Range-Restricted and Horn Interpolation through Clausal Tableaux

Christoph Wernhard

University of Potsdam

TABLEAUX 2023

Prague, Czech Republic, Sep 18–21, 2023

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project-ID 457292495.
The work was supported by the North-German Supercomputing Alliance (HLRN).

1. Query Reformulation as Craig Interpolation
2. Craig Interpolation with Clausal Tableaux
3. Considered Notions of Range-Restriction
4. Theorems on Interpolation, Range-Restriction and the Horn Property
5. On Proving the Theorems – The Hyper Property
6. Conclusion

1. Query Reformulation as Craig Interpolation

2. Craig Interpolation with Clausal Tableaux

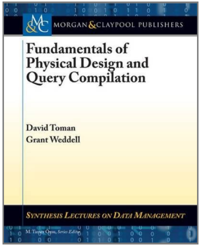
3. Considered Notions of Range-Restriction

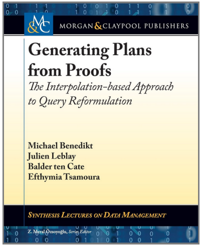
4. Theorems on Interpolation, Range-Restriction and the Horn Property

5. On Proving the Theorems – The Hyper Property

6. Conclusion

Query Refomulation as Craig Interpolation





[Craig 1957]
[Nash, Segoufin, Vianu 2005, 2010]
[Toman, Wedell 2011]
[Benedikt et al. 2016]

$$K \vDash \forall x\,(Qx \leftrightarrow Rx)$$

$$K \vDash Qx \rightarrow Rx \qquad\qquad K \vDash Rx \rightarrow Qx$$

$$K \wedge Qx \quad \vDash Rx \vDash \quad \neg K' \vee Q'x$$

Desirable Properties of Database Queries

$$K \models \forall x\,(Qx \leftrightarrow Rx)$$

$$K \land Qx \;\models Rx \models\; \neg K' \lor Q'x$$

- In DB/KR applications R should have **desirable properties**, in dependency of properties of K and Q
- In particular, query formulas should be “evaluable” – captured by domain independence
- Domain independence is undecidable, but there are various **syntactic restrictions** to ensure it

	Query	Domain independent
1	$\{x \mid \neg \mathsf{p}(x)\}$	
2	$\{x \mid \mathsf{p}(x) \land \neg \mathsf{q}(x)\}$	✓
3	$\{\langle x, y\rangle \mid \mathsf{p}(x) \lor \mathsf{q}(y)\}$	
4	$\{x \mid \mathsf{p}(x) \lor \exists y\, \mathsf{q}(x, y)\}$	✓

Range-Restricted and Horn Interpolation through Clausal Tableaux

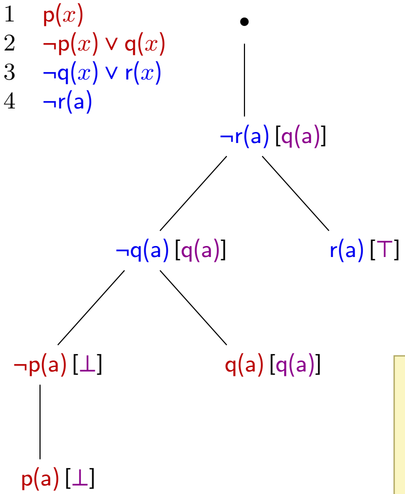
1. Query Reformulation as Craig Interpolation
2. **Craig Interpolation with Clausal Tableaux**
3. Considered Notions of Range-Restriction
4. Theorems on Interpolation, Range-Restriction and the Horn Property
5. On Proving the Theorems – The Hyper Property
6. Conclusion

- **A framework from fully automated first-order proving**
- Systems
  - Prolog Technology Theorem Prover [Stickel 1988]
  - SETHEO [Letz, Bibel et al. 1992]
  - CMProver [CW 1992]
  - leanCoP [Otten, Bibel 2003]
- Methodology
  - Connection method [Bibel 1982]
  - Model elimination [Loveland 1978]
  - **Clausal tableaux [Letz 1999]**
- Permits **Craig interpolation [CW JAR 2021]**

$\forall x\, \mathsf{p}(x) \land \forall x\, (\neg \mathsf{p}(x) \lor \mathsf{q}(x)) \;\models\; \forall x\, (\neg \mathsf{q}(x) \lor \mathsf{r}(x)) \rightarrow \mathsf{r}(\mathsf{a})$

1	$\mathsf{p}(x)$
2	$\neg \mathsf{p}(x) \lor \mathsf{q}(x)$
3	$\neg \mathsf{q}(x) \lor \mathsf{r}(x)$
4	$\neg \mathsf{r}(\mathsf{a})$

$\forall x\, \mathsf{p}(x) \land \forall x\, (\neg\mathsf{p}(x) \lor \mathsf{q}(x)) \models \forall x\, \mathsf{q}(x) \models \forall x\, (\neg\mathsf{q}(x) \lor \mathsf{r}(x)) \rightarrow \mathsf{r}(\mathsf{a})$



Procedure CTIF, a 2-Stage Interpolation Method

Input: First-order formulas F and G s.th. $F \models G$

Output: A Craig-Lyndon interpolant H of F and G

1. Free variables to placeholder constants
2. Skolemization and clausification of F and $\neg G$
3. Tableau computation by a prover
4. Tableau grounding
   Heuristics: choice of terms for grounding
5. Side assignment of the tableau clauses
   Heuristics: if a clause is from both F and $\neg G$
6. **"Stage 1" Ground interpolant extraction**

th variables

G

tifier order

les

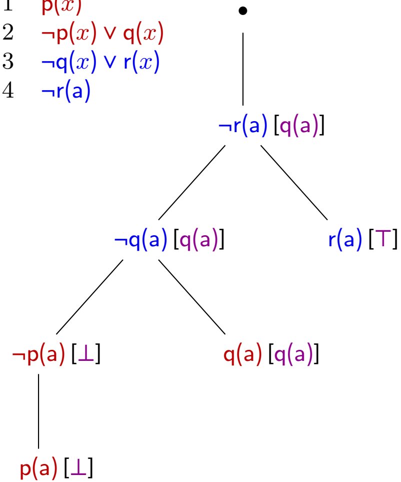
side(N)	side(tgt(N))	ipol(N)
F	F	$\bot$
F	G	lit(N)
G	F	$\overline{\text{lit}(N)}$
G	G	$\top$

side(N_1)	ipol(N)
F	$\bigvee_{i=1}^{n} \text{ipol}(N_i)$
G	$\bigwedge_{i=1}^{n} \text{ipol}(N_i)$

Craig Interpolation with Clausal Tableaux [CW JAR 2021]

$\forall x\, \mathsf{p}(x) \land \forall x\, (\neg \mathsf{p}(x) \lor \mathsf{q}(x)) \models \forall x\, \mathsf{q}(x) \models \forall x\, (\neg \mathsf{q}(x) \lor \mathsf{r}(x)) \rightarrow \mathsf{r}(\mathsf{a})$

1 $\mathsf{p}(x)$
2 $\neg \mathsf{p}(x) \lor \mathsf{q}(x)$
3 $\neg \mathsf{q}(x) \lor \mathsf{r}(x)$
4 $\neg \mathsf{r}(\mathsf{a})$



Procedure CTIF, a 2-Stage Interpolation Method
Input: First-order formulas F and G s.th. $F \models G$
Output: A Craig-Lyndon interpolant H of F and G

1. Free variables to placeholder constants
2. Skolemization and clausification of F and $\neg G$
3. Tableau computation by a prover
4. Tableau grounding
   Heuristics: choice of terms for grounding
5. Side assignment of the tableau clauses
   Heuristics: if a clause is from both F and $\neg G$
6. **"Stage 1" Ground interpolant extraction**
7. **"Stage 2" Lifting**: replacing terms with variables and adding a quantifier prefix
   Roughly: $\exists$ if term from F, $\forall$ if from G
   Heuristics: linearizing the partial quantifier order
8. Placeholder constants to free variables

1. Query Reformulation as Craig Interpolation
2. Craig Interpolation with Clausal Tableaux
3. Considered Notions of Range-Restriction
4. Theorems on Interpolation, Range-Restriction and the Horn Property
5. On Proving the Theorems – The Hyper Property
6. Conclusion

Definition

A formula $F(\mathcal{X})$ is ***VGT-range-restricted (VGT-RR)*** if cnf(F) = $Q\ M_{\mathrm{C}}$ and dnf(F) = $Q\ M_{\mathrm{D}}$, where

- Q is a quantifier prefix upon universal variables $\mathcal{U}$ and existential variables $\mathcal{E}$
- M_{C} is a CNF matrix
- M_{D} is a DNF matrix

such that

1. For all clauses C in M_{C} it holds that $\mathcal{V}ar(C) \cap \mathcal{U} \subseteq \mathcal{V}ar^{-}(C)$.
2. For all conjunctive clauses D in M_{D} it holds that $\mathcal{V}ar(D) \cap \mathcal{E} \subseteq \mathcal{V}ar^{+}(D)$.
3. For all conjunctive clauses D in M_{D} it holds that $\mathcal{X} \subseteq \mathcal{V}ar^{+}(D)$.

Example

Does some supplier supply all parts required for project a?

Let $F = \exists x \forall y\, (\neg \mathsf{r}(\mathsf{a}, y) \lor \mathsf{s}(x, y))$

$$\begin{array}{lcll} \mathrm{cnf}(F) & = & \exists x \forall y & \neg \mathsf{r}(\mathsf{a}, y) \lor \mathsf{s}(x, y) \\ \mathrm{dnf}(F) & = & \exists x \forall y & \neg \mathsf{r}(\mathsf{a}, y) \\ & & & \mathsf{s}(x, y) \end{array}$$

Example

Let $F = \exists x\, [(\mathsf{p}(x, y) \lor \mathsf{q}(y)) \land \neg \mathsf{r}(y)]$

$$\begin{array}{lcll} \mathrm{cnf}(F) & = & \exists x & \mathsf{p}(x, y) \lor \mathsf{q}(y) \\ & & & \neg \mathsf{r}(y) \\ \mathrm{dnf}(F) & = & \exists x & \mathsf{p}(x, y) \land \neg \mathsf{r}(y) \\ & & & \mathsf{q}(y) \land \neg \mathsf{r}(y) \end{array}$$

Note: $F \equiv (\exists x\, \mathsf{p}(x, y) \lor \mathsf{q}(y)) \land \neg \mathsf{r}(y)$

”Universal” Range-Restriction

Definition

A formula $F(\mathcal{X})$ is ***U-range-restricted (U-RR)*** if cnf$(F) = Q\, M_\mathrm{C}$ and ~~dnf$(F) = Q\, M_\mathrm{D}$~~, where

- Q is a quantifier prefix upon universal variables $\mathcal{U}$ and existential variables $\mathcal{E}$
- M_C is a CNF matrix
- ~~M_D is a DNF matrix~~

such that

1. For all clauses C in M_C it holds that $\mathcal{V}ar(C) \cap \mathcal{U} \subseteq \mathcal{V}ar^-(C)$.
2. ~~For all conjunctive clauses D in M_D it holds that $\mathcal{V}ar(D) \cap \mathcal{E} \subseteq \mathcal{V}ar^+(D)$.~~
3. ~~For all conjunctive clauses D in M_D it holds that $\mathcal{X} \subseteq \mathcal{V}ar^+(D)$.~~

If F is a sentence, then

- F is VGT-RR iff F and $\neg F$ are both U-RR
- If F is universal then F is VGT-RR iff F is U-RR
- If F is existential then F is VGT-RR iff $\neg F$ is U-RR

1. Query Reformulation as Craig Interpolation

2. Craig Interpolation with Clausal Tableaux

3. Considered Notions of Range-Restriction

4. Theorems on Interpolation, Range-Restriction and the Horn Property

5. On Proving the Theorems – The Hyper Property

6. Conclusion

Theorem (Interpolation and Range-Restriction)
Assume $F \models G$. If $F, \neg G$ satisfy the conditions specified in the table, then there exists a Craig-Lyndon interpolant H of F and G with the property given in the table.

Moreover, H can be effectively constructed from a clausal tableau proof of $F \models G$.

F	$\neg G$	H
U-RR		U-RR
U-RR, $\mathcal{V}ar(F) = \emptyset$	U-RR, $\mathcal{V}ar(\neg G) = \emptyset$	VGT-RR
U-RR, $\mathcal{V}ar(F) = \mathcal{X}$, $*$	U-RR, $\mathcal{V}ar(\neg G) = \mathcal{X}$, $*$	VGT-RR

$*$ Fineprint for case with free variables $\mathcal{X}$ in both F and $\neg G$:

1. No negative clause in cnf(F)
2. For all negative clauses C in cnf$(\neg G)$ it holds that $\mathcal{X} \subseteq \mathcal{V}ar^{-}(C)$
3. For all clauses C in cnf$(\neg G)$ it holds that $\mathcal{V}ar(C) \cap \mathcal{X} \subseteq \mathcal{V}ar^{-}(C)$

- Recall that $K \models \forall x\, (Qx \leftrightarrow Rx)$ iff $K \land Qx \models Rx \models \neg K' \lor Q'x$

Assuming K is a sentence, our theorem instantiates to

$K \land Qx$	$K \land \neg Qx$	Rx
U-RR		U-RR
U-RR, Q Boolean	U-RR, Q Boolean	VGT-RR
U-RR, $*$	U-RR, $*$	VGT-RR

$*$ Fineprint for non-Boolean Qx

1. No negative clause in cnf$(K \land Qx)$
2. For all negative clauses C in cnf$(\neg Qx)$ it holds that $x \in \mathcal{V}ar^{-}(C)$
3. For all clauses C in cnf$(K' \land \neg Q'x)$ it holds that if $x \in \mathcal{V}ar(C)$, then $x \in \mathcal{V}ar^{-}(C)$

Theorem (Horn Interpolation)
Assume $F \vDash G$. If $F, \neg G$ satisfy the conditions specified in the table, then there exists a Craig-Lyndon interpolant H of F and G with the properties given in the table.

Moreover, H can be effectively constructed from a clausal tableau proof of $F \vDash G$.

F	$\neg G$	H
Horn		**Horn**
U-RR, **Horn**		U-RR, **Horn**
U-RR, $\mathcal{V}ar(F) = \varnothing$, **Horn**	U-RR, $\mathcal{V}ar(\neg G) = \varnothing$	VGT-RR, **Horn**
U-RR, $\mathcal{V}ar(F) = \mathcal{X}$, $*$, **Horn**	U-RR, $\mathcal{V}ar(\neg G) = \mathcal{X}$, $*$	VGT-RR, **Horn**

$*$ Fineprint for case with free variables $\mathcal{X}$ in both F and $\neg G$:

1. No negative clause in cnf(F)
2. For all negative clauses C in cnf$(\neg G)$ it holds that $\mathcal{X} \subseteq \mathcal{V}ar^{-}(C)$
3. For all clauses C in cnf$(\neg G)$ it holds that $\mathcal{V}ar(C) \cap \mathcal{X} \subseteq \mathcal{V}ar^{-}(C)$

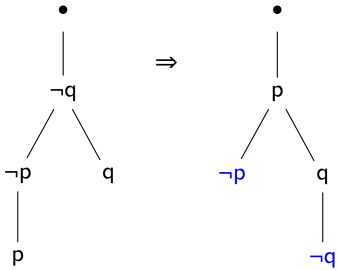
- Both the **cases for range-restriction and Horn are proven by induction on the same proof structures**

1. Query Reformulation as Craig Interpolation
2. Craig Interpolation with Clausal Tableaux
3. Considered Notions of Range-Restriction
4. Theorems on Interpolation, Range-Restriction and the Horn Property
5. On Proving the Theorems – The Hyper Property
6. Conclusion

The Hyper Property

Definition
A clausal tableau is called ***hyper*** if the nodes labeled with a negative literal are exactly the leaf nodes.

- On a clausal tableau with the hyper property, the **CTIF procedure computes interpolants according to our theorems**
- Arbitrary clausal tableaux **can be converted** such that they get the hyper property
- Also **resolution deduction trees** can be converted to clausal tableaux that are hyper

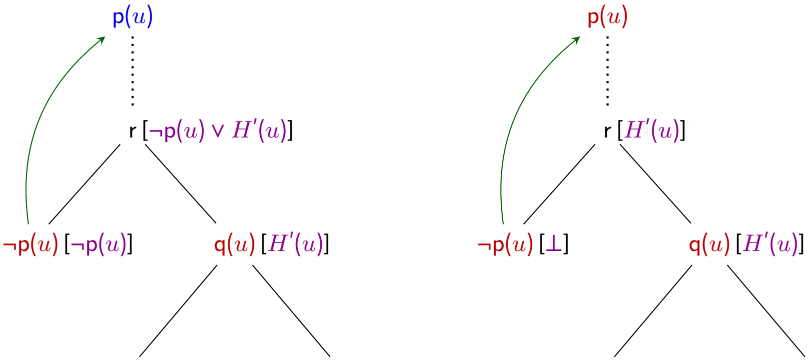


Basic use of the Hyper Property in the Proofs of the Theorems

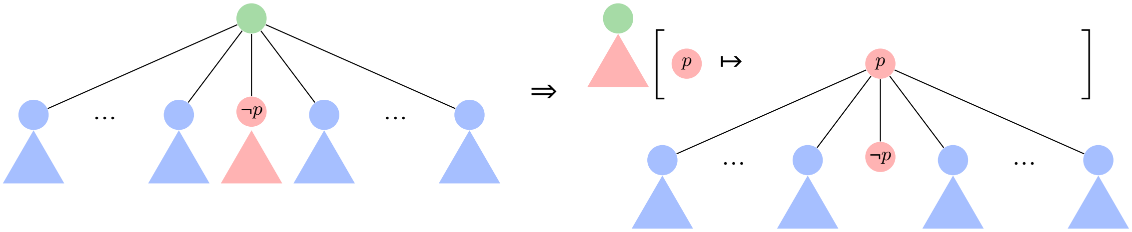
- We distinguish the terms that eventually will be converted to variables
- We show **invariants** in ground interpolant extraction by induction, e.g.

> If C in cnf(ipol(N)), then
> $\mathcal{V}\text{-}\mathcal{M}ax(C) \cap \mathcal{U} \subseteq \mathcal{V}\text{-}\mathcal{M}ax^{-}(C) \cup \mathcal{V}\text{-}\mathcal{M}ax^{+}(\mathsf{path}_{\mathsf{F}}(N))$

- Induction step, two cases:

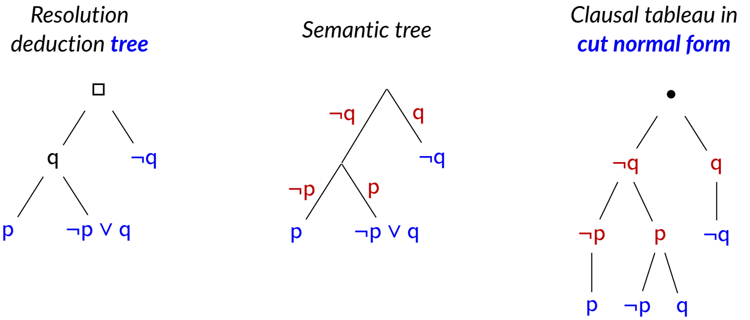


Hyper Conversion



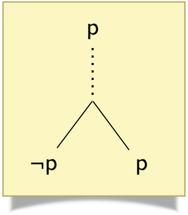
Handling Resolution Proofs

- These three representation are essentially **the same**



- We apply our **hyper conversion to the clausal tableau in cut normal form**

- It eliminates the atomic cuts: if the tableau is closed, regular and hyper it can not have atomic cuts



Is Resolution to Hyper Practically Feasible?

- We tried the Problems of the latest CASC that could be solved by Prover9

		Proof sizes			
Conversion step	*Proofs*	*Min*	*Max*	*Med*	*Time*
Solved by Prover9 in 400 s	**113**				
Prooftrans to binary resolution and paramodulation	112	12	919	55	
Paramodulation to binary resolution	112	10	4,833	**81**	fast
Expansion to cut normal form	**110**	20	97,866,317	**259**	fast, except one 121 s
Hyper conversion	**107**	11	3,110	**77**	fast, some up to 235 s

- Side observation: **the hyper conversion often reduced the proof size**
- The largest on which hyper conversion succeeded had size 51,359 and was reduced to 507

1. Query Reformulation as Craig Interpolation
2. Craig Interpolation with Clausal Tableaux
3. Considered Notions of Range-Restriction
4. Theorems on Interpolation, Range-Restriction and the Horn Property
5. On Proving the Theorems – The Hyper Property
6. **Conclusion**

Outlook and Related Work

- **Towards practice**
  - Systems currently suitable for our approach: CMProver, Prover9
  - Implementation under way
  - E, Vampire output proofs with gaps
- **Other approaches**
  - For query reformulation mostly sequent systems or analytic tableaux [Fitting 1995] are used
  - Vampire's interpolation targeted at verification [Benedikt 2017]
  - Princess' interpolation [Rümmer 2008]
  - So far no other work that considers a variant of range-restriction with general first-order ATP systems
- **Some open issues**
  - Special handling of equality; possible starting-points [Van Gelder, Topor 1993, Baumgartner, Schmidt 2020]
  - Structure preserving normal forms
  - Matching our theorems with DB/KR-relevant formula classes
- **The hyper property may be of further independent interest**
  - Proof presentation
  - Generalizable to "semantics"

- **Prenexing and clausification are utilized here for 3 reasons**
  - The efficient first-order provers
  - Van Gelder and Topor's characterization of range-restriction
  - Our two-stage interpolation method
- In general, an approach to **proof structures with a place for efficient fully automated first-order provers**
  - **Proof transformations** give them freedom to utilize their optimizations
  - The provers only must return a **clausal tableaux proof** or a **resolution proof**

References I

[Abiteboul et al., 1995] Abiteboul, S., Hull, R., and Vianu, V. (1995).
Foundations of Databases.
Addison Wesley.

[Baumgartner and Schmidt, 2020] Baumgartner, P. and Schmidt, R. A. (2020).
Blocking and other enhancements for bottom-up model generation methods.
J. Autom. Reasoning, 64:197–251.

[Benedikt et al., 2017] Benedikt, M., Kostylev, E. V., Mogavero, F., and Tsamoura, E. (2017).
Reformulating queries: Theory and practice.
In Sierra, C., editor, *IJCAI 2017*, pages 837–843. ijcai.org.

[Benedikt et al., 2016] Benedikt, M., Leblay, J., ten Cate, B., and Tsamoura, E. (2016).
Generating Plans from Proofs: The Interpolation-based Approach to Query Reformulation.
Morgan & Claypool.

[Bibel, 1987] Bibel, W. (1987).
Automated Theorem Proving.
Vieweg, Braunschweig.
First edition 1982.

References II

[Bibel and Otten, 2020] Bibel, W. and Otten, J. (2020).
From Schütte's formal systems to modern automated deduction.
In Kahle, R. and Rathjen, M., editors, *The Legacy of Kurt Schütte*, chapter 13, pages 215–249. Springer.

[Bonacina and Johansson, 2015] Bonacina, M. P. and Johansson, M. (2015).
On interpolation in automated theorem proving.
J. Autom. Reasoning, 54(1):69–97.

[Brillout et al., 2011] Brillout, A., Kroening, D., Rümmer, P., and Wahl, T. (2011).
Beyond quantifier-free interpolation in extensions of presburger arithmetic.
In Jhala, R. and Schmidt, D., editors, *VMCAI 2011*, pages 88–102. Springer.

[Craig, 1957] Craig, W. (1957).
Three uses of the Herbrand-Gentzen theorem in relating model theory and proof theory.
J. Symb. Log., 22(3):269–285.

[Huang, 1995] Huang, G. (1995).
Constructing Craig interpolation formulas.
In Du, D.-Z. and Li, M., editors, *COCOON '95*, volume 959 of *LNCS*, pages 181–190. Springer.

References III

[Kovács and Voronkov, 2017] Kovács, L. and Voronkov, A. (2017).
First-order interpolation and interpolating proof systems.
In Eiter, T. and Sands, D., editors, *LPAR-21*, volume 46 of *EPiC*, pages 49–64. EasyChair.

[Letz, 1999] Letz, R. (1999).
Tableau and Connection Calculi. Structure, Complexity, Implementation.
Habilitationsschrift, TU München.
Available from `http://www2.tcs.ifi.lmu.de/~letz/habil.ps`, accessed Jul 19, 2023.

[Letz et al., 1992] Letz, R., Schumann, J., Bayerl, S., and Bibel, W. (1992).
SETHEO: A high-performance theorem prover.
J. Autom. Reasoning, 8(2):183–212.

[Loveland, 1978] Loveland, D. W. (1978).
Automated Theorem Proving: A Logical Basis.
North-Holland, Amsterdam.

[Nash et al., 2010] Nash, A., Segoufin, L., and Vianu, V. (2010).
Views and queries: Determinacy and rewriting.
ACM Trans. Database Syst., 35(3):1–41.

References IV

[Otten and Bibel, 2003] Otten, J. and Bibel, W. (2003).
leanCoP: lean connection-based theorem proving.
J. Symb. Comput., 36(1-2):139–161.

[Stickel, 1988] Stickel, M. E. (1988).
A Prolog technology theorem prover: implementation by an extended Prolog compiler.
J. Autom. Reasoning, 4(4):353–380.

[Toman and Weddell, 2011] Toman, D. and Weddell, G. (2011).
Fundamentals of Physical Design and Query Compilation.
Morgan & Claypool.

[Van Gelder and Topor, 1991] Van Gelder, A. and Topor, R. W. (1991).
Safety and translation of relational calculus queries.
ACM Trans. Database Syst., 16(2):235–278.

[Wernhard, 2016] Wernhard, C. (2016).
The PIE system for proving, interpolating and eliminating.
In Fontaine, P., Schulz, S., and Urban, J., editors, *PAAR 2016*, volume 1635 of *CEUR Workshop Proc.*, pages 125–138. CEUR-WS.org.

[Wernhard, 2020] Wernhard, C. (2020).
Facets of the PIE environment for proving, interpolating and eliminating on the basis of first-order logic.
In Hofstedt, P. et al., editors, *DECLARE 2019, Revised Selected Papers*, volume 12057 of *LNCS (LNAI)*, pages 160–177. Springer.

[Wernhard, 2021] Wernhard, C. (2021).
Craig interpolation with clausal first-order tableaux.
J. Autom. Reasoning, 65(5):647–690.