Non-Iris Proofmode for Temporal Logic (WIP)

Jonas Kastberg Hinrichsen
Aalborg University

Elli Anastasiadi
Aalborg University

8. June, Iris Workshop 2026

Slides:

Work in Progress!

Feedback appreciated :)

Temporal Logic Primer

Temporal logic focus on proving when / if something happens

- Most typically liveness properties
- E.g.: If a request is made, a response will eventually happen

Temporal Logic Primer

Temporal logic focus on proving when / if something happens

- Most typically liveness properties
- E.g.: If a request is made, a response will eventually happen

Temporal logic propositions are typically predicates indexed by traces

- $\mathsf{tProp} \triangleq \mathsf{Trace} \rightarrow \mathsf{Prop}$
- Where Trace is a maximal labelled trace w.r.t. some LTS

Temporal Logic Primer

Temporal logic focus on proving when / if something happens

- Most typically liveness properties
- E.g.: If a request is made, a response will eventually happen

Temporal logic propositions are typically predicates indexed by traces

- $\mathsf{tProp} \triangleq \mathsf{Trace} \rightarrow \mathsf{Prop}$
- Where Trace is a maximal labelled trace w.r.t. some LTS

Most temporal logics are based on modalities, such as

- Next P ($\bigcirc P$): P holds after 1 step of the trace
- Globally P ($\Box P$): P holds for all suffixes of the trace
- Eventually P ($\Diamond P$): P holds for some suffix of the trace

Temporal Logic Primer

Temporal logic focus on proving when / if something happens

- Most typically liveness properties
- E.g.: If a request is made, a response will eventually happen

Temporal logic propositions are typically predicates indexed by traces

- $\mathsf{tProp} \triangleq \mathsf{Trace} \to \mathsf{Prop}$
- Where Trace is a maximal labelled trace w.r.t. some LTS

Most temporal logics are based on modalities, such as

- Next P ($\bigcirc P$): P holds after 1 step of the trace
- Globally P ($\Box P$): P holds for all suffixes of the trace
- Eventually P ($\Diamond P$): P holds for some suffix of the trace

For example:

- $\Box(\mathtt{request} \to \bigcirc \Diamond \mathtt{response}) \vdash \Diamond \mathtt{request} \to \bigcirc \Diamond \mathtt{response}$

Temporal Logic Primer

Temporal logic focus on proving when / if something happens

- Most typically liveness properties
- E.g.: If a request is made, a response will eventually happen

Temporal logic propositions are typically predicates indexed by traces

- $\mathsf{tProp} \triangleq \mathsf{Trace} \to \mathsf{Prop}$
- Where Trace is a maximal labelled trace w.r.t. some LTS

Most temporal logics are based on modalities, such as

- Next P ($\bigcirc P$): P holds after 1 step of the trace
- Globally P ($\Box P$): P holds for all suffixes of the trace
- Eventually P ($\Diamond P$): P holds for some suffix of the trace

For example:

- $\Box(\texttt{request} \to \bigcirc \Diamond \texttt{response}) \vdash \Diamond \texttt{request} \to \bigcirc \Diamond \texttt{response}$
- $\Box(P \to \bigcirc \Diamond Q) \vdash \Diamond P \to \bigcirc \Diamond Q$

Story Time

The need for mechanisation infrastructure for temporal properties

Three Options

Option 1: Going nuclear

- Remove "All results have been mechanised in Rocq"
- Give a paper proof / translation to model checker

Three Options

Option 1: Going nuclear

- Remove "All results have been mechanised in Rocq"
- Give a paper proof / translation to model checker

Option 2: Rolling up your sleeves

- Mechanise proof ad hoc
- Given a custom trace definition with suite of lemmas
- Alas, manually threading trace suffixes is a pain

Three Options

Option 1: Going nuclear

- Remove “All results have been mechanised in Rocq”
- Give a paper proof / translation to model checker

Option 2: Rolling up your sleeves

- Mechanise proof ad hoc
- Given a custom trace definition with suite of lemmas
- Alas, manually threading trace suffixes is a pain

Option 3: Using existing libraries

- Browse selection of existing (LTL) libraries in Rocq
- Realise they primarily focus on metatheory => not many helper lemmas
- Additionally, requires manually managing proof context

Goal: Mechanisation Infrastructure for Temporal Properties

MoSeL

MoSeL: Modal Separation Logic (Proofmode)

- “Extensible Modal Framework for Interactive Proofs (in Separation Logic)”
- Foundation for the Iris Proof Mode

MoSeL: Modal Separation Logic (Proofmode)

- "Extensible Modal Framework for Interactive Proofs (in Separation Logic)"
- Foundation for the Iris Proof Mode

Key features

- Propositional logic à la carte
- Managed proof context for spatial and persistent propositions
- Extensible handling of modality-based reasoning

MoSeL

MoSeL: Modal Separation Logic (Proofmode)

- “Extensible Modal Framework for Interactive Proofs (in Separation Logic)”
- Foundation for the Iris Proof Mode

Key features

- Propositional logic à la carte
- Managed proof context for spatial and persistent propositions
- Extensible handling of modality-based reasoning

Infrastructure based on modal laws

$\triangleright$-MONO
$$\frac{P \vdash Q}{\triangleright P \vdash \triangleright Q}$$

$\triangleright$-SEP
$$\triangleright P * \triangleright Q \dashv\vdash \triangleright(P * Q)$$

$\triangleright$-INTRO
$$P \vdash \triangleright P$$

MoSeL Demo

Key Idea:
Instantiate (and Extend) MoSeL
for Temporal Logic

Key Challenges

Retrofitting MoSeL infrastructure

- Enforces unused step-indexing (later) and resources (sep. conjunction)
- Requires modal law $\mathsf{True} \vdash \mathsf{M}\,\mathsf{True}$ for any modality M
- Does not support all conventional modal laws, e.g. those of eventually $\Diamond$

Key Challenges

Retrofitting MoSeL infrastructure

- Enforces unused step-indexing (later) and resources (sep. conjunction)
- Requires modal law $\mathsf{True} \vdash \mathsf{M}\,\mathsf{True}$ for any modality M
- Does not support all conventional modal laws, e.g. those of eventually $\Diamond$

Defining "persistently": $\boxdot P$

- Unclear if there is a canonical choice (alike Iris)
- Intuition is that P holds "independently" (in Iris, without owning anything)
- Current choice is temporal logic "globally": $\boxdot P \triangleq \Box P$
- Alternative choice is $\boxdot P \triangleq \lambda\_.\, \forall tr.\, P\, tr$

Key Challenges

Retrofitting MoSeL infrastructure

- Enforces unused step-indexing (later) and resources (sep. conjunction)
- Requires modal law $\mathsf{True} \vdash \mathsf{M}\,\mathsf{True}$ for any modality M
- Does not support all conventional modal laws, e.g. those of eventually $\Diamond$

Defining "persistently": $\boxdot P$

- Unclear if there is a canonical choice (alike Iris)
- Intuition is that P holds "independently" (in Iris, without owning anything)
- Current choice is temporal logic "globally": $\boxdot P \triangleq \Box P$
- Alternative choice is $\boxdot P \triangleq \lambda\_.\, \forall tr.\, P\, tr$

Defining model of traces

- Infinite vs possibly-infinite maximal traces

Steps Towards a Temporal Logic Proofmode

Trace Logic Foundation

Possibly-infinite maximal traces over a relation $R : S \to L \to S \to \mathsf{Prop}$

- $tr \in \mathsf{Trace} \triangleq \langle s \rangle \mid s \xrightarrow{\ell} tr \mid \bot$
- Bottom element is an absorbing state for terminated traces

Trace Logic Foundation

Possibly-infinite maximal traces over a relation $R : S \to L \to S \to \mathsf{Prop}$

- $tr \in \mathsf{Trace} \triangleq \langle s \rangle \mid s \xrightarrow{\ell} tr \mid \bot$
- Bottom element is an absorbing state for terminated traces

Propositions indexed by traces with expected definitions of usual connectives

- $P, Q \in \mathsf{tProp} \triangleq \mathsf{Trace} \to \mathsf{Prop}$
- $P \vdash Q \triangleq \forall tr.\, P\, tr \implies Q\, tr$
- $P \wedge Q \triangleq \lambda tr.\, P\, tr \wedge Q\, tr \qquad P \to Q \triangleq \lambda tr.\, P\, tr \to Q\, tr$

Trace Logic Foundation

Possibly-infinite maximal traces over a relation $R : S \to L \to S \to \mathsf{Prop}$

- $tr \in \mathsf{Trace} \triangleq \langle s \rangle \mid s \xrightarrow{\ell} tr \mid \bot$
- Bottom element is an absorbing state for terminated traces

Propositions indexed by traces with expected definitions of usual connectives

- $P, Q \in \mathsf{tProp} \triangleq \mathsf{Trace} \to \mathsf{Prop}$
- $P \vdash Q \triangleq \forall tr.\, P\, tr \implies Q\, tr$
- $P \wedge Q \triangleq \lambda tr.\, P\, tr \wedge Q\, tr \qquad P \to Q \triangleq \lambda tr.\, P\, tr \to Q\, tr$

Let sep. logic connectives coincide with propositional counterparts

- $P * Q \triangleq P \wedge Q \qquad P \mathrel{-\!\!*} Q \triangleq P \to Q$

Trace Logic Foundation

Possibly-infinite maximal traces over a relation $R : S \to L \to S \to \mathsf{Prop}$

- $tr \in \mathsf{Trace} \triangleq \langle s \rangle \mid s \xrightarrow{\ell} tr \mid \bot$
- Bottom element is an absorbing state for terminated traces

Propositions indexed by traces with expected definitions of usual connectives

- $P, Q \in \mathsf{tProp} \triangleq \mathsf{Trace} \to \mathsf{Prop}$
- $P \vdash Q \triangleq \forall tr.\, P\, tr \implies Q\, tr$
- $P \land Q \triangleq \lambda tr.\, P\, tr \land Q\, tr \qquad P \to Q \triangleq \lambda tr.\, P\, tr \to Q\, tr$

Let sep. logic connectives coincide with propositional counterparts

- $P * Q \triangleq P \land Q \qquad P \mathrel{-\!\!*} Q \triangleq P \to Q$

Let later modality be the identity

- $\triangleright P \triangleq P$

Trace Logic Foundation

Possibly-infinite maximal traces over a relation $R : S \to L \to S \to \mathrm{Prop}$

- $tr \in \mathrm{Trace} \triangleq \langle s \rangle \mid s \xrightarrow{\ell} tr \mid \bot$
- Bottom element is an absorbing state for terminated traces

Propositions indexed by traces with expected definitions of usual connectives

- $P, Q \in \mathrm{tProp} \triangleq \mathrm{Trace} \to \mathrm{Prop}$
- $P \vdash Q \triangleq \forall tr.\, P\, tr \implies Q\, tr$
- $P \wedge Q \triangleq \lambda tr.\, P\, tr \wedge Q\, tr \qquad P \to Q \triangleq \lambda tr.\, P\, tr \to Q\, tr$

Let sep. logic connectives coincide with propositional counterparts

- $P * Q \triangleq P \wedge Q \qquad P \mathrel{-\!\!*} Q \triangleq P \to Q$

Let later modality be the identity

- $\triangleright P \triangleq P$

Persistence is defined as globally

- $\boxdot P \triangleq \Box P$

Temporal Modalities

Modalities are inductively / coinductively defined with expected semantics

- Next: $\bigcirc P \equiv \lambda tr.\ P\,(\mathsf{drop}\ 1\ tr)$
- Eventually: $\Diamond P \equiv \lambda tr.\ \exists n.\ P\,(\mathsf{drop}\ n\ tr)$
- Globally: $\Box P \equiv \lambda tr.\ \forall n.\ P\,(\mathsf{drop}\ n\ tr)$

Temporal Modalities

Modalities are inductively / coinductively defined with expected semantics

- Next: $\bigcirc P \equiv \lambda tr.\, P\,(\mathsf{drop}\ 1\ tr)$
- Eventually: $\Diamond P \equiv \lambda tr.\, \exists n.\, P\,(\mathsf{drop}\ n\ tr)$
- Globally: $\square P \equiv \lambda tr.\, \forall n.\, P\,(\mathsf{drop}\ n\ tr)$

Possibility of finite traces complicates picture

- Recall required modal laws: $\mathsf{True} \vdash \bigcirc \mathsf{True}$ and $\mathsf{True} \vdash \square \mathsf{True}$
- Propositions on successive elements must be defined for singleton trace $\langle s \rangle$
  - $(\bigcirc P)\,\langle s \rangle \equiv P \bot \quad (\bigcirc P)\,\bot \equiv P \bot \qquad (\square P)\,\langle s \rangle \equiv P\,\langle s \rangle \wedge (\square P)\,\bot \quad (\square P)\,\bot \equiv P\,\bot$
- Yields crucial modal laws, as $\mathsf{True}\,\bot$ holds trivially

Temporal Modalities

Modalities are inductively / coinductively defined with expected semantics

- Next: $\bigcirc P \equiv \lambda tr.\, P\,(\mathsf{drop}\,1\,tr)$
- Eventually: $\Diamond P \equiv \lambda tr.\, \exists n.\, P\,(\mathsf{drop}\,n\,tr)$
- Globally: $\Box P \equiv \lambda tr.\, \forall n.\, P\,(\mathsf{drop}\,n\,tr)$

Possibility of finite traces complicates picture

- Recall required modal laws: $\mathsf{True} \vdash \bigcirc \mathsf{True}$ and $\mathsf{True} \vdash \Box \mathsf{True}$
- Propositions on successive elements must be defined for singleton trace $\langle s \rangle$
  - $(\bigcirc P)\,\langle s \rangle \equiv P \bot \quad (\bigcirc P) \bot \equiv P \bot \qquad (\Box P)\,\langle s \rangle \equiv P\,\langle s \rangle \wedge (\Box P) \bot \quad (\Box P) \bot \equiv P \bot$
- Yields crucial modal laws, as $\mathsf{True} \bot$ holds trivially

Modal laws of temporal modalities are partially supported by MoSeL

Modal Laws (an excerpt)

Globally ($\square$) vs. Iris Persistent ($\boxdot$)

$\square$-MONO

$$\frac{P \vdash Q}{\square P \vdash \square Q}$$

$\square$-AND

$$\square P \land \square Q \dashv\vdash \square(P \land Q)$$

$\square$-TAUT

$$\mathrm{True} \vdash \square\, \mathrm{True}$$

$\square$-EXISTS

$$\square\, \exists x.\, P \nvdash \exists x.\, \square\, P$$

Modal Laws (an excerpt)

Globally ($\Box$) vs. Iris Persistent ($\boxdot$)

$\Box$-MONO
$$\frac{P \vdash Q}{\Box P \vdash \Box Q}$$

$\Box$-AND
$$\Box P \land \Box Q \dashv\vdash \Box(P \land Q)$$

$\Box$-TAUT
$$\mathsf{True} \vdash \Box \mathsf{True}$$

$\Box$-EXISTS
$$\Box \exists x.\, P \nvdash \exists x.\, \Box P$$

Next ($\bigcirc$) vs. Iris Later ($\triangleright$)

$\bigcirc$-MONO
$$\frac{P \vdash Q}{\bigcirc P \vdash \bigcirc Q}$$

$\bigcirc$-AND
$$\bigcirc P \land \bigcirc Q \dashv\vdash \bigcirc(P \land Q)$$

$\bigcirc$-TAUT
$$\mathsf{True} \vdash \bigcirc \mathsf{True}$$

$\bigcirc$-INTRO
$$P \nvdash \bigcirc P$$

Modal Laws (an excerpt)

Globally ($\square$) vs. Iris Persistent ($\boxdot$)

□-MONO
$$\frac{P \vdash Q}{\square P \vdash \square Q}$$

□-AND
$$\square P \wedge \square Q \dashv\vdash \square(P \wedge Q)$$

□-TAUT
$$\text{True} \vdash \square\, \text{True}$$

□-EXISTS
$$\square\, \exists x.\, P \nvdash \exists x.\, \square\, P$$

Next ($\bigcirc$) vs. Iris Later ($\triangleright$)

◯-MONO
$$\frac{P \vdash Q}{\bigcirc P \vdash \bigcirc Q}$$

◯-AND
$$\bigcirc P \wedge \bigcirc Q \dashv\vdash \bigcirc(P \wedge Q)$$

◯-TAUT
$$\text{True} \vdash \bigcirc\, \text{True}$$

◯-INTRO
$$P \nvdash \bigcirc P$$

Eventually ($\Diamond$) vs. Iris Update ($\Rrightarrow$)

◇-MONO
$$\frac{P \vdash Q}{\Diamond P \vdash \Diamond Q}$$

◇-IDEMP
$$\Diamond P \dashv\vdash \Diamond \Diamond P$$

◇-AND-L
$$\Diamond(P \wedge Q) \vdash \Diamond P \wedge \Diamond Q$$

◇-AND-R
$$\Diamond P \wedge \Diamond Q \nvdash \Diamond(P \wedge Q)$$

Temporal Logic Proofmode Demo

Reflections on Definition of Persistent

Current definition of persistent ($\boxdot P \triangleq \Box P$) is incompatible with MoSeL

- MoSeL requires invalid law: $\Box \exists x.\, P \not\models \exists x.\, \Box P$
- But law might not be strictly required; mostly used in step-indexing

Reflections on Definition of Persistent

Current definition of persistent ($\boxdot P \triangleq \Box P$) is incompatible with MoSeL

- MoSeL requires invalid law: $\Box \exists x.\, P \not\vdash \exists x.\, \Box P$
- But law might not be strictly required; mostly used in step-indexing

Alternative definition ($\boxdot P \triangleq \lambda\_.\, \forall tr.\, P\, tr$) is unsatisfactory

- Does not let us leverage MoSeL infrastructure for persistent
- Requires treating $\Box$ as a separate modality
- Incompatible with current tooling for $\Diamond$
- Yields weaker logic-internal fixpoint theorems

More than just Linear Temporal Logic (LTL): Fixpoints

We have least (μ) and greatest (ν) fixpoints for any function monotone in $\boxdot$

- This is the case for any MoSeL instantiation

More than just Linear Temporal Logic (LTL): Fixpoints

We have least (μ) and greatest (ν) fixpoints for any function monotone in $\boxdot$

- This is the case for any MoSeL instantiation

Standard LTL constructions and more can be derived as follows

- $\Diamond P \triangleq \mu X.\, P \vee \bigcirc X$
- $P \cup Q \triangleq \mu X.\, Q \vee (P \wedge \bigcirc X)$
- $\widehat{\Box} P \triangleq \nu X.\, P \wedge \bigcirc X$
- $\texttt{every\_other}\, P \triangleq \nu X.\, P \wedge \bigcirc \bigcirc X$

More than just Linear Temporal Logic (LTL): Fixpoints

We have least (μ) and greatest (ν) fixpoints for any function monotone in $\boxdot$

- This is the case for any MoSeL instantiation

Standard LTL constructions and more can be derived as follows

- $\Diamond P \triangleq \mu X.\, P \lor \bigcirc X$
- $P \cup Q \triangleq \mu X.\, Q \lor (P \land \bigcirc X)$
- $\widehat{\Box} P \triangleq \nu X.\, P \land \bigcirc X$
- $\texttt{every\_other}\, P \triangleq \nu X.\, P \land \bigcirc \bigcirc X$

Monotonicity of the above w.r.t $\boxdot$ ($\triangleq \Box$) arise from monotonicity of $\bigcirc$ in $\Box$:

- $\Box(P \to Q) \vdash \bigcirc P \to \bigcirc Q$

More than just Linear Temporal Logic (LTL): Fixpoints

We have least (μ) and greatest (ν) fixpoints for any function monotone in $\boxdot$

- This is the case for any MoSeL instantiation

Standard LTL constructions and more can be derived as follows

- $\Diamond P \triangleq \mu X.\, P \vee \bigcirc X$
- $P \cup Q \triangleq \mu X.\, Q \vee (P \wedge \bigcirc X)$
- $\widehat{\square} P \triangleq \nu X.\, P \wedge \bigcirc X$
- $\texttt{every\_other}\, P \triangleq \nu X.\, P \wedge \bigcirc \bigcirc X$

Monotonicity of the above w.r.t $\boxdot$ ($\triangleq \square$) arise from monotonicity of $\bigcirc$ in $\square$:

- $\square(P \rightarrow Q) \vdash \bigcirc P \rightarrow \bigcirc Q$

Fixpoint theorems inherit choice of $\boxdot$

- Theorem: $\boxdot((P \vee \bigcirc Q) \rightarrow Q) \vdash \Diamond P \rightarrow Q$
- Stronger $\boxdot$: more fixpoints, weaker theorems (e.g. $\boxdot\, P \triangleq \forall tr.\, P\, tr$)
- Weaker $\boxdot$: less fixpoints, stronger theorems (e.g. $\boxdot\, P \triangleq \square\, P$)

More than just Linear Temporal Logic (LTL): Quantifiers

Temporal logic does not usually have impredicative quantification

- $\forall x.\, P \qquad \exists x.\, P$

More than just Linear Temporal Logic (LTL): Quantifiers

Temporal logic does not usually have impredicative quantification

- $\forall x.\, P \qquad \exists x.\, P$

Working in Rocq (and MoSeL), we get these for free

- $\forall x.\, P \triangleq \lambda tr.\, \forall x.\, P\, tr \qquad \exists x.\, P \triangleq \lambda tr.\, \exists x.\, P\, tr$

More than just Linear Temporal Logic (LTL): Quantifiers

Temporal logic does not usually have impredicative quantification

- $\forall x.\, P \qquad \exists x.\, P$

Working in Rocq (and MoSeL), we get these for free

- $\forall x.\, P \triangleq \lambda tr.\, \forall x.\, P\; tr \qquad \exists x.\, P \triangleq \lambda tr.\, \exists x.\, P\; tr$

But unclear if there is a good use case for this

- Iris-style abstract predicate use case seem to be about abstracting permissions
- We have yet to come up with a good example
- Ideas welcome :)!

To Conclude

We Need Help!

Settling on a definition of persistent

- Current definition is incompatible with required law: $\square \exists x.\, P \nvdash \exists x.\, \square\, P$
- Alternate does not allow us to leverage MoSeL persistent context for globally

We Need Help!

Settling on a definition of persistent

- Current definition is incompatible with required law: $\Box \exists x.\, P \nvdash \exists x.\, \Box P$
- Alternate does not allow us to leverage MoSeL persistent context for globally

Motivating / killer examples? Preferably using:

- Impredicative quantifiers
- Custom fixpoints

We Need Help!

Settling on a definition of persistent

- Current definition is incompatible with required law: $\Box\, \exists x.\, P \nvdash \exists x.\, \Box\, P$
- Alternate does not allow us to leverage MoSeL persistent context for globally

Motivating / killer examples? Preferably using:

- Impredicative quantifiers
- Custom fixpoints

Eyeing opportunities

- Step-indexing?
- Separation logic resources?
- Further generalisation of MoSeL modality infrastructure?

We Need Help!

Settling on a definition of persistent

- Current definition is incompatible with required law: $\square \exists x.\, P \nvdash \exists x.\, \square\, P$
- Alternate does not allow us to leverage MoSeL persistent context for globally

Motivating / killer examples? Preferably using:

- Impredicative quantifiers
- Custom fixpoints

Eyeing opportunities

- Step-indexing?
- Separation logic resources?
- Further generalisation of MoSeL modality infrastructure?

So far mostly an engineering project..

- Experiences in pitching proofmode projects appreciated

Conclusion

MoSeL seems mature enough to handle new domain logics

- Relatively easy to instantiate
- Framework can handle functionality beyond original intent
- Albeit some creativity is required

Conclusion

MoSeL seems mature enough to handle new domain logics

- Relatively easy to instantiate
- Framework can handle functionality beyond original intent
- Albeit some creativity is required

Modal temporal logic has correspondences to Iris-style reasoning

- Foundations in similar modal laws
- Albeit with key distinctions

Conclusion

MoSeL seems mature enough to handle new domain logics

- Relatively easy to instantiate
- Framework can handle functionality beyond original intent
- Albeit some creativity is required

Modal temporal logic has correspondences to Iris-style reasoning

- Foundations in similar modal laws
- Albeit with key distinctions

First(?) steps towards temporal separation logic

- Conceptually stimulating
- Practically unclear what it even means

$$\vdash \Box(\mathit{Question} \rightarrow \Diamond\, \mathit{Answer})$$

Rocq code available at: