universität
innsbruck

Improved Certification of Complexity Proofs for Term Rewrite Systems

René Thiemann

IFIP WG 1.6, Dortmund, June 26

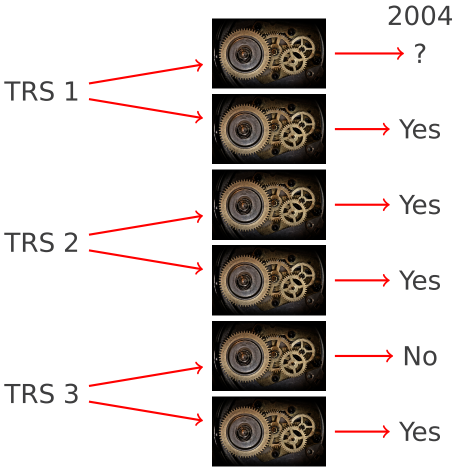
Supported by the Austrian Science Fund (FWF) project Y757

Overview

- IsaFoR + CeTA:
  Certifying Termination and Complexity Proofs
- Certifying Matrix Growth
- Formalization of the Perron–Frobenius Theorem

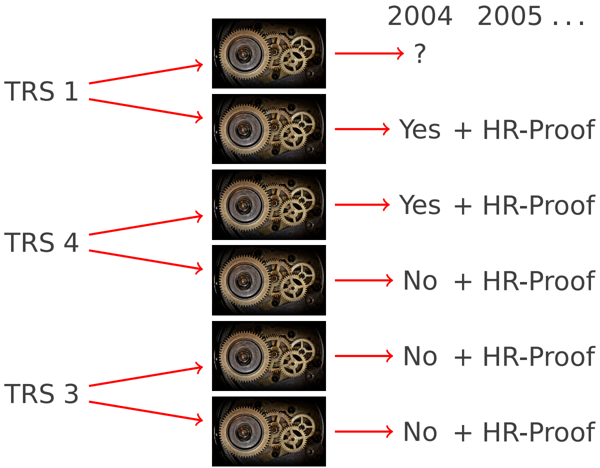
Annual International Termination Competition

-  automatic termination and complexity tools
  – powerful, complex, unreliable



Annual International Termination Competition

-  automatic termination and complexity tools – powerful, complex, unreliable



Annual International Termination Competition

-  automatic termination and complexity tools
  – powerful, complex, unreliable

2004 2005 . . . 2007 . . .

TRS 1

 → ?

 → Yes + HR-Proof + MR-Proof

TRS 4

 → Yes + HR-Proof + MR-Proof

 → Yes + HR-Proof + MR-Proof

TRS 5

 → No + HR-Proof + MR-Proof

 → Yes + HR-Proof + MR-Proof

Certification of Termination Proofs

-  automatic termination and complexity tools
  – powerful, complex, unreliable
-  certifiers
  – reliable, soundness proof in proof assistants
  – revealed errors in tools and papers
- certified termination and complexity analysis

TRS →  —answer / certificate→  → okay / reject, unsupported

Certification of Termination Proofs

-  automatic termination and complexity tools
  – powerful, complex, unreliable
-  certifiers
  – reliable, soundness proof in proof assistants
  – revealed errors in tools and papers
- certified termination and complexity analysis

TRS →  —answer / certificate→  → okay / reject
→ unsupported

- CeTA: certifier for termination, complexity, confluence, ..
- soundness of CeTA: Isabelle Formalization of Rewriting developed in collaboration with Christian Sternagel and . . .

Certification of Termination Proofs

-  automatic termination and complexity tools
  – powerful, complex, unreliable
-  certifiers
  – reliable, soundness proof in proof assistants
  – revealed errors in tools and papers
- certified termination and complexity analysis

TRS →  answer → certificate  → okay / reject → unsupported

- CeTA: certifier for termination, complexity, confluence, ..
- soundness of CeTA: Isabelle Formalization of Rewriting developed in collaboration with Christian Sternagel and ...
- this talk

  improvements of IsaFoR/CeTA for complexity proofs

Complexity of Term Rewrite Systems

$$
\begin{aligned}
\mathsf{sort}(\mathsf{Cons}(x, xs)) &\to \mathsf{insort}(x, \mathsf{sort}(xs)) \\
\mathsf{sort}(\mathsf{Nil}) &\to \mathsf{Nil} \\
\mathsf{insort}(x, \mathsf{Cons}(y, ys)) &\to \mathsf{Cons}(x, \mathsf{Cons}(y, ys)) && \mid x \leqslant y \\
\mathsf{insort}(x, \mathsf{Cons}(y, ys)) &\to \mathsf{Cons}(y, \mathsf{insort}(x, ys)) && \mid x \nleqslant y \\
\mathsf{insort}(x, \mathsf{Nil}) &\to \mathsf{Cons}(x, \mathsf{Nil})
\end{aligned}
$$

Aim: bound on maximal number of rewrite steps starting from

$$\mathsf{sort}(\mathsf{Cons}(x_1, \ldots \mathsf{Cons}(x_n, \mathsf{Nil})))$$

Running Automated Complexity tool

Running TCT on TRS yields $\mathcal{O}(n^2)$ + certificate

$$\llbracket \mathsf{sort} \rrbracket(xs) = \begin{pmatrix} 3 & 3 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} \cdot \llbracket xs \rrbracket$$

$$\llbracket \mathsf{insort} \rrbracket(x, xs) = \begin{pmatrix} 1 & 1 & 2 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} \cdot \llbracket xs \rrbracket + \begin{pmatrix} 2 \\ 1 \\ 2 \end{pmatrix}$$

$$\llbracket \mathsf{Cons} \rrbracket(x, xs) = \underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}}_{A} \cdot \llbracket xs \rrbracket + \begin{pmatrix} 0 \\ 1 \\ 2 \end{pmatrix}$$

$$\llbracket \mathsf{Nil} \rrbracket = \begin{pmatrix} 1 \\ 0 \\ 2 \end{pmatrix}$$

Certification — Step 1

- ensure termination:
  check strict decrease in every rewrite step
- for rewrite rule $\mathsf{sort}(\mathsf{Cons}(x, xs)) \to \mathsf{insort}(x, \mathsf{sort}(xs))$
  check

$$[\![\mathsf{sort}(\mathsf{Cons}(x, xs))]\!] =$$

$$\begin{pmatrix} 3 & 3 & 3 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} \cdot [\![xs]\!] + \begin{pmatrix} 3 \\ 2 \\ 2 \end{pmatrix} \begin{matrix} > \\ \geq \\ \geq \end{matrix} \begin{pmatrix} 3 & 3 & 3 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} \cdot [\![xs]\!] + \begin{pmatrix} 2 \\ 1 \\ 2 \end{pmatrix}$$

$$= [\![\mathsf{insort}(x, \mathsf{sort}(xs))]\!]$$

Certification — Step 2

- bound initial interpretation

$$\llbracket \mathsf{sort}(\mathsf{Cons}(x_1, \ldots \mathsf{Cons}(x_n, \mathsf{Nil}))) \rrbracket =$$
$$\begin{pmatrix} 3 & 3 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} \left(A^n \begin{pmatrix} 1 \\ 0 \\ 2 \end{pmatrix} + \sum_{i<n} A^i \begin{pmatrix} 0 \\ 1 \\ 2 \end{pmatrix} \right) \in \mathcal{O}(n \cdot A^n)$$

$\Longrightarrow$ key analysis: growth of values of A^n depending on n

Matrix Growth

- input: non-negative real matrix

$$A = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

- task: decide matrix growth

  how large do values in A^n get for increasing n?

Eigenvalues and eigenvectors

Matrix A has eigenvector $v \neq 0$ with eigenvalue λ if

$$Av = \lambda v$$

Consequences

- $A^n v = \lambda^n v$
- $|A^n v| = |\lambda|^n |v|$
- if $|\lambda| > 1$ then A^n grows exponentially

Eigenvalues and eigenvectors

Matrix A has eigenvector $v \neq 0$ with eigenvalue λ if

$$Av = \lambda v$$

Consequences

- $A^n v = \lambda^n v$
- $|A^n v| = |\lambda|^n |v|$
- if $|\lambda| > 1$ then A^n grows exponentially

Theorem

A^n grows polynomially if and only if

$|\lambda| \leqslant 1$

for all eigenvalues λ of A

Eigenvalues and eigenvectors

Matrix A has eigenvector $v \neq 0$ with eigenvalue λ if

$$Av = \lambda v$$

Consequences

- $A^n v = \lambda^n v$
- $|A^n v| = |\lambda|^n |v|$
- if $|\lambda| > 1$ then A^n grows exponentially

Theorem

A^n grows polynomially if and only if
$|\lambda| \leqslant 1$
for all eigenvalues λ of A

Remark

- λ is eigenvalue of A if and only if
  λ is root of characteristic polynomial χ_A

Eigenvalues and eigenvectors

Matrix A has eigenvector $v \neq 0$ with eigenvalue λ if

$$Av = \lambda v$$

Consequences

- $A^n v = \lambda^n v$
- $|A^n v| = |\lambda|^n |v|$
- if $|\lambda| > 1$ then A^n grows exponentially

Theorem

$A^n \in \mathcal{O}(n^d)$ if and only if
$|\lambda| \leqslant 1$ and $|\lambda| = 1 \longrightarrow$ max-size (Jordan Blocks λ) $\leqslant d + 1$
for all eigenvalues λ of A

Remark

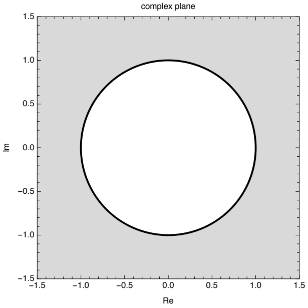
- λ is eigenvalue of A if and only if
  λ is root of characteristic polynomial χ_A

Old certification algorithm for $A^n \in \mathcal{O}(n^d)$

Input: Matrix A and degree d

Output: Accept or assertion failure

1. Compute all eigenvalues $\lambda_1, \ldots, \lambda_n$ of A (all complex roots of χ_A)
2. Compute spectral radius $\rho_A := \max_i |\lambda_i|$
3. Assert $\rho_A \leqslant 1$
4. For each λ_i with $|\lambda_i| = 1$, and Jordan block of A and λ_i with size s_i, assert $s_i \leqslant d + 1$
5. Accept



Example of linear growth

Input: Matrix A and degree d

Output: Accept or assertion failure

1. Compute all eigenvalues $\lambda_1, \ldots, \lambda_n$ of A (all complex roots of χ_A)
2. Compute spectral radius $\rho_A := \max_i |\lambda_i|$
3. Assert $\rho_A \leqslant 1$
4. For each λ_i with $|\lambda_i| = 1$, and Jordan block of A and λ_i with size s_i, assert $s_i \leqslant d + 1$
5. Accept

$$\text{Input: } A = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}, d = 1$$

1. $\lambda_1 = 1, \lambda_2 = 0$

2. $\rho_A = 1$

4. $s_1 = 2 \leqslant d + 1$

Another example

Input: $A = \frac{1}{2}\begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$

1. $\chi_A = \dfrac{(x-1)\left(8x^3-4x^2-2x-1\right)}{8}$

   $\lambda_1 = 1$

   $\lambda_2 = \left(\text{root \#1 of } f_1\right)$

   $\lambda_3 = \left(\text{root \#1 of } f_2\right) + \left(\text{root \#1 of } f_3\right)\mathrm{i}$

   $\lambda_4 = \left(\text{root \#1 of } f_2\right) + \left(\text{root \#2 of } f_3\right)\mathrm{i}$

   $f_1 = 8x^3 - 4x^2 - 2x - 1$

   $f_2 = 32x^3 - 16x^2 + 1$

   $f_3 = 1024x^6 + 512x^4 + 64x^2 - 11$

The problem and its solution

- old algorithm requires precise calculations ($|\lambda_i| = 1$)
- precise calculations are possible with algebraic numbers, but expensive
- aim: avoid explicit computation of eigenvalues
- solution: apply the Perron–Frobenius theorem

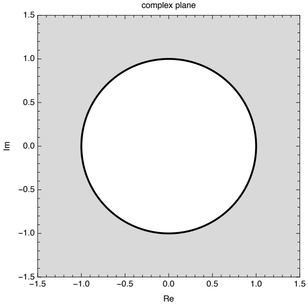
Perron–Frobenius, Part 1

Theorem (Perron–Frobenius)

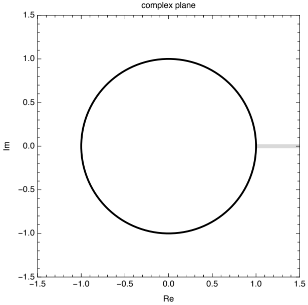
Let A be a non-negative real matrix

- *ρ_A is an eigenvalue of A*

Consequence



$\longrightarrow$



Perron–Frobenius, Part 2

Theorem (Perron–Frobenius)

Let A be a non-negative real and irreducible matrix

- ρ_A *is an eigenvalue of* A
- ρ_A *has multiplicity 1*
- ρ_A *is only eigenvalue with non-negative real eigenvector*
- $\exists f\, k.\ \chi_A = f \cdot (x^k - \rho_A^k) \wedge (f(y) = 0 \longrightarrow |y| < \rho_A)$
- $\ldots$

Perron–Frobenius, Part 2

Theorem (Perron–Frobenius)

Let A be a non-negative real and irreducible matrix

- ρ_A *is an eigenvalue of* A
- ρ_A *has multiplicity 1*
- ρ_A *is only eigenvalue with non-negative real eigenvector*
- $\exists f\, k.\ \chi_A = f \cdot (x^k - \rho_A^k) \wedge (f(y) = 0 \longrightarrow |y| < \rho_A)$
- $\ldots$

Consequences

- non-negative real and irreducible matrices have constant or exponential growth
- complexity proofs with irreducible matrices cannot prove runtime/derivational complexity $\mathcal{O}(n^d)$ for $d > 1$

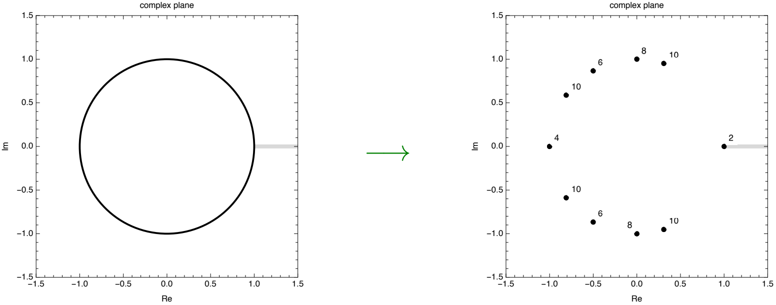
Perron–Frobenius, Part 3

Theorem

Let A be a non-negative real matrix

- ρ_A *is an eigenvalue of* A
- $\exists f\, K.\ \chi_A = f \cdot \prod_{k \in K}(x^k - \rho_A^k) \wedge (f(y) = 0 \longrightarrow |y| < \rho_A)$

Consequence



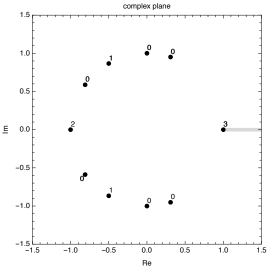
Uniqueness of f and K

Theorem

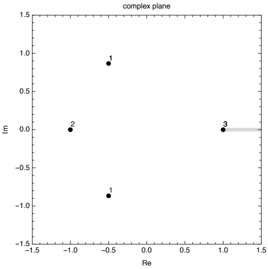
Let A be a non-negative real matrix

- ρ_A *is an eigenvalue of* A
- $\exists! f\, K.\ \chi_A = f \cdot \prod_{k \in K}(x^k - \rho_A^k) \wedge (f(y) = 0 \longrightarrow |y| < \rho_A)$
- decompose χ_A *computes* f *and* K *for* $\rho_A = 1$

Consequence



$\longrightarrow\ K = \{2, 2, 3\} +$



New certification algorithm for $A^n \in \mathcal{O}(n^d)$

$$\exists! f\, K.\ \chi_A = f \cdot \prod_{k \in K} (x^k - \rho_A^k) \wedge (f(y) = 0 \longrightarrow |y| < \rho_A)$$

Input: non-negative real matrix A and degree d

Output: Accept or assertion failure.

1. Assert that χ_A has no real roots in $(1, \infty)$ via Sturm's method
2. Compute K via decompose χ_A
3. For each $k \in \{1, \ldots, \max K\}$ do
   - $m_k := |\{k' \in K.\ k \text{ divides } k'\}|$
   - If $m_k > d + 1$ then check Jordan blocks for all primitive roots of unity of degree k, i.e., assert Jordan block size $\leqslant d + 1$
4. Accept

Experiments

large examples ($dim\ A = 21$)

- old: timeouts after 1 hour
- new: finished in fraction of second

matrices of termination competitions 2015–2018
($2 \leqslant dim\ A \leqslant 5$)

- new algorithm 5x faster

Unpublished new certification algorithm for $A^n \in \mathcal{O}(n^d)$

New Theorem

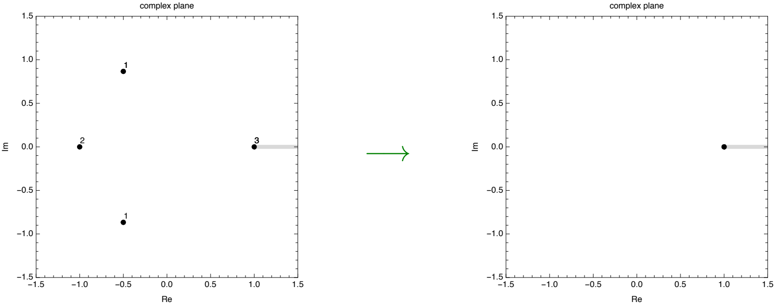
If A is non-negative real matrix and $\rho_A \leqslant 1$ then for every JB with $|\lambda| = 1$ there exists JB of 1 which is at least as large

Unpublished new certification algorithm for $A^n \in \mathcal{O}(n^d)$

New Theorem

If A is non-negative real matrix and $\rho_A \leqslant 1$ then for every JB with $|\lambda| = 1$ there exists JB of 1 which is at least as large

Consequence



Unpublished new certification algorithm for $A^n \in \mathcal{O}(n^d)$

New Theorem

If A is non-negative real matrix and $\rho_A \leqslant 1$ then for every JB with $|\lambda| = 1$ there exists JB of 1 which is at least as large

Input: non-negative real matrix A and degree d
Output: Accept or assertion failure

1. Assert that χ_A has no real roots in $(1, \infty)$ via Sturm's method
2. Assert that each Jordan block of eigenvalue 1 has size $s \leqslant d + 1$
3. Accept

certifying matrix growth for complexity proofs
without algebraic numbers

Improvements in Automation

- new certification algorithm runs in polynomial time

$\Longrightarrow$ there exists polynomial time SAT/SMT-encoding

$\Longrightarrow$ possibility to encode desired degree when searching for matrix interpretation

- currently investigated by TCT-team

Part of Paper Proof

Definitions

$$
\begin{aligned}
X &:= \{x \in \mathbb{R}^n \mid x \geq 0, x \neq 0\} \\
X_1 &:= \{x \in X \mid \|x\| = 1\} \\
Y &:= \{(A+I)^n x \mid x \in X_1\} \\
r(x) &:= \min_{j, x_j \neq 0} \frac{(Ax)_j}{x_j} \\
r_{max} &:= \max \{r(y) \mid y \in Y\}
\end{aligned}
$$

Lemmas

- X_1 and Y are compact
- r is continuous on Y
- r_{max} is well-defined (extreme value theorem)
- $r_{max} = \rho_A$
- $\chi'_A(\rho_A) = \sum_i \chi_{B_i}(\rho_A) > 0$ where $B_i =$ mat-delete A i i

Overview on Formalization

- HMA: Type-based vectors and matrices (ι :: finite $\rightarrow \alpha$)
- JNF: Carrier-based vectors and matrices ($\mathbb{N} \times (\mathbb{N} \rightarrow \alpha)$)

	HMA library	JNF library
compatible dimensions	type-system	explicit carrier
arithmetic, determinants, ...	✓	✓
continuity, compactness, ...	✓	
block-matrices, delete row, ...		✓

- formalization of Perron–Frobenius requires all features

$\Longrightarrow$ develop connection between both worlds: HMA connect

Overview of Formalization

Perron–Frobenius formalization

libraries HMA and JNF

Part 1
ρ_A is eigenvalue

Brouwer's fixpoint theorem

Part 2
more thms for irred. A

extreme value theorem

HMA connect

derivative of χ_A

HMA connect

Part 3
more thms for arbitr. A

block decomposition

HMA Connect

- main aim: establish connection between JNF and HMA
- tool: transfer
  - define correspondence-relation between vectors, matrices, . . .

$$HMA_{vec} :: \mathbb{N} \times (\mathbb{N} \to \alpha) \to (\iota \to \alpha) \to \mathsf{bool}$$
$$HMA_{vec}\ v\ w = (v = (\mathsf{CARD}(\iota), \lambda i. w_{\text{from-nat } i}))$$

    where from-nat is some bijection between ι and $\{0, \ldots, \mathsf{CARD}(\iota) - 1\} \subseteq \mathbb{N}$
  - prove transfer rules between constants of JNF and HMA

$$(HMA_{mat} \longrightarrow HMA_{mat} \longrightarrow HMA_{mat})\ \mathsf{op}\ +\ \mathsf{op}\ +$$
$$(HMA_{mat} \longrightarrow \mathsf{op}\ =)\ \mathsf{det}\ \mathsf{det}$$

  - finally transfer complex statements between JNF and HMA

Transferring Theorems from JNF to HMA

- JNF lemma for derivative of characteristic polynomial

$$A \in \text{carrier-mat}\ n\ n \longrightarrow$$
$$\text{pderiv}\ (\text{charpoly}\ A) = \sum_{i<n} \text{charpoly}\ (\text{mat-delete}\ A\ i\ i)$$

- transfer to HMA not yet possible: mat-delete not available
- solution: reformulate lemma

$$A \in \text{carrier-mat}\ n\ n \longrightarrow \text{monom}\ 1\ 1 *$$
$$\text{pderiv}\ (\text{charpoly}\ A) = \sum_{i<n} \text{charpoly}\ (\text{mat-erase}\ A\ i\ i)$$

- transfer to HMA

$$\text{monom}\ 1\ 1\ * \text{pderiv}\ (\text{charpoly}\ A) =$$
$$\sum_{i} \text{charpoly}\ (\text{mat-erase}\ A\ i\ i)$$

Transferring Theorems from HMA to JNF

- Perron–Frobenius Theorem Part 1 (HMA)

  $\text{real-non-neg-mat}\ A \longrightarrow \text{eigenvalue}\ A\ (\text{spectral-radius}\ A)$

- transfer to JNF

  $A \in \text{carrier-mat}\ (\text{CARD}(\iota))\ (\text{CARD}(\iota)) \longrightarrow$
  $\text{real-non-neg-mat}\ A \longrightarrow \text{eigenvalue}\ A\ (\text{spectral-radius}\ A)$

- post-processing with local type definition

  $A \in \text{carrier-mat}\ n\ n \longrightarrow n \neq 0 \longrightarrow$
  $\text{real-non-neg-mat}\ A \longrightarrow \text{eigenvalue}\ A\ (\text{spectral-radius}\ A)$

Summary

- formalization of Perron–Frobenius theorem:
  combination of two libraries via transfer + local types

- new theorem: Jordan blocks of spectral radius are largest

- improving IsaFoR/CeTA:
  certifying complexity proofs without algebraic numbers

joint work with Jose Divasón, Sebastiaan Joosten,
Ondřej Kunčar, and Akihisa Yamada

Future work / work in progress

Check termination proofs of programming languages

- formalize semantics of subset of LLVM IR in Isabelle (ongoing)
- verify translation to integer transition systems (future work)
- verify backend for integer transition systems
  - SMT-solver for LRA (basic solver available, ongoing)
  - bounds on integer solutions: LIA is in NP (unpublished)
  - theory-solver for LIA (ongoing)
  - SMT-solver for LIA (future work)