Interpolation and Quantifiers in Ortholattices

Simon Guilloud(✉), Sankalp Gambhir, and Viktor Kunčak

EPFL
School of Computer and Communication Sciences
Lausanne, Switzerland
{firstname.lastname}@epfl.ch

Abstract. We study quantifiers and interpolation properties in *orthologic*, a non-distributive weakening of classical logic that is sound for formula validity with respect to classical logic, yet has a quadratic-time decision procedure. We present a sequent-based proof system for quantified orthologic, which we prove sound and complete for the class of all complete ortholattices. We show that orthologic does not admit quantifier elimination in general. Despite that, we show that interpolants always exist in orthologic. We give an algorithm to compute interpolants efficiently. We expect our result to be useful to quickly establish unreachability as a component of verification algorithms.

1 Introduction

Interpolation-based techniques are important in hardware and software model checking [17, 19, 25, 30–33, 35, 40]. The interpolation theorem for classical propositional logic states that, for two formulas A and B such that $A \implies B$ is valid, there exists a formula I, with free variables among only those common to both A and B, such that both $A \implies I$ and $I \implies B$ are valid. All known algorithms for propositional logic have worst-case exponential size of proofs they construct, which is not surprising given that the validity problem is coNP-hard. Interpolation algorithms efficiently construct interpolants from such exponentially-sized proofs [39], which makes the overall process exponential. It is therefore interesting to explore whether there are logical systems for which proof search and interpolation are polynomial-time in the size of the input formulas.

Orthologic is a relaxation of classical logic, corresponding to the algebraic structure of ortholattices, where the law of distributivity does not necessarily hold, but where the weaker absorption law (V9) does (Table 1). In contrast to classical and intuitionistic logic, where the problem of deciding the validity of a formula is, respectively, coNP-complete and PSPACE-complete, there is a *quadratic-time* algorithm to decide validity in orthologic [14, 16]. Orthologic was first studied as a candidate for quantum logic, where distributivity fails [1, 2]. Due to its advantageous computational properties, orthologic has recently been suggested as a tool to reason about proofs and programs in formal verification in a way that is sound, efficient and predictable [14, 15], even if incomplete.

As a step towards enabling the use of state-of-the-art model checking techniques backed by orthologic, this paper studies interpolation, as well as properties of quantifiers in orthologic. The quantifier elimination property would immediately lead to the existence of interpolants. We show, however, that quantified orthologic does *not* admit quantifier elimination. To do so, we define semantics of quantified orthologic (QOL) using complete ortholattices. Furthermore, we present a natural sequent calculus proof system for QOL that we show to be sound and complete with respect to this semantics. We then consider the question of interpolation. We show that a refutation-based notion of interpolation fails. However, a natural notion of interpolants based on the lattice ordering of formulas yields interpolants in orthologic. Namely, if $A \leq B$ is provable, then there exists an interpolant I such that $A \leq I$ and $I \leq B$, where $\leq$ corresponds to implication. Moreover, these interpolants can be computed efficiently from a proof of $A \leq B$. We expect that this notion of interpolation can be used in future verification algorithms.

V1:	$x \vee y = y \vee x$	V1':	$x \wedge y = y \wedge x$
V2:	$x \vee (y \vee z) = (x \vee y) \vee z$	V2':	$x \wedge (y \wedge z) = (x \wedge y) \wedge z$
V3:	$x \vee x = x$	V3':	$x \wedge x = x$
V4:	$x \vee 1 = 1$	V4':	$x \wedge 0 = 0$
V5:	$x \vee 0 = x$	V5':	$x \wedge 1 = x$
V6:	$\neg\neg x = x$		
V7:	$x \vee \neg x = 1$	V7':	$x \wedge \neg x = 0$
V8:	$\neg(x \vee y) = \neg x \wedge \neg y$	V8':	$\neg(x \wedge y) = \neg x \vee \neg y$
V9:	$x \vee (x \wedge y) = x$	V9':	$x \wedge (x \vee y) = x$

Table 1: Axioms of orthologic, a generalization of classical logic corresponding to the algebraic variety of ortholattices. As lattices, ortholattices admit a partial order $\leq_{OL}$ defined by $a \leq_{OL} b$ iff $a \wedge b = a$ or, equivalently, $a \vee b = b$.

In some cases, ortholattices may be not only a relaxation of propositional logic but a direct intended interpretation of formulas. Indeed, lattices already play a crucial role in abstract interpretation [9, 38] and have been adopted by the Flix programming language [29]. Furthermore, De Morgan bi-semilattices and lattices (generalizations of ortholattices where law V6 of Table 1 does not necessarily hold) have been used to model multivalued logics with undefined states [4, 6]. Lattice automata [26] map final automaton states to elements of a finite distributive De Morgan lattice, which admits a notion of complement, but, in contrast to ortholattices, need not satisfy V7 or V7' of Table 1 (the chain $0 \leq 1/2 \leq 1$ is a finite distributive de Morgan lattice but not an ortholattice).

Proof-theoretic properties of propositional orthologic are presented in [16], but without discussion of interpolation and without the treatment of quantifiers as lattice operators. These topics are the subject of the present paper.

Contributions. We make the following contributions:

1. We define quantified orthologic, in the spirit of QBF, presenting its semantics in terms of validity in all complete ortholattices. We present a proof system for quantified orthologic, which extends an existing polynomial-time proof system for quantifier-free orthologic [16, 41] with rules for quantifier introduction and elimination. We show soundness and completeness of our proof system.
2. We show that quantified orthologic does not admit quantifier elimination. Consequently, quantifiers increase the class of definable relationships between ortholattice elements. This also makes the existence of interpolants a more subtle question than in classical propositional logic, where quantifier elimination alone guarantees that quantifier-free interpolants exist.
3. We consider a refutation-based interpolation property, which reduces to the usual one in classical logic. We show that orthologic does *not* satisfy this variant of interpolation.
4. We consider another notion of interpolation, one which is natural in any lattice-based logic. In the language of ortholattices (Table 2), given two formulas A and B such that $A \leq B$, an interpolant I is a formula such that $A \leq I$, $I \leq B$, and $\mathsf{FV}(I) \subseteq \mathsf{FV}(A) \cap \mathsf{FV}(B)$. While it is known [37] that orthologic admits such interpolants, we show using the sequent calculus proof system for OL that (a generalization of) such interpolants can always be computed efficiently. Specifically, we present an algorithm to compute interpolants from proofs of sequents in time linear in the size of the proof (where finding proofs in orthologic is worst-case quadratic time in the size of the inequality).

The final result yields an end-to-end polynomial-time algorithm that first finds a proof and then computes an interpolant I, where validity of both the input $A \implies B$ and the result $A \implies I$, $I \implies B$ is with respect to OL axioms.

P1:	$x \leq x$		
P2:	$x \leq y \ \& \ y \leq z \implies x \leq y$		
P3:	$0 \leq x$	P3':	$x \leq 1$
P4:	$x \wedge y \leq x$	P5':	$x \leq x \vee y$
P5:	$x \wedge y \leq y$	P6':	$y \leq x \vee y$
P6:	$x \leq y \ \& \ x \leq z \implies x \leq y \wedge z$	P6':	$x \leq z \ \& \ y \leq z \implies x \vee y \leq z$
P7:	$x \leq \neg\neg x$	P7':	$\neg\neg x \leq x$
P8:	$x \leq y \implies \neg y \leq \neg x$		
P9:	$x \wedge \neg x \leq 0$	P9':	$1 \leq x \vee \neg x$

Table 2: Axiomatization of ortholattices in the signature $(S, \leq, \wedge, \vee, 0, 1, \neg)$ as partially ordered sets. & denotes conjunction between atomic formulas of axioms, to differentiate it from the term-level lattice operation $\wedge$.

Preliminaries. We follow the definitions and notation of [16]. An ortholattice is an algebraic variety with language $(\wedge, \vee, \neg, 0, 1)$ and axioms in Table 1. As lattices, ortholattices can be described as a partially ordered set whose order relation, noted $\leq_{OL}$, is defined by $a \leq_{OL} b$ iff $a \wedge b = a$ or equivalently $a \vee b = b$ [22,36]. In both Boolean and Heyting algebras, this order relation corresponds to the usual logical implication. The (equivalent) axiomatization of ortholattices as a poset can be found in Table 2. We denote by $\mathcal{T}_{OL}$ the set of terms built as trees with nodes labelled by either by $(\wedge, \vee, \neg, 0, 1)$ or by symbols in a fixed, countably infinite set of variable symbols $\mathit{Var} = \{x, y, z, ...\}$. This corresponds precisely to the set of propositional formulas. Note that since $\wedge$ and $\vee$ are commutative, children of a node are described for simplicity as an unordered set. In particular, $x \wedge y$ and $y \wedge x$ denote the same term. Since 0 can always be represented as $x \wedge \neg x$, we sometimes omit it from case analysis for brevity, and similarly for 1.

2 Quantified Orthologic: Syntax, Semantics, and a Complete Proof System

We consider the extension of propositional orthologic to quantified orthologic, noted QOL, the analogue of QBF [8] for classical logic, or of System F [13] for intuitionistic logic. To do so, we extend the proof system of [16] by adding axiomatization of an existential quantifier ($\bigvee$) and a universal quantifier ($\bigwedge$). The deduction rules of QOL are in Figure 1. It is folklore that the sequent calculus LK [12] with arbitrarily many formulas on both sides corresponds to classical logic, while, if we restrict the right sides of sequents to only contain at most one formula, we obtain intuitionistic logic [43, section 7.1]. Orthologic exhibits a different natural restriction: sequents can only contain at most two formulas in both sides of the sequent combined. For this reason, it is convenient to represent sequents by decorating the formulas with superscript ^{L} or ^{R}, depending on whether they appear on the left or right side.

Definition 1 (From [16]). If ϕ is a formula, we call ϕ^L and ϕ^R annotated formulas. A *sequent* is a set of at most two annotated formulas. We use uppercase Greek letters (e.g. Γ and Δ) to represent sets that are either empty or contain exactly one annotated formula ($|\Gamma| \leq 1, |\Delta| \leq 1$).

Given formulas ϕ and ψ, we thus write ϕ^L, ϕ^R for a sequent often denoted $\phi \vdash \psi$.

Our use of quantifiers in this paper (quantified orthologic) is different from considering the first-order theory of ortholattices. In particular, the semantic of an existential quantifier ($\bigwedge x.\ t$) in QOL corresponds to the least upper bound of a (possibly infinite) family of *lattice* elements given by values of term t. In contrast, when considering a classical first-order theory of ortholattices, we would build an atomic formula such as $t_1 \leq t_2$, obtaining a definite truth or falsehood in the metatheory, and only then apply quantifiers to build formulas such as $\exists x.(t_1 \leq t_2)$. Such difference also exists in the case of Boolean algebras [24].

$$\dfrac{}{\phi^L, \phi^R}\ \text{Hyp}$$

$$\dfrac{\Gamma, \psi^R \qquad \psi^L, \Delta}{\Gamma, \Delta}\ \text{Cut}$$

$$\dfrac{\Gamma}{\Gamma, \Delta}\ \text{Weaken}$$

$$\dfrac{\Gamma, \phi^L}{\Gamma, (\phi \land \psi)^L}\ \text{LeftAnd} \qquad \dfrac{\Gamma, \phi^R \qquad \Gamma, \psi^R}{\Gamma, (\phi \land \psi)^R}\ \text{RightAnd}$$

$$\dfrac{\Gamma, \phi^L \qquad \Gamma, \psi^L}{\Gamma, (\phi \lor \psi)^L}\ \text{LeftOr} \qquad \dfrac{\Gamma, \phi^R}{\Gamma, (\phi \lor \psi)^R}\ \text{RightOr}$$

$$\dfrac{\Gamma, \phi^R}{\Gamma, (\neg\phi)^L}\ \text{LeftNot} \qquad \dfrac{\Gamma, \phi^L}{\Gamma, (\neg\phi)^R}\ \text{RightNot}$$

(a) Deduction rules of propositional Orthologic.

$$\dfrac{\Gamma, \phi[x := \gamma]^L}{\Gamma, (\bigwedge x.\phi)^L}\ \text{LeftForall} \qquad \dfrac{\Gamma, \phi[x := x']^R}{\Gamma, (\bigwedge x.\phi)^R}\ \begin{array}{c}\text{RightForall}\\ \scriptstyle (x' \text{ not free in } \Gamma)\end{array}$$

$$\dfrac{\Gamma, \phi[x := x']^L}{\Gamma, (\bigvee x.\phi)^L}\ \begin{array}{c}\text{LeftExists}\\ \scriptstyle (x' \text{ not free in } \Gamma)\end{array} \qquad \dfrac{\Gamma, \phi[x := \gamma]^R}{\Gamma, (\bigvee x.\phi)^R}\ \text{RightExists}$$

(b) Deduction rules of Quantified Orthologic.

Fig. 1: Deduction rules of Orthologic. Each holds for arbitrary Γ, Δ, ϕ, ψ. Sets Γ and Δ are either empty or contain a single annotated formula.

2.1 Complete Ortholattices

To model quantified Orthologic, we restrict ortholattices to complete ones.

Definition 2 (Complete Ortholattice). An ortholattice $\mathcal{O} = (O, \sqsubseteq, \sqcup, \sqcap, -)$ is *complete* if and only if for any possibly infinite set of elements $X \subseteq O$, there exist two elements noted $\bigsqcup X$ and $\bigsqcap X$ which are respectively the *lowest upper bound* and *greatest lower bound* of elements of X, with respect to $\sqsubseteq$:

$$\forall x \in X.\ \ x \sqsubseteq \bigsqcup X \text{ and } \bigsqcap X \sqsubseteq x\ ,$$

and, for all $y \in O$:

$$(\forall x \in X. x \sqsubseteq y) \implies (\bigsqcup X \sqsubseteq y)$$

$$(\forall x \in X. y \sqsubseteq x) \implies (y \sqsubseteq \bigsqcap X)$$

This definition coincides with the usual definition in complete lattices. Note that, in particular, all finite ortholattices are complete with bounds computed by iterating the binary operators $\sqcup$ and $\sqcap$.

Definition 3. $\mathcal{T}_{QOL}$ denotes the set of quantified orthologic formulas, i.e. $\mathcal{T}_{OL} \subset \mathcal{T}_{QOL}$ and for any $x \in \mathit{Var}$ and $\phi \in \mathcal{T}_{QOL}$,

$$\bigwedge x.\phi \in \mathcal{T}_{QOL} \quad \text{and} \quad \bigvee x.\phi \in \mathcal{T}_{QOL}.$$

Note that $\mathcal{T}_{QOL} = \mathcal{T}_{QBF}$, the set of quantified Boolean formulas. For two formulas $\phi, \psi \in \mathcal{T}_{QOL}$ and a variable x, let $\phi[x := \psi]$ denotes the usual capture-avoiding substitution of x by ψ inside ϕ.

We assume a representation of quantified formulas where alpha-equivalent terms are equal, so that capture-avoiding substitution is well-defined. It is easy to check that any construction in this paper (and in particular, provability) is consistent across alpha-equivalent formulas.

Definition 4 (Models and Interpretation). A *model* for QOL is a complete ortholattice $\mathcal{O} = (O, \sqsubseteq, \sqcup, \sqcap, -)$ and an assignment $\sigma : \mathit{Var} \to O$. The interpretation of a formula ϕ with respect to an assignment σ is defined recursively as usual:

$$\begin{aligned}
[\![x]\!]_\sigma &:= \sigma(x) \\
[\![\phi \wedge \psi]\!]_\sigma &:= [\![\phi]\!]_\sigma \sqcap [\![\psi]\!]_\sigma \\
[\![\phi \vee \psi]\!]_\sigma &:= [\![\phi]\!]_\sigma \sqcup [\![\psi]\!]_\sigma \\
[\![\neg\phi]\!]_\sigma &:= -[\![\phi]\!]_\sigma \\
[\![\bigvee x.\phi]\!]_\sigma &:= \bigsqcup\{[\![\phi]\!]_{\sigma[x:=e]} \mid e \in O\} \\
[\![\bigwedge x.\phi]\!]_\sigma &:= \bigsqcap\{[\![\phi]\!]_{\sigma[x:=e]} \mid e \in O\}
\end{aligned}$$

where $\sigma[x := e]$ denotes the assignment σ with its value at x changed to e and all other values unchanged.

The interpretation of a sequent is defined in the following way, as in [16]:

$$\begin{aligned}
[\![\phi^L, \psi^R]\!]_\sigma &:= [\![\phi]\!]_\sigma \sqsubseteq [\![\psi]\!]_\sigma \\
[\![\phi^L, \psi^L]\!]_\sigma &:= [\![\phi]\!]_\sigma \sqsubseteq -[\![\psi]\!]_\sigma \\
[\![\phi^R, \psi^R]\!]_\sigma &:= -[\![\phi]\!]_\sigma \sqsubseteq [\![\psi]\!]_\sigma \\
[\![\phi^L]\!]_\sigma &:= [\![\phi]\!]_\sigma \sqsubseteq 0_\mathcal{O} \\
[\![\phi^R]\!]_\sigma &:= 1_\mathcal{O} \sqsubseteq [\![\phi]\!]_\sigma \\
[\![\emptyset]\!]_\sigma &:= 1_\mathcal{O} \sqsubseteq 0_\mathcal{O}
\end{aligned}$$

Definition 5 (Entailment). If the sequent Γ, Δ is provable, we write $\vdash \Gamma, \Delta$. If for every complete ortholattice $\mathcal{O}$ and assignment $\sigma : \mathit{Var} \to O$, $[\![\Gamma, \Delta]\!]_\sigma$ is true, we write $\vDash \Gamma, \Delta$.

By slight abuse of notation, we sometimes write, e.g., $\phi \vdash \psi$ in place of $\vdash \phi^L, \psi^R$ to help readability.

Definition 6. Given formulas ϕ and ψ, let $\phi \dashv\vdash \psi$ denote the fact that both $\phi \vdash \psi$ and $\psi \vdash \phi$ are provable.

We show soundness and completeness of QOL with respect to the class of all complete ortholattices. Soundness is easy and direct, completeness less so.

2.2 Soundness

Lemma 1 (Soundness). *For every sequent S, if $\vdash S$ then $\vDash S$.*

Proof. We simply verify that every deduction rule of Figure 1 preserves truth of interpretation in any model. We show the case of LeftAnd as an example, as well as LeftForall and LeftExists. Other cases are easy or analogous.

Fix an arbitrary ortholattice $\mathcal{O} = (O, \sqsubseteq, \sqcup, \sqcap, -)$.

LeftAnd: For any assignment $\sigma : \mathit{Var} \to O$, the interpretation of the conclusion of a LeftAnd rule is

$$\llbracket \Gamma, (\phi \wedge \psi)^L \rrbracket_\sigma$$

Γ can be empty, a left formula or a right formula. If it is empty then

$$\llbracket \Gamma, (\phi \wedge \psi)^L \rrbracket_\sigma \iff \llbracket 0^R, (\phi \wedge \psi)^L \rrbracket_\sigma$$

If $\Gamma = \gamma^L$, then we have

$$\llbracket \Gamma, (\phi \wedge \psi)^L \rrbracket_\sigma \iff \llbracket (\neg\gamma)^R, (\phi \wedge \psi)^L \rrbracket_\sigma .$$

So without loss of generality we can assume $\Gamma = \gamma^R$ to be a right formula. Now,

$$\begin{aligned} \llbracket \gamma^R, (\phi \wedge \psi)^L \rrbracket_\sigma &\iff \\ \llbracket \phi \wedge \psi \rrbracket_\sigma \sqsubseteq \llbracket \gamma \rrbracket_\sigma &\iff \\ \llbracket \phi \rrbracket_\sigma \sqcap \llbracket \psi \rrbracket_\sigma \sqsubseteq \llbracket \gamma \rrbracket_\sigma & \end{aligned}$$

But using the premise of the LeftAnd rule and the induction hypothesis, we know $\llbracket \gamma^R, \phi^L \rrbracket_\sigma$ holds true. Hence,

$$\begin{aligned} \llbracket \gamma^R, \phi^L \rrbracket_\sigma &\iff \\ \llbracket \phi \rrbracket_\sigma \sqsubseteq \llbracket \gamma \rrbracket_\sigma &\implies \\ \llbracket \phi \rrbracket_\sigma \sqcap \llbracket \psi \rrbracket_\sigma \sqsubseteq \llbracket \gamma \rrbracket_\sigma & \end{aligned}$$

where the implication holds by the laws of ortholattices (Table 2).

LeftForall: For any assignment $\sigma : \mathit{Var} \to O$,

$$\left\llbracket \Gamma, (\bigwedge x.\phi)^L \right\rrbracket_\sigma$$

where we can again assume $\Gamma = \gamma^R$ without loss of generality. Then

$$\begin{aligned} \left\llbracket \gamma^R, (\bigwedge x.\phi)^L \right\rrbracket_\sigma &= \\ \left\llbracket \bigwedge x.\phi \right\rrbracket_\sigma \sqsubseteq \llbracket \gamma \rrbracket_\sigma &= \\ \bigsqcap \{ \llbracket \phi \rrbracket_{\sigma[x:=e]} \mid e \in O \} \sqsubseteq \llbracket \gamma \rrbracket_\sigma & \end{aligned}$$

Again, by hypothesis, there exists a formula ψ such that $\llbracket \gamma^R, \phi[x := \psi]^L \rrbracket_\sigma$ holds true. Finally,

$$\begin{aligned} \llbracket \gamma^R, \phi[x := \psi]^L \rrbracket_\sigma &\iff \\ \llbracket \phi[x := \psi] \rrbracket_\sigma \sqsubseteq \llbracket \gamma \rrbracket_\sigma &\iff \\ \llbracket \phi \rrbracket_{\sigma[x := \llbracket \psi \rrbracket_\sigma]} \sqsubseteq \llbracket \gamma \rrbracket_\sigma &\implies \\ \bigsqcap \{ \llbracket \phi \rrbracket_{\sigma[x:=e]} \mid e \in O \} \sqsubseteq \llbracket \gamma \rrbracket_\sigma & \end{aligned}$$

where the last implication holds by definition of $\bigsqcap$.
LeftExists: For any assignment $\sigma : \mathit{Var} \to O$,

$$\left\llbracket \Gamma, (\bigvee x.\phi)^L \right\rrbracket_\sigma$$

where we assume one last time without loss of generality that $\Gamma = \gamma^R$. Then

$$\begin{aligned} \left\llbracket \gamma^R, (\bigvee x.\phi)^L \right\rrbracket_\sigma &\iff \\ \left\llbracket \bigvee x.\phi \right\rrbracket_\sigma \sqsubseteq \llbracket \gamma \rrbracket_\sigma &\iff \\ \bigsqcup \{ \llbracket \phi \rrbracket_{\sigma[x:=e]} \mid e \in O \} \sqsubseteq \llbracket \gamma \rrbracket_\sigma & \end{aligned}$$

By hypothesis, $\llbracket \gamma^R, \phi^L \rrbracket_\tau$ holds for any assignment τ, and in particular for any τ of the form $\sigma[x := e]$. Since x does not appear in γ, $\llbracket \gamma \rrbracket_{\sigma[x:=e]} = \llbracket \gamma \rrbracket_\sigma$. Hence, for any $e \in O$, each of the following line holds true:

$$\begin{aligned} \llbracket \gamma^R, \phi^L \rrbracket_{\sigma[x:=e]} &\iff \\ \llbracket \phi \rrbracket_{\sigma[x:=e]} \sqsubseteq \llbracket \gamma \rrbracket_{\sigma[x:=e]} &\iff \\ \llbracket \phi \rrbracket_{\sigma[x:=e]} \sqsubseteq \llbracket \gamma \rrbracket_\sigma & \end{aligned}$$

By the least upper bound property of $\bigsqcup$, we obtain as desired the truth of:

$$\bigsqcup \{ \llbracket \phi \rrbracket_{\sigma[x:=e]} \mid e \in O \} \sqsubseteq \llbracket \gamma \rrbracket_\sigma$$

2.3 Completeness

In classical propositional logic, we can show completeness with respect to the $\{0, 1\}$ Boolean algebra, which is straightforward. In orthologic, however, we do not have completeness with respect to a simple finite structure; we will need to build an infinite complete ortholattices. The construction is distinct but not entirely unlike that of models for predicate orthologic [1,37]. In particular, MacNeille completion [28] is used to transform the initial incomplete model into a complete one.

Lemma 2 (Completeness). *For any sequent S, $\vDash S$ implies $\vdash S$.*

Proof. We prove the contraposition: if the sequent S is not provable, then there exists a complete ortholattice $\mathcal{O} = (O, \sqsubseteq, \sqcup, \sqcap, -)$ and an assignment $\sigma : \mathit{Var} \to O$ such that $[\![S]\!]_\sigma$ does not hold. We construct O from the set of syntactic terms of complete ortholattices themselves, similarly to a free algebra (but with quantifiers). Formally, let O be $\mathcal{T}_{QOL}/\dashv\vdash$, i.e. the quotient set of $\mathcal{T}_{QOL}$ by the relation $\dashv\vdash$.

It is immediate that the function symbols $\land, \lor, \neg$ and relation symbol $\vdash$ of $\mathcal{T}_{QOL}$ are consistent over the equivalence classes of O, allowing us to extend them to O:

$$\begin{aligned}
[\phi]_{\dashv\vdash} \sqcap [\psi]_{\dashv\vdash} &:= [\phi \land \psi]_{\dashv\vdash} \\
[\phi]_{\dashv\vdash} \sqcup [\psi]_{\dashv\vdash} &:= [\phi \lor \psi]_{\dashv\vdash} \\
-[\phi]_{\dashv\vdash} &:= [\neg\phi]_{\dashv\vdash} \\
[\phi]_{\dashv\vdash} \sqsubseteq [\psi]_{\dashv\vdash} &:= (\phi \vdash \psi) \text{ is provable}
\end{aligned}$$

It is also immediate that $\mathcal{O} = (O, \sqsubseteq, \sqcup, \sqcap, -)$ satisfies all the laws of ortholattices of Table 1. However, to interpret a quantified formula into $\mathcal{O}$, we would need $\mathcal{O}$ to be complete. It might not be complete, but it is "complete enough" to define all upper bounds of interest, as the following lemma shows.

Lemma 3. *For any $\sigma : \mathit{Var} \to O$, let $\sigma' : \mathit{Var} \to \mathcal{T}_{QOL}$ be such that for any x $[\sigma'(x)]_{\dashv\vdash} = \sigma(x)$. Let $\phi[\sigma']$ denote the simultaneous capture-avoiding substitution of variables in the formula ϕ with the assignments in σ'.*

Then, for any $\phi \in \mathcal{T}_{QOL}$, $[\![\phi]\!]_\sigma$ exists and $[\![\phi]\!]_\sigma = [\phi[\sigma']]_{\dashv\vdash}$.

Proof. First note that $[\phi[\sigma']]_{\dashv\vdash}$ is well-defined: it does not depend on the specific choice of assignment we make for σ'. Then, the proof works by structural induction on ϕ. If it is a variable x,

$$[\![x]\!]_\sigma = \sigma(x) = [\sigma'(x)]_{\dashv\vdash} = [x[\sigma']]_{\dashv\vdash}$$

by definition. Then, if $\phi = \phi_1 \land \phi_2$,

$$[\![\phi_1 \land \phi_2]\!]_\sigma = [\![\phi_1]\!]_\sigma \sqcap [\![\phi_2]\!]_\sigma = [\phi_1[\sigma']]_{\dashv\vdash} \sqcap [\phi_2[\sigma']]_{\dashv\vdash} = [\phi_1[\sigma'] \land \phi_2[\sigma']]_{\dashv\vdash}$$

where the first equality is the definition of $[\![\cdot]\!]$, the second equality the induction hypothesis and the third equality is the definition of $\sqcap$ in $\mathcal{O}$. $\lor$ and $\neg$ are similar. Consider now the interpretation of a formula $[\![\bigvee x.\phi]\!]_\sigma$. Since alpha-equivalence holds in our proof system and in the definition of the least upper bound, we assume to ease notation that x is fresh with respect to σ, i.e., that we don't need to signal explicitly capture-avoiding substitution. By definition of $[\![\cdot]\!]_\sigma$, we should have:

$$\left[\!\!\left[\bigvee x.\phi\right]\!\!\right]_\sigma = \bigsqcup\{[\![\phi]\!]_{\sigma[x:=e]} \mid e \in O\}$$

Does the right-hand side always exist in $\mathcal{O}$? We claim that it does, and that it is equal to $[(\bigvee x.\phi)[\sigma']]_{\dashv\vdash}$. Mainly, we need to show that it satisfies the two properties of the least upper bound. First, the *upper bound* property:

$$\forall a \in \{[\![\phi]\!]_{\sigma[x:=e]} \mid e \in O\}, \quad a \sqsubseteq \left[(\bigvee x.\phi)[\sigma']\right]_{\dashv\vdash}$$

Which is equivalent to $\forall e \in O$,

$$\begin{aligned}
[\![\phi]\!]_{\sigma[x:=e]} \sqsubseteq \left[(\bigvee x.\phi)[\sigma']\right]_{\dashv\vdash} &\qquad \iff &&(1)\\
\left[\phi[\sigma'_{[x:=e]}]\right]_{\dashv\vdash} \sqsubseteq \left[(\bigvee x.\phi)[\sigma']\right]_{\dashv\vdash} &\qquad \iff &&(2)\\
\phi[\sigma'_{[x:=e]}] \vdash (\bigvee x.\phi)[\sigma'] \text{ is provable} &\qquad \iff &&(3)\\
\phi[\sigma'_{[x:=e]}] \vdash \bigvee x.\phi[\sigma'_{[x:=x]}] \text{ is provable} & &&(4)
\end{aligned}$$

where (1) is the desired least upper bound property, (2) is equivalent by induction hypothesis and definition of σ', (3) by definition of $\sqsubseteq$ in $\mathcal{O}$ and (4) by definition of substitution. The last statement is indeed provable:

$$\cfrac{\cfrac{}{\phi[\sigma_{[x:=e]}]^L, \phi[\sigma_{[x:=e]}]^R}\ \text{Hyp}}{\phi[\sigma_{[x:=e]}]^L, (\bigvee x.\phi[\sigma'_{[x:=x]}])^R}\ \text{RightExists}$$

Secondly, we need to show the *least* upper bound property:

$$\forall a \in O.(\forall e \in O.\, [\![\phi]\!]_{\sigma[x:=e]} \sqsubseteq a) \implies (\left[(\bigvee x.\phi)[\sigma']\right]_{\dashv\vdash} \sqsubseteq a)$$

which is equivalent to

$$\forall \psi \in \mathcal{T}_{QOL}.(\forall e \in O.\, [\![\phi]\!]_{\sigma[x:=e]} \sqsubseteq [\psi]_{\dashv\vdash}) \implies (\left[(\bigvee x.\phi)[\sigma']\right]_{\dashv\vdash} \sqsubseteq [\psi]_{\dashv\vdash})$$

Fix an arbitrary ψ and assume $\forall e \in O.\, [\![\phi]\!]_{\sigma[x:=e]} \sqsubseteq [\psi]_{\dashv\vdash}$. Consider a variable x_2 which does not appear in ψ. Then, we have in particular,

$$[\![\phi]\!]_{\sigma[x:=x_2]} \sqsubseteq [\psi]_{\dashv\vdash}.$$

Then,

$$\begin{aligned}
[\![\phi]\!]_{\sigma[x:=x_2]} \sqsubseteq [\psi]_{\dashv\vdash} &\qquad \iff\\
\left[\phi[\sigma'_{[x:=x_2]}]\right]_{\dashv\vdash} \sqsubseteq [\psi]_{\dashv\vdash} &\qquad \iff\\
\phi[\sigma'_{[x:=x_2]}] \vdash \psi \text{ is provable} &\qquad \iff
\end{aligned}$$

Then using a proof of the last line, we can construct:

$$\cfrac{\phi[\sigma'_{[x:=x_2]}]^L, \psi^R}{(\bigvee x.\phi[\sigma'_{[x:=x]}])^L, \psi^R}\ \text{LeftExists}$$

We finally obtain our second property as desired:

$$\left[(\bigvee x.\phi)[\sigma']\right]_{\dashv\vdash} \sqsubseteq [\psi]_{\dashv\vdash}$$

To conclude the proof of Lemma 3, the case with $\bigwedge$ instead of $\bigvee$ is symmetrical.

Hence, our interpretation in $\mathcal{O}$ is guaranteed to be well-defined. However, $\mathcal{O}$ is not guaranteed to be complete for arbitrary sets of elements, which our definition of a model requires. To obtain a complete ortholattice, we will apply MacNeille completion to $\mathcal{O}$.

Definition 7 (MacNeille Completion, [28]). Given a lattice L, there exists a smallest complete lattice L' containing L as a sublattice with an embedding $i : L \to L'$ preserving the least upper bounds and greatest lower bounds of arbitrary (possibly infinite) subsets of L. This is the MacNeille completion of L.

Hence, there exists a complete lattice $\mathcal{O}'$ containing $\mathcal{O}$ as a sublattice and preserving the existing least upper bounds and greatest lower bound. But we also need $\mathcal{O}'$ to be an ortholattice, containing $\mathcal{O}$ as a subortholattice. Fortunately, this is true thanks to a theorem of Bruns.

Lemma 4 (Theorem 4.2 of [5]). *For every ortholattice $\mathcal{O}$, its MacNeille completion $\mathcal{O}'$ admits an orthocomplementation which extends the orthocomplementation of $\mathcal{O}$.*

Corollary 1. *There exists an injective ortholattice homomorphism $i : \mathcal{O} \to \mathcal{O}'$ such that*

$$\forall a, b \in O. a \leq_{\mathcal{O}} b \iff i(a) \leq_{\mathcal{O}'} i(b)$$

and for any $X \subset O$ such that $\bigsqcup X$ (resp. $\bigsqcap X$) exists, and

$$i(\bigsqcup X) = \bigsqcup(\{i(x) \mid x \in X\})$$
$$i(\bigsqcap X) = \bigsqcap(\{i(x) \mid x \in X\}).$$

We can now finish our completeness proof. Define $\sigma : \mathit{Var} \to O$ by $\sigma(x) = [x]_{\dashv\vdash}$. Then by Lemma 3, $[\![\phi]\!]_\sigma = [\phi]_{\dashv\vdash}$. Let γ, δ be the two formulas such that $[\![S]\!]_\sigma = ([\![\gamma]\!]_\sigma \sqsubseteq [\![\delta]\!]_\sigma)$, according to Definition 4. Remember that the sequent S is not provable by assumption, i.e. $[\![\gamma]\!]_\sigma \not\sqsubseteq [\![\delta]\!]_\sigma$, and hence:

$$[\gamma]_{\dashv\vdash} \not\sqsubseteq_{\mathcal{O}} [\delta]_{\dashv\vdash}$$

from which we deduce

$$i([\gamma]_{\dashv\vdash}) \not\sqsubseteq_{\mathcal{O}'} i([\delta]_{\dashv\vdash})$$

in the ortholattice $\mathcal{O}'$. We now define $\tau : \mathit{Var} \to O'$ such that $\tau(x) = i(\sigma(x))$, implying (by induction and Corollary 1) that for any ϕ,

$$i([\phi]_{\dashv\vdash}) = [\![\phi]\!]_\tau$$

and therefore, in $\mathcal{O}'$:

$$[\![\gamma]\!]_\tau \not\sqsubseteq [\![\delta]\!]_\tau$$

We have hence built a model with the complete ortholattice $\mathcal{O}'$ and the assignment τ in which $[\![\gamma]\!]_\tau \not\sqsubseteq [\![\delta]\!]_\tau$, so $[\![S]\!]_\tau$ does not hold, as desired.

Theorem 1. *QOL is sound and complete for complete ortholattices, i.e. for any sequent S:*

$$\vdash S \iff \models S$$

3 No Quantifier Elimination for Orthologic

Definition 8 (Quantifier Elimination). A quantified propositional logic admits *quantifier elimination* if for any term Q there exists a quantifier-free term E such that $Q \dashv\vdash E$.

Example 1. QBF, the theory of quantified classical propositional logic, admits quantifier elimination that replaces the quantified proposition $\exists x.F$ with the proposition $F[x := 0] \vee F[x := 1]$. This quantifier elimination approach is sound over Boolean algebras in general, thanks to the distributivity law.

Example 2. The theory of quantified intuitionistic propositional logic does not admit quantifier elimination. Whereas provability in quantifier-free intuitionistic propositional logic corresponds closely to inhabitation in simply typed lambda calculus and is PSPACE-complete [44], the quantified theory corresponds to System F, and is undecidable [11].

Note that quantifier elimination provides a solution to the interpolant problem for QBF (and QOL, if it were to admit quantifier elimination). Indeed, consider a provable sequent $A_{(\overrightarrow{x},\overrightarrow{y})} \vdash B_{(\overrightarrow{y},\overrightarrow{z})}$, and $\overrightarrow{x}, \overrightarrow{y}, \overrightarrow{z}$ the free variables in A and B. We ask for an interpolant such that

$$A_{(\overrightarrow{x},\overrightarrow{y})} \vdash I_{\overrightarrow{y}} \text{ and } I_{\overrightarrow{y}} \vdash B_{(\overrightarrow{y},\overrightarrow{z})}$$

By quantifier elimination, there exists a quantifier-free formula $I_{\overrightarrow{y}}$ equivalent to $\bigwedge z.B_{(\overrightarrow{y},\overrightarrow{z})}$. This $I_{\overrightarrow{y}}$ satisfies the interpolant condition.

$$\dfrac{A_{(\overrightarrow{x},\overrightarrow{y})} \vdash B_{(\overrightarrow{y},\overrightarrow{z})}}{A_{(\overrightarrow{x},\overrightarrow{y})} \vdash (\bigwedge z.B_{(\overrightarrow{y},\overrightarrow{z})})}\ \text{RightForall} \qquad \dfrac{\dfrac{}{B_{(\overrightarrow{y},\overrightarrow{z})} \vdash B_{(\overrightarrow{y},\overrightarrow{z})}}\ \text{Hyp}}{(\bigwedge z.B_{(\overrightarrow{y},\overrightarrow{z})}) \vdash B_{(\overrightarrow{y},\overrightarrow{z})}}\ \text{LeftForall}$$

However, the next theorem will show that QOL does not admit quantifier elimination in general, even though it still admits interpolation.

Theorem 1. *QOL does not admit quantifier elimination. In particular, there exists no quantifier free formula E such that*

$$E \dashv\vdash \bigvee x.\big(\neg x \wedge (y \vee x)\big)$$

Proof. For the sake of contradiction, suppose such an E exists. Let $y, w_1, ..., w_n$ be the free variables appearing in E. Since $\bigvee x.\neg x \wedge (y \vee x)$ is constant with respect to $w_1, ..., w_n$, E must be as well, and hence we can assume E only uses y as a variable. Moreover, the laws of OL in Table 1 imply that any quantifier free formula whose only variable is y is equivalent to one of $0, 1, y$ or $\neg y$. This can easily be shown by induction on the structure of the formula:

$$\begin{aligned}
0 \wedge 0 &= 0 & 0 \wedge 1 &= 1 \\
0 \wedge y &= 0 & 0 \wedge \neg y &= 0 \\
1 \wedge 1 &= 1 & 1 \wedge y &= y \\
1 \wedge \neg y &= \neg y & y \wedge y &= y \\
y \wedge \neg y &= 0 & \neg y \wedge \neg y &= \neg y \\
\neg 0 &= 1 & \neg 1 &= 0 \\
\neg\neg y &= y
\end{aligned}$$

and similarly for disjunction.
Now, consider the ortholattices M_2 and M_4 in Figure 2:

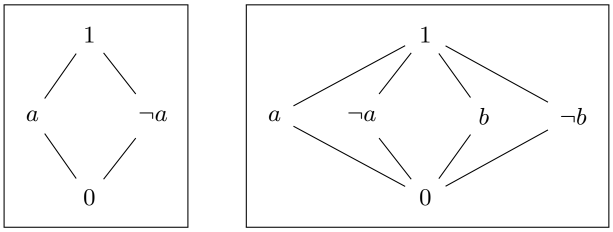

Fig. 2: The ortholattices M_2 and M_4. M_2 is distributive, but M_4 is not.

We use soundness of orthologic over ortholattices (Lemma 1) so that the formula E (if it exists) needs to be equal to $\bigvee x. \neg x \wedge (y \vee x)$ in all models. Since the model is finite, it is straightforward to compute in the ortholattice M_2 with the assignment $y := a$ that:

$$\left[\!\!\left[\bigvee x. \neg x \wedge (y \vee x) \right]\!\!\right]_{M_2, y := a} = a$$

And hence the only compatible formula for E is the atom y.

However, in M_4:

$$\left[\!\!\left[\bigvee x. \neg x \wedge (y \vee x) \right]\!\!\right]_{M_4, y := a} = 1$$

Hence, any expression for E among $0, 1, y, \neg y$ will fail to satisfy at least one of the two examples, and we conclude that there is no quantifier free formula E that is equivalent to $\bigvee x. \neg x \wedge (y \vee x)$. □

On one hand, this result shows that we can use quantifiers to define new operators, such as

$$\lceil y \rceil \equiv \bigvee x. \big(\neg x \wedge (y \vee x)\big)$$

while Theorem 1 shows that $\lceil y \rceil$ is not expressible without quantifiers. On the other hand, this result implies that we cannot use quantifier elimination to compute quantifier-free interpolants; such interpolants require a different approach.

4 Failure of a Refutation-Based Interpolation

We now consider a notion of interpolation based on orthologic with axioms. Using axioms makes the assumptions stronger and is a closer approximation of classical propositional logic.

Definition 9 (Refutation-Based Interpolation). Given an inconsistent pair of sequents A and B, i.e. there exists a proof of contradiction (the empty sequent) assuming them, a sequent I is said to be a refutation-based interpolant of (A, B) if

– I can be deduced from A alone,
– I and B are inconsistent, and
– $\mathsf{FV}(I) \subseteq \mathsf{FV}(A) \cap \mathsf{FV}(B)$.

We show, by counterexample, that such an interpolant does not exist in general in orthologic.

Theorem 2. *Given any inconsistent pair (A, B) of sequents, a refutation-based interpolant for it does not necessarily exist in orthologic. In particular, a refutation-based interpolant does not exist for the choice $A = (z \vee \neg y) \wedge (\neg z \vee \neg y)^R$ and $B = (x \wedge y) \vee (\neg x \wedge y)^R$.*

Proof. For the counterexample, let A be the sequent

$$(z \vee \neg y) \wedge (\neg z \vee \neg y)^R$$

and let B be

$$(x \wedge y) \vee (\neg x \wedge y)^R \ .$$

We show the proof of inconsistency of A and B in orthologic in Figure 3. For readability and space constraints, the proof is split into four parts, with Figure 3a showing a proof of y^R from B, Figure 3b and Figure 3c showing proofs of z^R and $\neg z^R$ respectively from y^R and A, and Figure 3d finally deriving the empty sequent from z^R and $\neg z^R$.

Given the inconsistent pair of A and B, however, we find that the only common variable between A and B is y. Sequents built only from the variable y are equivalent to one of $0^R, 1^R, y^R, (\neg y)^R$, none of which is a consequence of A while also being inconsistent with B.

Hence, the inconsistent sequent pair A, B as chosen does not admit a refutation-based interpolant according to Definition 9.

5 Interpolation for Orthologic Formulas

An arguably more natural definition of interpolation for a lattice-based logic (such as orthologic) using the $\leq$ relation is the following:

$$\dfrac{(x\wedge y)\vee(\neg x\wedge y)^R \qquad \dfrac{\dfrac{\dfrac{}{y^L, y^R}\,\text{Hyp}}{x\wedge y^L, y^R}\,\text{LeftAnd} \qquad \dfrac{\dfrac{}{y^L, y^R}\,\text{Hyp}}{\neg x\wedge y^L, y^R}\,\text{LeftAnd}}{(x\wedge y)\vee(\neg x\wedge y)^L, y^R}\,\text{LeftOr}}{y^R}\,\text{Cut}$$

(a) Proof of y^R from B.

$$\dfrac{(\neg z\vee\neg y)\wedge(z\vee\neg y)^R \qquad \dfrac{\dfrac{\dfrac{}{z^L, z^R}\,\text{Hyp} \qquad \dfrac{\dfrac{y^R}{\neg y^L}\,\text{LeftNot}}{\neg y^L, z^R}\,\text{Weaken}}{z\vee\neg y^L, z^R}\,\text{LeftOr}}{(\neg z\vee\neg y)\wedge(z\vee\neg y)^L, z^R}\,\text{LeftAnd}}{z^R}\,\text{Cut}$$

(b) Proof of z^R from y^R and A.

$$\dfrac{(\neg z\vee\neg y)\wedge(z\vee\neg y)^R \qquad \dfrac{\dfrac{\dfrac{}{\neg z^L, \neg z^R}\,\text{Hyp} \qquad \dfrac{\dfrac{y^R}{\neg y^L}\,\text{LeftNot}}{\neg y^L, \neg z^R}\,\text{Weaken}}{\neg z\vee\neg y^L, \neg z^R}\,\text{LeftOr}}{(\neg z\vee\neg y)\wedge(z\vee\neg y)^L, \neg z^R}\,\text{LeftAnd}}{\neg z^R}\,\text{Cut}$$

(c) Proof of $\neg z^R$ from y^R and A.

$$\dfrac{\neg z^R \qquad \dfrac{z^R}{\neg z^L}\,\text{LeftNot}}{\emptyset}\,\text{Cut}$$

(d) Proof of $\emptyset$ from z^R and $\neg z^R$.

Fig. 3: Proof of inconsistency of $A = (\neg z\vee\neg y)\wedge(z\vee\neg y)$ and $B = (x\wedge y)\vee(\neg x\wedge y)$.

Definition 1 (Implicational interpolation). *Given two propositional quantifier-free formulas A and B such that $A \le B$, an interpolant is a formula I such that $\mathsf{FV}(I) \subseteq \mathsf{FV}(A) \cap \mathsf{FV}(B)$ and $A \le I \le B$.*

For classical logic, this definition is equivalent to the one of Section 4, since $A \le_{CL} B$ if and only if the empty sequent is provable from $(A, \neg B)$. These definitions are however not equivalent in intuitionistic logic and orthologic.

We now prove that the theory of ortholattices admits this form of interpolation by showing a procedure constructing the interpolant inductively from a proof of the sequent A^L, B^R. For induction, we prove a slightly more general statement:

Theorem 3 (Interpolant for orthologic sequents). *There exists an algorithm (interpolate in Figure 4) , which, given a proof of a sequent Γ, Δ, computes a formula I called an* interpolant *for the ordered pair (Γ, Δ) such that*

$\mathsf{FV}(I) \subseteq \mathsf{FV}(\Gamma) \cap \mathsf{FV}(\Delta)$ and the sequents Γ, I^R and I^L, Δ are provable. The algorithm has runtime linear in the size of the given proof of Γ, Δ.

Note that interpolants for (Γ, Δ) and for (Δ, Γ) are distinct. In fact, in orthologic, they are negations of each other. Note also that if the sequent Γ, Δ is provable, then by [16] there is a proof of it of at most quadratic size. Hence, the interpolation algorithm runs in the worst-case in time quadratic in sizes of Γ and Δ.

We have made an executable Scala implementation of the interpolation algorithm alongside orthologic proof search (as described in [16]) open-source on GitHub [1].

Proof. We show correctness of the algorithm in Figure 4 with inputs Γ, Δ, P where P is a proof of the sequent $S = \Gamma, \Delta$. By the cut-elimination theorem of orthologic [16, 41], we assume that P is cut-free. We show that the result of interpolate(Γ, Δ, P) is an interpolant for (Γ, Δ).

We first deal with the particular case where either Γ or Δ is empty or when $\Gamma = \Delta$, as it will simplify the rest of the proof to assume that they are both non-empty and distinct.

- Suppose $(\Gamma, \Delta) = (\Pi, \emptyset)$. Then 0 is an interpolant, as both $\Pi, 0^R$ and $0^L, \emptyset$ are provable.
- Suppose $(\Gamma, \Delta) = (\emptyset, \Pi)$. Then 1 is an interpolant, as both $\emptyset, 1^R$ and $1^L, \Pi$ are provable
- Suppose $(\Gamma, \Delta) = (\Pi, \Pi)$. Then any formula ψ (in particular 0 and 1) is an interpolant as both Π, ψ^R and ψ^L, Π are provable by weakening.

In all other cases, the algorithm works recursively on the proof tree of P, starting from the concluding (root) step. At every step, the algorithm reduces the construction of the interpolant of S to those of its premises. By induction, assume that for a given proof P, the algorithm is correct for all proofs of smaller size (and in particular for the premises of P) and consider every proof step from Figure 1 with which P can be concluded:

- Hyp: suppose the concluding step is

$$\frac{}{\phi^L, \phi^R}\ \text{Hyp}\ ,$$

We must have $(\Gamma, \Delta) = (\phi^L, \phi^R)$, or $(\Gamma, \Delta) = (\phi^R, \phi^L)$. Assuming the former, consider the interpolant $I = \phi$. We then trivially have proofs of $(\Gamma, I^R) = (\phi^R, \phi^L)$ and $(I^L, \Delta^R) = (\phi^L, \phi^R)$:

$$\frac{}{\phi^L, \phi^R}\ \text{Hyp}\ , \qquad \text{and} \qquad \frac{}{\phi^L, \phi^R}\ \text{Hyp}\ ,$$

The latter case is symmetrical, with $I = \neg\phi$.

[1] https://github.com/sankalpgambhir/ol-interpolation

```
def interpolate(
    $\Gamma$: Option[AnnotatedFormula],
    $\Delta$: Option[AnnotatedFormula], // the input sequent $\Gamma, \Delta$
    $p$: ProofStep // proof of validity of the input sequent
): Formula =
($\Gamma$, $\Delta$) match
    case (Some($\Pi$), None) ⇒ 0
    case (None, Some($\Pi$)) ⇒ 1
    case (Some($\Pi$), Some($\Pi$)) ⇒ 0 // or 1
    case _ ⇒
        $p$ match
            case Hypothesis($\phi$) ⇒
                $\Gamma$ match
                    case Some($`\phi`^L$) ⇒ $\phi$
                    case Some($`\phi`^R$) ⇒ $\neg\phi$
            case Weaken($\Sigma$, $p'$) ⇒
                $\Gamma$ match
                    case $`\Sigma`$ ⇒ interpolate(None, $\Delta$, $p'$)
                    case _ ⇒ interpolate($\Gamma$, None, $p'$)
            case LeftAnd($\phi$, $\psi$, $p'$) ⇒
                $\Gamma$ match
                    case Some($`\phi` \wedge `\psi`^L$) ⇒ interpolate($\phi^L$, $\Delta$, $p'$)
                    case _ ⇒ interpolate($\Gamma$, $\phi^L$, $p'$)
            case RightAnd($\phi$, $\psi$, $p_1$, $p_2$) ⇒
                $\Gamma$ match
                    case Some($`\phi` \wedge `\psi`^R$) ⇒
                        interpolate($\phi^R$, $\Delta$, $p_1$) $\vee$ interpolate($\psi^R$, $\Delta$, $p_2$)
                    case _ ⇒
                        interpolate($\Delta$, $\phi^R$, $p_1$) $\wedge$ interpolate($\Delta$, $\psi^R$, $p_2$)
            case LeftOr($\phi$, $\psi$, $p_1$, $p_2$) ⇒
                $\Gamma$ match
                    case Some($`\phi` \vee `\psi`^L$) ⇒
                        interpolate($\phi^L$, $\Delta$, $p_1$) $\vee$ interpolate($\psi^L$, $\Delta$, $p_2$)
                    case _ ⇒
                        interpolate($\Delta$, $\phi^L$, $p_1$) $\wedge$ interpolate($\Delta$, $\psi^L$, $p_2$)
            case RightOr($\phi$, $\psi$, $p'$) ⇒
                $\Gamma$ match
                    case Some($`\phi` \vee `\psi`^R$) ⇒ interpolate($\phi^R$, $\Delta$, $p'$)
                    case _ ⇒ interpolate($\Gamma$, $\phi^R$, $p_1$)
            case LeftNot($\phi$, $p'$) ⇒
                $\Gamma$ match
                    case Some($\neg`\phi`^L$) ⇒ interpolate($\phi^R$, $\Delta$, $p'$)
                    case _ ⇒ interpolate($\Gamma$, $\phi^R$, $p'$)
            case RightNot($\phi$, $p'$) ⇒
                $\Gamma$ match
                    case Some($\neg`\phi`^R$) ⇒ interpolate($\phi^L$, $\Delta$, $p'$)
                    case _ ⇒ interpolate($\Gamma$, $\phi^L$, $p'$)
```

Fig. 4: The algorithm interpolate to produce an interpolant for any valid sequent, given a partition as an ordered pair and a proof. $`\phi`$ in a pattern match is Scala syntax to indicate that ϕ is an existing variable to be tested for equality, and not a fresh variable free to be assigned.

- Weaken: suppose the final inference is

$$\frac{\Pi}{\Pi, \Sigma}\ \text{Weaken}\ .$$

  As before, we must have $(\Gamma, \Delta) = (\Pi, \Sigma)$ or $(\Gamma, \Delta) = (\Sigma, \Pi)$. In the former case, consider the interpolant C for $(\Pi, \emptyset)$, the premise (in fact $C = 0$ or $C = 1$). By the hypothesis, the sequents

$$\Pi, C^R \text{ , and } \quad C^L, \emptyset$$

  are provable. Taking $I = C$, and applying Weaken on the second sequent, we obtain proofs of Γ, I^R and I^L, Δ:

$$\Pi, C^R \qquad \text{and} \qquad \frac{C^L}{C^L, \Delta}\ \text{Weaken}$$

  The case $\Gamma = \Delta$ is analogous, with $I = \neg C$.
- LeftAnd: suppose the final inference is

$$\frac{\Pi, \phi^L}{\Pi, \phi \wedge \psi^L}\ \text{LeftAnd}\ .$$

  We must have $(\Gamma, \Delta) = ((\phi \wedge \psi)^L, \Pi)$ or swapped. In the former case, by the induction hypothesis, consider an interpolant C for (ϕ^L, Π), such that the sequents

$$\phi^L, C^R \qquad\qquad C^L, \Pi$$

  are provable. For $I = C$ as interpolant, we have proofs of Γ, I^R and I^L, Δ:

$$\frac{\phi^L, C^R}{\phi \wedge \psi^L, C^R}\ \text{LeftAnd} \qquad \text{and} \qquad C^L, \Pi \quad .$$

  Since $\mathsf{FV}(\phi) \subseteq \mathsf{FV}(\phi \wedge \psi)$, $I = C$ is an interpolant for the conclusion as required. The case where $(\Gamma, \Delta) = (\Pi, (\phi \wedge \psi)^L)$ is analogous.
- RightAnd: suppose the final inference is

$$\frac{\Pi, \phi^R \qquad \Pi, \psi^R}{\Pi, (\phi \wedge \psi)^R}\ \text{RightAnd}\ .$$

  We have $(\Gamma, \Delta) = (\Pi, (\phi \wedge \psi)^R)$, or the other way round. Assume the former. Applying the induction hypothesis twice, we obtain an interpolant for each of the premises, C_ϕ and C_ψ, such that the sequents

$$\begin{array}{ll} \Pi, C_\phi^R \qquad\qquad & C_\phi^L, \phi^R \\ \Pi, C_\psi^R & C_\psi^L, \psi^R \end{array}$$

are valid. Take $I = C_\phi \wedge C_\psi$ as interpolant. Indeed, its free variables are contained in $\mathsf{FV}(\Pi) \cap (\mathsf{FV}(\phi) \cup \mathsf{FV}(\psi)) = \mathsf{FV}(\Gamma) \cap \mathsf{FV}(\phi \wedge \psi)$.
We then need proofs for Γ, I^R and I^L, Δ:

$$\dfrac{\Pi, C_\phi^R \qquad \Pi, C_\psi^R}{\Pi, (C_\phi \wedge C_\psi)^R}\ \text{RightAnd}$$

and

$$\dfrac{\dfrac{C_\phi^L, \phi^R}{C_\phi \wedge C_\psi^L, \phi^R}\ \text{LeftAnd} \qquad \dfrac{C_\psi^L, \psi^R}{C_\phi \wedge C_\psi^L, \psi^R}\ \text{LeftAnd}}{(C_\phi \wedge C_\psi)^L, (\phi \wedge \psi)^R}\ \text{RightAnd.}$$

showing that I is an interpolant for the pair (Γ, Δ).
Now in the other case $(\Gamma, \Delta) = ((\phi \wedge \psi)^R, \Pi)$, the induction hypothesis gives us the following interpolants:

$$\begin{array}{cc} \phi^R, D_\phi^R & D_\phi^L, \Pi \\ \psi^R, D_\psi^R & D_\psi^L, \Pi \end{array}$$

We can then take $I = (D_\phi \vee D_\psi)$ to obtain proofs of Γ, I^R and I^L, Δ:

$$\dfrac{\dfrac{\phi^R, D_\phi^R}{\phi^R, (D_\phi \vee D_\psi)^R}\ \text{RightOr} \qquad \dfrac{\psi^R, D_\psi^R}{\psi^R, (D_\phi \vee D_\psi)^R}\ \text{RightOr}}{(\phi \wedge \psi)^R, (D_\phi \vee D_\psi)^R}\ \text{RightAnd.}$$

and

$$\dfrac{D_\phi^L, \Pi \qquad D_\psi^L, \Pi}{(D_\phi \vee D_\psi)^L, \Pi}\ \text{LeftOr}$$

Note that we can show by induction that $D_\phi \vee D_\psi = \neg(C_\phi \wedge C_\psi)$.

- LeftNot: suppose the final inference is

$$\dfrac{\Pi, \phi^R}{\Pi, \neg\phi^L}\ \text{LeftNot}\ .$$

We have $(\Gamma, \Delta) = (\Pi, (\neg\phi)^L)$, or the other way round. Assume the former. We apply the induction hypothesis as before to obtain an interpolant C for (Π, ϕ^R) such that

$$\Pi, C^R \qquad\qquad C^L, \phi^R$$

are valid. $I = C$ suffices as an interpolant for the concluding sequent, with the proofs of Γ, I^R and I^L, Δ

$$\Pi, C^R \qquad \text{and} \qquad \frac{C^L, \phi^R}{C^L, \neg\phi^L}\text{ LeftNot} \quad .$$

The proofs for the remaining proof rules, LeftOr, RightOr, and RightNot, are analogous to the cases listed above.

Corollary 2 (Interpolation for Ortholattices). *Ortholattices admit interpolation, i.e., for any pair of formulas A, B in an ortholattice with $A \leq B$, there exists a formula I such that $A \leq I$ and $I \leq B$, with $\mathsf{FV}(I) \subseteq \mathsf{FV}(A) \cap \mathsf{FV}(B)$.*

6 Further Related Work

The best known interpolation result is Craig's interpolation theorem for first order logic [10], of which interpolation for classical propositional logic is a special case. Interpolation for predicate intuitionistic logic was first shown by [42]. The propositional case was further studied by [27].

Interpolation can be leveraged, among other applications, to solve constrained Horn clauses [30], for model checking [3, 32] or for invariant generation [23, 34]. Interpolants are computed by many existing solvers and provers such as Eldarica [19], Vampire [18] and Wolverine [25].

A sequent calculus proof system for orthologic was first described in [41], with cut elimination. [7] studied implication symbols in orthologic. [36] showed that orthologic with axioms is decidable. [37] showed that Orthologic admits the super amalgamation property, which implies that it admits interpolants. They also show that a predicate logic extension to orthologic admits interpolants, although the proof of both theorems are non-constructive and contain no discussion of algorithms or space and time complexity. Recently, orthologic was used in practice in a proof assistant [15], for modelling of epistemic logic [20, 21] and for normalizing formulas in software verification [14].

7 Conclusion

We showed that quantified orthologic, with a sequent-based proof system, is sound and complete with respect to all complete ortholattices. A soundness and completeness theorem typically allows demonstrating further provability results by using semantic arguments. We then showed that orthologic does not admit, in general, quantifier elimination. If such a procedure existed, it would have also allowed computing strongest and weakest interpolants. We instead presented an efficient algorithm, computing orthologic interpolants for two formulas, given a proof that one formula implies the other. Computing interpolants is a key part of some algorithms in model checking and program verification. Since orthologic has efficient algorithms to decide validity and compute proofs, which are necessary to compute interpolants, we expect that our present results will allow further development of orthologic-based tools and efficient algorithms for model checking and program verification.

References

1. Bell, J.L.: Orthologic, Forcing, and The Manifestation of Attributes. In: Chong, C.T., Wicks, M.J. (eds.) Studies in Logic and the Foundations of Mathematics. Studies in Logic and the Foundations of Mathematics, vol. 111, pp. 13–36. Elsevier, Singapore (Jan 1983). https://doi.org/10.1016/S0049-237X(08)70953-4
2. Birkhoff, G., Von Neumann, J.: The Logic of Quantum Mechanics. Annals of Mathematics **37**(4), 823–843 (1936). https://doi.org/10.2307/1968621
3. Bradley, A.R.: SAT-Based Model Checking without Unrolling. In: Jhala, R., Schmidt, D. (eds.) Verification, Model Checking, and Abstract Interpretation. pp. 70–87. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (2011). https://doi.org/10.1007/978-3-642-18275-4_7
4. Bruns, G., Godefroid, P.: Model Checking with Multi-valued Logics. In: Díaz, J., Karhumäki, J., Lepistö, A., Sannella, D. (eds.) Automata, Languages and Programming. pp. 281–293. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (2004). https://doi.org/10.1007/978-3-540-27836-8_26
5. Bruns, G.: Free Ortholattices. Canadian Journal of Mathematics **28**(5), 977–985 (Oct 1976). https://doi.org/10.4153/CJM-1976-095-6
6. Brzozowski, J.: De Morgan bisemilattices. In: Proceedings 30th IEEE International Symposium on Multiple-Valued Logic (ISMVL 2000). pp. 173–178 (May 2000). https://doi.org/10.1109/ISMVL.2000.848616
7. Chajda, I., Halaš, R.: An Implication in Orthologic. International Journal of Theoretical Physics **44**(7), 735–744 (Jul 2005). https://doi.org/10.1007/s10773-005-7051-1
8. Cook, S., Morioka, T.: Quantified propositional calculus and a second-order theory for NC1. Archive for Mathematical Logic **44**(6), 711–749 (Aug 2005). https://doi.org/10.1007/s00153-005-0282-2
9. Cousot, P., Cousot, R.: Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints. In: Proceedings of the 4th ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages. pp. 238–252. POPL '77, Association for Computing Machinery, New York, NY, USA (Jan 1977). https://doi.org/10.1145/512950.512973
10. Craig, W.: Three uses of the Herbrand-Gentzen theorem in relating model theory and proof theory. Journal of Symbolic Logic **22**(3), 269–285 (Sep 1957). https://doi.org/10.2307/2963594
11. Dudenhefner, A., Rehof, J.: A Simpler Undecidability Proof for System F Inhabitation. In: TYPES. p. 11 pages. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik GmbH, Wadern/Saarbruecken, Germany (2019). https://doi.org/10.4230/LIPICS.TYPES.2018.2
12. Gentzen, G.: Untersuchungen über das logische Schließen I. Mathematische Zeitschrift **39**, 176–210 (1935)
13. Girard, J.Y., Taylor, P., Lafont, Y.: Proofs and Types. Cambridge University Press, USA (Mar 1989)
14. Guilloud, S., Bucev, M., Milovancevic, D., Kuncak, V.: Formula Normalizations in Verification. In: 35th International Conference on Computer Aided Verification. pp. –. Lecture Notes in Computer Science, Springer, Paris (2023)
15. Guilloud, S., Gambhir, S., Kunčak, V.: LISA - A Modern Proof System. In: Naumowicz, A., Thiemann, R. (eds.) 14th International Conference on Interactive Theorem Proving (ITP 2023). Leibniz International Proceedings in

Informatics (LIPIcs), vol. 268, pp. 17:1–17:19. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany (2023). https://doi.org/10.4230/LIPIcs.ITP.2023.17, https://drops.dagstuhl.de/opus/volltexte/2023/18392

16. Guilloud, S., Kunčak, V.: Orthologic with axioms. Proc. ACM Program. Lang. **8**(POPL) (jan 2024)
17. Henzinger, T.A., Jhala, R., Majumdar, R., McMillan, K.L.: Abstractions from proofs. In: Jones, N.D., Leroy, X. (eds.) Proceedings of the 31st ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2004, Venice, Italy, January 14-16, 2004. pp. 232–244. ACM (2004). https://doi.org/10.1145/964001.964021
18. Hoder, K., Kovács, L., Voronkov, A.: Interpolation and Symbol Elimination in Vampire. In: Giesl, J., Hähnle, R. (eds.) Automated Reasoning, 5th International Joint Conference, IJCAR 2010, Edinburgh, UK, July 16-19, 2010. Proceedings. Lecture Notes in Computer Science, vol. 6173, pp. 188–195. Springer (2010). https://doi.org/10.1007/978-3-642-14203-1_16
19. Hojjat, H., Rummer, P.: The ELDARICA Horn Solver. 2018 Formal Methods in Computer Aided Design (FMCAD) pp. 1–7 (Oct 2018). https://doi.org/10.23919/FMCAD.2018.8603013
20. Holliday, W.H.: A Fundamental Non-Classical Logic. Logics **1**(1), 36–79 (Mar 2023). https://doi.org/10.3390/logics1010004
21. Holliday, W.H., Mandelkern, M.: The Orthologic of Epistemic Modals (2022). https://doi.org/10.48550/ARXIV.2203.02872
22. Kalmbach, G.: Orthomodular Lattices. Academic Press Inc, London ; New York (Mar 1983)
23. Kovács, L., Voronkov, A.: Finding Loop Invariants for Programs over Arrays Using a Theorem Prover. In: Chechik, M., Wirsing, M. (eds.) Fundamental Approaches to Software Engineering. pp. 470–485. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (2009). https://doi.org/10.1007/978-3-642-00593-0_33
24. Kozen, D.: Complexity of Boolean algebras. Theor. Comput. Sci. **10**, 221–247 (1980). https://doi.org/10.1016/0304-3975(80)90048-1
25. Kroening, D., Weissenbacher, G.: Interpolation-Based Software Verification with Wolverine. In: Gopalakrishnan, G., Qadeer, S. (eds.) Computer Aided Verification - 23rd International Conference, CAV 2011, Snowbird, UT, USA, July 14-20, 2011. Proceedings. Lecture Notes in Computer Science, vol. 6806, pp. 573–578. Springer (2011). https://doi.org/10.1007/978-3-642-22110-1_45
26. Kupferman, O., Lustig, Y.: Lattice automata. In: Cook, B., Podelski, A. (eds.) Verification, Model Checking, and Abstract Interpretation, 8th International Conference, VMCAI 2007, Nice, France, January 14-16, 2007, Proceedings. Lecture Notes in Computer Science, vol. 4349, pp. 199–213. Springer (2007). https://doi.org/10.1007/978-3-540-69738-1_14
27. de Lavalette, G.R.R.: Interpolation in Fragments of Intuitionistic Propositional Logic. The Journal of Symbolic Logic **54**(4), 1419–1430 (1989). https://doi.org/10.2307/2274823
28. MacNeille, H.M.: Partially ordered sets. Transactions of the American Mathematical Society **42**(3), 416–460 (1937). https://doi.org/10.1090/S0002-9947-1937-1501929-X
29. Madsen, M., Yee, M.H., Lhoták, O.: From Datalog to flix: A declarative language for fixed points on lattices. Proceedings of the 37th ACM SIGPLAN Conference

on Programming Language Design and Implementation pp. 194–208 (Jun 2016). https://doi.org/10.1145/2908080.2908096

30. McMillan, K., Rybalchenko, A.: Solving Constrained Horn Clauses using Interpolation. Tech. rep., Microsoft Research (2013)
31. McMillan, K.L.: Interpolation and sat-based model checking. In: Jr., W.A.H., Somenzi, F. (eds.) Computer Aided Verification, 15th International Conference, CAV 2003, Boulder, CO, USA, July 8-12, 2003, Proceedings. Lecture Notes in Computer Science, vol. 2725, pp. 1–13. Springer (2003). https://doi.org/10.1007/978-3-540-45069-6_1
32. McMillan, K.L.: Interpolation and SAT-Based Model Checking. In: Jr, W.A.H., Somenzi, F. (eds.) Computer Aided Verification, 15th International Conference, CAV 2003, Boulder, CO, USA, July 8-12, 2003, Proceedings. Lecture Notes in Computer Science, vol. 2725, pp. 1–13. Springer (2003). https://doi.org/10.1007/978-3-540-45069-6_1
33. McMillan, K.L.: Interpolants and Symbolic Model Checking. In: Cook, B., Podelski, A. (eds.) Verification, Model Checking, and Abstract Interpretation, 8th International Conference, VMCAI 2007, Nice, France, January 14-16, 2007, Proceedings. Lecture Notes in Computer Science, vol. 4349, pp. 89–90. Springer (2007). https://doi.org/10.1007/978-3-540-69738-1_6
34. McMillan, K.L.: Quantified Invariant Generation Using an Interpolating Saturation Prover. In: Ramakrishnan, C.R., Rehof, J. (eds.) Tools and Algorithms for the Construction and Analysis of Systems, 14th International Conference, TACAS 2008, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2008, Budapest, Hungary, March 29-April 6, 2008. Proceedings. Lecture Notes in Computer Science, vol. 4963, pp. 413–427. Springer (2008). https://doi.org/10.1007/978-3-540-78800-3_31
35. McMillan, K.L.: Interpolation and model checking. In: Clarke, E.M., Henzinger, T.A., Veith, H., Bloem, R. (eds.) Handbook of Model Checking, pp. 421–446. Springer International Publishing, Cham (2018). https://doi.org/10.1007/978-3-319-10575-8_14
36. Meinander, A.: A solution of the uniform word problem for ortholattices. Mathematical Structures in Computer Science **20**(4), 625–638 (Aug 2010). https://doi.org/10.1017/S0960129510000125
37. Miyazaki, Y.: The Super-Amalgamation Property of the Variety of Ortholattices. Reports Math. Log. **33**, 45–63 (1999)
38. Nielson, F., Nielson, H.R., Hankin, C.: Principles of Program Analysis. Springer Berlin Heidelberg, Berlin, Heidelberg (1999). https://doi.org/10.1007/978-3-662-03811-6
39. Pudlák, P.: The Lengths of Proofs. In: Studies in Logic and the Foundations of Mathematics, vol. 137, pp. 547–637. Elsevier (1998). https://doi.org/10.1016/S0049-237X(98)80023-2
40. Rümmer, P., Hojjat, H., Kuncak, V.: Disjunctive interpolants for horn-clause verification. In: Computer Aided Verification (CAV) (2013)
41. Schulte Mönting, J.: Cut elimination and word problems for varieties of lattices. Algebra Universalis **12**(1), 290–321 (Dec 1981). https://doi.org/10.1007/BF02483891
42. Schütte, K.: Der Interpolationssatz der intuitionistischen Prädikatenlogik. Mathematische Annalen **148**(3), 192–200 (Jun 1962). https://doi.org/10.1007/BF01470747

43. Sørensen, M., Urzyczyn, P.: Lectures on the Curry-Howard Isomorphism. Studies in Logic and the Foundations of Mathematics **149** (Oct 2010). https://doi.org/10.1016/S0049-237X(06)80005-4
44. Urzyczyn, P.: Inhabitation in typed lambda-calculi (a syntactic approach). In: Goos, G., Hartmanis, J., Leeuwen, J., Groote, P., Roger Hindley, J. (eds.) Typed Lambda Calculi and Applications. vol. 1210, pp. 373–389. Springer Berlin Heidelberg, Berlin, Heidelberg (1997). https://doi.org/10.1007/3-540-62688-3_47