intuitionistic modal axioms D,T,B,4, and 5 with corresponding frame conditions

name	axiom	intuitionistic frame correspondence	classical correspondence
T□: T◊:	□A → A A → ◊A	∀w. ∃u'. w ≤ u' ∧ u' R w ∀w. ∃u'. w ≤ u' ∧ w R u'	∀w. w R w (reflexivity)
B□: B◊:	A → □◊A ◊□A → A	∀w u. w R u → ∃u'. w ≤ u' ∧ u R u' ∀w u. w R u → ∃u'. u ≤ u' ∧ u' R w	∀w u. w R u → u R w (symmetry)
D:	□A → ◊A	∀w. ∃u'. w ≤ u' ∧ (∃x. u' R x ∧ w R x)	∀w. ∃u. w R u (seriality)
4◊: 4□:	◊◊A → ◊A □A → □□A	∀w v. (∃u. w R u ∧ u R v) → (∃u'. v ≤ u' ∧ w R u') ∀w v. (∃u. w R u ∧ u R v) → (∃u'. w ≤ u' ∧ u' R v)	∀w u v. w R u ∧ u R v → w R v (transitivity)
5◊: 5□:	◊A → □◊A ◊□A → □A	∀w v u. (w R u ∧ w R v) → (∃u'. u ≤ u' ∧ u' R v) ∀w v u. (w R u ∧ w R v) → (∃u'. u ≤ u' ∧ v R u')	∀w u v. w R u ∧ w R v → u R v (euclideaness)

5 An Embedding of Intuitionistic Modal Logics in HOL

As described in section 1 an embedding of IML in higher-order logic is used to verify the IML cube. This has already been done by Benzmüller et al. [4] in the scope of the modal logic cube and is further described in [6].

The latter emphasised that many problems can be encoded more elegantly in higher-order logics than in less expressive logics. This issue may be beyond efficiency, the problem could be impossible to solve. The authors of [6] give a noticeable example: In first-order logic the proof for George Boolos' curious inference is very long whereas in higher-order logic it is a one page proof.

In contrast to HOL the truth of a formula in IML is dependent from its context, called its *possible world*. To embed IML in HOL without losing this information, we need to store it somewhere. This is realised by introducing a special type for worlds ι. There is also a type for individuals, denoted by μ.

In section 3 we did not extend the concept of types to IML but it is easy to transfer. There are IML formulas of type μ, o, and combinations of these. An IML type α associates to HOL type $\lceil\alpha\rceil$ as follows:

$$\begin{aligned}\lceil\mu\rceil &= \mu\\ \lceil o\rceil &= \sigma = \iota \rightarrow \sigma\\ \lceil\alpha \rightarrow \beta\rceil &= \lceil\alpha\rceil \rightarrow \lceil\beta\rceil\end{aligned}$$

All types can be modelled in Isabelle in a straightforward fashion:

typedecl ι
typedecl μ
type-synonym $\sigma = (\iota \Rightarrow bool)$

The two relations are modelled as constants.

consts $R :: \iota \Rightarrow \iota \Rightarrow bool$ — accessibility relation
consts $le :: \iota \Rightarrow \iota \Rightarrow bool$ — ordering relation

A new evaluation function is defined. For example, in both IML and HOL the or connective has the type $o \to o \to o$. It takes two booleans as input and gives one as output. The embedding function resolves $\lceil A_o \vee B_o \rceil$ ($\vee$ connective of IML) to $\lambda w_\iota.\ \lceil A \rceil\ w \vee \lceil B \rceil\ w$ ($\vee$ connective of HOL). The type of this term is $\sigma \to \sigma \to \sigma$.

To distinguish between HOL and lifted operators we use of bold lettering. For example $\boldsymbol{\vee}_{\sigma \to \sigma \to \sigma} = \lambda\phi_\sigma.\ \lambda\psi_\sigma.\ \lambda w\mu.\ \phi\ w \vee \psi\ w$. Additionally to the bold lettering it is helpful to recall the type concept. The bold operators are used with terms of type σ, the HOL operators with those of type bool.

To retrieve all operators, we need to extend the $\lceil . \rceil$ function. In 3 we did not distinguish between constants and variables, but it is easy to do so. An IML term A is associated with a HOL term $\lceil A \rceil$ in the following way:

$$\begin{aligned} \lceil q_\alpha \rceil &= q_{\lceil \alpha \rceil} \\ \lceil A_o \vee_{o \to o \to o} B_o \rceil &= \lambda w_\iota.\ \lceil A \rceil\ w \vee \lceil B \rceil\ w \\ \lceil \Box_{o \to o} A_o \rceil &= \lambda w.\ \forall w'\ v'.\ w\ le\ w' \to (w'\ R\ v' \to \lceil A \rceil) \end{aligned}$$

This definition is not complete. All other formula liftings are given in the following code. To evaluate a term ϕ in a certain world w we only have to evaluate ϕ w. Whether the term contains or does not contain connectives it is possible to do exactly the same.

Unlike previously, in the Isabelle code, the terms ϕ and ψ are given as parameters but this is just for better readability. They are converted in lambda notation by Isabelle. For the same reason we switch to infix notation.

abbreviation $mand :: \sigma \Rightarrow \sigma \Rightarrow \sigma$ **where** $A \wedge B \equiv \lambda w.\ A\ w \wedge B\ w$
abbreviation $mor :: \sigma \Rightarrow \sigma \Rightarrow \sigma$ **where** $A \vee B \equiv \lambda w.\ A\ w \vee B\ w$
abbreviation $mimp :: \sigma \Rightarrow \sigma \Rightarrow \sigma$ **where** $A \to B \equiv \lambda w.\ \forall w'.\ w\ le\ w' \longrightarrow (A\ w' \longrightarrow B\ w')$
abbreviation $mbox :: \sigma \Rightarrow \sigma$ **where** $\Box A \equiv \lambda w.\ \forall w'\ v'.\ w\ le\ w' \longrightarrow (w'\ R\ v' \longrightarrow A\ v')$
abbreviation $mdia :: \sigma \Rightarrow \sigma$ **where** $\Diamond A \equiv \lambda w.\ \exists v\ .\ (w\ R\ v \wedge A\ v)$
abbreviation $mfalse :: \sigma$ **where** $\bot \equiv \lambda w.\ False$
abbreviation $mnot :: \sigma \Rightarrow \sigma$ **where** $\neg A \equiv \lambda w\ .\ (A\ \to \bot)\ w$

Quantifiers can be embedded additionally.

abbreviation *mexists* :: $('a \Rightarrow \sigma) \Rightarrow \sigma$ **where** $\exists\ \Phi \equiv (\lambda w.\ \exists x.\ \Phi\ x\ w)$
abbreviation *mforall* :: $('a \Rightarrow \sigma) \Rightarrow \sigma$ **where** $\forall\ \Phi \equiv (\lambda w.\ \forall x.\ \Phi\ x\ w)$

A formula is valid if it is true for all worlds. In section 2 three different notations of validity were introduced. It seems that valid here just means valid in a model but R and $\leq$ and the valuation function are arbitrary, so it means validity of a formula.

abbreviation *valid* :: $\sigma \Rightarrow bool$ **where** $\lfloor p \rfloor \equiv \forall w.\ p\ w$

The $\leq$ relation is partially ordered. Additionally, we define the conditions named in section 3.

axiomatization where *FR*: $\forall w.\ w\ le\ w$
axiomatization where *FT*: $\forall w\ u\ v.\ (w\ le\ u \wedge u\ le\ v) \longrightarrow w\ le\ v$
axiomatization where *FA*: $\forall w\ u.\ (w\ le\ u \wedge u\ le\ w) \longrightarrow (w{=}u)$
axiomatization where *F1*: $\forall w\ w'\ v.\ ((w\ le\ w' \wedge w\ R\ v) \longrightarrow (\exists v'.\ (w'\ R\ v' \wedge v\ le\ v')))$
axiomatization where *F3*: $\forall v\ v'\ w.\ ((v\ le\ v' \wedge w\ R\ v) \longrightarrow (\exists w'.\ (w'\ R\ v' \wedge w\ le\ w')))$

The property VMon can not be translated directly into HOL because the valuation function is included in the embedding. We can use the lemma of Monotonicity instead. VMon is included in it.

axiomatization where *Mon*: $\forall w\ w'.\ \forall\ A.\ (w\ le\ w' \wedge A\ w) \longrightarrow A\ w'$

Two last issues about this embedding shall be mentioned here. They are explained more in detail in [6].

In the embedding in [6] it was assumed that all constants are rigid. A constant q_α is rigid if $\exists$d. $\forall$w. $I_w(q_\alpha) = d$. This means that the interpretation of a constant is the same in all worlds. Constants of type o are an exception and still dependent on a world. This behaviour is called flexible.

To broaden the embedding to flexible constants, type-raising may be applied. An IML constant symbol of the type μ would be mapped to a HOL constant with the type $\iota \to \mu$. Because the information about the current world is not available anymore, it needs to be passed over.

In the embedding above this issue does not occur because there are no constants in IML.

The second issue deals with domains. Imagine that an individual in D_μ does not exist in all worlds. For example, Batman does not exist in the real world. Previously, we assumed that all domains are constant. In [6] the authors mention modifications to face varying domains.

6 Premise Check

This section is meant to validate premises introduced in the previous sections. Also, it is not necessary for the verification process of the IML cube to check them, their validity is so important that they are shown here nevertheless.

6.1 Negation Statements

Both statements are proposed by Plotkin et al. in [18]:

lemma $\lfloor(\forall\,(\lambda A.\ ((\Diamond A) \rightarrow \neg\Box\neg A)))\rfloor$ **by** (*meson FR Mon*)
lemma $\lfloor(\forall\,(\lambda A.\ ((\neg(\Diamond A)) \rightarrow \Box\neg A)))\rfloor$**by** (*meson Mon*)

6.2 K-Axioms

In this thesis the K-Axioms proposed in [18] are used. In classical modal logic all other K-Axioms would follow from k1 and the De Morgans laws. In an intuitionistic setting properties of the $\leq$ and R relation are necessary.

abbreviation *k1* **where** $k1 \equiv \lfloor(\forall\,(\lambda A.\ (\forall\,(\lambda B.\ ((\Box\ (A \rightarrow B)) \rightarrow ((\Box\ A) \rightarrow (\Box\ B)))))))\rfloor$
abbreviation *k2* **where** $k2 \equiv \lfloor(\forall\,(\lambda A.\ (\forall\,(\lambda B.\ ((\Box\ (A \rightarrow B)) \rightarrow ((\Diamond\ A) \rightarrow (\Diamond\ B)))))))\rfloor$
abbreviation *k3* **where** $k3 \equiv \lfloor(\forall\,(\lambda A.\ (\forall\,(\lambda B.\ ((\Diamond\ (A \vee B)) \rightarrow ((\Diamond\ A) \vee (\Diamond\ B)))))))\rfloor$
abbreviation *k4* **where** $k4 \equiv \lfloor(\forall\,(\lambda A.\ (\forall\,(\lambda B.(((\Diamond\ A) \rightarrow (\Box\ B)) \rightarrow (\Box(A \rightarrow B)))))))\rfloor$
abbreviation *k5* **where** $k5 \equiv \lfloor\neg\ (\Diamond\bot)\rfloor$

theorem *k1* **using** *FR FT* **by** *blast*
theorem *k2* **using** *FR* **by** *blast*
theorem *k3* **by** *blast*
theorem *k4* **by** (*meson F3 FR FT*)
theorem *k5* **by** *simp*

7 Proofs of the Frame Correspondences for IML

In this section proofs regarding the frame conditions explained in section 4 are given. They make aware of the challenges automatic theorem proving still has to face. All proofs were suggested by Sledgehammer. However, nine of them could not by reconstructed. Unfortunately, one statement (lemma A4-b-2) could not be verified at all. On the one hand this is a serious concern, on the other hand it may be enough to know that some proof exists.

Another interesting point connected to this matter is that, while we were working on this thesis, it was very difficult to prove some theorems on one of the computers. It was a 1.7GHz dual-core computer with 8GB memory. On an other computer a 2,5GHz quad-core with 16 GB memory it worked much better, showing how much influence the used infrastructure still has.

It must be mentioned that authors often do not distinguish between meta level argumentation and object level. To demonstrate the importance of this difference we give the following example.

When intuitionistic implication ($\boldsymbol{\rightarrow}$ operator) is used Nitpick quickly finds counterexamples for both of these two lemmata.

lemma $\lfloor(\forall(\lambda A.\ ((A \rightarrow (\Diamond\ A)) \rightarrow ((\Box\ A) \rightarrow A))))\rfloor$ **nitpick**[*user-axioms*] **sorry**
lemma $\lfloor(\forall(\lambda A.\ (((\Box\ A) \rightarrow A) \rightarrow (A \rightarrow (\Diamond\ A)))))\rfloor$ **nitpick**[*user-axioms*] **sorry**

On the opposite when using classical implication ($\rightarrow$ operator) it is possible to prove both statements.

lemma $\lfloor(\forall(\lambda A.\ (A \rightarrow (\Diamond\ A))))\rfloor \longrightarrow \lfloor(\forall(\lambda A.\ ((\Box\ A) \rightarrow A)))\rfloor$ — by (metis Mon) **sorry**
lemma $\lfloor(\forall(\lambda A.\ ((\Box\ A) \rightarrow A)))\rfloor \longrightarrow \lfloor(\forall(\lambda A.\ (A \rightarrow (\Diamond\ A))))\rfloor$ — by (metis FR Mon ext) **sorry**

Another option is to use intuitionistic implication ($\boldsymbol{\rightarrow}$ operator) and to quantify over each part individually. All statements can be proven.

lemma $\lfloor((\forall(\lambda A.\ ((A \rightarrow (\Diamond\ A))))) \rightarrow (\forall(\lambda A.((\Box\ A) \rightarrow A))))\rfloor$ — by (metis Mon) **sorry**
lemma $\lfloor((\forall(\lambda A.\ (((\Box\ A) \rightarrow A)))) \rightarrow (\forall(\lambda A.(A \rightarrow (\Diamond\ A)))))\rfloor$ — by (metis FR Mon ext) **sorry**

7.1 Intuitionistic modal axioms

abbreviation *T-dia* **where** *T-dia* $\equiv \lfloor(\forall(\lambda A.\ (A \rightarrow (\Diamond\ A))))\rfloor$
abbreviation *T-box* **where** *T-box* $\equiv \lfloor(\forall(\lambda A.\ ((\Box\ A) \rightarrow A)))\rfloor$
abbreviation *T* **where** *T* $\equiv \lfloor(\forall(\lambda A.\ (((\Box\ A) \rightarrow A) \wedge (A \rightarrow (\Diamond\ A)))))\rfloor$

abbreviation *B-dia* **where** *B-dia* $\equiv \lfloor(\forall(\lambda A.(\ (\Diamond(\Box\ A)) \rightarrow A)))\rfloor$
abbreviation *B-box* **where** *B-box* $\equiv \lfloor(\forall(\lambda A.(\ A \rightarrow (\Box\ (\Diamond\ A)))))\rfloor$
abbreviation *B* **where** *B* $\equiv \lfloor(\forall(\lambda A.(\ (\Diamond(\Box\ A)) \rightarrow A) \wedge (A \rightarrow (\Box\ (\Diamond\ A)))))\rfloor$

abbreviation *D* **where** *D* $\equiv \lfloor(\forall(\lambda A.((\Box\ A) \rightarrow (\Diamond\ A))))\rfloor$

abbreviation *IV-box* **where** *IV-box* $\equiv \lfloor(\forall(\lambda A.((\Box\ A) \rightarrow (\Box(\Box\ A)))))\rfloor$
abbreviation *IV-dia* **where** *IV-dia* $\equiv \lfloor(\forall(\lambda A.((\Diamond(\Diamond\ A)) \rightarrow (\Diamond\ A))))\rfloor$
abbreviation *IV* **where** *IV* $\equiv \lfloor(\forall(\lambda A.((\Box\ A) \rightarrow (\Box(\Box\ A)))\wedge((\Diamond(\Diamond\ A)) \rightarrow (\Diamond\ A))))\rfloor$

abbreviation *V-dia* **where** *V-dia* $\equiv \lfloor(\forall(\lambda A.((\Diamond(\Box\ A)) \rightarrow (\Box\ A))))\rfloor$

abbreviation $V\text{-}box$ **where** $V\text{-}box \equiv \lfloor(\forall (\lambda A.((\Diamond\ A) \to (\Box\ (\Diamond\ A)))))\rfloor$
abbreviation V **where** $V \equiv \lfloor(\forall (\lambda A.((\Diamond(\Box\ A)) \to (\Box\ A)) \wedge ((\Diamond\ A) \to (\Box\ (\Diamond\ A)))))\rfloor$

7.2 Frame Correspondences

abbreviation $FC\text{-}T\text{-}dia$ **where** $FC\text{-}T\text{-}dia \equiv \forall w.\ \exists u'.\ w\ le\ u' \wedge w\ R\ u'$
abbreviation $FC\text{-}T\text{-}box$ **where** $FC\text{-}T\text{-}box \equiv \forall w.\ \exists u'.\ w\ le\ u' \wedge u'\ R\ w$
abbreviation $FC\text{-}T$ **where** $FC\text{-}T \equiv FC\text{-}T\text{-}dia \wedge FC\text{-}T\text{-}box$

abbreviation $FC\text{-}B\text{-}dia$ **where** $FC\text{-}B\text{-}dia \equiv \forall w\ u.\ w\ R\ u \longrightarrow (\exists u'.\ u\ le\ u' \wedge u'\ R\ w)$
abbreviation $FC\text{-}B\text{-}box$ **where** $FC\text{-}B\text{-}box \equiv \forall w\ u.\ w\ R\ u \longrightarrow (\exists u'.\ w\ le\ u' \wedge u\ R\ u')$
abbreviation $FC\text{-}B$ **where** $FC\text{-}B \equiv FC\text{-}B\text{-}dia \wedge FC\text{-}B\text{-}box$

abbreviation $FC\text{-}D$ **where** $FC\text{-}D \equiv \forall w.\ \exists u'.\ w\ le\ u' \wedge (\exists x.\ u'\ R\ x \wedge w\ R\ x)$

abbreviation $FC\text{-}IV\text{-}dia$
  where $FC\text{-}IV\text{-}dia \equiv \forall w\ u.\ (\exists v.\ w\ R\ v \wedge v\ R\ u) \longrightarrow (\exists u'.\ u\ le\ u' \wedge w\ R\ u')$
abbreviation $FC\text{-}IV\text{-}box$
  where $FC\text{-}IV\text{-}box \equiv \forall w\ u.\ (\exists v.\ w\ R\ v \wedge v\ R\ u) \longrightarrow (\exists u'.\ w\ le\ u' \wedge u'\ R\ u)$
abbreviation $FC\text{-}IV$ **where** $FC\text{-}IV \equiv FC\text{-}IV\text{-}dia \wedge FC\text{-}IV\text{-}box$

abbreviation $FC\text{-}V\text{-}dia$
  where $FC\text{-}V\text{-}dia \equiv \forall w\ v\ u.\ (w\ R\ u \wedge w\ R\ v) \longrightarrow (\exists u'.\ u\ le\ u' \wedge u'\ R\ v)$
abbreviation $FC\text{-}V\text{-}box$
  where $FC\text{-}V\text{-}box \equiv \forall w\ v\ u.\ (w\ R\ u \wedge w\ R\ v) \longrightarrow (\exists u'.\ u\ le\ u' \wedge v\ R\ u')$
abbreviation $FC\text{-}V$ **where** $FC\text{-}V \equiv FC\text{-}V\text{-}dia \wedge FC\text{-}V\text{-}box$

7.3 Classical Frame Correspondences

abbreviation ref **where** $ref \equiv \forall w\ .\ w\ R\ w$
abbreviation ser **where** $ser \equiv \forall w\ .\ \exists v\ .\ w\ R\ v$
abbreviation sym **where** $sym \equiv \forall w\ u\ .\ w\ R\ u \longrightarrow u\ R\ w$
abbreviation $trans$ **where** $trans \equiv \forall w\ u\ v.\ w\ R\ u \wedge u\ R\ v \longrightarrow w\ R\ v$
abbreviation $eucl$ **where** $eucl \equiv \forall w\ u\ v.\ w\ R\ u \wedge w\ R\ v \longrightarrow u\ R\ v$

7.4 Proof of Frame Correspondences

7.4.1 Axiom T◇ corresponds to FC-T◇

lemma *A1-a-1*: $FC\text{-}T\text{-}dia \longrightarrow T\text{-}dia$ **using** *Mon* **by** *blast*
lemma *A1-a-2*: $T\text{-}dia \longrightarrow FC\text{-}T\text{-}dia$ **by** (*meson FR*)
theorem *A1-a*: $T\text{-}dia \longleftrightarrow FC\text{-}T\text{-}dia$ **using** *A1-a-1 A1-a-2* **by** *auto*

7.4.2 Axiom T□ corresponds to FC-T□

lemma *A1-b-1*: $T\text{-}box \longrightarrow FC\text{-}T\text{-}box$ — by (metis FR Mon ext) **sorry**
lemma *A1-b-2*: $FC\text{-}T\text{-}box \longrightarrow T\text{-}box$ **by** *fastforce*
theorem *A1-b*: $FC\text{-}T\text{-}box \longleftrightarrow T\text{-}box$ **using** *A1-b-1* **by** *blast*

7.4.3 T corresponds to reflexivity

lemma *A1-c-1*: $ref \longrightarrow T$ **using** *FR* **by** *blast*

lemma *A1-c-2*: $T \longrightarrow$ *ref* — by (meson FR) **sorry**
theorem *A1-c*: $T \longleftrightarrow$ *ref* **by** (*smt A1-c-1 A1-c-2*)

7.4.4 Axiom B◊ corresponds to FC-B◊

lemma *A2-a-1*: *FC-B-dia* $\longrightarrow$ *B-dia* **by** *blast*
lemma *A2-a-2*: *B-dia* $\longrightarrow$ *FC-B-dia* — by (metis FR Mon ext) **sorry**
theorem *A2-a*: *B-dia* $\longleftrightarrow$ *FC-B-dia* **using** *A2-a-2* **by** *blast*

7.4.5 Axiom B □ corresponds to FC-B□

theorem *A2-b*: *B-box* $\longleftrightarrow$ *FC-B-box* **by** (*meson FR Mon*)

7.4.6 Axiom B corresponds to symmetry

theorem *A2-c-1*: $B \longrightarrow$ *sym* — by (meson FR) **sorry**
lemma *A2-c-2*: *sym* $\longrightarrow B$ **using** *FR Mon* **by** *blast*
theorem *A2-c*: *sym* $\longleftrightarrow B$ **by** (*smt A2-c-1 A2-c-2*)

7.4.7 Axiom D corresponds to FC-D

theorem *A3-a*: $D \longleftrightarrow$ *FC-D* **using** *FR* **by** *blast*

7.4.8 Axiom D corrsponds to seriality

theorem *A3-b*: $D \longleftrightarrow$ *ser* **using** *FR* **by** *fastforce*

7.4.9 Axiom IV◊ corresponds to FC-IV◊

theorem *A4-a*: *IV-dia* $\longleftrightarrow$ *FC-IV-dia* **by** (*meson FR Mon*)

7.4.10 Axiom IV□ corresponds to FC-IV□

lemma *A4-b-1*: *FC-IV-box* $\longrightarrow$ *IV-box* **by** (*smt Mon*)
lemma *A4-b-2*: *IV-box* $\longrightarrow$ *FC-IV-box* **sorry**
theorem *A4-b*: *FC-IV-box* $\longleftrightarrow$ *IV-box* **by** (*smt A4-b-1 A4-b-2*)

7.4.11 Axiom IV corresponds to transitivity

lemma *A4-c-1*: *trans* $\longrightarrow$ *IV* **by** (*meson Mon*)
lemma *A4-c-2*: *IV* $\longrightarrow$ *trans* — by (metis FR) **sorry**
theorem *A4-c*: *IV* $\longleftrightarrow$ *trans* **using** *A4-c-1 A4-c-2* **by** *satx*

7.4.12 Axiom V◊ corresponds to FC-V◊

lemma *A5-a-1*: *FC-V-dia* $\longrightarrow$ *V-dia* **by** (*meson Mon*)
lemma *A5-a-2*: *V-dia* $\longrightarrow$ *FC-V-dia* **sorry**
lemma *A5*: *V-dia* $\longleftrightarrow$ *FC-V-dia* **using** *A5-a-1 A5-a-2* **by** *blast*

7.4.13 Axiom V□ corresponds to FC-V□

lemma *A5-b-1*: *FC-V-box* ⟶ *V-box* **by** (*meson Mon*)
lemma *A5-b-2*: *V-box* ⟶ *FC-V-box* **by** (*meson FR*)
theorem *A5-b*: *FC-V-box* ⟷ *V-box* **using** *A5-b-1 A5-b-2* **by** *blast*

7.4.14 Axiom V corresponds to euclideaness

lemma *A6-c-1*: *V-box* ⟶ *eucl* — by (metis FR) **sorry**
lemma *A6-c-2*: *eucl* ⟶ *V-box* **by** (*meson Mon*)
lemma *A6-c-3*: *V-dia* ⟶ *eucl* — by (metis FR Mon) **sorry**
lemma *A6-c-4*: *eucl* ⟶ *V-dia* **by** (*meson FR Mon*)
theorem *A6-c-5*: *eucl* ⟷ *V* **by** (*smt A6-c-1 A6-c-2 A6-c-4*)

8 Alternative Axiomatisations

As shown in Figure 4, in some cases the same logic can be obtained by adding different combinations of axioms. This is the case for the two logics **IS5** and **D4B**. For example, the axioms used to obtain **IS5**, namely V and T, are equivalent to the ones to obtain **D4B**, which are D, IV, and B.

In this section, proofs are given to show the equality of these logics. Therefore, the correspondence axioms from section 4 are used. It simplifies the proofs because there is no need to deal with formulae anymore. This means that instead of showing V ∧ T ≡ D ∧ V ∧ B, it is sufficient to show FC-V ∧ FC-T ≡ FC-D ∧ FC-IV ∧ FC-B.

To avoid redundancy, the helper lemmata defined in the next subsection are used. For H3 and H4 Sledgehammer found a proof but metis could not reconstruct it. This means that each proof in the sections 8.0.2 to 8.0.10 which is based on H3 or H4 can not be reconstructed either.

It is interesting that the axioms Mon and FR were used a lot, whereas the other properties of the ≤ relation were not used at all.

8.0.1 Helper lemmata

lemma *H1*: *FC-V* ∧ *FC-T* ⟶ *FC-B* **by** (*meson FR Mon*)
lemma *H2*: *FC-V* ∧ *FC-T* ⟶ *FC-D* **using** *FR* **by** *blast*
lemma *H3*: *FC-B* ∧ *FC-V* ⟶ *FC-IV* — by (metis Mon) **sorry**
lemma *H4*: *FC-B* ∧ *FC-IV* ⟶ *FC-V* — by (metis FR Mon) **sorry**
lemma *H5*: *FC-B* ∧ *FC-IV* ∧ *FC-D* ⟶ *FC-T* **using** *H4* **by** *meson*

8.0.2 IT5 ⟺ ITB5

theorem *B1*: (*FC-B* ∧ *FC-T* ∧ *FC-V*) ⟷ (*FC-T* ∧ *FC-V*) **using** *H1* **by** *fastforce*

8.0.3 IT5 $\Longleftrightarrow$ IT45

theorem *B2*: (*FC-T* $\wedge$ *FC-V* $\wedge$ *FC-IV*) $\longleftrightarrow$ (*FC-T* $\wedge$ *FC-V*) **using** *H3 H1* **by** *fastforce*

8.0.4 IT5 $\Longleftrightarrow$ IT4B5

theorem *B3*: (*FC-T* $\wedge$ *FC-V*) $\longleftrightarrow$ (*FC-T* $\wedge$ *FC-IV* $\wedge$ *FC-B* $\wedge$ *FC-V*) **using** *B1 B2* **by** *satx*

8.0.5 IT5 $\Longleftrightarrow$ IT4B

theorem *B4*: (*FC-T* $\wedge$ *FC-IV* $\wedge$ *FC-B*) $\longleftrightarrow$ (*FC-T* $\wedge$ *FC-V*) **using** *H4 B1 B2* **by** *satx*

8.0.6 IT5 $\Longleftrightarrow$ ID4B

theorem *B5*: (*FC-B* $\wedge$ *FC-IV* $\wedge$ *FC-D*) $\longleftrightarrow$ (*FC-T* $\wedge$ *FC-V*) **using** *H1 H2 H3 H4 H5* **by** *satx*

8.0.7 IT5 $\Longleftrightarrow$ ID4B5

theorem *B6*: (*FC-T* $\wedge$ *FC-V*) $\longleftrightarrow$ (*FC-B* $\wedge$ *FC-IV* $\wedge$ *FC-D* $\wedge$ *FC-V*) **using** *B5 H2* **by** *satx*

8.0.8 IT5 $\Longleftrightarrow$ IDB5

theorem *B7*: (*FC-B* $\wedge$ *FC-D* $\wedge$ *FC-V*) $\longleftrightarrow$ (*FC-T* $\wedge$ *FC-V*) **using** *H3 H5 B6* **by** *satx*

8.0.9 IKB5 $\Longleftrightarrow$ IK4B5

theorem *B8*: (*FC-IV* $\wedge$ *FC-B* $\wedge$ *FC-V*) $\longleftrightarrow$ (*FC-B* $\wedge$ *FC-V*) **using** *H3* **by** *satx*

8.0.10 IKB5 $\Longleftrightarrow$ IK4B

theorem *B9-a*: (*FC-IV* $\wedge$ *FC-B*) $\longleftrightarrow$ (*FC-B* $\wedge$ *FC-V*) **using** *H3 H4* **by** *satx*

9 Inclusion Relations

In the previous section we proved that some logics are actually the same. Now we want to show which logics differ. Analogous to [4] this thesis concentrates on the backward direction of an edge within the IML cube. The forward direction is always trivial. For example, to show that each theorem of ID45 is a theorem of logic ID5 it is enough to omit axiom 4. Thus, we want to examine whether there are theorems of ID5 that can not be proved in ID45. The notation A > B is used to indicate that in logic A strictly more theorems are provable than in logic B.

The following methodology is mostly adopted from [4]. That paper names several steps which are applied to all edges in the cube. Only one step was omitted because it was rarely possible to prove it.

9.1 Preparation

These three abbreviations are taken over from [4] directly:

abbreviation *one-world-model* :: $\iota \Rightarrow bool$ **where** $\#^1$ *w1* $\equiv \forall x.\ x = w1$
abbreviation *two-world-model* :: $\iota \Rightarrow \iota \Rightarrow bool$ **where** $\#^2$ *w1 w2* $\equiv (\forall x.\ x = w1 \vee x = w2) \wedge w1 \neq w2$
abbreviation *three-world-model* :: $\iota \Rightarrow \iota \Rightarrow \iota \Rightarrow bool$ **where** $\#^3$ *w1 w2 w3* $\equiv (\forall x.\ x = w1 \vee x = w2 \vee x = w3) \wedge w1 \neq w2 \wedge w1 \neq w3 \wedge w2 \neq w3$

They are needed because some logics are only equivalent if the model considered has enough worlds. For example, *two_world_model* forces that there are at least two worlds and that they are not equal. The idea behind Benzmüllers et al. methodology is to determine the minimum number of worlds which fulfil an inclusion relation.

consts *i1*::ι *i2*::ι *i3*::ι

i1, i2 and i3 are worlds. We will use them as arguments for the world-model operators and we also have to activate the Nitpick show constants option, otherwise no information about relations is shown.

nitpick-params [*user-axioms=true*,*format=2*,*max-threads=1*,*show-consts=true*]

To improve the readability the following abbreviations are defined:

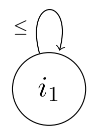
abbreviation *IT5* **where** *IT5* ≡ *FC-V* ∧ *FC-T*
abbreviation *IKB5* **where** *IKB5* ≡ *FC-B* ∧ *FC-V*
abbreviation *IK4* **where** *IK4* ≡ *FC-IV*
abbreviation *IKB* **where** *IKB* ≡ *FC-B*
abbreviation *IK5* **where** *IK5* ≡ *FC-V*
abbreviation *IK45* **where** *IK45* ≡ *FC-IV* ∧ *FC-V*
abbreviation *ID* **where** *ID* ≡ *FC-D*
abbreviation *IDB* **where** *IDB* ≡ *FC-D* ∧ *FC-B*
abbreviation *ID4* **where** *ID4* ≡ *FC-D* ∧ *FC-IV*
abbreviation *IT* **where** *IT* ≡ *FC-T*
abbreviation *IT4* **where** *IT4* ≡ *FC-T* ∧ *FC-IV*
abbreviation *ID5* **where** *ID5* ≡ *FC-D* ∧ *FC-V*
abbreviation *ID45* **where** *ID45* ≡ *FC-D* ∧ *FC-IV* ∧ *FC-V*
abbreviation *ITB* **where** *ITB* ≡ *FC-T* ∧ *FC-B*

9.2 Step A

Usually we would just try to prove the following statement:

lemma: ¬IKB

But as mentioned before, this lemma is only true if a model contains enough worlds. A counterexample for it is:



We need to obtain the information how many worlds a model has to have to make the inclusion true. Therefore, we assume that a false inclusion is valid and apply Nitpick on it. Nitpick will first try to test models with one world, then with two etc.

In the IKB > IK example nitpick can not find a counterexample with one world: For all models, containing only one world, the statement IKB is true. To retrieve the exact arity information, we assumed that each theorem in IKB is also a theorem in IK:

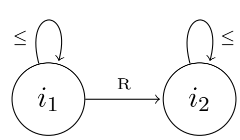
lemma C1-a: IKB

All lemmata produced by applying step A are named $C{*}\text{-}a$.

Nitpick generates a countermodel for each of it, 16 are countermodels with two worlds, 7 with one world and 1 with three worlds. For the IKB > IK example the countermodel found by Nitpick is:

$$R = (\lambda x.\ \text{-})\ ((\iota_1, \iota_1) := \mathit{False}, (\iota_1, \iota_2) := \mathit{True}, (\iota_2, \iota_1) := \mathit{False}, (\iota_2, \iota_2) := \mathit{False})$$
$$le = (\lambda x.\ \text{-})\ ((\iota_1, \iota_1) := \mathit{True}, (\iota_1, \iota_2) := \mathit{False}, (\iota_2, \iota_1) := \mathit{False}, (\iota_2, \iota_2) := \mathit{True})$$

This can be represented as a diagram:



IKB > IK may be valid only for models with two or more worlds because nitpick found a counterexample for the statement IKB $\ngtr$ IK with two wolds. . In Step C we will see how to prove that two is the minimal number of worlds for which the inclusion relation is valid.

9.3 Step B

Knowing how many worlds are needed we can use the abbreviations defined in 9.1 to enforce this number of worlds. In [4] it was possible to use the arity information directly. But when we tried to prove:

$\#^2$ i1 i2 → ¬ IKB

Nitpick found a counterexample. It seems that inclusion is dependent on properties of the R and the ≤ relation. It is sufficient that at least one R and one ≤ relation are existing in those respective frame A > B is valid. That is because a formula is only valid if it is valid in all frames.

Anyway, in [4] the Metis-based integration into Isabelle failed in a few cases. For these cases the authors took another method and used all information Nitpick gave them in its countermodel. In this thesis we will always use all information about the acessibility relation. In the IKB > IK example, we know for example that i1 ≤ i1 is true, whereas i1 ≤ i2 is false.

lemma C1-b: $\#^2$ i1 i2 ∧ ¬i1 R i1 ∧ i1 R i2 ∧ ¬i2 R i1 ∧ ¬i2 R i2 ∧ i1 le i1
∧ ¬i1 le i2 ∧ ¬i2 le i1 ∧ i2 le i2 → ¬ IKB

We named the resulting theorems uniformly C*-C.

In nearly all cases the Metis-based integration into Isabelle failed. However, every theorem except one (lemma C5-e) could be proven by CVC4 instead. With this step we show that the links in the intuitionistic modal cube are indeed correct.

9.4 Step C

Although the inclusions are already shown at this point, we want to determine whether the countermodels Nitpick produced have the minimal number of worlds. In the IKB > IK example we know that a minimum of two worlds is needed. Now we would prove:

lemma C1-c: $\#^1$ i1 → IKB

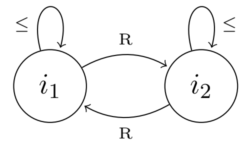
to show that having only one world is not enough. Indeed, in that case the statement IKB > IK is false for all relations R and ≤.

The resulting theorems are uniformly named C*-c. If a counterexample consists of one possible world only, it is not necessary to apply this step.

It is important to not get confused because by the omitted negation. Step B shows a method to prove that there are theorems valid in one logic that are not valid in the other when a certain number of worlds exists. Now we want to show that such theorems do not exist when we decrement the number of worlds by one. Thus, we prove the contrary. That means that there can not be any combination on R and ≤ that gives us a statement like that in Step B (with one world lesser).

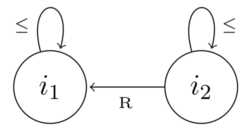
9.5 Proofs of Inclusion Relations

9.5.1 IK4 > IK



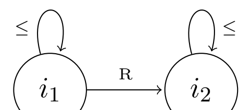
lemma *C1-a*: *IK4* **nitpick oops**
theorem *C1-b*: $(\#^2\ i1\ i2 \wedge i1\ le\ i1 \wedge i2\ le\ i2 \wedge \neg i1\ le\ i2 \wedge \neg i2\ le\ i1$
$\wedge \neg\ i1\ R\ i1 \wedge i1\ R\ i2 \wedge i2\ R\ i1 \wedge \neg i2\ R\ i2\ ) \longrightarrow \neg IK4$ **by** *smt*
lemma *C1-c*: $\#^1\ i1 \longrightarrow IK4$ **by** (*smt FR*)

9.5.2 IK5 > IK



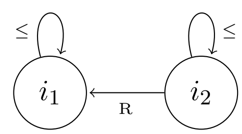
lemma *C2-a*: *IK5* **nitpick oops**
theorem *C2-b*: $(\#^2\ i1\ i2 \wedge \neg\ i1\ R\ i1 \wedge \neg\ i1\ R\ i2 \wedge i2\ R\ i1 \wedge \neg\ i2\ R\ i2$
$\wedge i1\ le\ i1 \wedge \neg\ i1\ le\ i2 \wedge \neg\ i2\ le\ i1 \wedge i2\ le\ i2\ ) \longrightarrow \neg IK5$ **by** *smt*
lemma *C2-c*: $\#^1\ i1 \longrightarrow IK5$ **by** (*smt FR*)

9.5.3 IKB > IK



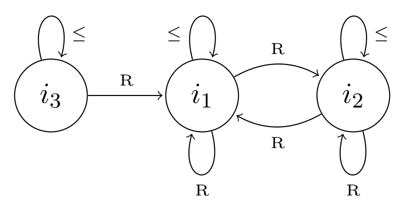
lemma *C3-a*: *IKB* **nitpick oops**
theorem *C3-b*: $(\#^2\ i1\ i2 \wedge \neg\ i1\ R\ i1 \wedge i1\ R\ i2 \wedge \neg\ i2\ R\ i1 \wedge \neg\ i2\ R\ i2$
$\wedge i1\ le\ i1 \wedge \neg\ i1\ le\ i2 \wedge \neg\ i2\ le\ i1 \wedge i2\ le\ i2) \longrightarrow \neg IKB$ **by** *smt*
lemma *C3-c*: $\#^1\ i1 \longrightarrow IKB$ **by** (*smt FR*)

9.5.4 IK45 > IK4



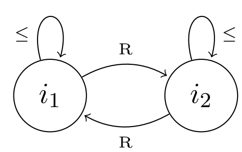
lemma *C4-a*: $IK4 \longrightarrow IK45$ **nitpick oops**
theorem *C4-b*: $(\#^2\ i1\ i2 \wedge \neg\ i1\ R\ i1 \wedge \neg\ i1\ R\ i2 \wedge i2\ R\ i1 \wedge \neg\ i2\ R\ i2$
$\wedge i1\ le\ i1 \wedge \neg\ i1\ le\ i2 \wedge \neg\ i2\ le\ i1 \wedge i2\ le\ i2) \longrightarrow \neg(IK4 \longrightarrow IK45)$ **by** *smt*
lemma *C4-c*: $\#^1\ i1 \longrightarrow (IK4 \longrightarrow IK45)$ **by** *smt*

9.5.5 IK45 > IK5



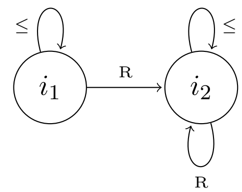
lemma *C5-a*: $IK5 \longrightarrow IK45$ **nitpick oops**
lemma *C5-d*: $(\#^3\ i1\ i2\ i3 \wedge i1\ R\ i1 \wedge i2\ R\ i2 \wedge \neg\ i3\ R\ i3 \wedge i1\ R\ i2$
$\wedge i2\ R\ i1\ \wedge \neg\ i2\ R\ i3 \wedge \neg\ i3\ R\ i2 \wedge \neg i1\ R\ i3 \wedge i3\ R\ i1 \wedge i1\ le\ i1$
$\wedge i2\ le\ i2 \wedge i3\ le\ i3 \wedge \neg\ i1\ le\ i2 \wedge \neg\ i2\ le\ i1 \wedge \neg\ i2\ le\ i3 \wedge \neg\ i3\ le\ i2$
$\wedge \neg\ i1\ le\ i3 \wedge \neg\ i3\ le\ i1) \longrightarrow \neg(IK5 \longrightarrow IK45)$ **by** *smt*
lemma *C5-e*: $(\#^2\ i1\ i2) \longrightarrow (IK5 \longrightarrow IK45)$ **sorry**

9.5.6 IKB5 > IKB



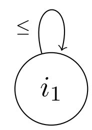
lemma *C6-a*: $IKB \longrightarrow IKB5$ **nitpick oops**
theorem *C6-b*: $(\#^2\ i1\ i2 \wedge \neg\ i1\ R\ i1 \wedge i1\ R\ i2 \wedge i2\ R\ i1 \wedge \neg\ i2\ R\ i2$
$\wedge i1\ le\ i1 \wedge \neg\ i1\ le\ i2 \wedge \neg\ i2\ le\ i1 \wedge i2\ le\ i2) \longrightarrow \neg(IKB \longrightarrow IKB5)$ **by** *smt*
lemma *C6-c*: $\#^1\ i1 \longrightarrow (IKB \longrightarrow IKB5)$ **by** (*smt FR*)

9.5.7 IKB5 > IK45



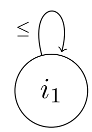
lemma *C7-a*: *IK45* ⟶ *IKB5* **nitpick oops**
theorem *C7-b*: ($\#^2$ *i1 i2* ∧ ¬ *i1 R i1* ∧ *i1 R i2* ∧ ¬ *i2 R i1* ∧ *i2 R i2* ∧ *i1 le i1* ∧ ¬ *i1 le i2* ∧ ¬ *i2 le i1* ∧ *i2 le i2*)⟶¬ (*IK45* ⟶ *IKB5*) **by** *smt*
lemma *C7-c*: $\#^1$ *i1* ⟶ (*IK45* ⟶ *IKB5*) **by** (*smt FR*)

9.5.8 ID > IK



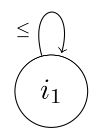
lemma *C8-a*: *ID* **nitpick oops**
theorem *C8-c*: ($\#^1$ *i1* ∧ ¬ *i1 R i1* ∧ *i1 le i1*) ⟶ ¬ *ID* **by** *smt*

9.5.9 ID4 > IK4



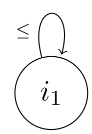
lemma *C9-a*: *IK4* ⟶ *ID4* **nitpick oops**
theorem *C9-b*: ($\#^1$ *i1* ∧ ¬ *i1 R i1* ∧ *i1 le i1*) ⟶ ¬ (*IK4* ⟶ *ID4*) **by** *smt*

9.5.10 ID5 > IK5



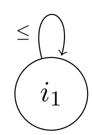
lemma *C10-a*: *IK5* ⟶ *ID5* **nitpick oops**
theorem *C10-b*: ($\#^1$ *i1* ∧ ¬ *i1 R i1* ∧ *i1 le i1*) ⟶ ¬ (*IK5* ⟶ *ID5*) **by** *smt*

9.5.11 ID45 > IK45



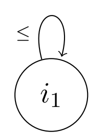
lemma *C11-a*: *IK45* ⟶ *ID45* **nitpick oops**
theorem *C11-b*: ($\#^1$ *i1* ∧ ¬ *i1 R i1* ∧ *i1 le i1*) ⟶ ¬ (*IK45* ⟶ *ID45*) **by** *smt*

9.5.12 IDB > IKB



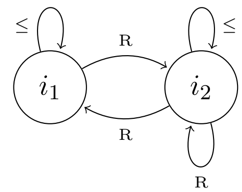
lemma *C12-a*: *IKB* ⟶ *IDB* **nitpick oops**
theorem *C12-b*: ($\#^1$ *i1* ∧ ¬ *i1 R i1* ∧ *i1 le i1*) ⟶ ¬ (*IKB* ⟶ *IDB*) **by** *smt*

9.5.13 IS5 > IKB5



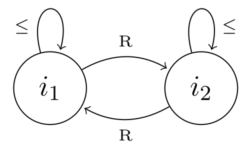
lemma *C13-a*: *IKB5* ⟶ *IT5* **nitpick oops**
theorem *C13-b*: ($\#^1$ *i1* ∧ ¬ *i1 R i1* ∧ *i1 le i1*) ⟶ ¬ (*IKB5* ⟶ *IT5*) **by** *smt*

9.5.14 ID4 > ID



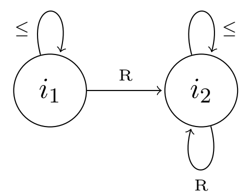
lemma *C14-a*: *ID* $\longrightarrow$ *ID4* **nitpick oops**
theorem *C14-b*: ($\#^2$ *i1 i2* $\wedge$ *i1 R i1* $\wedge$ *i1 R i2* $\wedge$ *i2 R i1* $\wedge$ $\neg$ *i2 R i2* $\wedge$ *i1 le i1* $\wedge$ $\neg$ *i1 le i2* $\wedge$ $\neg$ *i2 le i1* $\wedge$ *i2 le i2*)$\longrightarrow\neg$ (*ID* $\longrightarrow$ *ID4*) **by** *smt*
lemma *C14-c*: $\#^1$ *i1* $\longrightarrow$ (*ID* $\longrightarrow$ *ID4*) **by** (*smt FR*)

9.5.15 ID5 > ID



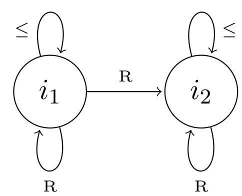
lemma *C15-a*: *ID* $\longrightarrow$ *ID5* **nitpick oops**
theorem *C15-b*: ($\#^2$ *i1 i2* $\wedge$ $\neg$ *i1 R i1* $\wedge$ *i1 R i2* $\wedge$ *i2 R i1* $\wedge$ $\neg$ *i2 R i2* $\wedge$ *i1 le i1* $\wedge$ $\neg$ *i1 le i2* $\wedge$ $\neg$ *i2 le i1* $\wedge$ *i2 le i2*)$\longrightarrow\neg$ (*ID* $\longrightarrow$ *ID5*) **by** *smt*
lemma *C15-c*: $\#^1$ *i1* $\longrightarrow$ (*ID* $\longrightarrow$*ID5*) **by** (*smt FR*)

9.5.16 IDB > ID



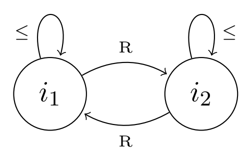
lemma *C16-a*: *ID* $\longrightarrow$ *IDB* **nitpick oops**
theorem *C16-b*: ($\#^2$ *i1 i2* $\wedge$ $\neg$ *i1 R i1* $\wedge$ *i1 R i2* $\wedge$ $\neg$ *i2 R i1* $\wedge$ *i2 R i2* $\wedge$ *i1 le i1* $\wedge$ $\neg$ *i1 le i2* $\wedge$ $\neg$ *i2 le i1* $\wedge$ *i2 le i2*)$\longrightarrow\neg$ (*ID* $\longrightarrow$ *IDB*) **by** *smt*
lemma *C16-c*: $\#^1$ *i1* $\longrightarrow$ (*ID* $\longrightarrow$ *IDB*) **by** *smt*

9.5.17 ID45 > ID4



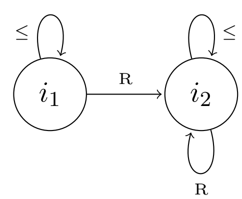
lemma *C17-a*: *ID4* $\longrightarrow$ *ID45* **nitpick oops**
theorem *C17-b*: ($\#^2$ *i1 i2* $\wedge$ *i1 R i1* $\wedge$ *i1 R i2* $\wedge$ $\neg$ *i2 R i1* $\wedge$ *i2 R i2* $\wedge$ *i1 le i1* $\wedge$ $\neg$ *i1 le i2* $\wedge$ $\neg$ *i2 le i1* $\wedge$ *i2 le i2*)$\longrightarrow\neg$ (*ID* $\longrightarrow$ *IDB*) **by** *smt*
lemma *C17-c*: $\#^1$ *i1* $\longrightarrow$ (*ID4* $\longrightarrow$ *ID45*) **by** (*smt FR*)

9.5.18 IT > ID



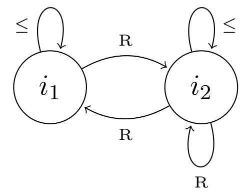
lemma *C18-a*: *ID* $\longrightarrow$ *IT* **nitpick oops**
theorem *C18-b*: ($\#^2$ *i1 i2* $\wedge$ $\neg$ *i1 R i1* $\wedge$ *i1 R i2* $\wedge$ *i2 R i1* $\wedge$ $\neg$ *i2 R i2* $\wedge$ *i1 le i1* $\wedge$ $\neg$ *i1 le i2* $\wedge$ $\neg$ *i2 le i1* $\wedge$ *i2 le i2*)$\longrightarrow\neg$ (*ID* $\longrightarrow$ *IT*) **by** *smt*
lemma *C18-c*: $\#^1$ *i1* $\longrightarrow$ (*ID* $\longrightarrow$ *IT*) **by** *smt*

9.5.19 IS4 > ID4



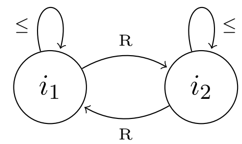
lemma *C19-a*: *ID4* $\longrightarrow$ *IT4* **nitpick oops**
theorem *C19-b*: ($\#^2$ *i1 i2* $\wedge$ $\neg$ *i1 R i1* $\wedge$ *i1 R i2* $\wedge$ $\neg$ *i2 R i1* $\wedge$ *i2 R i2* $\wedge$ *i1 le i1* $\wedge$ $\neg$ *i1 le i2* $\wedge$ $\neg$ *i2 le i1* $\wedge$ *i2 le i2*)$\longrightarrow$ $\neg$ (*ID4* $\longrightarrow$ *IT4*) **by** *smt*
lemma *C19-c*: $\#^1$ *i1* $\longrightarrow$ (*ID4* $\longrightarrow$ *IT4*) **by** *smt*

9.5.20 IS5 > ID45



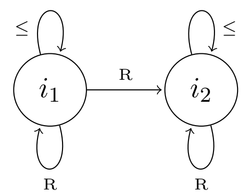
lemma *C20-a*: $ID45 \longrightarrow IT5$ **nitpick oops**
theorem *C20-b*: $(\#^2\ i1\ i2 \wedge \neg\ i1\ R\ i1 \wedge i1\ R\ i2 \wedge \neg\ i2\ R\ i1 \wedge i2\ R\ i2$ $\wedge\ i1\ le\ i1 \wedge \neg\ i1\ le\ i2 \wedge \neg\ i2\ le\ i1 \wedge i2\ le\ i2) \longrightarrow \neg\ (ID45 \longrightarrow IT5)$ **by** *smt*
lemma *C20-c*: $\#^1\ i1 \longrightarrow (ID45 \longrightarrow IT5)$ **by** *smt*

9.5.21 IB > IDB



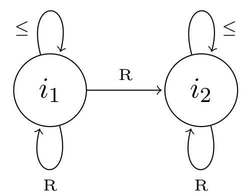
lemma *C21-a*: $IDB \longrightarrow ITB$ **nitpick oops**
theorem *C21-b*: $(\#^2\ i1\ i2 \wedge \neg\ i1\ R\ i1 \wedge i1\ R\ i2 \wedge i2\ R\ i1 \wedge \neg\ i2\ R\ i2$ $\wedge\ i1\ le\ i1 \wedge \neg\ i1\ le\ i2 \wedge \neg\ i2\ le\ i1 \wedge i2\ le\ i2) \longrightarrow \neg\ (IDB \longrightarrow IT5)$ **by** *smt*
lemma *C21-c*: $\#^1\ i1 \longrightarrow (IDB \longrightarrow ITB)$ **by** *smt*

9.5.22 IB > IT



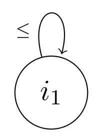
lemma *C22-a*: $IT \longrightarrow IKB$ **nitpick oops**
theorem *C22-b*: $(\#^2\ i1\ i2 \wedge i1\ R\ i1 \wedge i1\ R\ i2 \wedge \neg i2\ R\ i1 \wedge i2\ R\ i2$ $\wedge\ i1\ le\ i1 \wedge \neg\ i1\ le\ i2 \wedge \neg\ i2\ le\ i1 \wedge\ i2\ le\ i2) \longrightarrow \neg\ (IT \longrightarrow IKB)$ **by** *smt*
lemma *C22-c*: $\#^1\ i1 \longrightarrow (IT \longrightarrow IKB)$ **by** *smt*

9.5.23 IS5 > IS4



lemma *C23-a*: $IT4 \longrightarrow IT5$ **nitpick oops**
theorem *C23-b*: $(\#^2\ i1\ i2 \wedge i1\ R\ i1 \wedge i1\ R\ i2 \wedge \neg i2\ R\ i1 \wedge i2\ R\ i2$ $\wedge\ i1\ le\ i1 \wedge \neg\ i1\ le\ i2 \wedge \neg\ i2\ le\ i1 \wedge\ i2\ le\ i2) \longrightarrow \neg\ (IT4 \longrightarrow IT5)$ **by** *smt*
lemma *C23-c*: $\#^1\ i1 \longrightarrow (IT4 \longrightarrow IT5)$ **by** *smt*

9.5.24 IS5 > IKB



lemma *C24-a*: $IKB \longrightarrow IT5$ **nitpick oops**
theorem *C24-b*: $(\#^1\ i1 \wedge \neg\ i1\ R\ i1 \wedge i1\ le\ i1) \longrightarrow \neg\ (IKB \longrightarrow IT5)$ **by** *smt*

10 Comparison of LEO-II and Satallax

For ten theorems used in the verification process a proof was found by Sledgehammer but it could not be reconstructed. Two theorems remained unproven. Interestingly, it made a noticeable difference whether LEO-II or Satallax were used to find the proof. LEO-II found ten of those proofs, Satallax only three. The time limit for both provers was 60 seconds and we limited the statements Sledgehammer used to those which were really necessary. All results can be seen in Table 2. The × symbol denotes the cases in which a timeout occured, the ✓ is used when the prover found a proof.

Clearly, for this setting LEO-II was much more effective than Satallax. All data in the table was obtained from an rather slow 1.7GHz dual-core laptop with 8GB memory. But even when setting the timeout option for Satallax to over 60 seconds, it did mostly not show any improvement. In fact, LEO-II was often ready before the 60 seconds were over.

Table 2: Comparison of the number of proofs found by Satallax and LEO-II

lemma	Satallax	LEO-II
A1-b-1	×	✓
A1-c-2	✓	✓
A2-a-2	One line reconstruction failed but ISAR proof found	✓
A2-c-1	One line reconstruction failed but ISAR proof found	✓
A4-b-2	×	×
A4-c-2	×	✓
A5-a-2	×	✓
A6-c-1	×	✓
A6-c-3	×	✓
H3	×	✓
H4	×	✓
C5-c	×	×

11 Conclusion

In this thesis the intuitionistic modal logic cube was verified. Therefore, an embedding of IML in HOL was presented and used to show alternative axiomatisations and inclusion relationships.

All in all, two theorems could not be proven at all. One of it is not directly necessary for the verification process, the other states the equivalence of an intuitionistic modal axioms and its respective frame condition. For another ten theorems LEO-II found a proof but the integration into Isabelle failed. Further work remains to verify all theorems.

One of the most surprising findings of this thesis is that it was possible to show an equivalence between the classical frame correspondences and the intuitionistic modal axioms! It remains an open issue to find the reasons for this behaviour. If the equivalence is valid, it would be possible to verify the whole intuitionistic modal cube in the same way as in [2].

In summary, it is possible to verify all relationships in the modal logic cube. The methodology proposed in [4] could also be used to verify other cubes.

References

[1] R. Arisaka, A. Das, and L. Straßburger. On nested sequents for constructive modal logics. *arXiv:1505.06896*, 2015.

[2] C. Benzmüller. Verifying the modal logic cube is an easy task (for higher-order automated reasoners). In *Verification, Induction, Termination Analysis*, pages 117–128. Springer, 2010.

[3] C. Benzmüller, C. E. Brown, and M. Kohlhase. Higher-order semantics and extensionality. In *The Journal of Symbolic Logic*, volume 69, pages 1027–1088. Cambridge Univ Press, 2004.

[4] C. Benzmüller, M. Claus, and N. Sultana. Systematic verification of the modal logic cube in Isabelle/HOL. *arXiv:1507.08717*, 2015.

[5] C. Benzmüller, L. C. Paulson, F. Theiss, and A. Fietzke. LEO-II - a cooperative automatic theorem prover for classical higher-order logic (system description). In *International Joint Conference on Automated Reasoning*, pages 162–170. Springer, 2008.

[6] C. Benzmüller and B. Woltzenlogel Paleo. Higher-order modal logics: Automation and applications. In *Reasoning Web International Summer School*, pages 32–74. Springer, 2015.

[7] J. C. Blanchette and T. Nipkow. Nitpick: A counterexample generator for higher-order logic based on a relational model finder. In *International Conference on Interactive Theorem Proving*, pages 131–146. Springer, 2010.

[8] C. E. Brown. Satallax: An automatic higher-order prover. In *International Joint Conference on Automated Reasoning*, pages 111–117. Springer, 2012.

[9] A. Church. A formulation of the simple theory of types. In *The journal of symbolic logic*, volume 5, pages 56–68. Cambridge Univ Press, 1940.

[10] F. B. Fitch. Intuitionistic modal logic with quantifiers. In *Portugaliae mathematica*, volume 7, pages 113–118. Gazeta de matemática, 1948.

[11] L. Henkin. Completeness in the theory of types. In *The Journal of Symbolic Logic*, volume 15, pages 81–91. Cambridge Univ Press, 1950.

[12] J. Hurd. First-order proof tactics in higher-order logic theorem provers. In *Design and Application of Strategies/Tactics in Higher Order Logics, number NASA/CP-2003-212448 in NASA Technical Reports*, pages 56–68, 2003.

[13] K. Kojima. Which classical correspondence is valid in intuitionistic modal logic? In *Logic Journal of IGPL*, volume 20, pages 331–348. Oxford Univ Press, 2012.

[14] S. A. Kripke. Semantical analysis of intuitionistic logic I. volume 40, pages 92–130. Elsevier, 1965.

[15] R. Muskens. Higher order modal logic. volume 3, pages 621–653. Elsevier, 2007.

[16] T. Nipkow, L. C. Paulson, and M. Wenzel. *Isabelle/HOL: a proof assistant for higher-order logic*. Springer Science & Business Media, 2002.

[17] L. C. Paulson and J. C. Blanchette. Three years of experience with sledgehammer, a practical link between automatic and interactive theorem provers. *IWIL-2010*, 1, 2010.

[18] G. Plotkin and C. Stirling. A framework for intuitionistic modal logics. In *Proceedings of the 1986 Conference on Theoretical aspects of reasoning about knowledge*, pages 399–406. Morgan Kaufmann Publishers Inc., 1986.

[19] A. K. Simpson. *The proof theory and semantics of intuitionistic modal logic*. University of Edinburgh. College of Science and Engineering. School of Informatics., 1994.

[20] L. Straßburger. Cut elimination in nested sequents for intuitionistic modal logics. In *International Conference on Foundations of Software Science and Computational Structures*, pages 209–224. Springer, 2013.