Interpolant Strength Revisited

Georg Weissenbacher[1,2]

[1] Princeton University
[2] Vienna University of Technology, Austria

Craig's interpolation theorem has numerous applications in model checking, automated reasoning, and synthesis. There is a variety of interpolation systems which derive interpolants from refutation proofs; these systems are ad-hoc and rigid in the sense that they provide exactly one interpolant for a given proof. In previous work, we introduced a parametrised interpolation system which subsumes existing interpolation methods for propositional resolution proofs and enables the systematic variation of the logical strength and the elimination of non-essential variables in interpolants. In this paper, we generalise this system to propositional hyper-resolution proofs and discuss its application to proofs generated by contemporary SAT solvers. Finally, we show that, when applied to local (or split) proofs, our extension generalises two existing interpolation systems for first-order logic and relates them in logical strength.

1 Introduction

Craig interpolation [5] has proven to be an effective heuristic in applications such as model checking, where it is used as an approximate method for computing invariants of transition systems [18], and synthesis, where interpolants represent deterministic implementations of specifications given as relations [14]. The intrinsic properties of interpolants enable concise abstractions in verification and smaller circuits in synthesis. Intuitively, stronger interpolants provide more precision, and interpolants with fewer variables lead to smaller designs. However, interpolation is mostly treated as a black box, leaving no room for a systematic exploration of the solution space. In addition, the use of different interpolation systems complicates a comparison of their interpolants. We present a novel framework which generalises a number of existing interpolation techniques and supports a systematic variation and comparison of the generated interpolants.

Contributions. We present a novel *parametrised* interpolation system which extends our previous work on propositional interpolation [7].

- The extended system supports hyper-resolution (see § 3) and allows for systematic variation of the logical strength (with an additional degree of freedom over [7]) and the elimination of non-essential literals [6] in interpolants.
- We discuss (in § 4) the application of our interpolation system to hyper-resolution steps (introduced by pre-processing [10], for instance) and refutations generated by contemporary SAT solvers such as MiniSAT [8].
- When applied to local (or split) proofs [13], the extended interpolation system generalises the existing interpolation systems for first-order logic presented in [15] and [25] and relates them in logical strength (§ 5).

A. Cimatti and R. Sebastiani (Eds.): SAT 2012, LNCS 7317, pp. 312–326, 2012.

2 Background

This section introduces our notation (§ 2.1) and restates the main results of our previous paper on labelled interpolation systems [7] in § 2.2.

2.1 Formulae and Proofs

In our setting, the term *formula* refers to either a propositional logic formula or a formula in standard first-order logic.

Propositional Formulae. We work in the standard setting of propositional logic over a set X of propositional variables, the logical constants T and F (denoting true and false, respectively), and the standard logical connectives $\wedge$, $\vee$, $\Rightarrow$, and $\neg$ (denoting conjunction, disjunction, implication, and negation, respectively).

Moreover, let $\mathtt{Lit}_X = \{x, \overline{x} \,|\, x \in X\}$ be the set of literals over X, where $\overline{x}$ is short for $\neg x$. We write $\mathrm{var}(t)$ for the variable occurring in the literal $t \in \mathtt{Lit}_X$. A clause C is a set of literals. The empty clause $\square$ contains no literals and is used interchangeably with F. The disjunction of two clauses C and D is their union, denoted $C \vee D$, which is further simplified to $C \vee t$ if D is the singleton $\{t\}$. A propositional formula in Conjunctive Normal Form (CNF) is a conjunction of clauses, also represented as a set of clauses.

First-Order Logic. The logical connectives from propositional logic carry over into first-order logic. We fix an enumerable set of variables, function and predicate symbols over which formulae are built in the usual manner. The *vocabulary* of a formula A is the set of its function and predicate symbols. $\mathcal{L}(A)$ refers to the set of well-formed formulae which can be built over the vocabulary of A.

Variables may be universally ($\forall$) or existentially ($\exists$) quantified. A formula is *closed* if all its variables are quantified and *ground* if it contains no variables. As previously, conjunctions of formulae are also represented as sets.

Given a formula A in either first-order or propositional logic, we use $\mathrm{Var}(A)$ to denote the set of free (unquantified) variables in A.

Inference Rules and Proofs. We write $A_1, \cdots, A_n \models A$ to denote that the formula A holds in all models of $A_1, \ldots, A_n$ (where $n \geq 0$). An inference rule

$$\frac{A_1 \quad \cdots \quad A_n}{A} \tag{1}$$

associates zero or more *premises* (or *antecedents*) $A_1, \ldots, A_n$ with a *conclusion* A. The inference rule (1) is *sound* if $A_1, \ldots, A_n \models A$ holds. A (sound) inference system $\mathcal{I}$ is a set of (sound) inference rules.

Propositional *resolution*, for example, is a sound inference rule stating that an assignment satisfying the clauses $C \vee x$ and $D \vee \overline{x}$ also satisfies $C \vee D$:

$$\frac{C \vee x \qquad D \vee \overline{x}}{C \vee D}\ [\mathsf{Res}]$$

The clauses $C \vee x$ and $D \vee \overline{x}$ are the *antecedents*, x is the *pivot*, and $C \vee D$ is the *resolvent*. $\mathrm{Res}(C, D, x)$ denotes the resolvent of C and D with the pivot x.

Definition 1 (Proof). *A* proof *(or derivation) P in an inference system $\mathcal{I}_P$ is a directed acyclic graph $(V_P, E_P, \ell_P, \mathsf{s}_P)$, where V_P is a set of vertices, E_P is a set of edges, ℓ_P is a function mapping vertices to formulae, and $\mathsf{s}_P \in V_P$ is the sink vertex. An* initial vertex *has in-degree* 0. *All other vertices are* internal *and have in-degree ≥ 1. The sink has out-degree 0. Each internal vertex v with edges $(v_1, v), \ldots, (v_m, v) \in E_P$ is associated with an inference rule $\mathsf{Inf} \in \mathcal{I}_P$, i.e.,*

$$\frac{\ell_P(v_1) \quad \cdots \quad \ell_P(v_m)}{\ell_P(v)} \; [\mathsf{Inf}]\,.$$

The subscripts above are dropped if clear. A vertex v_i in P is a *parent* of v_j if $(v_i, v_j) \in E_P$. A proof P is a *refutation* if $\ell_P(\mathsf{s}_P) = \mathsf{F}$. Let A and B conjunctive formulae. A refutation P is an (A, B)*-refutation* of an unsatisfiable formula $A \wedge B$ if $\ell_P(v)$ is a conjunct of A or a conjunct of B for each initial vertex $v \in V_P$. A proof is *closed* (*ground*, respectively) if $\ell_P(v)$ is closed (ground) for all $v \in V_P$.

In the following, we use the propositional resolution calculus to instantiate Definition 1.

Definition 2 (Resolution Proof). *A* resolution proof *R is a proof in the inference system comprising only the resolution rule* Res. *Consequently, ℓ_R maps each vertex $v \in V_R$ to a clause, and all internal vertices have in-degree 2. Let piv_R be the function mapping internal vertices to pivot variables. For an internal vertex v and $(v_1, v), (v_2, v) \in E_R$, $\ell_R(v) = \mathrm{Res}(\ell_R(v_1), \ell_R(v_2), piv_R(v))$.*

Note that the value of ℓ_R at internal vertices is determined by that of ℓ_R at initial vertices and the pivot function piv_R. We write v^+ for the parent of v with $piv(v)$ in $\ell(v^+)$ and v^- for the parent with $\neg piv(v)$ in $\ell(v^-)$. A resolution proof R is a *resolution refutation* if $\ell_R(\mathsf{s}_R) = \square$.

2.2 Interpolation Systems and Labelling Functions

There are numerous variants and definitions of Craig's interpolation theorem [5]. We use the definition of a *Craig-Robinson interpolant* given by Harrison [11]:

Definition 3 (Interpolant). *A Craig-Robinson* interpolant *for a pair of formulae (A, B), where $A \wedge B$ is unsatisfiable, is a formula I whose free variables, function and predicate symbols occur in both A and B, such that $A \Rightarrow I$, and $B \Rightarrow \neg I$ holds.*

Craig's interpolation theorem guarantees the existence of such an interpolant for unsatisfiable pairs of formulae (A, B) in first order logic. Consequently, it also holds in the propositional setting, where the conditions of Definition 3 reduce to $A \Rightarrow I$, $B \Rightarrow \neg I$, and $\mathrm{Var}(I) \subseteq \mathrm{Var}(A) \cap \mathrm{Var}(B)$.

Numerous techniques to construct interpolants have been proposed (c.f. § 6). In particular, there is a class of algorithms that derive interpolants from proofs;

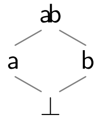
the first such algorithm for the sequent calculus is present in Maehara's constructive proof [17] of Craig's theorem. In this paper, we focus on *interpolation systems* that construct an interpolant from an (A, B)-refutation by mapping the vertices of a resolution proof to a formula called the *partial interpolant.*

Formally, an interpolation system Itp is a function that given an (A, B)-refutation R yields a function, denoted $\mathsf{Itp}(R, A, B)$, from vertices in R to formulae over $\mathrm{Var}(A) \cap \mathrm{Var}(B)$. An interpolation system is *correct* if for every (A, B)-refutation R with sink s, it holds that $\mathsf{Itp}(R, A, B)(\mathsf{s})$ is an interpolant for (A, B). We write $\mathsf{Itp}(R)$ for $\mathsf{Itp}(R, A, B)(\mathsf{s})$ when A and B are clear. Let v be a vertex in an (A, B)-refutation R. The pair $(\ell(v), \mathsf{Itp}(R, A, B)(v))$ is an *annotated clause* and is written $\ell(v)\ [\mathsf{Itp}(R, A, B)(v)]$ in accordance with [19].

In the following, we review the labelled interpolation systems we introduced in [7]. This approach generalises several existing propositional interpolation systems presented by Huang [12], Krajíček [16] and Pudlák [21], and McMillan [18]. A distinguishing feature of a labelled interpolation system is that it assigns an individual label $\mathsf{c} \in \{\perp, \mathsf{a}, \mathsf{b}, \mathsf{ab}\}$ to *each literal* in the resolution refutation.

Definition 4 (Labelling Function). *Let $(\mathcal{S}, \sqsubseteq, \sqcap, \sqcup)$ be the lattice below, where $\mathcal{S} = \{\perp, \mathsf{a}, \mathsf{b}, \mathsf{ab}\}$ is a set of symbols and $\sqsubseteq$, $\sqcap$ and $\sqcup$ are defined by the Hasse diagram to the right. A* labelling function $L_R : V_R \times \mathtt{Lit} \to \mathcal{S}$ *for a refutation R over a set of literals* $\mathtt{Lit}$ *satisfies that for all $v \in V_R$ and $t \in \mathtt{Lit}$:*

1. *$L_R(v, t) = \perp$ iff $t \notin \ell_R(v)$*
2. *$L_R(v, t) = L_R(v_1, t) \sqcup \cdots \sqcup L_R(v_m, t)$ for an internal vertex v, its parents $\{v_1, \cdots, v_m\}$, and literal $t \in \ell_R(v)$.*



Due to condition (2) above, the labels of literals at initial vertices completely determine the labelling function for literals at internal vertices. The following condition ensures that a labelling function respects the *locality* of a literal t with respect to (A, B). A literal t is A*-local* and therefore labelled a if $\mathrm{var}(t) \in \mathrm{Var}(A) \setminus \mathrm{Var}(B)$. Conversely, t is B*-local* and therefore labelled b if $\mathrm{var}(t) \in \mathrm{Var}(B) \setminus \mathrm{Var}(A)$. Literals t for which $\mathrm{var}(t) \in \mathrm{Var}(A) \cap \mathrm{Var}(B)$ are *shared* and can be labelled a, b, or ab (which generalises existing interpolation systems).

Definition 5 (Locality). *A labelling function for an (A, B)-refutation R* preserves locality *if for any initial vertex v and literal t in R*

1. $\mathsf{a} \sqsubseteq L(v, t)$ *implies that* $\mathrm{var}(t) \in \mathrm{Var}(A)$*, and*
2. $\mathsf{b} \sqsubseteq L(v, t)$ *implies that* $\mathrm{var}(t) \in \mathrm{Var}(B)$ *.*

For a given labelling function L, we define the downward *projection* of a clause at a vertex v with respect to $\mathsf{c} \in \mathcal{S}$ as $\ell(v)\lfloor_{\mathsf{c},L} \overset{\text{def}}{=} \{t \in \ell(v) \mid L(v, t) \sqsubseteq \mathsf{c}\}$. and the upward projection $\ell(v)\lceil_{\mathsf{c},L}$ as $\ell(v)\lceil_{\mathsf{c},L} \overset{\text{def}}{=} \{t \in \ell(v) \mid \mathsf{c} \sqsubseteq L(v, t)\}$. The subscript L is omitted if clear from the context.

Definition 6 (Labelled Interpolation System for Resolution). *Let L be a locality preserving labelling function for an (A, B)-refutation R. The labelled interpolation system* $\mathsf{Itp}(L)$ *maps vertices in R to partial interpolants as follows:*

For an initial vertex v with $\ell(v) = C$

$$(A\text{-clause})\ \frac{}{C \quad [C{\restriction_{\mathsf{b}}}]}\ \textit{if } C \in A \qquad\qquad (B\text{-clause})\ \frac{}{C \quad [\neg(C{\restriction_{\mathsf{a}}})]}\ \textit{if } C \in B$$

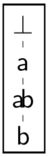
For an internal vertex v with $piv(v) = x$, $\ell(v^+) = C_1 \vee x$ and $\ell(v^-) = C_2 \vee \overline{x}$

$$\frac{C_1 \vee x \quad [I_1] \qquad C_2 \vee \overline{x} \quad [I_2]}{C_1 \vee C_2 \quad [I_3]}$$

$$\begin{array}{lll}
(A\text{-Res}) & \textit{if } L(v^+, x) \sqcup L(v^-, \overline{x}) = \mathsf{a},\ I_3 \overset{\text{def}}{=} & I_1 \vee I_2 \\
(AB\text{-Res}) & \textit{if } L(v^+, x) \sqcup L(v^-, \overline{x}) = \mathsf{ab},\ I_3 \overset{\text{def}}{=} & (x \vee I_1) \wedge (\overline{x} \vee I_2) \\
(B\text{-Res}) & \textit{if } L(v^+, x) \sqcup L(v^-, \overline{x}) = \mathsf{b},\ I_3 \overset{\text{def}}{=} & I_1 \wedge I_2
\end{array}$$

Labelling functions provide control over the interpolants constructed from a resolution proof. Firstly, labelled interpolation systems support the elimination of *non-essential* (*peripheral* [24], respectively) variables from interpolants [6]. Secondly, labelled interpolation systems – and their respective interpolants – are ordered by logical strength. A labelled interpolation system $\mathsf{Itp}(L)$ is *stronger than* $\mathsf{Itp}(L')$ if for all refutations R, $\mathsf{Itp}(L, R) \Rightarrow \mathsf{Itp}(L', R)$. The partial order $\preceq$ on labelling functions (first introduced in [7]) guarantees an ordering in strength:

Definition 7 (Strength Order). *We define the total order $\preceq$ on the lattice $\mathcal{S} = \{\bot, \mathsf{a}, \mathsf{b}, \mathsf{ab}\}$ as $\mathsf{b} \preceq \mathsf{ab} \preceq \mathsf{a} \preceq \bot$ (c.f. the Hasse diagram to the right). Let L and L' be labelling functions for an (A, B)-refutation R. The function L is stronger than L', denoted $L \preceq L'$, if for all $v \in V_R$ and $t \in \ell(v)$, $L(v, t) \preceq L'(v, t)$.*



Theorem 2 in [7] shows that if L is a stronger labelling function than L', the interpolant obtained from $\mathsf{Itp}(L)$ logically implies the one obtained from $\mathsf{Itp}(L')$.

3 Interpolation for Hyper-resolution

In this section, we extend labelled interpolation systems to a richer inference system, in particular, the inference system comprising (propositional) *hyper-resolution* [22]. Hyper-resolution is a condensation of a derivation consisting of several resolutions and avoids the construction of intermediate clauses. Hyper-resolution has several applications in propositional satisfiability checking, such as pre-processing [10] of formulae or as an integral part of the solver (e.g., [2]).

Positive hyper-resolution combines a single clause (called the *nucleus*) containing n negative literals $\overline{x}_1, \ldots, \overline{x}_n$ and n *satellite* clauses each of which contains one of the corresponding non-negated literals x_i (where $1 \leq i \leq n$):

$$\frac{\overbrace{(C_1 \vee x_1) \quad \cdots \quad (C_n \vee x_n)}^{\text{satellites}} \qquad \overbrace{(\overline{x}_1 \vee \cdots \vee \overline{x}_n \vee D)}^{\text{nucleus}}}{\bigvee_{i=1}^{n} C_i \vee D} \quad [\mathsf{HyRes}]$$

In *negative* hyper-resolution the rôles of x_i and $\overline{x}_i$ are exchanged.

Definition 8 (Hyper-resolution Proof). *A* hyper-resolution proof R *is a proof using only the inference rule* HyRes. *Accordingly,* ℓ_R *maps each vertex* $v \in V_R$ *to a clause, and all internal vertices have in-degree* ≥ 2. *Each internal vertex* v *has* $n \geq 1$ *parents* $v_1^+, \ldots, v_n^+$ *such that* $\ell_R(v_i^+) = C_i \vee x_i$ *and one parent* v^- *with* $\ell_R(v^-) = \overline{x}_1 \vee \cdots \vee \overline{x}_n \vee D$*, and consequently,* $\ell_R(v) = \bigvee_{i=1}^n C_i \vee D$.

The definition of labelling functions (Definition 4) readily applies to hyper-resolution proofs. Note that $\preceq$ is not a total order on labelling functions. Lemma 1 (a generalisation of Lemma 3 in [7] to hyper-resolution proofs) enables a comparison of labelling functions based solely on the values at the *initial* vertices.

Lemma 1. *Let* L *and* L' *be labelling functions for an* (A, B)*-refutation* R. *If* $L(v, t) \preceq L'(v, t)$ *for all initial vertices* v *and literals* $t \in \ell(v)$*, then* $L \preceq L'$.

The following definition provides a labelled interpolation system for hyper-resolution proofs.

Definition 9 (Labelled Interpolation System for Hyper-resolution). *Let* L *be a locality preserving labelling function for an* (A, B)*-refutation* R*, where* R *is a hyper-resolution proof. The labelled interpolation system* $\mathsf{Itp}(L)$ *maps vertices in* R *to partial interpolants as defined below.*[1]

For an initial vertex v *with* $\ell(v) = C$

$$(A\text{-clause})\ \frac{}{C \quad [C{\downharpoonright_{\mathsf{b}}}]}\ \textit{if } C \in A \qquad (B\text{-clause})\ \frac{}{C \quad [\neg(C{\downharpoonright_{\mathsf{a}}})]}\ \textit{if } C \in B$$

For an internal vertex v *with predecessors* $\{v_1^+, \ldots, v_n^+, v^-\}$ *(where* $n \geq 1$*) with* $\ell(v_i^+) = (C_i \vee x_i)$*, for* $1 \leq i \leq n$*, and* $\ell(v^-) = (D \vee \overline{x}_1 \vee \cdots \vee \overline{x}_n)$

$$\frac{C_1 \vee x_1 \quad [I_1] \qquad \cdots \qquad C_n \vee x_n \quad [I_n] \qquad \overline{x}_1 \vee \cdots \vee \overline{x}_n \vee D \quad [I_{n+1}]}{\bigvee_{i=1}^n C_i \vee D \quad [I]}$$

$(A\text{-HyRes})$ *if* $\forall i \in \{1..n\} \,.\, L(v_i^+, x_i) \sqcup L(v^-, \overline{x}_i) = \mathsf{a}$, $I \overset{\text{def}}{=} \bigvee_{i=1}^{n+1} I_i$

$(AB\text{-HyRes})$ *if* $\forall i \in \{1..n\} \,.\, L(v_i^+, x_i) \sqcup L(v^-, \overline{x}_i) = \mathsf{ab}$,

1.) $I \overset{\text{def}}{=} \bigwedge_{i=1}^n (x_i \vee I_i) \wedge (I_{n+1} \vee \bigvee_{i=1}^n \overline{x}_i)$, *or*

2.) $I \overset{\text{def}}{=} \bigvee_{i=1}^n (\overline{x}_i \wedge I_i) \vee (I_{n+1} \wedge \bigwedge_{i=1}^n x_i)$

$(B\text{-HyRes})$ *if* $\forall i \in \{1..n\} \,.\, L(v_i^+, x_i) \sqcup L(v^-, \overline{x}_i) = \mathsf{b}$, $I \overset{\text{def}}{=} \bigwedge_{i=1}^{n+1} I_i$

The system can be easily extended to *negative* hyper-resolution. In fact, Itp can be generalised by replacing the variables $x_1, \ldots, x_n$ in the definition with

[1] Note that unlike the interpolation system for ordinary resolution proofs presented in Definition 6, Itp is not total for hyper-resolution proofs (see discussion in § 4).

literals $t_1, \ldots, t_n$, since the proofs of our theorems below are not phase-sensitive. We avoid this generalisation to simplify the presentation.

Note that the interpolation system leaves us a choice for internal nodes AB$-$*HyRes*. We will use Itp_1 (Itp_2, respectively) to refer to the interpolation system that always chooses case 1 (case 2, respectively). Note furthermore that Definition 6 and Definition 9 are equivalent in the special case where $n = 1$.

Before we turn to the correctness of our novel interpolation system, we point out the limitation stated in Footnote 1. There are labelling functions L and proofs R for which the function $\mathsf{Itp}(L, R)$ is not total. This restriction is imposed by the case split in Definition 9 which requires the pivots of the hyper-resolution step to be uniformly labelled. We address this issue in § 4 and present a provisional conditional correctness result.

Theorem 1 (Correctness). *For any (A, B)-refutation R (where R is a hyper-resolution proof) and locality preserving labelling function L, $\mathsf{Itp}(L, R)$ (if defined) is an interpolant for (A, B).*

The proof[2] of Theorem 1 establishes that for each vertex $v \in V_R$ with $\ell_R(v) = C$ and $I = \mathsf{Itp}(L, R)(v)$, the following conditions hold:

- $A \wedge \neg(C{\restriction}_{\mathsf{a},L}) \Rightarrow I$,
- $B \wedge \neg(C{\restriction}_{\mathsf{b},L}) \Rightarrow \neg I$, and
- $\mathrm{Var}(I) \subseteq \mathrm{Var}(A) \cap \mathrm{Var}(B)$.

For $\ell_R(\mathsf{s}) = \Box$, this establishes the correctness of the system.

We emphasise that Theorem 1 does not constrain the choice for the case AB$-$*HyRes*. Since both $\mathsf{Itp}_1(L, R)$ and $\mathsf{Itp}_2(R, L)$ satisfy the conditions above, this choice does not affect the correctness of the interpolation system. In fact, it is valid to *mix* both systems by defining a choice function $\chi : V_R \to \{1, 2\}$ which determines which interpolation system is chosen at each internal node. We use $\mathsf{Itp}_\chi(L, R)$ to denote the resulting interpolation system. This modification, however, may have an impact on the logical strength of the resulting interpolant.

Theorem 2. *Let the hyper-resolution proof R be an (A, B)-refutation and L be a locality preserving labelling function. Moreover, let $\mathsf{Itp}_\chi(L, R)$ and $\mathsf{Itp}_{\chi'}(L, R)$ be labelled interpolation systems (defined for L, R) with the choice functions χ and χ', respectively. Then $\mathsf{Itp}_\chi(L, R) \Rightarrow \mathsf{Itp}_{\chi'}(L, R)$ if $\chi(v) \leq \chi'(v)$ for all internal vertices $v \in V_R$.*

Proof sketch: This follows (by structural induction over R) from

$$\left(\bigwedge_{i=1}^{n}(x_i \vee I_i) \wedge (I_{n+1} \vee \bigvee_{i=1}^{n} \overline{x}_i)\right) \Rightarrow \left(\bigvee_{i=1}^{n}(\overline{x}_i \wedge I_i) \vee (I_{n+1} \wedge \bigwedge_{i=1}^{n} x_i)\right). \quad \blacksquare$$

Note that the converse implication does not hold; a simple counterexample for an internal vertex with $n = 2$ is the assignment $x_1 = x_2 = \mathsf{F}$, $I_1 = \mathsf{T}$, and $I_2 = I_3 = \mathsf{F}$.

The final theorem in this section extends the result of Theorem 2 in [7] to hyper-resolution proofs:

[2] All proofs can be found in an extended version of the paper available from the author's website (`http://www.georg.weissenbacher.name`).

Theorem 3. *If L and L' are labelling functions for an (A, B)-refutation R (R being a hyper-resolution proof) and $L \preceq L'$ such that $\mathsf{Itp}_i(L, R)$ as well as $\mathsf{Itp}_i(L', R)$ are defined, then $\mathsf{Itp}_i(L, R) \Rightarrow \mathsf{Itp}_i(L', R)$ (for a fixed $i \in \{1, 2\}$).*

The proof of Theorem 3, is led by structural induction over R. For any vertex v in R, let I_v and I'_v be the partial interpolants due to $\mathsf{Itp}_i(L, R)$ and $\mathsf{Itp}_i(L', R)$, respectively. We show that $I_v \Rightarrow I'_v \vee \{t \in \ell_R(v) \mid L(v, t) \sqcup L'(v, t) = \mathsf{ab}\}$ for all vertices v, establishing $I_v \Rightarrow I'_v$ for the sink to show that $\mathsf{Itp}_i(L, R) \Rightarrow \mathsf{Itp}_i(L', R)$.

Theorems 2 and 3 enable us to fine-tune the strength of interpolants, since the sets of all labelling and choice functions ordered by $\preceq$ and $\leq$, respectively, form complete lattices (c.f. [7, Theorem 3]). Finally, we remark that the Theorems 2 and 3 are orthogonal. The former fixes the labelling function L, whereas the latter fixes the choice function χ.

4 Hyper-resolution and Resolution Chains

Contemporary proof-logging SAT solvers typically generate compacted proofs. MiniSAT [8], for example, discards all intermediate resolvents generated during the construction of a conflict clause and retains only resolution chains.

Definition 10 (Chain). *A (resolution)* chain *of length n is a tuple consisting of an input clause D_0 and an ordered sequence of clause-pivot pairs $\langle C_i, x_i \rangle$ (where $1 \leq i \leq n$). The final resolvent D_n of a resolution chain is defined inductively as $D_i = \mathrm{Res}(D_{i-1}, C_i, x_i)$.*

If D_0 is a nucleus and $C_1, \ldots, C_n$ are suitable satellites, the chain can be replaced by a hyper-resolution step if its conclusion D_n satisfies the HyRes rule. In general, this may not be the case: in the presence of merge literals [1], the final resolvent of a chain may depend on the order of the ordinary resolution steps. For example, the chain $(\{\overline{x}_1, x_2\}, [\langle \{\overline{x}_2\}, x_2 \rangle, \langle \{x_1, x_2\}, x_1 \rangle])$ yields the resolvent $\{x_2\}$, whereas swapping the clause-pivot pairs leads to the resolvent $\Box$. This is because the literal x_2 is re-introduced after being eliminated in the original chain, while it is *merged* and eliminated once and for all in the modified chain.

In the absence of merge literals, this issue does not arise. The following definition is a generalisation of *merge-free* edges (c.f. [7, § 5.1]) to chains.

Definition 11 (Strongly Merge-Free). *A chain*

$$(D_0, [\langle t_1 \vee C_1, \mathrm{var}(t_1) \rangle, \ldots, \langle t_n \vee C_n, \mathrm{var}(t_n) \rangle])$$

is strongly merge-free *if $\{\overline{t}_1, \cdots, \overline{t}_n\} \cap C_i = \emptyset$ for all $1 \leq i \leq n$.*

Strongly merge-free chains are insensitive to changes in the order of the resolution steps in the sense that any permutation of the clause-pivot sequence still represents a valid resolution proof (an immediate consequence of [7, Lemma 4]) with the final resolvent $(D_0 \setminus \{\overline{t}_1, \ldots, \overline{t}_n\}) \vee \bigvee_{i=1}^{n} C_i$. This property is stronger than just requiring that the sequence of resolution steps defined by a chain contains no merge literals; it demands that $\{\overline{t}_0, \ldots, \overline{t}_n\} \subseteq D_0$.[3]

Corollary 1. *Any strongly merge-free chain*

$$(\overline{x}_1 \vee \cdots \vee \overline{x}_n \vee D_0,\ [\langle x_1 \vee C_1, x_1\rangle, \ldots, \langle x_n \vee C_n, x_n\rangle])$$

corresponds to a hyper-resolution step

$$\frac{(C_1 \vee x_1) \quad \cdots \quad (C_n \vee x_n) \qquad (\overline{x}_1 \vee \cdots \vee \overline{x}_n \vee D)}{\bigvee_{i=1}^{n} C_i \vee D}\,.$$

Consequently, Definition 11 provides a sufficient (but not necessary) condition for replacing chains with hyper-resolution steps. We emphasise that Corollary 1 can be generalised by replacing the variables $x_1, \ldots, x_n$ in the respective definitions with literals $t_1, \ldots, t_n$ (c.f. § 3).

By definition, a single chain can be split into two consecutive chains, with the final resolvent of the first acting as the input clause of the second, without affecting the final result. Therefore, chains that are not merge-free can be split repeatedly until the resulting sub-sequences become strongly merge-free.

A further incentive for splitting is to enable interpolation. By splitting hyper-resolution steps whose literals are not uniformly labelled (recall the remark in § 3) we can *always* generate a labelled refutation for which ltp is a total function. The following example illustrates this transformation for a single resolution step:

$$\frac{(\overset{\mathsf{a}}{x_1} \vee C_1)\ (\overset{\mathsf{ab}}{x_2} \vee C_2)\ (\overset{\mathsf{a}}{x_3} \vee C_3)\ (\overset{\mathsf{a}}{x_4} \vee C_4) \quad (\overset{\mathsf{a}}{\overline{x}_1} \vee \overset{\mathsf{a}}{\overline{x}_2} \vee \overset{\mathsf{a}}{\overline{x}_3} \vee \overset{\mathsf{b}}{\overline{x}_4} \vee D)}{C_1 \vee C_2 \vee C_3 \vee C_4 \vee D}$$

$$\rotatebox{-90}{$\rightsquigarrow$}$$

$$\frac{(\overset{\mathsf{ab}}{x_2} \vee C_2)\ (\overset{\mathsf{a}}{x_4} \vee C_4) \qquad \dfrac{(\overset{\mathsf{a}}{x_1} \vee C_1)\ (\overset{\mathsf{a}}{x_3} \vee C_3) \quad (\overset{\mathsf{a}}{\overline{x}_1} \vee \overset{\mathsf{a}}{\overline{x}_2} \vee \overset{\mathsf{a}}{\overline{x}_3} \vee \overset{\mathsf{b}}{\overline{x}_4} \vee D)}{(\overset{\mathsf{a}}{\overline{x}_2} \vee \overset{\mathsf{b}}{\overline{x}_4} \vee C_1 \vee C_3 \vee D)}\ [A\text{-HyRes}]}{C_1 \vee C_2 \vee C_3 \vee C_4 \vee D}\ [AB\text{-HyRes}]$$

Each hyper-resolution step may need to be rewritten into at most three uniformly labelled steps (a, b, ab), thus changing the proof structure. Note that the results on the relative strength of interpolants in § 3 naturally only apply if both proofs have the same structure. The effect of the order of resolution steps on interpolants is discussed in [7, § 5.2] and exceeds the scope of this paper.

5 Local Refutations and Hyper-resolution

Jhala and McMillan demonstrate in [13, Theorem 3] that the applicability of propositional interpolation systems is not restricted to propositional logic. If a

[3] This condition, however, can be met by extending the chain with an additional resolution step $\mathrm{Res}(D_0, t \vee \overline{t} \vee T, \mathrm{var}(t))$ for any $t \in D_0$, which introduces the missing literals $T \subseteq \{\overline{t}_0, \ldots, \overline{t}_n\}$. This transformation is valid since $t \vee \overline{t} \vee T$ is a tautology.

first-order refutation R has a certain structure, namely if for each inference step in R the antecedents as well as the conclusion are either entirely in $\mathcal{L}(A)$ or in $\mathcal{L}(B)$, then one can use a propositional interpolation system (such as the ones in § 2.2 and § 3) to construct an interpolant that is a Boolean combination of the formulae in R. Kovács and Voronkov subsequently arrived at a similar result [15].

We recapitulate the results from [13,15] before we proceed to show that our interpolation system from Definition 9 generalises the system of [15] as well as a variation of [15] presented in [25].

Definition 12 (Local Refutation). *An (A, B)-refutation R in a given inference system for first-order logic is* local *if there exists a total* partitioning *function $\pi_R : V_R \to \{A, B\}$ such that for all edges $(v_1, v_2) \in E_R$ we have $\ell_R(v_1), \ell_R(v_2) \in \mathcal{L}(\pi_R(v_2))$.*

While proofs in general do *not* have this property, there is a variety of decision procedures that yield local (ground) refutations. The construction of local proofs is addressed in [13,20,9,15], to name only a few.

The following operation, which resembles the constructions in [15, Lemma 8], [13, Theorem 3], and [9, Section 5.5]), extracts a premise in $\mathcal{L}(A)$ ($\mathcal{L}(B)$, respectively) for a vertex $v \in V_R$ with $\pi(v) = A$ ($\pi(v) = B$, respectively) from a local refutation R.

Definition 13 (A-Premise, B-Premise). *Let R be a local (A, B)-refutation with partitioning function π, and let $v \in V_R$ such that $\pi(v) = A$. Then*

$$\begin{aligned} &A\text{-}premise\,(v) \overset{\text{def}}{=} \\ &\quad \{u \mid (u, v) \in E_R \text{ and } \pi(u) = B \text{ or } u \text{ is initial}\} \cup \\ &\quad \bigcup \{A\text{-}premise\,(u) \mid (u, v) \in E_R \text{ and } \pi(u) = A\}\,. \end{aligned}$$

B-premise(v) is defined analogously.

Intuitively, A-premise(v) comprises the leaves of the largest sub-derivation S rooted at v such that $\pi(u) = A$ for all internal vertices $u \in V_S$.[4] If the underlying inference system is sound, we have $\{\ell(u) \mid u \in A\text{-premise}(v)\} \models \ell(v)$. If, moreover, $\ell(v)$ as well as all formulae of A-premise(v) are *closed*, we make the following observation (c.f. related results in [15, Lemma 1] and [9, Lemma 3]):

Corollary 2. *Let R be a local closed refutation in a sound inference system, and let $v \in V_R$ an internal vertex such that $\pi_R(v) = A$. Then, the following Horn clause is a tautology:*

$$\bigvee_{u \in A\text{-}premise(v)} \neg \ell_R(u) \vee \ell_R(v) \tag{2}$$

A similar claim holds for the case in which $\pi(v) = B$.

[4] In particular, it is possible to choose π_R in such a manner that S is the largest sub-derivation rooted at v in R such that $\ell_R(u) \in \mathcal{L}(A)$ for all $u \in V_S$. This corresponds to the setting in [15, Lemma 8].

Corollary 2 is a pivotal element in our proof of the following theorem:

Theorem 4. *(c.f. [13, Theorem 3]) Let R be a closed local (A, B)-refutation in a sound inference system. Then one can extract a Craig-Robinson interpolant from R using a propositional interpolation system.*

Proof: Let $v \in V_R$ be such that $\pi(v) = A$. If v is initial, then either A or B contains the unit clause $C_v = \ell(v)$. Otherwise, according to Corollary 2, the clause $C_v = (\{\neg\ell(u) \,|\, u \in A\text{-premise}(v)\} \vee \ell(v))$ is tautological (and therefore implied by A). Moreover, it follows from Definition 12 that if $u \in A$-premise(v) is not an initial vertex of R then $\ell_R(u) \in \mathcal{L}(A) \cap \mathcal{L}(B)$ holds. Accordingly, $C_v \in \mathcal{L}(A)$, and we add C_v to A. A similar argument holds for $v \in V_R$ with $\pi(v) = B$.

By construction, the resulting set of clauses C_v, $v \in V_R$, is propositionally unsatisfiable [13,15]; also, each clause is implied by either A or B. Moreover, all literals with $t \in \mathcal{L}(A) \setminus \mathcal{L}(B)$ ($t \in \mathcal{L}(B) \setminus \mathcal{L}(A)$, respectively) are local to A (B, respectively). Accordingly, it is possible to construct an interpolant for (A, B) using the interpolation systems presented in § 2.2 and § 3. ■

Kovács and Voronkov avoid the explicit construction of a resolution proof by defining their interpolation system directly on the local proof [15, Theorem 11]:

Definition 14. *Let R be a local and closed (A, B)-refutation. The interpolation system Itp_{KV} maps vertices $v \in V_R$ for which $\ell_R(v) \in \mathcal{L}(A) \cap \mathcal{L}(B)$ holds to partial interpolants as defined below.*

For an initial vertex v

$$(A\text{-clause})\ \frac{}{\ell(v) \quad [\ell(v)]}\ \textit{if } \ell(v) \in A \qquad (B\text{-clause})\ \frac{}{\ell(v) \quad [\neg\ell(v)]}\ \textit{if } \ell(v) \in B$$

For an internal vertex v with $\{v_1, \ldots, v_n\} = \pi(v)$-premise$(v)$ such that

$$\ell(v_i) \in \mathcal{L}(A) \cap \mathcal{L}(B) \textit{ for } 1 \leq i \leq m \leq n \textit{ and}$$
$$\ell(v_j) \notin \mathcal{L}(A) \cap \mathcal{L}(B) \textit{ for } m < j \leq n\,.$$

$$\frac{\ell(v_1) \quad [I_1] \quad \cdots \quad \ell(v_m) \quad [I_m] \qquad \ell(v_{m+1}) \quad \cdots \quad \ell(v_n)}{\ell(v) \quad [I]}$$

$$(A\text{-justified})\ \textit{if } \pi(v) = A,\ I \stackrel{\text{def}}{=} \bigwedge_{i=1}^{m}(\ell(v_i) \vee I_i) \wedge \bigvee_{i=1}^{m} \neg\ell(v_i)$$
$$(B\text{-justified})\ \textit{if } \pi(v) = B,\ I \stackrel{\text{def}}{=} \bigwedge_{i=1}^{m}(\ell(v_i) \vee I_i)$$

Remark. In addition to the condition in Definition 12, Kovács and Voronkov require that for each $v \in V_R$ with predecessors $v_1, \ldots, v_n$, $\ell(v) \in \mathcal{L}(A) \cap \mathcal{L}(B)$ if $\ell(v_i) \in \mathcal{L}(A) \cap \mathcal{L}(B)$ for all $i \in \{1..n\}$. A local derivation satisfying this condition is *symbol-eliminating*, i.e., it does not introduce "irrelevant" symbols. This technical detail allows the leaves of R to be merely implied by A (or B) instead of being actual elements of A (B, respectively), while preserving the correctness of the interpolation system. This effectively enables interpolation for *non-closed* formulae (A, B).

We proceed to show one of the main results of this paper, namely that our interpolation system Itp from Definition 9 is able to simulate the interpolation system Itp_{KV}.

Theorem 5. *Let R be a local and closed (A, B)-refutation. Then we can construct a hyper-resolution refutation H of (A, B) and a locality preserving labelling function L such that for each $v \in V_R$ with $\ell_R(v) \in \mathcal{L}(A) \cap \mathcal{L}(B)$ there exists a corresponding vertex $u \in V_H$ such that $\mathsf{Itp}_{KV}(R)(v) \Leftrightarrow \mathsf{Itp}_1(L, H)(u)$.*

Proof sketch: We demonstrate that it is possible to construct a hyper-resolution refutation H of (A, B) in which each internal step of Itp_{KV} is simulated using *two* hyper-resolution steps. The induction hypothesis is that for each internal vertex $v \in V_R$ with $\{v_1, \ldots, v_n\} = \pi(v)$-premise$(v)$ and m as in Definition 14, we have vertices $\{u_1, \ldots, u_n\} \subseteq V_H$ such that

1. $\ell_H(u_i) = \ell_R(v_i)$ for $1 \le i \le n$, and
2. $\mathsf{Itp}_1(L, H)(u_i) \Leftrightarrow \mathsf{Itp}_{KV}(R)(v_i)$ for $1 \le i \le m$, and
3. $\mathsf{Itp}_1(L, H)(u_j) = \begin{cases} \mathsf{F} \text{ if } \ell(v_j) \in A \\ \mathsf{T} \text{ if } \ell(v_j) \in B \end{cases}$ for $m < j \le n$.

We add an auxiliary vertex labelled with the clause $\neg\ell_H(u_1) \vee \cdots \vee \neg\ell_H(u_n) \vee \ell_R(v)$, which, by Corollary 2 and by Definition 12, can be regarded as element of formula $\pi(v)$ (see proof of Theorem 4). The first hyper-resolution step eliminates the literals local to $\pi(v)$; the interpolants and labels are indicated for $\pi(v) = A$:

$$\frac{\ell_H(\overset{\mathsf{a}}{u_{m+1}})\ [\mathsf{F}]\ \cdots\ \ell_H(\overset{\mathsf{a}}{u_n})\ [\mathsf{F}] \qquad (\neg\ell_H(\overset{\mathsf{a}}{u_{m+1}}) \vee \cdots \vee \neg\ell_H(\overset{\mathsf{a}}{u_n}) \vee \cdots \vee \ell_R(\overset{\mathsf{a}}{v}))\,[\mathsf{F}]}{(\neg\ell_H(\overset{\mathsf{ab}}{u_1}) \vee \cdots \vee \neg\ell_H(\overset{\mathsf{ab}}{u_m}) \vee \ell_R(\overset{\mathsf{a}}{v})) \quad [\mathsf{F}]}$$

The second hyper-resolution step eliminates the shared literals $\ell_H(u_i)$ (for $1 \le i \le m$). Again, the labels and interpolants are for the case that $\pi(v) = A$:

$$\frac{\ell_H(u_1)\,[I_1]\ \cdots\ \ell_H(u_m)\,[I_m] \qquad (\neg\ell_H(\overset{\mathsf{ab}}{u_1}) \vee \cdots \vee \neg\ell_H(\overset{\mathsf{ab}}{u_m}) \vee \ell_R(\overset{\mathsf{a}}{v}))\ [\mathsf{F}]}{\ell_R(\overset{\mathsf{a}}{v}) \quad [\bigwedge_{i=1}^{m}(\ell_H(u_i) \vee I_i]) \wedge (\mathsf{F} \vee \bigvee_{i=1}^{m} \neg\ell_H(u_i))]}$$

The sink of this resolution step is the vertex $u \in V_H$ such that $\ell_H(u) = \ell_R(v)$ and $\mathsf{Itp}_1(L, H)(u) = \mathsf{Itp}_{KV}(v)$. ■

We proceed to show that our system for hyper-resolution also generalises another existing interpolation system for local refutations. In [25], we introduced the following variation of the interpolation system in Definition 14:

Definition 15. *Let Itp_W be the interpolation system as described in Definition 14, except for the following modification:*

$$\boxed{\begin{array}{l} (A\text{-justified})\ \textit{if}\ \pi(v) = A,\ I \overset{\text{def}}{=} \bigvee_{i=1}^{m}(\neg\ell(v_i) \wedge I_i) \\ (B\text{-justified})\ \textit{if}\ \pi(v) = B,\ I \overset{\text{def}}{=} \bigvee_{i=1}^{m}(\neg\ell(v_i) \wedge I_i) \vee \bigwedge_{i=1}^{m} \ell(v_i) \end{array}}$$

The following theorem states that the interpolation system in Definition 9 is powerful enough to simulate Itp_W.

Theorem 6. *Let R be a local and closed (A, B)-refutation. Then we can construct a hyper-resolution refutation H of (A, B) and a locality preserving labelling function L such that for each $v \in V_R$ with $\ell_R(v) \in \mathcal{L}(A) \cap \mathcal{L}(B)$ there exists a corresponding vertex $u \in V_H$ such that $\mathsf{Itp}_W(R)(v) \Leftrightarrow \mathsf{Itp}_2(L, H)(u)$.*

The proof is essentially equivalent to the proof of Theorem 5. Moreover, as a consequence of Theorem 2, Itp_{KV} is *stronger* than Itp_W.

Corollary 3. *Let R be a closed local (A, B)-refutation in a sound inference system. Then $\mathsf{Itp}_{KV}(R) \Rightarrow \mathsf{Itp}_W(R)$.*

6 Related Work

There is a vastly growing number of different interpolation techniques; a recent survey of interpolation in decision procedures is provided by [3]. An exposition of interpolation techniques for SMT solvers can be found in [4]. The work of Yorsh and Musuvathi [26] enables the combination of theory-specific and propositional interpolation techniques [12,16,21,18,7].

The novel interpolation system presented in Section 3 extends our prior work on propositional interpolation systems [7]. The idea of using labelling functions (initially introduced in [24] in the context of LTL vacuity detection to determine the *peripherality* of variables in resolution proofs) is common to both approaches.

A number of interpolation techniques provide local proofs (e.g., [13,20,9,15]). Not all interpolation techniques are based on local proofs, though: McMillan's interpolating inference system for equality logic with uninterpreted functions and linear arithmetic [19], for instance, performs an implicit conversion of the proof, and the approach presented in [23] avoids the construction of proofs altogether.

7 Consequences and Conclusion

We present a novel interpolation system for hyper-resolution proofs which generalises our previous work [7]. By applying our technique to local proofs, we combine a number of first-order [15,25] and propositional interpolation techniques [12,16,21,18] into one *uniform* interpolation approach. As in [13], our approach avoids an explicit theory combination step [26]. Therefore, it enables the variation of interpolant strength and the elimination of non-essential literals across the theory boundary. Finally, by defining a rule that addresses hyper-resolution steps (introduced by pre-processing or extracted from resolution chains), we avoid the construction of intermediate partial interpolants. An experimental evaluation of the benefit on overhead and interpolant size is future work.

References

1. Andrews, P.B.: Resolution with merging. J. ACM 15, 367–381 (1968)
2. Bacchus, F.: Enhancing Davis Putnam with extended binary clause reasoning. In: IAAI, pp. 613–619. AAAI Press / MIT Press (2002)
3. Bonacina, M.P., Johansson, M.: On Interpolation in Decision Procedures. In: Brünnler, K., Metcalfe, G. (eds.) TABLEAUX 2011. LNCS, vol. 6793, pp. 1–16. Springer, Heidelberg (2011)
4. Cimatti, A., Griggio, A., Sebastiani, R.: Efficient generation of Craig interpolants in satisfiability modulo theories. In: TOCL (2010)
5. Craig, W.: Linear reasoning. A new form of the Herbrand-Gentzen theorem. J. Symbolic Logic 22, 250–268 (1957)
6. D'Silva, V.: Propositional Interpolation and Abstract Interpretation. In: Gordon, A.D. (ed.) ESOP 2010. LNCS, vol. 6012, pp. 185–204. Springer, Heidelberg (2010)
7. D'Silva, V., Kroening, D., Purandare, M., Weissenbacher, G.: Interpolant Strength. In: Barthe, G., Hermenegildo, M. (eds.) VMCAI 2010. LNCS, vol. 5944, pp. 129–145. Springer, Heidelberg (2010)
8. Eén, N., Sörensson, N.: An Extensible SAT-solver. In: Giunchiglia, E., Tacchella, A. (eds.) SAT 2003. LNCS, vol. 2919, pp. 502–518. Springer, Heidelberg (2004)
9. Fuchs, A., Goel, A., Grundy, J., Krstić, S., Tinelli, C.: Ground Interpolation for the Theory of Equality. In: Kowalewski, S., Philippou, A. (eds.) TACAS 2009. LNCS, vol. 5505, pp. 413–427. Springer, Heidelberg (2009)
10. Gershman, R., Strichman, O.: Cost-Effective Hyper-Resolution for Preprocessing CNF Formulas. In: Bacchus, F., Walsh, T. (eds.) SAT 2005. LNCS, vol. 3569, pp. 423–429. Springer, Heidelberg (2005)
11. Harrison, J.: Handbook of Practical Logic and Automated Reasoning. Cambridge University Press (2009)
12. Huang, G.: Constructing Craig Interpolation Formulas. In: Li, M., Du, D.-Z. (eds.) COCOON 1995. LNCS, vol. 959, pp. 181–190. Springer, Heidelberg (1995)
13. Jhala, R., McMillan, K.L.: A Practical and Complete Approach to Predicate Refinement. In: Hermanns, H. (ed.) TACAS 2006. LNCS, vol. 3920, pp. 459–473. Springer, Heidelberg (2006)
14. Jiang, J.-H.R., Lin, H.-P., Hung, W.-L.: Interpolating functions from large Boolean relations. In: ICCAD, pp. 779–784. ACM (2009)
15. Kovács, L., Voronkov, A.: Interpolation and Symbol Elimination. In: Schmidt, R.A. (ed.) CADE 2009. LNCS, vol. 5663, pp. 199–213. Springer, Heidelberg (2009)
16. Krajíček, J.: Interpolation theorems, lower bounds for proof systems, and independence results for bounded arithmetic. J. Symbolic Logic 62, 457–486 (1997)
17. Maehara, S.: On the interpolation theorem of Craig. Sûgaku 12, 235–237 (1961)
18. McMillan, K.L.: Interpolation and SAT-Based Model Checking. In: Hunt Jr., W.A., Somenzi, F. (eds.) CAV 2003. LNCS, vol. 2725, pp. 1–13. Springer, Heidelberg (2003)
19. McMillan, K.L.: An interpolating theorem prover. TCS 345(1), 101–121 (2005)
20. McMillan, K.L.: Quantified Invariant Generation Using an Interpolating Saturation Prover. In: Ramakrishnan, C.R., Rehof, J. (eds.) TACAS 2008. LNCS, vol. 4963, pp. 413–427. Springer, Heidelberg (2008)
21. Pudlák, P.: Lower bounds for resolution and cutting plane proofs and monotone computations. J. Symbolic Logic 62, 981–998 (1997)
22. Robinson, J.: Automatic deduction with hyper-resolution. J. Comp. Math. 1 (1965)

23. Rybalchenko, A., Sofronie-Stokkermans, V.: Constraint Solving for Interpolation. In: Cook, B., Podelski, A. (eds.) VMCAI 2007. LNCS, vol. 4349, pp. 346–362. Springer, Heidelberg (2007)
24. Simmonds, J., Davies, J., Gurfinkel, A., Chechik, M.: Exploiting resolution proofs to speed up LTL vacuity detection for BMC. STTT 12, 319–335 (2010)
25. Weissenbacher, G.: Program Analysis with Interpolants. PhD thesis, Oxford (2010)
26. Yorsh, G., Musuvathi, M.: A Combination Method for Generating Interpolants. In: Nieuwenhuis, R. (ed.) CADE 2005. LNCS (LNAI), vol. 3632, pp. 353–368. Springer, Heidelberg (2005)