Testing for Termination with Monotonicity Constraints

Michael Codish

V. Lagoon   P. J. Stuckey

Department of Computer Science
Ben Gurion University

Department of CS and SE
University of Melbourne

ICLP 2005

Testing for Termination with Monotonicity Constraints

Michael Codish

Department of Computer Science
Ben Gurion University

V. Lagoon

Cadence

The MathWorks, Inc

P. J. Stuckey

Department of CS and SE
University of Melbourne

Dept. of Computing and
Information Systems

ICLP 2015

ICLP 2005

Termination Analysis
a logic programming success story

The whole concept of

"One Loop at a Time"

appeared very early in LP – way before
we understood that it was correct (and why)
(local ranking functions / per loop)

And, later became pivotal in many other termination analyzers (imperative languages)

The Early LP Termination Analyzers Used MC's (and other abstract domains)

Monotonicity Constraints

1989: Alexander Brodsky, Yehoshua Sagiv

A Semantic Basis for Termination of Logic Programs

1997: Michael Codish, Cohavit Taboch

Size Change Graphs

2001: Chin-Soon Lee, Neil Jones, Amir Ben-Amram

The Early LP Termination Analyzers
Used MC's (and other abstract domains)

TermiLog

1997: Lindenstrauss, Sagiv

1997: Lindenstrauss, Sagiv, Serebrenik

2001: Dershowitz, Linenstrauss, Sagiv, Serebrenik

TerminWeb

1997: Codish, Taboch

2003: Codish, Genaim (proving termination one loop at a time)

But. Implemented a "Test" for termination for each MC that later turned out to be the same as one used for SCG's (and "complete" for SCG's)

On our minds back in 2005

For a given abstraction, is there a complete form for a ranking function?

For a description of a program in the given abstraction, if there exists any proof of termination, then there exists one of the prescribed form.

What is the form of the ranking function? Is the SCG test complete for MC's? If not, then what is the complete form?

This question comes up both for standard (Global) ranking functions as well as for the (Local) ones we were considering.

Semantics for Termination

concrete:

abstract:

$$\text{PROGRAM} \Rightarrow \left\{ \begin{array}{c} \vdots \\ p(\bar{x}) \leftarrow \mu,\ p(\bar{y}) \\ \vdots \end{array} \right.$$

Herbrand Constraints
(Codish, Taboch, 1997)

Size-Change Graphs
$x_i > y_j,\ x_i \geq y_j,$
$x_i \in \bar{x},\ y_j \in \bar{y}$
(Lee,Jones,Ben-Amram 2001)

Monotonicity Constraints
$v > w,\ v \geq w,$
$v, w \in (\bar{x} \cup \bar{y})$
(Lindenstrauss,Sagiv, 1989)

Linear Constraints
$\sum c_i v_i > c$
$v_i \in (\bar{x} \cup \bar{y})$
(Podelski,Rybalchenko, 2004)

Semantics for Termination

concrete:

Herbrand Constraints (Codish, Taboch, 1987)

$$\text{PROGRAM} \Rightarrow \left\{ \begin{array}{c} \vdots \\ p(\bar{x}) \leftarrow \mu, p(\bar{y}) \\ \vdots \end{array} \right.$$

abstract:

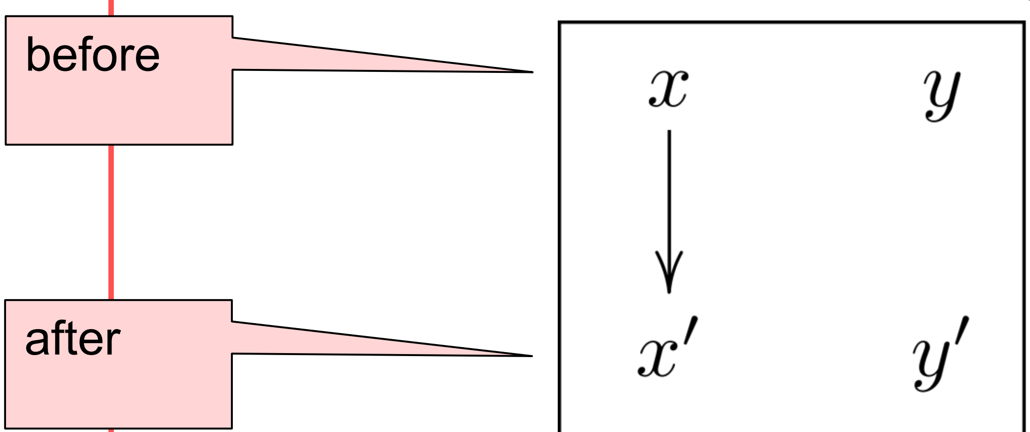
Many others since then

Size-Change Graphs by Example

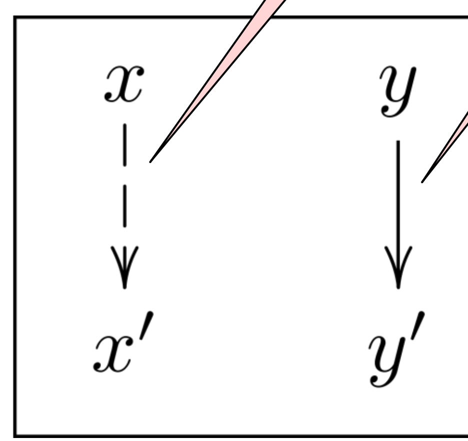
int Ack(int x, int y) {
    if $(x{=}{=}0)$ **return** $y+1$
    else if $(y{=}{=}0)$ **return** $\mathrm{Ack}(x-1,\ 1)$
    else return $\mathrm{Ack}(x-1,\ \mathrm{Ack}(x,\ y-1))$
}

$\mathrm{Ack}(x,\ y) \leftarrow \mathrm{Ack}(x-1,\ 1)$

$\mathrm{Ack}(x,\ y) \leftarrow \mathrm{Ack}(x-1,\ \mathrm{Ack}(x,\ y-1))$

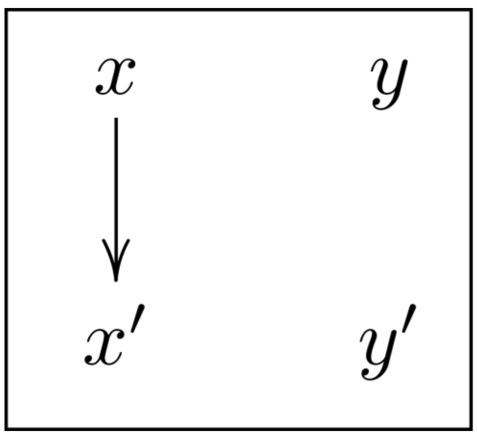


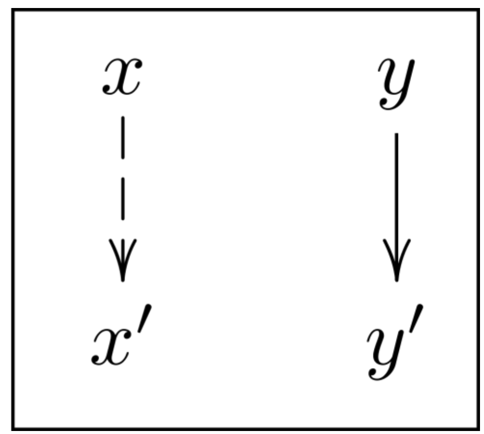
$\mathrm{Ack}(x,\ y) \leftarrow \mathrm{Ack}(x,\ y-1)$



Termination is Decidable for SCG's

2 loops



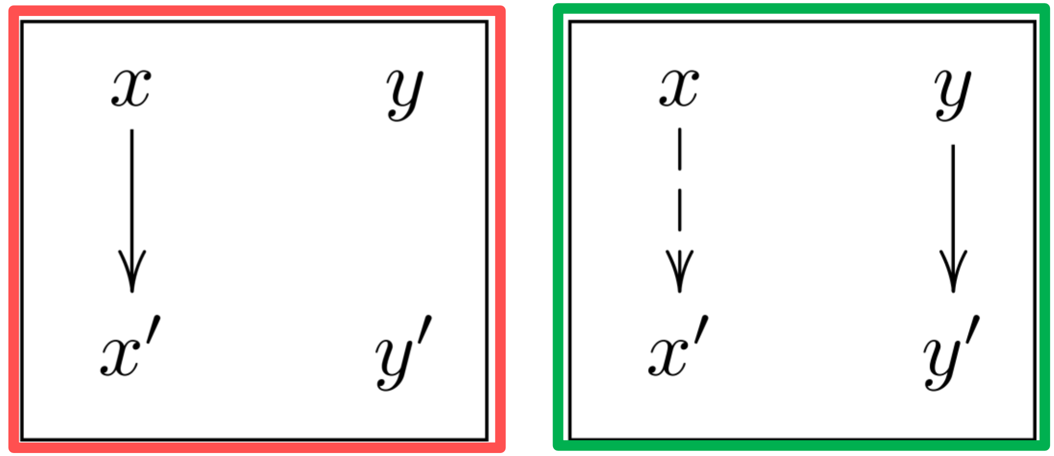


Global Approach: find a ranking function f

$$\exists f\ \forall loop.\ f \text{ decreases on the } loop$$

$$Ack(x, y)$$

$$f(x, y) = \langle x, y \rangle$$



Local Approach: (for a **price***, we can prove termination) ***one loop at a time***

$$\forall loop\ \exists f.\ f \text{ decreases on the } loop$$

$$f_1(x, y) = x \qquad f_2(x, y) = y$$

***price:** the loops are closed under composition

prize: the ranking functions are simpler

***price:** the loops are closed under composition

1. $p(x_1, x_2, x_3) \leftarrow x_1 > y_2, x_2 \geq y_2, x_3 > y_3, p(y_1, y_2, y_3).$
2. $p(x_1, x_2, x_3) \leftarrow x_1 > y_1, x_2 \geq y_1, p(y_1, y_2, y_3).$
3. $p(x_1, x_2, x_3) \leftarrow x_1 > y_2, x_2 > y_2, p(y_1, y_2, y_3).$

local	ranking functions

1. $f_1(u_1, u_2, u_3) = u_3$

2. $f_2(u_1, u_2, u_3) = u_1$

3. $f_3(u_1, u_2, u_3) = u_2$

*price: the loops are closed under composition

prize: the ranking functions are simpler

1. $p(x_1, x_2, x_3) \leftarrow x_1 > y_2, x_2 \geq y_2, x_3 > y_3, p(y_1, y_2, y_3).$
2. $p(x_1, x_2, x_3) \leftarrow x_1 > y_1, x_2 \geq y_1, p(y_1, y_2, y_3).$
3. $p(x_1, x_2, x_3) \leftarrow x_1 > y_2, x_2 > y_2, p(y_1, y_2, y_3).$

local	ranking functions	global

1. $f_1(u_1, u_2, u_3) = u_3$
2. $f_2(u_1, u_2, u_3) = u_1$
3. $f_3(u_1, u_2, u_3) = u_2$

3. $min(u_1, u_2)$

1,2. $\langle min(u_1, u_2), u_3 \rangle$

2,3. $\langle min(u_1, u_2), u_1 \rangle$

1,2,3. $*$

$*$ One can verify that there does not exist any function based on lexicographic ordering of linear functions (even allowing minimum and maximum functions) that is a global ranking function for this example.

On our minds back in 2005

For a SCGs, if there exists any proof of termination, then there exists one of the prescribed form.

What is the form of the ranking function?

Local:

price: exponentially many loops to consider;
prize: simple ranking functions.

Global: (Chin Soon Lee TOPLAS 2009)

The ranking function might be of triple exponential size

Correctness of the local approach

Let G be a set of size-change graphs. If every $\mu \in G^*$ has a ranking function then any program described by G terminates. (Dershowitz *et al*, 2001) (Lee,Jones,Ben-Amram 2001)

Ramsey's theorem (1930): Let X be some countably infinite set and colour the pairs in $X \times X$ in a finite number C of different colours. Then there exists some infinite $M \subset X$ such that the pairs of M all have the same colour.

Completeness (POPL 2001)

If there exists a ranking function then there exists one of a specified form

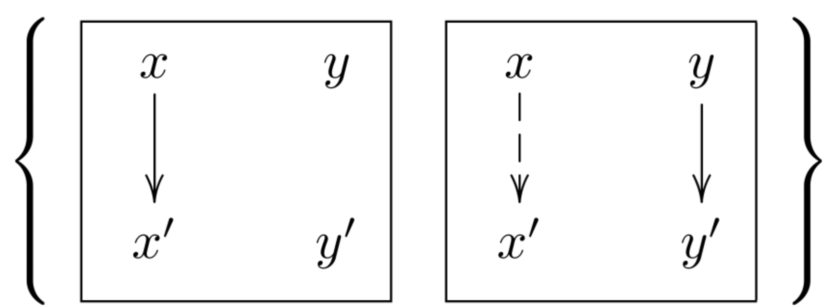
For an idempotent SCG if there is any ranking function f then there is one of the form $f(\bar{x}) = x_i$. (Lee etal, 2001)

The algorithm:

1. Compute the closure G^*
2. Compute the subset of idempotent graphs $I \subseteq G^*$
3. For each $\mu(\bar{x}, \bar{y}) \in I$ check that $\exists i . \mu(\bar{x}, \bar{y}) \rightarrow (x_i > y_i)$

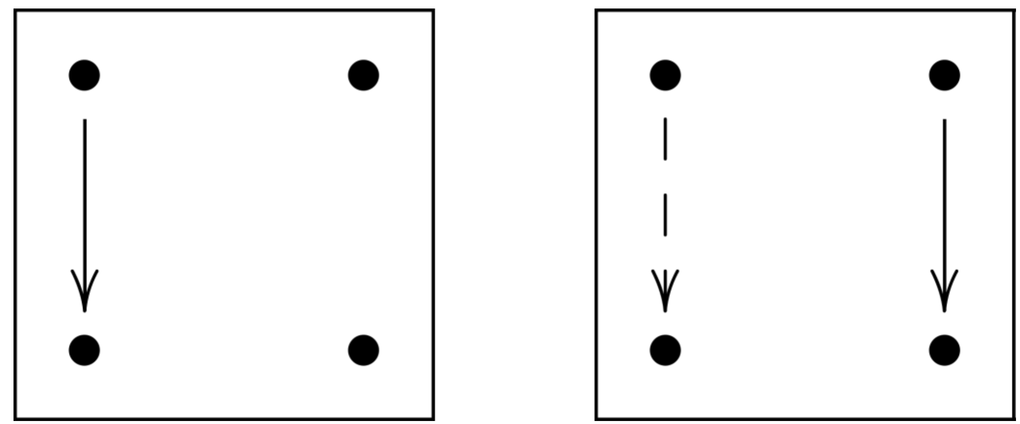
Example: $Ack(x, y)$ is terminating

$I = G^* =$

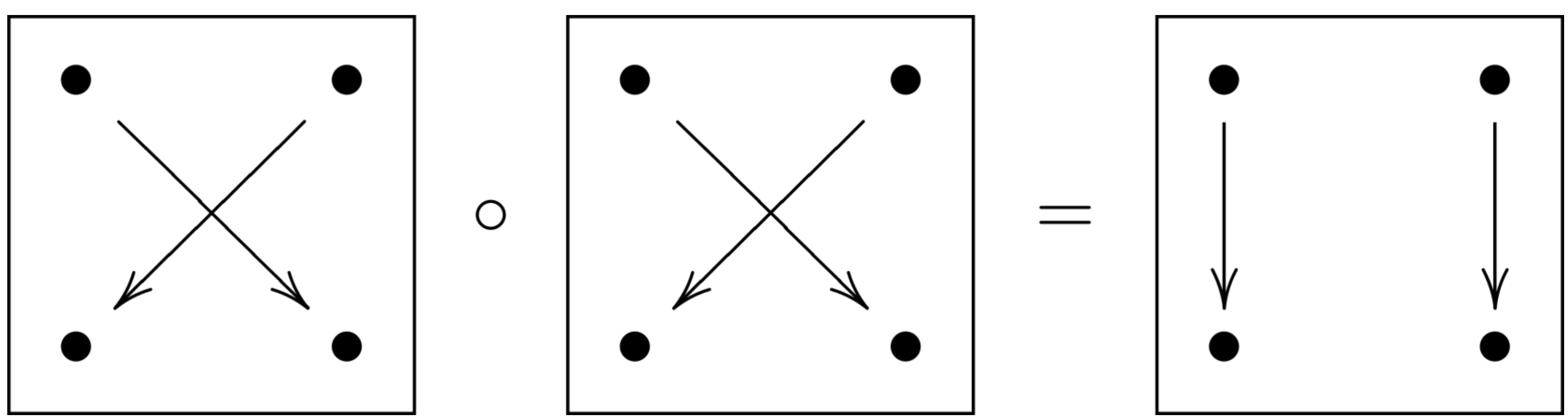


Idempotence

A size-change graph μ is idempotent if $\mu \circ \mu = \mu$. The two graphs of Ack are idempotent



A non-idempotent graph:



Completeness (ICLP 2005)

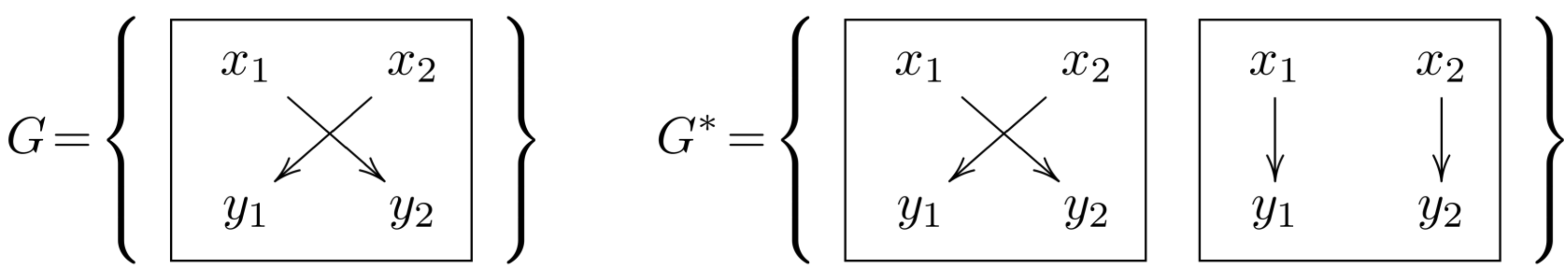
If there exists a ranking function then there exists one of a specified form

For any SCG if there is a ranking function f then there is one of the form $f(\bar{x}) = \sum_{i \in I} x_i$.

The algorithm:

1. Compute the closure G^*
2. For each $\mu(\bar{x}, \bar{y}) \in G^*$ check that $\mu(\bar{x}, \bar{y}) \models \bigvee_i (x_i > y_i)$

Example:

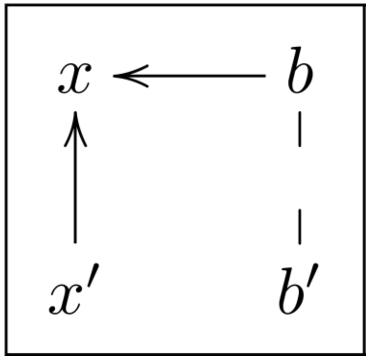


$$\mu(\bar{x}, \bar{y}) \models \bigvee_i (x_i > y_i) \;\equiv\; \neg\Big(\mu(\bar{x}, \bar{y}) \wedge \bigwedge_i (x_i \le y_i)\Big)$$

Monotonicity Constraints

Example: **while** `x<b` **do** `x=x+1`

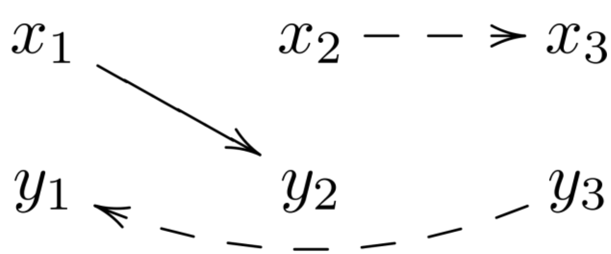
$$\mu(\langle x,b\rangle, \langle x',b'\rangle) = (x < x', x < b, b = b')$$



Neither $f(x,b) = x$ nor $f(x,b) = b$ is a ranking function.
A ranking function is $f(x,b) = b - x$.

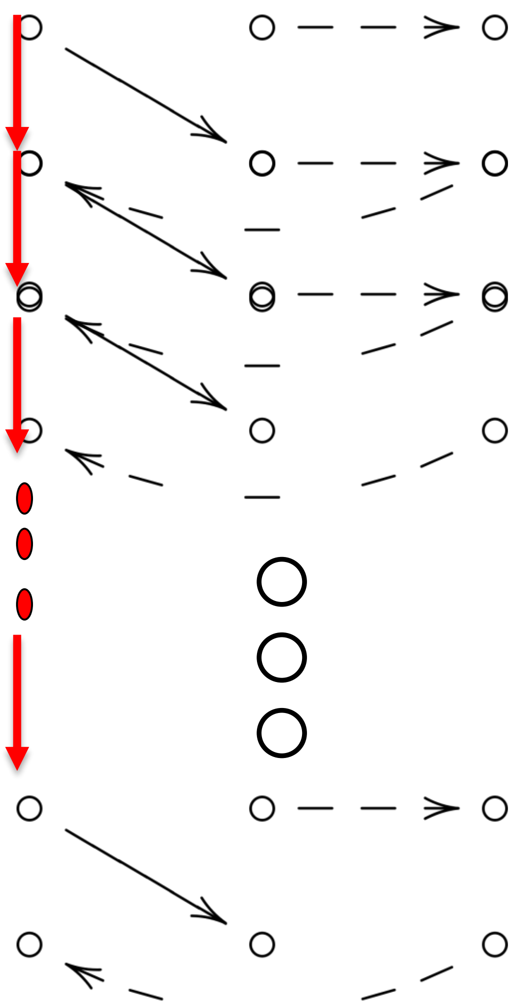
Is that form complete?

Balanced Monotonicity Constraints



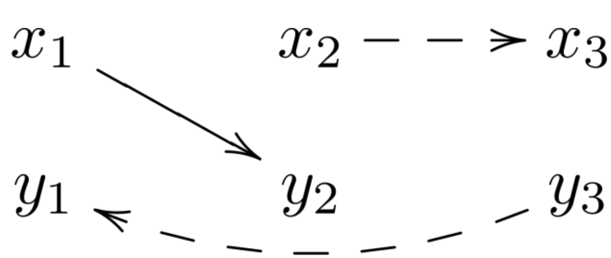
Idempotent

Balanced Monotonicity Constraints



no infinite decent

Balanced Monotonicity Constraints

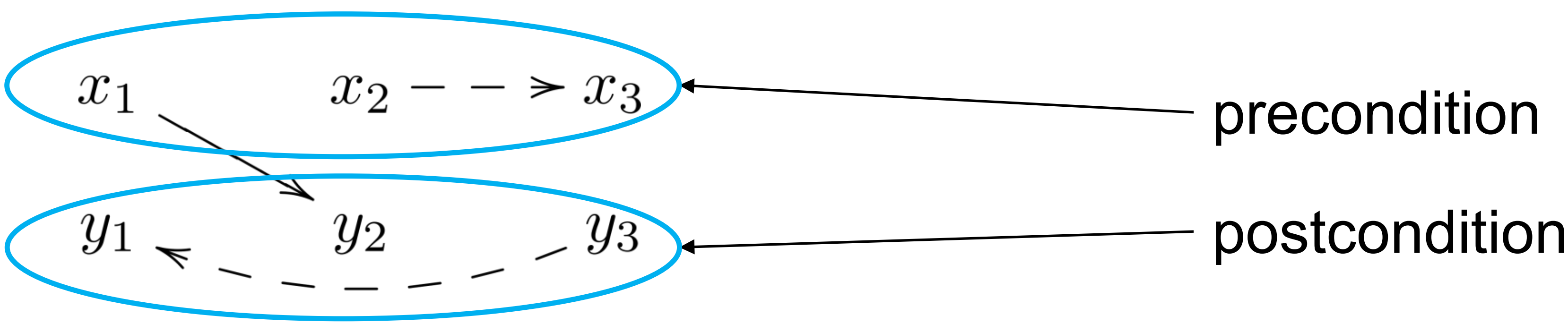


But no direct
down arc

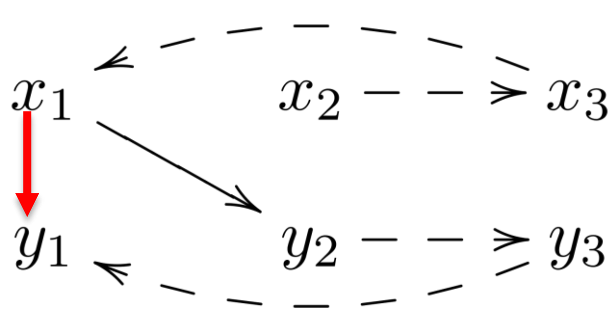
Balanced Monotonicity Constraints

Definition: A monotonicity constraint $\mu(\bar{x}, \bar{y})$ is *balanced* if

$$\mu(\bar{x}, \bar{y}) \models x_i \succ^b x_j \;\Leftrightarrow\; \mu(\bar{x}, \bar{y}) \models y_i \succ^b y_j$$



The balanced extension



Not a limitation because a postcondition is a precondition next turn around the loop

Completeness for Monotonicity Constraints

Theorem: for a *balanced idempotent* monotonicity constraint $\mu(\bar{x}, \bar{y})$ if there is a ranking function then there is a ranking function of the form $f(\bar{x}) = x_i$ or of the form $f(\bar{x}) = x_i - x_j$

Theorem: for a *balanced* monotonicity constraint $\mu(\bar{x}, \bar{y})$ if there is a ranking function then there is a linear ranking function.

à coeff = -1, 0, 1

The notion of balancing turned out to be important

Ben-Amram used it when extending the MC/SCG frameworks to constraints over any well-founded domain [CAV 2009], and then to constraints over integers [LMCS 2011]

Bozzelli and Pinchinat [VMCAI 2012] used it to extend the MC/SCG frameworks to Gap constraints

Bozga et al. [TACAS 2012] used it to show that a single loop with octagonal constraints terminates iff it eventually (i.e., after balancing) has a linear ranking function.

Conclusion

Sometimes – its all about how you ask the question

What is the form of the ranking function?
Is it complete for the given abstraction?

And, Sometimes the technicalities are “important”

The ICLP 2005 slides did not even include
the word “balanced” !