Machine-checked Interpolation Theorems for Substructural Logics using Display Calculi

Jeremy E. Dawson James Brotherston Rajeev Goré

Research School of Computer Science, Australian National University

University College London, UK

IJCAR, Coimbra, 28 June 2016

Craig interpolation

Definition

A (propositional) logic satisfies Craig interpolation iff for any provable $F \vdash G$ there exists an interpolant I s.t.:

$$F \vdash I \text{ provable and } I \vdash G \text{ provable and } \mathcal{V}(I) \subseteq \mathcal{V}(F) \cap \mathcal{V}(G)$$

($\mathcal{V}(X)$ is the set of propositional variables occurring in X)

Craig interpolation

Definition

A (propositional) logic satisfies Craig interpolation iff for any provable $F \vdash G$ there exists an interpolant I s.t.:

$$F \vdash I \text{ provable and } I \vdash G \text{ provable and } \mathcal{V}(I) \subseteq \mathcal{V}(F) \cap \mathcal{V}(G)$$

($\mathcal{V}(X)$ is the set of propositional variables occurring in X)

Applications in:

- logic: consistency; compactness; definability

Craig interpolation

Definition

A (propositional) logic satisfies Craig interpolation iff for any provable $F \vdash G$ there exists an interpolant I s.t.:

$$F \vdash I \text{ provable and } I \vdash G \text{ provable and } \mathcal{V}(I) \subseteq \mathcal{V}(F) \cap \mathcal{V}(G)$$

($\mathcal{V}(X)$ is the set of propositional variables occurring in X)

Applications in:

- logic: consistency; compactness; definability
- computer science: invariant generation; type inference; model checking; ontology decomposition

Interpolation via sequent calculi

Sequent Calculus:

$$(\vdash \wedge)\ \frac{\Gamma \vdash A, \Delta \qquad \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta} \qquad (\wedge \vdash)\ \frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

Cut Rule: usually eliminable

$$\frac{\Gamma \vdash A, \Delta \qquad \Gamma, A \vdash \Delta}{\Gamma \vdash \Delta}$$

Interpolation: constructive, by induction on cut-free proofs

$$(\vdash \wedge)\ \frac{\Gamma \vdash^{F_A} A, \Delta \qquad \Gamma \vdash^{F_B} B, \Delta}{\Gamma \vdash^{F_A \wedge F_B} A \wedge B, \Delta} \qquad (\wedge \vdash)\ \frac{\Gamma, A, B \vdash^{F_{A \wedge B}} \Delta}{\Gamma, A \wedge B \vdash^{F_{A \wedge B}} \Delta}$$

Display calculi: a modular sequent calculus framework

Structures: extra structural connectives beyond Gentzen's comma

$$X :== A \mid \emptyset \mid \sharp X \mid X ; X$$

Display Postulates: extra rules to dis-/re- assemble structures e.g.

$$X ; Y \vdash Z \quad \rightleftarrows_D \quad X \vdash \sharp Y ; Z \quad \rightleftarrows_D \quad Y ; X \vdash Z$$

Display Property: for any structure occurrence Z in $X \vdash Y$, one has either $X \vdash Y \equiv_D Z \vdash W$ or $X \vdash Y \equiv_D W \vdash Z$ for some W

Belnap: If rules meet 8 conditions then cut-elimination holds!

Question: can we obtain modular interpolation from such calculi?

Some proof rules

Identity rules:

$$\frac{}{P \vdash P} \qquad \frac{X' \vdash Y' \qquad X \vdash Y \equiv_D X' \vdash Y'}{X \vdash Y}$$

Logical rules, e.g.:

$$\frac{F \,;\, G \vdash X}{F\&G \vdash X} \qquad \frac{X \vdash F \qquad Y \vdash G}{X \,;\, Y \vdash F\&G}$$

Some proof rules

Identity rules:

$$\frac{}{P \vdash P} \qquad \frac{X' \vdash Y' \qquad X \vdash Y \equiv_D X' \vdash Y'}{X \vdash Y}$$

Logical rules, e.g.:

$$\frac{F \,;\, G \vdash X}{F \& G \vdash X} \qquad \frac{X \vdash F \qquad Y \vdash G}{X \,;\, Y \vdash F \& G}$$

Structural rules, e.g.:

$$\frac{W \,;\, (X \,;\, Y) \vdash Z}{(W \,;\, X) \,;\, Y \vdash Z} \qquad \frac{\emptyset \,;\, X \vdash Y}{X \vdash Y}$$

$$\frac{X \vdash Z}{X \,;\, Y \vdash Z} \qquad \frac{X \,;\, X \vdash Y}{X \vdash Y}$$

Interpolation: our approach

- Proof-theoretic strategy: by induction on cut-free proofs; from interpolants for the premises of a rule, construct an interpolant for its conclusion.

Interpolation: our approach

- Proof-theoretic strategy: by induction on cut-free proofs; from interpolants for the premises of a rule, construct an interpolant for its conclusion.
- But not enough info to do this for display steps, e.g.:

$$\frac{X \,;\, Y \vdash Z}{X \vdash \sharp Y \,;\, Z}$$

Local AD-interpolation (LADI) property

Let $\equiv_{AD}$ be the least equivalence closed under $\equiv_D$ and applications of associativity (α) (if present).

Local AD-interpolation (LADI) property

Let $\equiv_{AD}$ be the least equivalence closed under $\equiv_D$ and applications of associativity (α) (if present).

Definition

A proof rule with conclusion $\mathcal{C}$ has the LADI property if, given that for each premise of the rule $\mathcal{C}_i$ we have interpolants for all $\mathcal{C}'_i \equiv_{AD} \mathcal{C}_i$, we can construct interpolants for all $\mathcal{C}' \equiv_{AD} \mathcal{C}$.

Local AD-interpolation (LADI) property

Let $\equiv_{AD}$ be the least equivalence closed under $\equiv_D$ and applications of associativity (α) (if present).

Definition

A proof rule with conclusion $\mathcal{C}$ has the LADI property if, given that for each premise of the rule $\mathcal{C}_i$ we have interpolants for all $\mathcal{C}'_i \equiv_{AD} \mathcal{C}_i$, we can construct interpolants for all $\mathcal{C}' \equiv_{AD} \mathcal{C}$.

Proposition

If the proof rules of a display calculus $\mathcal{D}$ all have the LADI property then $\mathcal{D}$ enjoys Craig interpolation.

Highly technical pen-and-paper proofs: so are they correct?

Interactive Proof Assistants (Isabelle)

Examples: `Mizar`, `HOL4`, `Coq`, `LEGO`, `NuPrl`, `NqThm`, `Isabelle`, `λ-Prolog`, `HOL-Light`, `LF`, `ELF`, `Twelf` $\cdots$

Meta-Logic: LCF or Kripke-Platek Set Theory or LF Type Theory or Calculus of Constructions or ...

Implementation: small core of trusted ML code

User
Object Logics
Proof Assistant
Meta-Logic (LCF)

Int Proof Develop Env
`HOL` | `IFOL` | `FOL` | `Sequents` | ...
Untrusted (ML) Code
Trusted (ML) Code
(ML) Compiler
Machine Code

Trust: rests on strong typing and small core of (ML) code which is open to public scrutiny by experts

Proof Transcripts: can be cross-checked using other assistants

Deeply embed formulae, structures, sequents and rules

HOL Formula Type: `datatype formula =`

```
    Btimes formula formula | Bplus formula formula
  | Bneg formula | Btrue ("T") | Bfalse("F")
  | FV string              (* formula variable *)
  | PP string              (* prop variable    *)
```

HOL Structure Type: `datatype structr =`

```
    Comma structr structr | Star structr | I
  | Structform formula (* cast formula into structure *)
  | SV string    (* structure variable *)
```

HOL Sequent Type: `seq` $=$ `structr` $\vdash$ `structr`

HOL Rule Type: `inf` $=$ (`seq list` , `seq`) (* *ps*/*c* *)

Pretty Printing: term `Sequent (SV ''X'') (Structform (FV ''A''))` is printed and entered as `($''X'' |- ''A'')`.

Inductively Define Set of Basic Rule Instances: `rli :: inf set`

$([\, \mathtt{X} \vdash \{\mathtt{A}\}\, ,\ \mathtt{X} \vdash \{\mathtt{B}\}],\ \mathtt{X} \vdash \{\mathtt{A\&B}\}) \in \mathtt{rli}$

Intuitions: horizontal line encoded by , and rules by set `rli`

LADI: $(\&R)$

$$\frac{X \vdash F \qquad Y \vdash G}{X\ ;\ Y \vdash F\&G}$$

Need interpolant for arbitrary $W \vdash Z \equiv_{AD} X; Y \vdash F\&G$.

LADI: $(\& R)$

$$\frac{X \vdash F \qquad Y \vdash G}{X \,;\, Y \vdash F\&G}$$

Need interpolant for arbitrary $W \vdash Z \equiv_{AD} X; Y \vdash F\&G$.

Case: $F\&G$ occurs in Z.

LADI: ($\&R$)

$$\frac{X \vdash F \qquad Y \vdash G}{X \; ; \; Y \vdash F\&G}$$

Need interpolant for arbitrary $W \vdash Z \equiv_{AD} X; Y \vdash F\&G$.

Case: $F\&G$ occurs in Z.

Subcase: W built entirely from parts of X ($W \lhd X$).

LADI: $(\& R)$

$$\frac{X \vdash F \qquad Y \vdash G}{X \,;\, Y \vdash F \& G}$$

Need interpolant for arbitrary $W \vdash Z \equiv_{AD} X; Y \vdash F\&G$.

Case: $F\&G$ occurs in Z.

Subcase: W built entirely from parts of X ($W \lhd X$).

By a LEMMA $\exists U.\ X \vdash F \equiv_{AD} W \vdash U$.

LADI: $(\&R)$

$$\frac{X \vdash F \qquad Y \vdash G}{X\,;\,Y \vdash F\&G}$$

Need interpolant for arbitrary $W \vdash Z \equiv_{AD} X; Y \vdash F\&G$.

Case: $F\&G$ occurs in Z.

Subcase: W built entirely from parts of X ($W \lhd X$).

By a LEMMA $\exists U.\ X \vdash F \equiv_{AD} W \vdash U$.

Claim: interpolant I for $W \vdash U$ is an interpolant for $W \vdash Z$.

LADI: $(\& R)$

$$\frac{X \vdash F \qquad Y \vdash G}{X\, ;\, Y \vdash F \& G}$$

Need interpolant for arbitrary $W \vdash Z \equiv_{AD} X; Y \vdash F \& G$.

Case: $F \& G$ occurs in Z.

Subcase: W built entirely from parts of X ($W \lhd X$).

By a LEMMA $\exists U.\ X \vdash F \equiv_{AD} W \vdash U$.

Claim: interpolant I for $W \vdash U$ is an interpolant for $W \vdash Z$.

Main issue: show $I \vdash Z$ provable given $I \vdash U$ provable.

LADI: $(\& R)$

By display property we have $I \vdash U \equiv_D V \vdash F$.

LADI: ($\&R$)

By display property we have $I \vdash U \equiv_D V \vdash F$.

Next, we have:

$$W \vdash Z \quad \equiv_{AD} \quad X \vdash \sharp Y ; F \& G$$

LADI: $(\& R)$

By display property we have $I \vdash U \equiv_D V \vdash F$.

Next, we have:

$$
\begin{array}{rcl}
W \vdash Z & \equiv_{AD} & X \vdash \sharp Y ; F \& G \\
& = & X \vdash F[(\sharp Y ; F \& G)/F]
\end{array}
$$

LADI: $(\&R)$

By display property we have $I \vdash U \equiv_D V \vdash F$.

Next, we have:

$$\begin{array}{rcll} W \vdash Z & \equiv_{AD} & X \vdash \sharp Y; F\&G & \\ & = & X \vdash F[(\sharp Y; F\&G)/F] & \\ & \equiv_{AD} & W \vdash U[(\sharp Y; F\&G)/F] & \qquad \text{by an easy LEMMA} \end{array}$$

LADI: $(\&R)$

By display property we have $I \vdash U \equiv_D V \vdash F$.

Next, we have:

$$\begin{array}{rll} W \vdash Z & \equiv_{AD} & X \vdash \sharp Y; F\&G \\ & = & X \vdash F[(\sharp Y; F\&G)/F] \\ & \equiv_{AD} & W \vdash U[(\sharp Y; F\&G)/F] \qquad \text{by an easy LEMMA} \end{array}$$

Thus by a substitutivity LEMMA we obtain:

$$I \vdash Z \quad \equiv_{AD} \quad I \vdash U[(\sharp Y; F\&G)/F]$$

LADI: $(\&R)$

By display property we have $I \vdash U \equiv_D V \vdash F$.

Next, we have:

$$\begin{array}{lll} W \vdash Z & \equiv_{AD} & X \vdash \sharp Y; F\&G \\ & = & X \vdash F[(\sharp Y; F\&G)/F] \\ & \equiv_{AD} & W \vdash U[(\sharp Y; F\&G)/F] \qquad \text{by an easy LEMMA} \end{array}$$

Thus by a substitutivity LEMMA we obtain:

$$\begin{array}{lll} I \vdash Z & \equiv_{AD} & I \vdash U[(\sharp Y; F\&G)/F] \\ & \equiv_{AD} & V \vdash F[(\sharp Y; F\&G)/F] \end{array}$$

LADI: $(\& R)$

By display property we have $I \vdash U \equiv_D V \vdash F$.

Next, we have:

$$
\begin{array}{rcll}
W \vdash Z & \equiv_{AD} & X \vdash \sharp Y; F\&G & \\
 & = & X \vdash F[(\sharp Y; F\&G)/F] & \\
 & \equiv_{AD} & W \vdash U[(\sharp Y; F\&G)/F] & \quad \text{by an easy LEMMA}
\end{array}
$$

Thus by a substitutivity LEMMA we obtain:

$$
\begin{array}{rcl}
I \vdash Z & \equiv_{AD} & I \vdash U[(\sharp Y; F\&G)/F] \\
 & \equiv_{AD} & V \vdash F[(\sharp Y; F\&G)/F] \\
 & \equiv_{AD} & V; Y \vdash F\&G
\end{array}
$$

Need to reason about congruent parameters

$(U, V) \in \texttt{seqrep}\ b\ X\ Y$: if b is true/false then V is obtained by replacing some (or all or none) of the succedent/antecedent part occurrences of X in U by Y $\qquad (U \ {}^{X}{\rightsquigarrow}^{Y}\ V)$

Lemma (SF_some_sub)

For formula F, structure Z, and rule set rules, if

1. *the conclusions of rules do not contain formulae; and*
2. *the conclusion of a rule in rules does not contain more than one occurrence of any structure variable; and*
3. *the rules obeys Belnap's C4 condition and*
4. *concl is derivable from prems using rules; and*
5. *$\mathit{concl}\ {}^{F}{\rightsquigarrow}^{Z}\ \mathit{sconcl}$*

then there is a list sprems (of the same length as prems) such that

1. *sconcl is derivable from sprems using rules; and*
2. *$\mathit{prem}_n\ {}^{F}{\rightsquigarrow}^{Z}\ \mathit{sprem}_n$ holds for corresponding members prem_n of prems and sprem_n of sprems.*

Deletion Lemma

Definition (`seqdel`)

Define $(C, C') \in \texttt{seqdel}\ Fs$ to mean that C' is obtained from C by deleting one occurrence in C of a structure in the set Fs.

Then we proved the following result about deletion of a formula:

Lemma (`deletion`)

Let F be a formula or $F = \emptyset$. If sequent Cd is obtained from C by deleting an occurrence of some $\#^i F$, and if $C \to^*_{AD} C'$, then either

1. *there exists Cd', such that $Cd \to^*_{AD} Cd'$, and Cd' is obtained from C' by deleting an occurrence of some $\#^j F$, or*
2. *C' is of the form $\#^n F \vdash \#^m(Z_1; Z_2)$ or $\#^m(Z_1; Z_2) \vdash \#^n F$, where $Cd \to^*_{AD} (Z_1 \vdash \# Z_2)$, or $Cd \to^*_{AD} (\# Z_1 \vdash Z_2)$*

Thus the premise is that Cd is got from C by deleting instance(s) of the substructure formula F, possibly with some $\#$ symbols.

Caveats and Lessons learned

Note: our formalisation only includes "classical" substructural logics since implication is defined in terms of disjunction

Commutativity: of conjunction and disjunction is assumed

Programmable interface: ability to interact with Isabelle 2005 using plain ML was extremely useful to program the multiple case analyses