

Craig Interpolation of PDL

A report on the proof by Daniel Leivant (1981).

Malvin Gattinger*

Research project with Yde Venema, Winter 2013/14[†]

It seems to be an open question whether Propositional Dynamic Logic (PDL) has *Craig Interpolation*. In fact, it even seems to be an open question if this is an open question: Multiple proofs have been claimed to be found but all of them also have been claimed to be wrong, both more or less publicly.

This report tries to clarify the situation by discussing the proof by Daniel Leivant [11] from 1981 in detail, trying to explain the main methods and ideas. In particular we defend the proof against a criticism expressed by Marcus Kracht [8] that Leivant's proof would only show Craig Interpolation for finitary variants of PDL.

We then try to reproduce the proof and argue that it is correct.

Additionally, we show some basic results about PDL and finitary variants.

*Master of Logic Student, ILLC, Amsterdam. Contact: malvin@w4eg.de

[†]Last update of this file: **2014/10/10, 12:15:28**.

Contents

1 Preliminaries	3
1.1 Syntax and Semantics of Propositional Dynamic Logic	3
1.2 Size and Complexity	4
1.3 Craig's Interpolation Theorem	5
1.4 History	5
1.5 Uniform Interpolation	6
2 Leivant's Proof	6
2.1 Language and Notation	7
2.2 Segerberg's Axioms - The system S	7
2.3 Overview: Leivant's system D	8
2.4 A complete sequent calculus for PDL and PEL	8
2.5 Obtaining a finitary proof system for PDL	9
2.6 Intuitionistic PDL	10
2.7 Overview: The System CD	11
2.8 Counting in S	12
2.9 Proofs from non-initial sequents	12
2.10 Completeness and Closure under Iteration	16
2.11 Linear transformations	20
2.12 Extending Maehara's method for PDL	24
3 PDL and PDL^n	28
3.1 PDL is the intersection of all PDL^n s	29
3.2 Finitary Translations	30
3.3 Can we preserve CI along the chain?	31
4 Concluding Remarks	32
References	32

1 Preliminaries

1.1 Syntax and Semantics of Propositional Dynamic Logic

Definition: Syntax

Fix some set of propositional letters \mathfrak{P} .

The syntax of *Propositional Dynamic Logic* is:

$$\begin{aligned}\phi & ::= p \mid \neg\phi \mid \phi \vee \phi \mid \phi \wedge \phi \mid \phi \rightarrow \phi \mid \langle\alpha\rangle\phi \\ \alpha & ::= a \mid \alpha; \alpha \mid \alpha \cup \alpha \mid \alpha^* \mid \mathbf{1} \mid \mathbf{0}\end{aligned}$$

where $p \in \mathfrak{P}$. We write *PROG* for the set of all programs α and *PDL* for the set of all formulas ϕ according to this definition. The latter should not be confused with **PDL**, the set of all validities which we define later.

We also define the box operators as abbreviations: For any ϕ and α , let $[\alpha]\phi := \neg\langle\alpha\rangle\neg\phi$.

Note that usually also the test-programs $\phi?$ for any formula ϕ are included. Any axiomatization of **PDL** without tests can simply be extended using the scheme $\langle\phi?\rangle\psi \leftrightarrow (\phi \rightarrow \psi)$. Still, we have reasons not to bother with tests here: Our main interest is Craig Interpolation as defined below and by now it is well known that **PDL** without tests has this property iff **PDL** with tests has it (Theorem 10.6.2 in [8, p. 495]).

Definition: Semantics

The semantics of PDL can be given with Kripke models.

A PDL-model is a tuple $\mathcal{M} = (W, \mathcal{R}, V)$ where

- W is a non-empty set of worlds (also called states)
- $\mathcal{R} = (R_\alpha)_\alpha$ is a family of binary relations on W such that
 - $R_{\alpha;\beta} = R_\alpha; R_\beta$ (consecution)
 - $R_{\alpha \cup \beta} = R_\alpha \cup R_\beta$ (union)
 - $R_{\alpha^*} = (R_\alpha)^*$ (reflexive-transitive closure)
 - $R_1 = \{(s, t) \in W \times W \mid s = t\}$ (identity on W)
 - $R_0 = \emptyset$ (empty relation)

Note that here the symbols $;$, \cup and $*$ are twofold: On the left side of $=$ they are part of the language we are defining. On the right side their meaning is given by our set-theoretic meta-theory.

- $V : \mathfrak{P} \rightarrow \mathcal{P}(W)$ is a valuation function.

We say that such a model is based on the frame (W, \mathcal{R}) .

Definition: Truth

The satisfaction relation \models between pointed models and formulas is defined recursively:

- $\mathcal{M}, w \models p$ iff $w \in V(p)$
- $\mathcal{M}, w \models \neg\phi$ iff $\mathcal{M}, w \not\models \phi$
- $\mathcal{M}, w \models \phi \vee \psi$ iff $\mathcal{M}, w \models \phi$ or $\mathcal{M}, w \models \psi$

- $\mathcal{M}, w \models \phi \wedge \psi$ iff $\mathcal{M}, w \models \phi$ and $\mathcal{M}, w \models \psi$
- $\mathcal{M}, w \models \phi \rightarrow \psi$ iff $\mathcal{M}, w \not\models \phi$ or $\mathcal{M}, w \models \psi$
- $\mathcal{M}, w \models \langle \alpha \rangle \phi$ iff there is a $w' \in W$ such that $wR_\alpha w'$ and $\mathcal{M}, w' \models \phi$.

If we have $\mathcal{M}, w \models \phi$ we say that ϕ is (locally) *true* at w in \mathcal{M} .

When it is clear which \mathcal{M} we are considering, we also write $w \models \phi$.

Definition: Global Truth and Validity

We write $\mathcal{M} \models \phi$ and say ϕ is *globally true* in \mathcal{M} iff it is true in every world of this model. We write $\models \phi$ or say ϕ is *valid* iff ϕ is true in every world in every model, i.e. for all $\mathcal{M} = (W, R, V)$ and for all $w \in W$ we have $\mathcal{M}, w \models \phi$. The set of all valid formulas is **PDL**.

Definition: Consequence

We say that ψ is a *local consequence* of ϕ iff $\models \phi \rightarrow \psi$.

We say that ψ is a *global consequence* of ϕ iff: For all $\mathcal{M} = (W, R, V)$: If $\mathcal{M} \models \phi$, then $\mathcal{M} \models \psi$.

We say that ψ is a *superglobal consequence* of ϕ iff: If $\models \phi$, then $\models \psi$.

Consequence Lemma

Local consequence implies global and superglobal consequence.

Global consequence implies superglobal consequence.

1.2 Size and Complexity

Definition: Size of a formula

The size of a formula is the number of symbols it contains, counting all propositional letters, programs and boolean connectives. For example we have $\text{size}(p) = 1$ and $\text{size}((p \rightarrow [a]q) \rightarrow r) = 6$. The size of a finite set of formulas f is the sum of their sizes. We abbreviate both definitions with $|\phi|$ or $|f|$ respectively.

Definition: Lexicographic Order and Addition

We define the irreflexive lexicographic order $(\mathbb{N} \times \mathbb{N}, <)$ by

$$(a, b) < (c, d) : \iff (a < c \vee (a = c \wedge b < d))$$

where $<$ is the irreflexive order on the natural numbers.

Furthermore, let $(a, b) + (c, d) := (a + c, b + d)$.

Definition: Complexity of a formula

For any *PDL*-formula ϕ , its *complexity* $C(\phi)$ is a pair of numbers of which the sum is the maximum of nested connectives in ϕ where we count $*$ separately in the first coordinate and all other connectives in the second. Formally, we define $C : PDL \cup PROG \rightarrow \mathbb{N} \times \mathbb{N}$ recursively:

$$\begin{array}{ll} C(p) = (0, 1) & C(a) = (0, 1) \\ C(\phi \wedge \psi) = (0, 1) + \max\{C(\phi), C(\psi)\} & C(\sigma; \tau) = (0, 1) + \max\{C(\sigma), C(\tau)\} \\ C(\phi \vee \psi) = (0, 1) + \max\{C(\phi), C(\psi)\} & C(\sigma \cup \tau) = (0, 1) + \max\{C(\sigma), C(\tau)\} \\ C(\phi \rightarrow \psi) = (0, 1) + \max\{C(\phi), C(\psi)\} & C(\sigma^*) = (1, 0) + C(\sigma) \\ C(\langle \tau \rangle \phi) = C(\tau) + C(\phi) & \end{array}$$

where \max denotes a maximum according to the lexicographic order.

Examples

- $C(\langle a \rangle p \wedge q) = (0, 3)$
- $C((r \rightarrow \langle a; b \rangle p) \wedge \langle b^* \rangle q) = (1, 2)$
- $C(\langle \langle (a; b)^*; c \rangle^* \rangle p \wedge \langle a^* \rangle (p \rightarrow (q \wedge \langle d \rangle r))) = (0, 1) + \max\{(2, 4), (1, 5)\} = (2, 5)$

Furthermore, the *boolean complexity* C_b of a formula is the maximum number of nested boolean connectives. The *modal complexity* C_m of a formula is the maximum number of nested programs in nested modal operators. Formal definitions for these two can be obtained by replacing 1s with 0s in the left or right column above respectively.

Because the lexicographic order is well-founded, via C it also induces a well-order on all *PDL*-formulas and allows us to prove claims about all formulas ϕ by induction on their $C(\phi)$.

Our definition of complexity is mainly a preparation for proofs in section 3. It will enable us to apply induction hypotheses to formulas with more nested connectives than the the current one as long as they contain less nested stars.

1.3 Craig's Interpolation Theorem

Definition: Language of a formula

For any *PDL*-formula ϕ , its language $L(\phi)$ is the set of all atomic propositions and programs occurring in it. Formally, we define recursively:

$$\begin{array}{ll}
 L(p) = \{p\} & L(a) = \{a\} \\
 L(\phi \wedge \psi) = L(\phi) \cup L(\psi) & L(\sigma; \tau) = L(\sigma) \cup L(\tau) \\
 L(\phi \vee \psi) = L(\phi) \cup L(\psi) & L(\sigma \cup \tau) = L(\sigma) \cup L(\tau) \\
 L(\phi \rightarrow \psi) = L(\phi) \cup L(\psi) & L(\sigma^*) = L(\sigma) \\
 L(\langle \tau \rangle \phi) = L(\tau) \cup L(\phi) &
 \end{array}$$

Definition: Craig Interpolation

A logic, given as the set of its valid formulas Λ has *Craig Interpolation* (*CI*) iff for any two formulas ϕ and ψ such that $\phi \rightarrow \psi \in \Lambda$, there is a formula μ such that:

- $L(\mu) \subseteq L(\phi) \cap L(\psi)$,
- $\phi \rightarrow \mu \in \Lambda$
- and $\mu \rightarrow \psi \in \Lambda$.

In this case we also call μ an *interpolant* for $\phi \rightarrow \psi$.

1.4 History

At least the following attempts have been made to prove that **PDL** has *CI*.

- Daniel Leivant in a paper from 1981 [11],
- Manfred Borzeczowski in his Diplomarbeit from 1988 [3],
- Tomasz Kowalski in a paper from 2002 [6].

Leivant and Borzechowski are both aiming for a proof theoretic argument, using a sequent calculus and a tableaux system respectively. Kowalski on the other hand tried to employ duality results about free dynamic algebras, but his paper has been retracted in 2004 [7] after Yde Venema pointed out a flaw. Unfortunately there is no publication describing the problem or how it could be fixed.

Also the other two proofs by Leivant and Borzechowski have been criticized and the common opinion in the field seems to be that the question is still open. For example, see the chapter “The Unanswered Question” by Marcus Kracht [8, p. 493ff], the concluding remarks in Chapter 4 of [5, p. 99] or take the following quote from a paper by Johan van Benthem:

“The interpolation theorem has been claimed for [PDL] several times since the 1970s, including published papers in the Journal of Symbolic Logic, but so far, no proof has stood up to scrutiny.”[2, p. 456]

1.5 Uniform Interpolation

To avoid confusion we distinguish Craig Interpolation from the stronger property called Uniform Interpolation which is *not* an open question for **PDL** any more.

Definition

A logic Λ has *Uniform Interpolation* iff for all formulas ϕ and all subsets $A \subseteq L(\phi)$ of the language of ϕ there is a formula θ_A such that:

- $L(\theta_A) \subseteq A$
- $\phi \rightarrow \theta_A \in \Lambda$
- For all ψ such that $L(\phi) \cap L(\psi) \subseteq A$ and $\phi \rightarrow \psi \in \Lambda$ we also have $\theta_A \rightarrow \psi \in \Lambda$.

Note that θ_A only depends on the language of ψ and therefore on less information than μ in the definition of Craig Interpolation above. Hence Uniform Interpolation implies Craig Interpolation. However, this does not provide a method to show that *PDL* has *CI* because it has been shown in [1, p. 320] that uniform interpolation does not hold for *PDL*.

2 Leivant’s Proof

In this section we analyze the proof by Daniel Leivant in [11]. Inspired by the following objection we pay special attention if the proof contains an unwanted switch to a finitary variant of **PDL**.

“Twice a solution has been announced, in [[11]] and [[3]], but in neither case was it possible to verify the argument. The argument of Leivant makes use of the fact that if $\phi \vdash_{\mathbf{PDL}} \psi$ then we can bound the size of a possible countermodel so that the star a^* only needs to search up to a depth d which depends on ϕ and ψ . Once that is done, we have reduced **PDL** to **EPDL** which definitely has interpolation because it is a notational variant of polymodal K.”[8, p. 493]

A part of Leivant’s proof which Kracht might mean here is the finitary rule $*R$ and we discuss it in section 2.5. Because finitary variants of **PDL** are also interesting in isolation, we discuss them separately in section 3. For a proof of Craig Interpolation for polymodal K, see [12].

2.1 Language and Notation

It is important to note that Leivant does not use the full language we defined in section 1.1. For all the systems presented, the only basic boolean connectives are \rightarrow and \neg . The classical systems **D** and **S** still fit to our definitions because we can define the abbreviations $\phi \wedge \psi := \neg(\phi \rightarrow \neg\psi)$ and $\phi \vee \psi := \neg\phi \rightarrow \psi$. Furthermore, Leivant does not include the programs **1** and **0** in the language but uses them in the proof working on linear transformations. This does not pose a problem because one can always find equivalent formulas without these programs.

We adopt the following notational conventions from the original proof in [11].

- X, Y, Z and α, β, γ are variables for (possibly complex) formulas and programs
- f and g are variables for sets of formulas
- In a sequent calculus we write sequents as $f \vdash g$.
- Such a sequent is initial iff $f \cap g \neq \emptyset$.
- A proof is a tree of sequents such that (i) all branches are finite, (ii) the leaves are initial or from a given set of premises and (iii) all connections between nodes are instances of the rules of the given proof system.
- Depending on the rules, the tree might have infinitely many branches.
- The root is called the conclusion and proved by the tree.
- To define a new rule we annotate it with its name on the left whereas annotations on the right side indicate the application of a rule.

2.2 Segerberg's Axioms - The system **S**

Leivant starts out with a variant of Segerberg's axiomatization of **PDL** (See [13] for the original). The system **S** consists of the following schemes and rules.

Axiom schemes

- A1 All propositional tautologies.
- A2 $[\alpha]\top$
- A3 $[\alpha](X \wedge Y) \leftrightarrow [\alpha]X \wedge [\alpha]Y$
- A4 $[\alpha; \beta]X \leftrightarrow [\alpha][\beta]X$
- A5 $[\alpha \cup \beta]X \leftrightarrow [\alpha]X \wedge [\beta]X$
- A6 $[\alpha^*]X \leftrightarrow X \wedge [\alpha][\alpha^*]X$

Inference rules

- R1 If $\vdash X$ and $\vdash X \rightarrow Y$, then $\vdash Y$
(Detachment)
- R2 If $\vdash X \rightarrow Y$, then $\vdash [\alpha]X \rightarrow [\alpha]Y$
(Generalization)
- R3 If $\vdash X \rightarrow [\alpha]X$, then $\vdash X \rightarrow [\alpha^*]X$
(Induction)

Note that we removed the axiom scheme “A7 $[?X]Y \leftrightarrow (X \rightarrow Y)$ ” because we are not concerned with tests.

It is easy to check that **S** is sound for the Kripke semantics described above. One main goal of Leivant is to show the completeness of **S**.

2.3 Overview: Leivant's system D

Leivant presents the following sequent calculus called D. The most notable feature is the rule $\ast\text{R}$ which takes ω many premises. Moreover, note the absence of the cut rule.

$$\begin{array}{ll}
(\neg\text{R}) \frac{f, X \vdash g}{f, \vdash g, \neg X} & (\neg\text{L}) \frac{f \vdash g, X}{f, \neg X \vdash g} \\
(\rightarrow\text{R}) \frac{f, X \vdash g, Y}{f \vdash g, X \rightarrow Y} & (\rightarrow\text{L}) \frac{f \vdash g, X \quad f, Y \vdash g}{f, X \rightarrow Y \vdash g} \\
(; \text{R}) \frac{f \vdash g, [\alpha][\beta]X}{f \vdash g, [\alpha; \beta]X} & (; \text{L}) \frac{f, [\alpha][\beta]X \vdash g}{f, [\alpha; \beta]X \vdash g} \\
(\cup\text{R}) \frac{f \vdash g, [\alpha]X \quad f \vdash g, [\beta]X}{f \vdash g, [\alpha \cup \beta]X} & (\cup\text{L}) \frac{f, [\alpha]X, [\beta]X \vdash g}{f, [\alpha \cup \beta]X \vdash g} \\
(\ast\text{L}) \frac{f, X, [\alpha][\alpha^\ast]X \vdash g}{f, [\alpha^\ast]X \vdash g} & (\ast\text{R}) \frac{f \vdash g, \phi \quad f \vdash g, [\alpha]\phi \quad f \vdash g, [\alpha]^2\phi \quad \dots}{f \vdash g, [\alpha^\ast]\phi} \\
(\text{GEN}) \frac{f \vdash X}{[\alpha]f \vdash [\alpha]X} & (\text{WEAK}) \frac{f \vdash g}{f' \vdash g'} \\
& \text{where } f \subseteq f' \text{ and } g \subseteq g'.
\end{array}$$

2.4 A complete sequent calculus for PDL and PEL

In order to show that D is semantically complete for **PDL**, Leivant presents a variant of **PDL** called **PEL** in which the executions of programs are spelled out. Again, Leivant gives an infinitary sequent calculus called E and shows that **PDL** and **PEL** are equivalent in the following sense.

Proposition [11, 2.2.1]

E is sound for **PDL**: If $f \vdash_E g$ then $\bigwedge f \rightarrow \bigvee g$ is true in every state of every **PEL**- and thus **PDL**-model.

The system **PEL** is used to show that D is complete which then is one of the crucial steps towards the completeness of S and interpolation. For the rest of this report we rely on the following Lemma and Theorem but leave it open to check the proofs in [11, p. 361-363].

Lemma [11, 2.3.3]

The system D is complete for E.

Theorem [11, 2.4.2]

E and D are semantically complete.

2.5 Obtaining a finitary proof system for PDL

A crucial part of the system **D** is the infinitary rule

$$(*R) \frac{f \vdash g, X \quad f \vdash g, [\alpha]X \quad f \vdash g, [\alpha][\alpha]X \quad \dots}{f \vdash g, [\alpha^*]X}$$

where the dots indicate that we have all sequents $f \vdash g, [\alpha]^n X$ for $n \in \mathbb{N}$ as premises.

In his section [11, 2.5] Leivant argues that the proof system **D** can be turned in to a finitary one where proofs are trees of finite length, both horizontally and vertically. It is not clear at this point if the completeness and interpolation proofs depend on this claim and Leivant does not explicitly say in which (infinitary or finitary) system the rest of the paper is supposed to take place. However the rule replacement we will discuss now is a good candidate for what Kracht could have meant in the quote on page 6. Hence it deserves some attention.

It is a well-known result that **PDL** has the finite-model property (FMP).

An upper bound of the model size was found by Fischer and Ladner in [4].

Finite-Model Theorem (See [4, 3.2])

Let ϕ be a satisfiable formula. Then there is a model $\mathcal{M} = (W, \mathcal{R}, V)$ and a world $w \in W$ such that $\mathcal{M}, w \models \phi$ and $|W| \leq 2^{\text{size}(\phi)}$.

Using the Finite-Model Theorem, Leivant shows that the validity of a *PDL*-formula of the form $[\alpha^*]\phi$ already follows from the validity of finitely many formulas of the form $[\alpha^n]\phi$. We make this more explicit in the following theorem.

Theorem (implicit in [11, 2.5])

For any formula ϕ and any program α : If $[\alpha]^n \phi$ is valid for all $n \leq k := 2^{\text{size}([\alpha^*]\phi)}$, then $[\alpha^*]\phi$ is valid. In other words: $[\alpha^*]\phi$ is a superglobal consequence of $\bigwedge_{n \leq k} [\alpha]^n \phi$ where $k := 2^{\text{size}([\alpha^*]\phi)}$.

Proof. We have $\models [\alpha]^n \phi$ for all $n \leq k = 2^{\text{size}([\alpha^*]\phi)}$. In order to reach a contradiction, suppose $[\alpha^*]\phi$ is not valid. Then there is a model which falsifies it at some state. In particular, by [4, 3.2] there is such a model of size k . Fix such a model $\mathcal{M} = (W, \mathcal{R}, V)$, $s \not\models [\alpha^*]\phi$. Then $s \models \neg[\alpha^*]\phi$, thus $s \models \langle a^* \rangle \neg \phi$. Hence there is a chain of states $s = s_0 R_\alpha s_1 R_\alpha \dots R_\alpha s_m \not\models \phi$ for some $m \in \mathbb{N}$. We consider two cases.

- $m \leq k$. The first supposition gives us $s \models [\alpha]^m \phi$. It follows that $s_m \models \phi$.
- $m > k$. Then because $|W| = k$ there must be a subchain of s_0, \dots, s_m which still starts at s_0 and ends at s_m but has maximally k elements. Intuitively this subchain is obtained by “removing loops”. Now the first case applies to the subchain which implies $s_m \models \phi$.

In any case it follows that $s_m \models \phi$, a contradiction!

□

Leivant then claims that the last theorem allows us to replace the infinitary rule **R* in **D** with the following finitary one:¹

¹In [11] the first premise is $[\alpha]X$ instead of X . We consider this a typo.

$$(*R_{\leq}) \frac{f \vdash g, \phi \quad f \vdash g, [\alpha]\phi \quad \cdots \quad f \vdash g, [\alpha]^k \phi}{f \vdash g, [\alpha^*]\phi} \text{ where } k = 2^{|f|+|g|+|\phi|}.$$

This seems strange, because the inference rule then can also be applied for formulas which are not valid but merely follow from f and the negation of g . Hence the previous theorem does not suffice to justify this step for arbitrary f and g .

However, we can prove the following stronger version of the Theorem in which the sets f and g play the appropriate role to settle these worries.

Theorem

For any sets of formulas f and g and any formula ϕ : If $\bigwedge f \rightarrow \bigvee(g \cup \{[\alpha]^n \phi\})$ is valid for all $n \leq k = 2^{|f|+|g|+|\phi|}$, then $\bigwedge f \rightarrow \bigvee(g \cup \{[\alpha^*]\phi\})$ is valid.

Proof. Note that it suffices to consider a case where both f and g contain a single formula, for otherwise we can consider their conjunction and disjunction respectively. Let ψ and χ be these formulas and assume that $\psi \rightarrow \chi \vee [\alpha]^n \phi$ is valid for all $n \leq k = 2^{|\psi|+|\chi|+|\phi|}$. In order to reach a contradiction, suppose that $\psi \rightarrow \chi \vee [\alpha^*]\phi$ is not valid. Then there is a model which falsifies it. In particular by [4, 3.2] there is such a model of size k . Fix such a $\mathcal{M}, w \not\models \psi \rightarrow \chi \vee [\alpha^*]\phi$. Then $\mathcal{M}, w \models \psi$ and $\mathcal{M}, w \not\models \chi$ and $\mathcal{M}, w \not\models [\alpha^*]\phi$. From the first and second and our assumption we get that $\mathcal{M}, w \models [\alpha]^n \phi$ for all $n \leq k$. By the third there is a chain $w = w_0 \xrightarrow{\alpha} \cdots \xrightarrow{\alpha} w_m \not\models \phi$ and again we can consider two cases.

- $m \leq k$. Then we have $w_0 \models [\alpha]^m \phi$. It follows that $w_m \models \phi$.
- $m > k$. Then because our model is of size k there must be a subchain of w_0, \dots, w_m which still starts at w_0 and ends at w_m but has maximally k elements. Then the first case applies to this subchain and it follows that $w_m \models \phi$.

In any case it follows that $w_m \models \phi$, a contradiction!

□

We can conclude that the finitary rule is admissible, i.e. it is not stronger than the original. Furthermore, it is easy to see that the rule is not weaker than the original: Suppose we have a proof using the infinitary rule. Each occurrence can then be replaced with the finitary rule by removing the infinite many premises, keeping only the appropriate finite amount determined by the size of the formulas.

2.6 Intuitionistic PDL

To “kill two birds with one stone”[11, p. 365], namely dealing with both classical and intuitionistic **PDL**, Leivant considers constructive/intuitionistic versions of the systems **S** and **D**, called **CS** and **CD** respectively.

The system **CS** only differs from **S** in the axiom scheme A1, namely it contains only the intuitionistic propositional tautologies. The sequent calculus **CD** is “obtained by restricting the kind of sequents $f \vdash g$ used to ones where $|g| \leq 1$ ”[11, p. 366].² We list all the rules in section 2.7. By standard proof theoretic methods we can show that the intuitionistic and classical systems are related as follows.

²Please note that here $|g|$ is not the size of g as we defined above but just the cardinality of the set g .

Definition

For any X , let X° result from X by inserting $\neg\neg$ in front of atomic subformulas and in front of each logical operator.

Theorem [11, 3.2.1]

X is a theorem of S iff X° is a theorem of CS .

Theorem [11, 3.2.2]

X is a theorem of D iff X° is a theorem of CD .

2.7 Overview: The System CD

The following rules constitute the constructive/intuitionistic system in which Leivant's completeness and interpolation proofs take place. Again we leave out the rules $?R$ and $?L$ for tests and we only have the boolean connectives \neg and \rightarrow . Hence this is not a proof system for constructive **PDL** with all connectives because there "none of the propositional connectives \neg , \wedge , \vee , \rightarrow , is definable in terms of the others"[11, p. 366].

Proofs in CD are very similar to those in D as we described in 2.1, namely trees with the conclusion as their root, initial sequents as leaves and all connections according to the following rules. An additional demand for all rules is $g = \emptyset$ or $g = \{Z\}$ for some formula Z .

$$(\neg R) \frac{f, X \vdash}{f, \vdash \neg X}$$

$$(\neg L) \frac{f \vdash X}{f, \neg X \vdash}$$

$$(\rightarrow R) \frac{f, X \vdash Y}{f \vdash X \rightarrow Y}$$

$$(\rightarrow L) \frac{f \vdash X \quad f, Y \vdash g}{f, X \rightarrow Y \vdash g}$$

$$(; R) \frac{f \vdash [\alpha][\beta]X}{f \vdash [\alpha; \beta]X}$$

$$(; L) \frac{f, [\alpha][\beta]X \vdash g}{f, [\alpha; \beta]X \vdash g}$$

$$(\cup R) \frac{f \vdash [\alpha]X \quad f \vdash [\beta]X}{f \vdash [\alpha \cup \beta]X}$$

$$(\cup L) \frac{f, [\alpha]X, [\beta]X \vdash g}{f, [\alpha \cup \beta]X \vdash g}$$

$$(*L) \frac{f, X, [\alpha][\alpha^*]X \vdash g}{f, [\alpha^*]X \vdash g}$$

$$(*R) \frac{f \vdash \phi \quad f \vdash [\alpha]\phi \quad \dots \quad f \vdash [\alpha]^k\phi}{f \vdash [\alpha^*]\phi}$$

where $k = 2^{|f|+|g|+|\phi|}$

$$(\text{GEN}) \frac{f \vdash X}{[\alpha]f \vdash [\alpha]X}$$

$$(\text{WEAK}) \frac{f \vdash g}{f' \vdash g'}$$

where $f \subseteq f'$ and $g \subseteq g'$.

2.8 Counting in S

As a preparation for the completeness and interpolation proof, Leivant states:

Lemma [11, 4.1.1]

The following schemata are provable in **S** for any formula X and any program α .

- (i) $[(\alpha^w)^*][\alpha]X \rightarrow [\alpha][(\alpha^w)^*]X$
- (ii) $\bigwedge_{k < w} [\alpha^k][(\alpha^w)^*]X \rightarrow [\alpha^*]X$
- (iii) If $s_1, \dots, s_w \in \mathbb{N}$ are distinct modulo w , then

$$\left(\bigwedge_{i < \max\{s_1, \dots, s_k\}} [\alpha]^i X \wedge \bigwedge_{i < k} [\alpha^{s_i}][(\alpha^w)^*]X \right) \rightarrow [\alpha^*]X$$

(We reformulate this formula to clarify that one needs exactly w many s_i . While this is not implied by Leivant's notation, the formula is only valid for this case and he only uses the scheme in such situations.)

Note that Leivant claims this before having (and in order to show!) completeness of **S** for **PDL**. Hence purely syntactic proofs of the schemata are needed at this stage.

To see that (i) and (ii) are in **PDL** and hence the Lemma does not contradict the already given soundness of **S**, note that $\mathbb{N} + 1 = 1 + \mathbb{N}$. and $\bigcup_{k < w} (k + (w \cdot \mathbb{N})) = \mathbb{N}$ respectively. Instances of (iii) can be seen to be in **PDL** because for any s_1, \dots, s_n distinct modulo $n = w$ we have

$$\{x \mid x < \max\{s_1, \dots, s_k\}\} \cup \{s_k + w \cdot n \mid k \in \{1, \dots, w\} \wedge n \in \mathbb{N}\} = \mathbb{N}$$

2.9 Proofs from non-initial sequents

Definition

A proof of $f \vdash X$ from a set F of sequents is a proof-figure of **CD** where all leaves are initial or elements of F . The positive closure of a set f of formulas, denoted by $\text{PC}(f)$, is the smallest set $g \supseteq f$ such that for all formulas X and Y and all programs α and β :

- If $(X \rightarrow Y) \in g$, then $Y \in g$.
- If $[\alpha \cup \beta]X \in g$, then $[\alpha]X \in g$ and $[\beta]X \in g$.
- If $[\alpha]X \in g$, then $X \in g$.
- If $[\alpha^*]X \in g$, then $[\alpha][\alpha^*]X \in g$.
- If $[\alpha; \beta]X \in g$, then $[\alpha][\beta]X \in g$.
- (If $[X?]Y \in g$, then $Y \in g$.)

Note that we do not need clauses for disjunction or conjunction here because these are not part of the language of **CD**. Also note that if $|f| < \omega$, then also $|\text{PC}(f)| < \omega$.

The following two Lemmas play a crucial role in both Leivant's completeness and interpolation proofs. Unfortunately we could not reproduce the exact statement and proof for the first one as it appears in [11]. In particular we wonder why the formula $[\alpha]^r X$ should ever occur in a proof of $f \vdash [\alpha]g$, because **CD** does not have rules to remove boxes again in order to gain something which was not already given before. Furthermore, the application of the Lemma

later on suggests a typo, namely that the the resulting sequent should be $f \vdash [\alpha]^m q$ for any $m > r$. What we state below and refer to in the rest of the report is thus not a quote from [11] but a variant of the Theorem as we think it should be phrased and is applied later on. Both the revised statement for induction loading and the proof are from Yde Venema.

Lemma [11, 4.2.1] (revised)

Suppose P is a CD-proof of $f \vdash [\beta_1] \dots [\beta_k][\alpha]^m q$ from $\{f_i \vdash q\}_i$ where q does not occur in f and all β_i s are subprograms of α . If $f' \vdash [\alpha]^r q$ is a sequent in P under a non-initial leaf, then $PC(f') \subseteq PC(f)$.

Proof. By tree-induction on P , i.e. we distinguish which rule was applied in the last step. Easy cases are $\rightarrow L$, $;$ L, $\cup L$ and $*L$. If the last step is **WEAK**, then it introduces a formula $[\beta_1] \dots [\beta_k][\alpha]^m q$ on the right side which was previously empty. Then $[\alpha]^r q$ can not occur in P because q does not occur in f . The interesting case is $*R$ and we consider two possibilities.

First, suppose $k = 0$, i.e. there are no β_i s. Then $\alpha = \gamma^*$ for some program γ and P ends with:

$$\frac{\frac{P_0}{f \vdash [\gamma^*]^{m-1} q} \quad \frac{P_1}{f \vdash [\gamma][\gamma^*]^{m-1} q} \quad \dots \quad \frac{P_{k'}}{f \vdash [\gamma^{k'}][\gamma^*]^{m-1} q}}{f \vdash [\gamma^*][\gamma^*]^{m-1} q}$$

Now, if $r = m$, then $f' \vdash [\gamma^*]^r$ can only be the last line and $f = f'$. If $r < m$, then a sequent $f' \vdash [\gamma^*]^r$ can only occur in the proof parts P_i which are covered by the induction hypothesis.

Alternatively, w.l.o.g, suppose $k = 1$. Then P ends like this:

$$\frac{\frac{P_0}{f \vdash [\alpha]^m q} \quad \frac{P_1}{f \vdash [\beta][\alpha]^m q} \quad \dots \quad \frac{P_{k'}}{f \vdash [\beta^{k'}][\alpha]^m q}}{f \vdash [\beta^*][\alpha]^m q}$$

Now again a sequent $f' \vdash [\alpha]^r$ can only occur in the proof parts P_i .

□

Definition

We denote the result of substituting X for q in a proof P by $P[X/q]$.

Lemma [11, 4.2.2]

In CD: Suppose P is a proof of $f \vdash [\alpha]^r X$ from $\{f_i \vdash X\}_i$ where $X \notin PC(f)$. Then there is a proof P' of $f \vdash [\alpha]^r q$ from $\{f'_i \vdash q\}_i$ such that $P = P'[X/q]$.³

Intuitively, 4.2.2 tells us that a CD-proof of $f \vdash [\alpha]^r X$ where X is not in the positive closure of f will never take X apart and therefore can be obtained from substituting X for q in a proof of $f \vdash [\alpha]^r q$.

Proof. By tree-induction on P . In the induction step we have to consider all CD-rules (see section 2.7) which can end with a sequent of the form $f \vdash [\alpha]^r X$, namely $\rightarrow L$, $;$ R, $;$ L, $\cup R$, $\cup L$, $*L$, $*R$, **GEN** and **WEAK**. We will treat them all in detail because there are subtle differences between some of them and we can see that the different cases motivate the definition of the positive closure $PC(\cdot)$ above.

³Leivant writes that P' is a proof of $f \vdash [\alpha]q$, but this contradicts the substitution, so we assume $f \vdash [\alpha]^r q$.

For each case we show that the induction hypothesis can be applied to a part of P . To allow this the parts have to end with a sequent in which the consequent is not in the positive closure of the antecedent set. Let $\{f_i \vdash X\}_i$ be the set of premises of P .

1. Suppose P ends with \rightarrow L:

$$\frac{\frac{Q_L}{f \vdash Z} \quad \frac{Q_R}{Y \vdash [\alpha]^r X}}{f, Z \rightarrow Y \vdash [\alpha]^r X} (\rightarrow \text{L})$$

Then by $\text{PC}(Y) \subset \text{PC}(f, Z \rightarrow Y) \not\supset X$ we can apply the induction hypothesis to the part Q_R . This gives us a proof Q'_R of $Y \vdash [\alpha]^r q$ from a subset of $\{f_i \vdash q\}_i$ where q is new such that $Q_R = Q'_R[X/q]$. Hence the following is the desired P' :

$$\frac{\frac{Q_L}{f \vdash Z} \quad \frac{Q'_R}{Y \vdash [\alpha]^r q}}{f, Z \rightarrow Y \vdash [\alpha]^r q} (\rightarrow \text{L})$$

2. Suppose P ends with $;$ R:

$$\frac{Q}{f \vdash [\alpha][\beta][\alpha; \beta]^{r-1} X} (; \text{R})$$

Note that $[\beta][\alpha; \beta]^{r-1} X \notin \text{PC}(f)$ because otherwise we would have $X \in \text{PC}(f)$. Hence the induction hypothesis (for $r = 1$, i.e a formula of the shape $[\alpha]Y$) can be applied to Q and we get a proof Q' of $f \vdash [\alpha]q$ from $\{f_i \vdash q\}_i$ such that:

$$Q = Q'[[\beta][\alpha; \beta]^{r-1} X/q]$$

This substitution can be rewritten into two steps:

$$Q = (Q'[[\beta][\alpha; \beta]^{r-1} q/q]) [X/q]$$

Leaving out $[X/q]$, we can then use $Q'[[\beta][\alpha; \beta]^{r-1} q/q]$ as part of the desired P' :

$$\frac{\frac{Q'[[\beta][\alpha; \beta]^{r-1} q/q]}{f \vdash [\alpha][\beta][\alpha; \beta]^{r-1} q}}{f \vdash [\alpha; \beta]^r q} (; \text{R})$$

3. Suppose P ends with $;$ L:

$$\frac{Q}{\frac{f, [\beta][\gamma]Y \vdash [\alpha]^r X}{f, [\beta; \gamma]Y \vdash [\alpha]^r X}} (; \text{L})$$

Then $\text{PC}(f, [\beta][\gamma]Y) = \text{PC}(f, [\beta; \gamma]Y)$ by definition of $\text{PC}(\cdot)$. Hence $X \notin \text{PC}(f, [\beta][\gamma]Y)$ and thus we can apply the induction hypothesis to Q and obtain a proof Q' such that $Q = Q'[X/q]$. We immediately have P' :

$$\frac{Q'}{\frac{f, [\beta][\gamma]Y \vdash [\alpha]^r q}{f, [\beta; \gamma]Y \vdash [\alpha]^r q}} (; \text{L})$$

4. Suppose P ends with $\cup R$:

$$\frac{\frac{Q_0}{f \vdash [\alpha][\alpha \cup \beta]^{r-1} X} \quad \frac{Q_1}{f \vdash [\beta][\alpha \cup \beta]^{r-1} X}}{f \vdash [\alpha \cup \beta][\alpha \cup \beta]^{r-1} X} (\cup R)$$

This is similar to the $;R$ -case. Note that $[\alpha \cup \beta]^{r-1} X \notin \text{PC}(f)$ because otherwise we would have $X \in \text{PC}(f)$. Hence we can apply the induction hypothesis to Q_0 and Q_1 to obtain Q'_0 and Q'_1 such that

$$Q_0 = Q'_0[[\alpha \cup \beta]^{r-1} X/q] \quad \text{and} \quad Q_1 = Q'_1[[\alpha \cup \beta]^{r-1} X/q]$$

Both substitutions can be split in two steps:

$$Q_0 = (Q'_0[[\alpha \cup \beta]^{r-1} q/q]) [X/q] \quad \text{and} \quad Q_1 = (Q'_1[[\alpha \cup \beta]^{r-1} q/q]) [X/q]$$

Leaving out the $[X/q]$ we have the two needed parts for P' :

$$\frac{\frac{Q'_0[[\alpha \cup \beta]^{r-1} q/q]}{f \vdash [\alpha][\alpha \cup \beta]^{r-1} q} \quad \frac{Q'_1[[\alpha \cup \beta]^{r-1} q/q]}{f \vdash [\beta][\alpha \cup \beta]^{r-1} q}}{f \vdash [\alpha \cup \beta][\alpha \cup \beta]^{r-1} q} (\cup R)$$

5. Suppose P ends with $\cup L$:

$$\frac{Q}{\frac{f, [\beta]Y, [\gamma]Y \vdash [\alpha]^r X}{f, [\beta \cup \gamma]Y \vdash [\alpha]^r X}} (;L)$$

Note that $\text{PC}(f, [\beta]Y, [\gamma]Y) = \text{PC}(f, [\beta \cup \gamma]Y)$ by definition of $\text{PC}(\cdot)$. Hence we also have $X \notin \text{PC}(f, [\beta]Y, [\gamma]Y)$ and thus we can apply the induction hypothesis to Q and obtain a proof Q' such that $Q = Q'[X/q]$. Again, we immediately have P' :

$$\frac{Q'}{\frac{f, [\beta]Y, [\gamma]Y \vdash [\alpha]^r q}{f, [\beta \cup \gamma]Y \vdash [\alpha]^r q}} (;L)$$

6. Suppose P ends with $*L$:

$$\frac{Q}{\frac{f, Y, [\beta][\beta^*]Y \vdash [\alpha]^r X}{f, [\beta^*]Y \vdash [\alpha]^r X}} (;L)$$

Note that $\text{PC}(f, Y, [\beta][\beta^*]Y) = \text{PC}(f, [\beta^*]Y)$ by definition of $\text{PC}(\cdot)$. Hence we also have $X \notin \text{PC}(f, Y, [\beta][\beta^*]Y)$ and thus we can apply the induction hypothesis to Q and obtain a proof Q' such that $Q = Q'[X/q]$. Once more, we immediately have P' :

$$\frac{Q'}{\frac{f, Y, [\beta][\beta^*]Y \vdash [\alpha]^r q}{f, [\beta^*]Y \vdash [\alpha]^r q}} (;L)$$

7. Suppose P ends with $*R$:

$$\frac{\frac{Q_0}{f \vdash [\alpha^*]^{r-1} X} \quad \frac{Q_1}{f \vdash [\alpha][\alpha^*]^{r-1} X} \quad \frac{Q_2}{f \vdash [\alpha]^2[\alpha^*]^{r-1} X} \quad \dots}{f \vdash [\alpha^*][\alpha^*]^{r-1} X} \text{ (*R)}$$

Note that $[\alpha]^k[\alpha^*]^{r-1} X \notin \text{PC}(f)$ for all k because otherwise we would have $X \in \text{PC}(f)$. Hence the induction hypothesis can be applied to all Q_k s. For every $k \in \mathbb{N}$ we get a proof Q'_k of $f \vdash [\alpha^*]^k[\alpha^*]^{r-1} q$ from $\{f_i \vdash q\}_i$ such that $Q_k = Q'_k[[\alpha^*]^{r-1} X/q]$. Using them, we can assemble the desired P' :

$$\frac{\frac{Q'_0}{f \vdash [\alpha^*]^{r-1} q} \quad \frac{Q'_1}{f \vdash [\alpha][\alpha^*]^{r-1} q} \quad \frac{Q'_2}{f \vdash [\alpha]^2[\alpha^*]^{r-1} q} \quad \dots}{f \vdash [\alpha^*][\alpha^*]^{r-1} q} \text{ (*R)}$$

8. Suppose P ends with GEN. Then $f = [\alpha^*]g$ for some set g and P has this shape:

$$\frac{\frac{Q}{g \vdash [\alpha^*]^{r-1} X}}{f \vdash [\alpha^*][\alpha^*]^{r-1} X} \text{ (GEN)}$$

By $X \notin \text{PC}(f) = \text{PC}([\alpha^*]g)$ we also have $X \notin \text{PC}(g)$. Hence the induction hypothesis can be applied to Q , obtaining a Q' such that $Q = Q'[X/q]$ and we can assemble P' :

$$\frac{\frac{Q'}{g \vdash [\alpha^*]^{r-1} q}}{f \vdash [\alpha^*][\alpha^*]^{r-1} q} \text{ (GEN)}$$

9. Suppose P ends with WEAK. Then $f \supseteq g$ for some set g and P has this shape:

$$\frac{\frac{Q}{g \vdash [\alpha^*]^r X}}{f \vdash [\alpha^*]^r X} \text{ (WEAK)}$$

Then we also have $\text{PC}(f) \supseteq \text{PC}(g) \not\supseteq X$ and can apply the induction hypothesis to obtain Q' such that $Q = Q'[X/q]$ and P' :

$$\frac{\frac{Q'}{g \vdash [\alpha^*]^r q}}{f \vdash [\alpha^*]^r q} \text{ (WEAK)}$$

□

2.10 Completeness and Closure under Iteration

Using the machinery discussed so far, Leivant gives a new proof of the completeness of **S** for **PDL** via **D**. Although this proof is not our main interest we will discuss its main step because it introduces the proof technique also used for showing Interpolation later. In fact, Leivant only refers to the completeness proof at several steps.

Furthermore, Leivant seems to employ the following general property of **CD**: The rules listed in section 2.7 imply that a formula is built “step-by-step”. To clarify this central step in the proof we first single out the following Conjecture. Probably an elegant rephrasing of the Lemma [11, 4.2.1] would imply this additional claim.

Step-by-Step Conjecture

Suppose P is a CD-proof of $f \vdash [\alpha]^n X$. Then P consists of proof parts P_0, \dots, P_n which build up the $[\alpha]$ s “step by step” in the following sense.

- P_0 is a sequence of proofs of the elements of $\{f_j \vdash X\}_{j \in I_0}$ from the empty set, i.e. only using initial sequents as leaves.
- For each k such that $1 \leq k \leq n$ the part P_k is a sequence of proofs of the elements of $\{f_j \vdash [\alpha]^k X\}_{j \in I_k}$ from initial sequents and $\{f_j \vdash [\alpha]^{k-1} X\}_{j \in I_{k-1}}$.
- All f_j are sets of formulas such that $\text{PC}(f_j) \subseteq \text{PC}(f)$.

In the following proofs we adopt Leivant’s notation which depicts this situation as follows.

$$\frac{\frac{\frac{\frac{P_0}{\{f_j \vdash X\}_{j \in I_0}}{P_1}}{\{f_j \vdash [\alpha]X\}_{j \in I_1}}}{P_2}}{\{f_j \vdash [\alpha]^2 X\}_{j \in I_2}}}{\vdots}}{\{f_j \vdash [\alpha]^{n-1} X\}_{j \in I_{n-1}}}{P_n}}{f \vdash [\alpha]^n X}$$

Lemma [11, 4.3.1]

- If P is a proof in CD deriving $f \vdash X$, then $f \vdash_{\text{CS}} X$.
(I.e.: CS is complete for CD.)
- If P is a proof in CD deriving $f \vdash [\alpha]q$ from $\{f_i \vdash q\}_{i \in I}$, and $f_i \vdash_{\text{CS}} X$ for all $i \in I$, then $f \vdash_{\text{CS}} [\alpha]X$.

Proof. By tree-induction on P , simultaneously for (i) and (ii). The base case is trivial because for an initial sequent there is a corresponding one line CS-proof. For the induction step, the critical case is *R for (i). Furthermore, as Leivant writes, treating an arbitrary proof ending with this rule is also sufficient to prove (ii) at the same time.

Suppose P ends with *R. Let M be the number of needed premises according to the finitary rule *R which we discussed in section 2.5. Then there are proofs P_0, \dots, P_M which occur as parts of P :

$$\frac{\frac{P_0}{f \vdash X} \quad \frac{P_1}{f \vdash [\alpha]X} \quad \dots \quad \frac{P_M}{f \vdash [\alpha]^M X}}{f \vdash [\alpha]^* X} (*R)$$

Let $v := 2^{|\text{PC}(f)|}$, $d \in \omega$ such that $[\alpha]^d X \notin \text{PC}(f)$ and $h := 1 + v + d$. W.l.o.g. we can now assume that $h \leq M$.⁴ For the remainder of the proof we will focus on the part P_h together with its conclusion $f \vdash [\alpha]^h X$ as they appear in P :

⁴This is because we know from section 2.5 that the proof system with the finitary and infinitary rule are equivalent.

$$\frac{\frac{P_0}{f \vdash X} \quad \dots \quad \frac{P_h}{f \vdash [\alpha]^h X} \quad \dots \quad \frac{P_M}{f \vdash [\alpha]^M X}}{f \vdash [\alpha^*] X} \quad (*R)$$

Our motivation for the choice of h is that P_h should belong enough to apply both Lemma 4.2.1 and Lemma 4.2.2 to it. By our choice of h , P_h is a proof of $f \vdash [\alpha]^{1+v}[\alpha]^d X$:

$$\frac{P_h}{f \vdash [\alpha]^{1+v}[\alpha]^d X}$$

Now Leivant seems to employ the Step by Step Conjecture from above. In our case $1 + v + d$ many stacked $[\alpha]$ have to be introduced one after the other. In particular we can split the proof below the last occurrences of $[\alpha]^d X$:

$$\frac{\frac{Q_i}{\{f_i \vdash [\alpha]^d X\}_i}}{R}}{f \vdash [\alpha]^{1+v}[\alpha]^d X}$$

Here the $(f_i)_i$ are some sets such that $\text{PC}(f_i) \subseteq \text{PC}(f)$ for all i . Furthermore R (together with the lines above and below) is a proof of $f \vdash [\alpha]^{1+v}[\alpha]^d X$ from $\{f_i \vdash [\alpha]^d X\}_i$. Remember that a proof from any set can still have additional initial leaves. Hence R should not be thought of as a subtree but rather a sequence of subtrees. Because $[\alpha]^d X \notin \text{PC}(f)$, Lemma 4.2.2 tells us that R will not take $[\alpha]^d X$ apart and the whole proof has to be of this form:

$$\frac{\frac{Q_i}{\{f_i \vdash [\alpha]^d X\}_i}}{R'[[\alpha]^d X/q]}}{f \vdash [\alpha]^{1+v}[\alpha]^d X}$$

where q is new and R' is a proof of $f \vdash [\alpha]^{1+v}q$ from $\{f_i \vdash q\}_i$. Note that R' itself by the Step by Step Conjecture has to be of this form:

$$\frac{\frac{\frac{\frac{\frac{\frac{\{f_j \vdash q\}_{j \in I_0}}{R_j^1}}{\{f_j \vdash [\alpha]q\}_{j \in I_1}}{R_j^2}}{\{f_j \vdash [\alpha]^2 q\}_{j \in I_2}}}{\vdots}}{\{f_j \vdash [\alpha]^v q\}_{j \in I_v}}}{R_j^{1+v}}}{f \vdash [\alpha]^{1+v}q}$$

Note that we are still dealing with with branching trees. The proof part R_j^2 for example is a sequence of trees where the leaves are initial or in $\{f_j \vdash [\alpha]q\}_{j \in I_1}$. Moreover for every element of $\{f_j \vdash [\alpha]^2q\}_{j \in I_2}$ there is a tree in R_j^2 which has this element as its root.

By 4.2.1 we have for all $c \leq v$ and all $j \in I_c$ that $f_j \subseteq \text{PC}(f_j) \subseteq \text{PC}(f)$. Hence we get:

$$|\{\{f_j \mid j \in I_c\} \mid c \leq v\}| \leq |\mathcal{P}(\text{PC}(f))| = 2^{|\text{PC}(f)|} = v < v + 1 = |\{c \mid 0 \leq c \leq v\}|$$

Fact (\heartsuit)

For some $m \neq n$ we have $\{f_j \mid j \in I_m\} = \{f_j \mid j \in I_n\}$.

Therefore by reenumeration we can assume $I_m = I_n$.

W.l.o.g. we assume $m < n$. The proof parts R_m, \dots, R_n constitute a proof of $\{f_j \vdash [\alpha]^nq\}_{j \in I_n}$ from $\{f_j \vdash [\alpha]^mq\}_{j \in I_m}$. Now note that these parts add $r := n - m$ many boxes in front of q , but by \heartsuit the premises in their (non-initial) leafs and roots are the same. We call these parts T_j , i.e. for $j \in I_n$ we have that T_j is a proof of $f_j \vdash [\alpha]^r[\alpha]^mq$ from $\{f_{j'} \vdash [\alpha]^mq\}_{j' \in I_m}$.

Note that we also have $r < n < v$.

Moreover, $q \notin \text{PC}(f)$ because it was new. Hence we have by 4.2.1 that T_j also does not take $[\alpha]^mq$ apart, i.e. for all $j \in I_n$ there is a proof T'_j of $f_j \vdash [\alpha]^rp$ from $\{f_{j'} \vdash p\}_{j' \in I_m}$ such that $T_j = T'_j[[\alpha]^mq/p]$.

Now we will apply both induction hypotheses to the T'_j proofs. This is possible because they are shorter than P .

Let $Y = \bigvee_{j \in I_n} \bigwedge f_j$.⁵ Then we immediately have $f_j \vdash_{\text{CS}} Y$ for all $j \in I_n$.

Together with the T'_j s (where $j \in I_n$) this allows us to apply (ii) and we get $f_j \vdash_{\text{CS}} [\alpha]^rY$. Furthermore by disjunction-elimination (which is provable in CS), we get $Y \vdash_{\text{CS}} [\alpha]^rY$ and by the induction rule⁶ of S this implies for all $j \in I_n$:

$$f_j \vdash_{\text{CS}} [(\alpha^r)^*]Y$$

By applying (i) for all $k \in I_m$ we have $f_k \vdash_{\text{CS}} [\alpha]^mX$. Because $I_m = I_n$ also for all $j \in I_n$ we have $f_j \vdash_{\text{CS}} [\alpha]^mX$. Again by disjunction-elimination this implies $Y \vdash_{\text{CS}} [\alpha]^mX$, and by necessitation we get:

$$[(\alpha^r)^*]Y \vdash_{\text{CS}} [(\alpha^r)^*\alpha^m]X$$

Combining the last two results by transitivity of \vdash_{CS} we have:

$$f_j \vdash_{\text{CS}} [(\alpha^r)^*\alpha^m]X$$

Now let $s := h - n$ which means s is the number of boxes that are introduced in the remaining part of P_h , after sequents with the consequent $[\alpha]^nX$ occur. By applying the induction hypothesis (ii) s many times to $f_j \vdash_{\text{CS}} [(\alpha^r)^*\alpha^m]X$ and this remaining part of P_h we obtain:

$$f \vdash_{\text{CS}} [\alpha^s(\alpha^r)^*\alpha^m]X$$

⁵In Leivant's system CD the disjunction should probably be considered an abbreviation for an equivalent formula using negation and conjunction.

⁶See R3 in section 2.2. Note that CS differs in (propositional) axioms but has the same rules as S.

By m many applications of Lemma 4.1.1(i) (see page 12) this implies:

$$f \vdash_{\text{CS}} [\alpha^{s+m}(\alpha^r)^*]X$$

We are almost done. Only the additional α^{s+m} is not wanted and Lemma 4.1.1 (iii) seems to be the obvious way to remove it. To employ it, we just have to repeat the proof r many times with different values of h . This will yield different pairs of s and m at the end.

For reasons not entirely clear to us, Leivant does not proceed like this immediately. Instead of r he chooses a much higher modulus as follows. Let w be the least common multiple of $2, \dots, v$. Then by 4.1.1(i) and $r|w$ (because $r < v$) we also have:

$$f \vdash_{\text{CS}} [\alpha^{s+m}(\alpha^w)^*]X$$

Now we repeat the argument w times to obtain the following CS validities where all $s_x + m_x$ indexed by $x = 1, \dots, w$ are distinct modulo w .

$$\begin{aligned} f \vdash_{\text{CS}} [\alpha^{s_1+m_1}(\alpha^w)^*]X \\ f \vdash_{\text{CS}} [\alpha^{s_2+m_2}(\alpha^w)^*]X \\ \vdots \\ f \vdash_{\text{CS}} [\alpha^{s_w+m_w}(\alpha^w)^*]X \end{aligned}$$

Finally, note that by applying the induction hypothesis (i) to the corresponding P_k s we have $f \vdash_{\text{CS}} [\alpha]^k X$ for all $k \leq \max_x (s_x + m_x)$. Together with the above validities this fits nicely into Lemma 4.1.1(iii) from which we then get $f \vdash_{\text{CS}} [a^*]X$.

□

Theorem [11, 4.3.3]

S is semantically complete.

Proof. Suppose $X \in \mathbf{PDL}$. Then by 2.2.4⁷ X is provable in D . Hence by 3.2.2 it X^o is provable in CD . By 4.3.1⁸ this implies that X^o is provable in CS . Hence by 3.2.1 X is provable in S .

2.11 Linear transformations

To prepare the $*$ -case of the interpolation proof, Leivant discusses linear transformations of matrices and vectors which have *PDL*-programs and formulas as their entries, respectively.

Already Leivant attributes a central idea in this section to Dexter Kozen and we conjecture that by now the following Lemmas could also be obtained from more general results about action algebras and regular expressions. However, we try to reproduce the original proofs from [11] and refer the interested reader to [9] and [10] by Kozen.

⁷Leivant's "2.4.2" seems to be a typo here.

⁸Ditto for "4.4.1".

Notation

We write \vec{X} , \vec{Y} , \vec{Z} for vectors of formulas, for example $\vec{Y} = \langle Y_1, \dots, Y_k \rangle$ and (α) , (β) or (γ) for matrices of programs, e.g.

$$(\beta) = \begin{pmatrix} \beta_{1,1} & \cdots & \beta_{1,k} \\ \vdots & \ddots & \vdots \\ \beta_{k,1} & \cdots & \beta_{k,k} \end{pmatrix}$$

Definition

The linear transformation of a matrix of programs (β) and a vector of formulas $\vec{Y} = \langle Y_1, \dots, Y_k \rangle$ is the right scalar multiplication where the connective \wedge is the additive and the “boxing” map given by $(\beta, Y) \mapsto [\beta]Y$ is the multiplicative operator:

$$(\beta)\vec{Y} = \begin{pmatrix} \beta_{1,1} & \cdots & \beta_{1,k} \\ \vdots & \ddots & \vdots \\ \beta_{k,1} & \cdots & \beta_{k,k} \end{pmatrix} \begin{pmatrix} Y_1 \\ \vdots \\ Y_k \end{pmatrix} := \begin{pmatrix} [\beta_{1,1}]Y_1 \wedge \cdots \wedge [\beta_{1,k}]Y_k \\ \vdots \\ [\beta_{k,1}]Y_1 \wedge \cdots \wedge [\beta_{k,k}]Y_k \end{pmatrix}$$

Definition

The product of two matrices of programs is the product of the two linear transformations where the program connectives $;$ and \cup are the multiplicative and additive operator respectively:

$$\begin{aligned} (\beta)(\gamma) &= \begin{pmatrix} \beta_{1,1} & \cdots & \beta_{1,k} \\ \vdots & \ddots & \vdots \\ \beta_{k,1} & \cdots & \beta_{k,k} \end{pmatrix} \begin{pmatrix} \gamma_{1,1} & \cdots & \gamma_{1,k} \\ \vdots & \ddots & \vdots \\ \gamma_{k,1} & \cdots & \gamma_{k,k} \end{pmatrix} \\ &:= \begin{pmatrix} \beta_{1,1}; \gamma_{1,1} \cup \cdots \cup \beta_{1,k}; \gamma_{k,1} & \cdots & \beta_{1,1}; \gamma_{1,k} \cup \cdots \cup \beta_{1,k}; \gamma_{k,k} \\ \vdots & \ddots & \vdots \\ \beta_{k,1}; \gamma_{1,1} \cup \cdots \cup \beta_{k,k}; \gamma_{k,1} & \cdots & \beta_{k,1}; \gamma_{1,k} \cup \cdots \cup \beta_{k,k}; \gamma_{k,k} \end{pmatrix} \end{aligned}$$

Definition

Two vectors of formulas with k entries are equivalent, written $\vec{X} \equiv \vec{Y}$, iff we have for all $i \leq k$ that $X_i \leftrightarrow Y_i \in \mathbf{PDL}$. Any two $k \times k$ matrices (α) and (β) are equivalent, written $(\alpha) \equiv (\beta)$, iff for all vectors \vec{Y} with k entries we have that $(\alpha)\vec{Y} \equiv (\beta)\vec{Y}$.

Fact

If β and γ are matrices of programs and \vec{Y} is vector of formulas, then $(\beta)((\gamma)\vec{Y}) \equiv ((\beta)(\gamma))\vec{Y}$. This follows directly from the fact that boxes distribute over conjunctions and that conjunctions correspond to the \cup composition of programs as it can be see in the next example.

Example

$$\begin{aligned} \begin{pmatrix} a & b \\ c & d \end{pmatrix} \left(\begin{pmatrix} e & f \\ g & h \end{pmatrix} \begin{pmatrix} p \\ q \end{pmatrix} \right) &= \begin{pmatrix} a & b \\ c & d \end{pmatrix} \left(\begin{pmatrix} [e]p \wedge [f]q \\ [g]p \wedge [h]q \end{pmatrix} \right) \\ &= \begin{pmatrix} [a]([e]p \wedge [f]q) \wedge [b]([g]p \wedge [h]q) \\ [c]([e]p \wedge [f]q) \wedge [d]([g]p \wedge [h]q) \end{pmatrix} \equiv \begin{pmatrix} [(a; e) \cup (b; g)]p \wedge [(a; f) \cup (b; h)]q \\ [(c; e) \cup (d; g)]p \wedge [(b; g) \cup (d; h)]q \end{pmatrix} \\ &= \begin{pmatrix} (a; e) \cup (b; g) & (a; f) \cup (b; h) \\ (c; e) \cup (d; g) & (c; f) \cup (d; h) \end{pmatrix} \begin{pmatrix} p \\ q \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} e & f \\ g & h \end{pmatrix} \begin{pmatrix} p \\ q \end{pmatrix} \end{aligned}$$

Definition

The identity matrix I is given by:

$$I := \begin{pmatrix} 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & & & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & & & 1 & 0 \\ 0 & 0 & \dots & 0 & 1 \end{pmatrix}$$

Lemma [11, 5.2.1]

For every $k \times k$ matrix (β) of programs, there exists a matrix (γ) such that $I + (\beta)(\gamma) \equiv (\gamma)$.

Proof.

The proof is constructive. To keep the notation simple but still point out the main ideas we only consider the case $k = 2$ here. For a fully general but less detailed proof we refer to the original in [11, p. 370f].

Given a matrix $\beta = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ we have to find a matrix $\gamma = \begin{pmatrix} e & f \\ g & h \end{pmatrix}$ such that $I + (\beta)(\gamma) \equiv (\gamma)$ which means we need programs e, f, g, h such that:

$$\begin{pmatrix} 1 \cup (a; e) \cup (b; g) & (a; f) \cup (b; h) \\ (c; e) \cup (d; g) & 1 \cup (c; f) \cup (d; h) \end{pmatrix} \equiv \begin{pmatrix} e & f \\ g & h \end{pmatrix}$$

The first idea for the proof is to demand that the non-diagonal entries of γ are $;$ -multiples of the diagonal entries in the same column. In our simplified case this means there are λ_1 and λ_2 such that that $f = \lambda_1; h$ and $g = \lambda_2; e$. This allows us to rephrase and find solutions for the equivalences in the entries (1, 2) and (2, 1). The second idea for the proof is then to observe that the semantics of $*$ allow us to solve coinductive equations of programs.

From $(a; f) \cup (b; h) = f$ we get $(a; \lambda_1; h) \cup (b; h) = \lambda_1; h$ which is $((a; \lambda_1) \cup b); h = \lambda_1; h$ for which it suffices to have $(a; \lambda_1) \cup (b) = \lambda_1$. Note that $\lambda_1 := a^*; b$ solves this. Hence let $f := a^*; b; h$.

From $(c; e) \cup (d; g) = g$ we get $(c; e) \cup (d; \lambda_2; e) = \lambda_2; e$ which is $(c \cup (d; \lambda_2)); e = \lambda_2; e$ for which it suffices to have $(c \cup (d; \lambda_2)) = \lambda_2$. Note that $\lambda_2 := d^*; c$ solves this. Hence let $g := d^*; c; e$.

It remains to find e and h such that $1 \cup (a; e) \cup (b; g) = e$ and $1 \cup (c; f) \cup (d; h) = h$. But these now become $1 \cup (a; e) \cup (b; (d^*; c); e) = e$ and $1 \cup (c; a^*; b; h) \cup (d; h) = h$ which are equivalent to $1 \cup (a \cup (b; (d^*; c))); e = e$ and $1 \cup ((c; a^*; b) \cup d); h = h$.

Finally, we observe that $e := (a \cup (b; (d^*; c)))^*$ and $h := ((c; a^*; b) \cup d)^*$ solve these conditions and conclude with:

$$\gamma := \begin{pmatrix} (a \cup (b; (d^*; c)))^* & (a^*; b)((c; a^*; b) \cup d)^* \\ (d^*; c)(a \cup (b; (d^*; c)))^* & ((c; a^*; b) \cup d)^* \end{pmatrix}$$

It is noteworthy that all entries of the resulting matrix depend on all entries of the one we started with. This generalizes to the $k \times k$ case and should be kept in mind for applications of the Lemma.

□

Lemma [11, 5.2.2]

If for all n we have $(\beta)^n \neq (\beta)^{n+1}$, then (γ) in the previous Lemma is unique.

In fact, this Lemma can be strengthened to say that for any (β) we can find a unique (γ) such that $(\beta)^* \equiv (\gamma)$. In the case of $(\beta)^n = (\beta)^{n+1}$ we simply let $(\gamma) := (\beta)^n$, otherwise we take the (γ) from Lemma [11, 5.2.2].

But why are we interested in these fixpoint matrices? The key observation from the two previous Lemmas is that the fixpoint of a matrix of programs can itself be taken to be a matrix of programs. Hence we obtain an expressibility result, captured by the following definition and theorem.

Definition

For a fixed vector $\vec{Y} = \langle Y_1, \dots, Y_k \rangle$ and a matrix (β) and any $n < \omega$, let $Y^{(n)} := (\beta)^n \vec{Y}$. We write $Y_i^{(n)}$ for the i -th entry of $Y^{(n)}$.⁹

Applying Leivant's Lemmas 5.2.1 and 5.2.2 in such a situation shows that the infinite iteration of a linear transformation of programs can be expressed in PDL. We show that the PDL language can always express an infinite conjunction of the shape $\bigwedge_{n < \omega} Y_i^{(n)}$.

Fixpoint Expressibility Theorem (implicit in [11, p. 370])

For any k -tuple of formulas Y_1, \dots, Y_k , any $k \times k$ matrix of programs (β) and any $i \leq k$ there is a *PDL*-formula equivalent to the infinite conjunction $\bigwedge_{n < \omega} Y_i^{(n)}$.

Proof. By Lemma 5.2.1 there is a (γ) such that $I + (\beta)(\gamma) = (\gamma)$. Then we obtain the following equalities and equivalences:

$$\begin{aligned}
(\gamma) &= I + (\beta)(\gamma) \\
&= I + (\beta)(I + (\beta)(\gamma)) \\
&= I + (\beta) + (\beta)(\beta)(\gamma) \\
&= I + (\beta) + (\beta)(\beta)(I + (\beta)(\gamma)) \\
&= I + (\beta) + (\beta)(\beta) + (\beta)(\beta)(\beta)(\gamma) \\
&= \dots \qquad \qquad \qquad \equiv I + (\beta) + (\beta)^2 + (\beta)^3 + \dots =: (\beta)^*
\end{aligned}$$

Now note that $\bigwedge_{n < \omega} Y_i^{(n)}$ is equivalent to the i -th entry of $(\beta)^* \vec{Y}$. Let $\vec{Z} := (\gamma) \vec{Y}$ and let Z_i denote its i -th entry. Then $\bigwedge_{n < \omega} Y_i^{(n)} \equiv Z_i$ and Z_i is a *PDL*-formula. □

Example

For $\vec{Y} = \langle p, q \rangle$ and $(\beta) = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ we have:

$$Y^{(1)} = \begin{pmatrix} [a]p \wedge [b]q \\ [c]p \wedge [d]q \end{pmatrix} \quad \text{and} \quad Y^{(2)} = \begin{pmatrix} [a]([a]p \wedge [b]q) \wedge [b]([c]p \wedge [d]q) \\ [c]([a]p \wedge [b]q) \wedge [d]([c]q \wedge [d]q) \end{pmatrix} \quad \text{etc.}$$

⁹The notation $Y_i^{(n)}$ conceals that this formula is of course very dependent on the given β . Also note that we define $Y_i^{(n)}$ directly via matrices. Because matrix and vector multiplication are compatible as summarized in the fact on page 21, the result is still the same as Leivant's iterated definition of Y_i' .

$$\bigwedge_{n < \omega} Y_1^{(n)} = p \wedge [a]p \wedge ([a]p \wedge [b]q) \wedge ([a]([a]p \wedge [b]q) \wedge [b]([c]p \wedge [d]q)) \wedge \dots$$

$$\bigwedge_{n < \omega} Y_2^{(n)} = p \wedge [a]q \wedge ([c]p \wedge [d]q) \wedge ([c]([a]p \wedge [b]q) \wedge [d]([c]p \wedge [d]q)) \wedge \dots$$

Looking back at the end of the proof of 5.2.1 we can directly see that and how these infinite formulas can be expressed in *PDL*. As our matrix (β) is the same as in the proof, we can use the γ from there too. We get:

$$\bigwedge_{n < \omega} Y_1^{(n)} \equiv [(a \cup (b; (d^*; c)))^*]p \wedge [(a^*; b)((c; a^*; b) \cup d)^*]q$$

$$\bigwedge_{n < \omega} Y_2^{(n)} \equiv [(d^*; c)(a \cup (b; (d^*; c)))^*]p \wedge [(c; a^*; b) \cup d)^*]q$$

2.12 Extending Maehara's method for PDL

To prove interpolation for the obtained sequent calculus Leivant uses a well-known syntactical method originally by Maehara which is explained in [14, p. 33-35]. To employ this method, one first has to define the concept of an interpolant for a sequent with partitioned premises.

Definition

A partition of a set f of formulas is a pair g_1, g_2 such that $g_1 \cup g_2 = f$ and $g_1 \cap g_2 = \emptyset$. We write this as $g_1; g_2$. For notational simplicity we leave out the brackets of singletons in partitions. For example $X, f; g, Y$ denotes the partition of $\{X, Y\} \cup f \cup g$ into $f \cup \{X\}$ and $\{Y\} \cup g$.

Definition (from [11, Section 5.1])

Given a sequent $f \vdash X$ and a partition of f into $f^-; f^+$, we say that K is an interpolant for $f^-; f^+ \vdash X$ iff

$$L(K) \subseteq L(f^-) \cap L(f^+, X) \quad \text{and} \quad f^- \vdash K \quad \text{and} \quad f^+, K \vdash X$$

For the connection between these interpolants for partitions and interpolants as they occur in the definition of Craig Interpolation, see Theorem [11, 5.3.2 (i)] at the end of this section.

Lemma [11, 5.3.1]

Let $f^-; f^+$ be any partition of a set of formulas f . Let X be any formula and q an atomic proposition not occurring in f .

- (i) Suppose P is a proof in CD of $f \vdash X$. Then there is an interpolant K for $f^-; f^+ \vdash X$.
- (ii) Suppose P is a proof of $f \vdash [\alpha]q$ from $\{f_i \vdash q\}_{i < k}$ where q does not occur in f . Let $f_i^-; f_i^+$ be the partitions of f_i ($i < k$) induced¹⁰ by $f^-; f^+$. If K_i is an interpolant for $f_i^-; f_i^+ \vdash X$ ($i < k$), then there is an interpolant K of the form $\bigwedge_i [\beta_i]K_i$ for $f^-; f^+ \vdash [\alpha]X$.

Proof. By tree-induction on P , simultaneously for (i) and (ii).

Base case

¹⁰We give a definition of *induced* in the induction step.

If P is of length 1, then it consists of a single sequent $f \vdash X$ which is initial, i.e. we have $f \cap \{X\} \neq \emptyset$ and thus $X \in f$. Now, given a partition of f into $f^-; f^+$ we have $X \in f^-$ or $X \in f^+$. In the first case, let $K := X$. In the second case, let $K := \top$. In both cases it is easy to check that K is an interpolant for $f^-; f^+ \vdash X$. This shows (i).

There is no base case for (ii) because P in (ii) is always of a length greater than 1.

Induction Hypothesis

Suppose that for all partitions we have that (i) and (ii) hold for all proofs shorter than P .

Induction Step

We consider different cases for each rule which is applied at the end of P . Note that for some rules we also have to distinguish different possible partitions because active formulas left of \vdash can be demanded to be in the left or right part of the partition.

In every case the strategy is as follows. Given a partition of the last sequent we choose partitions of the premises. Then by the induction hypothesis we obtain interpolants for the partitioned premises. We have to find those partitions of the premises which yield interpolants out of which we can build an interpolant of the last sequent. We call these partitions *induced* by the given partition and these are what part (ii) talks about.

$\rightarrow L$ Suppose the last step of P is

$$\frac{f \vdash X \quad f, Y \vdash Z}{f, X \rightarrow Y \vdash Z} (\rightarrow L)$$

Now, given an arbitrary partition $f^-; f^+$ of f we have to consider the two possible ways to extend this to a partition of $f, X \rightarrow Y$.

- $f^-, X \rightarrow Y; f^+$

By induction hypothesis we get interpolants for the following two sequents.

For $f^+; f^- \vdash X$ (note the changed partition!) we have a formula K_1 such that

$$L(K_1) \subseteq L(f^+) \cap L(f^-, X) \text{ and } f^+ \vdash K_1 \text{ and } f^-, K_1 \vdash X$$

For $f^-, Y; f^+ \vdash Z$ we have a formula K_2 such that

$$L(K_2) \subseteq L(f^-, Y) \cap L(f^+, Z) \text{ and } f^-, Y \vdash K_2 \text{ and } f^+, K_2 \vdash Z$$

Now we can show that $K_1 \rightarrow K_2$ is an interpolant for $f^-, X \rightarrow Y; f^+ \vdash Z$.

$$\begin{aligned} L(K_1 \rightarrow K_2) &= L(K_1) \cup L(K_2) \\ &\subseteq (L(f^+) \cap L(f^-, X)) \cup (L(f^-, Y) \cap L(f^+, Z)) \\ &\subseteq (L(f^+, Z) \cap L(f^-, X)) \cup (L(f^-, Y) \cap L(f^+, Z)) \\ &= (L(f^-, X) \cap L(f^+, Z)) \cup (L(f^-, Y) \cap L(f^+, Z)) \\ &= (L(f^-, X) \cup L(f^-, Y)) \cap L(f^+, Z) \\ &= L(f^-, X \rightarrow Y) \cap L(f^+, Z) \end{aligned}$$

$$\frac{\frac{f^-, K_1 \vdash X \quad f^-, Y \vdash K_2}{f^-, X \rightarrow Y, K_1 \vdash K_2} (\rightarrow L)}{f^-, X \rightarrow Y \vdash (K_1 \rightarrow K_2)} (\rightarrow R)$$

$$\frac{f^+ \vdash K_1 \quad f^+, K_2 \vdash Z}{f^+, K_1 \rightarrow K_2 \vdash Z} (\rightarrow L)$$

- $f^-; X \rightarrow Y, f^+$

By induction hypothesis we get interpolants for the following two sequents.

For $f^-; f^+ \vdash X$ we have a formula K_1 such that

$$L(K_1) \subseteq L(f^-) \cap L(f^+, X) \text{ and } f^- \vdash K_1 \text{ and } f^+, K_1 \vdash X$$

For $f^-; Y, f^+ \vdash Z$ we have a formula K_2 such that

$$L(K_2) \subseteq L(f^-) \cap L(Y, f^+, Z) \text{ and } f^- \vdash K_2 \text{ and } f^+, Y, K_2 \vdash Z$$

Now we can show that $K_1 \wedge K_2$ is an interpolant for $f^-; X \rightarrow Y, f^+ \vdash Z$. (In Leivant's system which does not contain \wedge we would equivalently choose $\neg(K_1 \rightarrow \neg K_2)$.)

$$\begin{aligned} L(K_1 \wedge K_2) &= L(K_1) \cup L(K_2) \\ &\subseteq (L(f^-) \cap L(f^+, X)) \cup (L(f^-) \cap L(Y, f^+, Z)) \\ &\subseteq (L(f^-) \cap L(f^+, X, Y, Z)) \cup (L(f^-) \cap L(Y, f^+, X, Z)) \\ &= L(f^-) \cap L(f^+, X, Y, Z) \\ &= L(f^-) \cap L(X \rightarrow Y, f^+, Z) \end{aligned}$$

$$\frac{f^- \vdash K_1 \quad f^- \vdash K_2}{f^- \vdash K_1 \wedge K_2} (\wedge R)$$

$$\frac{\frac{\frac{f^+, K_1 \vdash X}{f^+, K_1, K_2 \vdash X} (\text{WEAK}) \quad \frac{f^+, Y, K_2 \vdash Z}{f^+, Y, K_1, K_2 \vdash Z} (\text{WEAK})}{f^+, K_1 \wedge K_2 \vdash X} (\wedge L) \quad \frac{f^+, Y, K_1 \wedge K_2 \vdash Z}{f^+, Y, K_1 \wedge K_2 \vdash Z} (\wedge L)}{f^+, X \rightarrow Y, K_1 \wedge K_2 \vdash Z} (\rightarrow L)$$

- *R This is the main challenge of showing CI for PDL. Suppose the last step of P is (*R). Then there is some M and for all $h \in \{0, \dots, M\}$ there is a proof P_h which is part of P :

$$\frac{\frac{P_0}{f \vdash X} \quad \frac{P_1}{f \vdash [\alpha]X} \quad \dots \quad \frac{P_M}{f \vdash [\alpha]^M X}}{f \vdash [\alpha^*]X} (*R)$$

The good news are that all the active formulas in *R are right of \vdash . Hence, given a partition f^-, f^+ of f it comes natural to use it without further manipulation. Our goal is to find a formula K such that

$$L(K) \subseteq L(f^-) \cap L(f^+, [\alpha^*]X) \text{ and } f^- \vdash K \text{ and } f^+, K \vdash [\alpha^*]X$$

For each $h = 1, \dots, M$ let P_h be the proof of $f \vdash [\alpha]^h X$ occurring in P above this premise.

As in the proof of 4.3.1, by Lemmas 4.2.1, 4.2.2 and the Step by Step Conjecture we can now find a large enough h such that for some m, s and r such that $h = m + r + s$ we can split P_h into the following parts:¹¹

¹¹The substitutions given in [11, p. 372] seem to be wrong. For example the part R_j starts with $[\alpha]^m X$ and does not take this formula apart, not $[\alpha]^r X$. Hence the former has to be substituted for q , not the latter.

$$\frac{\frac{Q_i}{\frac{\{f_i^-; f_i^+ \vdash [\alpha]^m X\}_{i \in I}}{R'_j[[\alpha]^m X/q]}}}{\frac{\{f_j^-; f_j^+ \vdash [\alpha]^{m+r} X\}_{j \in I}}{U'[[\alpha]^{m+r} X/q]}}}{f_i^-; f_i^+ \vdash [\alpha]^{m+r+s} X}$$

By applying the induction hypothesis (i) to the second line we get a vector \vec{K} such that every K_i is an interpolant for $f_i^-; f_i^+ \vdash [\alpha]^m X$.

Furthermore, by using the induction hypothesis (ii) r times, whenever we have a vector \vec{M} such that each M_i is an interpolant for $f_i^-; f_i^+ \vdash Y$ then there is a matrix (β) such that $(\beta)M$ is a vector such that each $((\beta)M)_i$ is an interpolant for $f_i^-; f_i^+ \vdash [\alpha]^r Y$.

Now, for all n , by applying the latter to the former n times we get:

$$f_i^- \vdash ((\beta)^n K)_i \quad \text{and} \quad f_i^+, ((\beta)^n K)_i \vdash [\alpha]^m [\alpha]^{r \times n} X$$

Now we consider two cases how we can reach a fixpoint of (β) .

- Suppose for some n we have $(\beta)^n \equiv (\beta)^{n+1}$. Then we have $(\beta)^n \equiv (\beta)^{n+k}$ for all k . Let $(\gamma) := I + (\beta) + (\beta)^2 + \dots + (\beta)^n$.
- Alternatively, we have for all n that $(\beta)^n \neq (\beta)^{n+1}$. Then we can apply 5.2.2 to obtain a unique (γ) such that $(\gamma) \equiv (\beta)^*$.

In both cases, γ is such that $((\gamma)K)_i$ is an interpolant for $f_i^-; f_i^+ \vdash [\alpha]^m [(\alpha^r)^*] X$:

$$f_i^- \vdash ((\gamma)K)_i \quad \text{and} \quad f_i^+, ((\gamma)K)_i \vdash [\alpha]^m [(\alpha^r)^*] X$$

Now note that U' is a CD-proof of $f^-; f^+ \vdash [\alpha]^s q$ from $\{f_j^-; f_j^+ \vdash q\}_{j \in I}$. Hence by induction hypothesis (ii) and the $((\gamma)K)_i$ s we get an interpolant H_1 for $f^-; f^+ \vdash [\alpha]^{m+s} [(\alpha^r)^*] X$.

(Now Leivant introduces w which “is a sufficiently large number depending only on $f^-; f^+$ ” [11, p. 327]. It is not clear to us where this w comes from or why we can choose it arbitrarily large, hence we try to rephrase and be a bit more explicit in the last steps.)

We note that r could have been chosen as any larger number from the beginning, where the minimum only depended on $f^-; f^+$. Furthermore, the same holds for m and s . Hence we can now fix some r and then repeat the proof for r many sums $m+s$ which are pairwise different modulo r . We write these sums as $m_1 + s_1, \dots, m_r + s_r$. Now for every $k \leq r$ we have an interpolant H_k for $f_i^-; f_i^+ \vdash [\alpha]^{m_r+s_r} [(\alpha^r)^*]$. Letting $K := \bigwedge_{i \in I} H_i$, we can then use 4.1.1 to show that K is an interpolant for $f^-; f^+ \vdash [\alpha^*] X$.

□

It remains to show that Lemma 5.3.1 implies our original definition of Craig Interpolation for CD and thus for **PDL**.

Theorem [11, 5.3.2 (i)]

PDL has Craig Interpolation.

Proof.

Suppose $X \rightarrow Y$ is a theorem of **PDL**.

By Theorem [11, 2.4.2] (see section 2.4) D is complete for **PDL**, hence we have $\vdash_D X \rightarrow Y$.

By 3.2.2 we have $\vdash_{CD} X^o \rightarrow Y^o$ and thus $X^o \vdash_{CD} Y^o$.¹² Now consider the partition $X^o; \emptyset$.

Then by 5.3.1 applied to $X^o; \emptyset \vdash Y^o$ there is a formula Z for which we have

- $L(Z) \subseteq L(X^o) \cap L(\emptyset, Y^o)$,
- $X^o \rightarrow Z \in \mathbf{PDL}$ and $Z \rightarrow Y^o \in \mathbf{PDL}$

Furthermore we have that in **PDL** (and thus in D and in S) both $X^o \equiv X$ and $Y^o \equiv Y$. Also note that $L(X^o) = L(X)$ and $L(Y^o) = L(Y)$ because the formulas only differ in negation symbols. Hence we can see that Z is in fact a Craig interpolant for $X \rightarrow Y$ because we have:

- $L(Z) \subseteq L(X) \cap L(Y)$,
- $X \rightarrow Z \in \mathbf{PDL}$ and $Z \rightarrow Y \in \mathbf{PDL}$

□

3 PDL and \mathbf{PDL}^n

In the passage we mentioned at the beginning of section 2, Marcus Kracht describes a mistake in Leivant's proof. One could read it as accusing the step in which Leivant replaces the ω -rule in D with a finitary one. But as we have seen in section 2.5, the finitary rule is both admissible and strong enough to replace the original. Kracht continues his description of Leivant's mistake as follows.

“However, this is tantamount to the following. Abbreviate by \mathbf{PDL}^n the strengthening of **PDL** by axioms of the form $[a^*]p \leftrightarrow [a^{\leq n}]p$ for all a . Then, by the finite model property of **PDL**, **PDL** is the intersection of the logics \mathbf{PDL}^n . Unfortunately, it is not so that interpolation is preserved under intersection. A counterexample is the logic **G.3** which fails to have interpolation while all proper extensions have interpolation[.]”[8, p. 493]

This remark is not a proof of **PDL** not having interpolation, because we can easily show that intersection does not always break interpolation. In this section we will discuss the ideas from this quote in detail, in order to explore the relation between **PDL** and \mathbf{PDL}^n

¹²Leivant writes “Then $X^o \rightarrow Y^o$ is a theorem of D , by 3.2.2”. This seems to be a typo.

3.1 PDL is the intersection of all \mathbf{PDL}^n s

Definition

- For any set of formulas A we define its semantic closure $\mathbf{SCL}(A) := \{\phi \mid A \models \phi\}$.
- For any $\phi \in \mathbf{PDL}$, $\alpha \in \mathbf{PROG}$ and $n \in \omega$ we define two abbreviations:
 - $\langle \alpha^{\leq n} \rangle \phi := \phi \vee \langle \alpha \rangle \phi \vee \langle \alpha; \alpha \rangle \phi \vee \dots \vee \langle \alpha^n \rangle \phi$
 - $[\alpha^{\leq n}] \phi := \phi \wedge [\alpha] \phi \wedge [\alpha; \alpha] \phi \wedge \dots \wedge [\alpha^n] \phi$
- Remember that \mathfrak{P} is our set of propositional letters. For any $n \in \omega$, let

$$\mathbf{PDL}^n := \mathbf{SCL}(\mathbf{PDL} \cup \{[\alpha^*]p \leftrightarrow [\alpha^{\leq n}]p \mid \alpha \in \mathbf{PROG} \wedge p \in \mathfrak{P}\})$$

Intersection Theorem

$$\mathbf{PDL} = \bigcap_n \mathbf{PDL}^n$$

Proof.

\subseteq : By definition of \mathbf{PDL}^n .

\supseteq : By contraposition. Take any formula $\phi \notin \mathbf{PDL}$. Then there is a model $\mathcal{M} = (W, \mathcal{R}, V)$ and a state $s \in W$ such that $\mathcal{M}, s \not\models \phi$. In particular, by the finite model property, we can assume W to be finite. Let $k = |W|$ and note that for all $p \in \mathfrak{P}$ and $\alpha \in \mathbf{PROG}$, $[\alpha^*]p \leftrightarrow [\alpha^{\leq k}]p$ is globally true in \mathcal{M} . Therefore \mathcal{M} is also a \mathbf{PDL}^k model. Hence $\phi \notin \mathbf{PDL}^k$ and thereby $\phi \notin \bigcap_n \mathbf{PDL}^n$.

□

Chain Theorem

For any $k, n \in \omega$ such that $k < n$ we have $\mathbf{PDL}^k \supseteq \mathbf{PDL}^n$.

In fact we have:

$$\mathbf{PDL}^0 \supseteq \mathbf{PDL}^1 \supseteq \mathbf{PDL}^2 \supseteq \dots \supseteq \mathbf{PDL}$$

Proof. Note that it suffices to show that for any k, p and a we have $[a^*]p \leftrightarrow [a^{\leq k+1}]p \in \mathbf{PDL}^k$.

\leftarrow follows directly from $[a^*]p \leftrightarrow [a^{\leq k}]p \in \mathbf{PDL}^k$.

\rightarrow follows from the $*$ -axiom and $[a^*]p \leftrightarrow [a^{\leq k}]p \in \mathbf{PDL}^k$.¹³

$$\frac{\frac{[a^*]p}{p \wedge [a][a^*]p} (*)}{\frac{p \wedge [a]([a^{\leq k}]p)}{[a^{\leq k+1}]p}} ([a^*]p \rightarrow [a^{\leq k}]p)$$

□

¹³NB: This proof-tree is a sketch, not to be taken as a proof in a sequent calculus.

3.2 Finitary Translations

Definition

For any *PDL*-formula ϕ we define its n -translation¹⁴ $\phi^{[n]}$ where $\phi \mapsto \phi^{[n]}$ is defined by:

$$\begin{array}{ll}
 p \mapsto p & \langle a \rangle \phi \mapsto \langle a \rangle (\phi^{[n]}) \\
 \neg \phi \mapsto \neg (\phi^{[n]}) & \langle \alpha; \beta \rangle \phi \mapsto (\langle \alpha \rangle \langle \beta \rangle \phi)^{[n]} \\
 \phi \wedge \psi \mapsto \phi^{[n]} \wedge \psi^{[n]} & Z \langle \alpha \cup \beta \rangle \phi \mapsto (\langle \alpha \rangle \phi)^{[n]} \vee (\langle \beta \rangle \phi)^{[n]} \\
 \phi \vee \psi \mapsto \phi^{[n]} \vee \psi^{[n]} & \langle \alpha^* \rangle \phi \mapsto (\langle \alpha^{\leq n} \rangle \phi)^{[n]} \\
 \phi \rightarrow \psi \mapsto \phi^{[n]} \rightarrow \psi^{[n]} &
 \end{array}$$

Remember that $[\alpha]\phi$ is just $\neg \langle \alpha \rangle \neg \phi$, so the translation for boxed formulas is defined implicitly.

Examples

$$\begin{array}{ll}
 \phi & \phi^{[2]} \\
 \hline
 [a^*]q & \mapsto \neg(\neg q \wedge \langle a \rangle \neg q \wedge \langle a; a \rangle \neg q) \\
 \langle a \cup b^* \rangle p & \mapsto \langle a \rangle p \vee (p \vee \langle b \rangle p \vee \langle b; b \rangle p) \\
 p \rightarrow \langle b^*; a \rangle q & \mapsto p \rightarrow (\langle a \rangle q \vee \langle b \rangle \langle a \rangle q \vee \langle b; b \rangle \langle a \rangle q)
 \end{array}$$

We now might wonder if $\phi \in \mathbf{PDL} \iff \exists n : \phi^{[n]} \in \mathbf{PDL}^n$ holds. The left to right direction indeed does but we skip the proof as it is implied by a theorem below. On the other hand, the right to left direction can be refuted as follows. Consider $\phi = \langle a^* \rangle p \rightarrow (\neg \langle a \rangle p \rightarrow p)$. Then $\phi^{[1]} = (p \vee \langle a \rangle p) \rightarrow (\neg \langle a \rangle p \rightarrow p)$ and it is easy to show that $\phi^{[1]} \in \mathbf{PDL}$ and thus $\phi^{[1]} \in \mathbf{PDL}^1$. Therefore the right side holds for this ϕ . But for the left side note that $\phi \notin \mathbf{PDL}$ because $\mathcal{M} := (W = 3, R_a = \{(0, 1), (1, 2)\}, V(p) = \{2\}), 0 \models \neg \phi$. Hence the left side does not hold and this direction does not hold in general. Still we can prove the following similar Lemma which is the main ingredient for the Correspondence Theorem.

Lemma

For any $n \in \omega$ and any ϕ we have $(\phi \leftrightarrow \phi^{[n]}) \in \mathbf{PDL}^n$.

Proof. We fix an arbitrary n and do an induction on $C(\phi)$. The base case is trivial as $C(\phi) = (0, 1)$ implies $p = \phi = \phi^{[n]} = p$ for some p from which it follows that $\phi \leftrightarrow \phi^{[n]} = p \leftrightarrow p \in \mathbf{PDL} \subseteq \mathbf{PDL}^n$.

For the induction step, take any formula ϕ and suppose that for all formulas ψ such that $C(\psi) < C(\phi)$ we have $(\psi \leftrightarrow \psi^{[n]}) \in \mathbf{PDL}^n$. Similarly to the base case, the boolean connectives and all programs but the star do not generate any differences in truth conditions between ϕ and $\phi^{[n]}$. It remains to consider the case $\phi = \langle \alpha^* \rangle \psi$ in which our definition of complexity turns out to be very helpful. Let $(a, b) := C(\psi)$ and $(c, d) := C(\alpha)$. Then $C(\phi) = C(\langle \alpha^* \rangle \psi) = (1, 0) + C(\alpha) + C(\psi) = (1 + a + c, b + d)$. We spell out $\phi^{[n]}$:

$$\phi^{[n]} = (\langle \alpha^* \rangle \psi)^{[n]} = (\langle \alpha^{\leq n} \rangle \psi)^{[n]} = \psi^{[n]} \vee (\langle \alpha \rangle \psi)^{[n]} \vee \dots \vee (\langle \alpha^n \rangle \psi)^{[n]}$$

¹⁴Note that these are not translations between two different languages. \mathbf{PDL} and \mathbf{PDL}^n are merely two different logics based on the same language. In particular, not all n -translations are in \mathbf{PDL}^n , only theorems.

Here all the disjuncts are of lower complexity¹⁵ than ϕ :

$$\begin{aligned}
& \max\{C(\psi), \dots, C(\langle \alpha^n \rangle \psi)\} \\
& \max\{(a, b), \dots, (a, b) + (c, d + n)\} \\
= & \max\{(a, b), \dots, (a + c, b + d + n)\} \\
= & (a + c, b + d + n) \\
< & (1 + a + c, b + d) \\
= & C(\phi)
\end{aligned}$$

Hence by the induction hypothesis the disjuncts are \mathbf{PDL}^n -equivalent to their n -translations:

$$\{\psi(n) \leftrightarrow (\psi)^{[n]}, \dots, \langle \alpha^n \rangle \psi \leftrightarrow (\langle \alpha^n \rangle \psi)^{[n]}\} \subseteq \mathbf{PDL}^n$$

This allows us to show $\phi^{[n]} \leftrightarrow \phi \in \mathbf{PDL}^n$, namely $(\langle \alpha^* \rangle \psi)^{[n]} \leftrightarrow \langle \alpha^* \rangle \psi \in \mathbf{PDL}^n$ as follows.

For one direction, suppose $(\langle \alpha^* \rangle \psi)^{[n]}$ is true at some state in some \mathbf{PDL}^n -model. Then one of its disjuncts has to hold, i.e. for some $k \leq n$ we have $(\langle \alpha^k \rangle \psi)^{[n]}$ true there. By $\langle \alpha^k \rangle \psi \leftrightarrow (\langle \alpha^k \rangle \psi)^{[n]} \in \mathbf{PDL}^n$ also $\langle \alpha^k \rangle \psi$ has to hold. By $\mathbf{PDL} \subseteq \mathbf{PDL}^n$ this implies $\langle \alpha^* \rangle \psi$. Hence $(\langle \alpha^* \rangle \psi)^{[n]} \rightarrow \langle \alpha^* \rangle \psi \in \mathbf{PDL}^n$.

For the other direction, suppose $\langle \alpha^* \rangle \psi$ is true at some state in some \mathbf{PDL}^n -model. Then because $\langle \alpha^* \rangle \psi \leftrightarrow \langle \alpha^{\leq n} \rangle \psi \in \mathbf{PDL}^n$ also $\langle \alpha^{\leq n} \rangle \psi$ has to be true there, and thus for some $k \leq n$ we have $\langle \alpha^k \rangle \psi$. Now by $\langle \alpha^k \rangle \psi \leftrightarrow (\langle \alpha^k \rangle \psi)^{[n]} \in \mathbf{PDL}^n$ we also have $(\langle \alpha^k \rangle \psi)^{[n]}$ true there which is one of the disjuncts of $(\langle \alpha^* \rangle \psi)^{[n]}$. Hence $\langle \alpha^* \rangle \psi \rightarrow (\langle \alpha^* \rangle \psi)^{[n]} \in \mathbf{PDL}^n$.

□

Correspondence Theorem

For all PDL -formulas ϕ we have $\phi \in \mathbf{PDL}$ iff $\forall n : \phi^{[n]} \in \mathbf{PDL}^n$.

Proof.

\Rightarrow Fix any $\phi \in \mathbf{PDL}$ and $n \in \omega$. By $\mathbf{PDL} \subseteq \mathbf{PDL}^n$ we have $\phi \in \mathbf{PDL}^n$. By the Lemma we have $\phi \leftrightarrow \phi^{[n]} \in \mathbf{PDL}^n$ and therefore also $\phi^{[n]} \in \mathbf{PDL}^n$.

\Leftarrow Suppose we have $\forall n : \phi^{[n]} \in \mathbf{PDL}^n$. Fix some n . Then by the previous Lemma we have $(\phi \leftrightarrow \phi^{[n]}) \in \mathbf{PDL}^n$ and therefore also $\phi \in \mathbf{PDL}^n$. Because n was arbitrary we have $\forall n : \phi \in \mathbf{PDL}^n$. Then by the Intersection Theorem we have $\phi \in \mathbf{PDL}$.

□

3.3 Can we preserve CI along the chain?

Now, to show that \mathbf{PDL} has interpolation, we do not need that it is preserved under intersection. The statement that CI is preserved under the particular infinite descending chain of \mathbf{PDL}^n 's would suffice. But does this hold and could the results from the previous section be of any help? We finish this chapter with a sketch of what unfortunately seems to be a dead-end.

¹⁵While different bracketing of formulas and programs does not change their truth conditions, it can very well change their complexity, for example we have $C((p \wedge q) \wedge (r \wedge s)) = (0, 3) < (0, 4) = C(((p \wedge q) \wedge r) \wedge s)$. In such cases we tacitly assume the bracketing which yields the lowest complexity.

Given any $\phi \rightarrow \psi \in \mathbf{PDL}$, we have for all k that $\phi \rightarrow \psi \in \mathbf{PDL}^k$. Each \mathbf{PDL}^k has CI because it is a notational variant of multi-modal logic, hence we get for each k a formula γ_k such that γ_k is a \mathbf{PDL}^k -interpolant for $\phi \rightarrow \psi$, i.e.

- $\phi \rightarrow \gamma_k \in \mathbf{PDL}^k$ and $\gamma_k \rightarrow \psi \in \mathbf{PDL}^k$
- $L(\gamma_k) \subseteq L(\phi) \cap L(\psi)$

Furthermore, by the Chain Theorem we have for any $m, n \in \mathbb{N}$ that if $m < n$, then every \mathbf{PDL}^n -interpolant is also a \mathbf{PDL}^m -interpolant. Hence note that we can choose an arbitrarily high k , for example depending on the size of ϕ and ψ .

After fixing a k , in order to reach a contradiction suppose that γ_k is a \mathbf{PDL}^k -interpolant but *not* a \mathbf{PDL} -interpolant. Then $\phi \rightarrow \gamma_k \notin \mathbf{PDL}$ or $\gamma_k \rightarrow \psi \notin \mathbf{PDL}$. Hence there is a \mathbf{PDL} -model \mathcal{M} such that $\mathcal{M} \models \phi \wedge \neg\gamma_k$ or $\mathcal{M} \models \gamma_k \wedge \neg\psi$. In both cases we know that \mathcal{M} can be taken to be finite and furthermore by [4, 3.2] the size $m := |\mathcal{M}|$ only depends on ϕ , γ_k and ψ .

The missing link now seems to be how choosing a k such that a \mathbf{PDL}^m interpolant for $\phi \rightarrow \psi$ helps us to reach a contradiction. Probably, a limit on the size of the interpolants themselves would be useful here. If we could reach a contradiction, CI for \mathbf{PDL} would follow.

4 Concluding Remarks

We finish this report with a different yet still not satisfying view on the question of Craig Interpolation for \mathbf{PDL} . We could show that the criticism by Kracht in [8] does not affect Leivant's proof in [11] insofar that the latter does not contain any implicit or explicit switch to a finitary variant of \mathbf{PDL} .

However, we are still not able to reconstruct the whole proof and settle the question. Maeharas method and the linear transformations seem to be free from problems and by today they also are not a particularity of this proof any more. The main obstacles are Lemma 4.2.1 and the Step-by-Step Conjecture. Also the final steps in the completeness and interpolation proofs (4.3.1 and 5.3.1) deserve further explanation than we could provide so far. Hence we think that further discussion of the proof should focus on these parts. We hope this report contributes towards a renewed discussion, possibly leading to an answer to the age-old question.

References

- [1] Giovanna D'Agostino, Marco Hollenberg: *Logical Questions Concerning the μ -Calculus: Interpolation, Lyndon and Łoś-Tarski*. In: *Journal of Symbolic Logic*, Volume 65, Issue 1, March 2000, pp 310-332.
<http://www.jstor.org/stable/2586539>
- [2] Johan van Benthem: *The many faces of interpolation*. In: *Synthese*, Volume 164, 2008, pp 451-460.
<http://link.springer.com/article/10.1007/s11229-008-9351-5>
- [3] Manfred Borzeczowski: *Tableau-Kalkül für PDL und Interpolation*. Diplomarbeit, Department of Mathematics, FU Berlin, 1988.

- [4] Michael J. Fischer, Richard E. Ladner: *Propositional dynamic logic of regular programs*. In: *Journal of Computer and System Sciences*, Volume 18, Issue 2, April 1979, pp 194-211. [http://dx.doi.org/10.1016/0022-0000\(79\)90046-1](http://dx.doi.org/10.1016/0022-0000(79)90046-1)
- [5] Amélie P. Gheerbrant: *Fixed-point logics on trees*. Dissertation, University of Amsterdam 2010. <http://dare.uva.nl/en/record/360229>
- [6] Tomasz Kowalski: *PDL has Interpolation*. In: *Journal of Symbolic Logic*, Volume 67, Issue 3, September 2002, pp 933-946. <http://www.jstor.org/stable/3648548>
- [7] Tomasz Kowalski: *Retraction note for »PDL has interpolation«*. In: *Journal of Symbolic Logic*, Volume 69, Issue 3, September 2004, p 935. <http://www.jstor.org/stable/30041768>
- [8] Marcus Kracht: *Tools and techniques in modal logic*. Series: *Studies in logic and the foundations of mathematics*, Volume 142, Elsevier, 1999. <http://wwwhomes.uni-bielefeld.de/mkracht/html/tools/book.pdf>
- [9] Dexter Kozen: *A Completeness Theorem for Kleene Algebras and the Algebra of Regular Events*. In: *Information and Computation*, Volume 110, Issue 2, 1994, pp 366–390. <http://dx.doi.org/10.1006/inco.1994.1037>
- [10] Dexter Kozen: *On Action Algebras*. In: Jan van Eijck, Albert Visser (Editors): *Logic and Information Flow*. MIT Press, 1994, pp 78-88. <http://books.google.com/books?isbn=0262220474>
- [11] Daniel Leivant: *Proof theoretic methodology for propositional dynamic logic*. In: *Lecture Notes in Computer Science*, Volume 107, 1981, pp 356-373. http://dx.doi.org/10.1007/3-540-10699-5_111
- [12] Judit Madarász: *The Craig interpolation theorem in multi-modal logics*. In: *Bulletin of the Section of Logic*, Volume 3.24, 1995, pp 147-151. http://www.filozof.uni.lodz.pl/bulletin/pdf/24_3_5.pdf
- [13] Krister Segerberg: *A completeness theorem in the modal logic of programs*. In: *Notices American Mathematical Society*, Volume 24, 1977, A522. Republished in: *Banach Center Publications*, Volume 9.1, 1982, pp 31-46. <https://eudml.org/doc/209235>
- [14] Gaisi Takeuti: *Proof Theory*. North-Holland, Amsterdam/Oxford 1975. <http://www.getcited.org/pub/101589483>