(Un)professional Publishing
A Drama in Five Proofs

Uwe Waldmann*

August 2023

Prologue

As a scientist, I am used to being treated badly by academic publishers. I am used to publishers who ask me to submit LaTeX code for my accepted paper, but who send me the galley proofs in pdf format, rather than as LaTeX code, and who ask me to check them within 48 hours. (Even though there are tools like `diffpdf` that do a decent job to compare the manuscript pdf and the proof pdf file, a textual `diff` of the manuscript LaTeX file and the proof LaTeX file would be much more informative.) I am used to the fact that the galley proofs are decorated with watermarks and use fonts that are different from the manuscript, making a mechanical comparison even more difficult. I am used to copy editors who insert or delete spaces randomly, who turn aligned lines into non-aligned lines, and who turn balanced vertical space into non-balanced vertical space. I am used to copy editors who turn roman subscripts in formulas into italic subscripts, replace long arrow by short arrows (or vice versa), or replace correctly spelled names in the bibliography by incorrectly spelled names. I have even seen copy editors who turn two non-related tables into a single table. (Sample images can be found in the appendix.) But things can always get worse . . .

Proof #1 "Hanlon's Razor"

The *Journal of Automated Reasoning (JAR)* published by Springer-Verlag is one of the leading journals in my reseach area. In May 2021, my colleagues and me submitted a paper [4] to a special issue of the JAR. The paper was accepted, and in March 2022, we sent the camera-ready manuscript to the publisher. Six weeks later, we received the publisher's proof, with best regards from the Springer Nature Correction Team, Chennai, India. "Please submit your corrections within 2 working days," as usual.

*MPI for Informatics, Saarland Informatics Campus E1.4, 66123 Saarbrücken, Germany

I consider myself a typography nerd, and one of my coauthors likes to read English style guides as a pastime. You may assume that the manuscript that we submitted did not require any orthographical, typographical, or stylistic corrections. That didn't stop Springer's "Correction Team" from editing it nevertheless—with the expected results.

The JAR asked its authors to format manuscripts using the LaTeX style `svjour3.cls`. The final version, however, was produced using a substantially different style. And it was not only the fonts that were different (which would be understandable since the printed version may use non-free fonts), but also headline styles and spacing. For instance, LaTeX `\paragraph` headlines, which were clearly discernible in our manuscript, had become almost invisible (Fig. 1).

There were more problems. Readers familiar with typographic rules will know that there are two different conventions for typesetting dashes in the middle of an English sentence: either using a closed-up em dash ("as—even moderately gifted—copy editors should know") or using a spaced en (or rarely em) dash ("as – even moderately gifted – copy editors should know"). The first style is preferred in the US, the second one in the UK. In our manuscript, we had consistently used the first one. It turned out that the "Correction Team" had inserted spaces at random spots, in particular after (but not before!) 12 out of 33 dashes in the paper. (Fig. 1). They had also indented some lines randomly and turned some math symbols that should have remained upright into italics. Furthermore, for unknown reasons, some references (2–6, 17–19, 36–37) were no longer alphabetically sorted (Fig. 2).

These errors are, unfortunately, rather common. I have seen them in other proofs in the previous years. Some of them are probably systematic, caused by bugs in the converter that the publisher uses to process LaTeX files, and the others can be explained by negligence or incompetence, so Hanlon's Razor applies: "Never attribute to malice that which is adequately explained by stupidity." The next problem was different, however: One of the reviewers had asked us to provide an index of concepts, and we had formatted the index entries in the usual way: concept, comma, space, page number. In the publisher's proof, however, the space following the comma had been deleted—not once, not everywhere, but randomly in 51 out of 113 index entries (Fig. 3). Frankly, I have no idea how such a change could be explained by stupidity, and I wonder whether the people who did it were perhaps paid by the number of edits they made, rather than by the number of useful edits.

At this point, we stopped reading the proof and wrote an angry letter to the "Correction Team", asking them to start from scratch.

For some of the notions in Sections 2.1 and 2.2 one can find alternative definitions in the literature.

Redundancy Criteria As in Bachmair and Ganzinger's chapter [6, Section 4.1], we have specified in condition (R1) of redundancy criteria that the deletion of redundant formulas must preserve inconsistency. Alternatively, one can require that redundant formulas must be entailed by the nonredundant ones—i.e., $N \setminus Red_{\mathrm{F}}(N) \models Red_{\mathrm{F}}(N)$—leading to some obvious changes in Lemmas 10 and 37.

Bachmair and Ganzinger's definition of a redundancy criterion differs from ours in that they require only conditions (R1)–(R3). They call a redundancy criterion *effective* if an inference $\iota \in \mathit{Inf}$ is in $Red_{\mathrm{I}}(N)$ whenever $concl(\iota) \in N \cup Red_{\mathrm{F}}(N)$. As demonstrated by Lemma 1, that condition is equivalent to our condition (R4).

Inferences from Redundant Premises Inferences from redundant premises are sometimes excluded in the definitions of saturation, fairness, and refutational completeness [6], and sometimes not [5, 10, 30, 44].[1] Similarly, redundancy of inferences is sometimes defined in such a way that inferences from redundant premises are necessarily redundant them-

---

For some of the notions in Sects. 2.1 and 2.2 one can find alternative definitions in the literature.

Redundancy Criteria As in Bachmair and Ganzinger's chapter [5, Sect. 4.1], we have specified in condition (R1) of redundancy criteria that the deletion of redundant formulas must preserve inconsistency. Alternatively, one can require that redundant formulas must be entailed by the nonredundant ones— i.e., $N \setminus Red_{\mathrm{F}}(N) \models Red_{\mathrm{F}}(N)$— leading to some obvious changes in Lemmas 10 and 37.

Bachmair and Ganzinger's definition of a redundancy criterion differs from ours in that they require only conditions (R1)–(R3). They call a redundancy criterion *effective* if an inference $\iota \in \mathit{Inf}$ is in $Red_{\mathrm{I}}(N)$ whenever $concl(\iota) \in N \cup Red_{\mathrm{F}}(N)$. As demonstrated by Lemma 1, that condition is equivalent to our condition (R4).

Inferences from Redundant Premises Inferences from redundant premises are sometimes excluded in the definitions of saturation, fairness, and refutational completeness [5], and sometimes not [4, 10, 30, 44].[1] Similarly, redundancy of inferences is sometimes defined in such a way that inferences from redundant premises are necessarily redundant themselves [4,

Figure 1: Top: authors' manuscript; bottom: proof #1. The paragraph headlines "Redundancy Criteria" and "Inferences from Redundant Premises" are barely recognizable as headlines. Spurious spaces have been inserted after the dashes and before "[5]".

2. Bachmair, L., Dershowitz, N., Plaisted, D.A.: Completion without failure. In: H. Aït-Kaci, M. Nivat (eds.) Rewriting Techniques—Resolution of Equations in Algebraic Structures, vol. 2, pp. 1–30. Academic Press (1989)
3. Bachmair, L., Ganzinger, H.: On restrictions of ordered paramodulation with simplification. In: M.E. Stickel (ed.) CADE-10, *LNCS*, vol. 449, pp. 427–441. Springer (1990). DOI 10.1007/3-540-52885-7_105
4. Bachmair, L., Ganzinger, H.: Associative-commutative superposition. In: N. Dershowitz, N. Lindenstrauss (eds.) CTRS-94, *LNCS*, vol. 968, pp. 1–14. Springer (1994)
5. Bachmair, L., Ganzinger, H.: Rewrite-based equational theorem proving with selection and simplification. J. Log. Comput. **4**(3), 217–247 (1994)
6. Bachmair, L., Ganzinger, H.: Resolution theorem proving. In: A. Robinson, A. Voronkov (eds.) Handbook of Automated Reasoning, vol. I, pp. 19–99. Elsevier and MIT Press (2001)
7. Bachmair, L., Ganzinger, H., Waldmann, U.: Superposition with simplification as a desision procedure for the monadic class with equality. In: G. Gottlob, A. Leitsch, D. Mundici (eds.) KGC '93, *LNCS*, vol. 713, pp. 83–96. Springer (1993). DOI 10.1007/BFb0022557

2. Bachmair, L., Ganzinger, H.: On restrictions of ordered paramodulation with simplification. In: Stickel, M.E. (ed.) CADE-10. LNCS, vol. 449, pp. 427–441. Springer, Heidelberg (1990). https://doi.org/10.1007/3-540-52885-7_105
3. Bachmair, L., Ganzinger, H.: Associative-commutative superposition. In: Dershowitz, N., Lindenstrauss, N. (eds.) CTRS-94. LNCS, pp. 1–14. Springer, Heidelberg (1994)
4. Bachmair, L., Ganzinger, H.: Rewrite-based equational theorem proving with selection and simplification. J. Log. Comput. **4**(3), 217–247 (1994)
5. Bachmair, L., Ganzinger, H.: Resolution theorem proving. In: Robinson, A., Voronkov, A. (eds.) Handbook of Automated Reasoning, vol. I, pp. 19–99. Elsevier and MIT Press, Cambridge (2001)
6. Bachmair, L., Dershowitz, N., Plaisted, D.A.: Completion without failure. In: Aït-Kaci, H., Nivat, M. (eds.) Rewriting Techniques-Resolution of Equations in Algebraic Structures, vol. 2, pp. 1–30. Academic Press, Boston (1989)
7. Bachmair, L., Ganzinger, H., Waldmann, U.: Superposition with simplification as a desision procedure for the monadic class with equality. In: Gottlob, G., Leitsch, A., Mundici, D. (eds.) KGC '93. LNCS, vol. 713, pp. 83–96. Springer, Heidelberg (1993). https://doi.org/10.1007/BFb0022557

Figure 2: Top: authors' manuscript; bottom: proof #1. Reference 6 should be placed before reference 2.

A-sequence, 5	A-sequence,5
standard redundancy criterion, 5	standard redundancy criterion, 5
statically refutationally complete, 5	statically refutationally complete,5
T, 30	T, 30
T_∞, 6	T_∞,6
$\lceil T \rceil$, 31	$\lceil T \rceil$,31
tiebreaker ordering, 16	tiebreaker ordering, 17
trivial redundancy criterion, 5	trivial redundancy criterion,5
undef, 14	*undef*,14
Zipperposition loop prover, 33	Zipperposition loop prover,33
ZL, 33	ZL, 33

Figure 3: Left: authors' manuscript; right: proof #1. Spaces after commas have been deleted randomly.

Proof #2 "A New Hope"

The second proof arrived. The "Correction Team" had fixed some of the bugs we had complained about (missing spaces in the index, superfluous spaces after dashes, unrecognizable headlines), but it was obvious that they had *not* started from scratch. The bibliography was still incorrectly sorted (Fig. 4), there were still italic letters in mathematical formulas that should have been upright (Fig. 5), and the misaligned lines that we had mentioned earlier were still misaligned (Fig. 6). Furthermore one paragraph had been split into four single-sentence paragraphs, probably in order to simplify pagebreaking (Fig. 7). We wrote one more letter to the "Correction Team" and asked them to fix the remaining bugs.

2. Bachmair, L., Ganzinger, H.: On restrictions of ordered paramodulation with simplification. In: Stickel, M.E. (ed.) CADE-10. LNCS, vol. 449, pp. 427–441. Springer, Heidelberg (1990). https://doi.org/10.1007/3-540-52885-7_105
3. Bachmair, L., Ganzinger, H.: Associative-commutative superposition. In: Dershowitz, N., Lindenstrauss, N. (eds.) CTRS-94. LNCS, pp. 1–14. Springer, Heidelberg (1994)
4. Bachmair, L., Ganzinger, H.: Rewrite-based equational theorem proving with selection and simplification. J. Log. Comput. **4**(3), 217–247 (1994)
5. Bachmair, L., Ganzinger, H.: Resolution theorem proving. In: Robinson, A., Voronkov, A. (eds.) Handbook of Automated Reasoning, vol. I, pp. 19–99. Elsevier and MIT Press, Cambridge (2001)
6. Bachmair, L., Dershowitz, N., Plaisted, D.A.: Completion without failure. In: Aït-Kaci, H., Nivat, M. (eds.) Rewriting Techniques-Resolution of Equations in Algebraic Structures, vol. 2, pp. 1–30. Academic Press, Boston (1989)
7. Bachmair, L., Ganzinger, H., Waldmann, U.: Superposition with simplification as a desision procedure for the monadic class with equality. In: Gottlob, G., Leitsch, A., Mundici, D. (eds.) KGC '93. LNCS, vol. 713, pp. 83–96. Springer, Heidelberg (1993). https://doi.org/10.1007/BFb0022557

Figure 4: Publisher's proof #2. Reference 6 should be placed before reference 2.

Lemma 66 *Every* $\Longrightarrow_{\mathsf{GC}}$*-derivation is a* $\rhd_{Red^{\cap\mathcal{G}_{\mathrm{L}},\sqsupset}}$*-derivation.*

Proof We need to show that every labeled formula that is deleted in a $\Longrightarrow_{\mathsf{GC}}$-step is $Red^{\cap\mathcal{G}_{\mathrm{L}},\sqsupset}$-redundant w.r.t. the remaining labeled formulas. For PROCESS, this is trivial. For

Lemma 66 *Every* $\Longrightarrow_{\mathit{GC}}$*-derivation is a* $\rhd_{Red^{\cap\mathcal{G}_{\mathrm{L}},\sqsupset}}$*-derivation.*

Proof We need to show that every labeled formula that is deleted in a $\Longrightarrow_{\mathsf{GC}}$-step is $Red^{\cap\mathcal{G}_{\mathrm{L}},\sqsupset}$-redundant w.r.t. the remaining labeled formulas. For PROCESS, this is trivial. For INFER, the

Figure 5: Top: authors' manuscript; bottom: proof #2. The subscript GC should always be in an upright sans-serif font (as in line 1027), even if the surrounding text is italic.

Given two sets of formulas **F** and **G**, an **F**-inference system *FInf*, a **G**-inference system *GInf*, and a redundancy criterion *Red* for *GInf*, let $\mathcal{G}$ be a function that maps every formula in **F** to a subset of **G** and every **F**-inference in *FInf* to *undef* or to a subset of *GInf*. The function $\mathcal{G}$ is called a *grounding function* if

(G1) for every $\bot \in \mathbf{F}_\bot$, $\emptyset \neq \mathcal{G}(\bot) \subseteq \mathbf{G}_\bot$;
(G2) for every $C \in \mathbf{F}$, if $\bot \in \mathcal{G}(C)$ and $\bot \in \mathbf{G}_\bot$ then $C \in \mathbf{F}_\bot$;
(G3) for every $\iota \in \mathit{FInf}$, if $\mathcal{G}(\iota) \neq \mathit{undef}$, then $\mathcal{G}(\iota) \subseteq \mathit{Red}_{\mathrm{I}}(\mathcal{G}(\mathit{concl}(\iota)))$.

The function $\mathcal{G}$ is extended to sets $N \subseteq \mathbf{F}$ by defining $\mathcal{G}(N) := \bigcup_{C \in N} \mathcal{G}(C)$ for all N. Analogously, for a set $I \subseteq \mathit{FInf}$, $\mathcal{G}(I) := \bigcup_{\iota \in I,\, \mathcal{G}(\iota) \neq \mathit{undef}} \mathcal{G}(\iota)$.

Remark 27 Conditions (G1) and (G2) express that *false* formulas may only be mapped to sets of *false* formulas, and that only *false* formulas may be mapped to sets of *false* formulas. For most applications, it would be possible to replace condition (G3) by

(G3′) for every $\iota \in \mathit{FInf}$, if $\mathcal{G}(\iota) \neq \mathit{undef}$, then $\mathit{concl}(\mathcal{G}(\iota)) \subseteq \mathcal{G}(\mathit{concl}(\iota))$,

which implies (G3) by property (R4). There are some calculi, however, for which (G3′) is too strong. Typical examples are calculi where the **F**-inferences include some normalization

---

Given two sets of formulas **F** and **G**, an **F**-inference system *FInf*, a **G**-inference system *GInf*, and a redundancy criterion *Red* for *GInf*, let $\mathcal{G}$ be a function that maps every formula in **F** to a subset of **G** and every **F**-inference in *FInf* to *undef* or to a subset of *GInf*. The function $\mathcal{G}$ is called a *grounding function* if

(G1) for every $\bot \in \mathbf{F}_\bot$, $\emptyset \neq \mathcal{G}(\bot) \subseteq \mathbf{G}_\bot$;
(G2) for every $C \in \mathbf{F}$, if $\bot \in \mathcal{G}(C)$ and $\bot \in \mathbf{G}_\bot$ then $C \in \mathbf{F}_\bot$;
(G3) for every $\iota \in \mathit{FInf}$, if $\mathcal{G}(\iota) \neq \mathit{undef}$, then $\mathcal{G}(\iota) \subseteq \mathit{Red}_{\mathrm{I}}(\mathcal{G}(\mathit{concl}(\iota)))$.

The function $\mathcal{G}$ is extended to sets $N \subseteq \mathbf{F}$ by defining $\mathcal{G}(N) := \bigcup_{C \in N} \mathcal{G}(C)$ for all N. Analogously, for a set $I \subseteq \mathit{FInf}$, $\mathcal{G}(I) := \bigcup_{\iota \in I,\, \mathcal{G}(\iota) \neq \mathit{undef}} \mathcal{G}(\iota)$.

Remark 27 Conditions (G1) and (G2) express that *false* formulas may only be mapped to sets of *false* formulas, and that only *false* formulas may be mapped to sets of *false* formulas. For most applications, it would be possible to replace condition (G3) by

(G3′) for every $\iota \in \mathit{FInf}$, if $\mathcal{G}(\iota) \neq \mathit{undef}$, then $\mathit{concl}(\mathcal{G}(\iota)) \subseteq \mathcal{G}(\mathit{concl}(\iota))$,

which implies (G3) by property (R4). There are some calculi, however, for which (G3′) is too strong. Typical examples are calculi where the **F**-inferences include some normalization or

Figure 6: Top: authors' manuscript; bottom: proof #2. The lines starting with (G1), (G2), (G3), and (G3′) should have been aligned.

Example 46 For resolution or superposition in standard first-order logic, we can define the *instantiation* quasi-ordering $\gtrsim$ on clauses by $C \gtrsim C'$ if and only if $C = C'\sigma$ for some substitution σ. In particular, if C and C' are α-renamings of each other, then $C \gtrsim C'$ and $C \lesssim C'$. The instantiation ordering $\gtrdot := \gtrsim \setminus \lesssim$ is well founded. By choosing $\sqsupset := \gtrdot$, we obtain a criterion $Red^{\mathcal{G},\sqsupset}$ that includes standard redundancy (Example 3) and also supports subsumption deletion. (It is customary to define subsumption so that C is subsumed by C' if $C = C'\sigma \vee D$ for some substitution σ and some possibly empty clause D, but since the case where D is nonempty is already supported by the standard redundancy criterion, the instantiation ordering $\gtrdot$ is sufficient.)

Similarly, for proof calculi modulo commutativity (C) or associativity and commutativity (AC), we can let $C \gtrsim C'$ be true if there exists a substitution σ such that C equals $C'\sigma$ up to the equational theory (C or AC). The relation $\gtrdot = \gtrsim \setminus \lesssim$ is then again well founded.

Example 46 For resolution or superposition in standard first-order logic, we can define the *instantiation* quasi-ordering $\gtrsim$ on clauses by $C \gtrsim C'$ if and only if $C = C'\sigma$ for some substitution σ.

In particular, if C and C' are α-renamings of each other, then $C \gtrsim C'$ and $C \lesssim C'$.

The instantiation ordering $\gtrdot := \gtrsim \setminus \lesssim$ is well founded. By choosing $\sqsupset := \gtrdot$, we obtain a criterion $Red^{\mathcal{G},\sqsupset}$ that includes standard redundancy (Example 3) and also supports subsumption deletion.

(It is customary to define subsumption so that C is subsumed by C' if $C = C'\sigma \vee D$ for some substitution σ and some possibly empty clause D, but since the case where D is nonempty is already supported by the standard redundancy criterion, the instantiation ordering $\gtrdot$ is sufficient.)

Similarly, for proof calculi modulo commutativity (C) or associativity and commutativity (AC), we can let $C \gtrsim C'$ be true if there exists a substitution σ such that C equals $C'\sigma$ up to the equational theory (C or AC). The relation $\gtrdot = \gtrsim \setminus \lesssim$ is then again well founded.

Figure 7: Top: authors' manuscript; bottom: proof #2. Splitting this paragraph into four paragraphs does not make sense.

Proof #3 "Groundhog Day"

The third proof arrived, the bibliography was still incorrectly sorted, and the subscripts that should have been upright were still italic. In one formula, a spurious comma had been inserted (Fig. 8). The misaligned lines that we had complained about in the second letter were finally aligned, but the prime mark in "(G3′)" had been turned into a closing quote: "(G3')" (Fig. 9).

We wrote a third letter to the "Correction Team" and asked them to fix the remaining bugs.

Proof #4 "Close, But No Cigar"

The fourth proof arrived. The bibliography entries were finally sorted alphabetically, and the subscripts that should have upright were finally upright. The spurious comma had been eliminated, but instead, one additional space had been inserted (Fig. 10). Even more surprising, the double arrow that formerly had an italic subscript had been turned into a single arrow (Fig. 11).

We wrote the fourth letter to the "Correction Team" and asked them to fix the remaining bugs.

Proof #5 "Rejoice!"

The fifth proof arrived. After just five iterations, the "Springer Nature Correction Team" had finally managed to correct all the errors they had previously made. And even better, they had not introduced any new ones. We could still have complained about some details, such as insufficient horizontal space between letters and superscripts, but we felt that it might be a bad idea to push our luck. After all, five iterations to reach a publishable version are not that bad—I've been told that colleagues of mine recently needed more than *ten* iterations.

Proof To show that (iii) implies (i), assume that $(\mathit{Inf},\mathit{Red}')$ is statically refutationally complete. That is, the property

$$N \models \{\bot\} \text{ for some } \bot \in \mathbf{F}_\bot \text{ implies } \bot' \in N \text{ for some } \bot' \in \mathbf{F}_\bot \qquad (*)$$

Proof To show that (iii) implies (i), assume that $(\mathit{Inf}, \mathit{Red}')$ is statically refutationally complete. That is, the property

$$N \models \{\bot\} \text{ for some, } \bot \in \mathbf{F}_\bot \text{ implies } \bot' \in N \text{ for some } \bot' \in \mathbf{F}_\bot \qquad (*)$$

Figure 8: Top: authors' manuscript; bottom: proof #3. The inserted comma makes no sense here.

Given two sets of formulas **F** and **G**, an **F**-inference system *FInf*, a **G**-inference system *GInf*, and a redundancy criterion *Red* for *GInf*, let $\mathcal{G}$ be a function that maps every formula in **F** to a subset of **G** and every **F**-inference in *FInf* to *undef* or to a subset of *GInf*. The function $\mathcal{G}$ is called a *grounding function* if

(G1) for every $\bot \in \mathbf{F}_\bot$, $\emptyset \neq \mathcal{G}(\bot) \subseteq \mathbf{G}_\bot$;
(G2) for every $C \in \mathbf{F}$, if $\bot \in \mathcal{G}(C)$ and $\bot \in \mathbf{G}_\bot$ then $C \in \mathbf{F}_\bot$;
(G3) for every $\iota \in \mathit{FInf}$, if $\mathcal{G}(\iota) \neq \mathit{undef}$, then $\mathcal{G}(\iota) \subseteq \mathit{Red}_\mathrm{I}(\mathcal{G}(\mathit{concl}(\iota)))$.

The function $\mathcal{G}$ is extended to sets $N \subseteq \mathbf{F}$ by defining $\mathcal{G}(N) := \bigcup_{C \in N} \mathcal{G}(C)$ for all N. Analogously, for a set $I \subseteq \mathit{FInf}$, $\mathcal{G}(I) := \bigcup_{\iota \in I,\, \mathcal{G}(\iota) \neq \mathit{undef}} \mathcal{G}(\iota)$.

Remark 27 Conditions (G1) and (G2) express that *false* formulas may only be mapped to sets of *false* formulas, and that only *false* formulas may be mapped to sets of *false* formulas. For most applications, it would be possible to replace condition (G3) by

(G3′) for every $\iota \in \mathit{FInf}$, if $\mathcal{G}(\iota) \neq \mathit{undef}$, then $\mathit{concl}(\mathcal{G}(\iota)) \subseteq \mathcal{G}(\mathit{concl}(\iota))$,

which implies (G3) by property (R4). There are some calculi, however, for which (G3′) is too strong. Typical examples are calculi where the **F**-inferences include some normalization or

Given two sets of formulas **F** and **G**, an **F**-inference system *FInf*, a **G**-inference system *GInf*, and a redundancy criterion *Red* for *GInf*, let $\mathcal{G}$ be a function that maps every formula in **F** to a subset of **G** and every **F**-inference in *FInf* to *undef* or to a subset of *GInf*. The function $\mathcal{G}$ is called a *grounding function* if

(G1) for every $\bot \in \mathbf{F}_\bot$, $\emptyset \neq \mathcal{G}(\bot) \subseteq \mathbf{G}_\bot$;
(G2) for every $C \in \mathbf{F}$, if $\bot \in \mathcal{G}(C)$ and $\bot \in \mathbf{G}_\bot$ then $C \in \mathbf{F}_\bot$;
(G3) for every $\iota \in \mathit{FInf}$, if $\mathcal{G}(\iota) \neq \mathit{undef}$, then $\mathcal{G}(\iota) \subseteq \mathit{Red}_\mathrm{I}(\mathcal{G}(\mathit{concl}(\iota)))$.

The function $\mathcal{G}$ is extended to sets $N \subseteq \mathbf{F}$ by defining $\mathcal{G}(N) := \bigcup_{C \in N} \mathcal{G}(C)$ for all N. Analogously, for a set $I \subseteq \mathit{FInf}$, $\mathcal{G}(I) := \bigcup_{\iota \in I,\, \mathcal{G}(\iota) \neq \mathit{undef}} \mathcal{G}(\iota)$.

Remark 27 Conditions (G1) and (G2) express that *false* formulas may only be mapped to sets of *false* formulas, and that only *false* formulas may be mapped to sets of *false* formulas. For most applications, it would be possible to replace condition (G3) by

(G3’) for every $\iota \in \mathit{FInf}$, if $\mathcal{G}(\iota) \neq \mathit{undef}$, then $\mathit{concl}(\mathcal{G}(\iota)) \subseteq \mathcal{G}(\mathit{concl}(\iota))$,

which implies (G3) by property (R4). There are some calculi, however, for which (G3′) is too strong. Typical examples are calculi where the **F**-inferences include some normalization or

Figure 9: Top: proof #2; bottom: proof #3. The lines starting with (G1), (G2), (G3), and (G3′) are finally aligned, but the symbol following “(G3” should be a prime mark (as in line 552), not a closing quote.

Proof To show that (iii) implies (i), assume that (Inf, Red') is statically refutationally complete. That is, the property

$$N \models \{\bot\} \text{ for some } \bot \in \mathbf{F}_\bot \text{ implies } \bot' \in N \text{ for some } \bot' \in \mathbf{F}_\bot \qquad (*)$$

Proof To show that (iii) implies (i), assume that (Inf, Red') is statically refutationally complete. That is, the property

$$N \models \{\bot\} \text{ for some, } \bot \in \mathbf{F}_\bot \text{ implies } \bot' \in N \text{ for some } \bot' \in \mathbf{F}_\bot \qquad (*)$$

Proof To show that (iii) implies (i), assume that (Inf, Red') is statically refutationally complete. That is, the property

$$N \models \{\bot\} \text{ for some } \bot \in \mathbf{F}_\bot \text{ implies } \bot' \in N \text{ for some } \ \bot' \in \mathbf{F}_\bot \qquad (*)$$

Figure 10: Top: authors' manuscript; middle: proof #3; bottom: proof #4. The comma has been eliminated, but the space after "for some" is now too large.

Lemma 66 *Every* $\Longrightarrow_{\mathsf{GC}}$*-derivation is a* $\rhd_{Red^{\cap \mathcal{G}_\mathbf{L}}, \sqsupset}$*-derivation.*

Proof We need to show that every labeled formula that is deleted in a $\Longrightarrow_{\mathsf{GC}}$-step is $Red^{\cap \mathcal{G}_\mathbf{L}, \sqsupset}$-redundant w.r.t. the remaining labeled formulas. For PROCESS, this is trivial. For

Lemma 66 *Every* $\Longrightarrow_{\mathsf{GC}}$*-derivation is a* $\rhd_{Red^{\cap \mathcal{G}_\mathbf{L}}, \sqsupset}$*-derivation.*

Proof We need to show that every labeled formula that is deleted in a $\Longrightarrow_{\mathsf{GC}}$-step is $Red^{\cap \mathcal{G}_\mathbf{L}, \sqsupset}$-redundant w.r.t. the remaining labeled formulas. For PROCESS, this is trivial. For INFER, the

Lemma 66 *Every* $\rightarrow_{\mathsf{GC}}$*-derivation is a* $\rhd_{Red^{\cap \mathcal{G}_\mathbf{L}}, \sqsupset}$*-derivation.*

Proof We need to show that every labeled formula that is deleted in a $\Longrightarrow_{\mathsf{GC}}$-step is $Red^{\cap \mathcal{G}_\mathbf{L}, \sqsupset}$-redundant w.r.t. the remaining labeled formulas. For PROCESS, this is trivial. For INFER, the

Figure 11: Top: authors' manuscript; middle: proof #3; bottom: proof #4. The single arrow should have been a double arrow as in line 1026.

Epilogue

What we experienced with this submission to the Journal of Automated Reasoning was special in that some of the changes made by the typesetters could not, by the best of intentions, be explained by accidents or stupidity. On the other hand, from an author's point of view, errors introduced by utter incompetence and gross negligence are just as annoying as errors introduced by intentional sabotage. This problem is not restricted to this particular paper, not restricted to the Journal of Automated Reasoning, and not restricted to Springer Nature. (Examples can be found in the appendix.) In fact, Elsevier also has a reputation for mistreating properly formatted manuscripts—I just did not have any publications at Elsevier journals in the last years, so all my examples are from Springer Nature.

I have seen enough badly formatted manuscripts to know that some authors are completely clueless when it comes to typesetting rules and that they would profit significantly from the help of competent professional copy editors. The key words are "competent" and "professional", though. If copy editors do not recognize when something needs to be repaired in a manuscript and when it doesn't, they are not competent. If copy editors do not know what is good typesetting practice and what's not or if they are not familiar with the tools they are using, they are not competent. If copy editors deliberately screw up authors' manuscripts, they behave thoroughly unprofessionally. If copy editors lack competence or professional ethics, then their involvement in the production of a publication is not only pointless but damaging. And if publishers do not manage to organize their business processes so that their employees (or their subcontractors' employees) do not cause damage, there is no place for them.

Addendum, February 20, 2024

I tried to contact Springer Nature in this matter. Unfortunately, Springer Nature makes every effort to shield its management from authors' complaints, therefore I was unable to figure out who is in charge of this. The only thing I found was the contact form of the compliance department. So I used that one. I got a reply from a manager of the compliance department, who told me that my enquiry was not within the Governance, Risk and Compliance team's remit, but that she had passed it on to the correct department internally on September 20, 2023, and that she had asked them to follow up with me directly. That was five months ago. Since then, nothing happened.

Appendix

Bad copy editing is not restricted to academic journals. This appendix shows a collection of bloopers from publisher's proofs of recent conference proceeding volumes.

Vertical Spacing

Irregular vertical spacing is one of the most common problems introduced by publishers' copy editors. Fig. 12 shows an example where the vertical space above a displayed formula has been doubled, whereas the vertical space below the same formula has been kept constant.

Example 1. Applied variables give rise to subtle situations with no counterparts in first-order logic. Consider the clauses

$$\mathsf{f\,a} \approx \mathsf{c} \qquad\qquad \mathsf{h}\,(y\,\mathsf{b})\,(y\,\mathsf{a}) \not\approx \mathsf{h}\,(\mathsf{g}\,(\mathsf{f\,b}))\,(\mathsf{g\,c})$$

where $\mathsf{f\,a} \succ \mathsf{c}$. It is easy to see that the clause set is unsatisfiable, by grounding the second clause with $\theta = \{y \mapsto (\lambda x.\,\mathsf{g}\,(\mathsf{f}\,x))\}$. However, to mimic the superposition inference that can be performed at the ground level, it is necessary to superpose at

Example 1. Applied variables give rise to subtle situations with no counterparts in first-order logic. Consider the clauses

$$\mathsf{f\,a} \approx \mathsf{c} \qquad\qquad \mathsf{h}\,(y\,\mathsf{b})\,(y\,\mathsf{a}) \not\approx \mathsf{h}\,(\mathsf{g}\,(\mathsf{f\,b}))\,(\mathsf{g\,c})$$

where $\mathsf{f\,a} \succ \mathsf{c}$. It is easy to see that the clause set is unsatisfiable, by grounding the second clause with $\theta = \{y \mapsto (\lambda x.\,\mathsf{g}\,(\mathsf{f}\,x))\}$. However, to mimic the superposition inference that can be performed at the ground level, it is necessary to

Figure 12: Top: authors' manuscript; bottom: proof #1. The vertical space above and below the formula should be the same. [2]

Fig. 13 shows a similar problem. Again the vertical space above the formula has been roughly doubled. In addition the closing parenthesis has been moved left so that it overlaps the preceding symbol. In the second proof, the overlap has been removed, but the vertical space above and below the formula is still not equal. Furthermore, in the preceding paragraph (and some others) the interword space has almost disappeared completely (note in particular the fourth line) and the final linebreak before "**F**" is hideous.

Intersections of Liftings. The above results can be extended in a straightforward way to intersections of lifted redundancy criteria. As before, let $\mathbf{F}$ and $\mathbf{G}$ be two sets of formulas, and let *FInf* be an $\mathbf{F}$-inference system. In addition, let Q be a set. For every $q \in Q$, let $\models^q$ be a consequence relation over $\mathbf{G}$, let GInf^q be a $\mathbf{G}$-inference system, let Red^q be a redundancy criterion for $\models^q$ and GInf^q, and let $\mathcal{G}^q$ be a grounding function from $\mathbf{F}$ and *FInf* to $\mathbf{G}$ and GInf^q. Let $\sqsupset$ be a well-founded strict partial ordering on $\mathbf{F}$.

For each $q \in Q$, we know by Theorem 6 that the $(\mathcal{G}^q, \emptyset)$-lifting $\mathit{Red}^{q,\mathcal{G}^q,\emptyset} = (\mathit{Red}_{\mathrm{I}}^{q,\mathcal{G}^q}, \mathit{Red}_{\mathrm{F}}^{q,\mathcal{G}^q,\emptyset})$ and the $(\mathcal{G}^q, \sqsupset)$-lifting $\mathit{Red}^{q,\mathcal{G}^q,\sqsupset} = (\mathit{Red}_{\mathrm{I}}^{q,\mathcal{G}^q}, \mathit{Red}_{\mathrm{F}}^{q,\mathcal{G}^q,\sqsupset})$ of Red^q are redundancy criteria for $\models^q_{\mathcal{G}^q}$ and *FInf*. Consequently, the intersections

$$\mathit{Red}^{\cap\mathcal{G}} := (\mathit{Red}_{\mathrm{I}}^{\cap\mathcal{G}}, \mathit{Red}_{\mathrm{F}}^{\cap\mathcal{G}}) := \left(\bigcap_{q\in Q} \mathit{Red}_{\mathrm{I}}^{q,\mathcal{G}^q}, \bigcap_{q\in Q} \mathit{Red}_{\mathrm{F}}^{q,\mathcal{G}^q,\emptyset}\right) \text{ and}$$
$$\mathit{Red}^{\cap\mathcal{G},\sqsupset} := (\mathit{Red}_{\mathrm{I}}^{\cap\mathcal{G},\sqsupset}, \mathit{Red}_{\mathrm{F}}^{\cap\mathcal{G},\sqsupset}) := \left(\bigcap_{q\in Q} \mathit{Red}_{\mathrm{I}}^{q,\mathcal{G}^q}, \bigcap_{q\in Q} \mathit{Red}_{\mathrm{F}}^{q,\mathcal{G}^q,\sqsupset}\right)$$

are redundancy criteria for $\models^{\cap}_{\mathcal{G}} := \bigcap_{q\in Q} \models^q_{\mathcal{G}^q}$ and *FInf*.

Intersections of Liftings. The above results can be extended in a straightforward way to intersections of lifted redundancy criteria. As before, let $\mathbf{F}$ and $\mathbf{G}$ be two sets of formulas, and let *FInf* be an $\mathbf{F}$-inference system. In addition, let Q be a set. For every $q \in Q$, let $\models^q$ be a consequence relation over $\mathbf{G}$, let GInf^q be a $\mathbf{G}$-inference system, let Red^q be a redundancy criterion for $\models^q$ and GInf^q, and let $\mathcal{G}^q$ be a grounding function from $\mathbf{F}$ and *FInf* to $\mathbf{G}$ and GInf^q. Let $\sqsupset$ be a well-founded strict partial ordering on $\mathbf{F}$.

For each $q \in Q$, we know by Theorem 6 that the $(\mathcal{G}^q, \emptyset)$-lifting $\mathit{Red}^{q,\mathcal{G}^q,\emptyset} = (\mathit{Red}_{\mathrm{I}}^{q,\mathcal{G}^q}, \mathit{Red}_{\mathrm{F}}^{q,\mathcal{G}^q,\emptyset})$ and the $(\mathcal{G}^q, \sqsupset)$-lifting $\mathit{Red}^{q,\mathcal{G}^q,\sqsupset} = (\mathit{Red}_{\mathrm{I}}^{q,\mathcal{G}^q}, \mathit{Red}_{\mathrm{F}}^{q,\mathcal{G}^q,\sqsupset})$ of Red^q are redundancy criteria for $\models^q_{\mathcal{G}^q}$ and *FInf*. Consequently, the intersections

$$\mathit{Red}^{\cap\mathcal{G},\sqsupset} := (\mathit{Red}_{\mathrm{I}}^{\cap\mathcal{G},\sqsupset}, \mathit{Red}_{\mathrm{F}}^{\cap\mathcal{G},\sqsupset}) := \left(\bigcap_{q\in Q} \mathit{Red}_{\mathrm{I}}^{q,\mathcal{G}^q}, \bigcap_{q\in Q} \mathit{Red}_{\mathrm{F}}^{q,\mathcal{G}^q,\emptyset}\right) \text{ and}$$
$$\mathit{Red}^{\cap\mathcal{G},\sqsupset} := (\mathit{Red}_{\mathrm{I}}^{\cap\mathcal{G},\sqsupset}, \mathit{Red}_{\mathrm{F}}^{\cap\mathcal{G},\sqsupset}) := \left(\bigcap_{q\in Q} \mathit{Red}_{\mathrm{I}}^{q,\mathcal{G}^q}, \bigcap_{q\in Q} \mathit{Red}_{\mathrm{F}}^{q,\mathcal{G}^q,\sqsupset}\right)$$

are redundancy criteria for $\models^{\cap}_{\mathcal{G}} := \bigcap_{q\in Q} \models^q_{\mathcal{G}^q}$ and *FInf*.

Intersections of Liftings. The above results can be extended in a straightforward way to intersections of lifted redundancy criteria. As before, let $\mathbf{F}$ and $\mathbf{G}$ be two sets of formulas, and let *FInf* be an $\mathbf{F}$-inference system. In addition, let Q be a set. For every $q \in Q$, let $\models^q$ be a consequence relation over $\mathbf{G}$, let GInf^q be a $\mathbf{G}$-inference system, let Red^q be a redundancy criterion for $\models^q$ and GInf^q, and let $\mathcal{G}^q$ be a grounding function from $\mathbf{F}$ and *FInf* to $\mathbf{G}$ and GInf^q. Let $\sqsupset$ be a well-founded strict partial ordering on $\mathbf{F}$.

For each $q \in Q$, we know by Theorem 6 that the $(\mathcal{G}^q, \emptyset)$-lifting $\mathit{Red}^{q,\mathcal{G}^q,\emptyset} = (\mathit{Red}_{\mathrm{I}}^{q,\mathcal{G}^q}, \mathit{Red}_{\mathrm{F}}^{q,\mathcal{G}^q,\emptyset})$ and the $(\mathcal{G}^q, \sqsupset)$-lifting $\mathit{Red}^{q,\mathcal{G}^q,\sqsupset} = (\mathit{Red}_{\mathrm{I}}^{q,\mathcal{G}^q}, \mathit{Red}_{\mathrm{F}}^{q,\mathcal{G}^q,\sqsupset})$ of Red^q are redundancy criteria for $\models^q_{\mathcal{G}^q}$ and *FInf*. Consequently, the intersections

$$\mathit{Red}^{\cap\mathcal{G},\sqsupset} := (\mathit{Red}_{\mathrm{I}}^{\cap\mathcal{G},\sqsupset}, \mathit{Red}_{\mathrm{F}}^{\cap\mathcal{G},\sqsupset}) := \left(\bigcap_{q\in Q} \mathit{Red}_{\mathrm{I}}^{q,\mathcal{G}^q}, \bigcap_{q\in Q} \mathit{Red}_{\mathrm{F}}^{q,\mathcal{G}^q,\emptyset}\right) \text{ and}$$
$$\mathit{Red}^{\cap\mathcal{G},\sqsupset} := (\mathit{Red}_{\mathrm{I}}^{\cap\mathcal{G},\sqsupset}, \mathit{Red}_{\mathrm{F}}^{\cap\mathcal{G},\sqsupset}) := \left(\bigcap_{q\in Q} \mathit{Red}_{\mathrm{I}}^{q,\mathcal{G}^q}, \bigcap_{q\in Q} \mathit{Red}_{\mathrm{F}}^{q,\mathcal{G}^q,\sqsupset}\right)$$

are redundancy criteria for $\models^{\cap}_{\mathcal{G}} := \bigcap_{q\in Q} \models^q_{\mathcal{G}^q}$ and *FInf*.

Figure 13: Top: authors' manuscript; middle: proof #1; bottom: proof #2. The vertical space above and below the formula should be the same, and the closing parenthesis should not overlap the preceding formula. Furthermore, in proof #2, the interword spacing is just weird (in particular in lines 2 and 4) and the final linebreak before "**F**" is hideous. [3]

Missing and Spurious Spaces and Symbols

Spaces and symbols are often inserted or deleted randomly. Fig. 14, 16, and 15 show some examples.

Standard Lifting. Given two sets of formulas $\mathbf{F}$ and $\mathbf{G}$, an $\mathbf{F}$-inference system $FInf$, a $\mathbf{G}$-inference system $GInf$, and a redundancy criterion Red for $GInf$, let $\mathcal{G}$ be a function that maps every formula in $\mathbf{F}$ to a subset of $\mathbf{G}$ and every $\mathbf{F}$-inference

Standard Lifting. Given two sets of formulas $\mathbf{F}$ and $\mathbf{G}$, an $\mathbf{F}$-inference system $FInf$, a $\mathbf{G}$-inference system $GInf$, and a redundancy criterion Red for $GInf$,let $\mathcal{G}$ be a function that maps every formula in $\mathbf{F}$ to a subset of $\mathbf{G}$ and every $\mathbf{F}$-inference

Figure 14: Top: authors' manuscript; bottom: proof #1. The space between "*GInf*," and "let" has been deleted. [3]

Although unfailing completion predates the introduction of Bachmair–Ganzinger-style redundancy, it can be incorporated into that framework by defining that formulas (i.e., rewrite rules and equations) and inferences (i.e., orientation and critical pair computation) are redundant if for every rewrite proof using that rewrite rule, equation, or critical peak, there exists a smaller rewrite proof. The requirement that the redundancy criterion must be obtained by lifting (which is necessary to introduce the labeling) can then be trivially fulfilled by "self-lifting"—i.e., by defining $\mathbf{G} := \mathbf{F}$ and $\mathrel{\dot\succ} := \emptyset$ and by taking $\mathcal{G}$ as the function that maps every formula or inference to the set of its α-renamings.

Although unfailing completion predates the introduction of Bachmair–Ganzinger-style redundancy, it can be incorporated into that framework by defining that formulas (i.e., rewrite rules and equations) and inferences (i.e., orientation and critical pair computation are redundant if for every rewrite proof using that rewrite rule, equation, or critical peak, there exists a smaller rewrite proof. The requirement that the redundancy criterion must be obtained by lifting (which is necessary to introduce the labeling) can then be trivially fulfilled by "self-lifting"— i.e., by defining $\mathbf{G} := \mathbf{F}$ and $\mathrel{\dot\succ} := \emptyset$ and by taking $\mathcal{G}$ as the function that maps every formula or inference to the set of its α-renamings.

Figure 15: Top: authors' manuscript; bottom: proof #1. The closing parenthesis after "computation" has been deleted and a space has been inserted after the dash. [3]

x [25]. It replaces fluid terms t by fresh variables z_t and maps type arguments to term arguments; thus, $[\lambda x{:}\iota.\,\lambda y{:}\iota.\,x] = \mathsf{lam}\,\iota\,(\mathsf{lam}\,\iota\,(\mathsf{db}_1\,\iota))$ and $[\mathsf{f}\langle\iota\rangle(y\,\mathsf{a})] = \mathsf{f}\,\iota\,z_{y\,\mathsf{a}}$.
bound variables x [25]. It replaces fluid terms t by fresh variables z_t and maps type arguments to term arguments; thus, $[\lambda x : \iota.\,\lambda ys : \iota.\,x] = \mathsf{lam}\,\iota\,(\mathsf{lam}\,\iota\,(\mathsf{db}_1\,\iota))$ and $[\mathsf{f}\langle\iota\rangle(y\,\mathsf{a})] = \mathsf{f}\,\iota\,z_{y\,\mathsf{a}}$. We then define the *metaorder* $\succ_{\mathsf{meta}}$ induced by $\succ_{\mathsf{base}}$ in

Figure 16: Top: authors' manuscript; bottom: proof #1. The variable "y" has been replaced by "ys". [2]

Misspelled Proper Names

Publishers love to replace bibliographic references in the manuscript by bibliographic references from their own database. Which would be fine if the entries in their own database were spelled correctly. In Fig. 17, they are not. (Note that this is an article in a Springer conference volume referencing an article in a Springer conference volume.)

[43] Kohlhase, M.: Higher-order tableaux. In: Baumgartner, P., Hähnle, R., Posegga, J. (eds.) TABLEAUX '95. LNCS, vol. 918, pp. 294–309. Springer (1995)
43. Kohlhase, M.: Higher-order tableaux. In: Baumgartner, P., Hähnle, R., Possega, J. (eds.) TABLEAUX 1995. LNCS, vol. 918, pp. 294–309. Springer, Heidelberg (1995). https://doi.org/10.1007/3-540-59338-1_43

Figure 17: Top: authors' manuscript; bottom: proof #1. The editor's name is Posegga, not Possega. [2]

Merged Tables

Fig. 18 shows a particularly weird edit. The manuscript contained two related tables, whose columns are not related, though. The copy editor decided to turn them into a single table. (Note that the caption talks about "The first table" and "The second table".)

Category	ARI	DAT	GEG	HWV	MSC	NUM	PUZ	SEV	SWV	SWW	SYN	SYO
Total	539	103	5	88	2	43	1	6	2	177	1	3
Solved	531	98	5	0	2	41	1	2	2	97	0	2

Rating	≥ 0.0	≥ 0.1	≥ 0.2	≥ 0.3	≥ 0.4	≥ 0.5	≥ 0.6	≥ 0.7	≥ 0.8	≥ 0.9	1.0
Total	972	853	771	527	391	343	253	180	129	97	97
Solved	781	666	584	340	210	162	85	29	12	2	2

Table 1: *Beagle* performance on the TPTP "theorem" or "unsatisfiable" problems. The first table breaks down the number of solved problems by category. The second table filters by problem rating. The column $\geq$ 0.6, for instance, means "all problems with a rating 0.6 or higher."

Table 1. *Beagle* performance on the TPTP "theorem" or "unsatisfiable" problems. The first table breaks down the number of solved problems by category. The second table filters by problem rating. The column $\geq$ 0.6, for instance, means "all problems with a rating 0.6 or higher".

Category	ARI	DAT	GEG	HWV	MSC	NUM	PUZ	SEV	SWV	SWW	SYN	SYO
Total	539	103	5	88	2	43	1	6	2	177	1	3
Solved	531	98	5	0	2	41	1	2	2	97	0	2
Rating	≥ 0.0	≥ 0.1	≥ 0.2	≥ 0.3	≥ 0.4	≥ 0.5	≥ 0.6	≥ 0.7	≥ 0.8	≥ 0.9	1.0	
Total	972	853	771	527	391	343	253	180	129	97	97	
Solved	781	666	584	340	210	162	85	29	12	2	2	

Figure 18: Top: authors' manuscript; bottom: proof #1. The two independent tables have been turned into a single one. Note that the columns of the two tables are not related. [1]

References

[1] Peter Baumgartner, Joshua Bax, and Uwe Waldmann. Beagle – a hierarchic superposition theorem prover. In Amy P. Felty and Aart Middeldorp, eds., *Automated Deduction – CADE-25*, Berlin, Germany, 2015, *LNAI 9195*, pp. 367–377. Springer.

[2] Alexander Bentkamp, Jasmin Christian Blanchette, Sophie Tourret, Petar Vukmirović, and Uwe Waldmann. Superposition with lambdas. In Pascal Fontaine, ed., *Automated Deduction – CADE 27*, Natal, Brazil, 2019, *LNAI 11716*, pp. 55–73. Springer.

[3] Uwe Waldmann, Sophie Tourret, Simon Robillard, and Jasmin Blanchette. A comprehensive framework for saturation theorem proving. In Nicolas Peltier and Viorica Sofronie-Stokkermans, eds., *Automated Reasoning (IJCAR 2020)*, Paris, France (Virtual Conference), 2020, *LNAI 12166*, pp. 316–334. Springer.

[4] Uwe Waldmann, Sophie Tourret, Simon Robillard, and Jasmin Blanchette. A comprehensive framework for saturation theorem proving. *Journal of Automated Reasoning*, 66:499–539, 2022.