# A proof theory for model checking

Dale Miller

Inria Saclay & LIX, École Polytechnique
Palaiseau, France

Linearity 2016, Porto, Portugal, 25 June 2016

Linear Logic 2016, Lyon, 9 November 2016

Joint work with Quentin Heath. Extended draft available on web.

# Historically speaking

Model checking was introduced in the early 1980's as a way to establish properties about (concurrent) computer programs that were hard or impossible to do then using traditional, axiomatic proof techniques of Floyd and Hoare.

If you cannot prove a property, at least you can get help looking for counterexamples: eg, a path to a state where two processes are both in their critical section.

I will argue that model checking can, in fact, be given an appealing proof theoretical foundation.

# Proof theoretical ingredients

Some historical high-points.

- Gentzen's sequent calculus [1935]
- Girard's linear logic [1987]
- Andreoli's focused proof system for linear logic [1991]
- Schroeder-Heister's and Girard's treatment of equality and fixed points [1992]
- Baelde, McDowell, M, and Tiu developed proof theory for least and greatest fixed points [1997-2012].

All these developments except for the first item come after the start of model checking.

# Why promote a proof theoretic framework?

Proof theory can suggest certificates for model checking.

It should be conceptually easier to integrate inductive theorem provers and model checking since they operate in the same (or similar) logics.

Generalization of model checking are natural to consider. For example, Bedwyr generalized model checking by allowing linguistic expressions including bindings within states.

A proof-theoretic framework for logic programming was proposed in 1986. This work extends that work.

# Three ways to move beyond MALL

1. Girard [1987] added the *exponentials* to get linear logic.
2. Liang and M [2009] added *classical and intuitionistic connectives* to get LKU. (Exponentials are behind this design.)
3. Baelde and M [2007] added *fixed points* to get $\mu$MALL.

A goal of this talk is to illustrate how (first-order) $\mu$MALL is better suited for model checking and inductive/coinductive theorem proving than linear logic.

# Three ways to move beyond MALL

1. Girard [1987] added the *exponentials* to get linear logic.
2. Liang and M [2009] added *classical and intuitionistic connectives* to get LKU. (Exponentials are behind this design.)
3. Baelde and M [2007] added *fixed points* to get $\mu$MALL.

A goal of this talk is to illustrate how (first-order) $\mu$MALL is better suited for model checking and inductive/coinductive theorem proving than linear logic.

Note:

- Fixed point unfolding resembles contraction: $\mu B\bar{t} = B(\mu B)\bar{t}$.
- In linear logic, it is the case that $B \equiv !\,B$ when $B$ is *purely positive*. In linear logic, this is a trivial collection of formulas; in $\mu$MALL, this is a rich collection of formulas.

# What is an additive inference rule?

Truth is central to the way that model checking is understood. The notion of *additive* inference rules seem to be a treatment of truth.

- Linear logic provides examples of additive inference rules.
- Hintikka games provide another treatment: two player use a board containing one formula.
- No comma is needed in a sequent, etc.

Instead of attempting a definition, we state four properties of a class of additive connectives that seem desirable to maintain.

# Additive connectives

Let $\mathcal{A}$ be the set of formulas built from the propositional connectives $\{\wedge, \mathit{tt}, \vee, \mathit{ff}\}$ (no propositional constants included).

Consider the proof system given by the following one-sided sequent calculus inference rules.

$$\frac{\vdash B_1, \Delta \quad \vdash B_2, \Delta}{\vdash B_1 \wedge B_2, \Delta} \qquad \frac{}{\vdash \mathit{tt}, \Delta} \qquad \frac{\vdash B_1, \Delta}{\vdash B_1 \vee B_2, \Delta} \qquad \frac{\vdash B_2, \Delta}{\vdash B_1 \vee B_2, \Delta}$$

Here, $\mathit{tt}$ is the unit of $\wedge$, and $\mathit{ff}$ is the unit of $\vee$. Notice that $\vee$ has two introduction rules while $\mathit{ff}$ has none,

The de Morgan duals are $\mathit{tt}$ / $\wedge$ with $\mathit{ff}$ / $\vee$. By $\neg B$ we mean the de Morgan dual of all connectives in $B$.

The multiset $\Delta$ is provable if and only if there is a proof of $\vdash \Delta$ using these inference rules.

# Some properties of additives

Let $\Delta$, $\Delta_1$, $\Delta_2$ be multisets of $\mathcal{A}$-formulas and let $B$ be an $\mathcal{A}$-formula.

**Theorem (Strengthening)**

*If $\vdash \Delta$ has a proof, then there is a $B \in \Delta$ such that $\vdash B$.*

**Theorem (Weakening & contraction admissibility)**

*If $\Delta_1 \subseteq \Delta_2$ and $\vdash \Delta_1$ is provable then $\vdash \Delta_2$ is provable.*

**Theorem (Initial admissibility)**

*$\vdash B, \neg B$ is provable.*

**Theorem (Cut admissibility)**

*If $\vdash B, \Delta_1$ and $\vdash \neg B, \Delta_2$, then $\vdash \Delta_1, \Delta_2$.*

# Truth-tables evaluation

These properties allow the following definition.

Define $v(\cdot) : \mathcal{A} \longrightarrow \{tt, ff\}$ such that

- $v(B) = tt$ if $\vdash B$ is provable and
- $v(B) = ff$ if $\vdash \neg B$ is provable.

Initial admissibility implies that $v(\cdot)$ is total.

Cut admissibility implies that $v(\cdot)$ is functional.

The introduction rules yield the truth table definition for $v(\cdot)$:

e.g., $v(A \wedge B)$ is the truth-functional conjunction of $v(A)$ and $v(B)$ (similarly for $\vee$).

Of course, the logic of $\mathcal{A}$-formulas is essentially trivial. When we build synthetic rules that must also be additive, we will arrange that their provability is independent of their context.

# Term equality and quantification

A *ranked signature* $\Sigma$ associates to every constructor a natural number indicating that constructor's arity.

A $\Sigma$-*term* is a (closed) term built from only constructors in $\Sigma$ and obeying the rank restrictions.

For example, if $\Sigma$ is $\{a/0, b/0, f/1, g/2\}$, then $a$, $(f\ a)$, and $(g\ (f\ a)\ b)$ are all $\Sigma$-terms.

$$\frac{}{\vdash t = t, \Delta} \qquad \frac{}{\vdash t \neq s, \Delta}\ t \text{ and } s \text{ differ}$$

Here, $t$ and $s$ are $\Sigma$-terms for some ranked signature $\Sigma$.

$$\frac{\vdash B[t/x], \Delta}{\vdash \exists x.B, \Delta}\ \exists \qquad \frac{\{\ \vdash B[t/x], \Delta \ \mid\ \Sigma\text{-term } t\ \}}{\vdash \forall x.B, \Delta}\ \forall\text{-ext}$$

All of these inference rules are additive but at the cost of using infinitary proofs.

# There is no algorithm here

Let $\Sigma$ contain the ranked symbols $z/0$ and $s/1$. Abbreviate $z$, $(s\ z)$, $(s\ (s\ z))$, $(s\ (s\ (s\ z)))$, etc by $\mathbf{0}$, $\mathbf{1}$, $\mathbf{2}$, $\mathbf{3}$, etc.

Let $A$ and $B$ be the set of terms $\{\mathbf{0}, \mathbf{1}\}$ and $\{\mathbf{0}, \mathbf{1}, \mathbf{2}\}$, respectively. These sets can be encoded as the predicate expressions $\lambda x.\ x = \mathbf{0} \vee x = \mathbf{1}$ and $\lambda x.\ x = \mathbf{0} \vee x = \mathbf{1} \vee x = \mathbf{2}$.

The fact that $A$ is a subset of $B$ can be denoted by the formula $\forall x. \neg(A\,x) \vee B\,x$ or, equivalently, as

$$\forall x. (x \neq \mathbf{0} \wedge x \neq \mathbf{1}) \vee x = \mathbf{0} \vee x = \mathbf{1} \vee x = \mathbf{2}$$

Proving this formula requires an infinite number of premises of the form $(t \neq \mathbf{0} \wedge t \neq \mathbf{1}) \vee t = \mathbf{0} \vee t = \mathbf{1} \vee t = \mathbf{2}$.

# Multiplicative connectives: implication and conjunction

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

The strengthening property is lost: $\vdash (p \supset q), p$.

Currying $A \supset B \supset C \equiv (A \wedge B) \supset C$ yields a multiplicative conjunction.

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

For symmetry, we rename $\wedge$ as $\wedge^-$ and $tt$ to $tt^-$.

In linear logic, one writes &, $\top$, $\otimes$, $\mathbf{1}$ for $\wedge^-$, $tt^-$, $\wedge^+$, $tt^+$.

Similar, $\supset$ corresponds to $\multimap$ in linear logic. The multiplicative disjunction plays no central role here. The multiplicative false $ff^-$ exists as $t \neq t$ (for closed term $t$).

# Multiplicative connectives: quantifiers and eigenvariables

The multiplicative treatment of quantifiers employs *eigenvariables*.

Let the set $\mathcal{X}$ denote *first-order variables*.

Let $\Sigma(\mathcal{X})$ denote all terms built from constructors in $\Sigma$ and from the variables $\mathcal{X}$: variables act as constructors of arity 0.

Sequents are now written as $\mathcal{X}\,;\,\Gamma \vdash \Delta$: the variables in $\mathcal{X}$ are bound over the formulas in $\Gamma$ and $\Delta$: formulas in $\Gamma$ and $\Delta$ are $\Sigma(\mathcal{X})$-formulas.

$$\frac{\mathcal{X}\,;\,\Gamma \vdash B[t/x], \Delta}{\mathcal{X}\,;\,\Gamma \vdash \exists x.B, \Delta}\ \exists \qquad \frac{\mathcal{X}, y\,;\,\Gamma \vdash B[y/x], \Delta}{\mathcal{X}\,;\,\Gamma \vdash \forall x.B, \Delta}\ \forall$$

where $t$ is a $\Sigma(\mathcal{X})$-term and $y \notin \Sigma$. Dually, for the left introduction rules.

# Equality with open terms

When $t$ and $s$ are not unifiable,

$$\frac{}{\mathcal{X}\,;\,\Gamma, t = s \vdash \Delta} \qquad \frac{}{\mathcal{X}\,;\,\Gamma \vdash t \neq s, \Delta}$$

Otherwise, set $\theta = \mathrm{mgu}(t, s)$:

$$\frac{\theta\mathcal{X}\,;\,\theta\Gamma \vdash \theta\Delta}{\mathcal{X}\,;\,\Gamma, t = s \vdash \Delta} \qquad \frac{\theta\mathcal{X}\,;\,\theta\Gamma \vdash \theta\Delta}{\mathcal{X}\,;\,\Gamma \vdash t \neq s, \Delta}$$

Here, $\theta\mathcal{X}$ is the result of removing from $\mathcal{X}$ variables in the domain of $\theta$ and then adding the variables free in the codomain of $\theta$.

This treatment of equality was developed independently by Schroeder-Heister and Girard in [1991/92]. It has been extended to include simply typed $\lambda$-terms in Bedwyr and Abella.

Unification is a black box attached to sequent calculus.

# Return to subset example

Let $\Sigma$ and the sets $A$ and $B$ be

$$\lambda x.\ x = \mathbf{0} \vee x = \mathbf{1} \quad \text{and} \quad \lambda x.\ x = \mathbf{0} \vee x = \mathbf{1} \vee x = \mathbf{2}.$$

To prove that $A$ is a subset of $B$ requires proving the formula $\forall x. Ax \supset Bx$ is provable.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\cdot\,;\ \cdot \vdash \mathbf{0} = \mathbf{0}}}{\cdot\,;\ \cdot \vdash \mathbf{0} = \mathbf{0} \vee \mathbf{0} = \mathbf{1} \vee \mathbf{0} = \mathbf{2}}}{x;\ x = \mathbf{0} \vdash x = \mathbf{0} \vee x = \mathbf{1} \vee x = \mathbf{2}} \quad \dfrac{\dfrac{\dfrac{}{\cdot\,;\ \cdot \vdash \mathbf{1} = \mathbf{1}}}{\cdot\,;\ \cdot \vdash \mathbf{1} = \mathbf{0} \vee \mathbf{1} = \mathbf{1} \vee \mathbf{1} = \mathbf{2}}}{x;\ x = \mathbf{1} \vdash x = \mathbf{0} \vee x = \mathbf{1} \vee x = \mathbf{2}}}{x\,;\ x = \mathbf{0} \vee x = \mathbf{1} \vdash x = \mathbf{0} \vee x = \mathbf{1} \vee x = \mathbf{2}}}{\cdot\,;\ \cdot \vdash \forall x.(x = \mathbf{0} \vee x = \mathbf{1}) \supset (x = \mathbf{0} \vee x = \mathbf{1} \vee x = \mathbf{2})}$$

This proof accounts for reachability: we only consider checking membership in set $B$ for those elements "reached" in $A$.

# Fixed points

The least fixed point $\mu$ and greatest fixed point $\nu$ are actually a series of operators depending on the arity of the relationship they define. We leave this arity implicit. Unfolding $\mu B t_1 \dots t_n$ and $\nu B t_1 \dots t_n$ yields

$$B(\mu B) t_1 \dots t_n \quad \text{and} \quad B(\nu B) t_1 \dots t_n, \quad \text{respectively.}$$

Here, $\mu$ and $\nu$ have type $(i \to \cdots \to i \to o) \to i \to \cdots \to i \to o$

$$\frac{\mathcal{X}\,;\,\Gamma \vdash B(\mu B)\bar{t}, \Delta}{\mathcal{X}\,;\,\Gamma \vdash \mu B \bar{t}, \Delta}\ \mu R \qquad \frac{\mathcal{X}\,;\,\Gamma, S\bar{t} \vdash \Delta \quad \mathcal{X}, \bar{x}\,;\,BS\bar{x} \vdash S\bar{x}}{\mathcal{X}\,;\,\Gamma, \mu B \bar{t} \vdash \Delta}\ \mu L$$

$$\frac{\mathcal{X}\,;\,\Gamma, B(\nu B)\bar{t} \vdash \Delta}{\mathcal{X}\,;\,\Gamma, \nu B \bar{t} \vdash \Delta}\ \nu L \qquad \frac{\mathcal{X}\,;\,\Gamma \vdash S\bar{t}, \Delta \quad \bar{x}\,;\,S\bar{x} \vdash BS\bar{x}}{\mathcal{X}\,;\,\Gamma \vdash \nu B \bar{t}, \Delta}\ \nu R$$

Rules for $\mu$-unfolding on the left and $\nu$-unfolding on the right are admissible. Also, $\mu$ is positive and $\nu$ is negative.

# Horn clauses yield least fixed points

Horn clauses (Prolog) can be encoded as purely positive fixed point expressions. For example, for specifying a (tiny) graph and its transitive closure:

```
step a b.  step b c.  step c b.
path X Z :- step X Z.
path X Z :- step X Y, path Y Z.
```

Write the `step` as the least fixed point expression

$$\mu(\lambda A \lambda x \lambda y.\, (x = a \wedge^{+} y = b) \vee (x = b \wedge^{+} y = c) \vee (x = c \wedge^{+} y = b))$$

Likewise, `path` can be encoded as the relation $path(\cdot, \cdot)$:

$$\mu(\lambda A \lambda x \lambda z.\ \texttt{step}\ x\ z \vee (\exists y.\, \texttt{step}\ x\ y \wedge^{+} A\, y\, z)).$$

These expressions use only positive connectives and no non-logical predicates.

# Proofs with fixed points

There is no proof that there is a step from $a$ to $c$.

$$\dfrac{\dfrac{\mathit{fail}}{\vdash (a = a \wedge^{+} c = b) \vee (a = b \wedge^{+} c = c) \vee (a = c \wedge^{+} c = b)}}{\vdash \texttt{step}\ a\ c}$$

There is a proof that there is a path from $a$ to $c$.

$$\dfrac{\dfrac{\dfrac{\vdash \texttt{step}\ a\ b \qquad \vdash \mathit{path}\, b\, c}{\vdash \texttt{step}\ a\ b \wedge^{+} \mathit{path}\, b\, c}}{\vdash \exists y.\, \texttt{step}\ a\ y \wedge^{+} \mathit{path}\, y\, c}}{\dfrac{\vdash \texttt{step}\ a\ c \vee (\exists y.\, \texttt{step}\ a\ y \wedge^{+} \mathit{path}\, y\, c)}{\vdash \mathit{path}(a, c)}}$$

# Proof with fixed points

Below is a proof that the node $a$ is not adjacent to $c$.

$$\dfrac{\dfrac{\dfrac{}{a = a, c = b \vdash \cdot}}{a = a \wedge^{+} c = b \vdash \cdot} \qquad \dfrac{\dfrac{}{a = b, c = c \vdash \cdot}}{a = b \wedge^{+} c = c \vdash \cdot} \qquad \dfrac{\dfrac{}{a = c, c = b \vdash \cdot}}{a = c \wedge^{+} c = b \vdash \cdot}}{\dfrac{(a = a \wedge^{+} c = b) \vee (a = b \wedge^{+} c = c) \vee (a = c \wedge^{+} c = b) \vdash \cdot}{\texttt{step}\ a\ c \vdash \cdot}}$$

In general, proofs by negation-as-finite-failures yield sequent calculus proofs in this setting.

# More examples

Definitions of relations for natural numbers, addition, less-than.

$$
\begin{aligned}
nat =& \mu\lambda N\lambda n(n = z \vee \exists n'(n = s\ n' \wedge^{+} N\ n')) \\
plus =& \mu\lambda P\lambda n\lambda m\lambda p\ ((n = z \wedge^{+} m = p)\vee \\
& \qquad \exists n'\exists p'(n = s\ n' \wedge^{+} p = s\ p' \wedge^{+} P\ n'\ m\ p')) \\
lt =& \mu\lambda L\lambda x\lambda y((x = z \wedge^{+} \exists y'.y = sy')\vee \\
& \qquad (\exists x'\exists y'.x = sx' \wedge^{+} y = sy' \wedge^{+} L\ x'\ y'))
\end{aligned}
$$

while the following formula requires induction to be proved

$$\forall n\forall m\forall p.nat\ n \supset nat\ m \supset plus\ n\ m\ p \supset plus\ m\ n\ p,$$

the following formula can be proved by a model checker.

$$\forall n\forall m\forall p(lt\ n\ \mathbf{10} \supset lt\ m\ \mathbf{10} \supset plus\ n\ m\ p \supset plus\ m\ n\ p)$$

# Synthetic inference rules via focusing

Sequents in the focused proof system come in three styles.

- *up-arrow* sequents: $\Sigma : \mathcal{N} \Uparrow \Gamma \vdash \Delta \Uparrow \mathcal{P}$.
- *left-down-arrow* sequent: $\Sigma : \mathcal{N} \Downarrow B \vdash \mathcal{P}$.
- *right-down-arrow* sequent: $\Sigma : \mathcal{N} \vdash B \Downarrow \mathcal{P}$.

The zone marked by $\mathcal{N}$ is a multiset of negative formulas
The zone marked by $\mathcal{P}$ is a multiset of positive formulas

However, both $\Delta$ and $\Gamma$ are *lists* of formulas.

$\Sigma$ is a signature as we have seen before.

# A focused proof system: negative rules

$$\frac{\Sigma\theta : \mathcal{N}\theta \Uparrow \Gamma\theta \vdash \Delta\theta \Uparrow \mathcal{P}\theta}{\Sigma : \mathcal{N} \Uparrow s = t, \Gamma \vdash \Delta \Uparrow \mathcal{P}}\dagger \qquad \frac{\Sigma\theta : \mathcal{N}\theta \Uparrow \cdot \vdash \cdot \Uparrow \mathcal{P}\theta}{\Sigma : \mathcal{N} \Uparrow \cdot \vdash s \neq t \Uparrow \mathcal{P}}\dagger \qquad \frac{}{\Sigma : \mathcal{N} \Uparrow s = t, \Gamma \vdash \Delta \Uparrow \mathcal{P}}\ddagger$$

$$\frac{}{\mathcal{N} \Uparrow \cdot \vdash \cdot \Uparrow s \neq t, \Delta, \mathcal{P}}\ddagger \qquad \frac{\mathcal{N} \Uparrow \Gamma \vdash \Delta \Uparrow \mathcal{P}}{\mathcal{N} \Uparrow \mathit{tt}^+, \Gamma \vdash \Delta \Uparrow \mathcal{P}} \qquad \frac{\mathcal{N} \Uparrow \cdot \vdash \Delta \Uparrow \mathcal{P}}{\mathcal{N} \Uparrow \cdot \vdash \mathit{ff}^-, \Delta \Uparrow \mathcal{P}}$$

$$\frac{\mathcal{N} \Uparrow A_1, \Gamma \vdash \Delta \Uparrow \mathcal{P} \qquad \mathcal{N} \Uparrow A_2, \Gamma \vdash \Delta \Uparrow \mathcal{P}}{\mathcal{N} \Uparrow A_1 \vee A_2, \Gamma \vdash \Delta \Uparrow \mathcal{P}}$$

$$\frac{\mathcal{N} \Uparrow \cdot \vdash A_1 \Uparrow \mathcal{P} \qquad \mathcal{N} \Uparrow \cdot \vdash A_2 \Uparrow \mathcal{P}}{\mathcal{N} \Uparrow \cdot \vdash A_1 \wedge^- A_2 \Uparrow \mathcal{P}}$$

$$\frac{\mathcal{N} \Uparrow A_1, A_2, \Gamma \vdash \Delta \Uparrow \mathcal{P}}{\mathcal{N} \Uparrow A_1 \wedge^+ A_2, \Gamma \vdash \Delta \Uparrow \mathcal{P}} \qquad \frac{\mathcal{N} \Uparrow A_1 \vdash A_2, \Delta \Uparrow \mathcal{P}}{\mathcal{N} \Uparrow \cdot \vdash A_1 \supset A_2, \Delta \Uparrow \mathcal{P}} \qquad \frac{}{\mathcal{N} \Uparrow \mathit{ff}^+, \Gamma \vdash \Delta \Uparrow \mathcal{P}}$$

$$\frac{}{\mathcal{N} \Uparrow \cdot \vdash \mathit{tt}^-, \Delta \Uparrow \mathcal{P}} \qquad \frac{\Sigma,\ y : \mathcal{N} \Uparrow C\,y, \Gamma \vdash \Delta \Uparrow \mathcal{P}}{\Sigma : \mathcal{N} \Uparrow \exists x.\, C\,x, \Gamma \vdash \Delta \Uparrow \mathcal{P}} \qquad \frac{\Sigma,\ y : \mathcal{N} \Uparrow \cdot \vdash C\,y, \Delta \Uparrow \mathcal{P}}{\Sigma : \mathcal{N} \Uparrow \cdot \vdash \forall x.\, C\,x, \Delta \Uparrow \mathcal{P}}$$

$$\frac{\mathcal{N} \Uparrow B(\mu B)\bar{t}, \Gamma \vdash \Delta \Uparrow \mathcal{P}}{\mathcal{N} \Uparrow \mu B\,\bar{t}, \Gamma \vdash \Delta \Uparrow \mathcal{P}} \qquad \frac{\mathcal{N} \Uparrow \cdot \vdash B(\nu B)\bar{t}, \Delta \Uparrow \mathcal{P}}{\mathcal{N} \Uparrow \cdot \vdash \nu B\,\bar{t}, \Delta \Uparrow \mathcal{P}}$$

Proviso $\dagger$: $\theta = mgu(s, t)$ and $\ddagger$: $s$ and $t$ not unifiable.

# A focused proof system: positive and structural rules

$$\frac{}{\mathcal{N} \Downarrow t \neq t \vdash \mathcal{P}} \qquad \frac{}{\mathcal{N} \vdash t = t \Downarrow \mathcal{P}} \qquad \frac{}{\mathcal{N} \Downarrow \mathit{ff}^- \vdash \mathcal{P}} \qquad \frac{}{\mathcal{N} \vdash \mathit{tt}^+ \Downarrow \mathcal{P}}$$

$$\frac{\mathcal{N}_1 \cdot \vdash A_1 \Downarrow \mathcal{P}_1 \quad \mathcal{N}_2 \Downarrow A_2 \vdash \mathcal{P}_2}{\mathcal{N}_1, \mathcal{N}_2 \Downarrow A_1 \supset A_2 \vdash \mathcal{P}_1, \mathcal{P}_2} \qquad \frac{\mathcal{N}_1 \vdash A_1 \Downarrow \mathcal{P}_1 \quad \mathcal{N}_2 \vdash A_2 \Downarrow \mathcal{P}_2}{\mathcal{N}_1, \mathcal{N}_2 \vdash A_1 \wedge^+ A_2 \Downarrow \mathcal{P}_1, \mathcal{P}_2}$$

$$\frac{\mathcal{N} \Downarrow A_i \vdash \mathcal{P}}{\mathcal{N} \Downarrow A_1 \wedge^- A_2 \vdash \mathcal{P}} \qquad \frac{\mathcal{N} \vdash A_i \Downarrow \mathcal{P}}{\mathcal{N} \vdash A_1 \vee A_2 \Downarrow \mathcal{P}}$$

$$\frac{\mathcal{N} \Downarrow C\, t \vdash \mathcal{P}}{\mathcal{N} \Downarrow \forall x.\, C\, x \vdash \mathcal{P}} \qquad \frac{\mathcal{N} \vdash C\, t \Downarrow \mathcal{P}}{\mathcal{N} \vdash \exists x.\, C\, x \Downarrow \mathcal{P}}$$

$$\frac{\mathcal{N} \Downarrow B(\nu B)\bar{t} \vdash \mathcal{P}}{\mathcal{N} \Downarrow \nu B\, \bar{t} \vdash \mathcal{P}} \qquad \frac{\mathcal{N} \vdash B(\mu B)\bar{t} \Downarrow \mathcal{P}}{\mathcal{N} \vdash \mu B\, \bar{t} \Downarrow \mathcal{P}}$$

store

$$\frac{\mathcal{N}, N \Uparrow \Gamma \vdash \Delta \Uparrow \mathcal{P}}{\mathcal{N} \Uparrow N, \Gamma \vdash \Delta \Uparrow \mathcal{P}} \qquad \frac{\mathcal{N} \Uparrow \cdot \vdash \Delta \Uparrow P, \mathcal{P}}{\mathcal{N} \Uparrow \cdot \vdash P, \Delta \Uparrow \mathcal{P}}$$

release

$$\frac{\mathcal{N} \Uparrow P \vdash \cdot \Uparrow \mathcal{P}}{\mathcal{N} \Downarrow P \vdash \mathcal{P}} \qquad \frac{\mathcal{N} \Uparrow \cdot \vdash N \Uparrow \mathcal{P}}{\mathcal{N} \vdash N \Downarrow \mathcal{P}}$$

decide

$$\frac{\mathcal{N} \Downarrow N \vdash \mathcal{P}}{\mathcal{N}, N \Uparrow \cdot \vdash \cdot \Uparrow \mathcal{P}} \qquad \frac{\mathcal{N} \vdash P \Downarrow \mathcal{P}}{\mathcal{N} \Uparrow \cdot \vdash \cdot \Uparrow P, \mathcal{P}}$$

Here, unfolding replaces induction and coinduction.

# Synthetic inference rules

Sequents of the form $\Sigma\colon \mathcal{N} \Uparrow \cdot \vdash \cdot \Uparrow \mathcal{P}$ are *border* sequents.

*Synthetic inference rules* have border sequents as conclusion and as premises.

A border sequent $\Sigma\colon \mathcal{N} \Uparrow \cdot \vdash \cdot \Uparrow \mathcal{P}$ where $\mathcal{P} \cup \mathcal{N}$ is a singleton multiset is called a *singleton* border sequent.

Such a sequent is of the form

$$\Sigma\colon N \Uparrow \cdot \vdash \cdot \Uparrow \cdot \quad \text{or} \quad \Sigma\colon \cdot \Uparrow \cdot \vdash \cdot \Uparrow P$$

These sequent represent proving $\neg N$ (for a negative formula $N$) or proving $P$ for a positive formula $P$.

Only the decide rules can have such a sequent as its conclusion and there is only one choice for the focus.

# Synthetic inference rules: purely positive formulas

$$P := tt^{+} \mid t = s \mid \mu\lambda A\lambda\bar{x}.P \mid P \wedge^{+} P \mid P \vee P \mid \exists x.P$$

Consider a border sequent with a purely positive $P$ on the right.

$$\frac{\begin{array}{c}\Xi \\ \Sigma : \cdot \vdash P \Downarrow \cdot\end{array}}{\Sigma : \cdot \Uparrow \cdot \vdash \cdot \Uparrow P}$$

If a complete proof $\Xi$ exists, it is entirely one (positive) phase. An entire (non-deterministic) computation is placed into one synthetic inference rule.

For example, Prolog-like computations can be forced into one phase. Obviously, checking such synthetic inference rules is undecidable in general.

# Additive synthetic connectives

In order to build on *additive synthetic connectives*, we need to restrict occurrence of the multiplicative connectives $\supset$ and $\wedge^+$.

A $\mu$MALL$^=$ formula is *switchable* if

- whenever a subformula $C \wedge^+ D$ occurs negatively (under an odd number of implications), either $C$ or $D$ is purely positive;
- whenever a subformula $C \supset D$ occurs positively (under an even number of implications), either $C$ is purely positive or $D$ is purely negative.

Note: purely positive formulas and purely negative formulas are switchable.

An occurrence of a formula $B$ in a sequent is *switchable* if it appears on the right-hand side (resp. left-hand side) and $B$ (resp. $B \supset ff^-$) is switchable.

# Example: simulation

Let $P \xrightarrow{A} Q$ be a labeled transition system between processes and actions and that it is defined as a purely positive expression.

If $p, q \in P$ and $a \in A$ then both $P \xrightarrow{A} Q$ and $(P \xrightarrow{A} Q) \supset ff^-$ are switchable formulas.

The following two greatest fixed point expressions define simulation and bisimulation for this label transition systems.

$$\nu\big(\lambda S \lambda p \lambda q.\, \forall a \forall p'.\, p \xrightarrow{a} p' \supset \exists q'.\, q \xrightarrow{a} q' \wedge^+ S\, p'\, q'\big)$$

$$\begin{aligned}\nu(\lambda B \lambda p \lambda q. \quad & (\forall a \forall p'.\, p \xrightarrow{a} p' \supset \exists q'.\, q \xrightarrow{a} q' \wedge^+ B\, p'\, q') \\ & \wedge^- (\forall a \forall q'.\, q \xrightarrow{a} q' \supset \exists p'.\, p \xrightarrow{a} p' \wedge^+ B\, q'\, p'))\end{aligned}$$

These are switchable formulas. Note that bisimulation has both conjunctions.

# Switchable formulas yield additive synthetic rules

The following theorem is proved by a simple induction on the structure of $\mu\mathsf{MALL}^{=}$ proofs.

### Theorem (switchability)

*Let $\Pi$ be a $\mu MALLF^{=}$ derivation of either $\Sigma\colon A \Uparrow \cdot \vdash \cdot \Uparrow \cdot$ or $\Sigma\colon \cdot \Uparrow \cdot \vdash \cdot \Uparrow A$ where the occurrence of $A$ is switchable. Also assume that every invariant $S$ is purely positive. Then every sequent in $\Pi$ that is the conclusion of a rule that switches phases (either a decide or a release rule) contains exactly one occurrence of a formula and that occurrence is switchable.*

# An example of a synthetic inference rules

$$
\begin{array}{c}
\dfrac{\cdot : \cdot \Uparrow \cdot \vdash sim(p_i, q_i) \Uparrow \cdot}{\cdot : \cdot \vdash sim(p_i, q_i) \Downarrow \cdot} \\
\hline\hline
\cdot : \cdot \vdash \exists Q'.q_0 \xrightarrow{a_i} Q' \wedge^+ sim(p_i, Q') \Downarrow \cdot \quad C \\
\hline
\cdot : \cdot \Uparrow \cdot \vdash \cdot \Uparrow \exists Q'.q_0 \xrightarrow{a_i} Q' \wedge^+ sim(p_i, Q') \\
\hline
\cdots \quad \cdot : \cdot \Uparrow \cdot \vdash \exists Q'.q_0 \xrightarrow{a_i} Q' \wedge^+ sim(p_i, Q') \Uparrow \cdot \quad \cdots \\
\hline\hline
P', A : p_0 \xrightarrow{A} P' \Uparrow \exists Q'.q_0 \xrightarrow{A} Q' \wedge^+ sim(P', Q') \vdash \cdot \Uparrow \quad B \\
\hline\hline
\cdot : \cdot \Uparrow \cdot \vdash sim(p_0, q_0) \Uparrow \cdot \quad A
\end{array}
$$

$A$ contain introduction rules for $\forall$ and $\supset$.

$B$ consists of $\Uparrow$ rules that generates all $a_i$ and $p_i$ such that $p_0 \xrightarrow{a_i} p_i$.

$C$ is a sequence of $\Downarrow$ rules that prove that $q_0 \xrightarrow{a_i} q_i$.

Finally, the top-most inference rule is a release rule.

# Some applications

The model checker *Bedwyr*, built before the designing of $\mu$MALL, is now understood as proof search in $\mu$MALL.
The interactive theorem prover *Abella* is based on a similar logic but emphasizes induction and coinduction.
With Rob Blanco and Quentin Health, we are looking at means to develop proof certificates for model checking. In that way, Abella could check a Bedwyr proof and vice versus.

Producing certificates for model checking seems rather difficult in general. We have shown the following.

- That a path in a graph can be proof certificate for *reachability*.
- Connected components can be a proof certificate for *non-reachability*.
- A bisimulation can be a proof certificate for bisimilarity.
- A Hennessy-Milner modal formula can be a proof certificate for *non-bisimilarity*.

# Thank you