Interactive Proofmode for Temporal Logic in Rocq

Jonas Kastberg Hinrichsen
Aalborg University

Elli Anastasiadi
Aalborg University

20. August, CoPLaWS 2026

Slides:

Context:

We care about mechanisation and liveness

Problem:

Limited support for mechanised temporal properties

Solution:

Interactive Proofmode for Temporal Logic in Rocq

Context:

We care about mechanisation and liveness

Liveness Properties and Temporal Logic

Liveness: something good eventually happens

- E.g.: If a request is made, a response will <u>eventually</u> happen
- Often studied in terms of temporal logic

Liveness Properties and Temporal Logic

Liveness: something good eventually happens

- E.g.: If a request is made, a response will eventually happen
- Often studied in terms of temporal logic

Temporal logic: propositions indexed by traces

- $P \vdash Q \triangleq \forall tr.\, P\, tr \rightarrow Q\, tr$ where $P, Q \in \texttt{Trace} \rightarrow \mathsf{Prop}$
- Here, `Trace` is a maximal labelled trace w.r.t. some LTS

Liveness Properties and Temporal Logic

Liveness: something good eventually happens

- E.g.: If a request is made, a response will eventually happen
- Often studied in terms of temporal logic

Temporal logic: propositions indexed by traces

- $P \vdash Q \triangleq \forall tr.\, P\, tr \to Q\, tr$ where $P, Q \in \mathtt{Trace} \to \mathsf{Prop}$
- Here, `Trace` is a maximal labelled trace w.r.t. some LTS

Most temporal logics are based on modalities, such as

- Next P: $\circ P \equiv \lambda tr.\, P(\mathsf{tail}\, tr)$
- Globally P: $\Box P \equiv \lambda tr.\, \forall n.\, P(\mathsf{tail}^n\, tr)$
- Eventually P: $\Diamond P \equiv \lambda tr.\, \exists n.\, P(\mathsf{tail}^n\, tr)$

Liveness Properties and Temporal Logic

Liveness: something good eventually happens

- E.g.: If a request is made, a response will eventually happen
- Often studied in terms of temporal logic

Temporal logic: propositions indexed by traces

- $P \vdash Q \triangleq \forall tr.\, P\, tr \to Q\, tr$ where $P, Q \in \texttt{Trace} \to \mathsf{Prop}$
- Here, `Trace` is a maximal labelled trace w.r.t. some LTS

Most temporal logics are based on modalities, such as

- Next P: $\circ P \equiv \lambda tr.\, P(\mathsf{tail}\, tr)$
- Globally P: $\Box P \equiv \lambda tr.\, \forall n.\, P(\mathsf{tail}^n\, tr)$
- Eventually P: $\Diamond P \equiv \lambda tr.\, \exists n.\, P(\mathsf{tail}^n\, tr)$

For example:

- $\Box(\texttt{request} \to \circ \Diamond \texttt{response}) \vdash \Diamond \texttt{request} \to \circ \Diamond \texttt{response}$

Liveness Properties and Temporal Logic

Liveness: something good eventually happens

- E.g.: If a request is made, a response will eventually happen
- Often studied in terms of temporal logic

Temporal logic: propositions indexed by traces

- $P \vdash Q \triangleq \forall tr.\, P\, tr \to Q\, tr$ where $P, Q \in \mathtt{Trace} \to \mathsf{Prop}$
- Here, `Trace` is a maximal labelled trace w.r.t. some LTS

Most temporal logics are based on modalities, such as

- Next P: $\circ P \equiv \lambda tr.\, P(\mathsf{tail}\, tr)$
- Globally P: $\Box P \equiv \lambda tr.\, \forall n.\, P(\mathsf{tail}^n\, tr)$
- Eventually P: $\Diamond P \equiv \lambda tr.\, \exists n.\, P(\mathsf{tail}^n\, tr)$

For example:

- $\Box(\mathtt{request} \to \circ \Diamond \mathtt{response}) \vdash \Diamond \mathtt{request} \to \circ \Diamond \mathtt{response}$
- $\Box(P \to \circ \Diamond Q) \vdash \Diamond P \to \circ \Diamond Q$

Mechanisation

Mechanisation: machine-checked proofs

- Trust through rigor; no stone unturned
- Reusable proof infrastructure
- Increasingly becoming a community expectation

Mechanisation

Mechanisation: machine-checked proofs

- Trust through rigor; no stone unturned
- Reusable proof infrastructure
- Increasingly becoming a community expectation

Interactive theorem provers (ITPs): Rocq, Lean, Isabelle, ...

- Step-wise deduction of proof, with interactive proof context
- Time consuming, especially with limited support

Mechanisation

Mechanisation: machine-checked proofs

- Trust through rigor; no stone unturned
- Reusable proof infrastructure
- Increasingly becoming a community expectation

Interactive theorem provers (ITPs): Rocq, Lean, Isabelle, ...

- Step-wise deduction of proof, with interactive proof context
- Time consuming, especially with limited support

Support is limited for temporal logic

- Primarily studied in more automated verification (e.g. model checking)
- Usage alongside ITPs introduces a gap in the trusted computing base

Problem:

Limited support for mechanised temporal properties

Three Immediate Options

Option 1: Going nuclear

- Give a paper proof / translation to model checker
- Increases the trusted computing base
- Remove "All results have been mechanised in Rocq"

Three Immediate Options

Option 1: Going nuclear

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

Option 2: Rolling up your sleeves

- Mechanise proof ad hoc
- Given a custom trace definition with suite of lemmas

Three Immediate Options

Option 1: Going nuclear

- Give a paper proof / translation to model checker
- Increases the trusted computing base
- Remove "All results have been mechanised in Rocq"

Option 2: Rolling up your sleeves

- Mechanise proof ad hoc
- Given a custom trace definition with suite of lemmas

Option 3: Using existing libraries

- Browse selection of existing (LTL) libraries in Rocq
- Realise they primarily focus on metatheory => not many helper lemmas
- Additionally, working with an object logic is very verbose

Solution:

Interactive Proofmode for Temporal Logic in Rocq

Contributions

Introducing Temporal Logic Proof Mode (TPM)

- Proofmode context for temporal (object) logic
- Tactics for working with modalities

Modality-compatible model of possibly-infinite well-formed traces

- Next ($\circ$) and now ($\downarrow$) modalities for possiblity-infinite traces
- Well-behaved derivation of globally, eventually, and until via next

Support for least and greatest fixpoints

- Define logic-level fixpoints with fixpoint theorems for free
- E.g. $P \cup Q \triangleq \mu X.\, Q \vee (P \wedge \circ X)$

Infrastructure for instantiation with custom LTS models

- Derive logic-level axioms in terms of LTS stepping relation
- Extract meta-level theorems about traces

Proof of eventual delivery for model of Stenning protocol

- In an unreliable network with a fair delivery assumption

Temporal Logic Proof Mode Demo

Developing the Temporal Logic Proof Mode

MoSeL: Modal Separation Logic (Proofmode)

- “Extensible Modal Framework for Interactive Proofs (in Separation Logic)”
- Foundation for the Iris Proof Mode (facilitating 150+ papers since 2015)
- Now also foundation for the Temporal Logic Proof Mode

Developing the Temporal Logic Proof Mode

MoSeL: Modal Separation Logic (Proofmode)

- "Extensible Modal Framework for Interactive Proofs (in Separation Logic)"
- Foundation for the Iris Proof Mode (facilitating 150+ papers since 2015)
- Now also foundation for the Temporal Logic Proof Mode

Key features

- Propositional logic à la carte
- Managed proof context for spatial and persistent propositions
- Extensible handling of modality-based reasoning
- Least and greatest monotone fixpoint constructions
- (Opt-in) step-indexing and separation logic

Developing the Temporal Logic Proof Mode

MoSeL: Modal Separation Logic (Proofmode)

- "Extensible Modal Framework for Interactive Proofs (in Separation Logic)"
- Foundation for the Iris Proof Mode (facilitating 150+ papers since 2015)
- Now also foundation for the Temporal Logic Proof Mode

Key features

- Propositional logic à la carte
- Managed proof context for spatial and persistent propositions
- Extensible handling of modality-based reasoning
- Least and greatest monotone fixpoint constructions
- (Opt-in) step-indexing and separation logic

Infrastructure based on modal laws

M-AND
$$M\,P \land M\,Q \vdash M\,(P \land Q)$$

M-MONO
$$\frac{P \vdash Q}{M\,P \vdash M\,Q}$$

M-INTRO
$$\mathsf{True} \vdash M\,\mathsf{True}$$

To Conclude

Conclusion

Interactive proofs of temporal properties is beneficial

- Avoids an extended TCB by using external tools

Conclusion

Interactive proofs of temporal properties is beneficial

- Avoids an extended TCB by using external tools

Modal temporal logic is suitable for interactive verification

- Modal laws allow intuitive proof flows

Conclusion

Interactive proofs of temporal properties is beneficial

- Avoids an extended TCB by using external tools

Modal temporal logic is suitable for interactive verification

- Modal laws allow intuitive proof flows

MoSeL seems mature enough to handle new domain logics

- Relatively easy to instantiate
- Framework can handle functionality beyond original intent
- I encourage trying it out!

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

Rocq code available at: