Testing for Termination with Monotonicity Constraints

Michael Codish[1]*, Vitaly Lagoon[2], and Peter J. Stuckey[2,3]

[1] Department of Computer Science, Ben-Gurion University, Israel
[2] Department of Computer Science and Software Engineering
The University of Melbourne, Australia
[3] NICTA Victoria Laboratory
mcodish@cs.bgu.ac.il, {lagoon,pjs}@cs.mu.oz.au

Abstract. Termination analysis is often performed over the abstract domains of monotonicity constraints or of size change graphs. First, the transition relation for a given program is approximated by a set of descriptions. Then, this set is closed under a composition operation. Finally, termination is determined if all of the idempotent loop descriptions in this closure have (possibly different) ranking functions. This approach is complete for size change graphs: An idempotent loop description has a ranking function if and only if it has one which indicates that some specific argument decreases in size. In this paper we generalize the size change criteria for size change graphs which are not idempotent. We also illustrate that proving termination with monotonicity constraints is more powerful than with size change graphs and demonstrate that the size change criteria is incomplete for monotonicity constraints. Finally, we provide a complete termination test for monotonicity constraints.

1 Introduction

Termination analysis is often performed by approximating the transition relation induced by a program. For logic programs this is a relation on the calls to predicates encountered during computation. A transition from call $p(\bar{t})$ to a subsequent call $q(\bar{s})$ in some computation can be represented as a binary clause of the form $p(\bar{t}) \leftarrow q(\bar{s})$. A semantics which specifies this transition relation is introduced and shown to make calls observable in [6]. It is shown to make termination observable in [2]. The TerminWeb termination analyzer for logic programs [2] is basically, a meta-interpreter for an abstraction of this semantics with transitions approximated by monotonicity constraints [1].

Size change graphs were introduced in [7] and are similar to monotonicity constraints. These two domains are used by an increasing number of termination analyzers for a variety of languages including: TermiLog [9, 8] and TerminWeb [2] for logic programs, implementations for simple first-order functional languages [13, 4] and the AProVE analyzer for term rewrite systems [12].

* Research performed at the University of Melbourne

Monotonicity constraints and size change graphs can be represented as (abstract) binary clauses of the form $p(\bar{x}) \leftarrow \mu(\bar{x}, \bar{y}), q(\bar{y})$ where $\bar{x}$ and $\bar{y}$ are tuples of distinct variables and $\mu(\bar{x}, \bar{y})$ is a conjunction of binary constraints of the form $u > v$,$u \geq v$ on the sizes of the data before and after a corresponding concrete transition. For monotonicity constraints u and v are any variables among $\bar{x}$ and $\bar{y}$. Size change graphs are more restricted with u in $\bar{x}$ and v in $\bar{y}$. When p and q are the same symbol the (abstract) binary clause is recursive and describes a loop.

Example 1. *Consider the predicate* `ackerman/3` *below (on the left) which computes Ackerman's function. The size change graphs (on the right) describe the induced transition relation. In subsequent calls to this predicate, either the first argument decreases in size or else it does not increase in size and the second argument decreases in size.*

$ackerman(0, N, s(N)).$
$ackerman(s(M), 0, Res) \leftarrow$
$\quad ackerman(M, s(0), Res).$
$ackerman(s(M), s(N), Res) \leftarrow$
$\quad ackerman(s(M), N, Res1),$
$\quad ackerman(M, Res1, Res).$

$ackermann(x_1, x_2, x_3) \leftarrow$
$\quad x_1 > y_1, ackermann(y_1, y_2, y_3).$
$ackermann(x_1, x_2, x_3) \leftarrow$
$\quad x_1 \geq y_1, x_2 > y_2, ackermann(y_1, y_2, y_3).$

This paper is not concerned with how approximations of transition relations are obtained, but rather with the question of how termination is proven given such an approximation. Existing analyzers provide the approximations as a starting point for this paper.

In the classic approach, to prove termination of a program P one should identify a ranking function f from program states to the elements of a well founded domain and show that f decreases as execution proceeds through all of the loops in P. For example, one might show that the function $f(u_1, u_2, u_3) = \langle u_1, u_2 \rangle$ decreases with respect to the lexicographic ordering for both of the loop descriptions in Example 1. This is a *global* ranking function — it is shown to decrease for all loop descriptions in the analysis.

An alternative approach is based on the application of *local* ranking functions. In this approach, under the condition that the set of loop descriptions is "closed under composition" (resolution of abstract binary clauses), termination is guaranteed if for each individual loop description a (possibly different) ranking function is shown to decrease when execution goes through that loop.

The main advantage in applying local ranking functions is that they take a simpler form than corresponding global ranking functions and are easy to find automatically. Moreover, it is sufficient to find ranking functions only for those descriptions which are idempotent (a description is idempotent if it remains invariant when composed with itself) [7]. There is also a disadvantage: as illustrated in [7], for size change graphs, there is a worst case exponential growth factor (in the number of arguments) associated with the computation of closure under composition. The following example illustrates the advantage. The disadvantage is the topic of another paper.

Example 2. *Consider the following three loop descriptions (which are idempotent and closed under composition)* [1]*.*

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

Local ranking functions are respectively $f_1(u_1, u_2, u_3) = u_3$*,* $f_2(u_1, u_2, u_3) = u_1$*, and* $f_3(u_1, u_2, u_3) = u_2$*. The function* $min(u_1, u_2)$ *decreases for the third loop description and does not increase for the first two descriptions. The functions* $\langle min(u_1, u_2), u_3\rangle$ *and* $\langle min(u_1, u_2), u_1\rangle$ *decrease (with respect to the lexicographic ordering) for the first two and last two loop descriptions respectively. 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.*

The correctness of the local approach is first given by Dershowitz *et al.* [3] and is based on the application of Ramsey's Theorem [11]. The approach is also complete [7] in the sense that an idempotent size change graph has a ranking function if and only if it has one of the form $f(\bar{u}) = u_i$. Hence an algorithm to decide termination for size change graphs is obtained.

The first contribution of this paper is to generalize the completeness result for size change graphs which are not necessarily idempotent. Here, if there exists any ranking function then there exists one of the form $f(u_1, \ldots, u_n) = \Sigma a_i u_i$ with all coefficients $a_i \in \{0, 1\}$. In [8] the authors suggest a termination test for monotonicity constraints which is the one implemented in TermiLog and TerminWeb. We show that this test provides a simple decision procedure for the existence of a ranking function for a size change graph (idempotent or not).

We proceed to illustrate that size change termination is incomplete for monotonicity constraints which are not size change graphs. This fact has been overlooked until now. Both TermiLog and TerminWeb implement for monotonicity constraints the test which is complete for size change graphs. The second contribution of this paper is to provide completeness results for monotonicity constraints: for an idempotent monotonicity constraint, if there exists any ranking function then there exists one of the form $f(u_1, \ldots, u_n) = u_i$ or of the form $f(u_1, \ldots, u_n) = u_i - u_j$. For arbitrary monotonicity constraints if there exists a ranking function then there exists one which is linear.

In [10], the authors present an efficient test for termination for loop descriptions for a domain which is more general than monotonicity constraints. Their approach is complete with respect to linear ranking functions: if there exists a linear ranking function then the proposed procedure will succeed (and synthesize it). However, if the procedure fails, it could be the case that there exists a ranking function which is non-linear. Our result implies that the test presented in [10] is complete for monotonicity constraints.

The remainder of this paper is structured as follows: Section 2 introduces monotonicity constraints and size change graphs. Section 3 describes the com-

[1] This example was suggested by Amir Ben Amram.

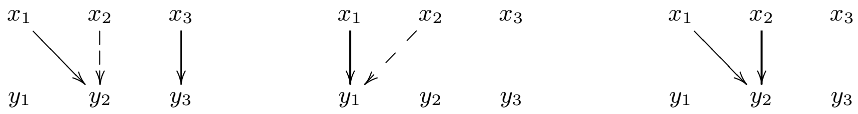


Fig. 1. Size change graphs from Example 2.

pleteness result for idempotent size change graphs and extends it for arbitrary size change graphs. Section 4 illustrates that size change termination is incomplete for monotonicity constraints and provides two completeness results: first for idempotent descriptions and second for descriptions which are not necessarily idempotent. Section 5 concludes.

2 Monotonicity Constraints and Size change graphs

Let $\bar{x} = \langle x_1, \dots, x_n \rangle$ and $\bar{y} = \langle y_1, \dots, y_n \rangle$ denote n-tuples of variables taking non-negative integer values. When clear from the context we let these denote the corresponding sets of variables. Intuitively, these values correspond to the sizes of terms in a computation, with respect to a given norm function.

Definition 1 (monotonicity constraint, size change graph). *A monotonicity constraint is a binary clause of the form $p(\bar{x}) \leftarrow \mu(\bar{x}, \bar{y}), q(\bar{y})$ where $\mu(\bar{x}, \bar{y})$ is a conjunction of constraints of the form $u \geq v + b$, denoted also $u \succ^b v$, with $u, v \in \bar{x} \cup \bar{y}$ and $b \in \{0, 1\}$. We write also $u > v$ and $u \geq v$ when respectively $b = 1$ or $b = 0$ or $u \succ v$ when not distinguishing between the two cases. If constraints are restricted so that $u \in \bar{x}$, $v \in \bar{y}$ then $\mu(\bar{x}, \bar{y})$ is called a size change graph. When clear from the context we refer to $\mu(\bar{x}, \bar{y})$ as the monotonicity constraint (or size change graph).*

A monotonicity constraint $\mu(\bar{x}, \bar{y})$ can be viewed as a directed graph with nodes $\bar{x} \cup \bar{y}$ and an edge labeled by b from u to v if and only if $\mu(\bar{x}, \bar{y}) \models u \succ^b v$. For size change graphs this view gives a directed bipartite graph. In the examples, graphs are depicted with solid and dashed arrows representing edges of the form $u > v$ and $u \geq v$ respectively. We often omit edges which can be inferred from those drawn. A monotonicity constraint is satisfiable if and only if its graph representation has no cycle with a solid edge. Note that a size change graph is always satisfiable. The size change graphs from Example 2 are depicted in Figure 1.

Monotonicity constraints induce corresponding transition relations.

Definition 2 (transition relation).
A monotonicity constraint $p(\bar{x}) \leftarrow \mu(\bar{x}, \bar{y}), q(\bar{y})$ induces a transition relation $\sqsubseteq_\mu$ on labeled vectors of non-negative integers given by $p(\bar{a}) \sqsubseteq_\mu q(\bar{b})$ if and only if $\mu(\bar{a}, \bar{b})$ is valid. When clear from the context we drop the labels p and q.

A derivation for a set of monotonicity constraints is a chain in the corresponding transition relations. For the completeness results of this paper it is sufficient to consider derivations induced by a single recursive monotonicity constraint.

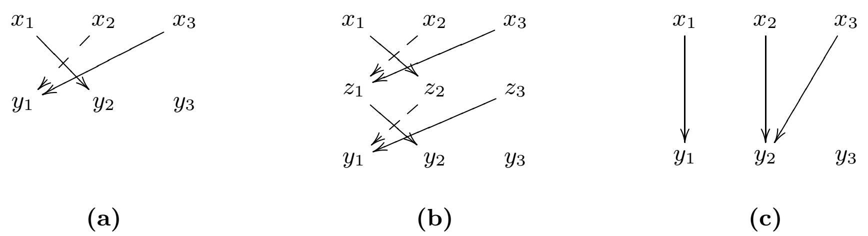


Fig. 2. Constraints of Example 3: **(a)** monotonicity constraint $\mu(\bar{x}, \bar{y})$, **(b)** derivation constraint $\mu(\bar{x}, \bar{z}, \bar{y})$, and **(c)** self composition $\mu^2(\bar{x}, \bar{y})$.

Definition 3 (derivation, derivation constraint). *Let $p(\bar{x}) \leftarrow \mu(\bar{x}, \bar{y}), p(\bar{y})$ be a recursive monotonicity constraint. A derivation of $\mu(\bar{x}, \bar{y})$ is a chain of non-negative integer vectors of the form $\bar{a}^0 \sqsubseteq_\mu \bar{a}^1 \sqsubseteq_\mu \cdots \sqsubseteq_\mu \bar{a}^k$ which may also be infinite. For a finite derivation, there is a corresponding derivation constraint $\mu(\bar{x}^0, \ldots, \bar{x}^k) = \mu(\bar{x}^0, \bar{x}^1) \wedge \cdots \wedge \mu(\bar{x}^{k-1}, \bar{x}^k)$. A derivation constraint can also be viewed as a directed graph.*

The composition of monotonicity constraints is defined through renaming of variables (graph nodes), conjunction and entailment. Derivation constraints and composition are illustrated in Figure 2.

Definition 4 (composition of monotonicity constraints). *The composition of monotonicity constraints $\mu_1(\bar{x}, \bar{y})$ and $\mu_2(\bar{x}, \bar{y})$ is given by*

$$\mu_1(\bar{x}, \bar{y}) \circ \mu_2(\bar{x}, \bar{y}) = \bigwedge \left\{ u \succ^b v \,\middle|\, u, v \in \bar{x} \cup \bar{y}, \mu_1(\bar{x}, \bar{z}) \wedge \mu_2(\bar{z}, \bar{y}) \models u \succ^b v \right\}.$$

We denote by $\mu^k(\bar{x}, \bar{y})$ the composition of $\mu(\bar{x}, \bar{y})$ with itself k times.

Definition 5 (idempotent monotonicity constraints). *A monotonicity constraint, $\mu(\bar{x}, \bar{y})$, is* idempotent *if and only if $\mu^2(\bar{x}, \bar{y}) = \mu(\bar{x}, \bar{y})$.*

Example 3. *The monotonicity constraint $\mu(\bar{x}, \bar{y}) = x_1 > y_2 \wedge x_2 \geq y_1 \wedge x_3 > y_1$ is depicted as Figure 2(a). The derivation constraint $\mu(\bar{x}, \bar{z}, \bar{y})$ is shown in Figure 2(b) and consists of 2 copies of $\mu(\bar{x}, \bar{y})$. The self composition $\mu^2(\bar{x}, \bar{y})$ is depicted in Figure 2(c). The constraint of Figure 2(a) is not idempotent but the constraint of Figure 2(c) is. Note that $\mu(\bar{x}, \bar{z}, \bar{y}) \models x_3 \geq y_2 + 2$ while $\mu^2(\bar{x}, \bar{y}) \models x_3 \geq y_2 + 1$. This illustrates that (projections of) derivation constraints are not monotonicity constraints.*

Definition 6 (closure under composition). *Let $\mathcal{G}$ be a set of monotonicity constraints. We denote by $\mathcal{G}^*$ the closure of $\mathcal{G}$ under composition. This is the smallest superset of $\mathcal{G}$ such that if $\mu_1(\bar{x}, \bar{y}) \in \mathcal{G}^*$ and $\mu_2(\bar{x}, \bar{y}) \in \mathcal{G}^*$ then also $\mu_1(\bar{x}, \bar{y}) \circ \mu_2(\bar{x}, \bar{y}) \in \mathcal{G}^*$.*

A central notion when proving termination is that of a ranking function. We focus on ranking functions for individual monotonicity constraints.

Definition 7 (ranking function). *A ranking function for a monotonicity constraint $\mu(\bar{x}, \bar{y})$ is a mapping f from tuples of non-negative integers to a well founded domain $(D, \prec_D)$ such that $\mu(\bar{x}, \bar{y}) \models f(\bar{y}) \prec_D f(\bar{x})$.*

In this paper we often choose ranking functions mapping to the domain of natural numbers $(\mathcal{N}, <)$ with the standard ordering.

Example 4. *Consider the size change graph $\mu(\bar{x}, \bar{y})$ from Figure 2(a). The function $f(\bar{u}) = u_1 + u_2$ on the natural numbers is a ranking function for $\mu(\bar{x}, \bar{y})$.*

Theorem 1 (correctness [3, 7]). *Let $\mathcal{G}$ be a finite set of monotonicity constraints which approximates the loops in the transition relation of a program P. If for each of the idempotent monotonicity constraints in $\mathcal{G}^*$ there exists a (possibly different) ranking function then P terminates.*

The proofs in [3] and [7] are for monotonicity constraints and size change graphs respectively. The proofs are essentially the same and presented as applications of Ramsey's Theorem [11].

3 Completeness for Size Change Graphs

In [7] the authors present a completeness result for size change graphs. We rephrase this result in terms of ranking functions.

Theorem 2 (completeness – idempotent size change graphs). *Let $\mu(\bar{x}, \bar{y})$ be an idempotent size change graph. If there exists a ranking function for $\mu(\bar{x}, \bar{y})$ then there exists one mapping to $(\mathcal{N}, <)$ of the form $f(u_1, \ldots, u_n) = u_i$.*

We present a proof, different from the one in [7]. Following this proof will help understand its generalizations in the remainder of this paper. The proof relies on the following two lemmata.

Lemma 1. *If $\mu(\bar{x}, \bar{y})$ is an idempotent monotonicity constraint which implies $x_i \succ^{b_1} y_j$ and $x_j \succ^{b_2} y_k$ (or $y_i \succ^{b_1} x_j$ and $y_j \succ^{b_2} x_k$) then it implies also $x_i \succ^{b_1 \vee b_2} y_k$ (or $y_i \succ^{b_1 \vee b_2} x_k$).*

Proof. By definitions of composition and idempotence.

Lemma 2. *If $\mu(\bar{x}, \bar{y})$ is an idempotent size change graph, then either for some argument i, $\mu(\bar{x}, \bar{y}) \models x_i > y_i$; or $\mu(\bar{x}, \bar{y}) \wedge \bar{x} = \bar{y}$ is satisfiable.*

Proof. Let $\mu(\bar{x}, \bar{y})$ be an idempotent size change graph and assume that the second condition does not hold. Since $\mu(\bar{x}, \bar{y})$ is a size change graph, $\mu(\bar{x}, \bar{y})$ must be satisfiable and there must be an alternating sequence of constraints implied by $\mu(\bar{x}, \bar{y})$ and $\bar{x} = \bar{y}$, forming a simple cycle, as depicted in Figure 3 and of the form: $x_{i_1} \succ^{b_1} y_{i_2} = x_{i_2} \succ^{b_2} y_{i_3} \ldots y_{i_k} = x_{i_k} \succ^{b_k} y_{i_1} = x_{i_1}$ with $b_1 \vee \cdots \vee b_k = 1$. From idempotence using Lemma 1 since the constraints $x_{i_1} \succ^{b_1} y_{i_2}$, $x_{i_2} \succ^{b_2} y_{i_3}$, $\ldots$, $x_{i_k} \succ^{b_k} y_{i_1}$ are implied by $\mu(\bar{x}, \bar{y})$ then so is the constraint $x_{i_1} > y_{i_1}$. Hence the first condition must hold.

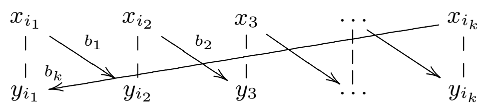


Fig. 3. An alternating cycle of inconsistent constraints from $\mu(\bar{x}, \bar{y})$ and $\bar{x} = \bar{y}$.

For the proof of Theorem 2, we know only that $\mu(\bar{x}, \bar{y})$ has some ranking function of an unknown form. This is sufficient to guarantee that the transition relation induced by $\mu(\bar{x}, \bar{y})$ has no infinite derivations.

Proof. (of Theorem 2) Assume $\mu(\bar{x}, \bar{y})$ has a ranking function. So $\mu(\bar{x}, \bar{y}) \wedge \bar{x} = \bar{y}$ is not satisfiable. Otherwise there would be a vector $\bar{a}$ such that $\bar{a} \sqsubseteq_\mu \bar{a}$ giving an infinite derivation. By Lemma 2, $\mu(\bar{x}, \bar{y}) \models x_i > y_i$ and so $f(u_1, \ldots, u_n) = u_i$ is a ranking function.

Theorems 1 and 2 provide the basis for a decision procedure [7]. For a set of size change graphs $\mathcal{G}$, first compute $\mathcal{G}^*$ and then check for each $\mu(\bar{x}, \bar{y}) \in \mathcal{G}^*$ if it is idempotent and if: $\bigvee_{1 \leq i \leq n} (\mu(\bar{x}, \bar{y}) \models x_i > y_i)$. We can strengthen this statement checking instead for each idempotent size graph the condition:

$$\mu(\bar{x}, \bar{y}) \models \bigvee_{1 \leq i \leq n} (x_i > y_i) \tag{1}$$

This is justified by the following result.

Lemma 3. *Let $\mu(\bar{x}, \bar{y})$ be an idempotent size change graph. Then*

$$\bigvee_{1 \leq i \leq n} (\mu(\bar{x}, \bar{y}) \models x_i > y_i) \quad \Leftrightarrow \quad \mu(\bar{x}, \bar{y}) \models \bigvee_{1 \leq i \leq n} (x_i > y_i).$$

The graph in Figure 2(a) demonstrates that this result does not hold for non-idempotent size change graphs.

Proof. (of Lemma 3) **(⇒)** Obvious. **(⇐)** If $\mu(\bar{x}, \bar{y}) \models \bigvee_i x_i > y_i$ then $\mu(\bar{x}, \bar{y}) \wedge \bigwedge_i (x_i = y_i)$ is not satisfiable. Hence, by Lemma 2, $\mu(\bar{x}, \bar{y})$ implies a constraint of the form $x_i > y_i$ as required.

It is interesting to note that the condition of Equation (1) is precisely that implemented in TermiLog and TerminWeb where no test for idempotence is applied. We proceed to generalize the completeness result of Theorem 2 to apply for size change graphs which are not necessarily idempotent. We show the condition of Equation (1) is complete for arbitrary size change graphs, idempotent or not. It means that we need not test for idempotence in an implementation.

Theorem 3 (Completeness – arbitrary size change graphs). *Let $\mu(\bar{x}, \bar{y})$ be a size change graph (not necessarily idempotent). If there exists any ranking function for $\mu(\bar{x}, \bar{y})$ then there exists one mapping to the non-negative integers of the form $f(u_1, \ldots, u_n) = \sum a_i u_i$ with all coefficients $a_i \in \{0, 1\}$.*

The proof follows from the observation that a set of size change graphs closed under composition is a finite semigroup with composition as the operator.

Lemma 4 (idempotent self composition). *Let $\mu(\bar{x}, \bar{y})$ be a monotonicity constraint. Then there exists a positive integer k such that $\mu^k(\bar{x}, \bar{y})$ is idempotent.*

Proof. A finite non-empty semigroup of the form $\{\ a^k \mid k \in \mathcal{Z}^+\ \}$ contains precisely one idempotent element [5].

Proof. (of Theorem 3) Let f mapping to $(D, \prec_D)$ be a ranking function for size change graph $\mu(\bar{x}, \bar{y})$. By Lemma 4, there exists a positive k such that $\mu^k(\bar{x}, \bar{y})$ is idempotent. By transitivity of $\prec_D$, f is also a ranking function for $\mu^k(\bar{x}, \bar{y})$. By Theorem 2, $\mu^k(\bar{x}, \bar{y})$ has a ranking function of the form $f'(u_1, \ldots, u_n) = u_i$ indicating the presence of a corresponding strict down arrow $\mu^k(\bar{x}, \bar{y}) \models x_i > y_i$. It follows that $\mu(\bar{x}^0, \bar{x}^1), \ldots, \mu(\bar{x}^{k-1}, \bar{x}^k)$ imply respectively constraints forming a chain of the form $x^0_{i_0} \succ^{b_1} x^1_{i_1} \succ^{b_2} x^2_{i_2} \succ \cdots \succ x^{k-1}_{i_{k-1}} \succ^{b_k} x^k_{i_k}$ with $i_0 = i_k = i$ such that at least one of $b_1, \ldots, b_k$ is 1 (i.e., strict). It follows that $\mu(\bar{x}, \bar{y})$ implies corresponding constraints $x_{i_0} \succ^{b_1} y_{i_1}$, $x_{i_1} \succ^{b_2} y_{i_2}$, …, $x_{i_{k-2}} \succ^{b_{k-1}} y_{i_{k-1}}$, $x_{i_{k-1}} \succ^{b_k} y_{i_k}$ with $i_0 = i_k = i$. Summing these constraints we get $x_{i_0} + \cdots + x_{i_{k-1}} > y_{i_0} + \cdots + y_{i_{k-1}}$.

We can assume without loss of generality that there are no repeated indices among $i_0, \ldots, i_{k-1}$. Hence we obtain the required result taking coefficients $a_{i_0}, \ldots, a_{i_{k-1}}$ equal to one and all others equal to zero.

If there were a repeated index $i_\ell = i_{\ell'} = i'$ then the sequence would be of the form $x^0_{i_0} \succ x^1_{i_1} \succ \cdots \succ x^\ell_{i_\ell} \succ \cdots \succ x^{\ell'}_{i_{\ell'}} \succ \cdots \succ x^k_{i_k}$. At least one of the shorter sequences: that starting and ending in argument position i without the segment from i' to i', or that starting and ending in argument position i' must contain a strict relation and can be chosen instead.

Theorem 3 does indicate an efficient test for termination and it would seem to require checking all possible combinations of coefficients $a_i \in \{0, 1\}$. We show that the, easy to implement, condition of Equation (1) is a complete test for non-idempotent graphs.

Corollary 1 (detecting ranking functions). *A size change graph $\mu(\bar{x}, \bar{y})$ has a ranking function if and only if*

$$\mu(\bar{x}, \bar{y}) \models \bigvee_{1 \le i \le n} x_i > y_i.$$

Proof. **(⇒)** Assume to the contrary that $\mu(\bar{x}, \bar{y})$ has a ranking function and $\mu(\bar{x}, \bar{y}) \wedge \neg \bigvee_i (x_i > y_i)$ has a solution. So $\mu(\bar{x}, \bar{y}) \wedge \bigwedge_i (x_i \le y_i)$ has a solution which implies that $\mu(\bar{x}, \bar{y}) \wedge (\Sigma_i a_i x_i \le \Sigma_i a_i y_i)$ has a solution for any coeeficients a_i. This is a contradiction because by Theorem 3, $\mu(\bar{x}, \bar{y}) \models \Sigma_i a_i x_i > \Sigma_i a_i y_i$ for some coefficients $a_i \in \{0, 1\}$. **(⇐)** If $\mu(\bar{x}, \bar{y}) \models \bigvee_i (x_i > y_i)$ then $\mu(\bar{x}, \bar{y}) \wedge \bar{x} = \bar{y}$ is not satisfiable and follow the proof of Theorem 2 up till the point when we get a simple cycle of constraints of the form depicted in Figure 3 (this part does not rely on idempotence). Summing these constraints gives a ranking function of the form $f(\bar{u}) = \Sigma_i a_i u_i$ with $a_i \in \{0, 1\}$.

4 Completeness for Monotonicity Constraints

Monotonicity constraints are more expressive than size change graphs. They may contain relations of the form $y_i \succ x_j$, going "up" in the graph representation, and also "horizontal" *loop invariants* of the form $x_i \succ x_j$ or of the form $y_i \succ y_j$. An analyzer based on size change graphs cannot prove termination when the size of an argument is increasing in a loop towards an upper bound. The following generic example illustrates that size change termination is incomplete for monotonicity constraints.

Example 5. *Consider a program involving a loop of the form* `while (a1 < a2) a1 := a1 + 1`*. A corresponding loop description involves a monotonicity constraint of the form* $\mu(\bar{x}, \bar{y}) = x_1 < x_2, x_1 < y_1, x_2 = y_2$ *which is idempotent but not a size change graph. While the loop clearly terminates, neither* $f(u_1, u_2) = u_1$ *nor* $f(u_1, u_2) = u_2$ *is a ranking function. There is however a ranking function of the form* $f(u_1, u_2) = u_2 - u_1$*.*

We provide a completeness result for monotonicity constraints. If $\mu(\bar{x}, \bar{y})$ is an idempotent monotonicity constraint and has a ranking function, then it has a ranking function of the form $f(u_1, \ldots, u_n) = u_i$ or of the form $f(u_1, \ldots, u_n) = u_i - u_j$ for $1 \leq i, j \leq n$. If $\mu(\bar{x}, \bar{y})$ is not idempotent, then it has a linear ranking function.

The extra expressiveness of monotonicity constraints introduces several problems. First, a monotonicity constraint $\mu(\bar{x}, \bar{y})$ or one of its derivation constraints may be unsatisfiable and hence have no infinite derivations. For example, $\mu(\bar{x}, \bar{y}) = x_1 > x_2 \wedge y_1 \leq y_2$ is satisfiable but $\mu(\bar{x}, \bar{z}, \bar{y})$ is not. A second problem is illustrated in Figure 4. The constraint in Figure 4(a) is idempotent and has no infinite derivations because the value in its first argument is strictly decreasing in any such derivation. However there is no direct down arc in $\mu(\bar{x}, \bar{y})$. If we restrict attention to constraints with *balanced* invariants we avoid both problems. This is not a limitation for termination analysis as every postcondition of a loop is the precondition for the next time around.

Definition 8 (balanced constraint). *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 $\mu_B(\bar{x}, \bar{y})$ *of* $\mu(\bar{x}, \bar{y})$ *is the smallest monotonicity constraint which includes* $\mu(\bar{x}, \bar{y})$ *and is balanced. We define* $bal(\mu)(\bar{x}, \bar{y})) = \mu(\bar{x}, \bar{y}) \wedge \{x_i \succ^b x_j \mid \mu(\bar{x}, \bar{y}) \models y_i \succ^b y_j\} \wedge \{y_i \succ^b y_j \mid \mu(\bar{x}, \bar{y}) \models x_i \succ^b x_j\}$*. Clearly* $\mu(\bar{x}, \bar{y})_B = bal^{4n(n-1)}(\mu)(\bar{x}, \bar{y}))$ *since there are at most* $4n(n-1)$ *constraints that can be added by* bal*. There are tighter bounds but that will suffice for our purposes.*

The balanced extension of the constraint in Figure 4(a) is shown in Figure 4(b) and with transitive closure in Figure 4(c). The downwards paths are now explicit. The balanced extension of a constraint is almost equivalent to the original, particularly in its powers. Figures 4(d) and (e) illustrate the similarity. For termination analysis we can restrict our attention to balanced extensions because of the following two Lemmata.

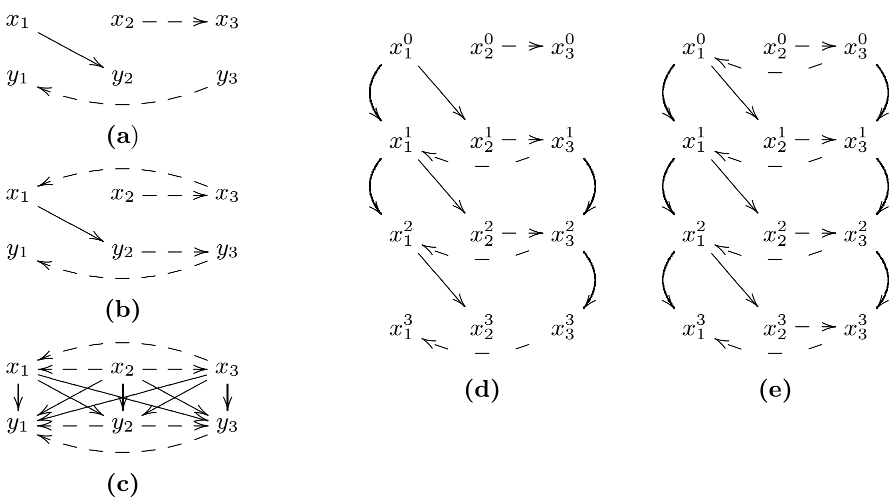


Fig. 4. (a) An unbalanced but idempotent constraint $\mu(\bar{x},\bar{y})$, (b) its balanced extension $\mu_B(\bar{x},\bar{y})$, and (c) $\mu_B(\bar{x},\bar{y})$ indicating also transitive paths, (d) derivation constraint $\mu(\bar{x}^0,\bar{x}^1,\bar{x}^3,\bar{x}^4)$, and (e) $\mu_B(\bar{x}^0,\bar{x}^1,\bar{x}^3,\bar{x}^4)$.

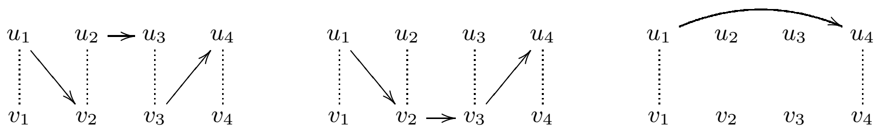
Lemma 5. *If monotonicity constraint $\mu(\bar{x},\bar{y})$ is balanced, then either the derivation constraint $\mu(\bar{x}^0,\dots,\bar{x}^m)$ is satisfiable for all $m > 0$ or $\mu(\bar{x},\bar{y})$ is unsatisfiable.*

Proof. *Assume that $\mu(\bar{x},\bar{y})$ is balanced and let $m > 0$ be such that $\mu(\bar{x}^0,\dots,\bar{x}^m)$ is not satisfiable. Hence for some variable x_i^k (at level k in argument i) there is a strict cycle of constraints implied by $\mu(\bar{x}^0,\dots,\bar{x}^m)$ of the form $x_i^k \succ x_j^{k'} \succ x_{j'}^{k'} \succ x_{i'}^k \succ x_i^k$ such that: (i) if $k = k'$ then $i \neq j$, $i = i'$ and $j = j'$; or (ii) $k' = k \pm 1$. Thus, as $\mu(\bar{x},\bar{y})$ is balanced it must also imply a strict cycle of the form $x_i \succ x_j \succ x_i$ (if $k = k'$) or of the form $x_i \succ y_j \succ y_{j'} \succ x_{i'} \succ x_i$ (if $k' = k+1$) or of the form $y_i \succ x_j \succ x_{j'} \succ y_{i'} \succ y_i$ (if $k' = k-1$). Hence $\mu(\bar{x},\bar{y})$ is not satisfiable.*

Lemma 6. *Monotonicity constraint $\mu(\bar{x},\bar{y})$ has an infinite derivation if and only if its balanced extension $\mu_B(\bar{x},\bar{y})$ has an infinite derivation.*

Proof. (sketch) **($\Leftarrow$)** *Let $\bar{b}^0 \sqsubseteq_{\mu_B} \bar{b}^1 \sqsubseteq_{\mu_B} \bar{b}^2 \sqsubseteq_{\mu_B} \cdots$ be an infinite derivation for the balanced extension. Since $\mu_B(\bar{x},\bar{y}) \models \mu(\bar{x},\bar{y})$, the infinite derivation $\bar{b}^0 \sqsubseteq_\mu \bar{b}^1 \sqsubseteq_\mu \bar{b}^2 \sqsubseteq_\mu \cdots$ exists.* **($\Rightarrow$)** *Let $\bar{a}^0 \sqsubseteq_\mu \cdots \bar{a}^k \sqsubseteq_\mu \bar{a}^{k+1} \sqsubseteq_\mu \cdots$ be an infinite derivation. One can show by induction that for any k and ℓ such that $0 \leq \ell \leq k$, $\bar{a}^k \sqsubseteq_{bal^\ell(\mu)} \bar{a}^{k+1}$. Now given that $\mu_B(\bar{x},\bar{y}) = bal^{4n(n-1)}(\mu(\bar{x},\bar{y}))$ we have the infinite derivation $\bar{a}^{4n(n-1)} \sqsubseteq_{\mu_B} \bar{a}^{4n(n-1)+1} \sqsubseteq_{\mu_B} a^{4n(n-1)+2} \sqsubseteq_{\mu_B} \cdots$.*

Theorem 4 (Completeness for idempotent monotonicity constraints). *Let $\mu(\bar{x},\bar{y})$ be a balanced idempotent monotonicity constraint. If there exists any ranking function for $\mu(\bar{x},\bar{y})$ then there exists one mapping to $(\mathcal{N},<)$ of the form $f(u_1,\dots,u_n) = u_i$ or of the form $f(u_1,\dots,u_n) = u_i - u_j$ for some $1 \leq i,j \leq n$.*



(a) **(b)** **(c)**

Fig. 5. Illustrating the proof of Lemma 7: **(a)** An alternating path from v_1 to v_4 with horizontal relation $u_2 > u_3$ **(b)** The graph is balanced so it contains also $v_2 > v_3$ **(c)** By transitivity it contains also $u_1 > u_4$ giving a shorter alternating path from v_1 to v_4.

The proof strategy for Theorem 4 is similar to that for Theorem 2. We will show that if there exists no ranking function for $\mu(\bar{x}, \bar{y})$ of the prescribed form then there is an infinite chain in $\sqsubseteq_\mu$ implying that there exists no ranking function of any form for $\mu(\bar{x}, \bar{y})$. We will need the following lemma.

Lemma 7. *If $\mu(\bar{x}, \bar{y})$ is a satisfiable balanced idempotent monotonicity constraint then either: (a) $\mu(\bar{x}, \bar{y})$ implies a constraint of the form $x_i > y_i$ or of the form $y_i > x_i$ or (b) $\mu(\bar{x}, \bar{y}) \wedge \bar{x} = \bar{y}$ is satisfiable.*

Proof. Let $\mu(\bar{x}, \bar{y})$ be a satisfiable balanced idempotent monotonicity constraint and assume that condition (b) does not hold. Given that $\mu(\bar{x}, \bar{y})$ is satisfiable and since $\mu(\bar{x}, \bar{y}) \wedge \bar{x} = \bar{y}$ is not satisfiable, it must be the case that there is a sequence of constraints from $\mu(\bar{x}, \bar{y})$ and from $\bar{x} = \bar{y}$ giving a contradiction. Without loss of generality, otherwise applying transitivity, we may assume the sequence is alternating and hence of the form:

$$u_{i_1} \succ^{b_1} v_{i_2} = u_{i_2} \succ^{b_2} v_{i_3} \ldots = u_{i_k} \succ^{b_k} v_{i_1} = u_{i_1}$$

with $b_1 \vee \cdots \vee b_k = 1$. Given that $\mu(\bar{x}, \bar{y})$ is balanced, we may also assume without loss of generality that the sequence does not involve "horizontal" relations as these could be removed by transitivity. See Figure 5. It follows that all of the constraints $u_{i_j} \succ^{b_1} v_{i_{j+1}}$ are in the same direction (downwards or upwards). Namely, that for all $1 \leq j \leq k$ either $u_{i_j} \in \bar{x}$ and $v_{i_j} \in \bar{y}$ or $u_{i_j} \in \bar{y}$ and $v_{i_j} \in \bar{x}$. From idempotence using Lemma 1 we get that $\mu(\bar{x}, \bar{y})$ contains a constraint of the form $x_i > y_i$ or of the form $y_i > x_i$.

Lemma 8. *If $\mu(\bar{x}, \bar{y})$ is a balanced idempotent monotonicity constraint where for all i $\mu(\bar{x}, \bar{y}) \not\models x_i \succ y_i$, then for all i, j, $\mu(\bar{x}, \bar{y}) \not\models x_i \succ y_j$.*

Proof. Assume to the contrary that $\mu(\bar{x}, \bar{y}) \models x_i \succ y_j$. Then since $\mu(\bar{x}, \bar{y})$ is idempotent and balanced there must exist two constraints $x_i \succ y_{k_1}$ and $x_{k_1} \succ y_j$ implied by $\mu(\bar{x}, \bar{y})$ to ensure that $x_i \succ y_j$ is in the self composition. If $k_1 \in \{i, j\}$ we have a contradiction. So $k_1 \notin \{i, j\}$. Now consider the constraint $x_i \succ y_{k_1}$ implied by $\mu(\bar{x}, \bar{y})$. Using the same reasoning there must be constraints $x_i \succ y_{k_2}$ and $x_{k_2} \succ y_{k_1}$ implied by $\mu(\bar{x}, \bar{y})$. If $k_2 = i$ or $k_2 = k_1$ we immediately have a contradiction. If $k_2 = j$, then we have $x_j = x_{k_2} \succ y_{k_1}$ and $x_{k_1} \succ y_j$ implied by

$\mu(\bar{x}, \bar{y})$ and hence by Lemma 1 also $x_j \succ y_j$ implied by $\mu(\bar{x}, \bar{y})$. Contradiction. Hence $k_2 \notin \{i, j, k_1\}$. We can now consider the constraint $x_i \succ y_{k_2}$ to generate $x_i \succ y_{k_3}$ and $x_{k_3} \succ y_{k_2}$, where $k_3 \notin \{i, j, k_1, k_2\}$. Following the same reasoning we eventually run out of argument positions. Contradiction.

Lemma 9. *For satisfiable, balanced and idempotent monotonicity constraint $\mu(\bar{x}, \bar{y})$, if for all $1 \leq i, j \leq n$, $\mu(\bar{x}, \bar{y}) \not\models x_i > y_i$ and $\mu(\bar{x}, \bar{y}) \not\models x_i \geq y_i \wedge x_i \geq x_j \wedge y_j > x_j$ then there is an infinite derivation using $\mu(\bar{x}, \bar{y})$.*

Proof. (Sketch)
Construction: *Let μ be the set of binary relations of the form $u \succ v$ implied by $\mu(\bar{x}, \bar{y})$. Let $U \subseteq \{1, \ldots, n\}$ be the set of arguments j which have a strict up arrow $y_j > x_j \in \mu$ and arguments i such that $j \in U$ and $u_i \succ v_j \in \mu$. Let $E = \{1, \ldots, n\} - U$ be the rest of the arguments. Let $V_E = \cup\{\{x_i, y_i\} \mid i \in E\}$ and $V_U = \cup\{\{x_i, y_i\} \mid i \in U\}$. We partition μ into three disjoint sets (conjunctions) of constraints: μ_U — the restriction of μ to the arguments U, μ_E — its restriction to the arguments in E, and μ_{EU} — the rest. So, $\mu(\bar{x}, \bar{y}) = \mu_U \wedge \mu_E \wedge \mu_{EU}$. This partitioning is given by: $\mu_E = \{u \succ v \in \mu \mid \{u, v\} \subseteq V_E\}$, $\mu_U = \{u \succ v \in \mu \mid \{u, v\} \subseteq V_U\}$ and $\mu_{EU} = \mu - \mu_E - \mu_U$.*

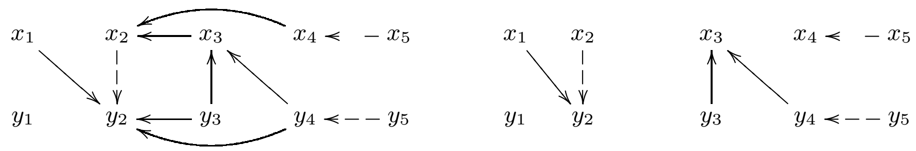
The "equals" part: *First we show that $\mu_E \wedge \bigwedge_{i \in E} x_i = y_i$ is satisfiable. By Lemma 7 either this holds or there exists $x_i > y_i$ in μ contradicting the assumption of the Lemma or $y_i > x_i$ in μ for $i \in E$ contradicting the definition of E. Hence there is a solution $\bar{a}_E$ of $\mu_E(\bar{x}, \bar{y}) \wedge \bigwedge_{i \in E} x_i = y_i$ and so $\bar{a}_E \sqsubseteq_{\mu_E} \bar{a}_E$.*

The "up" part: *Now let us consider μ_U. From the assumption of the Lemma there can be no $i \in U$ with $x_i > y_i \in \mu_U$. We show by the construction and the preconditions that there is no $i \in U$ with $x_i \geq y_i \in \mu_U$.*

First we show that for each $i \in U$ either $y_i > x_i$ or there exists k where $y_k > x_k$ and $x_i \succ x_k$ or $y_i \succ x_k$. The first case is straightforward from the definition of U. For the second, suppose i is added to U because j is already in U. Then either (a) $x_i \succ x_j$ (and $y_i \succ y_j$), (b) $x_i \succ y_j$ or (c) $y_i \succ x_j$. If $y_j > x_j$ then in all three cases we get the result. Otherwise by induction, we have $x_j \succ x_k$ or $y_j \succ x_k$. For case (a) if $x_j \succ x_k$ then we have by transitivity $x_i \succ x_k$ or if $y_j \succ x_k$ we have $y_i \succ x_k$. For case (b) if $x_j \succ x_k$ then by balance we have that $y_j \succ y_k$ and by transitivity ($x_i \succ y_j, y_j \succ y_k, y_k > x_k$) we have $x_i \succ x_k$, or if $y_j \succ x_k$ then by transitivity we have $x_i \succ x_k$. For (c) if $x_j \succ x_k$ then by transitivity we have $y_i \succ x_k$, and if $y_j \succ x_k$ we have by idempotence that $y_i \succ x_k$.

Suppose that $x_i \geq y_i$ for some $i \in U$. Then for some j with $y_j > x_j$ we have either $x_i \succ x_j$ or $y_i \succ x_j$. In the first case this contradicts the preconditions of the lemma. In the second case since $x_i \geq y_i$ and $y_i \succ x_j$ by transitivity we have $x_i \succ x_j$ again contradicting the preconditions of the lemma. So we have that there are no directly down arcs in U. By Lemma 8 we have there are no downwards arcs (direct or indirect) amongst arguments in U (E cannot be involved since there are no arcs from arguments in E to arguments in U).

Ordering the "up" part: *We now partition U into (disjoint) sets $U = U_1 \cup \cdots \cup U_l$ where for each $1 \leq k \leq l$ and $\{i, j\} \subseteq U_k$ we have $\mu_U \models x_i = x_j$ and for*



(a) **(b)** **(c)**

Fig. 6. The monotonicity constraints from Example 6: **(a)** the monotonicity constraint $\mu(\bar{x}, \bar{y})$, **(b)** the "equals" part μ_E, and **(c)** the "up" part μ_U.

each $1 \leq k_1 < k_2 \leq l$, $i_1 \in U_{k_1}$ *and* $i_2 \in U_{k_2}$ *we do not have* $\mu_U \models x_{i_1} \succ x_{i_2}$. *This is possible since* μ *is satisfiable. This provides a total order on equivalence classes of arguments such that variables in* U_k *are not constrained from above by any arguments in* E *or* $U_1 \cup \cdots \cup U_{k-1}$.

An infinite derivation: *We can now build an infinite derivation for* μ. *To build* $\bar{a}^0$ *set each position in* E *to the value in* $\bar{a}_E$. *Then for* U_1 *set all positions* $j_1 \in U_1$ *to the least integer satisfying all the constraints with respect to arguments in* E. *For* $k = 2, \ldots l$ *set all positions* j_k *to the least integer satisfying all constraints with respect to arguments in* $E \cup U_1 \cup \cdots \cup U_{k-1}$.

To build $\bar{a}^{k+1}$ *from* a^k *is similar but also taking into account all (lower bounding) constraints with respect to* $\bar{a}^k$.

Example 6. *Consider the satisfiable balanced idempotent monotonicity constraint depicted in Figure 6***(a)**. *Building* $U = \{3, 4, 5\}$*:* $3 \in U$ *because of the constraint* $y_3 > x_3$, $4 \in U$ *because of 3 and the constraint* $y_4 \geq x_3$ *and* $5 \in U$ *because of 4 and the constraint* $x_5 \geq x_4$. *The remaining indices are* $E = \{1, 2\}$. *Partitioning* U*: we can take* $U_1 = \{3\}$, $U_2 = \{4\}$ *and* $U_3 = \{5\}$ *(*U_3 *must be last in the ordering). The constraints* μ_E *and* μ_U *are depicted as Figures 6***(b)** *and* **(c)**. *The constraint* $\mu_{EU} = \{u > v \mid u \in \{x_3, x_4, x_5, y_3, y_4, y_5\}, v \in \{x_2, y_2\}\}$

We build an infinite derivation as follows. Pick a solution for $\mu_E \wedge x_1 = y_1 \wedge x_2 = y_2$, *say* $x_1 = y_1 = 1$, $x_2 = y_2 = 0$. $\bar{a}_E = (1, 0)$. *Create* $\bar{a}_0$ *starting from* $\bar{a}_E$*, and filling in the argument positions in* U_1*,* U_2*, and* U_3 *with the least value satisfying constraints in filled in positions. Since there are no arcs from an argument position to an unfilled position this is always possible. We find* $\bar{a}_0 = (1, 0, 1, 1, 1)$, $\bar{a}_1 = (1, 0, 2, 2, 2)$, $\bar{a}_2 = (1, 0, 3, 3, 3)$,...

Proof. (of Theorem 4) If $\mu(\bar{x}, \bar{y})$ is unsatisfiable then any ranking function is suitable (in particular one the form required by the theorem). Otherwise the conditions of Lemma 9 hold and since there exists a ranking function there can be no infinite derivation. Hence either (a) for some i, $\mu(\bar{x}, \bar{y}) \models x_i > y_i$ and the function $f(\bar{u}) = u_i$ is thus a ranking function, or (b) for some i, j, $\mu(\bar{x}, \bar{y}) \models x_i \geq y_i \wedge x_i \geq x_j \wedge y_j > x_j$ from which it follows that $f(\bar{u}) = u_i - u_j$ is a ranking function.

Theorem 5 (Completeness for arbitrary monotonicity constraints). *Let* $\mu(\bar{x}, \bar{y})$ *be a balanced monotonicity constraint (not necessarily idempotent). If* *there exists any ranking function for $\mu(\bar{x},\bar{y})$ then there exists a linear ranking function for $\mu(\bar{x},\bar{y})$.*

Proof. Assume balanced $\mu(\bar{x},\bar{y})$ has a ranking function. Assume $\mu(\bar{x},\bar{y})$ is satisfiable otherwise the result is trivial. It follows that $\mu^k(\bar{x},\bar{y})$ is satisfiable, balanced and has a ranking function for any positive k. By Lemma 4 there exists a k such that $\mu^k(\bar{x},\bar{y})$ is idempotent and hence by Theorem 4, $\mu^k(\bar{x},\bar{y})$ has a ranking function of the form $f(u_1,\ldots,u_n) = u_i$ or of the form $f(u_1,\ldots,u_n) = u_i - u_j$. If the first case then the proof is identical to that of Theorem 3. If the second case then $\mu^k(\bar{x},\bar{y}) \models x_i \geq y_i \wedge x_i \geq x_j \wedge y_j > x_j$ and similar to the proof of Theorem 3 the following two sequences of "down" and "up" constraints are implied by $\mu(\bar{x},\bar{y})$:

$$\begin{array}{llll} x_{i_0} \geq y_{i_1}, & x_{i_1} \geq y_{i_2}, & \ldots, & x_{i_{k-2}} \geq y_{i_{k-1}}, \quad x_{i_{k-1}} \geq y_{i_k} \\ y_{j_1} \succ x_{j_0}, & y_{j_2} \succ x_{j_1}, & \ldots, & y_{j_{k-1}} \succ x_{j_{k-2}}, \quad y_{j_k} \succ x_{j_{k-1}} \end{array}$$

where at least one of the inequalities in the second ("up") sequence is strict. Adding these inequalities pairwise we get the sequence:

$$x_{i_0} - x_{j_0} \succ y_{i_1} - y_{j_1}, \quad x_{i_1} - x_{j_1} \succ y_{i_2} - y_{j_2}, \quad \ldots, \quad x_{i_{k-1}} - x_{j_{k-1}} \succ y_{i_k} - y_{j_k}$$

with at least one strict inequality. Summing this sequence and observing that $i_0 = i_k = i$ and $j_0 = j_k = j$ we obtain $\Sigma_{\ell=1}^k (x_{i_\ell} - x_{j_\ell}) > \Sigma_{\ell=1}^k (y_{i_\ell} - y_{j_\ell})$ which is of the form $\Sigma a_i x_i > \Sigma a_i y_i$ with coefficients determined by the number of repetitions of the constraints in the two sequences. Positive coefficients originate from "downwards" constraints and negative coefficients from the "upwards". We take $f(\bar{u}) = \Sigma a_i u_i$.

We now show that $\mu(\bar{x},\bar{y}) \models f(\bar{x}) \geq 0$. We have $\mu^k(\bar{x},\bar{y}) \models x_i \geq x_j$ which implies that $\mu(\bar{x},\bar{y}) \models x_i \geq x_j$ and from balance $\mu(\bar{x},\bar{y}) \models y_i \geq y_j$. Recalling that $i = i_k$ and $j = j_k$ we have $\mu(\bar{x},\bar{y}) \models y_{i_k} \geq y_{j_k}$ and $\mu(\bar{x},\bar{y}) \models x_{i_k} \geq x_{j_k}$. From transitivity (with the last constraints in the "down" and "up" sequences) that $\mu(\bar{x},\bar{y}) \models x_{i_{k-1}} \geq x_{j_{k-1}}$ and from balance $\mu(\bar{x},\bar{y}) \models y_{i_{k-1}} \geq y_{j_{k-1}}$. In a similar way we obtain that $\mu(\bar{x},\bar{y})$ implies the constraints $x_{i_\ell} \geq x_{j_\ell}$ for $\ell \in \{1,\ldots,k\}$. Summing these constraints gives $f(\bar{x}) = \Sigma_{\ell=1}^k (x_{i_\ell} - x_{j_\ell}) \geq 0$.

Example 7. *Consider the (balanced extension of) monotonicity constraint $x_1 \geq y_2, x_2 \geq y_1, x_2 \geq x_3, y_3 > x_3$. The "down" and "up" sequences from the proof of Theorem 5 are respectively $x_1 \geq y_2, x_2 \geq y_1$ and $y_3 < x_3, y_3 < x_3$. Summing these gives $x_1 + x_2 - 2x_3 > y_1 + y_2 - 2y_3$. A ranking function of the form $f(\bar{u}) = u_1 + u_2 - 2u_3$ exists. The constraints $x_1 \geq x_3$ and $x_2 \geq x_3$ imply that $f(\bar{x}) = x_1 + x_2 - 2x_3 \geq 0$.*

5 Conclusion

This paper makes two contributions. For size change graphs we establish that the termination test implemented in analyzers such as TermiLog and TerminWeb is complete for size change graphs and incomplete for monotonicity constraints. In particular there is no loss of precision when not checking for idempotence. For idempotent monotonicity constraints, we prove that if there exists any ranking function for a loop description then there exists one of a simple form: a single argument or the difference between two arguments is decreasing. Moreover, for loop descriptions which are not idempotent if there exists a ranking function then there exists one which is linear.

References

1. A. Brodsky and Y. Sagiv. Inference of monotonicity constraints in Datalog programs. In *Proceedings of the Eighth ACM SIGACT-SIGART-SIGMOD Symposium on Principles of Database Systems*, pages 190–199, 1989.
2. M. Codish and C. Taboch. A semantic basis for the termination analysis of logic programs. *The Journal of Logic Programming*, 41(1):103–123, 1999.
3. N. Dershowitz, N. Lindenstrauss, Y. Sagiv, and A. Serebrenik. A general framework for automatic termination analysis of logic programs. *Applicable Algebra in Engineering, Communication and Computing*, 12(1-2):117–156, 2001.
4. C. C. Frederiksen. A simple implementation of the size-change termination principle. Technical Report D-442, DIKU, Copenhagen University, Denmark, 2001.
5. G. Frobenius. Über endliche gruppen. *Sitzungsber. Preuss. Akad. Wiss. Berlin*, pages 163–194, 1895.
6. M. Gabbrielli and R. Giacobazzi. Goal independency and call patterns in the analysis of logic programs. In *Proceedings of the Ninth ACM Symposium on Applied Computing*, pages 394–399. ACM Press, 1994.
7. C. S. Lee, N. D. Jones, and A. M. Ben-Amram. The size-change principle for program termination. *ACM SIGPLAN Notices*, 36(3):81–92, 2001. Proceedings of POPL'01.
8. N. Lindenstrauss and Y. Sagiv. Automatic termination analysis of logic programs. In L. Naish, editor, *Proceedings of the Fourteenth International Conference on Logic Programming*, pages 63–77, Leuven, Belgium, 1997. The MIT Press.
9. N. Lindenstrauss, Y. Sagiv, and A. Serebrenik. Termilog: A system for checking termination of queries to logic programs. In O. Grumberg, editor, *CAV*, volume 1254 of *Lecture Notes in Computer Science*, pages 444–447. Springer, 1997.
10. A. Podelski and A. Rybalchenko. A complete method for the synthesis of linear ranking functions. In B. Steffen and G. Levi, editors, *Verification, Model Checking, and Abstract Interpretation, 5th International Conference*, volume 2937 of *Lecture Notes in Computer Science*, pages 239–251. Springer, 2004.
11. F. Ramsey. On a problem of formal logic. In *Proceedings London Mathematical Society*, volume 30, pages 264–286, 1930.
12. R. Thiemann and J. Giesl. Size-change termination for term rewriting. In *Proceedings of the 14th International Conference on Rewriting Techniques and Applications (RTA-03)*, volume 2706 of *Lecture Notes in Computer Science*, pages 264–278. Springer-Verlag, 2003.
13. D. Wahlstedt. Detecting termination using size-change in parameter values. Master's thesis, Göteborgs Universitet, 2000. `http://www.cs.chalmers.se/~davidw/`.