An Agda Implementation of the Modal Logic S4.2: First Investigations

Riccardo Borsetto[1], Margherita Zorzi[1]

[1]Università di Verona, Dipartimento di Informatica, Strada le Grazie 15, 37134 Verona, Italy

Abstract

This paper is the first step towards a new mechanization of modal logics strongly oriented to constructive mathematics. We introduce $E_{S4.2}$, a sequent calculus for the logic S4.2, that extends S4 with the axiom $\Diamond\Box\phi \rightarrow \Box\Diamond\phi$ and emerges as the underlying logic in different fields. We build on previous investigations where formulas are equipped with a *position*, i.e. set of uninterpreted tokens used to manage modal information. The calculus is designed to enjoy strong proof-theoretical properties, such as a direct syntactical proof of cut-elimination, which in turn yields the consistency of the system and the subformula property. We implement in Agda the system and we present the implementation of the proof of the Cut-Elimination Theorem as a work in progress.

Keywords

Proof theory, sequent calculus, cut-elimination, modal logic, 2-sequents

1. Introduction

The modal logic S4.2 holds a significant position in the landscape of non-classical logics. It is axiomatically defined by extending the well-known system S4 (which includes axioms K, T, and 4) with the Geach axiom G (also denoted as .2): $\Diamond\Box\phi \rightarrow \Box\Diamond\phi$. Semantically, S4.2 is characterized by reflexive and transitive Kripke frames that satisfy a confluence (or directedness) property, such as being weakly directed. The interest in S4.2 stems from its role as the underlying logic in a variety of distinct mathematical and philosophical domains. In a seminal work by Goldblatt [1], the Diodorean interpretation of modality allows for modeling time using four-dimensional Minkowski spacetime, and modal sentences valid in this structure are shown to be exactly the theorems of S4.2. A suitable interpretation of the box ($\Box$) and diamond ($\Diamond$) modalities demonstrates that S4.2 is the logic of the forcing technique in axiomatic set theory ZFC [2] and also applies to abstract algebra as the modal logic of abelian groups [3]. Moreover, within the context of potential infinity, S4.2 offers an interesting perspective on convergent expansions of infinite collections [4, 5].

Last but not least, S4.2 has been advocated by many philosophers and epistemologists as *the* correct logic of knowledge [6, 7, 8, 9]. This suggests that S4.2 is a promising formal system for integration into the emerging field connecting knowledge representation and Explainable Artificial Intelligence, especially in contexts involving non-monotonic cognitive agents.

The versatility of S4.2 highlights the importance of developing robust and well-behaved proof systems, which can facilitate more effective reasoning within these contexts. In terms of proof theory, this corresponds to developing deductive systems that are as simple as possible and enjoy strong structural properties such as analyticity and cut-elimination [10].

This paper studies the theory and presents the implementation in the Agda proof assistant [11] of $E_{S4.2}$, a sequent calculus for S4.2 designed with the aforementioned proof-theoretic desiderata in mind. Our approach builds upon the general framework of extended sequent calculi, which have been successfully developed for classical modal logics (such as K, D, T, and S4) to provide modular systems and direct syntactical proofs of cut-elimination [12]. We list the main features of our system: i) formulas are marked by a *position*, i.e., a set of uninterpreted tokens (in [12], positions are modeled as lists); ii) the right rule for $\Box$, and its dual left rule for $\Diamond$, are formulated using constraints on positions, drawing

ICTCS 2025: Italian Conference on Theoretical Computer Science, September 10–12, 2025, Pescara, 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)

a strong analogy with the eigenvariable conditions of the right ∀ rule (and ∃ left rule, respectively) in first-order logic; iii) the accessibility relation is not formalized directly; iv) only modal operators can change the positions of formulas.

Developing $\mathsf{E}_{\mathsf{S4.2}}$ as a position-based sequent calculus is motivated by the goal of a full formalization of the system and its meta-properties in the Agda proof assistant. The use of positions makes proofs more manageable, as the representation of the underlying Kripke semantics is transparent. Leveraging [12], we exploit the fact that the proof-theoretical distance between classical logic and the system under consideration is very small. The choice of Agda over other powerful assistants such as Coq, Lean, or Isabelle is based on its specific advantages for our approach. While these systems are highly capable, they primarily rely on an imperative "proof script" style. In contrast, Agda's functional "proof term" approach, combined with its clean, Unicode-based syntax, allows formalized definitions and proofs to remain remarkably 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. Furthermore, Agda's foundation in constructive type theory aligns naturally with the proof-theoretic focus of our work and facilitates future extensions, such as developing an intuitionistic version of S4.2. This represents an alternative approach to systems like Lean, which have a stronger focus on classical mathematics.

The main contribution of this work is the presentation of the $\mathsf{E}_{\mathsf{S4.2}}$ calculus and its ongoing formalization in Agda, including the syntax and a proof of the Cut-Elimination Theorem. As a corollary, we obtain the subformula property and a syntactical proof of the consistency of the system. We also discuss our ongoing work and future research directions.

Related work The proof theory of S4.2 has been previously investigated in several papers, which differ methodologically from the approach we propose. In [13, 14], S4.2 is analysed through labelled natural deduction systems. Sequent calculi for S4.2 have also been explored in [15], where, within a labelled framework S4.2 is characterized as the *modal companion* of Jan-De Morgan logic, and in [16], where a restricted form of the cut rule is introduced in order to obtain a subformula property that enables significant corollaries, such as the interpolation property.

On the implementation side, in [17, 18], the authors present a mechanization of GL in [18]. Building upon these contributions, the approach has been generalized in the HOLMS framework to encompass a broader class of normal modal logics [19]. This project constitutes a closely related reference, as our proposed implementation is part of an ongoing effort to develop a modular implementation of all normal extensions of K up to S5 (see Section 4).

Outline of the paper. The paper is structured as follows: in Section 2 we present syntax and rules of $\mathsf{E}_{\mathsf{S4.2}}$; Section 3 is dedicated to the Cut-Elimination Theorem and part of the Agda implementation is shown; Section 4 discusses our work in progress and future directions of the investigation.

2. The sequent calculus $\mathsf{E}_{\mathsf{S4.2}}$

The language $\mathscr{L}$ consists of a countably infinite set of propositional symbols $p_0, p_1, \dots$, propositional connectives $\wedge, \vee, \rightarrow, \neg, \bot$, modal operators $\Box, \Diamond$, and auxiliary symbols (,).

Our main syntactical objects are *position-formulas* (p-formulas), expressions of the form A^s, where A is a modal formula and s is a *position* – a finite set of uninterpreted syntactic objects called *tokens*, denoted by meta-variables x, y, z, possibly indexed. We use $\mathscr{T}$ to denote a denumerable set of tokens. Meta-variables s, t, u range over positions. The use of *sets* of tokens for positions is a key feature of $\mathsf{E}_{\mathsf{S4.2}}$, distinguishing it from earlier extended sequent calculi that typically employed *lists* of tokens [12]. This choice is motivated by recent developments in natural deduction for S4.2 [20] and aligns naturally with its semantics based on semilattices with a minimum. The set-based nature allows positions to represent collections of modal constraints or "worlds" without an artificial order, which is crucial for capturing the confluence property inherent in S4.2. The union operation ($s \cup t$ or $s \cup \{x\}$) on positions reflects the "merging" of modal information.

A token is implemented in the proof assistant as a term of type Fin n. In Agda's standard library [21], Fin n represents the finite set of natural numbers $\{0, 1, ..., n-1\}$, where the parameter $n : \mathbb{N}$ determines the size of the set. This choice provides a concrete, finite, and enumerable set of syntactic objects used to decorate formulas. Decidable equality on Fin n is crucial for implementing the "freshness" constraints in the modal rules, which require introducing a token not previously used in a specific context. A position is implemented as Subset n, representing a subset of Fin n.

An *extended sequent* (e-sequent) is an expression $\Gamma \vdash \Delta$, where Γ and Δ are finite sequences of p-formulas.

We present the rules of $\mathsf{E}_{\mathsf{S4.2}}$. The propositional rules are standard, adapted to operate on p-formulas defined at the same position. The structural rules (Weakening, Contraction, Exchange) are also naturally extended to p-formulas. The core of the calculus lies in the modal rules, which manipulate the positions associated with modal formulas. In analogy with first-order logic, the rules $\vdash \Box$ and $\Diamond \vdash$ introduce a fresh token x (where $x \notin s$ and x is new to the rest of the sequent) that we refer to as an *eigentoken*.

Identity rules

$$\frac{}{A^s \vdash A^s}\,Ax \qquad \frac{\Gamma_1 \vdash A^s, \Delta_1 \qquad \Gamma_2, A^s \vdash \Delta_2}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2}\,Cut$$

Structural rules

$$\frac{\Gamma \vdash \Delta}{\Gamma, A^s \vdash \Delta}\,W\vdash \qquad \frac{\Gamma \vdash \Delta}{\Gamma \vdash A^s, \Delta}\,\vdash W$$

$$\frac{\Gamma, A^s, A^s \vdash \Delta}{\Gamma, A^s \vdash \Delta}\,C\vdash \qquad \frac{\Gamma \vdash A^s, A^s, \Delta}{\Gamma \vdash A^s, \Delta}\,\vdash C$$

$$\frac{\Gamma_1, A^s, B^t, \Gamma_2 \vdash \Delta}{\Gamma_1, B^t, A^s, \Gamma_2 \vdash \Delta}\,Exc\vdash \qquad \frac{\Gamma \vdash \Delta_1, A^s, B^t, \Delta_2}{\Gamma \vdash \Delta_1, B^t, A^s, \Delta_2}\,\vdash Exc$$

Propositional rules

$$\frac{\Gamma, A^s \vdash \Delta}{\Gamma, A \land B^s \vdash \Delta}\,\land_1\vdash \qquad \frac{\Gamma, B^s \vdash \Delta}{\Gamma, A \land B^s \vdash \Delta}\,\land_2\vdash \qquad \frac{\Gamma_1 \vdash A^s, \Delta_1 \qquad \Gamma_2 \vdash B^s, \Delta_2}{\Gamma_1, \Gamma_2 \vdash A \land B^s, \Delta_1, \Delta_2}\,\vdash\land$$

$$\frac{\Gamma_1, A^s \vdash \Delta_1 \qquad \Gamma_2, B^s \vdash \Delta_2}{\Gamma_1, \Gamma_2, A \lor B^s \vdash \Delta_1, \Delta_2}\,\lor\vdash \qquad \frac{\Gamma \vdash A^s, \Delta}{\Gamma \vdash A \lor B^s, \Delta}\,\vdash\lor_1 \qquad \frac{\Gamma \vdash B^s, \Delta}{\Gamma \vdash A \lor B^s, \Delta}\,\vdash\lor_2$$

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

Modal rules

$$\frac{\Gamma, A^{s,t} \vdash \Delta}{\Gamma, \Box A^s \vdash \Delta}\,\Box\vdash \qquad \frac{\Gamma \vdash A^{s,x}, \Delta}{\Gamma \vdash \Box A^s, \Delta}\,\vdash\Box$$

$$\frac{\Gamma, A^{s,x} \vdash \Delta}{\Gamma, \Diamond A^s \vdash \Delta}\,\Diamond\vdash \qquad \frac{\Gamma \vdash A^{s,t}, \Delta}{\Gamma \vdash \Diamond A^s, \Delta}\,\vdash\Diamond$$

with the constraints: $x \notin s$ and x fresh for Γ, Δ for $\vdash \Box$ and $\Diamond \vdash$.

We show here the Agda implementation of the $\Box$ introduction rule.

```
⊢□ : ∀ {n A} {x : token {n}} {s : position {n}} {Γ Δ : List (pf {n})}
    → x ∉ s
    → fresh x Γ
    → fresh x Δ
    → Proof (Γ ⊢ [(A ^ s ∪ ⁅ x ⁆)] , Δ)
    → Proof (Γ ⊢ [(□ A ^ s)] , Δ)
```

As a work in progress, we are implementing in Agda the weak completeness theorem for S4.2: if $\vdash_{\mathsf{H}_{\mathsf{S4.2}}} A$ then $\vdash_{\mathsf{E}_{\mathsf{S4.2}}} A^{\emptyset}$, where $\mathsf{H}_{\mathsf{S4.2}}$ is the Hilbert-style axiomatization of S4.2 (see e.g. [20]). In the following the derivation of the characteristic axiom:

```
geachAxiom : ∀ {n A}
  → (Proof {suc (suc n)} ([] ⊢ [((◇ (□ A)) ⇒ (□ (◇ A)) ^ ⊥)]))
geachAxiom {A = A} =
  ⊢⇒
  (◇⊢ {x = x} {Γ = []} ∉⊥ (λ ()) (λ ())
  (⊢□ {x = y} ∉⊥ (y-fresh-helper {A = A}) ∉⊥
  (⊢◇ {s = ⊥ ∪ ⁅ y ⁆} {t = ⁅ x ⁆}
  (□⊢ {A = A} {s = ⊥ ∪ ⁅ x ⁆} {t = ⁅ y ⁆} {Γ = []}
  Ax))))
```

3. The cut-elimination theorem in Agda

We present the ongoing implementation in the Agda proof assistant of the Cut-Elimination Theorem for $E_{S4.2}$. The theoretical result is complete and proven on paper in [12] using standard Gentzen techniques adapted for modal position-based sequent calculi. The proof establishes termination through a careful induction on the complexity of proofs, as measured by the δ function. This approach mirrors classical cut-elimination proofs but requires handling of position information throughout the elimination process.

The Agda implementation proceeds by structural induction on proof terms, following the same logical structure as the theoretical proof. The `cutElimination` function is fully defined for all inference rules, implementing the complete elimination strategy with proper case analysis for each logical connective and modal operator. However, three crucial auxiliary lemmas (`mixLemma`, `removeOneMore`, and `weakeningRemove`) are currently postulated—their types are declared, but their proofs are not yet complete. Additionally, we have not yet resolved how to convince Agda's termination checker using the δ complexity measure, as the recursive calls operate on transformed rather than structurally smaller proof terms.

Theorem 3.1 (Cut-Elimination). *If* Π *is a proof of* $\Gamma \vdash \Delta$*, then there exists a cut-free proof* Π^* *of* $\Gamma \vdash \Delta$*.*

Proof. Below we present the code for the proof.

```
cutElimination : ∀ {n Γ Δ} → (Π : Proof {n} (Γ ⊢ Δ)) → CutFreeProof (Γ ⊢ Δ)
cutElimination Ax                  = Ax
cutElimination (Cut {A = A} {Γ₁ = Γ₁} {Γ₂ = Γ₂} {Δ₁ = Δ₁} {s = s} Π₁ Π₂)
  with δ (Cut Π₁ Π₂) | δ-cut-≢-zero Π₁ Π₂
... | zero | p = ⊥-elim (p refl)
... | suc m | _ =
  weakeningRemove {Γ₁ = Γ₁} {Δ₁ = Δ₁} (A ^ s) (
    cutElimination (
      removeOneMore {Γ₁ = Γ₁} {Γ₂ = Γ₂} (A ^ s)
      (proj₁ (
        mixLemma A s refl
        (cutFreeToStandard (cutElimination Π₁))
        (cutFreeToStandard (cutElimination Π₂))
        (rankCutFreeIsZero (cutElimination Π₁))
        (rankCutFreeIsZero (cutElimination Π₂))
        ))))
cutElimination (W⊢ Π)          = W⊢ (cutElimination Π)
cutElimination (⊢W Π)          = ⊢W (cutElimination Π)
cutElimination (C⊢ Π)          = C⊢ (cutElimination Π)
cutElimination (⊢C Π)          = ⊢C (cutElimination Π)
cutElimination (Exc⊢ {Γ₁ = Γ₁} Π) = Exc⊢ {Γ₁ = Γ₁} (cutElimination Π)
cutElimination (⊢Exc {Δ₁ = Δ₁} Π) = ⊢Exc {Δ₁ = Δ₁} (cutElimination Π)
cutElimination (¬⊢ Π)          = ¬⊢ (cutElimination Π)
cutElimination (⊢¬ Π)          = ⊢¬ (cutElimination Π)
cutElimination (∧₁ Π)          = ∧₁ (cutElimination Π)
cutElimination (∧₂ Π)          = ∧₂ (cutElimination Π)
cutElimination (⊢∧ Π₁ Π₂)      = ⊢∧ (cutElimination Π₁) (cutElimination Π₂)
cutElimination (∨⊢ Π₁ Π₂)      = ∨⊢ (cutElimination Π₁) (cutElimination Π₂)
cutElimination (⊢∨₁ Π)         = ⊢∨₁ (cutElimination Π)
cutElimination (⊢∨₂ Π)         = ⊢∨₂ (cutElimination Π)
cutElimination (⇒⊢ Π₁ Π₂) = ⇒⊢ (cutElimination Π₁) (cutElimination Π₂)
cutElimination (⊢⇒ Π)       = ⊢⇒ (cutElimination Π)
cutElimination (□⊢ Π)          = □⊢ (cutElimination Π)
cutElimination (⊢□ x x₁ x₂ Π) = ⊢□ x x₁ x₂ (cutElimination Π)
cutElimination (◊⊢ x x₁ x₂ Π) = ◊⊢ x x₁ x₂ (cutElimination Π)
cutElimination (⊢◊ Π)          = ⊢◊ (cutElimination Π)
```

where

```
removeOneMore : ∀ {n} {Γ₁ Γ₂ Δ₁ Δ₂ : List (pf {n})} (P : pf {n}) →
  Proof (Γ₁ , ((Γ₂ , [(P)]) - P) ⊢ (([(P)] , Δ₁) - P) , Δ₂) →
  Proof (Γ₁ , Γ₂ - P ⊢ Δ₁ - P , Δ₂)
```

and

```
weakeningRemove : ∀ {n} {Γ₁ Γ₂ Δ₁ Δ₂ : List (pf {n})} (P : pf {n}) →
  CutFreeProof (Γ₁ , (Γ₂ - P) ⊢ (Δ₁ - P) , Δ₂) →
  CutFreeProof (Γ₁ , Γ₂ ⊢ Δ₁ , Δ₂)
```

□

The auxiliary procedures deserve some explanations. The `removeOneMore` function addresses a technical challenge during mix operations: when combining cut-free proofs, redundant occurrences of the cut formula may appear in intermediate contexts and require careful removal. The `weakeningRemove` function performs the complementary operation, systematically restoring formulas that were temporarily filtered out and ensuring the final proof maintains proper sequent structure.

The correctness of this process rests on other key lemmas: cutFreeToStandard expresses that a cut-free proof is a proof of the statement, rankCutFreeIsZero states that a cut-free proof has rank zero, and the δ-`cut-≢-zero` lemma ensures that no Cut rule can have zero rank, preventing infinite recursion while maintaining the rank bounds essential for termination.

The *subformula property* follows as a corollary of the Cut-Elimination Theorem,.

Corollary 1 (Subformula Property)**.** *Any formula occurring in a cut-free proof* Π *of* $\Gamma \vdash \Delta$ *is a subformula of some formula in the end-sequent* $\Gamma \vdash \Delta$*.*

Moreover, we also obtain a purely syntactical proof of the consistency of the system:

Corollary 2 (Consistency)**.** *The empty sequent* $\vdash$ *is not provable in* $\mathsf{E}_{S4.2}$*.*

4. Conclusions and future work

In this paper, we introduced $\mathsf{E}_{S4.2}$, an extended sequent calculus for the modal logic S4.2 which utilizes *positions*, i.e., sets of uninterpreted tokens, to annotate formulas and manage modal information. We implemented the system in the Agda proof assistant and are working on providing a formal syntactic proof of the Cut-Elimination Theorem, from which the consistency of the system and the subformula property follow. We presented the main structure of the proof, and as a work in progress, we are implementing the auxiliary lemmata. Following [20], we are defining semantics based on semilattices with a minimum, which allow for a sound interpretation of position-formulas.
As part of our ongoing work, we are formalizing in Agda the soundness theorem and the weak completeness theorem with respect to the Hilbert-style axiomatization. Moreover, motivated by applications to mathematics, we are mechanizing the semantics and proofs presented in [3], where S4.2 is interpreted as the modal logic of abelian groups.
Finally, this work represents the first step toward the development of a unified framework, NAMOR (New Agda MOdal Realization) [22], which builds upon the theory of 2-sequents and extended sequents [12] and aims to encompass the entire range of normal modal logics from K to S5. Following the underlying theoretical systems, NAMOR is designed to be highly parametric, as all normal modal systems share the same set of rules.
Beyond its implementation aspects, S4.2 also raises several open theoretical questions, such as the correct Hilbert-style axiomatization of the intuitionistic counterpart of the system: although the transformation from classical to intuitionistic proof systems is well-known at the proof-theoretic level, the axiomatization of the intuitionistic version of S4.2 remains unclear.

Acknowledgments

Margherita Zorzi's work is partially supported by INDAM-Istituto Nazionale di Alta Matematica “Francesco Severi”, group GNSAGA-Logica matematica e applicazioni.

Declaration on generative AI

During the preparation of this work, the authors used DeepL, 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] R. Goldblatt, Diodorean modality in Minkowski spacetime, Studia Logica 39 (1980) 219–236. URL: http://www.jstor.org/stable/20014982.

[2] J. D. Hamkins, B. Loewe, The modal logic of forcing, TRANS. of the AMS 360(4), 1793–1817 (2008).

[3] S. Berger, A. Block, B. Loewe, The modal logic of abelian groups, Algebra Univers. 84, 25 (2023).

[4] E. Brauer, The modal logic of potential infinity: Branching versus convergent possibilities, Erkenntnis 87 (2020) 1–19. doi:10.1007/s10670-020-00296-3.

[5] O. Linnebo, S. Shapiro, Actual and potential infinity, Noûs 53 (2017) 160–191. doi:10.1111/nous.12208.

[6] 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.

[7] W. Lenzen, Epistemic logic (2004) 963–983. URL: https://doi.org/10.1007/978-1-4020-1986-9_26. doi:10.1007/978-1-4020-1986-9_26.

[8] S. Negri, E. Pavlović, A proof-theoretic approach to formal epistemology, in: Y. Weiss, R. Birman (Eds.), Saul Kripke on Modal Logic, 2024, pp. 303–345.

[9] R. Stalnaker, On logics of knowledge and belief, Philosophical Studies 128 (2006) 169–199. doi:10.1007/s11098-005-4062-y.

[10] S. Negri, Proof theory for modal logic, Philosophy Compass 6 (2011) 523–538.

[11] T. A. Team, Agda documentation, 2025. URL: https://agda.readthedocs.io/en/latest/index.html.

[12] 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.

[13] L. Viganò, Labelled non-classical logics, Kluwer Academic Publishers (1997).

[14] D. A. Basin, S. Matthews, L. Viganò, Natural deduction for non-classical logics, Stud Logica 60 (1998) 119–160. URL: https://doi.org/10.1023/A:1005003904639. doi:10.1023/A:1005003904639.

[15] R. Dyckhoff, S. Negri, Admissibility and cut-elimination for modal logics, Journal of Logic and Computation (2012).

[16] M. Takano, A modified subformula property for the modal logic s4.2, Bulletin of the Section of Logic 48 (2019) null. URL: http://eudml.org/doc/295583.

[17] 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.

[18] M. Maggesi, C. Perini Brogi, Mechanising gödel-löb provability logic in HOL light, J. Autom. Reason. 67 (2023) 29. URL: https://doi.org/10.1007/s10817-023-09677-z. doi:10.1007/S10817-023-09677-Z.

[19] 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.

[20] S. Martini, A. Masini, M. Zorzi, A natural deduction calculus for s4.2, Notre Dame Journal of Formal Logic 65 (2024) 127–150. doi:10.1215/00294527-2024-0011.

[21] T. A. Team, Documentation for the agda standard library, 2025. URL: https://agda.github.io/agda-stdlib/.

[22] R. Borsetto, M. Zorzi, Namor: a new agda library for modal extended sequents, in: OVERLAY 2025, 7th International Workshop on Artificial Intelligence and fOrmal VERification, Logic, Automata, and sYnthesis October 26th, Bologna, Italy, CEUR Conference Proceedings, to appear, 2025.