Curry-Howard Correspondance between Temporal Logic and Reactive Programming

Esaïe Bauer

Alexis Saurin & Thomas Ehrhard

13 octobre 2021
**Who am I?**

**Today**
PhD student under the direction of Alexis Saurin & Thomas Ehrhard.
### Who am I?

#### Today

PhD student under the direction of Alexis Saurin & Thomas Ehrhard.

#### Internships

Worked with Marie Kerjean & Olivier Laurent.
Table des matières

1. Context

2. Line of work
Table des matières

1 Context

2 Line of work
Temporal logic is a logic system where formulas have a truth value evolving over time.
Informal definitions

Temporal logic is a logic system where formulas have a truth value evolving over time.

Reactive programming is a programm paradigm concerned by propagating a reactive input (such as stream) to ensure properties or modify value over time.

Example: spreadsheet, graphical interface, web app
Temporal logic

Three connectives

We introduce three connectives:

\( \circ A, \Diamond A \) and \( \square A \)
### Temporal logic

**Three connectives**

We introduce three connectives:

\[ \square A, \quad \Diamond A \quad \text{and} \quad \Diamond A \]

**Formulas**

\[ F ::= \text{Var} \mid F \lor F \mid F \land F \mid F \rightarrow F \mid \Diamond F \mid \mu X.F \mid \nu X.F \mid \bot \mid 1 \]
Fixed point

Constructions with fixed point

\[ \Diamond A ::= \mu X. A \vee \Diamond X \lor \Box A ::= \nu X. A \wedge \Box X \]

or \[ A \cup B ::= \mu X. ((A \wedge \Box X) \vee B) \]

Fixed point over \( \Box \) and \( \Diamond \)

\[ \Box \Diamond A \quad \text{and} \quad \nu X. \mu Y. ((A \wedge \Box X) \vee \Box Y) \]

both express that there will be an infinite number of \( A \)
Reactive programming and causality

Causality

\[ f(s_1, \ldots, s_n, s_{n+1}, \ldots) = f(s_1, \ldots, s_n, s'_{n+1}, \ldots) \text{ at time } n \]
Reactive programming and causality

Causality

\[ f(s_1, \ldots, s_n, s_{n+1}, \ldots) = f(s_1, \ldots, s_n, s'_{n+1}, \ldots) \text{ at time } n \]

\[ \bigcirc(A \lor B) \rightarrow \bigcirc A \lor \bigcirc B \]

should not be provable.
FRP (Cave, Ferreira, Panangaden & Pientka)

\[\text{let } \bullet x = t_1 \text{ in } t_2 \quad \bullet t\]

Sequents

[\Theta; \Gamma \vdash A]

Typing rules

\[
\begin{align*}
\Theta; \Gamma \vdash t : A & \quad \Rightarrow \quad \Theta; \Gamma \vdash \bullet t : \circ A \\
\Theta; \Gamma \vdash t_1 : \circ A & \quad \Rightarrow \quad \Theta, x : A; \Gamma \vdash t_2 : B \quad \circ_e
\end{align*}
\]
Reduction rules and normalization

Règle de réduction

\[ \text{let } \bullet x = \bullet t_1 \text{ in } t_2 \rightarrow t_2[x := t_1] \bullet \]
Reduction rules and normalization

Règle de réduction

$\text{let } \bullet x = \bullet t_1 \text{ in } t_2 \rightarrow t_2[x := t_1] \bullet$

Strong Normalization

If $\Theta; \Gamma \vdash t : A$ is provable then $t$ is strongly normalizing
Table des matières

1. Context

2. Line of work
Transforming FRP

FRP typing system seen as a sequent calculus, with step-indexed formula led to:

$$; \odot A \rightarrow \odot B \not\vdash \odot(A \rightarrow B)$$
FRP typing system seen as a sequent calculus, with step-indexed formula led to:

; ∪A → ∪B ⊬ ∪(A → B)

Transforming FRP into a $\bar{\lambda}_\mu$-FRP
Transforming FRP

FRP typing system seen as a sequent calculus, with step-indexed formula led to:

; $\circ A \rightarrow \circ B \not\vdash \circ(A \rightarrow B)$

Transforming FRP into a $\bar{\lambda}_\mu$-FRP

Problem: in classical LTL, $\circ(A \lor B) \vdash \circ A \lor \circ B$ is a theorem.
Circular proofs

Circular proofs is a way of formalizing induction and coinduction in rule system.
Circular proofs

Circular proofs is a way of formalizing induction and coinduction in rule system.

Formalizing such a system in a Coq Library