Ranking Templates for Linear Loops

Jan Leike
The Australian National University

Matthias Heizmann
University of Freiburg

Termination

- safety reduced to reachability - liveness reduced to termination

Termination

- safety reduced to reachability - liveness reduced to termination
- neither provable nor refutable by testing

Termination

- safety reduced to reachability - liveness reduced to termination
- neither provable nor refutable by testing
- computing fixpoint on sets of states does not work

Termination

- safety reduced to reachability - liveness reduced to termination
- neither provable nor refutable by testing
- computing fixpoint on sets of states does not work
- ranking function (decreasing, bounded, contradiction!)

Research directions

1. practical tools for termination analysis

Urban, Miné **An Abstract Domain to Infer Ordinal-Valued Ranking Functions** (ESOP 2014)

Brockschmidt, Cook, Fuhs **Better Termination Proving through Cooperation** (CAV 2013)

Kroening, Sharygina, Tsitovich, Wintersteiger **Termination analysis with compositional transition invariants** (CAV 2010)

Cook, B., Podelski, A., Rybalchenko, A. **Terminator: Beyond safety** (CAV 2006)

...

2. decidability of terminaton for restricted classes of programs

Ben-Amram, Genaim **Ranking functions for linear-constraint loops** (POPL 2013)

Ben-Amram, Genaim, Masud **On the Termination of Integer Loops** (VMCAI 2012)

Tiwari **Termination of Linear Programs** (CAV 2004)

...

3. constraint-based synthesis of ranking functions for loops

Cook, Kroening, Rümmer, Wintersteiger **Ranking function synthesis for bit-vector relations** (FMSD 2013)

Rybalchenko **Constraint solving for program verification theory and practice by example** (CAV 2010)

Colón, Sankaranarayanan, Sipma **Linear invariant generation using non-linear constraint solving** (CAV 2003)

...

Ranking functions for loops - applications

- termination analysis for programs
  - Terminator (Cook, Rybalchenko, et al.)
  - T2 (Brockschmidt, et al.)
  - Tan (Chen, Kroening, Wintersteiger, et al.)
  - Ultimate Büchi Automizer (H. et al.)

Ranking functions for loops - applications

- termination analysis for programs
  - Terminator (Cook, Rybalchenko, et al.)
  - T2 (Brockschmidt, et al.)
  - Tan (Chen, Kroening, Wintersteiger, et al.)
  - Ultimate Büchi Automizer (H. et al.)
- cost analysis
- stability of hybrid systems

- affine-linear ranking functions

Colón, Sipma **Synthesis of Linear Ranking Functions** (TACAS 2001)

Podelski, Rybalchenko **A complete method for the synthesis of linear ranking functions** (VMCAI 2004)

Bradley, Manna, Sipma **Termination Analysis of Integer Linear Loops** (CONCUR 2005)

Cook, Kroening, Rümmer, Wintersteiger **Ranking function synthesis for bit-vector relations** (FMSD 2013)

Ben-Amram, Genaim **Ranking functions for linear-constraint loops** (POPL 2013)

- affine-linear ranking functions

Colón, Sipma **Synthesis of Linear Ranking Functions** (TACAS 2001)

Podelski, Rybalchenko **A complete method for the synthesis of linear ranking functions** (VMCAI 2004)

Bradley, Manna, Sipma **Termination Analysis of Integer Linear Loops** (CONCUR 2005)

Cook, Kroening, Rümmer, Wintersteiger **Ranking function synthesis for bit-vector relations** (FMSD 2013)

Ben-Amram, Genaim **Ranking functions for linear-constraint loops** (POPL 2013)

- lexicographic linear ranking functions

Bradley, Manna, Sipma **Linear ranking with reachability** (CAV 2005)

Alias, Darte, Feautrier, Gonnord **Multi-dimensional Rankings, Program Termination, and Complexity Bounds of Flowchart Programs** (SAS 2010)

Cook, See, Zuleger **Ramsey vs. Lexicographic Termination Proving** (TACAS 2013)

- affine-linear ranking functions

Colón, Sipma **Synthesis of Linear Ranking Functions** (TACAS 2001)

Podelski, Rybalchenko **A complete method for the synthesis of linear ranking functions** (VMCAI 2004)

Bradley, Manna, Sipma **Termination Analysis of Integer Linear Loops** (CONCUR 2005)

Cook, Kroening, Rümmer, Wintersteiger **Ranking function synthesis for bit-vector relations** (FMSD 2013)

Ben-Amram, Genaim **Ranking functions for linear-constraint loops** (POPL 2013)

- lexicographic linear ranking functions

Bradley, Manna, Sipma **Linear ranking with reachability** (CAV 2005)

Alias, Darte, Feautrier, Gonnord **Multi-dimensional Rankings, Program Termination, and Complexity Bounds of Flowchart Programs** (SAS 2010)

Cook, See, Zuleger **Ramsey vs. Lexicographic Termination Proving** (TACAS 2013)

- piecewise linear ranking functions

Urban, Miné **An Abstract Domain to Infer Ordinal-Valued Ranking Functions** (ESOP 2014)

- affine-linear ranking functions

Colón, Sipma **Synthesis of Linear Ranking Functions** (TACAS 2001)

Podelski, Rybalchenko **A complete method for the synthesis of linear ranking functions** (VMCAI 2004)

Bradley, Manna, Sipma **Termination Analysis of Integer Linear Loops** (CONCUR 2005)

Cook, Kroening, Rümmer, Wintersteiger **Ranking function synthesis for bit-vector relations** (FMSD 2013)

Ben-Amram, Genaim **Ranking functions for linear-constraint loops** (POPL 2013)

- lexicographic linear ranking functions

Bradley, Manna, Sipma **Linear ranking with reachability** (CAV 2005)

Alias, Darte, Feautrier, Gonnord **Multi-dimensional Rankings, Program Termination, and Complexity Bounds of Flowchart Programs** (SAS 2010)

Cook, See, Zuleger **Ramsey vs. Lexicographic Termination Proving** (TACAS 2013)

- piecewise linear ranking functions

Urban, Miné **An Abstract Domain to Infer Ordinal-Valued Ranking Functions** (ESOP 2014)

- multiphase ranking functions

Bradley, Manna, Sipma **The polyranking principle** (ICALP 2005)

one method to synthesize them all

Ranking function

$Loop(\mathbf{x}, \mathbf{x'})$

Ranking function

$$\forall x x'.\ Loop(x, x') \rightarrow \underbrace{f(x) > f(x')}_{\text{decreasing}} \wedge \underbrace{f(x) > 0}_{\text{bounded}}$$

Synthesis of ranking function

$$\forall \mathbf{x}\mathbf{x}'.\ \mathit{Loop}(\mathbf{x}, \mathbf{x}') \rightarrow \underbrace{\mathbf{f}(\mathbf{x}) > \mathbf{f}(\mathbf{x}')}_{\text{decreasing}} \wedge \underbrace{\mathbf{f}(\mathbf{x}) > 0}_{\text{bounded}}$$

Idea:

- write definition as logical formula,
- let theorem prover find satisfying assignment for **free variables**

Synthesis of ranking function

$$\forall \mathbf{x}\mathbf{x'}.\ Loop(\mathbf{x}, \mathbf{x'}) \rightarrow \underbrace{\mathbf{f}(\mathbf{x}) > \mathbf{f}(\mathbf{x'})}_{\text{decreasing}} \wedge \underbrace{\mathbf{f}(\mathbf{x}) > 0}_{\text{bounded}}$$

Idea:

- write definition as logical formula,
- let theorem prover find satisfying assignment for **free variables**

Problem:

- no theorem prover for domain of functions

Solution:

- use template $\mathrm{T}(\mathbf{x}, \mathbf{x'})$

Synthesis of ranking function

$\forall \mathbf{x}\mathbf{x'}.\ \mathit{Loop}(\mathbf{x}, \mathbf{x'}) \rightarrow\ \textsc{t}(\mathbf{x}, \mathbf{x'})$

Idea:

- write definition as logical formula,
- let theorem prover find satisfying assignment for **free variables**

Problem:

- no theorem prover for domain of functions

Solution:

- use template $\textsc{t}(\mathbf{x}, \mathbf{x'})$

Synthesis of affine-linear ranking function

$\forall \mathbf{x}\mathbf{x'}.\ \mathit{Loop}(\mathbf{x}, \mathbf{x'}) \rightarrow \mathrm{T}(\mathbf{x}, \mathbf{x'})$

where the template $\mathrm{T}(\mathbf{x}, \mathbf{x'})$ is $f(\mathbf{x}) > f(\mathbf{x'}) \wedge f(\mathbf{x}) > 0$

decreasing

bounded

and $f(\mathbf{x})$ is a shorthand for the affine-linear term $\mathbf{c}_1 \cdot \mathbf{x}_1 + \cdots + \mathbf{c}_n \cdot \mathbf{x}_1 + \mathbf{c}_0$

Synthesis of affine-linear ranking function

$\forall \mathbf{x}\mathbf{x}'.\ \mathit{Loop}(\mathbf{x}, \mathbf{x}') \rightarrow \mathrm{T}(\mathbf{x}, \mathbf{x}')$

where the template $\mathrm{T}(\mathbf{x}, \mathbf{x}')$ is $\underbrace{f(\mathbf{x}) > f(\mathbf{x}')}_{\text{decreasing}} \wedge \underbrace{f(\mathbf{x}) > 0}_{\text{bounded}}$

and $f(\mathbf{x})$ is a shorthand for the affine-linear term $\mathbf{c}_1 \cdot \mathbf{x}_1 + \cdots + \mathbf{c}_n \cdot \mathbf{x}_1 + \mathbf{c}_0$

Difficult!

- universal quantification ($\forall \mathbf{x}$ ...)
- nonlinear arithmetic ($\mathbf{c}_1 \cdot \mathbf{x}_1$ ...)

Synthesis of affine-linear ranking function

$\forall \mathbf{x}\mathbf{x}'.\ \mathit{Loop}(\mathbf{x}, \mathbf{x}') \rightarrow \mathrm{T}(\mathbf{x}, \mathbf{x}')$

where the template $\mathrm{T}(\mathbf{x}, \mathbf{x}')$ is $\underbrace{f(\mathbf{x}) > f(\mathbf{x}')}_{\text{decreasing}} \wedge \underbrace{f(\mathbf{x}) > 0}_{\text{bounded}}$

and $f(\mathbf{x})$ is a shorthand for the affine-linear term $\mathbf{c}_1 \cdot \mathbf{x}_1 + \cdots + \mathbf{c}_n \cdot \mathbf{x}_1 + \mathbf{c}_0$

Difficult!

- universal quantification ($\forall \mathbf{x}$ ...)
- nonlinear arithmetic ($\mathbf{c}_1 \cdot \mathbf{x}_1$ ...)

Lemma (Farkas)

$$\forall x.\ (\ldots \rightarrow \ldots) \quad \textit{iff} \quad \exists \vec{\lambda}\ (\ldots)$$

Synthesis of affine-linear ranking function

$\forall \mathbf{x}\mathbf{x}'.\ Loop(\mathbf{x}, \mathbf{x}') \rightarrow \mathrm{T}(\mathbf{x}, \mathbf{x}')$

where the template $\mathrm{T}(\mathbf{x}, \mathbf{x}')$ is $\underbrace{f(\mathbf{x}) > f(\mathbf{x}')}_{\text{decreasing}} \wedge \underbrace{f(\mathbf{x}) > 0}_{\text{bounded}}$

and $f(\mathbf{x})$ is a shorthand for the affine-linear term $\mathbf{c}_1 \cdot \mathbf{x}_1 + \cdots + \mathbf{c}_n \cdot \mathbf{x}_1 + \mathbf{c}_0$

Difficult!

- universal quantification ($\forall \mathbf{x}$ ...)
- nonlinear arithmetic ($\mathbf{c}_1 \cdot \mathbf{x}_1$ ...)

Lemma (Farkas)

$$\forall x.\ (... \rightarrow ...) \qquad \textit{iff} \qquad \exists \vec{\lambda}\ (...)$$

Colón, Sipma **Synthesis of Linear Ranking Functions** (TACAS 2001)

Lexicographic ranking function

Lexicographic ranking function

$$\text{program state} \mapsto \text{lexicographic ordered tuple}$$

Lexicographic order: e.g. $(2, 3, 4) \geq (2, 2, 9)$

Lexicographic ranking function

Lexicographic ranking function

$$\text{program state} \mapsto \text{lexicographic ordered tuple}$$

Lexicographic order: e.g. $(2, 3, 4) \geq (2, 2, 9)$

Linear lexicographic ranking function

each entry of the tuple defined by linear function

$$\mathbf{f}(x) = (\ \mathbf{f}_1(x),\ \ldots,\ \mathbf{f}_k(x)\ )$$

Lexicographic ranking function

Lexicographic ranking function

program state $\mapsto$ lexicographic ordered tuple

Lexicographic order: e.g. $(2,3,4) \geq (2,2,9)$

Linear lexicographic ranking function
each entry of the tuple defined by linear function

$$\mathbf{f}(x) = (\ \mathbf{f}_1(x),\ \ldots,\ \mathbf{f}_k(x)\ )$$

Recall:

Idea:

- write definition as logical formula,
- let theorem prover find satisfying assignment for **free variables**

Linear lexicographic ranking functions

$$\mathrm{T}(\mathbf{x}, \mathbf{x'}) := \overbrace{\mathbf{f}_1(\mathbf{x}) \geq 0}^{\mathbf{f}_1 \text{ bounded}} \wedge \overbrace{\mathbf{f}_2(\mathbf{x'}) \geq 0}^{\mathbf{f}_2 \text{ bounded}} \wedge$$

$$\Big(\underbrace{\mathbf{f}_1(\mathbf{x}) > \mathbf{f}_1(\mathbf{x'})}_{\mathbf{f}_1 \text{ decreasing}} \vee \underbrace{\mathbf{f}_1(\mathbf{x}) \geq \mathbf{f}_1(\mathbf{x'})}_{\mathbf{f}_1 \text{ not increasing}} \wedge \underbrace{\mathbf{f}_2(\mathbf{x}) > \mathbf{f}_2(\mathbf{x'})}_{\mathbf{f}_2 \text{ decreasing}} \Big)$$

Linear lexicographic ranking functions

$$\mathrm{T}(\mathbf{x}, \mathbf{x'}) := \overset{\textbf{f}_1 \text{ bounded}}{\mathbf{f}_1(\mathbf{x}) \geq 0} \wedge \overset{\textbf{f}_2 \text{ bounded}}{\mathbf{f}_2(\mathbf{x'}) \geq 0} \wedge$$

$$\Big(\underset{\textbf{f}_1 \text{ decreasing}}{\mathbf{f}_1(\mathbf{x}) > \mathbf{f}_1(\mathbf{x'})} \vee \underset{\textbf{f}_1 \text{ not increasing}}{\mathbf{f}_1(\mathbf{x}) \geq \mathbf{f}_1(\mathbf{x'})} \wedge \underset{\textbf{f}_2 \text{ decreasing}}{\mathbf{f}_2(\mathbf{x}) > \mathbf{f}_2(\mathbf{x'})} \Big)$$

each $\mathbf{f}(\mathbf{x})$ is a shorthand for an affine-linear term $\mathbf{c}_1 \cdot \mathbf{x}_1 + \cdots + \mathbf{c}_n \cdot \mathbf{x}_1 + \mathbf{c}_0$

Linear lexicographic ranking functions

$$\mathrm{T}(\mathbf{x}, \mathbf{x'}) := \overbrace{\mathbf{f}_1(\mathbf{x}) \geq 0}^{\mathbf{f}_1\text{ bounded}} \wedge \overbrace{\mathbf{f}_2(\mathbf{x'}) \geq 0}^{\mathbf{f}_2\text{ bounded}} \wedge \Big(\underbrace{\mathbf{f}_1(\mathbf{x}) > \mathbf{f}_1(\mathbf{x'})}_{\mathbf{f}_1\text{ decreasing}} \vee \underbrace{\mathbf{f}_1(\mathbf{x}) \geq \mathbf{f}_1(\mathbf{x'})}_{\mathbf{f}_1\text{ not increasing}} \wedge \underbrace{\mathbf{f}_2(\mathbf{x}) > \mathbf{f}_2(\mathbf{x'})}_{\mathbf{f}_2\text{ decreasing}} \Big)$$

$$\forall \mathbf{x}\mathbf{x'}.\ \mathit{Loop}(\mathbf{x}, \mathbf{x'}) \rightarrow \mathrm{T}(\mathbf{x}, \mathbf{x'})$$

Linear lexicographic ranking functions

$$\mathrm{T}(\mathbf{x}, \mathbf{x'}) := \overbrace{\mathbf{f}_1(\mathbf{x}) \geq 0}^{\mathbf{f}_1 \text{ bounded}} \wedge \overbrace{\mathbf{f}_2(\mathbf{x'}) \geq 0}^{\mathbf{f}_2 \text{ bounded}} \wedge \left(\underbrace{\mathbf{f}_1(\mathbf{x}) > \mathbf{f}_1(\mathbf{x'})}_{\mathbf{f}_1 \text{ decreasing}} \vee \underbrace{\mathbf{f}_1(\mathbf{x}) \geq \mathbf{f}_1(\mathbf{x'})}_{\mathbf{f}_1 \text{ not increasing}} \wedge \underbrace{\mathbf{f}_2(\mathbf{x}) > \mathbf{f}_2(\mathbf{x'})}_{\mathbf{f}_2 \text{ decreasing}} \right)$$

$\forall \mathbf{x}\mathbf{x'}.\ \mathit{Loop}(\mathbf{x}, \mathbf{x'}) \rightarrow \mathrm{T}(\mathbf{x}, \mathbf{x'})$

Lemma (Farkas)

$$\forall x.\ (\ldots \rightarrow \ldots) \qquad \textit{iff} \qquad \exists \vec{\lambda}\ (\ldots)$$

Linear lexicographic ranking functions

$$\mathrm{T}(\mathbf{x}, \mathbf{x'}) := \underbrace{\mathbf{f}_1(\mathbf{x}) \geq 0}_{\mathbf{f}_1 \text{ bounded}} \wedge \underbrace{\mathbf{f}_2(\mathbf{x'}) \geq 0}_{\mathbf{f}_2 \text{ bounded}} \wedge \Big(\underbrace{\mathbf{f}_1(\mathbf{x}) > \mathbf{f}_1(\mathbf{x'})}_{\mathbf{f}_1 \text{ decreasing}} \vee \underbrace{\mathbf{f}_1(\mathbf{x}) \geq \mathbf{f}_1(\mathbf{x'})}_{\mathbf{f}_1 \text{ not increasing}} \wedge \underbrace{\mathbf{f}_2(\mathbf{x}) > \mathbf{f}_2(\mathbf{x'})}_{\mathbf{f}_2 \text{ decreasing}} \Big)$$

$$\forall \mathbf{x}\mathbf{x'}.\ \mathit{Loop}(\mathbf{x}, \mathbf{x'}) \rightarrow \mathrm{T}(\mathbf{x}, \mathbf{x'})$$

Lemma (Farkas)

$$\forall x.\ (\ldots \rightarrow \ldots) \quad \mathit{iff} \quad \exists \vec{\lambda}\ (\ldots)$$

Theorem (Motzkin)

$$\forall x.\ \neg(\ \ldots \leq \ldots \ \wedge\ \ldots < \ldots\ ) \quad \mathit{iff} \quad \exists \vec{\lambda}\ (\ldots)$$

Ranking Template

$\forall \mathbf{x}\mathbf{x}'.\ \mathit{Loop}(\mathbf{x}, \mathbf{x}') \rightarrow \mathrm{T}(\mathbf{x}, \mathbf{x}')$

- "building blocks" linear functions $f(\mathbf{x}) = \mathbf{c}_1 \cdot \mathbf{x}_1 + \cdots + \mathbf{c}_n \cdot \mathbf{x}_1 + \mathbf{c}_0$

Ranking Template

$\forall \mathbf{x}\mathbf{x'}.\ \mathit{Loop}(\mathbf{x}, \mathbf{x'}) \rightarrow \mathrm{T}(\mathbf{x}, \mathbf{x'})$

- "building blocks" linear functions $f(\mathbf{x}) = \mathbf{c}_1 \cdot \mathbf{x}_1 + \cdots + \mathbf{c}_n \cdot \mathbf{x}_1 + \mathbf{c}_0$
- boolean combinations of linear inequalities (Motzkin applicable)

Ranking Template

$\forall \mathbf{x}\mathbf{x}'.\ \mathit{Loop}(\mathbf{x}, \mathbf{x}') \rightarrow \mathrm{T}(\mathbf{x}, \mathbf{x}')$

- "building blocks" linear functions $f(\mathbf{x}) = \mathbf{c}_1 \cdot \mathbf{x}_1 + \cdots + \mathbf{c}_n \cdot \mathbf{x}_1 + \mathbf{c}_0$
- boolean combinations of linear inequalities (Motzkin applicable)
- well-founded

Ranking Template

$\forall \mathbf{x}\mathbf{x'}.\ \mathit{Loop}(\mathbf{x}, \mathbf{x'}) \rightarrow \mathrm{T}(\mathbf{x}, \mathbf{x'})$

- "building blocks" linear functions $f(\mathbf{x}) = \mathbf{c}_1 \cdot \mathbf{x}_1 + \cdots + \mathbf{c}_n \cdot \mathbf{x}_1 + \mathbf{c}_0$
- boolean combinations of linear inequalities (Motzkin applicable)
- well-founded

Definition (Linear ranking template)

A linear ranking template $\mathrm{T}(\mathbf{x}, \mathbf{x'})$ is a

- boolean combination whose atoms are of the following form

$$\sum_{\mathbf{f} \in F} \alpha_{\mathbf{f}} \cdot \mathbf{f}(\mathbf{x}) + \beta_{\mathbf{f}} \cdot \mathbf{f}(\mathbf{x'}) \ \rhd \ 0 \ ,$$

  where each $\mathbf{f}(\mathbf{x})$ is an affine-linear term $\mathbf{c}_1 \cdot \mathbf{x}_1 + \cdots + \mathbf{c}_n \cdot \mathbf{x}_1 + \mathbf{c}_0$, each $\alpha_{\mathbf{f}}$, and $\beta_{\mathbf{f}}$ is a constant and $\rhd \in \{\geq, >\}$.
- such that each instance of $\mathrm{T}(\mathbf{x}, \mathbf{x'})$ defines a well-founded relation.

- affine-linear ranking function
  template $\mathrm{T}_{\mathsf{affine}}(\mathbf{x}, \mathbf{x'})$

Lemma

the template $\mathrm{T}_{\mathit{affine}}(\mathbf{x}, \mathbf{x'})$ is a linear ranking template.

- linear lexicographic ranking function
  k-lexicographic template $\mathrm{T}_{\mathsf{k\text{-}lex}}(\mathbf{x}, \mathbf{x'})$

Lemma

the template $\mathrm{T}_{\mathit{k\text{-}lex}}(\mathbf{x}, \mathbf{x'})$ is a linear ranking template, for each k

- piecewise linear ranking function
  k-piece template $\mathrm{T}_{\mathsf{k\text{-}piece}}(\mathbf{x}, \mathbf{x'})$

Lemma

the template $\mathrm{T}_{\mathit{k\text{-}piece}}(\mathbf{x}, \mathbf{x'})$ is a linear ranking template, for each k

- multiphase linear ranking function
  k-phase template $\mathrm{T}_{\mathsf{k\text{-}phase}}(\mathbf{x}, \mathbf{x'})$

Lemma

the template $\mathrm{T}_{\mathit{k\text{-}phase}}(\mathbf{x}, \mathbf{x'})$ is a linear ranking template, for each k

Why has no one used this method before?

Our explanation: recent progress in solving nonlinear arithmetic

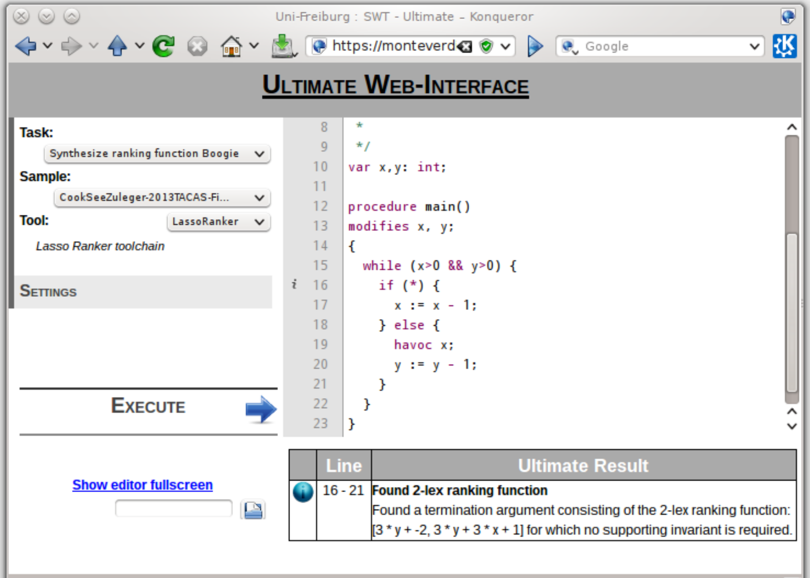
Jovanovic,Moura	**Solving non-linear arithmetic**	(IJCAR 2012)

SMT solver Z3

`http://z3.codeplex.com/`

Our tool: LassoRanker

`http://ultimate.informatik.uni-freiburg.de/LassoRanker/`



New kind of ranking function?

You can synthesize your new ranking function automatically in three steps.

- write down a template $\mathrm{T}(\mathbf{x}, \mathbf{x'})$ for this ranking function
- prove that each instance of $\mathrm{T}(\mathbf{x}, \mathbf{x'})$ is a well-founded relation
- add template $\mathrm{T}(\mathbf{x}, \mathbf{x'})$ to our tool LassoRanker