NAMOR: a New Agda Library for Modal Extended Sequents

Riccardo Borsetto[1], Margherita Zorzi[1]

[1]University of Verona, Department of Computer Science, Strada le Grazie 15, 37134 Verona, Italy

Abstract

This paper introduces NAMOR (New Agda MOdal Realization), a comprehensive Agda library for mechanizing modal proof theory through *extended sequents*. Our approach employs position-based reasoning that mirrors first-order logic structures while maintaining implicit accessibility relations. Building on our previous investigation of extended deductive systems, NAMOR provides a unified framework for formalizing proof-theoretical properties across normal extensions of modal logic K, from basic K through S5. The library is founded on our novel sequent calculus E_{pos}, which exhibits remarkable modularity: all captured logics share identical inference rules, differing mainly in their modal constraints. This enables systematic verification of meta-theoretical properties while preserving the intuitive structure of paper-and-pencil proofs. By leveraging Agda's expressive type system, NAMOR establishes a direct correspondence between formal proofs and their mathematical counterparts, creating a robust platform for symbolic reasoning in domains where modal logic naturally applies.

Keywords

Agda proof assistant, modal logic, proof theory, sequent calculus, extended-sequents, cut elimination

1. Introduction

The recent renaissance in modal proof theory has introduced diverse deductive styles, including labelled systems [1, 2], 2-sequents [3], nested sequents [4, 5], and hyper-sequents [6]. This work focuses on *extended or extensible sequents*, which build upon the original formulation of 2-sequents [7]. The key ideas for extended sequents are as follows: formulas are marked with a "spatial coordinate" called *position*; rules do not contain explicit reference to the accessibility relation (a crucial difference from other approaches such as labeled frameworks); and only modal operators, treated analogously to first-order quantifiers, can "change" the position of formulas. These properties yield several benefits. First, one obtains "small" deductive systems (in terms of the number of rules) that allow reasoning not only on the metatheory of the captured logics but also in other settings where modal logic serves as a knowledge tool, such as Artificial Intelligence and Epistemology. Second, the transparent representation of the semantics allows it to be changed without altering the set of rules. This is useful for exploring semantics different from the usual Kripke semantics, such as the BHK interpretation in the intuitionistic setting, or for logics characterized by multiple different Kripke models (e.g., the intermediate modal logic S4.2 [8]). Finally, we show in a previous investigation that position-based deductive systems enjoy a very strong notion of *modularity*, which can be also imported into implementations: a wide spectrum of normal extensions of the basic modal logic K share the same set of rules and a specific system can be easily obtained simply by "tuning" some constraints on modal and cut rules (see[9]).

It is with these advantages in mind that we present NAMOR, a new Agda library for normal modal logics designed to be as close as possible to pen-and-paper theory. Its main contributions are:

- A new sequent calculus, E_{pos}, that provides a unified theoretical foundation for normal modal logics. Building on our previous work [9, 10], E_{pos} achieves a strong form of modularity by exploiting the similarity between the position-based interpretation of modalities and first-order

7th International Workshop on Artificial Intelligence and Formal Verification, Logic, Automata, and Synthesis (OVERLAY 2025), October 26, 2025, Bologna, Italy
riccardo.borsetto@univr.it (R. Borsetto); margherita.zorzi@univr.it (M. Zorzi)
0009-0001-4751-1516 (R. Borsetto); 0000-0002-3285-9827 (M. Zorzi)

logic. This allows a wide range of normal modal logics—from K to S5—to be captured using an identical set of deductive rules, distinguished only by their modal constraints.

- The implementation of this calculus in NAMOR, a new Agda library. We generalize the core ideas from our previous implementation for S4.2 [10] to the full E_{pos} calculus, creating a flexible and robust library for mechanizing modal proof theory across a wide spectrum of normal modal systems. Our implementation uses a deep embedding approach, where the syntax and inference rules of the calculus are faithfully represented within Agda's type system, enabling formal verification of meta-theoretical properties.
- The choice of Agda over other powerful proof assistants such as Coq, Lean, or Isabelle has multiple reasons. While the systems cited above are highly capable, they primarily rely on an imperative "proof script" style for writing proofs. In contrast, Agda's functional "proof term" approach and clean, Unicode-based syntax allow for formalized definitions and proofs that remain exceptionally close to their on-paper mathematical counterparts. This closeness, supported by Agda's interactive development features, is particularly valuable for a system like ours, where the manipulation of positions is central.

1.1. Synopsis

In Section 2 we present the sequent calculus E_{pos}, we show some sample of the Agdaimplementation and we provide an example of axiom derivation in NAMOR; Related work are discussed in Section 3; Section 4 is devoted to discussion and future work.

2. NAMOR: Implementing the sequent calculus E_{pos}

We first present the position-based sequent calculus E_{pos}. According to some constraints of the rules $\Box\vdash$, $\vdash\Diamond$ and Cut, E_{pos} captures several normal extensions of the logic K by incorporating the basic axiom $K \equiv \Box(A \rightarrow B) \rightarrow (\Box A \rightarrow \Box B)$ and one or more of the following axioms: $D \equiv \Box A \rightarrow \Diamond A$, $T \equiv \Box A \rightarrow A$, $4 \equiv \Box A \rightarrow \Box\Box A$, $.2 \equiv \Diamond\Box A \rightarrow \Box\Diamond A$ and $5 \equiv A \rightarrow \Diamond\Box A$. This results in K (containing only the basic axiom K), D (K+D), T (K+T), K4 (K+4), D4 (K+D+4), S4 (K+T+4), S4.2 (K+T+4+.2) and S5 (K+T+4+5). The system is inspired by our previous work [9, 11].

The language $\mathscr{L}$ consists of a countably infinite set of propositional symbols $p_0, p_1, \ldots$, propositional connectives $\wedge, \vee, \rightarrow, \neg, \bot$, modal operators $\Box, \Diamond$, and a denumerable set $\mathscr{T}$ of *tokens* (ranged over by x, y, z), which act as atomic world-identifiers. Modal formulas are defined inductively as usual.

A *position* α (ranged over by α, β, γ) is a finite sequence of tokens, i.e., an element of $\mathscr{T}^*$. We denote a sequence as $\langle x_1, \ldots, x_n\rangle$, with $\langle\,\rangle$ being the empty sequence. The (associative) concatenation of positions is defined as $\langle x_1, \ldots, x_n\rangle \circ \langle z_1, \ldots, z_m\rangle = \langle x_1, \ldots, x_n, z_1, \ldots, z_m\rangle$. On positions, we define the successor relation $s \triangleleft t \Leftrightarrow \exists x \in \mathscr{T}.\ t = s \circ \langle x\rangle$, and its reflexive ($\triangleleft^0$), transitive ($\sqsubset$), and reflexive-transitive ($\sqsubseteq$) closures.

A *position-formula* (or *p-formula*) is an expression A^α, where A is a modal formula and α is a position. An *extended sequent* (or *e-sequent*) is an expression $\Gamma \vdash \Delta$, where Γ and Δ are finite sequences of p-formulas.

The rules of E_{pos} are in Figure 1. Modal rules represent the core of the system and are common to all the logics. The system is designed as much as possible in analogy with first-order logic quantifiers as suggested by the constraints on $\vdash\Box$ and $\Diamond\vdash$, that are reminiscent of the analogous for $\forall$ and $\exists$: in the rules $\vdash\Box$ and $\Diamond\vdash$, one has $\alpha \circ \langle x\rangle \notin \mathfrak{Init}[\Gamma, \Delta]$, where Γ and Δ are the lists of (open) assumptions on which $A^{\alpha\circ\langle x\rangle}$ depends and $\mathfrak{Init}[\Sigma] = \{\beta : \exists A^\alpha \in \Sigma.\ \beta \sqsubseteq \alpha\}$.

As previously said, all logics in the normal spectrum from K to S5 share the rules above; a specific system is obtained simply by "tuning" the constraints on the $\Box\vdash$, $\vdash\Diamond$ and on the cut rule for systems K and K4. Constraints on modal and cut rules are in Table 1 and Table 2, respectively.

Axiom and Cut Rule

$$\dfrac{}{A^\alpha \vdash A^\alpha}\,Ax \qquad \dfrac{\Gamma_1 \vdash A^\alpha, \Delta_1 \qquad \Gamma_2, A^\alpha \vdash \Delta_2}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2}\,Cut$$

Structural Rules

$$\dfrac{\Gamma \vdash \Delta}{\Gamma, A^\alpha \vdash \Delta}\,W\vdash \qquad \dfrac{\Gamma \vdash \Delta}{\Gamma \vdash A^\alpha, \Delta}\,\vdash W$$

$$\dfrac{\Gamma, A^\alpha, A^\alpha \vdash \Delta}{\Gamma, A^\alpha \vdash \Delta}\,C\vdash \qquad \dfrac{\Gamma \vdash A^\alpha, A^\alpha, \Delta}{\Gamma \vdash A^\alpha, \Delta}\,\vdash C$$

$$\dfrac{\Gamma_1, A^\alpha, B^\beta, \Gamma_2 \vdash \Delta}{\Gamma_1, B^\beta, A^\alpha, \Gamma_2 \vdash \Delta}\,Exc\vdash \qquad \dfrac{\Gamma \vdash \Delta_1, A^\alpha, B^\beta, \Delta_2}{\Gamma \vdash \Delta_1, B^\beta, A^\alpha, \Delta_2}\,\vdash Exc$$

Propositional Rules

$$\dfrac{\Gamma, A^\alpha \vdash \Delta}{\Gamma, A \wedge B^\alpha \vdash \Delta}\,\wedge_1\vdash \qquad \dfrac{\Gamma, B^\alpha \vdash \Delta}{\Gamma, A \wedge B^\alpha \vdash \Delta}\,\wedge_2\vdash \qquad \dfrac{\Gamma_1 \vdash A^\alpha, \Delta_1 \qquad \Gamma_2 \vdash B^\alpha, \Delta_2}{\Gamma_1, \Gamma_2 \vdash A \wedge B^\alpha, \Delta_1, \Delta_2}\,\vdash\wedge$$

$$\dfrac{\Gamma_1, A^\alpha \vdash \Delta_1 \qquad \Gamma_2, B^\alpha \vdash \Delta_2}{\Gamma_1, \Gamma_2, A \vee B^\alpha \vdash \Delta_1, \Delta_2}\,\vee\vdash \qquad \dfrac{\Gamma \vdash A^\alpha, \Delta}{\Gamma \vdash A \vee B^\alpha, \Delta}\,\vdash\vee_1 \qquad \dfrac{\Gamma \vdash B^\alpha, \Delta}{\Gamma \vdash A \vee B^\alpha, \Delta}\,\vdash\vee_2$$

$$\dfrac{\Gamma_1 \vdash A^\alpha, \Delta_1 \qquad \Gamma_2, B^\alpha \vdash \Delta_2}{\Gamma_1, \Gamma_2, A \to B^\alpha \vdash \Delta_1, \Delta_2}\,\to\vdash \qquad \dfrac{\Gamma, A^\alpha \vdash B^\alpha, \Delta}{\Gamma \vdash A \to B^\alpha, \Delta}\,\vdash\to$$

Modal Rules

$$\dfrac{\Gamma, A^\beta \vdash \Delta}{\Gamma, \Box A^\alpha \vdash \Delta}\,\Box\vdash \qquad \dfrac{\Gamma \vdash A^{\alpha\circ\langle x\rangle}, \Delta}{\Gamma \vdash \Box A^\alpha, \Delta}\,\vdash\Box$$

$$\dfrac{\Gamma, A^{\alpha\circ\langle x\rangle} \vdash \Delta}{\Gamma, \Diamond A^\alpha \vdash \Delta}\,\Diamond\vdash \qquad \dfrac{\Gamma \vdash A^\beta, \Delta}{\Gamma \vdash \Diamond A^\alpha, \Delta}\,\vdash\Diamond$$

Figure 1: The sequent calculus E_{pos}

Calculus	**Constraints on the rules $\Box\vdash$ and $\vdash\Diamond$**
E_{S5}	no constraints
$E_{S4.2}$	$\alpha \subseteq \beta$
E_{S4}	$\alpha \sqsubseteq \beta$
E_T	$\alpha \lhd^0 \beta$
E_D	$\alpha \lhd \beta$
E_{D4}	$\alpha \sqsubset \beta$
E_{K4}	$\alpha \sqsubset \beta$ there is at least a formula $B^{\alpha\circ\beta\circ\eta}$ in either Γ or Δ
E_K	$\alpha \lhd \beta$ there is at least a formula $B^{\alpha\circ\beta\circ\eta}$ in either Γ or Δ

Table 1: Constraints for modal rules $\Box\vdash$ and $\vdash\Diamond$

	Constraints on the cut rule
E_D, E_T, E_{D4}, E_{S4}, $E_{S4.2}$, E_{S5}	no constraints
E_K, E_{K4}	$\alpha \in \mathfrak{Init}[\Gamma_1, \Delta_1 - A^\alpha]$ or $\alpha \in \mathfrak{Init}[\Gamma_2 - A^\alpha, \Delta_2]$

Table 2: Constraints for cut rule

We reflect the modularity in the Agda implementation:

```
□⊢ : ∀ {M A α β Γ Δ}
    → modalConstraint M α β Γ Δ
    → Proof (Γ ,, [ A ^ β ] ⊢ Δ)
    → Proof (Γ ,, [ □ A ^ α ] ⊢ Δ)

⊢◊ : ∀ {M A α β Γ Δ}
    → modalConstraint M α β Γ Δ
    → Proof (Γ ⊢ [ A ^ β ] ,, Δ)
    → Proof (Γ ⊢ [ ◊ A ^ α ] ,, Δ)
```

Following Table 1, the specific constraint for each logic is implemented in the proof-assistant by the **modalConstraints** function, as follows:

```
modalConstraint : Logic → position → position → List pf → List pf → Type
modalConstraint S5 α β Γ Δ  = Unit
modalConstraint S4dot2 α β Γ Δ = α ⊆ β
modalConstraint S4 α β Γ Δ  = α ⊑ β
modalConstraint T α β Γ Δ   = α ◃⁰ β
modalConstraint D α β Γ Δ   = α ◃ β
modalConstraint D4 α β Γ Δ = α ⊏ β
modalConstraint K4 α β Γ Δ = α ⊏ β × (Γ ,, Δ) has β
modalConstraint K α β Γ Δ   = α ◃ β × (Γ ,, Δ) has β
```

Finally, following Table 2, constraints for the cut- rule are implemented in Agda as

```
cutConstraint : Logic → mf → position → List pf → List pf → List pf → List pf → Type
cutConstraint K A α Γ₁ Γ₂ Δ₁ Δ₂ = (α ∈ Init (Γ₁ ,, (Δ₁ - A ^ α))) ⊎ (α ∈ Init ((Γ₂ - A ^ α) ,, Δ₂))
cutConstraint K4 A α Γ₁ Γ₂ Δ₁ Δ₂ = (α ∈ Init (Γ₁ ,, (Δ₁ - A ^ α))) ⊎ (α ∈ Init ((Γ₂ - A ^ α) ,, Δ₂))
cutConstraint _ A α Γ₁ Γ₂ Δ₁ Δ₂ = Unit
```

In the next section we propose an example to show how both the constraints and the implementation work.

2.1. Working example: deriving the Geach axiom (twice)

To show how the framework E_{pos} works, we propose two derivations of the Geach axiom, also called .2: $\Diamond\Box A \rightarrow \Box\Diamond A$. The Geach axiom is characteristic for the logic S4.2, an intermediate system between S4 and S5, that is particulary interesting and enjoy several applications both in mathematics and in computer science.

We first try to derive the axiom in the wrong system S4, whose constraint, correctly, does not allow the derivation. Then we show the correct derivation in the system S4.2.

Example 1 (Derivation of the Geach axiom)**.**

Let's start observing what happens if we try to derive .2 by setting NAMOR to logic S4*:*

```
axiomG-S4-FAILED : ∀ {A} → (Proof ([] ⊢ [((◊ (□ A)) ⇒ (□ (◊ A)) ^ [])]))
axiomG-S4-FAILED =
  ⊢⇒
  (◊⊢ {x = "x"} {Γ = []} (λ { (here ()) ; (there ()) })
  (⊢□ {x = "y"} (λ { (there (here ())) ; (there (there ())) })
  (⊢◊ {M = S4} {α = ⟨ "y" ⟩} {β = ⟨ "x" ⟩ ∘ ⟨ "y" ⟩} {!  !} {!  !}))) -- FAILURE
```

The logic S4 has constraint

$$\textit{modalConstraint}\ S4\ \alpha\ \beta\ \Gamma\ \Delta\ =\ \alpha\ \sqsubseteq\ \beta$$

so for the $\vdash\Diamond$ *rule we need* $\langle\ "y"\rangle \sqsubseteq (\langle\ "x"\rangle \circ \langle\ "y"\rangle)$. *The meaning is that we are requiring* $\langle\ "y"\rangle$ *is a prefix of* $\langle\ "x",\ "y"\rangle$, *but it is not, so the constraint is not satisfied and the derivation fails.*

One might wonder if choosing different tokens or swapping "x" and "y" could make the derivation work in S4. *However, the failure is* structural, *not dependent on variable names. The modal rules force a specific order of token introduction: the* $\Diamond$ *elimination introduces "x" first, then the* $\Box$ *introduction introduces "y". The resulting constraint always requires the second token to be a prefix of the concatenated position, which is impossible under the prefix relation regardless of the chosen variable names.*

For the second derivation we set the constraints to capture S4.2 *and then we have:*

```
axiomG : ∀ {A} → (Proof ([] ⊢ [((◇ (□ A)) ⇒ (□ (◇ A)) ^ [])]))
axiomG =
  ⊢⇒
  (◇⊢ {x = "x"} {Γ = []} (λ { (here ()) ; (there ()) })
  (⊢□ {x = "y"} (λ { (there (here ())) ; (there (there ())) })
  (⊢◇ {M = S4dot2} {α = ⟨ "y"⟩} {β = ⟨ "x"⟩ ∘ ⟨ "y"⟩} (λ {x} → there)
  (□⊢ {M = S4dot2} {α = ⟨ "x"⟩} {β = ⟨ "x"⟩ ∘ ⟨ "y"⟩} {Γ = []} (xs⊆xs++ys ⟨ "x" ⟩ ⟨ "y" ⟩)
  Ax ))))
```

The logic S4.2 has constraint:

$$\textit{modalConstraint}\ S4.2\ \alpha\ \beta\ \Gamma\ \Delta\ =\ \alpha\ \subseteq\ \beta$$

For the $\vdash\Diamond$ *rule we need to specify* $\langle\ "y"\rangle \subseteq (\langle\ "x"\rangle \circ \langle\ "y"\rangle)$. *This constraint is satisfied by* $(\lambda\ \{x\} \to$ *there*) *that shows "y" is at position there in* ["x", "y"]. *For the* $\Box\vdash$ *rule we need* $\langle\ "x"\rangle \subseteq (\langle\ "x"\rangle \circ \langle\ "y"\rangle)$. *This constraint is satisfied by* xs⊆xs++ys *that shows* ["x"] ⊆ ["x"] ++ ["y"]. *So the derivation succeeds by applying the axiom rule.*

We spend some words about here, there, and freshness patterns. In Agda, proofs of list membership are constructed with here (for the head) and there (for the tail). We use these to enforce freshness constraints *in modal rules, which is analogous to the eigen-variable conditions in traditional sequent calculi.*

For instance, to prove a token is fresh for an empty context, we must show it cannot be a member. We do this with a function that uses Agda's absurd pattern *() to demonstrate that both cases—the token being here* or *there—are impossible. This pattern-matching against an absurd case is a standard Agda idiom for proving negative properties, ensuring that new tokens are genuinely fresh.*

The success of S4.2 lies in the flexibility of the subset relation (⊆) compared to the prefix relation. While positions in S4 must be ordered as prefixes (reflecting the transitivity of accessibility), S4.2 allows positions to be interpreted as sets, where subset inclusion is more permissive.

3. Related work

The automation of modal reasoning is receiving increasing attention from the research community. In this context, it is worth mentioning recent developments in the HOL Light proof assistant by Maggesi et al., which are closely aligned with our work with partially different motivations. In [12], the authors present a formalized proof of modal completeness for Gödel–Löb provability logic (GL) within HOL Light in a labelled setting [1]. This work is further extended towards a full mechanization of GL in [13]. Building upon these contributions, the approach has been generalized in the HOLMS framework to encompass a broader class of normal modal logics [14]. This project is a key reference, as we are also developing a modular library for normal extensions of K, albeit with a different approach to modal proof theory.

Our motivations are complementary. While HOLMS moves towards automated deduction, we position our work at the level of meta-theoretical formalization. This is reflected in our choice of a classical, LK-style calculus, which is well-suited for studying meta-properties, in contrast to the G3-style systems (see, e.g., [15, 16]) often preferred for automated proof search due to their invertible rules. Our short-term goal is not automated deduction in a strict algorithmic sense; rather, we aim to provide a correct and robust implementation of our theoretical results. We exploit the full expressivity of Agda to formally verify properties of our meta-theory and plan to use NAMOR as a reasoning tool for applications in modal logic semantics (see Section 4).

4. Conclusions and ongoing work

The development of NAMOR is an ongoing project that opens several directions for future work. Our current goal is to formally verify the Cut-Elimination Theorem, a central results in proof-theory, for the calculus E_{pos}. Following our approach in [9, 10], the modularity of the system will allow us to prove this fundamental property once for the entire spectrum of captured logics, from K to S5.

Beyond this foundational work, our main short-term goal is to develop and implement semantics for specific instances of E_{pos}. We are particularly focused on S4.2 [11], a system with applications in AI, epistemology [8, 10], and mathematics, including as the logic of abelian groups [17] and of set-theoretic forcing [18]. The position-based nature of our framework is a key advantage here, as it allows us to tailor the semantics to the desired interpretation.

Finally, we plan to explore the computational interpretations of our work. The logic S5, for instance, has deep connections to computation [19]. Following the work in [20], we aim to implement the Brouwer-Heyting-Kolmogorov (BHK) semantics for the intuitionistic version of our S5 calculus. This will allow us, via a Curry-Howard correspondence, to study the resulting dependent type calculus.

Declaration on Generative AI

During the preparation of this work, the authors used Gemini 2.5 Pro in order to: Grammar and spelling check, Paraphrase and reword. After using this tool/service, the authors reviewed and edited the content as needed and takes full responsibility for the publication's content.

References

[1] S. Negri, Proof theory for modal logic, Philosophy Compass 6 (2011) 523–538. doi:10.1111/j.1747-9991.2011.00418.x.

[2] R. Dyckhoff, S. Negri, Admissibility and cut-elimination for modal logics, Journal of Logic and Computation (2012). doi:10.1007/s00153-011-0254-7.

[3] A. Masini, 2-sequent calculus: intuitionism and natural deduction, Journal of Logic and Computation 3 (1993) 533–562. doi:10.1093/logcom/3.5.533.

[4] M. Fitting, Prefixed tableaus and nested sequents, Annals of Pure and Applied Logic 163 (2012) 291–313. doi:10.1016/j.apal.2011.09.004.

[5] B. Lellmann, E. Pimentel, Modularisation of sequent calculi for normal and non-normal modalities, ACM Transactions on Computational Logic 20 (2019) 7:1–7:46. doi:10.1145/3288757.

[6] A. Ciabattoni, R. Ramanayake, H. Wansing, Hypersequent and display calculi—a unified perspective, Studia Logica 102 (2014) 1245–1294. doi:10.1007/s11225-014-9566-z.

[7] A. Masini, 2-sequent calculus: A proof theory of modalities, Annals of Pure and Applied Logic 58 (1992) 229–246. URL: https://www.sciencedirect.com/science/article/pii/016800729290029Y. doi:10.1016/0168-0072(92)90029-Y.

[8] A. Chalki, C. D. Koutras, Y. Zikos, A quick guided tour to the modal logic S4.2, Logic Journal of the IGPL 26 (2018) 429–451. URL: https://doi.org/10.1093/jigpal/jzy008. doi:10.1093/jigpal/jzy008. arXiv:https://academic.oup.com/jigpal/article-pdf/26/4/429/25207558/jzy008.pdf.

[9] S. Martini, A. Masini, M. Zorzi, Cut elimination for extended sequents, Bulletin of the Section of Logic. Published online: August 15, 2023; 36 pages (2023). doi:https://doi.org/10.18778/0138-0680.2023.22.

[10] R. Borsetto, M. Zorzi, An agda implementation of the modal logic s4.2: first investigations, preprint, submitted (2025). URL: https://drive.google.com/file/d/1dNdyJbGW7J5ImsyuqaqqGDnGnwUv6Qa_/view?usp=sharing.

[11] R. Borsetto, M. Zorzi, Namor: a new agda library for modal extended sequents, in: ICTCS 2025, 20th Italian Conference on Theoretical Computer Science, October 26th, Bologna, Italy, CEUR Conference Proceedings, to appear, 2025.

[12] M. Maggesi, C. Perini Brogi, A Formal Proof of Modal Completeness for Provability Logic, in: L. Cohen, C. Kaliszyk (Eds.), 12th International Conference on Interactive Theorem Proving (ITP 2021), volume 193 of *Leibniz International Proceedings in Informatics (LIPIcs)*, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2021, pp. 26:1–26:18. URL: https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.ITP.2021.26. doi:10.4230/LIPIcs.ITP.2021.26.

[13] M. Maggesi, C. Perini Brogi, Mechanising gödel-löb provability logic in HOL light, Journal of Automated Reasoning 67 (2023) 29. doi:10.1007/S10817-023-09677-Z.

[14] A. Bilotta, M. Maggesi, C. Perini Brogi, Growing holms, a hol light library for modal systems, in: OVERLAY 2024, 6th International Workshop on Artificial Intelligence and Formal Verification, Logic, Automata, and Synthesis, November 28–29, 2024, Bolzano, Italy, volume 3904 of *CEUR Conference Proceedings*, 2024, pp. 26:1–26:18. URL: https://ceur-ws.org/Vol-3904/paper5.pdf.

[15] S. Negri, J. von Plato, A. Ranta, Structural Proof Theory, Cambridge University Press, 2001. doi:10.1017/CBO978113923619.

[16] G. Fellin, S. Negri, E. Orlandelli, Glivenko sequent classes and constructive cut elimination in geometric logics, Archive for Mathematical Logic 62 (2023) 657–688. doi:10.1007/s00153-022-0857-z.

[17] S. Berger, A. C. Block, B. Löwe, The modal logic of abelian groups, Algebra Universalis 84 (2023) 25. URL: https://doi.org/10.1007/s00012-023-00821-9. doi:10.1007/s00012-023-00821-9.

[18] J. D. Hamkins, B. Loewe, The modal logic of forcing, Transactions of the American Mathematical Society 360 (2008) 1793–1817. doi:10.1305/ndjfl/1093894722.

[19] T. Murphy, K. Crary, R. Harper, F. Pfenning, A symmetric modal lambda calculus for distributed computing, in: Proceedings of the 19th Annual IEEE Symposium on Logic in Computer Science, 2004., 2004, pp. 286–295. doi:10.1109/LICS.2004.1319623.

[20] S. Guerrini, A. Masini, M. Zorzi, Natural deduction calculi for classical and intuitionistic S5, Journal of Applied Non-Classical Logics 33 (2023) 165–205. doi:10.1080/11663081.2023.2233750.