Craig-Lyndon Interpolation for the Logic of Here and There with a Variation of Mints' Sequent System

Christoph Wernhard

University of Potsdam

CI-BD-SOQE 2026
Lisbon, July 24, 2026

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project-ID 457292495

HT and Strong Equivalence of Answer Set Programs Under Stable Models Semantics

Program	*Stable Models*
f ← b ∧ ¬p b	{{b, f}}
f ← b ∧ ¬p b p	{{b, p}}
p	{{p}}
p ← ¬q	{{p}}
p q	{{p, q}}
p ← ¬q q	{{q}}
p q	{{p, q}}
p q ← p	{{p, q}}

[Lifschitz, Pearce, Valverde: *Strongly equivalent logic programs*, 2001]

Definition

P and Q are ***strongly equivalent*** iff

For all R: StableModels($P \cup R$) = StableModels($Q \cup R$)

Theorem

P and Q are **strongly equivalent** (wrt. the stable model semantics) iff

P and Q are **equivalent in the logic of here and there (HT)**

[Heuer, W: *Synthesizing Strongly Equivalent Logic Programs: Beth Definability for Answer Set Programs via Craig Interpolation in First-Order Logic*, IJCAR 2024]

Synthesizing for given programs P, Q programs R in a given vocabulary V s.th.

$P \cup R$ and $P \cup Q$ are **strongly equivalent**

$$Q = \begin{array}{l} \mathsf{p} \leftarrow \mathsf{q} \land \mathsf{r} \\ \mathsf{p} \lor \mathsf{q} \leftarrow \mathsf{r} \\ \mathsf{q} \leftarrow \mathsf{q} \land \mathsf{s} \end{array} \qquad V = \{\mathsf{p}, \mathsf{r}\}$$

$$R = \mathsf{p} \leftarrow \mathsf{r}$$

$$P = \mathsf{p} \leftarrow \mathsf{q} \qquad Q = \begin{array}{l} \mathsf{r} \leftarrow \mathsf{p} \\ \mathsf{r} \leftarrow \mathsf{q} \end{array} \qquad V = \{\mathsf{p}, \mathsf{r}\}$$

$$R = \mathsf{r} \leftarrow \mathsf{p}$$

$$P = \bot \leftarrow \mathsf{p} \land \mathsf{q} \qquad Q = \mathsf{r} \leftarrow \mathsf{p} \land \neg \mathsf{q} \qquad V = \{\mathsf{p}, \mathsf{r}\}$$

$$R = \mathsf{r} \leftarrow \mathsf{p}$$

$$P = \begin{array}{l} \mathsf{p} \leftarrow \mathsf{q} \land \neg \mathsf{r} \\ \mathsf{p} \leftarrow \mathsf{s} \\ \neg \mathsf{r} \lor \mathsf{s} \leftarrow \mathsf{p} \\ \mathsf{q} \lor \mathsf{s} \leftarrow \mathsf{p} \end{array} \qquad Q = \mathsf{t} \leftarrow \mathsf{p} \qquad V = \{\mathsf{q}, \mathsf{r}, \mathsf{s}, \mathsf{t}\}$$

$$R = \begin{array}{l} \mathsf{t} \leftarrow \mathsf{q} \land \neg \mathsf{r} \\ \mathsf{t} \leftarrow \mathsf{s} \end{array}$$

- Idea: P expresses a schema mapping from client predicate p to KB predicates V
  R is a rewriting of client query Q in terms of KB predicates
- Only the first two rules of P actually describe the mapping, the other two complete them
- Effects unfolding of p
- Also works with R and Q switched and $V = \{\mathsf{p}, \mathsf{t}\}$: then it effects folding into p

HT and a Variation of HT

- The greatest intermediate, or superintuitionistic, logic properly included in classical propositional logic
- Introduced by Heyting in 1930
- Aka: **Gödel's G$_3$**, SmL (Smetanich Logic)
- ***HT-formula***: $A, B \;:=\; Atom \mid \bot \mid \top \mid \neg A \mid (A \lor B) \mid (A \land B) \mid (A \to B)$
- We also consider a variation of HT, with a **further logic operator** nh ***(not here)***:

  NHNNF-formula: $A, B \;:=\; Atom \mid \neg Atom \mid \neg\neg Atom \mid \mathrm{nh}(Atom) \mid \bot \mid \top \mid (A \lor B) \mid (A \land B)$

Here	There	Value		
$\bot$	$\bot$	F	*false*	0
$\bot$	$\top$	NF	*not false*	0.5
$\top$	$\top$	T	*true*	1

$\bot$	$\top$
F	T

A	$\neg A$	nh(A)
F	T	T
NF	F	T
T	F	F

A	B	$A \lor B$	$A \land B$	$A \to B$
F	F	F	F	T
F	NF	NF	F	T
F	T	T	F	T
NF	F	NF	F	F
NF	NF	NF	NF	T
NF	T	T	NF	T
T	F	T	F	F
T	NF	T	NF	NF
T	T	T	T	T

More on HT and Variations

A	$\neg A$	nh(A)
F	T	T
NF	F	T
T	F	F

A	B	$A \vee B$	$A \wedge B$	$A \to B$
F	F	F	F	T
F	NF	NF	F	T
F	T	T	F	T
NF	F	NF	F	F
NF	NF	NF	NF	T
NF	T	T	NF	T
T	F	T	F	F
T	NF	T	NF	NF
T	T	T	T	T

- $A \models B$ iff for all assignments v: $\text{Val}_v(A) \leq \text{Val}_v(B)$
- For **HT-formulas** this is equivalent to
  for all assignments v: $\text{Val}_v(A) = \text{T}$ implies $\text{Val}_v(B) = \text{T}$
- The following statements are equivalent:
  (i) $A \models B$ (ii) $\models A \to B$ (iii) For all assignments v: $\text{Val}_v(A \to B) = \text{T}$
- Let $A \leftrightarrow B \ := \ (A \to B) \wedge (B \to A)$
  The following statements are equivalent:
  (i) $A \equiv B$ (ii) $\models A \leftrightarrow B$ (iii) For all assignments v: $\text{Val}_v(A) = \text{Val}_v(B)$
- Some propositions

$$
\begin{array}{rcl}
A \wedge \neg A & \equiv & \bot \\
 & \not\models & A \vee \neg A \\
 & \models & \neg A \vee \neg\neg A \\
A & \models & \neg\neg A \\
(A \wedge B) \to C & \equiv & A \to (B \to C) \\
\neg(A \to B) & \equiv & \neg(\neg A \vee B) \ \equiv \ \neg\neg A \wedge \neg B
\end{array}
$$

$$
\begin{array}{rcl}
A \wedge \text{nh}(A) & \not\equiv & \bot \\
 & \models & A \vee \text{nh}(A) \\
\neg A & \models & \text{nh}(A) \\
\text{nh}(\neg A) & \equiv & \neg\neg A \\
\text{nh}(A \wedge B) & \equiv & \text{nh}(A) \vee \text{nh}(B) \\
\text{nh}(A \vee B) & \equiv & \text{nh}(A) \wedge \text{nh}(B) \\
A \to B & \equiv & (\text{nh}(A) \vee B) \wedge (\neg A \vee \neg\neg B)
\end{array}
$$

[Mints: *Cut-free Formulations for a Quantified Logic of Here and There*, 2010]

Axioms (here with atom/literal restrictions)

Ax-1	$A, \Gamma \Rightarrow \Delta, A$	A a literal	$\models (A \land B) \to (C \lor A)$
Ax-2	$A, \neg A, \Gamma \Rightarrow \Delta$	A an atom	$\models (A \land \neg A \land B) \to C$

Familiar classical rules for $\land$, $\lor$, new rules for $\to$

$$\to\Rightarrow \quad \frac{\neg A, \Gamma \Rightarrow \Delta \qquad \Gamma \Rightarrow \Delta, A, \neg B \qquad B, \Gamma \Rightarrow \Delta}{(A \to B), \Gamma \Rightarrow \Delta}$$

$$\begin{aligned} &((A \to B) \land C) \to D \\ \equiv\ &((\neg A \land C) \to D) \land \\ &(C \to (D \lor A \lor \neg B)) \land \\ &((B \land C) \to D) \end{aligned}$$

$$\Rightarrow\to \quad \frac{A, \Gamma \Rightarrow \Delta, B \qquad \neg B, \Gamma \Rightarrow \Delta, \neg A}{\Gamma \Rightarrow \Delta, (A \to B)}$$

$$\begin{aligned} &C \to (D \lor (A \to B)) \\ \equiv\ &((A \land C) \to (D \lor B)) \land \\ &((\neg B \land C) \to (D \lor \neg A)) \end{aligned}$$

Rules for moving negation inside over $\land$, $\lor$

$\neg\neg$ rules

$$\frac{\Gamma \Rightarrow \Delta, \neg A}{\neg\neg A, \Gamma \Rightarrow \Delta} \qquad \begin{aligned} &(\neg\neg A \land C) \to D \\ \equiv\ &C \to (D \lor \neg A) \end{aligned}$$

$$\frac{\neg A, \Gamma \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, \neg\neg A} \qquad \begin{aligned} &C \to (D \lor \neg\neg A) \\ \equiv\ &(\neg A \land C) \to D \end{aligned}$$

Axioms

Ax-nh-1	$\Gamma \Rightarrow \Delta, A, \mathrm{nh}(A)$	A an atom	$\models B \to (C \vee A \vee \mathrm{nh}(A))$
Ax-nh-2	$\neg A, \Gamma \Rightarrow \Delta, \mathrm{nh}(A)$	A an atom	$\models (\neg A \wedge B) \to (C \vee \mathrm{nh}(A))$

Rules based on the following equivalences:

$$\mathrm{nh}(\neg A) \equiv \neg\neg A \qquad \mathrm{nh}(A \wedge B) \equiv \mathrm{nh}(A) \vee \mathrm{nh}(B) \qquad \mathrm{nh}(A \vee B) \equiv \mathrm{nh}(A) \wedge \mathrm{nh}(B)$$

A rule for ¬nh on the left

$$\neg\mathrm{nh}{\Rightarrow} \quad \frac{\Gamma \Rightarrow \mathrm{nh}(A), \Delta}{\neg\mathrm{nh}(A), \Gamma \Rightarrow \Delta} \qquad \begin{array}{ll} & (\neg\mathrm{nh}(A) \wedge B) \to C \\ \equiv & B \to (\mathrm{nh}(A) \vee C) \end{array}$$

A third new rule for implication, required for interpolation

$$\Rightarrow\to* \quad \frac{\Gamma \Rightarrow \Delta, \mathrm{nh}(A), B \qquad \neg B, \Gamma \Rightarrow \Delta, \neg A}{\Gamma \Rightarrow \Delta, (A \to B)} \qquad \begin{array}{ll} & C \to (D \vee (A \to B)) \\ \equiv & ((C \to (D \vee \mathrm{nh}(A) \vee B)) \wedge \\ & ((\neg B \wedge C) \to (D \vee \neg A)) \end{array}$$

Recall Mints' corresponding rule

$$\Rightarrow\to \quad \frac{A, \Gamma \Rightarrow \Delta, B \qquad \neg B, \Gamma \Rightarrow \Delta, \neg A}{\Gamma \Rightarrow \Delta, (A \to B)} \qquad \begin{array}{ll} & C \to (D \vee (A \to B)) \\ \equiv & ((A \wedge C) \to (D \vee B)) \wedge \\ & ((\neg B \wedge C) \to (D \vee \neg A)) \end{array}$$

Craig-Lyndon Interpolation for HT in Two Stages – via Two Techniques

It is common to **verify strong equivalence (equivalence in HT) via classical encodings**

- Atoms a are duplicated to a^h and a^t, with axioms $a^h \rightarrow a^t$
- Implications $(p \land \neg q) \rightarrow (r \lor \neg s)$ become: $(p^h \land \neg q^t) \rightarrow (r^h \lor \neg s^t)$
  $(p^t \land \neg q^t) \rightarrow (r^t \lor \neg s^t)$

Here	There	Value
$\bot$	$\bot$	F
$\bot$	$\top$	NF
$\top$	$\top$	T

A	$\neg A$
F	T
NF	F
T	F

We compute Craig-Lyndon interpolants H in HT for HT-formulas A, B

Via classical encoding [HW 2024]

1. Compute a classical Craig-Lyndon interpolant H' for classical encodings A', B' of A, B
   - H' might not encode a HT-formula
2. Convert H' to a stronger classical formula H'' that encodes a Craig-Lyndon interpolant H for A, B
   - Involves CNF conversion
   - Semantic justification relies on modifying a proof of $A' \models H'$

- Implemented, on basis of *Prover9* or *CMProver*
- Straightforward first-order generalization

Via a variation of Mints' sequent system

1. Compute a Craig-Lyndon interpolant H' for A, B with Maehara's method applied to $A \Rightarrow B$
   - H' is an NHNNF-formula
2. Convert H' to a stronger formula H in HT that is a Craig-Lyndon interpolant for A, B
   - Involves CNF conversion
   - Semantic justification relies on modifying a proof of $\neg\neg A \models \neg\neg H'$

Split-sequents

$$\Gamma^{\mathsf{L}}, \Gamma^{\mathsf{R}} \overset{H}{\Longrightarrow} \Delta^{\mathsf{L}}, \Delta^{\mathsf{R}}$$

where $\bigwedge \Gamma^{\mathsf{L}} \models H \lor \bigvee \Delta^{\mathsf{L}}$
$\bigwedge \Gamma^{\mathsf{R}} \land H \models \bigvee \Delta^{\mathsf{R}}$
$\mathsf{voc}^{\pm}(H) \subseteq \mathsf{voc}^{\pm}(\bigwedge \Gamma^{\mathsf{L}} \land \neg \bigvee \Delta^{\mathsf{L}}) \cap \mathsf{voc}^{\pm}(\neg \bigwedge \Gamma^{\mathsf{R}} \lor \bigvee \Delta^{\mathsf{R}})$

Split variations of axioms

$$A^{\mathsf{L}}, \Gamma \overset{\bot}{\Longrightarrow} \Delta, A^{\mathsf{L}} \qquad A^{\mathsf{L}}, \Gamma \overset{A}{\Longrightarrow} \Delta, A^{\mathsf{R}} \qquad A^{\mathsf{R}}, \Gamma \overset{\neg A}{\Longrightarrow} \Delta, A^{\mathsf{L}} \qquad A^{\mathsf{R}}, \Gamma \overset{\top}{\Longrightarrow} \Delta, A^{\mathsf{R}}$$

Split variation of rules, e.g.

$$\Rightarrow\land\mathsf{L}\ \frac{\Gamma \overset{H_1}{\Longrightarrow} \Delta, A^{\mathsf{L}} \qquad \Gamma \overset{H_2}{\Longrightarrow} \Delta, B^{\mathsf{L}}}{\Gamma \overset{H_1 \lor H_2}{\Longrightarrow} \Delta, (A \land B)^{\mathsf{L}}} \qquad \Rightarrow\land\mathsf{R}\ \frac{\Gamma \overset{H_1}{\Longrightarrow} \Delta, A^{\mathsf{R}} \qquad \Gamma \overset{H_2}{\Longrightarrow} \Delta, B^{\mathsf{R}}}{\Gamma \overset{H_1 \land H_2}{\Longrightarrow} \Delta, (A \land B)^{\mathsf{R}}}$$

Root

$$A^{\mathsf{L}} \overset{H}{\Longrightarrow} B^{\mathsf{R}}$$

Proving, provenance propagation

Interpolant construction

Theorem 1

Let A, B be **HT-formulas** s.th. $A \models B$. Then there exists an **NHNNF-formula** C' s.th.

1. $A \models C'$ and $C' \models B$, and
2. $\mathrm{voc}^{\pm}(C') \subseteq \mathrm{voc}^{\pm}(A) \cap \mathrm{voc}^{\pm}(B)$

Moreover, C' can be **constructed** from a proof of $A \models B$ in our variation of Mints' system

We assume B is ***body-normalized***: no implication occurs in the antecedent of another implication, w.l.o.g:

$$(A \to B) \to C \equiv (\neg A \to C) \land (A \lor \neg B \lor C) \land (B \to C) \qquad (A \lor B) \to C \equiv (A \to C) \land (B \to C)$$
$$(A \land B) \to C \equiv (A \to C) \lor (B \to C) \qquad \neg A \to B \equiv \neg\neg A \lor B$$

Lemma

For all split-sequents $\Gamma \xRightarrow{H} \Delta$ in a proof of $A^{\mathsf{L}} \xRightarrow{C'} B^{\mathsf{R}}$ in our system:

(i) It is a proper split sequent
(ii) H is an NHNNF-formula
(iii) All occurrences of nh have provenance R
(iv) All occurrences of nh are as top-level operators of members of Δ
(v) Argument formulas of nh have no occurrences of $\to$ and nh
(vi) All members of Γ^{R} are negations, i.e., have $\neg$ as top-level operator

$$\bigwedge \Gamma^{\mathsf{L}} \models H \lor \bigvee \Delta^{\mathsf{L}}$$
$$\bigwedge \Gamma^{\mathsf{R}} \land H \models \bigvee \Delta^{\mathsf{R}}$$
$$\mathrm{voc}^{\pm}(H) \subseteq \mathrm{voc}^{\pm}(\ldots) \cap \mathrm{voc}^{\pm}(\ldots)$$

			$\bigwedge \Gamma^{\mathsf{L}} \models H \lor \bigvee \Delta^{\mathsf{L}}$	$\bigwedge \Gamma^{\mathsf{R}} \land H \models \bigvee \Delta^{\mathsf{R}}$
Ax-1-LL	$A^{\mathsf{L}}, \Gamma \xrightarrow{\bot} \Delta, A^{\mathsf{L}}$	A a literal		
Ax-1-LR	$A^{\mathsf{L}}, \Gamma \xrightarrow{A} \Delta, A^{\mathsf{R}}$	A a literal		
Ax-1-RL	$(\neg A)^{\mathsf{R}}, \Gamma \xrightarrow{\neg\neg A} \Delta, (\neg A)^{\mathsf{L}}$	A an atom	$\bigwedge \Gamma^{\mathsf{L}} \models \neg\neg A \lor \bigvee \Delta^{\mathsf{L}} \lor \neg A$	$\neg A \land \bigwedge \Gamma^{\mathsf{R}} \land \neg\neg A \models \bigvee \Delta^{\mathsf{R}}$
Ax-1-RR	$(\neg A)^{\mathsf{R}}, \Gamma \xrightarrow{\top} \Delta, (\neg A)^{\mathsf{R}}$	A an atom		
Ax-2-LL	$A^{\mathsf{L}}, (\neg A)^{\mathsf{L}}, \Gamma \xrightarrow{\bot} \Delta$	A an atom		
Ax-2-LR	$A^{\mathsf{L}}, (\neg A)^{\mathsf{R}}, \Gamma \xrightarrow{A} \Delta$	A an atom	$A \land \bigwedge \Gamma^{\mathsf{L}} \models A \lor \bigvee \Delta^{\mathsf{L}}$	$\bigwedge \Gamma^{\mathsf{R}} \land \neg A \land A \models \bigvee \Delta^{\mathsf{R}}$
Ax-2-LR-Alt	$A^{\mathsf{L}}, (\neg A)^{\mathsf{R}}, \Gamma \xrightarrow{\neg\neg A} \Delta$	A an atom	$A \land \bigwedge \Gamma^{\mathsf{L}} \models \neg\neg A \lor \bigvee \Delta^{\mathsf{L}}$	$\bigwedge \Gamma^{\mathsf{R}} \land \neg A \land \neg\neg A \models \bigvee \Delta^{\mathsf{R}}$

Remarks

- By Lemma (vi), split variations Ax-1 where an R-atom is a member of the antecedent are superfluous
- For **Ax-1-RL the restriction to negative literals as principal formulas is crucial** since for $A^{\mathsf{R}}, \Gamma \xrightarrow{H} \Delta, A^{\mathsf{L}}$ there is no H s.th.

$$\bigwedge \Gamma^{\mathsf{L}} \models H \lor \bigvee \Delta^{\mathsf{L}} \lor A \quad \text{and} \quad A \land \bigwedge \Gamma^{\mathsf{R}} \land H \models \bigvee \Delta^{\mathsf{R}}$$

Interpolating Variations of Ax-nh-1 and Ax-nh-2

			$\bigwedge \Gamma^{\mathsf{L}} \models H \vee \bigvee \Delta^{\mathsf{L}}$	$\bigwedge \Gamma^{\mathsf{R}} \wedge H \models \bigvee \Delta^{\mathsf{R}}$
Ax-nh-1-LR	$\Gamma \overset{\mathsf{nh}(A)}{\Longrightarrow} \Delta, A^{\mathsf{L}}, \mathsf{nh}(A)^{\mathsf{R}}$	A an atom	$\bigwedge \Gamma^{\mathsf{L}} \models \mathsf{nh}(A) \vee \bigvee \Delta^{\mathsf{L}} \vee A$	$\Gamma^{\mathsf{R}} \wedge \mathsf{nh}(A) \models \bigvee \Delta^{\mathsf{R}} \vee \mathsf{nh}(A)$
Ax-nh-1-RR	$\Gamma \overset{\top}{\Longrightarrow} \Delta, A^{\mathsf{R}}, \mathsf{nh}(A)^{\mathsf{R}}$	A an atom		
Ax-nh-2-LR	$(\neg A)^{\mathsf{L}}, \Gamma \overset{\neg A}{\Longrightarrow} \Delta, \mathsf{nh}(A)^{\mathsf{R}}$	A an atom		
Ax-nh-2-RR	$(\neg A)^{\mathsf{R}}, \Gamma \overset{\top}{\Longrightarrow} \Delta, \mathsf{nh}(A)^{\mathsf{R}}$	A an atom		

Remarks

- By Lemmas (iii) and (iv) axioms Ax-nh-1 and Ax-nh-2 are only needed in the shown split variations
- **Ax-nh-1-LR is the only place where nh is introduced into the constructed interpolant**

For provenance L we use Mints' rules

$$\to\Rightarrow\mathsf{L}\quad \frac{\neg A^{\mathsf{L}}, \Gamma \overset{H_1}{\Longrightarrow} \Delta \qquad \Gamma \overset{H_2}{\Longrightarrow} \Delta, A^{\mathsf{L}}, (\neg B)^{\mathsf{L}} \qquad B^{\mathsf{L}}, \Gamma \overset{H_3}{\Longrightarrow} \Delta}{(A \to B)^{\mathsf{L}}, \Gamma \overset{H_1 \vee H_2 \vee H_3}{\Longrightarrow} \Delta}$$

$$\Rightarrow\to\mathsf{L}\quad \frac{A^{\mathsf{L}}, \Gamma \overset{H_1}{\Longrightarrow} \Delta, B^{\mathsf{L}} \qquad (\neg B)^{\mathsf{L}}, \Gamma \overset{H_2}{\Longrightarrow} \Delta, (\neg A)^{\mathsf{L}}}{\Gamma \overset{H_1 \vee H_2}{\Longrightarrow} \Delta, (A \to B)^{\mathsf{L}}}$$

- By Lemma (vi) a split variation for implication with provenance R in the antecedent is superfluous

For implication with provenance R in the succedent we need our new rule

$$\Rightarrow\to{*}\mathsf{R}\quad \frac{\Gamma \overset{H_1}{\Longrightarrow} \Delta, \mathsf{nh}(A)^{\mathsf{R}}, B^{\mathsf{R}} \qquad (\neg B)^{\mathsf{R}}, \Gamma \overset{H_2}{\Longrightarrow} \Delta, (\neg A)^{\mathsf{R}}}{\Gamma \overset{H_1 \wedge H_2}{\Longrightarrow} \Delta, (A \to B)^{\mathsf{R}}}$$

- Justification of the semantic conditions:
  If $\bigwedge \Gamma^{\mathsf{L}} \vDash H_1 \vee \bigvee \Delta^{\mathsf{L}}$ and $\bigwedge \Gamma^{\mathsf{L}} \vDash H_2 \vee \bigvee \Delta^{\mathsf{L}}$, then $\bigwedge \Gamma^{\mathsf{L}} \vDash (H_1 \wedge H_2) \vee \bigvee \Delta^{\mathsf{L}}$
  If $\bigwedge \Gamma^{\mathsf{R}} \wedge H_1 \vDash \bigvee \Delta^{\mathsf{R}} \vee \mathsf{nh}(A) \vee B$ and $\neg B \wedge \bigwedge \Gamma^{\mathsf{R}} \wedge H_2 \vDash \bigvee \Delta^{\mathsf{R}} \vee \neg A$, then
  $\bigwedge \Gamma^{\mathsf{R}} \wedge (H_1 \wedge H_2) \vDash \bigvee \Delta^{\mathsf{R}} \vee (A \to B)$

Omitted Interpolating Rule Variations

- Axioms and rules for the truth value constants
- 8 split variations of the familiar rules for conjunction and disjunction
- 4 rules for $\neg\neg$
- 12 rules for moving negation inward
- 1 rule for eliminating nh over negation
- 2 rules for moving nh inward over $\land$, $\lor$
- By Lemma (iv) the rule $\neg\mathsf{nh}{\Rightarrow}$ is superfluous

- **Termination**: each rule reduces $\langle |BinaryLogops|, |UnaryLogops| \rangle$
- If $\Gamma \overset{H}{\Longrightarrow} \Delta$ meets Lemmas (iii)–(vi) and Γ or Δ has a member that is not A, $\neg A$, or nh(A), with A an atom, then some rule is applicable
- For a leaf $\Gamma \overset{H}{\Longrightarrow} \Delta$ to which no rule is applicable and which is no instance of an axiom we can **construct a refuting model** v by assigning truth values to all atoms A as follows

If $A \in \Gamma$, then $A :=$ T,
else if nh$(A) \in \Delta$, then $A :=$ T,
else if $\neg A \in \Delta$, then $A :=$ NF,
else $A :=$ F

$\Delta \backslash \Gamma$	–	A	$\neg A$
–	F	T	F
A	F	×	F
$\neg A$	NF	T	×
nh(A)	T	T	×
$A, \neg A$	NF	×	×
$\neg A$, nh(A)	T	T	×

A	$\neg A$	nh(A)
F	T	T
NF	F	T
T	F	F

$A, \Gamma \Rightarrow \Delta, A$	A literal
$A, \neg A, \Gamma \Rightarrow \Delta$	A atom
$\Gamma \Rightarrow \Delta, A$, nh$(A)$	A atom
$\neg A, \Gamma \Rightarrow \Delta$, nh$(A)$	A atom

  - We have **Val**$_v(\bigwedge \Gamma) =$ T, while **Val**$_v(\bigvee \Delta) \in \{F, NF\}$. Hence, **Val**$_v(\bigwedge \Gamma \to \bigvee \Delta) \in \{F, NF\} \neq$ T
  - Sine our rules are invertible and sound, $A \to B$ for the interpolation arguments A, B is equivalent to the conjunction of $\bigwedge \Gamma \to \bigvee \Delta$ for all leaves of the proof tree that are not closed by an axiom
  - If for one of these “leaf conjuncts” $\bigwedge \Gamma \to \bigvee \Delta$ it holds that Val$_v(\bigwedge \Gamma \to \bigvee \Delta) \in \{F, NF\}$, then Val$_v(A \to B) \in \{F, NF\} \neq$ T

Theorem 2

Let A be an **HT-formula** and let C' be an **NHNNF-formula** s.th. $A \models C'$.
Then there exists an **HT-formula** C s.th. $A \models C$, $C \models C'$ and $\mathrm{voc}^{\pm}(C) \subseteq \mathrm{voc}^{\pm}(C')$

Moreover, C can be **constructed** from C'

The Construction

Convert C' to a CNF with clauses $\mathrm{nh}(E_1) \lor \ldots \lor \mathrm{nh}(E_n) \lor F$, where F has no occurrences of nh, followed by rewriting each clause to $(E_1 \land \ldots \land E_n) \rightarrow F$

Proof Sketch

(1) Let $D(\mathrm{nh}) = \mathrm{nh}(E_1) \lor \ldots \lor \mathrm{nh}(E_n) \lor F$ be a clause of the CNF of C'
(2) $A \models D(\mathrm{nh})$
(3) $\neg\neg A \models \neg\neg D(\mathrm{nh})$
(4) There is a proof $\mathit{Proof}(\mathrm{nh}, \text{Ax-nh-2})$ s.th. $\mathit{Proof}(\mathrm{nh}, \text{Ax-nh-2}) \vdash \neg\neg A \Rightarrow \neg\neg D(\mathrm{nh})$
(5) $\mathit{Proof}(\neg, \text{Ax-1}) \vdash \neg\neg A \Rightarrow \neg\neg D(\neg)$
(6) $A \models \neg\neg A \models \neg\neg D(\neg)$
(7) $A \models D(\mathrm{nh}) \land \neg\neg D(\neg)$
(8) $A \models (E_1 \land \ldots \land E_n) \rightarrow F$

- Ax-nh-1 ($\Gamma \Rightarrow \Delta, A, \mathrm{nh}(A)$; A an atom) is not in Proof: the sequents have no unsigned atoms as members
- For (5) we replace in Proof nh by $\neg$ and Ax-nh-2 ($\neg A, \Gamma \Rightarrow \Delta, \mathrm{nh}(A)$) by Ax-1 ($\neg A, \Gamma \Rightarrow \Delta, \neg A$)
- For (8) we use

$$
\begin{aligned}
& D(\mathrm{nh}) \land \neg\neg D(\neg) \\
\equiv\ & (\mathrm{nh}(E_1) \lor \ldots \lor \mathrm{nh}(E_n) \lor F) \land \neg\neg(\neg E_1 \lor \ldots \lor \neg E_n \lor F) \\
\equiv\ & (\mathrm{nh}(E_1) \lor \ldots \lor \mathrm{nh}(E_n) \lor F) \land (\neg E_1 \lor \ldots \lor \neg E_n \lor \neg\neg F) \\
\equiv\ & (E_1 \land \ldots \land E_n) \rightarrow F
\end{aligned}
$$

Theorem 1

Let A, B be **HT-formulas** s.th. $A \models B$. Then there exists an **NHNNF-formula** C' s.th.

1. $A \models C'$ and $C' \models B$, and
2. $\text{voc}^{\pm}(C') \subseteq \text{voc}^{\pm}(A) \cap \text{voc}^{\pm}(B)$

Moreover, C' can be constructed from a proof of $A \models B$ in our variation of Mints' system

Theorem 2

Let A be an **HT-formula** and let C' be an **NHNNF-formula** s.th. $A \models C'$.
Then there exists an **HT-formula** C s.th. $A \models C$, $C \models C'$ and $\text{voc}^{\pm}(C) \subseteq \text{voc}^{\pm}(C')$

Moreover, C can be constructed from C'

Theorem 3

Let A, B be **HT-formulas** such that $A \models B$. Then there exists an **HT-formula** C s.th.

1. $A \models C$ and $C \models B$, and
2. $\text{voc}^{\pm}(C) \subseteq \text{voc}^{\pm}(A) \cap \text{voc}^{\pm}(B)$

Moreover, C can be constructed from a proof of $A \models B$ in our variation of Mints' system

Some Open Issues, Outlook

- The construction involves **two expensive steps** (aside of proving):
  1. In stage 1 preprocessing the second input B to **body-normalized** form
  2. In stage 2 the **CNF conversion**

  - Logic programs typically come in CNF-like and body-normalized form
  - Mints' system can be seen as a compiler to this CNF-like form
  - Are definitional normal forms known/used, e.g., for HT?

- **Polarity** ("Lyndon"-interpolation) seems refinable to **4 cases** (see also [HW 2024]): $(p \land \neg q) \to (r \lor \neg s)$

- On **uniform interpolation for HT**
  - [Baaz, Veith 1999]: $\exists p\, A(p, q, r) \;\equiv\; A(\top, q, r) \lor A(\bot, q, r) \lor A(q, q, r) \lor A(r, q, r)$
  - Left-uniform in two stages (work in progress):
    1. $\forall p\, A(p) \;\equiv\; A(\top) \land A(\bot) \land A(\mathsf{nf})$
    2. Strengthening to a HT-formula, utilizing an interesting characteristics [Aguado et al. 2015] of HT as *total-closed*: if $\mathrm{Val}_v(A) = \top$, then also $\mathrm{Val}_{v\{\mathrm{NF}\mapsto\mathrm{T}\}}(A) = \top$

- Is there something deeper behind the apparent necessity of **two stages**?

Summary: Craig-Lyndon Interpolation for HT in Two Stages

Via classical encoding [HW 2024]

1. Compute a **classical Craig-Lyndon interpolant** H' for **classical encodings** A', B' of A, B
   - H' might **not encode a HT-formula**
2. Convert H' to a **stronger classical formula** H'' **that encodes** a Craig-Lyndon interpolant H for A, B
   - Involves CNF conversion
   - Semantic justification relies on **modifying a proof** of $A' \models H'$

- Implemented, on basis of *Prover9* or *CMProver*
- Straightforward first-order generalization

Via a variation of Mints' sequent system

1. Compute a Craig-Lyndon interpolant H' for A, B with **Maehara's method** applied to $A \Rightarrow B$
   - H' is an **NHNNF-formula**
2. Convert H' to a **stronger formula** H **in HT** that is a Craig-Lyndon interpolant for A, B
   - Involves CNF conversion
   - Semantic justification relies on **modifying a proof** of $\neg\neg A \models \neg\neg H'$

References I

[Aguado et al., 2015] Aguado, F., Cabalar, P., Pearce, D., Pérez, G., and Vidal, C. (2015).
A denotational semantics for equilibrium logic.
Theory Pract. Log. Program., 15(4–5):620–634.

[Baaz and Veith, 1999] Baaz, M. and Veith, H. (1999).
Interpolation in fuzzy logic.
Arch. Math. Logic, 38:461–489.

[Heuer and Wernhard, 2024] Heuer, J. and Wernhard, C. (2024).
Synthesizing strongly equivalent logic programs: Beth definability for answer set programs via Craig interpolation in first-order logic.
In Benzmüller, C., Heule, M., and Schmidt, R., editors, *IJCAR 2024*, volume 14739 of *LNCS (LNAI)*, pages 172–193. Springer.

[Lifschitz et al., 2001] Lifschitz, V., Pearce, D., and Valverde, A. (2001).
Strongly equivalent logic programs.
ACM Trans. Comp. Log., 2(4):526–541.

[Mints, 2010] Mints, G. (2010).
Cut-free formulations for a quantified logic of here and there.
Ann. Pure Appl. Log., 162(3):237–242.