Soundness-Preserving Fusion of Modal Logics in Coq

Miguel A. Nunes✉[1]   Karina G. Roggia[2]   Paulo Torrens[3]
[1]miguel.alfredo.nunes@gmail.com
[2]karina.roggia@udesc.br
[3]paulotorrens@gnu.org

[1]IFCH - Unicamp   [2]DCC - UDESC   [3]University of Kent

2024-12-05

Supported by:
CNPq   fapesc Fundação de Amparo à Pesquisa e Inovação do Estado de Santa Catarina

Sumary

SBMF 2024
27th Brazilian Symposium on Formal Methods

0. Introduction

Introduction

- Modal logics are well established as tools useful for applications in philosophy, mathematics and computer science [6];
- However, modal logics with a single modal operator remain the most well used and well known kinds of modal logic, something that philosopher Dana Scott [18, pp. 161] was quite dissatisfied by some fifty years ago:

Introduction

- Modal logics are well established as tools useful for applications in philosophy, mathematics and computer science [6];
- However, modal logics with a single modal operator remain the most well used and well known kinds of modal logic, something that philosopher Dana Scott [18, pp. 161] was quite dissatisfied by some fifty years ago:

Introduction

- Modal logics are well established as tools useful for applications in philosophy, mathematics and computer science [6];
- However, modal logics with a single modal operator remain the most widely used and well known kinds of modal logic, something that philosopher Dana Scott [18, pp. 161] was quite dissatisfied by some fifty years ago:

  "Here is what I consider one of the biggest mistakes of all in modal logic: concentration on a system with just *one* modal operator."

Introduction

- Modal logics with several distinct modal operators (known as *multimodal*) are far more expressive than ones with a single modal operator;
- These logics may be obtained by defining them from the ground up or by combining several existing modal logics;
- The latter method has the benefit of ensuring that certain interesting properties of the logics being combined are carried over to the combined logic.

Introduction

- Modal logics with several distinct modal operators (known as *multimodal*) are far more expressive than ones with a single modal operator;
- These logics may be obtained by defining them from the ground up or by combining several existing modal logics;
- The latter method has the benefit of ensuring that certain interesting properties of the logics being combined are carried over to the combined logic.

Introduction

- Modal logics with several distinct modal operators (known as *multimodal*) are far more expressive than ones with a single modal operator;
- These logics may be obtained by defining them from the ground up or by combining several existing modal logics;
- The latter method has the benefit of ensuring that certain interesting properties of the logics being combined are carried over to the combined logic.

Introduction

- To combine logics, however, is no easy feat, as it requires understanding of both the logics being combined and the method being utilized;
- As such, it is very important to take great care to not forget any minor detail in a definition or any elusive step in a proof;
- This suggests that proof assistants may be useful tools when attempting to combine logics.

Introduction

- To combine logics, however, is no easy feat, as it requires understanding of both the logics being combined and the method being utilized;
- As such, it is very important to take great care to not forget any minor detail in a definition or any elusive step in a proof;
- This suggests that proof assistants may be useful tools when attempting to combine logics.

Introduction

- To combine logics, however, is no easy feat, as it requires understanding of both the logics being combined and the method being utilized;
- As such, it is very important to take great care to not forget any minor detail in a definition or any elusive step in a proof;
- This suggests that proof assistants may be useful tools when attempting to combine logics.

Introduction

- In order to investigate the use of proof assistants for combining modal logics, in this work we present a formalization of the method of fusion of modal logics in the Coq proof assistant;
- This method allows the combination of arbitrary modal logics, while also ensuring that soundness is preserved - i.e. if all logics being combined are sound then so is the resulting logic [14];

Introduction

- In order to investigate the use of proof assistants for combining modal logics, in this work we present a formalization of the method of fusion of modal logics in the Coq proof assistant;
- This method allows the combination of arbitrary modal logics, while also ensuring that soundness is preserved - i.e. if all logics being combined are sound then so is the resulting logic [14];

Introduction

- This work extends a previous work on a library of modal logic in Coq by da Silveira et al. [9];
- Our work is available online at `https://github.com/funcao/LML`, under a permissive free software license.

0.1 Related Work

Related Work

- Proof assistants have widely been used to represent and reason with or about logics and their combinations;
- We found that the work that are the most closely related to ours can roughly be divided in three main areas.

Related Work

- Proof assistants have widely been used to represent and reason with or about logics and their combinations;
- We found that the work that are the most closely related to ours can roughly be divided in three main areas.

First Area

Proving metaproperties of modal logics in proof assistants and automated theorem provers.

1. Bentzen (2021) [3];
2. Maggesi and Brogi (2021) [16].

Second Area

Formalizing argumentation and reasoning with proof assistants and automated theorem provers.

1. Fuenmayor and Benzmüller (2019) [12];
2. Fuenmayor and Benzmüller (2019a) [11];
3. Benzmüller, Parent and van der Torre (2020) [5].

Third Area

SBMF 2024
27th Brazilian Symposium on
Formal Methods

Formalizing the combination of logics in proof assistants.

1. Benzmüller (2010) [4];
2. Rabe (2017) [17];
3. Lescanne and Puisségur (2007) [15].

Related Work

- Of all works analyzed, none attempted to tackle methods of combining logics themselves;
- The works present logics resulting from combination but did not attempt to combine them in the tools utilized nor did they attempt to show the preservation of properties;
- To the best of the authors' knowledge, we believe our main result to be a novel approach as we aim at the fusion of arbitrary modal logic systems.

Related Work

- Of all works analyzed, none attempted to tackle methods of combining logics themselves;
- The works present logics resulting from combination but did not attempt to combine them in the tools utilized nor did they attempt to show the preservation of properties;
- To the best of the authors' knowledge, we believe our main result to be a novel approach as we aim at the fusion of arbitrary modal logic systems.

Related Work

- Of all works analyzed, none attempted to tackle methods of combining logics themselves;
- The works present logics resulting from combination but did not attempt to combine them in the tools utilized nor did they attempt to show the preservation of properties;
- To the best of the authors' knowledge, we believe our main result to be a novel approach as we aim at the fusion of arbitrary modal logic systems.

1. Core Concepts

1.1 Multimodal Logics

Multimodal Logic

- Multimodal logic is an extension of modal logic that allows the representation of many distinct modalities in the same language;
- For example, in a multimodal language, it is possible to have modalities that represent both time and knowledge of an agent;
- It is worth noting, however, that modal logics with two (or more) modalities that are interdefinable is not multimodal, for a logic to be classified as multimodal the modalities must be distinct.

Multimodal Logic

- Multimodal logic is an extension of modal logic that allows the representation of many distinct modalities in the same language;
- For example, in a multimodal language, it is possible to have modalities that represent both time and knowledge of an agent;
- It is worth noting, however, that modal logics with two (or more) modalities that are interdefinable is not multimodal, for a logic to be classified as multimodal the modalities must be distinct.

Multimodal Logic

- Multimodal logic is an extension of modal logic that allows the representation of many distinct modalities in the same language;
- For example, in a multimodal language, it is possible to have modalities that represent both time and knowledge of an agent;
- It is worth noting, however, that modal logics with two (or more) modalities that are interdefinable is not multimodal, for a logic to be classified as multimodal the modalities must be distinct.

The Language of Multimodal Logics

For the grammar of multimodal logics we must replace the connectives $\Box$ and $\Diamond$ present in the grammar of modal logic each by a list $\Box_1, \ldots, \Box_n$ and $\Diamond_1, \ldots, \Diamond_n$, as such we have:

Definition (Grammar)

$$\varphi ::= p \mid \top \mid \bot \mid \neg\varphi \mid \Box_i\varphi \mid \Diamond_i\varphi \mid (\varphi_1 \wedge \varphi_2) \mid (\varphi_1 \vee \varphi_2) \mid (\varphi_1 \rightarrow \varphi_2)$$

Where $p \in \mathcal{P}$ and $i, n \in \mathbb{N}$ and $1 \leq i \leq n$

The connectives $\Box_i$ and $\Diamond_i$ are dual and each instance $\Box_i\varphi$ (resp. $\Diamond_i\varphi$) may be replaced by $\neg\Diamond_i\neg\varphi$ (resp. $\neg\Box_i\neg\varphi$).

The Components of Multimodal Logic

For the sake of brevity, I will omit the definitions of semantics, syntax and metaproperties of both modal logics and multimodal logics, as we use the same definitions that are usual in the literature, such as in Blackburn, de Rijke and Venema [6] or Gabbay [13].

1.2 Combining Logics

Combining Logics

SBMF 2024
27th Brazilian Symposium on Formal Methods

- Combinations of logics is a relatively new topic in the field of modern logic [8];
- To combine logics is to use some method to obtain new logical systems from preexisting ones, whether through joining up several systems or breaking up one system into simpler ones [7];
- This work focuses on the method of fusion of modal logics, the simplest and oldest generic method to combine logics [13, 8].

Combining Logics

- Combinations of logics is a relatively new topic in the field of modern logic [8];
- To combine logics is to use some method to obtain new logical systems from preexisting ones, whether through joining up several systems or breaking up one system into simpler ones [7];
- This work focuses on the method of fusion of modal logics, the simplest and oldest generic method to combine logics [13, 8].

Combining Logics

- Combinations of logics is a relatively new topic in the field of modern logic [8];
- To combine logics is to use some method to obtain new logical systems from preexisting ones, whether through joining up several systems or breaking up one system into simpler ones [7];
- This work focuses on the method of fusion of modal logics, the simplest and oldest generic method to combine logics [13, 8].

The Fusion of Modal Logics

- The fusion of modal logics is a binary operation defined on the class of modal logics semantically defined by Kripke semantics and syntactically defined by a Hilbert calculus [7];
- Given two logics $\mathcal{L}_1$ and $\mathcal{L}_2$ that can be fused together, they may be fused together into a new logic $\mathcal{L}_3$ that has two $\Box$ connectives, two instances of the K axiom and the necessitation rule, and all their frames will have two accessibility relations, one for each $\Box$;
- Simply put, the fusion of two monomodal logics will generate a bimodal logic.

The Fusion of Modal Logics

- The fusion of modal logics is a binary operation defined on the class of modal logics semantically defined by Kripke semantics and syntactically defined by a Hilbert calculus [7];
- Given two logics $\mathcal{L}_1$ and $\mathcal{L}_2$ that can be fused together, they may be fused together into a new logic $\mathcal{L}_3$ that has two $\Box$ connectives, two instances of the K axiom and the necessitation rule, and all their frames will have two accessibility relations, one for each $\Box$;
- Simply put, the fusion of two monomodal logics will generate a bimodal logic.

The Fusion of Modal Logics

- The fusion of modal logics is a binary operation defined on the class of modal logics semantically defined by Kripke semantics and syntactically defined by a Hilbert calculus [7];
- Given two logics $\mathcal{L}_1$ and $\mathcal{L}_2$ that can be fused together, they may be fused together into a new logic $\mathcal{L}_3$ that has two $\Box$ connectives, two instances of the K axiom and the necessitation rule, and all their frames will have two accessibility relations, one for each $\Box$;
- Simply put, the fusion of two monomodal logics will generate a bimodal logic.

The Fusion of Modal Logics

- The fusion of modal logics, like many other methods of combining logics, preserves certain properties of the combined logics;
- Given any two modal logics, $\mathcal{L}_1$ and $\mathcal{L}_2$, that respect a certain property $\mathbf{P}$, then the logic resulting from their fusion will also respect property $\mathbf{P}$ [14];
- Some properties preserved by fusion are soundness, completeness, finite model property, decidability and interpolation [13].

The Fusion of Modal Logics

SBMF 2024
27th Brazilian Symposium on Formal Methods

- The fusion of modal logics, like many other methods of combining logics, preserves certain properties of the combined logics;
- Given any two modal logics, $\mathcal{L}_1$ and $\mathcal{L}_2$, that respect a certain property **P**, then the logic resulting from their fusion will also respect property **P** [14];
- Some properties preserved by fusion are soundness, completeness, finite model property, decidability and interpolation [13].

The Fusion of Modal Logics

SBMF 2024 27th Brazilian Symposium on Formal Methods

- The fusion of modal logics, like many other methods of combining logics, preserves certain properties of the combined logics;
- Given any two modal logics, $\mathcal{L}_1$ and $\mathcal{L}_2$, that respect a certain property **P**, then the logic resulting from their fusion will also respect property **P** [14];
- Some properties preserved by fusion are soundness, completeness, finite model property, decidability and interpolation [13].

2. Modal and Multimodal Logics in Coq

2.1 Base Library

A Brief Overview

- A modal logic library developed by da Silveira et al. [9] was used as a basis for this work;
- Modal logic is represented in the style of a deep embedding in this library, the semantics is an implementation of Kripke semantics and the syntax is an implementation of Hilbert calculus, with proofs of both soundness and weak completeness;
- An embedding of a logic $\mathcal{L}_1$ inside another logic $\mathcal{L}_2$ is an encoding of the components of $\mathcal{L}_1$ in the language of $\mathcal{L}_2$. An embedding is said to be *deep* if both semantics and syntax of $\mathcal{L}_1$ is defined inside of $\mathcal{L}_2$ [1].

A Brief Overview

- A modal logic library developed by da Silveira et al. [9] was used as a basis for this work;
- Modal logic is represented in the style of a deep embedding in this library, the semantics is an implementation of Kripke semantics and the syntax is an implementation of Hilbert calculus, with proofs of both soundness and weak completeness;
- An embedding of a logic $\mathcal{L}_1$ inside another logic $\mathcal{L}_2$ is an encoding of the components of $\mathcal{L}_1$ in the language of $\mathcal{L}_2$. An embedding is said to be *deep* if both semantics and syntax of $\mathcal{L}_1$ is defined inside of $\mathcal{L}_2$ [1].

A Brief Overview

SBMF 2024 27th Brazilian Symposium on Formal Methods

- A modal logic library developed by da Silveira et al. [9] was used as a basis for this work;
- Modal logic is represented in the style of a deep embedding in this library, the semantics is an implementation of Kripke semantics and the syntax is an implementation of Hilbert calculus, with proofs of both soundness and weak completeness;
- An embedding of a logic $\mathcal{L}_1$ inside another logic $\mathcal{L}_2$ is an encoding of the components of $\mathcal{L}_1$ in the language of $\mathcal{L}_2$. An embedding is said to be *deep* if both semantics and syntax of $\mathcal{L}_1$ is defined inside of $\mathcal{L}_2$ [1].

A Brief Overview

- This library has a custom notation that allows intuitive representation of modal formulas inside of Coq: $\Box p_0 \to (p_1 \wedge \Diamond p_2)$ becomes `[! []#0 -> (#1 /\ <>#2) !]`;
- Evaluation of formulas and semantic entailment are direct translations into Coq of the pen and paper definitions;
- Axiom systems and syntactical entailment require more complicated inductive definitions to be expressed;
- Many usual modal systems are defined, both in semantical and syntactical terms, and new systems can be easily defined by adding their respective frame conditions or axioms (or both).

A Brief Overview

SBMF 2024
27th Brazilian Symposium on Formal Methods

- This library has a custom notation that allows intuitive representation of modal formulas inside of Coq: $\Box p_0 \to (p_1 \wedge \Diamond p_2)$ becomes `[! []#0 -> (#1 /\ <>#2) !]`;
- Evaluation of formulas and semantic entailment are direct translations into Coq of the pen and paper definitions;
- Axiom systems and syntactical entailment require more complicated inductive definitions to be expressed;
- Many usual modal systems are defined, both in semantical and syntactical terms, and new systems can be easily defined by adding their respective frame conditions or axioms (or both).

A Brief Overview

- This library has a custom notation that allows intuitive representation of modal formulas inside of Coq: $\Box p_0 \to (p_1 \wedge \Diamond p_2)$ becomes `[! []#0 -> (#1 /\ <>#2) !]`;
- Evaluation of formulas and semantic entailment are direct translations into Coq of the pen and paper definitions;
- Axiom systems and syntactical entailment require more complicated inductive definitions to be expressed;
- Many usual modal systems are defined, both in semantical and syntactical terms, and new systems can be easily defined by adding their respective frame conditions or axioms (or both).

A Brief Overview

- This library has a custom notation that allows intuitive representation of modal formulas inside of Coq: $\Box p_0 \rightarrow (p_1 \wedge \Diamond p_2)$ becomes `[! []#0 -> (#1 /\ <>#2) !]`;
- Evaluation of formulas and semantic entailment are direct translations into Coq of the pen and paper definitions;
- Axiom systems and syntactical entailment require more complicated inductive definitions to be expressed;
- Many usual modal systems are defined, both in semantical and syntactical terms, and new systems can be easily defined by adding their respective frame conditions or axioms (or both).

2.2 Our Modifications

Towards Multiple Modalities

SBMF 2024 27th Brazilian Symposium on Formal Methods

- To be able to represent multimodal logics it was first necessary to define how to represent indexes of modalities such that $\Box_i \neq \Box_j$ iff $i \neq j$;
- This was done by defining a small type of indexes as an implicit argument.

```
Context {I: Set}. (* Universe of all indexes *)
```

Towards Multiple Modalities

The type of indexes serves as a base to define a typeclass that represents the set of actual indexes, this simply being the characteristic function over the base type of indexes.

```
(* Set of indexes that are actually used *)
Class modal_index_set: Type := {
    C: I → Prop (* C is a set of I, a small type. *)
}.
```

Towards Multiple Modalities

With this set of indexes, we then define an instance of this set of indexes as an implicit generalizing argument as a `Context`.

```
(* Implicitly setting the indexes as arguments
   to all following definitions *)
Context `{X: modal_index_set}.
```

Towards Multiple Modalities

Finally, we define a `Structure` that represents the actual indexes which will be used to build the modal formulas. This structure has two components, an index and a proof that this index is valid.

```
(* Set of indexes that will build
   the formulas of a given logic *)
Structure modal_index: Set := {
    index: I;
    index_valid: C index (* Proof index is an element of C. *)
}.
```

Indexing Indexes

```
Context {I: Set}. (* Universe of all indexes *)

(* Set indexes that are actually used *)
Class modal_index_set: Type := {
    C: I → Prop (* C is a set of I, a small type. *)
}.

(* Implicitly setting the indexes as arguments
    to all following definitions *)
Context `{X: modal_index_set}.

(* Set of indexes that will build
    the formulas of a given logic *)
Structure modal_index: Set := {
    index: I;
    index_valid: C index (* Proof index is an element of C. *)
}.
```

The Language of Multimodal Logic in Coq

We are now able to define the language of multimodal logic in Coq by making minor modifications to the definition in the base library

```
(*some cases omitted*)
Inductive formula: Set :=
    | Lit    : nat → formula
    | Neg    : formula → formula
    | Box    : modal_index → formula → formula
    | Dia    : modal_index → formula → formula
```

This simple change, allows us to extend the base library to deal with multimodal logics, while also preserving the ability to reason about monomodal logics, by simply defining a singleton set as the index set.

The Components of Multimodal Logic in Coq

SBMF 2024 27th Brazilian Symposium on Formal Methods

- For the sake of brevity I will not explain in detail the remaining modifications, I will only list out the major changes;
- Some modifications had to be done to the definition of semantics to represent multimodal frames and to use them in the valuation function;
- The notation system had to be extended to show the indexes of modalities;
- The deductive system had to be extended to be able to represent and deduce formulas with many distinct modalities;
- The proofs of both soundness and weak completeness had to be extended to multimodal systems.

The Components of Multimodal Logic in Coq

- For the sake of brevity I will not explain in detail the remaining modifications, I will only list out the major changes;
- Some modifications had to be done to the definition of semantics to represent multimodal frames and to use them in the valuation function;
- The notation system had to be extended to show the indexes of modalities;
- The deductive system had to be extended to be able to represent and deduce formulas with many distinct modalities;
- The proofs of both soundness and weak completeness had to be extended to multimodal systems.

The Components of Multimodal Logic in Coq

- For the sake of brevity I will not explain in detail the remaining modifications, I will only list out the major changes;
- Some modifications had to be done to the definition of semantics to represent multimodal frames and to use them in the valuation function;
- The notation system had to be extended to show the indexes of modalities;
- The deductive system had to be extended to be able to represent and deduce formulas with many distinct modalities;
- The proofs of both soundness and weak completeness had to be extended to multimodal systems.

The Components of Multimodal Logic in Coq

SBMF 2024 27th Brazilian Symposium on Formal Methods

- For the sake of brevity I will not explain in detail the remaining modifications, I will only list out the major changes;
- Some modifications had to be done to the definition of semantics to represent multimodal frames and to use them in the valuation function;
- The notation system had to be extended to show the indexes of modalities;
- The deductive system had to be extended to be able to represent and deduce formulas with many distinct modalities;
- The proofs of both soundness and weak completeness had to be extended to multimodal systems.

The Components of Multimodal Logic in Coq

- For the sake of brevity I will not explain in detail the remaining modifications, I will only list out the major changes;
- Some modifications had to be done to the definition of semantics to represent multimodal frames and to use them in the valuation function;
- The notation system had to be extended to show the indexes of modalities;
- The deductive system had to be extended to be able to represent and deduce formulas with many distinct modalities;
- The proofs of both soundness and weak completeness had to be extended to multimodal systems.

3. Combining Modal Logics in Coq

Combining Modal Logics in Coq

To represent the fusion of modal logics, the first problem we had to tackle was how to represent the set of indexes a system resulting from fusion.

This is quite simple, as this set is the disjoint union of the set of indexes of each fused system.

```
Context {I1: Set}. (* universe of first logic *)
Context {I2: Set}. (* universe of second logic *)
Definition fusion: Set := I1 + I2. (* Disjoint
                        union of types (universes). *)
```

Indexes

Next, we had to define an instance of the `modal_index_set` typeclass and prove that this instance exists, i.e. that there are indexes in this set. This being done by showing that any index in this set belongs to either base set.

```
Context {X1: @modal_index_set I1}.
Context {X2: @modal_index_set I2}.
Instance fusion_index_set: @modal_index_set fusion := {|
    C i := (* Disjoint union of sets. *)
        match i with
        | inl a ⇒ @C I1 X1 a
        | inr b ⇒ @C I2 X2 b
        end
|}.
```

Lifiting

- During the development of this section on fusion, we encountered a problem: Coq's type theory does not have a subtype relation, so it does not allow an object to belong to two types at once;
- This is a problem because in the standard pen and paper formulations, it is supposed that certain objects will belong to two sets at once, so when converting the proof to type theory, we somehow had to show that certain components belong to two types at once.

Lifiting

This was solved by defining a typeclass that "lifts" objects from some basic type to a fused type, which is simply a function from some type A to some type B:

```
Class lift_conv (A B: Type): Type := {
    lift: A → B
}.
```

And defining some instances of this typeclass to relevant components:

```
Instance index_lift1: lift_conv modal_index1 modal_indexF.
Instance formula_lift1: lift_conv formula1 formulaF.
Instance axiom_lift1: lift_conv axiom1 axiomF.
(* proofs omitted *)
```

Notation

We defined some local notations to ease the development of the remaining section.

```
Local Notation modal_index1 := (@modal_index I1 X1).
Local Notation modal_index2 := (@modal_index I2 X2).
Local Notation modal_indexF :=
        (@modal_index fusion fusion_index_set).
```

Frames

Frames may be fused together if they have the same sets of worlds, if they do then we can construct a frame that has all the relations of both the base frames.

```
Definition join_frames (f1: Frame1) (f2: Frame2)
                         (H: W f2 = W f1): FrameF.
Proof.
    (* Proof omitted. *)
Defined.
```

Splitting and Splicing

Next, it was necessary to show that given a frame (resp. model) of a logic resulting from fusion, it was possible to split this frame (resp. model) into frames (resp. models) of the base logics that were fused together and that a split model could be lifted back up into a fusion model:

```
Lemma split_frame1: FrameF → Frame1.
Lemma split_model1: ModelF → Model1.
Instance lift_split_model1 M:
    @lift_conv (W (F (split_model1 M))) (W (F M)).
(* proofs omitted *)
```

Splitting and Splicing

Now we could prove that fusion preserves formula evaluation, that is, if a formula φ is true/false in a model $\mathcal{M}_1$ that was split from a fusion model $\mathcal{M}_\mathcal{F}$, then φ lifted up to the fusion indexes will have the same truth value in the model $\mathcal{M}_\mathcal{F}$:

```
Lemma split_model1_coherence:
    forall M f w,
    fun_validation (split_model1 M) w f ↔
    fun_validation M (lift w) (lift f).
Proof.
    (* Proof omitted. *)
Qed.
```

Classes of Frames

Another important component is the definition of the classes of frames that respect a same property (imposed on their accessibility relation) for both the monomodal frames and the fusion frames.

As these properties are defined as functions from frames to Coq's type of propositions Prop in the base library, the definition for the fusion case is straightforward, it is simply a conjunction of the properties of both base frames.

```
Variable P1: Frame1 → Prop.
Variable P2: Frame2 → Prop.

Definition PF: FrameF → Prop :=
    (* Satisfy the frame condition from both logics. *)
    fun F ⇒ P1 (split_frame1 F) ∧ P2 (split_frame2 F).
```

Axiom Systems

The axiom system of the logic resulting from fusion is simply the union of the axiom systems of each of the components logics, with the indexes lifted up to the indexes of the fusion.

```
Variable A1: axiom1 → Prop.
Variable A2: axiom2 → Prop.

Inductive fusion_axioms: axiomF → Prop :=
| fusion_axioms1: (* Include all axioms from system 1. *)
    forall p, A1 p → fusion_axioms (lift p)
| fusion_axioms2: (* Include all axioms from system 2. *)
    forall p, A2 p → fusion_axioms (lift p).
```

Soundness

Before being able to define the preservation of soundness, it was necessary to give a generic definition of the concept of soundness wrt a class of frames, as the definition on the base library was exclusive to system K.

```
Definition entails_modal_class (G: theory) (p: formula): Prop :=
    forall M,  P (F M) → entails M G p.

Definition sound I `{X: @modal_index_set I} P A: Prop :=
    forall G p, (A;  G |-- p) → entails_modal_class P G p.
```

Soundness Preservation

At last, we can define the preservation of soundness by fusion. If two axioms systems A_1 and A_2 are sound with respect to classes of frames $\mathcal{P}_1$ and $\mathcal{P}_2$, then the axiom system A_f resulting from the fusion of A_1 and A_2 is sound with respect to the class of frames $\mathcal{P}_f$.

```
Theorem soundness_transfer:
    forall I1 `{X1: @modal_index_set I1} P1 A1,
    forall I2 `{X2: @modal_index_set I2} P2 A2,
    sound I1 P1 A1 →
    sound I2 P2 A2 →
    sound fusion (PF P1 P2) (fusion_axioms A1 A2).
Proof.
    (* proof omitted *)
Qed.
```

Example: Fusion of S4 with S4

We can use this result to model a proper fusion of modal logics, in this case, the fusion of two S4 systems such as the one considered by van Benthem et al. in [2]. First we define the indexes of the system.

```
Local Instance unit_index: @modal_index_set unit := {|
    C x := True (* Use the whole universe (i.e., unit). *)
|}.

(* The only possible index in the system. *)
Local Definition idx: modal_index :=
    Build_modal_index tt I.
```

Example: Fusion of S4 with S4

Next, we define the fusion system to simply be the fusion of S4 with itself.

```
(* We define X as the fusion of two copies of
   System S4 on idx. *)
Local Definition X :=
   fusion_axioms (S4 idx) (S4 idx).
```

To prove preservation of soundness, we must define the classes of frames relevant for this system, in this case, frames where the relation is a pre-order.

```
(* Condition on frames for X: both need to be pre-orders. *)
Local Definition P: Frame → Prop :=
   fun F ⇒ preorder_frame F idx.
```

Example: Fusion of S4 with S4

Finally, to prove that the fusion is sound, it is enough to prove that each component of the fusion is sound, which can be done simply by using the components of the base library.

```
(* We prove System X is sound from
   soundness of System S4 alone. *)
Goal (* Use P as the condition for both frames. *)
   sound fusion (PF P P) X.
Proof.
   apply soundness_transfer.
   - intros G p ? M ?.
     now apply soundness_S4 with (idx := idx).
   - intros G p ? M ?.
     now apply soundness_S4 with (idx := idx).
Qed.
```

4. Conclusions

Conclusions

- We presented a implementation of the fusion of modal logics in the Coq proof assistant, together with a proof that fusion preserves the property of soundness;
- The modal logic library that served as a basis for this work is highly expressive and allows us to define arbitrary normal modal logics, even nonalethic ones, and has proofs of soundness and weak completeness for the K system, with little effort required to extend those proofs to other modal systems;
- Currently, we know of no other implementation of fusion in a proof assistant, let alone one that proves soundness preservation.

4.1 Future Work

Future Work

SBMF 2024
27th Brazilian Symposium on Formal Methods

1. Carry on improving the base library;
2. Adding the possibility to express bridge principles of fusion, thus allowing the addition of new axioms only to fused systems;
3. Prove the preservation of (at least weak) completeness by fusion, which may be done via the method of stepwise construction of multimodal models presented by Fine and Schurz in [10];
4. Implement different methods of combination, such as fibring [7].

5. References

References

[1] Azurat, A., Prasetya, W.: A survey on embedding programming logics in a theorem prover. Tech. rep., Utrecht University: Institute of Information and Computing Sciences, Utrecht (03 2002)

[2] Benthem, J.v., Bezhanishvili, G., Cate, B.t., Sarenac, D.: Multimodal logics of products of topologies. Studia Logica **84**(3), 369–392 (2006)

[3] Bentzen, B.: A henkin-style completeness proof for the modal logic s5. In: Logic and Argumentation: 4th International Conference, CLAR 2021, Hangzhou, China, October 20–22, 2021, Proceedings 4. pp. 459–467. Springer (2021)

[4] Benzmüller, C.: Combining logics in simple type theory. In: International Workshop on Computational Logic in Multi-Agent Systems. pp. 33–48. Springer, Berlin (2010)

References

[5] Benzmüller, C., Parent, X., van der Torre, L.: Designing normative theories for ethical and legal reasoning: Logikey framework, methodology, and tool support. Artificial Intelligence **287**, 103348 (2020). https://doi.org/https://doi.org/10.1016/j.artint.2020.103348, https://www.sciencedirect.com/science/article/pii/S0004370219301110

[6] Blackburn, P., De Rijke, M., Venema, Y.: Modal logic, vol. 53. Cambridge University Press, Cambridge (2001)

[7] Carnielli, W., Coniglio, M., Gabbay, D.M., Gouveia, P., Sernadas, C.: Analysis and Synthesis of Logics: how to cut and paste reasoning systems, vol. 35. Springer Science & Business Media, Netherlands (2008)

References

[8] Carnielli, W., Coniglio, M.E.: Combining Logics. In: Zalta, E.N. (ed.) The Stanford Encyclopedia of Philosophy. Metaphysics Research Lab, Stanford University, Stanford, Fall 2020 edn. (2020)

[9] Da Silveira, A.A., Ribeiro, R., Nunes, M.A., Torrens, P., Roggia, K.: A sound deep embedding of arbitrary normal modal logics in coq. In: XXVI Brazilian Symposium on Programming Languages. pp. 1–7. SBLP 2022, Association for Computing Machinery, New York, NY, USA (2022). https://doi.org/10.1145/3561320.3561329

[10] Fine, K., Schurz, G.: Transfer theorems for multimodal logics. Logic and reality: essays on the legacy of Arthur Prior pp. 169–213 (1996)

References

[11] Fuenmayor, D., Benzmüller, C.: Harnessing higher-order (meta-)logic to represent and reason with complex ethical theories. In: Nayak, A.C., Sharma, A. (eds.) PRICAI 2019: Trends in Artificial Intelligence. pp. 418–432. Springer International Publishing, Cham (2019)

[12] Fuenmayor, D., Benzmüller, C.: Mechanised assessment of complex natural-language arguments using expressive logic combinations. In: International Symposium on Frontiers of Combining Systems. pp. 112–128. Springer, London (2019)

[13] Gabbay, D.: Many-dimensional Modal Logics: Theory and Applications. Studies in logic and the foundations of mathematics, North Holland Publishing Company, Amsterdam (2003)

References

[14] Kracht, M., Wolter, F.: Properties of independently axiomatizable bimodal logics. The Journal of Symbolic Logic **56**(4), 1469–1485 (1991)

[15] Lescanne, P., Puisségur, J.: Dynamic logic of common knowledge in a proof assistant. arXiv preprint arXiv:0712.3146 (2007)

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

References

[17] Rabe, F.: How to identify, translate and combine logics? Journal of Logic and Computation **27**(6), 1753–1798 (2017)

[18] Scott, D.: Advice on Modal Logic, pp. 143–173. Springer Netherlands, Dordrecht (1970)

Thanks

Our work can be found here: