universität innsbruck

supported by FWF project I5943-N

Automated Analysis of Logically Constrained Rewrite Systems

Jonas Schöpf

PhD Rigorosum
24th October 2025

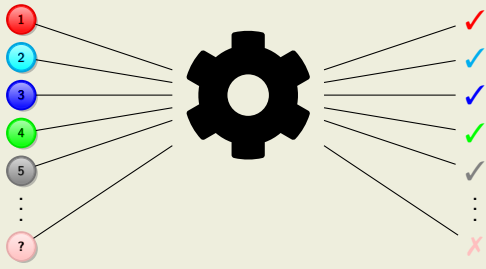
Table of Contents

Table of Contents

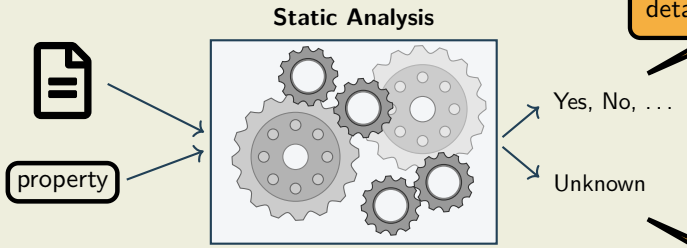
How to Avoid Bugs?

- program carefully? skilled programmers?
- bugs may not be detected by testing

Testing?



Static Program Analysis



Properties

- **termination** — does the program finish in a finite amount of time?
- **confluence** — does the program compute unique solutions?
- **complexity** — how long does the program run?
- ...

Computational Model

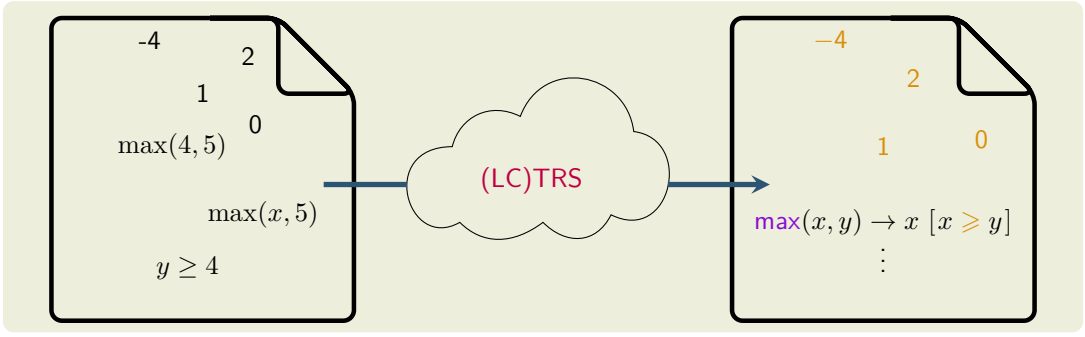


Table of Contents

Term Rewrite Systems (TRSs)

set of variables	$\{x, y, z, \dots\}$
set of function symbols	$\{\mathsf{max}, 0\}$
terms	$\mathsf{max}(x, y), \mathsf{max}(\mathsf{max}(x, 0), z), \dots$
rules	$\mathsf{max}(\mathsf{s}(0), 0) \to \mathsf{s}(0)$
set of rules	$\{\mathsf{max}(\mathsf{s}(0), 0) \to \mathsf{s}(0), \dots\}$

- terms represent program states
- rewriting represents computation
- term rewriting is Turing-complete

Term Rewrite System

signature $\{\mathsf{max}, \mathsf{ite}, \mathsf{s}, \mathsf{p}, \mathsf{geq}, \mathsf{geq2}, \mathsf{0}, \mathsf{true}, \mathsf{false}\}$ and rules

$$\mathsf{max}(x,y) \to \mathsf{ite}(\mathsf{geq}(x,y),x,y) \qquad \mathsf{ite}(\mathsf{true},x,y) \to x$$
$$\mathsf{max}(x,y) \to \mathsf{max}(y,x) \qquad \mathsf{ite}(\mathsf{false},x,y) \to y$$
$$\mathsf{s}(\mathsf{p}(x)) \to x \qquad \mathsf{p}(\mathsf{s}(x)) \to x$$
$$\mathsf{geq}(x,y) \to \mathsf{geq2}(x,y,\mathsf{0},\mathsf{0}) \qquad \mathsf{geq2}(\mathsf{s}(x),y,z,u) \to \mathsf{geq2}(x,y,\mathsf{s}(z),u)$$
$$\mathsf{geq2}(\mathsf{p}(x),y,z,u) \to \mathsf{geq2}(x,y,z,\mathsf{s}(u)) \qquad \mathsf{geq2}(\mathsf{0},\mathsf{s}(x),y,z) \to \mathsf{geq2}(\mathsf{0},x,y,\mathsf{s}(z))$$
$$\mathsf{geq2}(\mathsf{0},\mathsf{p}(x),y,z) \to \mathsf{geq2}(\mathsf{0},x,\mathsf{s}(y),z) \qquad \mathsf{geq2}(\mathsf{0},\mathsf{0},\mathsf{s}(x),\mathsf{s}(y)) \to \mathsf{geq2}(\mathsf{0},\mathsf{0},x,y)$$
$$\mathsf{geq2}(\mathsf{0},\mathsf{0},x,\mathsf{0}) \to \mathsf{true} \qquad \mathsf{geq2}(\mathsf{0},\mathsf{0},\mathsf{0},\mathsf{s}(x)) \to \mathsf{false}$$

compute $\mathrm{max}(4,5)$: $\mathsf{max}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{0})))),\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{0}))))))$

$$\begin{aligned}\mathsf{max}(\mathsf{s}^4(\mathsf{0}),\mathsf{s}^5(\mathsf{0})) &\to \mathsf{ite}(\mathsf{geq}(\mathsf{s}^4(\mathsf{0}),\mathsf{s}^5(\mathsf{0})),\mathsf{s}^4(\mathsf{0}),\mathsf{s}^5(\mathsf{0}))\\ &\to \mathsf{ite}(\mathsf{geq2}(\mathsf{s}^4(\mathsf{0}),\mathsf{s}^5(\mathsf{0})),\mathsf{s}^4(\mathsf{0}),\mathsf{s}^5(\mathsf{0}),\mathsf{0},\mathsf{0})\\ &\to^{12} \mathsf{s}^5(\mathsf{0}) = \mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{0}))))))\end{aligned}$$

Logically Constrained Rewrite System (LCTRS)

LCTRS $\mathcal{M}$

$$\mathcal{I}_{\mathsf{Bool}} = \mathbb{B} \qquad \mathcal{I}_{\mathsf{Int}} = \mathbb{Z}$$

$$\mathcal{F}_{\mathsf{te}} = \ldots, -1, 0, 1, \ldots : \mathsf{Int} \qquad \mathsf{max} : [\mathsf{Int}] \Rightarrow \mathsf{Int}$$

$$\mathcal{F}_{\mathsf{th}} = \ldots, -1, 0, 1, \ldots : \mathsf{Int} \qquad \wedge : [\mathsf{Bool} \times \mathsf{Bool}] \Rightarrow \mathsf{Bool}$$

$$\mathsf{true}, \mathsf{false} : \mathsf{Bool} \qquad +, - : [\mathsf{Int} \times \mathsf{Int}] \Rightarrow \mathsf{Int}$$

$$\neg : [\mathsf{Bool}] \Rightarrow \mathsf{Bool} \qquad \leqslant, \geqslant, = : [\mathsf{Int} \times \mathsf{Int}] \Rightarrow \mathsf{Bool}$$

$$\mathcal{M} = \quad \mathsf{max}(x, y) \to x \ [x \geqslant y] \qquad \mathsf{max}(x, y) \to y \ [y \geqslant x] \qquad \mathsf{max}(x, y) \to \mathsf{max}(y, x)$$

compute $\max(4, 5)$: $\mathsf{max}(4, 5) \to 5$

$$\mathsf{max}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(0)))), \mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(0))))))) \to^{14} \mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(\mathsf{s}(0)))))$$

Confluence

- undecidable in general
- critical pairs
- sufficient criteria

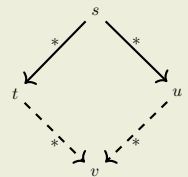


Table of Contents

Automation of Rewriting Infrastructure (ARI)

Confluence Criteria

- sufficient criteria
- (parallel) critical pairs
- based on TRS methods

Automation

- ~~Ctrl~~ $\Rightarrow$ crest
- decidable conditions
- participate in CoCo

LCTRS Semantics

- constrained equations
- inference system
- algebraic semantics

LCTRS category CoCo

- LCTRS CoCo format
- supported sub-categories
- benchmarks for database

Automation of Rewriting Infrastructure (ARI)

Confluence Criteria

- sufficient criteria
- (parallel) critical pairs
- based on TRS methods

Automation

- ~~Ctrl~~ $\Rightarrow$ crest
- decidable conditions
- participate in CoCo

LCTRS Semantics

- constrained equations
- inference system
- algebraic semantics

LCTRS category CoCo

- LCTRS CoCo format
- supported sub-categories
- benchmarks for database

CADE 2023 – Confluence Criteria

- adapt critical pair definition
- strong closedness
- parallel rewriting
- replaying proofs

- input format for LCTRSs
- transformations
- prototype
- experiments

Theorems

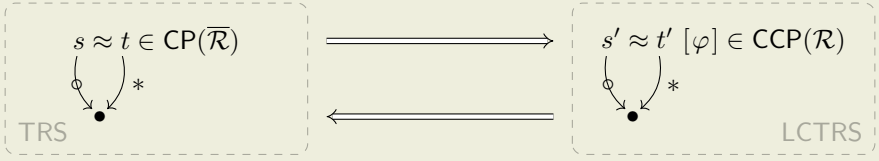
- linear strongly closed LCTRSs are confluent — CADE 2023
- left-linear almost parallel closed LCTRSs are confluent — CADE 2023

IJCAR 2024 – Confluence Criteria Revisited

- transformation LCTRS to TRS
- critical pair correspondence
- repair parallel rewriting

- parallel critical pairs
- multi-step rewriting
- **simpler proofs!**

Main Idea



Theorems

- local confluence is undecidable for terminating LCTRSs — IJCAR 2024
- left-linear almost development closed LCTRSs are confluent — IJCAR 2024
- left-linear LCTRSs with parallel closed parallel critical pairs are confluent — IJCAR 2024

Example

rewrite rules

$$\mathsf{max}(x, y) \to x \ [x \geqslant y] \qquad \mathsf{max}(x, y) \to y \ [y \geqslant x] \qquad \mathsf{max}(x, y) \to \mathsf{max}(y, x)$$

constrained critical pairs

$$x \approx y \ [y \geqslant x \wedge x \geqslant y] \qquad x \approx \mathsf{max}(y, x) \ [x \geqslant y] \qquad y \approx \mathsf{max}(y, x) \ [y \geqslant x]$$

confluence criterion

$$\ldots \qquad x \approx \mathsf{max}(y, x) \ [x \geqslant y] \to x \approx x \ [x \geqslant y] \qquad \ldots$$

$\Longrightarrow$ confluence

Automation of Rewriting Infrastructure (ARI)

Confluence Criteria

- sufficient criteria
- (parallel) critical pairs
- based on TRS methods

Automation

- ~~Ctrl~~ $\Rightarrow$ crest
- decidable conditions
- participate in CoCo

LCTRS Semantics

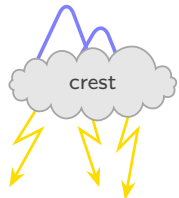
- constrained equations
- inference system
- algebraic semantics

LCTRS category CoCo

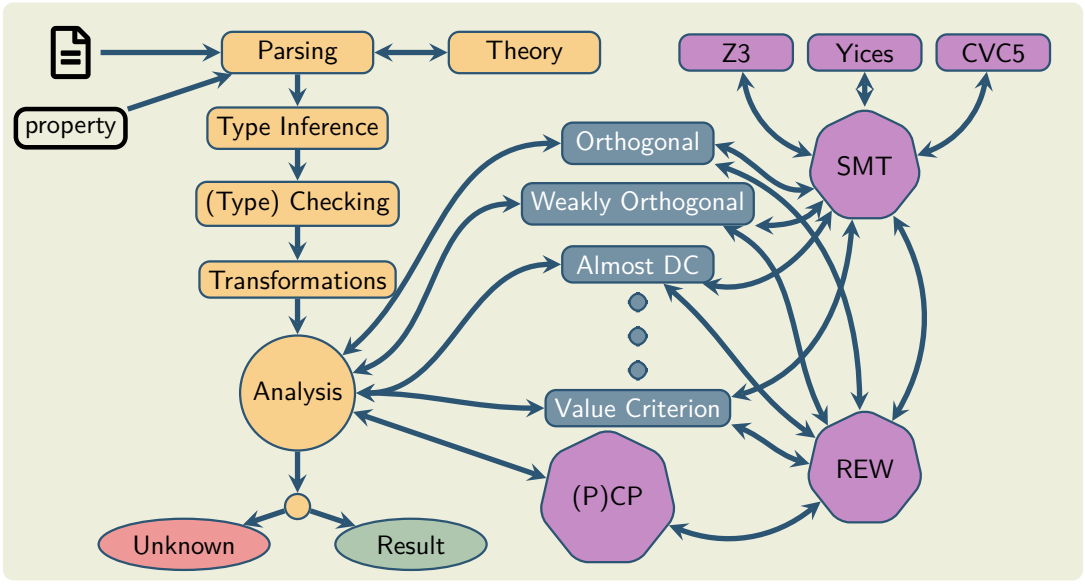
- LCTRS CoCo format
- supported sub-categories
- benchmarks for database

TACAS 2025 – Automation in crest

- Constrained REwriting Software Tool (crest)
- (non-)confluence and termination analysis
- Haskell
- open source
- non-confluence criterion
- transformations



Simplified Overview of crest



Automation of Rewriting Infrastructure (ARI)

Confluence Criteria

- sufficient criteria
- (parallel) critical pairs
- based on TRS methods

Automation

- ~~Ctrl~~ $\Rightarrow$ crest
- decidable conditions
- participate in CoCo

LCTRS Semantics

- constrained equations
- inference system
- algebraic semantics

LCTRS category CoCo

- LCTRS CoCo format
- supported sub-categories
- benchmarks for database

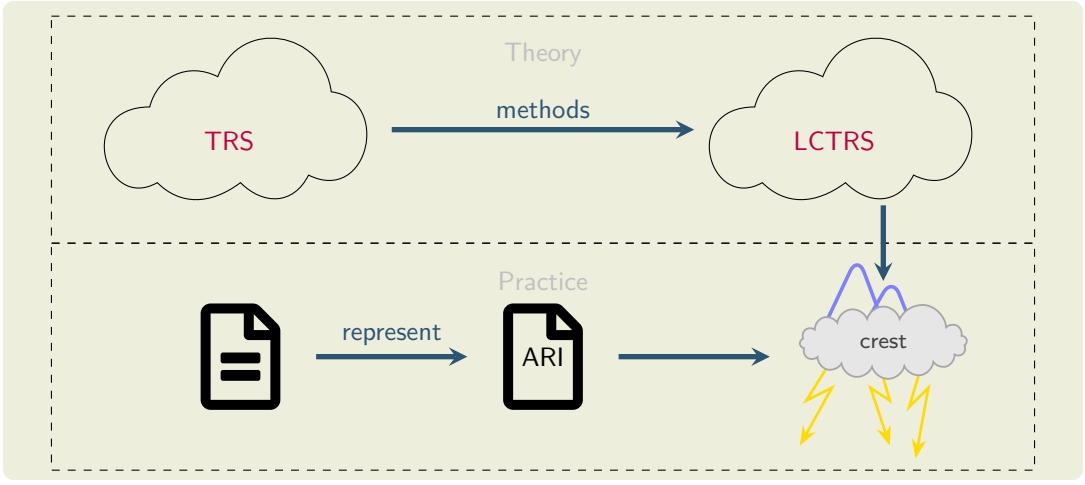
ARI Format & Database

- ARI format
- examples in literature (by hand)
- 113 benchmarks
- **more needed!**

Confluence Competition

- since 2012
- LCTRS category since 2024
- first place in 2024 and 2025

Simplified Summary



Automation of Rewriting Infrastructure (ARI)

Confluence Criteria

- sufficient criteria
- (parallel) critical pairs
- based on TRS methods

Automation

- ~~Ctrl~~ $\Rightarrow$ crest
- decidable conditions
- participate in CoCo

LCTRS Semantics

- constrained equations
- inference system
- algebraic semantics

LCTRS category CoCo

- LCTRS CoCo format
- supported sub-categories
- benchmarks for database

FSCD 2024 – Semantics of LCTRSs

- introduce Π notion
- constrained equations (CEs)
- constrained equational theory

- CE-validity
- inference system
- algebraic semantics

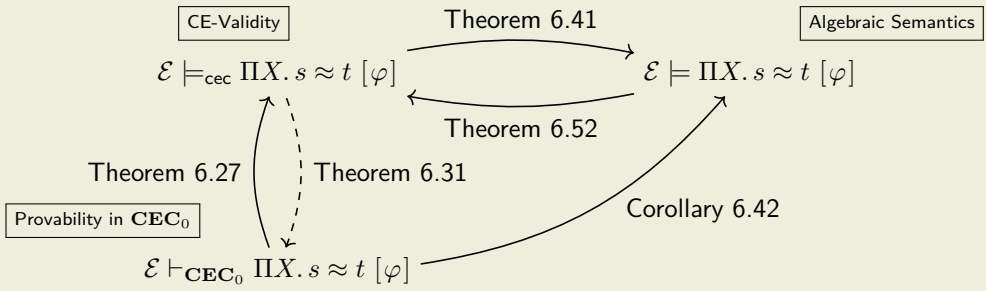
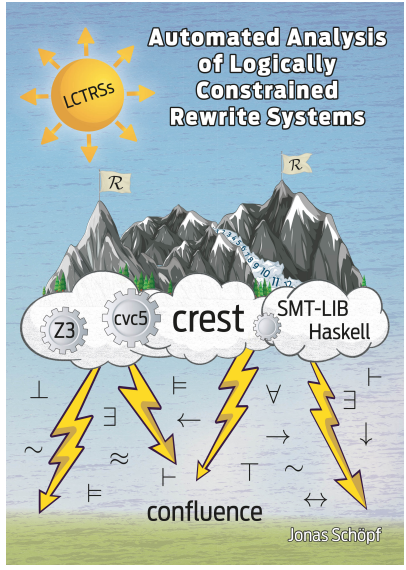


Table of Contents

Done?

- labeling techniques, critical pair closing systems, more compositional criteria, ...
- improve non-confluence analysis
- constrained simultaneous critical pairs
- implementation of the reduction method
- (non-)termination methods
- narrowing constrained terms
- formally verifying the obtained results
- connection to the real-world
- ...

PhD Thesis



Thank you for your attention!