PIE Document: WLP 2019 Examples

Revision: August 29, 2019; Rendered: May 25, 2021

Examples from the DECLARE/WLP 2019 presentation *PIE – Proving, Interpolating and Eliminating on the Basis of First-Order Logic.*

1 Abduction with the Weakest Sufficient Condition

---

kb_1

Defined as

$$\begin{array}{ll}(\mathsf{sprinkler\_was\_on} \rightarrow \mathsf{wet}(\mathsf{grass})) & \land \\ (\mathsf{rained\_last\_night} \rightarrow \mathsf{wet}(\mathsf{grass})) & \land \\ (\mathsf{wet}(\mathsf{grass}) \rightarrow \mathsf{wet}(\mathsf{shoes})). & \end{array}$$

A knowledge base.

$explanation(Kb, Na, Obs)$

Defined as

$$\forall Na\,(Kb \rightarrow Obs).$$

The weakest sufficient condition of observation Obs on the complement of Na as assumables within knowledge base Kb.

The expression $explanation(kb_1, [\mathsf{wet}], \mathsf{wet}(\mathsf{shoes}))$ expands into:

$$\begin{array}{ll}\forall p\,((\mathsf{sprinkler\_was\_on} \rightarrow p(\mathsf{grass})) \land & \\ \quad (\mathsf{rained\_last\_night} \rightarrow p(\mathsf{grass})) \land & \\ \quad (p(\mathsf{grass}) \rightarrow p(\mathsf{shoes})) & \rightarrow \\ \quad p(\mathsf{shoes})). & \end{array}$$

Second-order quantifier elimination computes the abductive explanation:

Input: $\mathit{explanation}(\mathit{kb_1}, [\mathsf{wet}], \mathsf{wet}(\mathsf{shoes}))$.
Result of elimination:

$$\mathsf{rained\_last\_night} \lor \mathsf{sprinkler\_was\_on}.$$

The following is shown by invoking a first-order prover, Prover9 by default:

This formula is valid: $\mathit{kb_1} \land \mathsf{rained\_last\_night} \rightarrow \mathsf{wet}(\mathsf{shoes})$.

2 A Simple Example of Second-Order Quantifier Elimination

Input: $\exists p\, (\forall x\, (\mathsf{q}(x) \rightarrow p(x)) \land \forall x\, (p(x) \rightarrow \mathsf{r}(x)))$.
Result of elimination:

$$\forall x\, (\mathsf{q}(x) \rightarrow \mathsf{r}(x)).$$

3 Predicate Circumscription

$\mathit{circ}(P, F)$

Defined as

$$F \land \neg\exists P'\, (F' \land T_1 \land \neg T_2),$$

where

$$\begin{array}{l} F' := F[P \mapsto P'], \\ A := \text{arity of } P \text{ in } F, \\ T_1 := \text{transfer clauses } [P/A\text{-}\mathsf{n}] \rightarrow [P'], \\ T_2 := \text{transfer clauses } [P'] \rightarrow [P/A\text{-}\mathsf{n}]. \end{array}$$

Predicate circumscription of a single predicate. The formula $\mathit{circ}(\mathsf{p}, \mathsf{p}(\mathsf{a}))$, for example, expands into:

$$\begin{array}{ll} \mathsf{p}(\mathsf{a}) & \land \\ \neg\exists q\, (q(\mathsf{a}) \land \forall x\, (q(x) \rightarrow \mathsf{p}(x)) \land \neg\forall x\, (\mathsf{p}(x) \rightarrow q(x))). & \end{array}$$

Second-order quantifier elimination can be applied to to compute that circumscription:

Input: $\mathit{circ}(\mathsf{p}, \mathsf{p}(\mathsf{a}))$.
Result of elimination:

$$\mathsf{p}(\mathsf{a}) \land \forall x\, (\mathsf{p}(x) \rightarrow x = \mathsf{a}).$$

Similarly we can compute the circumscription of wet in $\mathit{kb_1}$:

Input: $circ(\mathsf{wet}, kb_1)$.
Result of elimination:

$$\begin{array}{ll}
(\mathsf{rained_last_night} \rightarrow \mathsf{wet}(\mathsf{grass})) & \land \\
(\mathsf{sprinkler_was_on} \rightarrow \mathsf{wet}(\mathsf{grass})) & \land \\
(\mathsf{wet}(\mathsf{grass}) \rightarrow \mathsf{wet}(\mathsf{shoes})) & \land \\
\forall x\, (\mathsf{wet}(x) \rightarrow \mathsf{rained_last_night} \lor \mathsf{sprinkler_was_on}) & \land \\
\forall x\, (\mathsf{wet}(x) \land \mathsf{wet}(\mathsf{grass}) \rightarrow x = \mathsf{grass} \lor x = \mathsf{shoes}). &
\end{array}$$

4 Computing Modal Correspondences

$\Box p \rightarrow p$, known as axiom M or T, corresponds to reflexivity of the accessibility relation:
Input: $\forall p\, \forall v\, (\forall w\, (\mathsf{r}(v, w) \rightarrow p(w)) \rightarrow p(v))$.
Result of elimination:

$$\forall x\, \mathsf{r}(x, x).$$

$\Box p \rightarrow \Box\Box p$, known as axiom 4, corresponds to transitivity:
Input: $\forall p\, \forall v\, (\forall w\, (\mathsf{r}(v, w) \rightarrow p(w)) \rightarrow \forall w\, (\mathsf{r}(v, w) \rightarrow \forall w_1\, (\mathsf{r}(w, w_1) \rightarrow p(w_1))))$.
Result of elimination:

$$\forall x \forall y\, (\mathsf{r}(x, y) \rightarrow \forall z\, (\mathsf{r}(y, z) \rightarrow \mathsf{r}(x, z))).$$

5 Craig Interpolation

A simple propositional example for Craig interpolation:

---

ip_1

Defined as

$$\mathsf{p} \land \mathsf{q} \rightarrow \mathsf{p} \lor \mathsf{r}.$$

Input: ip_1.
Result of interpolation:

$$\mathsf{p}.$$

A first-order example for Craig interpolation with combined universal and existential quantification:

ip2

Defined as

$$\forall x\, \mathsf{p}(\mathsf{a}, x) \land \mathsf{q} \to \exists x\, \mathsf{p}(x, \mathsf{b}) \lor \mathsf{r}.$$

Input: *ip2*.
Result of interpolation:

$$\exists x\, \forall y\, \mathsf{p}(x, y).$$

A simple first-order example for Craig interpolation, with displayed underlying tableau:

ip3

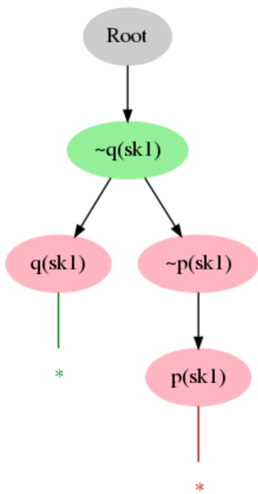
Defined as

$$\forall x\, \mathsf{p}(x) \land \forall x\, (\mathsf{p}(x) \to \mathsf{q}(x)) \to \mathsf{q}(\mathsf{c}).$$

Input: *ip3*.
Result of interpolation:

$$\forall x\, \mathsf{q}(x).$$

The clausal tableau proof underlying interpolant extraction can be visualized, colors representing the sides with respect to interpolation. The color of the closing marks indicate the side of the connection partner:



6 Definientia, Definability

definiens(G, F, P)

Defined as

$$\exists P\,(F \land G) \rightarrow \forall P\,(F \rightarrow G).$$

The interpolants of the left and right side of this implication are exactly the definientia of G in F in terms of all predicates not in P. The implication is valid if and only if definability holds.

Her is an example:

kb_2

Defined as

$$\begin{array}{ll} \forall x\,(\mathsf{p}(x) \rightarrow \mathsf{q}(x) \land \mathsf{s}(x)) & \land \\ \forall x\,(\mathsf{s}(x) \rightarrow \mathsf{r}(x)) & \land \\ \forall x\,(\mathsf{q}(x) \land \mathsf{r}(x) \rightarrow \mathsf{p}(x)). & \end{array}$$

We verify definability of p:

This formula is valid: *definiens*$(\mathsf{p}(\mathsf{a}), kb_2, [\mathsf{p}, \mathsf{s}])$.

Craig interpolation can now be applied to compute a definiens:

Input: *definiens*$(\mathsf{p}(\mathsf{a}), kb_2, [\mathsf{p}, \mathsf{s}])$.
Result of interpolation:

$$\mathsf{q}(\mathsf{a}) \land \mathsf{r}(\mathsf{a}).$$

Since a does not occur free in kb_2, it may be considered as representing a variable in a definition of p. That is, we can verify:

This formula is valid: $kb_2 \rightarrow \forall a\,(\mathsf{p}(a) \leftrightarrow \mathsf{q}(a) \land \mathsf{r}(a))$.

7 Graph Colorability

$col_2(E)$

Defined as

$$\begin{array}{ll} \exists r \exists g\,(\forall x\,(r(x) \lor g(x)) & \land \\ \quad \forall x \forall y\,(E(x,y) \rightarrow \neg(r(x) \land r(y)) \land \neg(g(x) \land g(y)))). & \end{array}$$

2-colorability as an existential second-order formula. The predicate describing the graph is exported as a parameter. Predicate parameters may be instantiated with a constant or a λ-expression. For example, $col_2(\lambda(u, v).(u = 1 \land v = 2) \lor (u = 2 \land v = 3))$ expands into:

$$\exists p \exists q\, (\forall x\, (p(x) \lor q(x)) \land \forall x \forall y\, ((x = 1 \land y = 2) \lor (x = 2 \land y = 3) \rightarrow \neg(p(x) \land p(y)) \land \neg(q(x) \land q(y)))).$$

We now perform some computations by elimination based on the inner first-order component of the above specification of 2-colorability:

$fo\_col_2(E)$

Defined as

$$\forall x\, (\mathsf{r}(x) \lor \mathsf{g}(x)) \land \forall x \forall y\, (E(x, y) \rightarrow \neg(\mathsf{r}(x) \land \mathsf{r}(y)) \land \neg(\mathsf{g}(x) \land \mathsf{g}(y))).$$

We can instantiate the graph with a predicate constant and eliminate one of the color predicates:

Input: $\exists g\, fo\_col_2(\mathsf{e})$.
Result of elimination:

$$\forall x \forall y\, (\mathsf{e}(x, y) \rightarrow \neg(\mathsf{r}(y) \land \mathsf{r}(x)) \land (\mathsf{r}(y) \lor \mathsf{r}(x))).$$

We can evaluate 2-colorability for a given graph by second-order quantifier elimination. In the current version of *PIE* this does not suceed in a single call to elimination, but works in two steps with different elimination configurations:

$ex\_rg(F)$

Defined as

$$\exists r\, F_1,$$

where

$$\mathrm{ppl\_elim(ex2([g], F), [elim\_options = [pre = [c6]], printing = false, r = F1])}.$$

Input: $ex\_rg(fo\_col_2(\lambda(u, v).(u = 1 \land v = 2) \lor (u = 2 \land v = 3)))$.
Result of elimination:

$$1 \neq 2 \land 2 \neq 3.$$

Index