Advanced Topics in Termination

3rd International School on Rewriting
Obergurgl, Austria, 21-23 July 2008

Dieter Hofbauer
BA Nordhessen
Germany

Rewriting

Why study rewriting? Well . . .

- oriented equations
- universal computation model
- model for non-deterministic processes

Specific classes of rewriting systems:
string / term / higher-order / graph / . . .

String Rewriting

Why study string rewriting?

- oriented equations $\rightsquigarrow$ (semi-) group theory
- universal computation model $\rightsquigarrow$ recursion / complexity theory
- particular case of linear term rewriting (why?)
- prototype for more general rewriting systems:
  - concepts easier to invent
  - concepts easier to explain
  - concepts often generalize (to linear rewriting ...)
  - undecidability results transfer

String Rewriting: Definitions

- Letter: element of a set Σ, the alphabet
- String: sequence of letters. Σ^* is the set of strings over Σ
- String rewriting system: set of rules of the form $\ell \to r$, i.e. a set $R \subseteq \Sigma^* \times \Sigma^*$
- Rewrite step: replace the left hand side of rule $\ell \to r$ by its right hand side: $x\ell y \to_R xry$ within context $x, y \in \Sigma^*$
- Derivation: chain of rewrite steps

Term Rewriting: Definitions

- Symbol: element of a set Σ, the signature
- Term: tree.
  $\mathcal{T}_\Sigma$ is the set of ground terms over Σ,
  $\mathcal{T}_\Sigma(\mathcal{V})$ is the set of terms with variables from $\mathcal{V}$
- Term rewriting system: set of rules of the form $\ell \longrightarrow r$,
  i.e. a set $R \subseteq \mathcal{T}_\Sigma(\mathcal{V}) \times \mathcal{T}_\Sigma(\mathcal{V})$
- Rewrite step: replace the left hand side of rule $\ell \longrightarrow r$ by its right hand side:
  $c[\ell\sigma] \longrightarrow_R c[r\sigma]$ within context c under substitution σ
- Derivation: chain of rewrite steps

Termination

Why study termination? Well ...

System R is *terminating*
if any R-derivation contains only finitely many steps.

- Notation $\mathrm{SN}(R)$: R is *strongly normalizing*
- That is, $\to_R^+$ is well-founded.

Expl.s of terminating (why?) systems:

- $\{aab \to ba\}$
- $\{ab \to ba\}$
- $\{ab \to baa\}$
- $\{aa \to aba\}$

Example

$R = \{aa \to bc,\ bb \to ac,\ cc \to ab\}$ induces derivation

$$b\,b\,\boxed{a\,a} \to_R$$
$$b\,\boxed{b\,b}\,c \to_R$$
$$b\,a\,\boxed{c\,c} \to_R$$
$$b\,\boxed{a\,a}\,b \to_R$$
$$\boxed{b\,b}\,c\,b \to_R$$
$$a\,\boxed{c\,c}\,b \to_R$$
$$a\,a\,b\,b \to_R \cdots\cdots\cdots$$

- Is there an infinite derivation?

Example

$R = \{aa \to bc,\ bb \to ac,\ cc \to ab\}$ induces derivation

$$b\,b\,\boxed{a\,a} \to_R$$
$$b\,\boxed{b\,b}\,c \to_R$$
$$b\,a\,\boxed{c\,c} \to_R$$
$$b\,\boxed{a\,a}\,b \to_R$$
$$\boxed{b\,b}\,c\,b \to_R$$
$$a\,\boxed{c\,c}\,b \to_R$$
$$a\,a\,b\,b \to_R \cdots\cdots\cdots\cdots\cdots$$

- Is there an infinite derivation?
  No (was open for some time)
- How long can derivations get?

Example

$R = \{aa \to bc,\ bb \to ac,\ cc \to ab\}$ induces derivation

$$
\begin{array}{l}
b\, b\, \boxed{a\, a} \to_R \\
b\, \boxed{b\, b}\, c \to_R \\
b\, a\, \boxed{c\, c} \to_R \\
b\, \boxed{a\, a}\, b \to_R \\
\boxed{b\, b}\, c\, b \to_R \\
a\, \boxed{c\, c}\, b \to_R \\
a\, a\, b\, b \to_R \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots
\end{array}
$$

- Is there an infinite derivation?
  No (was open for some time)
- How long can derivations get?
  Exponential bound in size of starting string (trivial)
  Open problem: polynomial upper bound?

Derivational Complexity: Definition

The *derivation height* of term t modulo system R is the maximal length of an R-derivation starting in t:

$$\boxed{\mathrm{dh}_R(t) = \max\{n \mid \exists s : t \to_R^n s\}}$$

The *derivational complexity* of R maps natural number n to the maximal derivation height of terms of size at most n:

$$\boxed{\mathrm{dc}_R(n) = \max\{\mathrm{dh}_R(t) \mid \mathrm{size}(t) \leq n\}}$$

This is a worst case complexity measure.

Exercise: How about the following systems?

- $\{aab \to ba\}$, $\{ab \to ba\}$, $\{ab \to baa\}$, $\{aa \to aba\}$

Derivational Complexity: Exercises

Find lower bounds for the derivational complexity of

- $R_1 = \{ba \to acb,\, bc \to abb\}$
- $R_2 = \{ba \to acb,\, bc \to cbb\}$
- $R_3 = \{ba \to aab,\, bc \to cbb\}$

Hint: one system is doubly exponential, one is multiply exponential, one is non-terminating.

A lower bound is proven by presenting a family of derivations that achieves the desired length.

Relative Termination

allows to remove rules successively $\rightsquigarrow$
modular termination proofs

System R is *terminating relative to* system S
if any $R \cup S$-derivation contains only finitely many R-steps.

- Notation: $\mathrm{SN}(R/S)$
- That is, $(\to_S^* \circ \to_R \circ \to_S^*)^+$ is well-founded

Expl: $\{aa \to aba\}$ is terminating relative to $\{b \to bb\}$.

$\mathrm{SN}(R/S)$ and $\mathrm{SN}(S)$ imply $\mathrm{SN}(R \cup S)$

Course Outline

- Termination proofs
  - direct / incremental / transformations
- Match bounds
  - automata / regularity preservation
- Matrix interpretations
  - heuristics / weighted automata
- Derivational complexity
  - interpretations / context-dependent int's
  - path orders
  - relative termination
- Miscellaneous
  - competition
  - live demos

www.termination-portal.org

- people
- workshop on termination (1st WST'93 – 9th WST'07)
- termination competition ('04 – '07)
- tools, e.g.
  - AProVE [Giesl et al.]
  - Jambox [Endrullis]
  - Matchbox [Waldmann]
  - MultumNonMulta [Hofbauer]
  - Torpa [Zantema]
  - TTT(2) [Middeldorp et al.]
- problems
  termination problem data base (tpdb) at
  `www.lri.fr/~marche/termination-competition/`

Termination via Interpretations

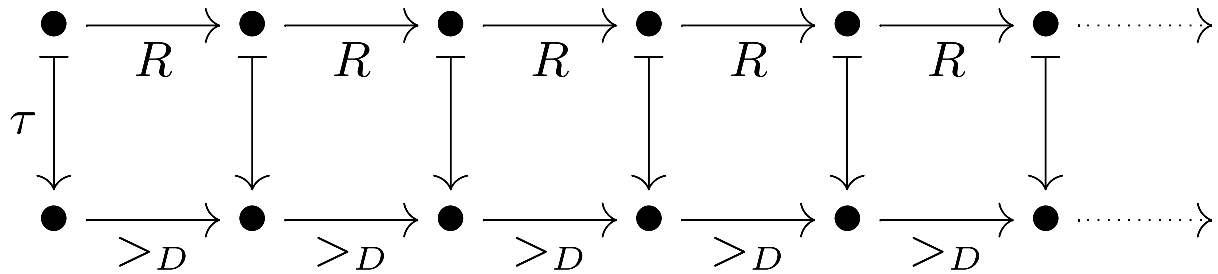
Interpretations as order preserving mappings
into well-founded domains:

- Let R be a rewriting system over Σ.
- Let $(D, \geq_D)$ be a well-founded partial order.

If a mapping $\tau : \mathcal{T}_\Sigma \to D$ is order preserving (monotone)

- from $(\mathcal{T}_\Sigma, {\to_R}^+)$ to $(D, >_D)$

then R is terminating.



Relative Termination

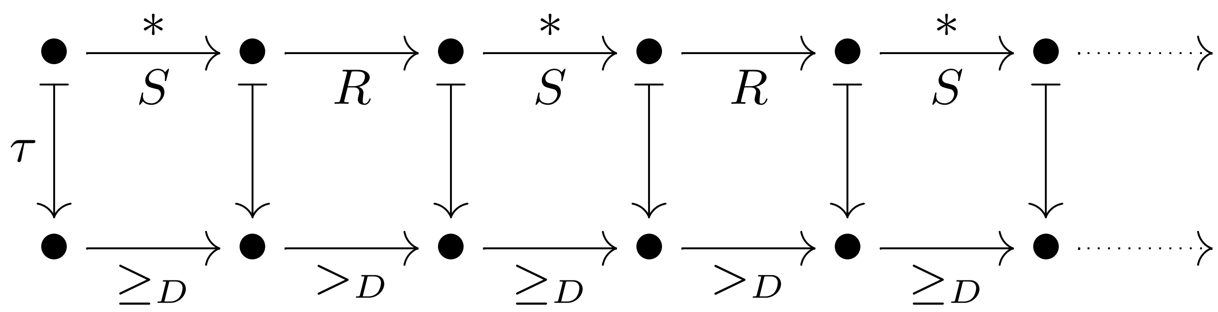
Straightforward generalization to relative termination:

- Let R and S be rewriting systems over Σ.
- Let $(D, \geq_D)$ be a well-founded partial order.

If a mapping $\tau : \mathcal{T}_\Sigma \to D$ is order preserving

- from $(\mathcal{T}_\Sigma, \to_R{}^+)$ to $(D, >_D)$ and
- from $(\mathcal{T}_\Sigma, \to_S{}^+)$ to $(D, \geq_D)$,

then R is terminating relative to S.



Interpretations (cont'd)

R is terminating iff there is a well-founded ordering $>$ on $\mathcal{T}_\Sigma$ such that, for all $t, t' \in \mathcal{T}_\Sigma$,

$$t \rightarrow_R t' \quad \text{implies} \quad t > t'$$

(Exercise: show "only if".)

- For *interpretations* choose $>$ as an ordering *induced by a function* $\tau : \mathcal{T}_\Sigma \rightarrow D$ as above:

τ is an *interpretation for* R *into* $(D, \geq_D)$ if, for all $t, t' \in \mathcal{T}_\Sigma$,

$$t \rightarrow_R t' \quad \text{implies} \quad \tau(t) >_D \tau(t')$$

- Are interpretations a "*universal*" proof method, i.e., do they apply to *all* terminating rewriting systems?
- In which cases can D be specialized to $\mathbb{N}$?

Interpretations (cont'd)

- Are interpretations a *"universal"* proof method, i.e., do they apply to *all* terminating rewriting systems?

  Yes: Let $D = \mathcal{T}_\Sigma$, $>_D = \to_R^+$, τ the identity on $\mathcal{T}_\Sigma$. R is terminating if and only if an interpretation for R into some well-founded partial ordering exists.

- In which cases can D be specialized to $\mathbb{N}$?

  For finitely branching terminating systems: Let $\tau = \mathrm{dh}_R$. (Note that dh_R is well-defined for finitely branching R.) R is terminating if and only if an interpretation for R into $(\mathbb{N}, \geq)$ exists.

  Exercise: show that no interpretation for $\{a \to f^i(b) \mid i \in \mathbb{N}\} \cup \{f(b) \to b\}$ into $(\mathbb{N}, \geq)$ exists.

Homomorphic Interpretations

Each function symbol f is associated with a function f_τ of same arity on the underlying well-founded set (Σ-*algebra*). Ground terms are interpreted via *homomorphic extension*:

$$\tau\big(f(t_1,\ldots,t_n)\big) = f_\tau\big(\tau(t_1),\ldots,\tau(t_n)\big)$$

- Expl.: A homomorphic interpretation for $\{ffx \to fgfx\}$ over $\Sigma = \{a, f, g\}$ into $(\mathbb{N}, \geq)$: Choose $a_\tau = 1$ and

$$f_\tau = \begin{cases} n+2 & \text{if } n \text{ is even} \\ n-1 & \text{else} \end{cases} \qquad g_\tau = \begin{cases} n+1 & \text{if } n \text{ is even} \\ n & \text{else} \end{cases}$$

Hint: show $\tau(t) = 2k$ if $t = f(\ldots)$, else $\tau(t) = 2k+1$, where k is the number of factors ff in t.

Homomorphic Int's (cont'd)

- Again, homomorphic interpretations are *"universal"*.
  Let $D = \mathcal{T}_\Sigma$, $>_D = \rightarrow_R^+$ as before.
  Choose $f_\tau = f$, thus $\tau(t) = t$.

- A simple algebraic characterization:
  $\tau : \mathcal{T}_\Sigma \rightarrow D$ is a Σ-homomorphism iff
  $\tau(t_i) = \tau(t_i')$ implies $\tau\big(f(t_1, \ldots, t_n)\big) = \tau\big(f(t_1', \ldots, t_n')\big)$.
  - E.g., all *injective* interpretations can be expressed as homomorphic ones.
  - But derivation height functions dh_R of terminating systems R typically not (why?). Nevertheless:

Homomorphic Int's (cont'd)

- A finitely branching system R is terminating if and only if a homomorphic interpretation for R into $(\mathbb{N}, \geq)$ exists.
  - Proof: exercise
  - Hint: define an appropriate bijection between $\mathcal{T}_\Sigma$ and $\mathbb{N}$ that respects $\rightarrow_R$.
  - Remark: this even gives *recursive* functions f_τ in case sets $\rightarrow_R(t)$ can be computed, thus in particular for *finite* systems R.

Monotone Interpretations

Using *strictly monotone* functions f_τ ensures that it suffices to consider (ground) instances $\ell\gamma \to r\gamma$ of rewrite rules $\ell \to r$ within homomorphic interpretations into $(D, \geq_D)$:

$$d >_D d' \quad \text{implies} \quad f_\tau(\dots, d, \dots) >_D f_\tau(\dots, d', \dots)$$

Such an interpretation is called *monotone*. Then:

- $t \to_R t'$ implies $\tau(t) >_D \tau(t')$ if and only if, for all rules $\ell \to r$ and ground substitutions γ, $\tau(\ell\gamma) >_D \tau(r\gamma)$.
- Thus, R is terminating if $\tau(\ell\gamma) >_D \tau(r\gamma)$ for all rules $\ell \to r$ and ground substitutions γ.

Monotone Interpretations (cont'd)

- Again, monotone interpretations are *"universal"*.
- But unlike homomorphic interpretations in general, for monotone interpretations the restriction to $(\mathbb{N}, \geq)$ is no longer universal: An interpretation into a totally ordered domain induces a total ordering on ground terms. But for

$$\{g(a) \to g(b),\ f(b) \to f(a)\}$$

this is impossible (why?).

Challenging Problems

- z086 [Zantema]

$$\{aa \to bc,\ bb \to ac,\ cc \to ab\}$$

- z001 [Zantema]

$$\{aabb \to bbbaaa\}$$

Automata theory can help ...

Preserving Regularity

Given: A *string rewriting system* R over alphabet Σ.
The set of *descendants* of a language $L \subseteq \Sigma^*$ modulo R is

$$\to_R^*(L) = \{y \in \Sigma^* \mid \exists x \in L : x \to_R^* y\} = R^*(L)$$

R *preserves regularity*: If L is regular then $\to_R^*(L)$ is regular.

R *preserves context-freeness*: analogously

- Aiming at *syntactic criteria* guaranteeing regularity preservation – despite known undecidability results.

Example [Book, Jantzen, Wrathall 1982]:
Inverse context-free rules: $|\text{right hand side}| \leq 1$.
Monadic rules: Inverse context-free and length-reducing.

Deleting String Rewriting Systems

The system R is *deleting* if there is a *precedence* (irreflexive partial order) $>$ on Σ so that for each rule $\ell \to r$ in R:

$$\boxed{\forall \text{ letter } b \text{ in } r \; \exists \text{ letter } a \text{ in } \ell : a > b}$$

Hibbard (1974) calls the inverse system
$R^- = \{r \to \ell \mid \ell \to r \text{ in } R\}$ *context-limited*.

- Deleting systems preserve regularity.
  [H, Waldmann 2003]

- Inverse deleting systems preserve context-freeness.
  [Hibbard 1974]

Example

$$\boxed{R = \{ba \to cb, \;\; bd \to d, \;\; cd \to de, \;\; d \to \epsilon\}}$$

is deleting for the precedence

$$\boxed{a > b > d, \;\; a > c > e, \;\; c > d}$$

For instance,

$$\to_R^*(ba^*d) \cap \mathrm{NF}(R) = c^*b \cup c^*e^*$$

where $\mathrm{NF}(R)$ denotes the set of R-normal forms.

A Decomposition Theorem

For each deleting system R over Σ there are

- a finite substitution s from Σ to some alphabet $\Gamma \supseteq \Sigma$,
- an inverse context-free ($\approx$ monadic) system M over Γ

so that

$$\boxed{\rightarrow_R^* = (s \circ \rightarrow_M^*)|_\Sigma}$$

A Decomposition Theorem

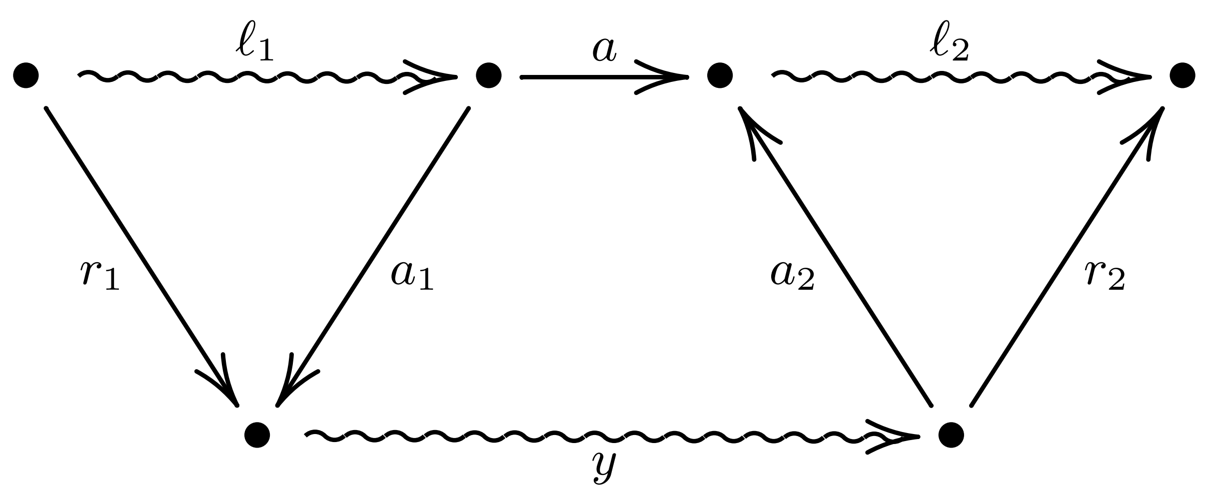
For each deleting system R over Σ there are

- a finite substitution s from Σ to some alphabet $\Gamma \supseteq \Sigma$,
- an inverse context-free ($\approx$ monadic) system M over Γ

so that

$$\boxed{\to_R^* = (s \circ \to_M^*)|_\Sigma}$$

Proof sketch: Replace $\ell_1 a \ell_2 \to r_1 y r_2$ (where a is $>$-maximal) with $\{a \to a_1 y a_2,\ \ell_1 a_1 \to r_1,\ a_2 \ell_2 \to r_2\}$ (a_1, a_2 new letters).



Example cont'd

$$R = \{ba \to cb, \;\; bd \to d, \;\; cd \to de, \;\; d \to \epsilon\}$$

i	pivot rule	S_i	M_i	N_i
0		$\emptyset$	$bd \to d, d \to \epsilon$	$ba \to cb, cd \to de$
1	$ba \to cb$	$a \to a_1 a_2$	$bd \to d, d \to \epsilon,$ $ba_1 \to c, a_2 \to b$	$cd \to de$
2	$cd \to de$	$c \to c_1 c_2$	$bd \to d, d \to \epsilon, ba_1 \to c,$ $a_2 \to b, c_1 \to d, c_2 d \to e$	$ba_1 \to c_1 c_2$
3	$ba_1 \to c_1 c_2$	$a_1 \to a_{1,1} a_{1,2}$	$bd \to d, d \to \epsilon, ba_1 \to c,$ $a_2 \to b, c_1 \to d, c_2 d \to e,$ $ba_{1,1} \to c_1, a_{1,2} \to c_2$	$\emptyset$

Example cont'd

$$R = \{ba \longrightarrow cb, \quad bd \longrightarrow d, \quad cd \longrightarrow de, \quad d \longrightarrow \epsilon\}$$

i	pivot rule	S_i	M_i	N_i
0		$\emptyset$	$bd \longrightarrow d, d \longrightarrow \epsilon$	$ba \longrightarrow cb, cd \longrightarrow de$
1	$ba \longrightarrow cb$	$a \longrightarrow a_1a_2$	$bd \longrightarrow d, d \longrightarrow \epsilon,$ $ba_1 \longrightarrow c, a_2 \longrightarrow b$	$cd \longrightarrow de$
2	$cd \longrightarrow de$	$c \longrightarrow c_1c_2$	$bd \longrightarrow d, d \longrightarrow \epsilon, ba_1 \longrightarrow c,$ $a_2 \longrightarrow b, c_1 \longrightarrow d, c_2d \longrightarrow e$	$ba_1 \longrightarrow c_1c_2$
3	$ba_1 \longrightarrow c_1c_2$	$a_1 \longrightarrow a_{1,1}a_{1,2}$	$bd \longrightarrow d, d \longrightarrow \epsilon, ba_1 \longrightarrow c,$ $a_2 \longrightarrow b, c_1 \longrightarrow d, c_2d \longrightarrow e,$ $ba_{1,1} \longrightarrow c_1, a_{1,2} \longrightarrow c_2$	$\emptyset$

Example cont'd

$$R = \{ba \to cb,\ \ bd \to d,\ \ cd \to de,\ \ d \to \epsilon\}$$

i	pivot rule	S_i	M_i	N_i
0		$\emptyset$	$bd \to d, d \to \epsilon$	$ba \to cb, cd \to de$
1	$ba \to cb$	$a \to a_1a_2$	$bd \to d, d \to \epsilon,$ $ba_1 \to c, a_2 \to b$	$cd \to de$
2	$cd \to de$	$c \to c_1c_2$	$bd \to d, d \to \epsilon, ba_1 \to c,$ $a_2 \to b, c_1 \to d, c_2d \to e$	$ba_1 \to c_1c_2$
3	$ba_1 \to c_1c_2$	$a_1 \to a_{1,1}a_{1,2}$	$bd \to d, d \to \epsilon, ba_1 \to c,$ $a_2 \to b, c_1 \to d, c_2d \to e,$ $ba_{1,1} \to c_1, a_{1,2} \to c_2$	$\emptyset$

Why does the transformation terminate?
Here, $(\text{base}\, N_i)_i$ is $\{a, c\} <_{\text{mset}} \{c\} <_{\text{mset}} \{a\} <_{\text{mset}} \emptyset$.

Corollaries

Deleting systems preserve REG.

Proof:

$$R^*(L) = (s \circ M^*)|_\Sigma(L) = M^*(s(L)) \cap \Sigma^*.$$

And REG is closed under finite substitution,
inverse context-free rewriting, and intersection with Σ^*.

Inverse deleting systems preserve CF.

Proof:

$$R^{-*}(L) = (R^*)^-(L) = ((s \circ M^*)|_\Sigma)^-(L) = s^-(M^{-*}(L)).$$

And CF is closed under context-free rewriting
and inverse finite substitution.

Application 1: Prefix Rewriting

For a given prefix rewriting system P define
a (standard) rewriting system

$$\boxed{P_\nabla = \{\nabla\ell \to r \mid \ell \to r \text{ in } P\}}$$

over $\Sigma \cup \{\nabla\}$. Note that P_∇ is deleting (choose $\nabla > a \in \Sigma$).

Then

$$\boxed{\nabla^* \cdot P^*(L) = {P_\nabla}^*(\nabla^* \cdot L)}$$

for $L \subseteq \Sigma^*$, thus $P^*(L) = \pi_\nabla({P_\nabla}^*(\nabla^* \cdot L))$,
and regularity of L implies regularity of $P^*(L)$ [Büchi 1964].

Application 2: Monadic Rewriting

For a given monadic rewriting system M define a rewriting system

$$\begin{aligned} M_\triangle = &\{h_\triangle(x) \to \epsilon \mid x \to \epsilon \text{ in } M\} \cup \\ &\{h_\triangle(x)\, a \to b \mid xa \to b \text{ in } M,\ a, b \in \Sigma\} \end{aligned}$$

over $\Sigma \cup \{\triangle\}$, where $h_\triangle : a \mapsto a\triangle$ for $a \in \Sigma$. Again, $M_\triangle$ is deleting. Then

$$M^*(L) = \pi_\triangle({M_\triangle}^*(h_\triangle(L)))$$

for $L \subseteq \Sigma^*$, and regularity of L implies regularity of $M^*(L)$ [Book, Jantzen, Wrathall 1982].

Further Applications

- Mixed prefix-, suffix-, and monadic rewriting (choose $\nabla > \Delta > a \in \Sigma$)
- Transductions
- $\ldots$
- Match-bounded rewriting [Geser, H, Waldmann 2003]

Match-Heights and -Bounds

Annotate letters by natural numbers (*heights*).
Let height in reduct $= 1 +$ minimum height in redex:
For R over Σ define (infinite) system $\mathrm{match}(R)$ over $\Sigma \times \mathbb{N}$:

$$\{\ell' \to \mathrm{lift}_{1+m}(r) \mid (\ell \to r) \in R,\ \mathrm{base}(\ell') = \ell,\ m = \min \mathrm{height}(\ell')\}$$

with morphisms

- $\mathrm{height} : \Sigma \times \mathbb{N} \to \mathbb{N} : (a, h) \mapsto h$
- $\mathrm{base} : \Sigma \times \mathbb{N} \to \Sigma : (a, h) \mapsto a$
- $\mathrm{lift}_h : \Sigma \to \Sigma \times \mathbb{N} : a \mapsto (a, h)$

Example: $\mathrm{match}(\{ab \to bc\}) = \{a_0 b_0 \to b_1 c_1, a_0 b_1 \to b_1 c_1, a_1 b_0 \to b_1 c_1, a_1 b_1 \to b_2 c_2, a_0 b_2 \to b_1 c_1, \ldots\}$

Match–Bounded Systems

System R is *match-bounded* for $L \subseteq \Sigma^*$ by $c \in \mathbb{N}$ if all heights in $\mathrm{match}(R)$-derivations starting from $\mathrm{lift}_0(L)$ are $\leq c$.

$$\boxed{\mathrm{match}_c(R) = \mathrm{match}(R)|_{\Sigma \times \{0,\ldots,c\}}}$$

- Observation: $\mathrm{match}_c(R)$ is deleting.
  Proof: Use precedence $(x, m) > (y, n)$ iff $m < n$.
- Example: Rule $a_0 b_2 \to b_1 c_1$ is deleting,
  since $a_0 > b_1$ and $a_0 > c_1$, since $0 < 1$.

Properties of Match-Bounded Systems

Basic observation: If R is match-bounded by c then

$$\boxed{R^* = \mathrm{lift}_0 \circ \mathrm{match}_c(R)^* \circ \mathrm{base}}$$

- If R is match-bounded (for L), then R is linearly terminating (on L).
- If R is match-bounded, then R preserves REG, and R^- preserves CF.
- "Is R match-bounded by c for $L \in \mathrm{REG}$?" is decidable.

Match-Bounded Systems: Examples

- $Z = \{a^2b^2 \to b^3a^3\}$ is match-bounded by 4.
  Thus, the system has linear derivational complexity [Tahhan-Bittar].

- Peg solitaire is a one-person game: remove pegs from a board by one peg X hopping over an adjacent peg Y. After the hop, Y is removed. Peg solitaire on a one-dimensional board corresponds to

$$P = \{\blacksquare\blacksquare\square \to \square\square\blacksquare, \square\blacksquare\blacksquare \to \blacksquare\square\square\}$$

  The language of all positions that can be reduced to one single peg: $P^{-*}(\square^*\blacksquare\square^*)$
  Regularity of $P^{-*}(\square^*\blacksquare\square^*)$ is a “folklore theorem”.
  P^- is match-bounded by 2, so we obtain yet another proof of that result.

Related Work: Change-Bounds

For R over Σ define (infinite) system $\text{change}(R)$ over $\Sigma \times \mathbb{N}$:

$$\{\ell' \to r' \mid (\ell \to r) \in R,\ \text{base}(\ell') = \ell,\ \text{base}(r') = r,$$
$$\text{height}(\text{successor}\, \ell') = \text{height}(r')\}$$

for *length-preserving* R, where $\text{successor}(x, c) = (x, c + 1)$.

Example: $\text{change}(\{ab \to bc\}) = \{a_0 b_0 \to b_1 c_1, a_0 b_1 \to b_1 c_2,$
$a_1 b_0 \to b_2 c_1, a_1 b_1 \to b_2 c_2, a_0 b_2 \to b_1 c_3, \ldots\}$.

[Ravikumar 1997]: R change-bounded $\Rightarrow$ R preserves REG.

New proof since R change-bounded $\Rightarrow$ R match-bounded.

Actually, $\Leftrightarrow$ holds.

Inverse Deleting Systems

$$\boxed{\mathrm{Inf}(R^*) = \{x \mid \exists^\infty y : x \to_R^* y\}}$$

Theorem [Geser, H, Waldmann 2003]
R inverse deleting $\Rightarrow$ $\mathrm{Inf}(R^*)$ regular (effectively).

Corollary

- R inverse deleting $\Rightarrow$ termination of R decidable.
- R inverse match-bounded $\Rightarrow$ termination of R decidable.

Proof: Check $\mathrm{Inf}(R^*) = \emptyset$. (Note that cycles are impossible.)

Example: Z^- is match-bounded by 2, and $\mathrm{Inf}(Z^*) = \emptyset$.
Thus Z is terminating.

Inverse Deleting Systems (cont'd)

Corollary

- R inverse deleting and L regular $\Rightarrow$ termination of R on L decidable.

- R inverse match-bounded and L regular $\Rightarrow$ termination of R on L decidable.

Proof: Check $\mathrm{Inf}(R^*) \cap L = \emptyset$.

Examples

- termination on one string: $L = \{x\}$
- termination on all strings: $L = \Sigma^*$

Inverse Deleting Systems (cont'd)

The following reachability problem is decidable:

GIVEN: An inverse match-bounded system R;
a context-free language L; a regular language M.

QUESTION: $\exists x \in L \; \exists y \in M : x \to_R^* y$?

Proof: Check $R^*(L) \cap M \neq \emptyset$.
Note: $R^*(L)$ is effectively context-free.

Example: The following reachability problem is decidable:

GIVEN: An inverse match-bounded system R over Σ;
two strings $x, y \in \Sigma^*$.

QUESTION: $\exists u, v \in \Sigma^* : x \to_R^* uyv$?

Proof: Choose $L = \{x\}$ and $M = \Sigma^*\{y\}\Sigma^*$.

No Match-Bounds

Exercise

- Show that

$$\boxed{\{ab \to ba\}}$$

is not match-bounded.

- How many proofs can you find?

Forward-Closures and Termination

Right forward closures modulo R:
$\mathrm{RFC}(R)$ is the least set $F \subseteq \Sigma^*$
that contains $\mathrm{rhs}(R)$ and is closed under

- rewriting:
  $u \in F \wedge u \rightarrow_R v \Rightarrow v \in F$
- right extension:
  $u\ell_1 \in F \wedge (\ell_1\ell_2 \rightarrow r) \in R \wedge \ell_1, \ell_2 \neq \epsilon \Rightarrow ur \in F$

Example: For $R = \{ba \rightarrow aab\}$, $\mathrm{RFC}(R) = a^{2*}b$.

Theorem [Dershowitz 1981]
R terminating on Σ^* iff R terminating on $\mathrm{RFC}(R)$.

Match-Bounds for Forward-Closures

$$\boxed{R_\# = R \cup \{\ell_1\# \to r \mid (\ell_1\ell_2 \to r) \in R,\ \ell_1, \ell_2 \neq \epsilon\}}$$

For $L = \mathrm{rhs}(R) \cdot \#^*$ we get

$$\boxed{\mathrm{RFC}(R) = {R_\#}^*(L) \cap \Sigma^*}$$

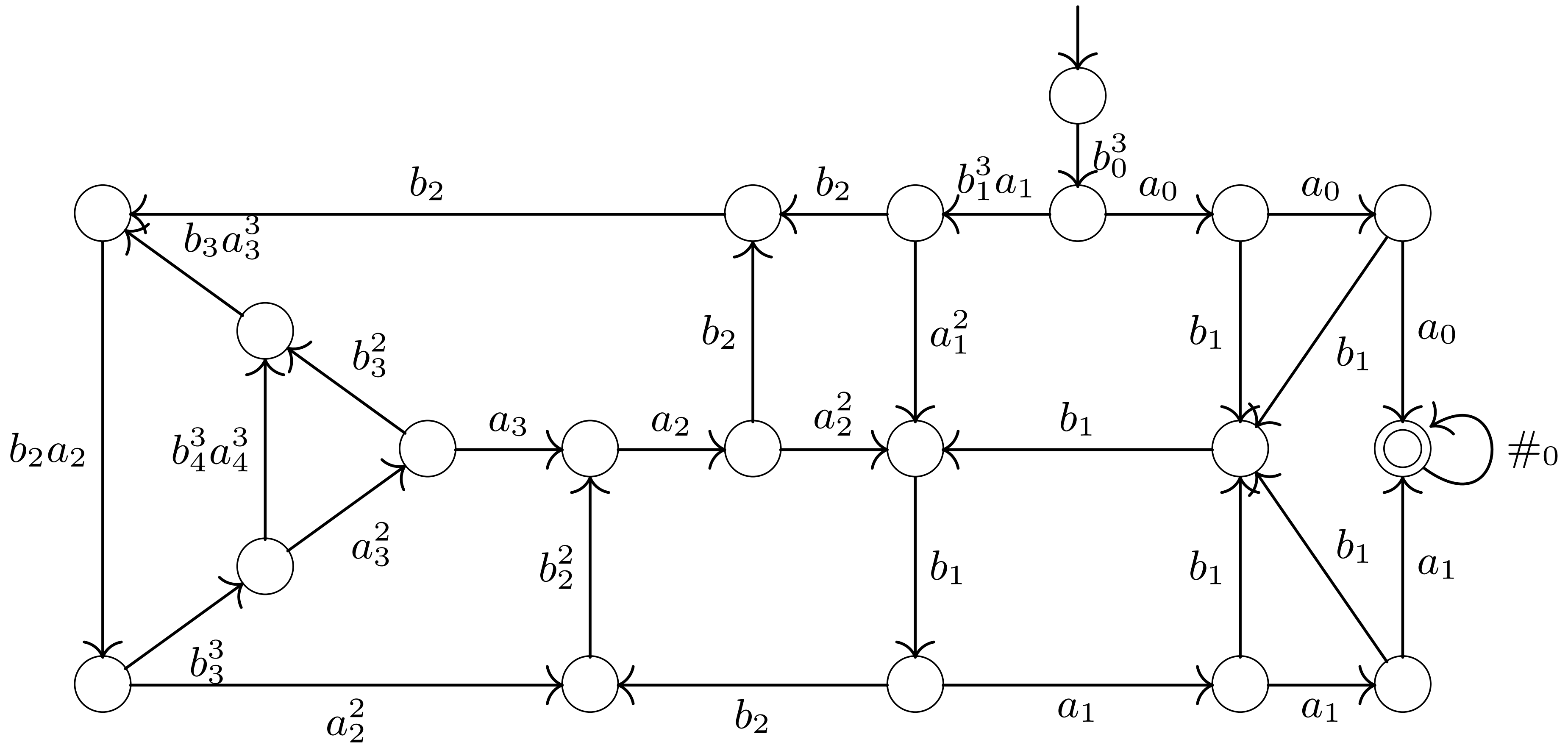
Theorem: $R_\#$ match-bounded for $L \Rightarrow R$ terminating on Σ^*.
Proof: $R \subseteq R_\#$ and $\mathrm{RFC}(R) \subseteq {R_\#}^*(L)$.

Remark: R linearly terminating on L,
but not necessarily linearly on Σ^* (example $\{ab \to ba\}$).

Example: $Z = \{a^2b^2 \to b^3a^3\}$

$Z_\# = \{a^2b^2,\ a^2b\#,\ a^2\#,\ a\#\} \to b^3a^3$.

Automaton for $\mathrm{match}(Z_\#)^*(\mathrm{lift}_0(\mathrm{rhs}(Z) \cdot \#^*))$:



Match-bound for $\mathrm{RFC}(Z)$ is 4 $\Rightarrow$ Z terminating.

Compatible Finite Automata

Automaton A is *compatible* with R over Σ and $L \subseteq \Sigma^*$ if

- $L \subseteq \mathcal{L}(A)$
- $p \xrightarrow{\ell}_A q$ implies $p \xrightarrow{r}_A q$ for states p, q and rules $\ell \to r$

Then $\to_R^*(L) \subseteq \mathcal{L}(A)$: *"overapproximation"*

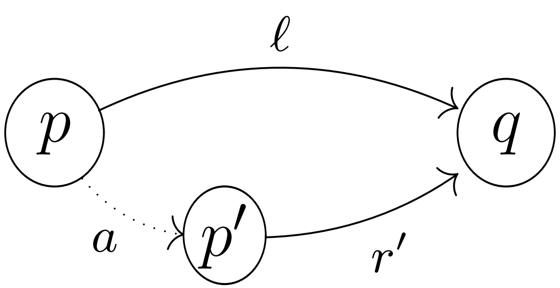
- A (possibly infinite) rewriting system R over a (possibly infinite) alphabet is *locally terminating* if every restriction of R to a finite subalphabet is terminating.
- If some *finite* automaton is compatible with R and L, and R is locally terminating, then R is terminating on L.
- Thus, if some finite automaton is compatible with $\mathrm{match}(R)$ and $\mathrm{lift}_0(L)$, then R is terminating on L.

Completion Strategies

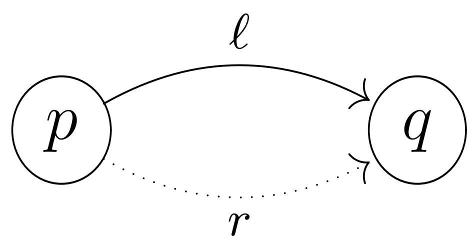
While A is not compatible repeat: If $p \xrightarrow{\ell}_A q$ and not $p \xrightarrow{r}_A q$ then add suitable transitions and states such that $p \xrightarrow{r}_A q$.

Implemented in Torpa, Matchbox, AProVE, TTT2.

TORPA heuristic

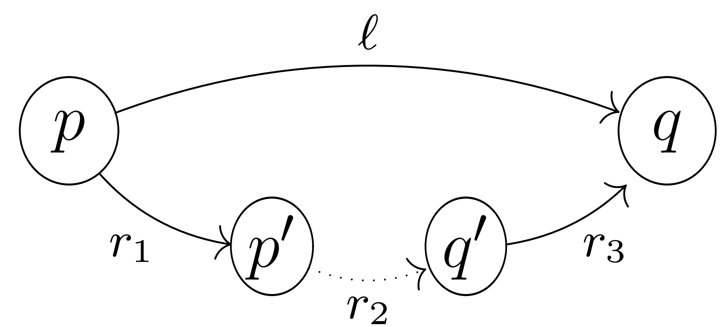


$r = ar', a \in \Gamma, r' \in \Gamma^*$



else

Matchbox heuristic



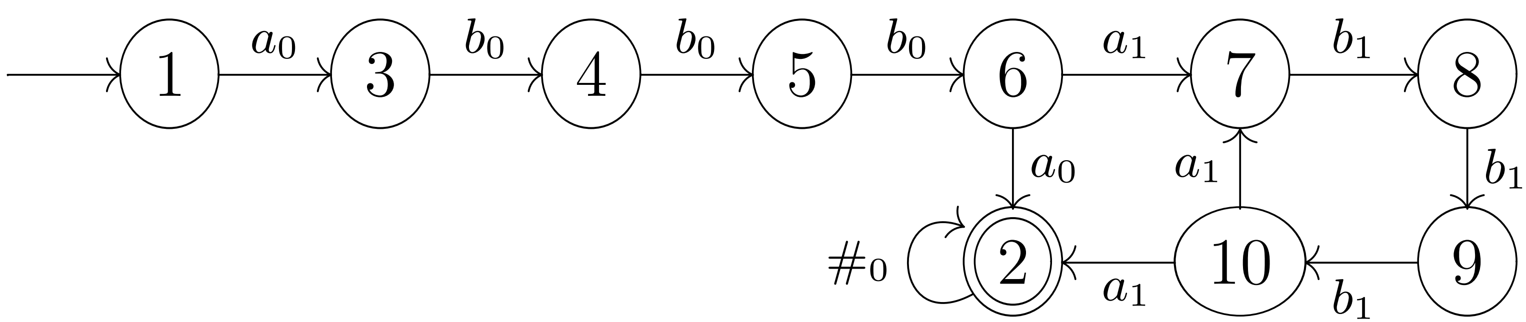
$r = r_1 r_2 r_3 \in \Gamma^*$

Compatibility: Example

Consider $R = \{aba \to abbba\}$. Then
$R_\# = \{aba \to abbba, a\# \to abbba, ab\# \to abbba\}$

$$\begin{aligned}\text{match}(R_\#) = &\{a_i b_j a_k \to a_m b_m b_m b_m a_m \mid m = \min\{i,j,k\}+1\} \cup \\ &\{a_i \#_j \to a_m b_m b_m b_m a_m \mid m = \min\{i,j\}+1\} \cup \\ &\{a_i b_j \#_k \to a_m b_m b_m b_m a_m \mid m = \min\{i,j,k\}+1\}\end{aligned}$$

This automaton is compatible with $\text{match}(R_\#)$
and $a_0 b_0 b_0 b_0 a_0 \#_0^*$, thus certifies match-bound 1:



Fast versus Exact

- exact approach is complete, but maybe intractable
- approx. approach is incomplete, but often successful

Fast versus Exact

- exact approach is complete, but maybe intractable
- approx. approach is incomplete, but often successful
- Good news
  - [Endrullis 2005] fast and exact decomposition
  - $\rightsquigarrow$ extra slides
  - $\rightsquigarrow$ live demo

Match-bounds for Term Rewriting

- Definition of match-heights and -bounds for TRSs is obvious, but the exact approach needs REG-preservation $\rightsquigarrow$ a decomposition result for *"deleting" TRSs*.

- Bad news: M.b.ness does not imply REG-preservation:

$$\{g(f(x,y)) \to f(g(x),g(y))\} \text{ on } g^*(f(a,a))$$

Match-bounds for Term Rewriting

- Definition of match-heights and -bounds for TRSs is obvious, but the exact approach needs REG-preservation $\rightsquigarrow$ a decomposition result for *"deleting" TRSs*.

- Bad news: M.b.ness does not imply REG-preservation:

$$\{g(f(x,y)) \to f(g(x),g(y))\} \text{ on } g^*(f(a,a))$$

- Alternative: Use the approximation approach to construct compatible tree automata
  - (left-)linear [Geser, H, Waldmann, Zantema 2005] using non-deterministic tree automata

Match-bounds for Term Rewriting

- Definition of match-heights and -bounds for TRSs is obvious, but the exact approach needs REG-preservation $\rightsquigarrow$ a decomposition result for *"deleting" TRSs*.

- Bad news: M.b.ness does not imply REG-preservation:

$$\{g(f(x,y)) \to f(g(x),g(y))\} \text{ on } g^*(f(a,a))$$

- Alternative: Use the approximation approach to construct compatible tree automata
  - (left-)linear [Geser, H, Waldmann, Zantema 2005] using non-deterministic tree automata
  - non-linear [Korp, Middeldorp 2007] using *"quasi-deterministic"* tree automata $\rightsquigarrow$ live demo

Matrix Interpretations

Expl.: z001 as a test case for automated termination methods

$$a \mapsto \begin{pmatrix} \boxed{1} & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \boxed{1} \end{pmatrix} \qquad b \mapsto \begin{pmatrix} \boxed{1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 2 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & \boxed{1} \end{pmatrix}$$

$$(\ell \to r) \mapsto \begin{pmatrix} 1 & 2 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 1 & 0 & 1 \\ 0 & 4 & 2 & 0 & 2 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} - \begin{pmatrix} 1 & 2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 1 \\ 0 & 1 & 2 & 0 & 2 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 & \boxed{1} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

- This interpretation proves termination since *all* entries are ≥ 0 and *marked* entries are $\boxed{\geq 1}$
- Found automatically / underlying theory elementary / fast verification

Ring Interpretations

Interpret the free monoid of strings in a ring:

- concatenation of factors $\mapsto$ multiplication
- replacement of factors $\mapsto$ subtraction

Ring Interpretations

Interpret the free monoid of strings in a ring:

- concatenation of factors $\mapsto$ multiplication
- replacement of factors $\mapsto$ subtraction

For termination: Use an (infinite) ordered ring, which is well-founded (on its "positive cone").

- Expl: $(\mathbb{Z}, 0, 1, +, \cdot)$ works for $\{aab \to ba\}$, but doesn't work for $\{ab \to ba\}$ as multiplication is commutative.

$\rightsquigarrow$ Use a non-commutative ring, e.g., a matrix ring

Well-founded Rings

A *partially ordered ring* $(D, 0, 1, +, \cdot, \geq)$:

- $(D, 0, +)$ an Abelian group, $(D, 1, \cdot)$ a monoid.
- Multiplication distributes over addition from both sides. (Multiplication not necessarily commutative / invertible.)
- $\geq$ is a compatible partial order:

$$a \geq b \Rightarrow a + c \geq b + c$$
$$a \geq b \wedge c \geq 0 \Rightarrow a \cdot c \geq b \cdot c \wedge c \cdot a \geq c \cdot b$$

Its *positive cone*: $N = \{d \in D \mid d \geq 0\}$,
its *strictly positive cone*: $P = N \setminus \{0\} = \{d \in D \mid d > 0\}$.
The ring is *well-founded* if $>$ is well-founded on N.

- Note: The order is uniquely determined by these cones:
  $a \geq b$ iff $a - b \in N$ and $a > b$ iff $a - b \in P$.
- Note: $N \cdot N \subseteq N$, but $P \cdot P \not\subseteq P$ if zero divisors exist.

Ring Interpretations (cont'd)

A *ring interpretation* of alphabet Σ is a mapping $i : \Sigma \to D$

- extended to a mapping $i : \Sigma^* \to D$ on strings by

$$\boxed{i(s_1 \cdot \ldots \cdot s_n) = i(s_1) \cdot \ldots \cdot i(s_n)}$$

- extended to a mapping $i : \Sigma^* \times \Sigma^* \to D$ on rules by

$$\boxed{i(\ell \to r) = i(\ell) - i(r)}$$

Ring Interpretations (cont'd)

Apply ring interpretations for proving termination:
Ensure $i(x\ell y) > i(xry)$ for each step $x\ell y \to_R xry$, i.e.,

$$\begin{aligned} i(x\ell y) - i(xry) &= i(x)i(\ell)i(y) - i(x)i(r)i(y) \\ &= \boxed{i(x)\Big(i(\ell) - i(r)\Big)i(y) \in P} \qquad (*) \end{aligned}$$

Given the set of interpretations of letters $i(\Sigma) = A$, what is the set of admissible interpretations of rules $i(R) = B$?

Ring Interpretations (cont'd)

Apply ring interpretations for proving termination:
Ensure $i(x\ell y) > i(xry)$ for each step $x\ell y \to_R xry$, i.e.,

$$\begin{aligned} i(x\ell y) - i(xry) &= i(x)i(\ell)i(y) - i(x)i(r)i(y) \\ &= \boxed{i(x)\Big(i(\ell) - i(r)\Big)i(y) \in P} \qquad (*) \end{aligned}$$

Given the set of interpretations of letters $i(\Sigma) = A$, what is the set of admissible interpretations of rules $i(R) = B$?
From $(*)$ it is obvious that $A^*BA^* \subseteq P$ is necessary.
The largest such set B is

$$\boxed{\mathrm{core}(A) = \{d \in D \mid A^*dA^* \subseteq P\}}$$

Example: For $A = \{\left(\begin{smallmatrix}1 & 0\\ 0 & 0\end{smallmatrix}\right)\}$ we get $\mathrm{core}(A) = \{d \mid d \geq \left(\begin{smallmatrix}1 & 0\\ 0 & 0\end{smallmatrix}\right)\}$.

Core Facts

- Increasing the range of interpretations of letters typically reduces the set that can safely be chosen as interpretations of rules:

$$\boxed{\text{If } A_1 \subseteq A_2, \text{ then } \mathrm{core}(A_1) \supseteq \mathrm{core}(A_2)}$$

- The range of all interpretations is upward closed: W.l.o.g. for the interpretation of letters by

$$\boxed{\mathrm{core}(A + N) = \mathrm{core}(A)}$$

and for the interpretation of rules by

$$\boxed{\mathrm{core}(A) + N = \mathrm{core}(A)}$$

Ring Interpretations (cont'd)

Let R be a string rewriting system over Σ.
An interpretation $i : \Sigma \to N$ into a p.o.-ring is order preserving

- from $(\Sigma^*, \to_R)$ to $(D, >)$ iff $i(R) \subseteq \mathrm{core}(i(\Sigma))$

Definition: Let A be a subset of the positive cone of a well-founded ring. Then $i : \Sigma \to A$ is an *A-interpretation* for R if

$$\boxed{i(R) \subseteq \mathrm{core}(A)}$$

Theorem:

- If there is an A-interpretation for R, then R is terminating.

Ring Interpretations (cont'd)

Let R, S be string rewriting systems over Σ.
An interpretation $i : \Sigma \to N$ into a p.o.-ring is order preserving

- from $(\Sigma^*, \to_R)$ to $(D, >)$ iff $i(R) \subseteq \mathrm{core}(i(\Sigma))$
- from $(\Sigma^*, \to_S)$ to $(D, \geq)$ iff $i(S) \subseteq N$

Definition: Let A be a subset of the positive cone of a well-founded ring. Then $i : \Sigma \to A$ is an *A-interpretation* for R if

$$\boxed{i(R) \subseteq \mathrm{core}(A)}$$

Theorem:

- If there is an A-interpretation i for R with $i(S) \subseteq N$, then R is terminating relative to S.

Matrix Interpretations

Consider the p.o. ring of square matrices
of a fixed dimension n over the integers: $D = \mathbb{Z}^{n \times n}$

- Addition / multiplication as usual.
- 0 and 1 are the zero and the identity matrix resp.
- The order is defined component-wise:
  $d \geq e$ if $\forall i, j : d_{i,j} \geq e_{i,j}$.
- The positive cone is $N = \mathbb{N}^{n \times n}$, and $P = N \setminus \{0\}$.
- The p.o. is well-founded on the positive cone.
- For $n > 1$, the p.o. is not total.

In order to apply the previous theorem we need
a set of matrices $A \subseteq N$ with non-empty $\mathrm{core}(A)$.

Matrix Classes

Two particular instances of the above method:

- Choose $A = M_I$ with $\mathrm{core}(A) = M_I$.
- Choose $A = E_I$ with $\mathrm{core}(A) = P_I$.

All these are simple "syntactically" defined subsets of N, parameterized by a set of matrix indices $I \subseteq \{1, \ldots, n\}$:

$$M_I = \{d \in N \mid \forall i \in I \exists j \in I : d_{i,j} > 0\}$$

$$E_I = M_I \cap M_I^{\mathrm{T}}$$

$$P_I = \{d \in N \mid \exists i \in I \exists j \in I : d_{i,j} > 0\}$$

Consider only entries $d_{i,j}$ with $i, j \in I$:

- M_I: no zero row
- E_I: no zero row, no zero column

Example: $\{aa \to aba\}/\{b \to bb\}$

$$i(a) = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \qquad i(b) = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$$

is an E_1-interpretation with
$i(aa \to aba) = i(aa) - i(aba) = \left(\begin{smallmatrix} 2 & 1 \\ 1 & 1 \end{smallmatrix}\right) - \left(\begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix}\right) = \left(\begin{smallmatrix} 1 & 0 \\ 0 & 0 \end{smallmatrix}\right) \in P_1$
and $i(b \to bb) = i(b) - i(bb) = 0 \in N$.

Alternatively, use the M_2-interpretation

$$i(a) = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \qquad i(b) = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}$$

with $i(aa \to aba) = \left(\begin{smallmatrix} 2 & 1 \\ 1 & 1 \end{smallmatrix}\right) - \left(\begin{smallmatrix} 2 & 0 \\ 1 & 0 \end{smallmatrix}\right) = \left(\begin{smallmatrix} 0 & 1 \\ 0 & 1 \end{smallmatrix}\right) \in M_2$ and
$i(b \to bb) = 0$. (This interpretation is not E_I for any I.)

Example: $\{aabb \to bbbaaa\}$

$$a \mapsto \begin{pmatrix} \boxed{1} & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \boxed{1} \end{pmatrix} \qquad b \mapsto \begin{pmatrix} \boxed{1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 2 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & \boxed{1} \end{pmatrix}$$

$$(\ell \to r) \mapsto \begin{pmatrix} 1 & 2 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 1 & 0 & 1 \\ 0 & 4 & 2 & 0 & 2 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} - \begin{pmatrix} 1 & 2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 1 \\ 0 & 1 & 2 & 0 & 2 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 & \boxed{1} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

This is an $E_{\{1,5\}}$-interpretation.

Example: Linear Interpretations

- All termination proofs by additive natural weights can be expressed as matrix interpretations: $(\mathbb{N}, +)$ is isomorphic to $(\{\left(\begin{smallmatrix} 1 & n \\ 0 & 1 \end{smallmatrix}\right) \mid n \in \mathbb{N}\}, \cdot\,)$ since

$$\left(\begin{smallmatrix} 1 & m \\ 0 & 1 \end{smallmatrix}\right) \cdot \left(\begin{smallmatrix} 1 & n \\ 0 & 1 \end{smallmatrix}\right) = \left(\begin{smallmatrix} 1 & m+n \\ 0 & 1 \end{smallmatrix}\right)$$

- More general: Linear interpretations
  - Interpret letters by functions $\lambda n.an + b$ on $\mathbb{N}$ with $a, b \in \mathbb{N}$ and $a \geq 1$,
  - concatenation is interpreted by function composition,
  - proof obligation is $\forall n : i(\ell)(n) > i(r)(n)$.

  This corresponds to matrix interpretations with matrices of the form $\left(\begin{smallmatrix} a & b \\ 0 & 1 \end{smallmatrix}\right)$.

A Normal Form for E_I-Proofs

Matrix interpretations are invariant under permutations:

- If i is an E_I- or M_I-interpretation for R,
- and if π is a permutation on the index set $\{1, \ldots, n\}$,
- then there is also an $E_{\pi(I)}$- / $M_{\pi(I)}$-interpretation for R.

A Normal Form for E_I-Proofs

Matrix interpretations are invariant under permutations:

- If i is an E_I- or M_I-interpretation for R,
- and if π is a permutation on the index set $\{1, \ldots, n\}$,
- then there is also an $E_{\pi(I)}$- / $M_{\pi(I)}$-interpretation for R.

$\Rightarrow$ W.l.o.g. we can replace an arbitrary set I by $\{1, \ldots, |I|\}$.

A Normal Form for E_I-Proofs

Matrix interpretations are invariant under permutations:

- If i is an E_I- or M_I-interpretation for R,
- and if π is a permutation on the index set $\{1, \ldots, n\}$,
- then there is also an $E_{\pi(I)}$- / $M_{\pi(I)}$-interpretation for R.

$\Rightarrow$ W.l.o.g. we can replace an arbitrary set I by $\{1, \ldots, |I|\}$.

$\Rightarrow$ A normal form: Choose $J = \{1, n\}$.

- A proof of $\mathrm{SN}(R/S)$ via some E_I-interpretation can be replaced by a sequence of E_J-interpretations which successively remove the same rules.

Implementations: Performance

Percentage of YES in the 2006 SRS competition:

- MultumNonMulta (H) 51 %
  matrix interpretations only
- Matchbox/Satelite (Waldmann) 68 %
  labelling, matrices, RFC match-bounds
- TORPA (Zantema) 75 %
  various techniques, including 3×3 matrices
- Jambox (Endrullis) 94 %
  $\approx$ Matchbox $+$ dependency pairs

(2007 competition of partial significance . . .)

Implementations: TORPA

Random guesses or complete enumeration, using matrix shape

$$\begin{pmatrix} 0 & * & \boxed{+} \\ 0 & * & * \\ 0 & 0 & 0 \end{pmatrix} \subseteq \text{core} \begin{pmatrix} 1 & * & * \\ 0 & * & * \\ 0 & 0 & 1 \end{pmatrix}$$

with $* \in \{0, 1, 4\}$. Occurs in 36% of its proofs, e.g. z007:

```
TORPA 1.6 is applied to
a b  -> b a , b a  -> a a c b ,
[A] Choose interpretation in NxN,
order :  (x,y) > (x',y') <==> x > x' & y >= y'
a : lambda (x,y) . (x+y,4y)
b : lambda (x,y) . (x,4y+1)
c : lambda (x,y) . (x,0)
remove: a b  -> b a
```

Implementations: MultumNonMulta

- Random guesses, random restart hill climbing; complete enumeration, ... (not in the competition version)
- Backward completion, see below $\rightsquigarrow$ live demo
  - Examples: z061 / z062 / ...
  - Example: Waldmann/r10

$$\mathrm{SN}(\{ba^2b \to a^4, ab^2a \to b^4\}/\{b \to b^3\})$$

Sparse 14×14 matrices (250 sec '06 / 10 sec '07)

- Determine additive weights using the GNU Linear Programming Kit.
  - Example: z112 / ...

Implementations: SAT Solving

- Fix dimension, say 5 $\rightsquigarrow$ Constraint system
  - $|\Sigma| \cdot d^2$ unknowns (matrix entries) and
  - $|R| \cdot d^2$ constraints (entries in differences).
- Fix maximal value for entries, say $7 = 2^3 - 1$ $\rightsquigarrow$ Finite domain constraint system
  - Binary encoding of entries $\rightsquigarrow$ boolean SAT problem: e.g. 15.000 variables, 90.000 clauses, 300.000 literals
  - Solve by SAT solver, e.g. SatELiteGTI. Expl: z001 takes 7 seconds
- Jambox: Linear programming + SAT solving.
- Matchbox: Likewise, but using only one bit per entry: Computation in $\{0, 1\} \subset \mathbb{N}$, so $1 + 1$ is "forbidden".

Limitation: Derivational complexity

In a product of k matrices from a finite set,
entries are bounded by an exponential function in k.
Assume R has derivational complexity above exponential.

Limitation: Derivational complexity

In a product of k matrices from a finite set,
entries are bounded by an exponential function in k.
Assume R has derivational complexity above exponential.

$\Rightarrow$ There can be no strict matrix interpretation for R.
Expl: $\{ab \to baa,\ cb \to bbc\}$

- Derivational complexity doubly exponential.
- But: "Relative" matrix proof with step-wise removal of rules is possible (first remove $cb \to bbc$).

Limitation: Derivational complexity

In a product of k matrices from a finite set, entries are bounded by an exponential function in k.
Assume R has derivational complexity above exponential.

$\Rightarrow$ There can be no strict matrix interpretation for R.
Expl: $\{ab \to baa,\ cb \to bbc\}$

- Derivational complexity doubly exponential.
- But: "Relative" matrix proof with step-wise removal of rules is possible (first remove $cb \to bbc$).

$\Rightarrow$ There can be no matrix interpretation at all for R if each rule occurs "equally often".
Expl: $\{ab \to bca, cb \to bbc\}$ (z018, z020)

- Derivational complexity tower of exponentials.
- But: Terminating via DP $+$ matrix interpretations
- (and RPO-terminating).

Limitation: Dimension restrictions

A matrix ring is not *free*: Certain polynomial identities hold.

- Dimension 1: $[A, B] = 0$

  where $[A, B] = AB - BA$ (*commutator*)

  $\Rightarrow$ No 1-dim termination proof for $\{ab \to ba\}$.

Limitation: Dimension restrictions

A matrix ring is not *free*: Certain polynomial identities hold.

- Dimension 1: $[A, B] = 0$

  where $[A, B] = AB - BA$ (*commutator*)

  $\Rightarrow$ No 1-dim termination proof for $\{ab \to ba\}$.
- Dimension 2: $[[A, B]^2, C] = 0$

  $\Rightarrow$ No 2-dim termination proof for

  $\{abcbc \to cbcba, acbcb \to bcbca, bccba \to abccb, cbbca \to acbbc\}$

  (Is RFC match-bounded. Matrix proof not known.)

Similar identities hold for matrix rings of any dimension.

Limitation: Dimension restrictions

A matrix ring is not *free*: Certain polynomial identities hold.

- Dimension 1: $[A, B] = 0$
  where $[A, B] = AB - BA$ (*commutator*)
  $\Rightarrow$ No 1-dim termination proof for $\{ab \to ba\}$.
- Dimension 2: $[[A, B]^2, C] = 0$
  $\Rightarrow$ No 2-dim termination proof for
  $\{abcbc \to cbcba, acbcb \to bcbca, bccba \to abccb, cbbca \to acbbc\}$
  (Is RFC match-bounded. Matrix proof not known.)

Similar identities hold for matrix rings of any dimension.

Define SRS hierarchy by "minimal matrix proof dimension":

- Is every level inhabited? Which levels are decidable?
  [Gebhardt, Waldmann 2008]

Proof Verification

- Although probably hard to find, a termination proof via matrix interpretations is easy to verify ...

- ... and verification is fast: PTIME

Proof Verification

- Although probably hard to find, a termination proof via matrix interpretations is easy to verify . . .

- . . . and verification is fast: PTIME

- Even if the matrix type is not “syntactically” specified:
  - It is decidable whether an arbitrary matrix interpretation i satisfies $i(R) \subseteq \mathrm{core}(i(\Sigma))$.
  - Even more: we can effectively determine a finite set $C \subseteq P$ such that $\mathrm{core}(i(\Sigma)) = \{d \geq c \mid c \in C\}$.

Weighted Automata

Transitions have a natural number as *weight*:

A *weighted automaton* "is" a mapping $Q \times \Sigma \times Q \to \mathbb{N}$.

This mapping is extended to $Q \times \Sigma^* \times Q \to \mathbb{N}$:

- multiply weights along a single path,
- add weights of different paths.

Weighted Automata

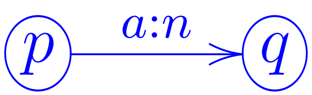
Transitions have a natural number as *weight*:

A *weighted automaton* “is” a mapping $Q \times \Sigma \times Q \to \mathbb{N}$.

This mapping is extended to $Q \times \Sigma^* \times Q \to \mathbb{N}$:

- multiply weights along a single path,
- add weights of different paths.

W.l.o.g. $Q = \{1, \ldots, n\}$.
For a transition from state p to state q with weight n for letter a, the following representations are equivalent:

- State diagram: 

- Matrix interpretation: $i(a)_{p,q} = n$

Weighted Automata (cont'd)

- Matrix multiplication computes the transitive closure:

  For $x \in \Sigma^*$, the $\boxed{\text{weight of path } (p) \xrightarrow{x} (q) \text{ is } i(x)_{p,q}}$

Weighted Automata (cont'd)

- Matrix multiplication computes the transitive closure:

  For $x \in \Sigma^*$, the weight of path $(p) \xrightarrow{x} (q)$ is $i(x)_{p,q}$

- "Standard" automata: $Q \times \Sigma \times Q \to \{0, 1\}$.

- Other (semi-)rings possible ...

Zantema's System (cont'd)

The above matrix interpretation:

$$a \mapsto \begin{pmatrix} \boxed{1} & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \boxed{1} \end{pmatrix} \qquad b \mapsto \begin{pmatrix} \boxed{1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 2 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & \boxed{1} \end{pmatrix}$$

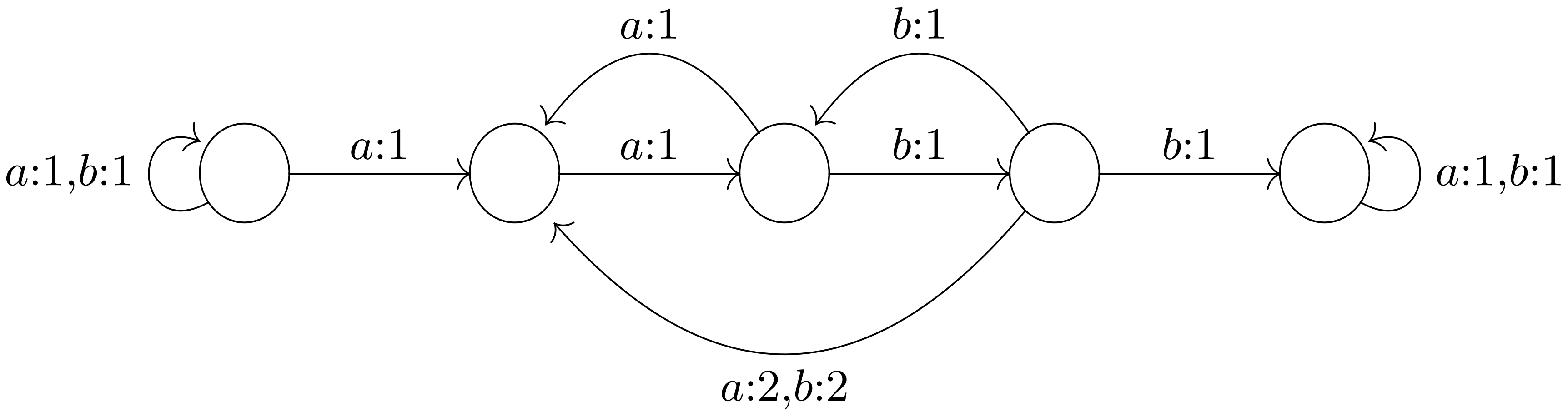
$$(\ell \to r) \mapsto \begin{pmatrix} 1 & 2 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 1 & 0 & 1 \\ 0 & 4 & 2 & 0 & 2 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} - \begin{pmatrix} 1 & 2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 1 \\ 0 & 1 & 2 & 0 & 2 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 & \boxed{1} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

proves termination since

- *all* entries are ≥ 0 and
- *marked* entries are $\boxed{\geq 1}$

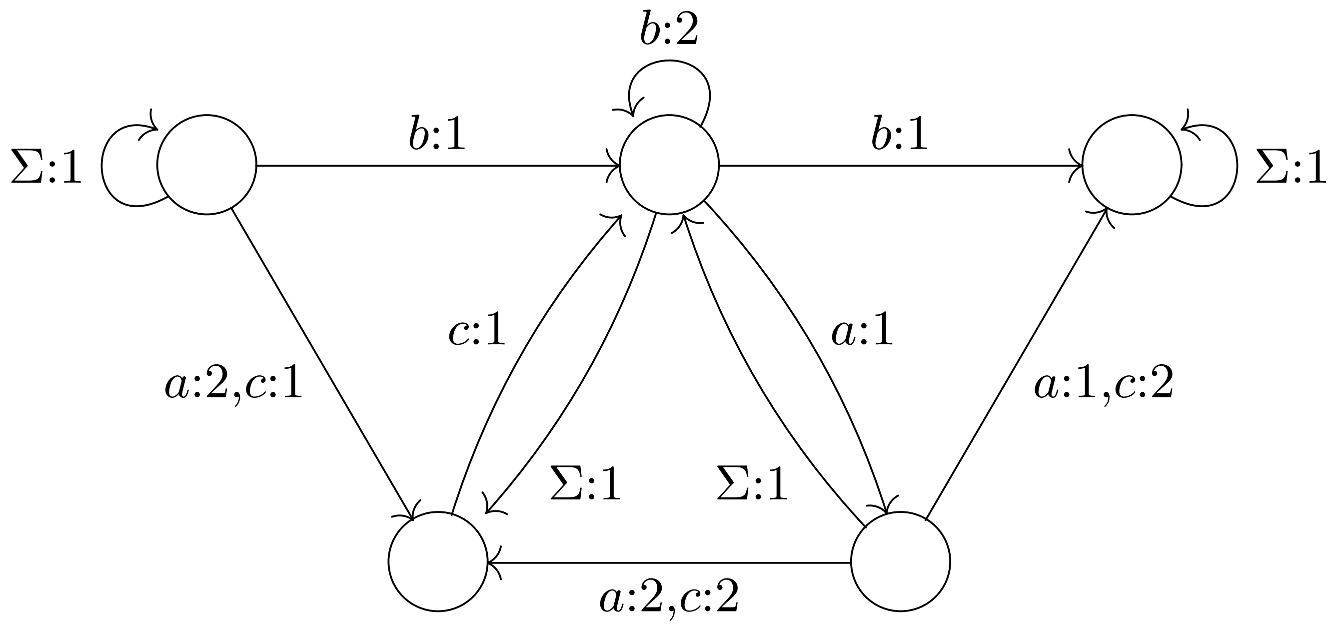
Zantema's System (cont'd)

The same termination proof as a weighted automaton:



Example: $\{aa \to bc,\ bb \to ac,\ cc \to ab\}$

Solution for RTA List of Open Problems $\#104$:



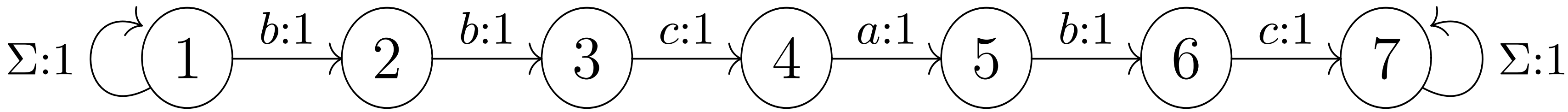
A variant was published as a *monotone algebra* in IPL'06.

Automata: Large and Sparse

- Example: $\{bbcabc \rightarrow abbcbca\}$ (z061)

Automata: Large and Sparse

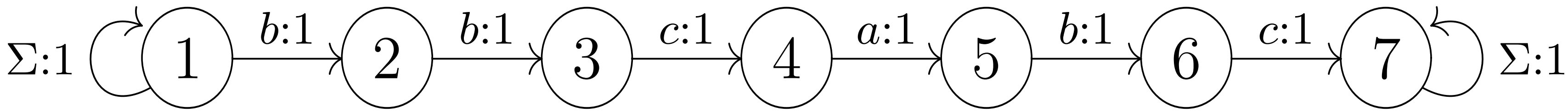
- Example: $\{bbcabc \rightarrow abbcbca\}$ (z061)



Done.

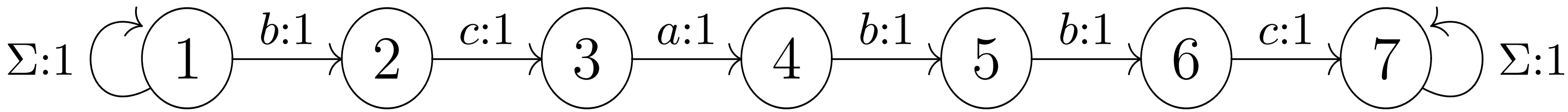
Automata: Large and Sparse

- Example: $\{bbcabc \rightarrow abbcbca\}$ (z061)



Done.

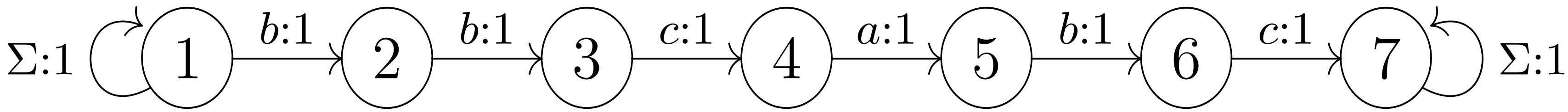
- Example: $\{bcabbc \rightarrow abcbbca\}$ (z062)



No: $\text{weight}\left(\textcircled{1} \xrightarrow{bcabbc} \textcircled{4}\right) = 0 \not\geq 1 = \text{weight}\left(\textcircled{1} \xrightarrow{abcbbca} \textcircled{4}\right)$

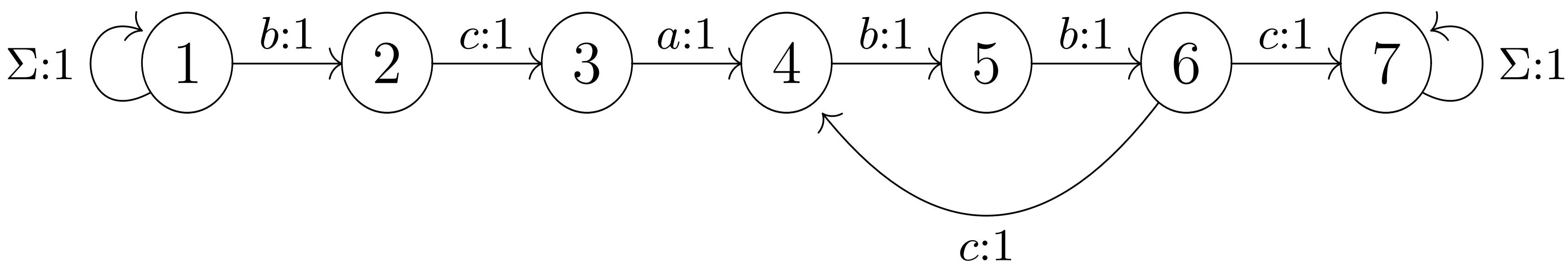
Automata: Large and Sparse

- Example: $\{bbcabc \to abbcbca\}$ (z061)



Done.

- Example: $\{bcabbc \to abcbbca\}$ (z062)



Done: $\text{weight}\left(1 \xrightarrow{bcabbc} 4\right) = 1 = \text{weight}\left(1 \xrightarrow{abcbbca} 4\right)$

Matrix Int's for Term Rewriting

Linear combinations of matrix interpretations
[Endrullis, Waldmann, Zantema 2006]

- monotone algebra framework
- vectors as domain: $\mathbb{N}^n$
- interpretations of the form

$$\boxed{f_\tau(\vec{v_1}, \ldots, \vec{v_n}) = M_1 \vec{v_1} + \cdots + M_n \vec{v_n} + \vec{v}}$$

where $M_i \in \mathbb{N}^{n \times n}$ with $\boxed{(M_i)_{1,1} > 0}$ and $\vec{v} \in \mathbb{N}^n$

Matrix Int's for Terms (cont'd)

Dependency pairs [Arts, Giesl 2000]

$$\mathrm{SN}(R) \quad \text{iff} \quad \mathrm{SN}(\mathrm{DP}(R)_{\mathrm{top}}/R)$$

- The matrix method supports relative termination $\Rightarrow$ it supports this basic version of the DP method
- Marker symbols encode the idea that $\mathrm{DP}(R)$ steps only happen at the left end (for terms: top position). [Endrullis, Waldmann, Zantema 2006]: the matrix method can be adapted to relative top-termination
- and can be combined with refinements [Hirokawa, Middeldorp 2004]

Problems

- Further instances of the general scheme are conceivable: Other matrix classes?
- Explain the relationship between proofs via E_I and via M_I.
- Explain the relationship between proofs via M_I and via $M_{I'}$ for $I \neq I'$.
- A normal form for M_I-proofs?
- Good heuristics for backward completion

Grand Unified Theory

- Matrix interpretations are weighted finite automata.
- The method of (RFC) match-bounds also builds on weighted (annotated) automata.

Unified view $\rightsquigarrow$ [Waldmann, work in progress]

- Natural semi-ring $(\mathbb{N}, +, \cdot, 0, 1)$
- Boolean semi-ring $(\{0, 1\}, +, \cdot, 0, 1)$
- Tropical semi-ring $(\mathbb{N} \cup \{\infty\}, \min, +, \infty, 0)$
  [W '08, unpublished]: subsumes match-boundedness
- Arctic semi-ring $(\mathbb{N} \cup \{-\infty\}, \max, +, -\infty, 0)$
  [W '07]: subsumes *quasi-periodic interpretations* by [W, Zantema '07]
- . . . below zero $(\mathbb{Z} \cup \{-\infty\}, \max, +, -\infty, 0)$
  [Koprowski, W '08]

Derivational Complexity

Research program

- Deduce upper/lower bounds on derivation lengths from termination proofs.
- Characterize complexity classes via termination proof methods.

The *derivation height* of term t modulo system R is

$$\mathrm{dh}_R(t) = \max\{n \mid \exists s : t \to_R^n s\}$$

The *derivational complexity* of R is

$$\mathrm{dc}_R(n) = \max\{\mathrm{dh}_R(t) \mid \mathrm{size}(t) \leq n\}$$

- exercise: show $\mathrm{dc}_R(n+1) \geq \mathrm{dc}_R(n)$
- exercise: show $\mathrm{dc}_R(n) \in \Omega(n)$ for non-trivial R

String Rewriting: Examples

1. $R = \{aa \to aba\}$, $\mathrm{dc}_R \in \Theta(n)$

String Rewriting: Examples

1. $R = \{aa \to aba\}$, $\mathrm{dc}_R \in \Theta(n)$
2. $R = \{ab \to ba\}$, $\mathrm{dc}_R \in \Theta(n^2)$

String Rewriting: Examples

1. $R = \{aa \to aba\}$, $\mathrm{dc}_R \in \Theta(n)$
2. $R = \{ab \to ba\}$, $\mathrm{dc}_R \in \Theta(n^2)$
3. $R = \{ab \to baa\}$, $\mathrm{dc}_R \in \Theta(2^n)$

String Rewriting: Examples

1. $R = \{aa \to aba\}$, $\mathrm{dc}_R \in \Theta(n)$
2. $R = \{ab \to ba\}$, $\mathrm{dc}_R \in \Theta(n^2)$
3. $R = \{ab \to baa\}$, $\mathrm{dc}_R \in \Theta(2^n)$
4. $R = \{aabab \to aPb, aP \to PAa, aA \to Aa,$
   $bP \to bQ, QA \to aQ, Qa \to babaa\}$
   dc_R not primitive recursive (Ackermann)

String Rewriting: Examples

1. $R = \{aa \to aba\}$, $\mathrm{dc}_R \in \Theta(n)$
2. $R = \{ab \to ba\}$, $\mathrm{dc}_R \in \Theta(n^2)$
3. $R = \{ab \to baa\}$, $\mathrm{dc}_R \in \Theta(2^n)$
4. $R = \{aabab \to aPb, aP \to PAa, aA \to Aa, bP \to bQ, QA \to aQ, Qa \to babaa\}$

   dc_R not primitive recursive (Ackermann)
5. Etc. (string rewriting is computationally complete)

String Rewriting: Examples

1. $R = \{aa \to aba\}$, $\mathrm{dc}_R \in \Theta(n)$
2. $R = \{ab \to ba\}$, $\mathrm{dc}_R \in \Theta(n^2)$
3. $R = \{ab \to baa\}$, $\mathrm{dc}_R \in \Theta(2^n)$
4. $R = \{aabab \to aPb, aP \to PAa, aA \to Aa, bP \to bQ, QA \to aQ, Qa \to babaa\}$
   dc_R not primitive recursive (Ackermann)
5. Etc. (string rewriting is computationally complete)

We can deduce some of these bounds automatically:

1. via match bounds
2. via upper triangular 3×3 matrix interpretations
3. via matrix interpretations

Some Results for Term Rewriting

- polynomial interpretations $\leadsto$ doubly exponential [Lautemann / Geupel / H / Zantema / . . .]
- multiset path orders $\leadsto$ primitive recursive [H]
- lexicographic path orders $\leadsto$ multiple recursive [Weiermann]
- Knuth-Bendix orders $\leadsto$ multiple recursive (2-rec) [H, Lautemann / Touzet / Lepper / Bonfante / Moser]
- Related [Buchholz / Touzet / Weiermann / Moser . . .]
- match bounds $\leadsto$ linear [Geser, H, Waldmann]
- matrix interpretations $\leadsto$ exponential [H, Waldmann], polynomial in particular cases [Waldmann]
- context-dependent interpretations $\leadsto$ see below [H]

Research Problem

Challenge: *Small* complexity classes.
Here, upper bound results heavily overestimate dc_R.

Some remedies:

- Syntactic restrictions of standard path orders
  - light multiset path order LMPO [Marion 2003]
  - polynomial path order POP*: innermost derivations on constructor-based terms [Avanzini, Moser 2008], cf. [Bellantoni, Cook 1992]
- Matrix interpretations of particular shape [Waldmann 2007]
- Context-dependent interpretations [H 2001 / Schnabl, Moser 2008]

Interpretations and Derivation Lengths

For an interpretation τ for R into a Σ-algebra over $\mathbb{N}$, $s \to_R t$ implies $\tau(s) > \tau(t)$. Thus, for $t \in \mathcal{T}_\Sigma$,

$$\boxed{\mathrm{dh}_R(t) \leq \tau(t)}$$

- Main Lemma. Let τ be a monotone interpretation for R into $(\mathbb{N}, \geq)$ and let $p : \mathbb{N} \to \mathbb{N}$ be strictly monotone such that for all $f \in \Sigma$ and $k \in \mathbb{N}$, $p(k) \geq f_\tau(k, \ldots, k)$. Then

$$\mathrm{dh}_R(t) \leq p^{\mathrm{depth}(t)}(0)$$

$$\mathrm{dc}_R(n) \leq p^n(0)$$

- Proof: exercise (hints: induction on t; $\mathrm{depth}(t) \leq \mathrm{size}(t)$)

Corollaries

1. If p is a linear function, then $\mathrm{dc}_R(n) \in 2^{O(n)}$.
2. If p is a polynomial, then $\mathrm{dc}_R(n) \in 2^{2^{O(n)}}$.
3. If p is an exponential function, then $\mathrm{dc}_R(n) \in E_4$.
4. If $p \in E_k$, then $\mathrm{dc}_R(n) \in E_{k+1}$, for $k \geq 2$.

Here, E_k denotes the k-th level of the Grzegorczyk hierarchy.

Remark: 2. and 3. are special cases of 4.

Example

Consider the (length preserving) system FIB

$$\{aab \to bba,\ b \to a\}$$

Example

Consider the (length preserving) system FIB

$$\{aab \to bba,\ b \to a\}$$

- exponential lower bound:
  $b^n \to^k b^{n-1}a$ where $k \geq \text{fib}(n)$ (Fibonacci number)

  $b^n \to^{\geq \text{fib}(n-1)} b^{n-2}ab \to^{\geq \text{fib}(n-2)} b^{n-3}aab \to b^{n-3}bba = b^{n-1}a$

Example

Consider the (length preserving) system FIB

$$\{aab \to bba,\ b \to a\}$$

- exponential lower bound:
  $b^n \to^k b^{n-1}a$ where $k \geq \mathrm{fib}(n)$ (Fibonacci number)

  $$b^n \to^{\geq \mathrm{fib}(n-1)} b^{n-2}ab \to^{\geq \mathrm{fib}(n-2)} b^{n-3}aab \to b^{n-3}bba = b^{n-1}a$$

- termination proof by linear functions:

  $$\tau : a \mapsto \lambda n.2n,\ b \mapsto \lambda n.2n+1$$

  thus $\tau(aabw) = 8\tau(w) + 4 > 8\tau(w) + 3 = \tau(bbaw)$,
  which implies a single exponential upper bound
  by the main lemma: choose $p = \tau(b)$

Example

Consider the system CNF

$$\neg(x \wedge y) \to \neg(x) \vee \neg(y)$$

$$\neg(x \vee y) \to \neg(x) \wedge \neg(y)$$

$$x \vee (y \wedge z) \to (x \vee y) \wedge (x \vee z)$$

$$(x \wedge y) \vee z \to (x \vee z) \wedge (y \vee z)$$

- CNF allows derivation heights not bounded by any elementary function (exercise), thus by the above corollary no *polynomial interpretation* can prove termination, as conjectured by Dershowitz.
- Termination *can* be proven using exponential functions, however (exercise).

Embedding Relations

From homeomorphic embedding to path orderings . . .

- Define the rewriting system HE as

$$f(x_1, \ldots, x_n) \to x_i$$

  The *homeomorphic embedding* relation is $>_{\mathsf{HE}} = \to^+_{\mathsf{HE}}$.

- For a given *precedence* $>$ (well-founded ordering on Σ), define the rewriting system HP as

$$f(x_1, \ldots, x_n) \to c_{<f}[x_1, \ldots, x_n]$$

  where $c_{<f}$ denotes any context with symbols $< f$.
  $>_{\mathsf{HP}} = \to^+_{\mathsf{HP}}$ is the *homeomorphic embedding with precedence*.

Embedding Relations (cont'd)

- For a given *precedence* $>$ the rewriting system PE is

$$f(x_1, \ldots, x_n) \to c_{<f}[x_1, \ldots, x_n]$$

$$f(x_1, \ldots, g(y_1, \ldots, y_m), \ldots, x_n) \to$$
$$c_{<f}[f(x_1, \ldots, y_1, \ldots, x_n), \ldots, f(x_1, \ldots, y_m, \ldots, x_n)]$$

$>_{\mathsf{PE}} = \to^{+}_{\mathsf{PE}}$ is called *primitive embedding*.

- similarly: *generalized embedding*

- *multiset path order*

- *lexicographic/recursive path order*

Embedding Relations (cont'd)

Via the Key Lemma:

- *homeomorphic embedding* implies
  linear upper bound on dc_R
- *homeo. embedding with precedence* implies
  single exponential upper bound on dc_R
- *primitive / generalized embedding / mpo* imply
  primitive recursive upper bound on dc_R
- etc.

Traditional Interpretations

For an interpretation τ for R into a Σ-algebra over $\mathbb{N}$, $s \rightarrow_R t$ implies $\tau(s) - \tau(t) \geq 1$. Thus

$$\boxed{\mathrm{dh}_R(t) \leq \tau(t)}$$

- τ as a Σ-homomorphism:

$$\tau(f(\ldots t \ldots)) = f_\tau(\ldots \tau(t) \ldots)$$

- all functions f_τ strictly monotone

Then it suffices to show $\tau(\ell\gamma) - \tau(r\gamma) \geq 1$.

Example $abx \to bax$

Choose

$$a_\tau(n) = 2n$$
$$b_\tau(n) = 1 + n$$
$$c_\tau = 0$$

Then $\tau(abt) - \tau(bat) = 2(1 + \tau(t)) - (1 + 2\tau(t)) = 1$.
Both a_τ and b_τ are strictly monotone.

For instance $\boxed{\tau(a^n b^m c) = 2^n \cdot m}$ but $\boxed{\mathrm{dh}_R(a^n b^m c) = n \cdot m}$
HUGE GAP. Problem:

$$\tau(a^k\ ab\ t) - \tau(a^k\ ba\ t) = 2^k,$$

reflecting *one* rewrite step.

Context-dependent Interpretations

- Now, interpretation τ is parameterized with $\Delta \in \mathbb{Q}_0^+$.
- Show $s \to_R t$ implies $\tau[\Delta](s) - \tau[\Delta](t) \geq \Delta$. Then

$$\boxed{\mathrm{dh}_R(t) \leq \tau[\Delta](t)/\Delta}$$

  Thus

$$\boxed{\mathrm{dh}_R(t) \leq \inf_{\Delta > 0} \frac{\tau[\Delta](t)}{\Delta}}$$

- Term evaluation now depends on Δ:

$$\tau[\Delta](f(\ldots t_i \ldots)) = f_\tau[\Delta](\ldots \tau[f_\tau^i(\Delta)](t_i) \ldots)$$

- Extra constraints to ensure that
  $\tau[\Delta](\ell\gamma) - \tau[\Delta](r\gamma) \geq \Delta$ suffices: Δ-*monotonicity*

Example $abx \to bax$ (cont'd)

Idea: introduce parameter via $2 \mapsto 1 + \Delta$.

From here on, no *creative step* is needed at all. Choose

$$a_\tau[\Delta](z) = (1 + \Delta)z$$
$$b_\tau[\Delta](z) = 1 + z$$
$$c_\tau[\Delta] = 0$$

The Δ-monotonicity constraint is (analogously for b_τ)

$$a_\tau[\Delta](z + a_\tau^1(\Delta)) - a_\tau[\Delta](z) \geq \Delta$$

That is, $a_\tau[\Delta]$ propagates a difference of at least Δ, provided a difference of at least $a_\tau^1(\Delta)$ (in argument 1) is given.

Example $abx \rightarrow bax$ (cont'd)

Solving these constraints gives

$$a^1_\tau(\Delta) \geq \frac{\Delta}{1+\Delta}$$
$$b^1_\tau(\Delta) \geq \Delta$$

Choosing $=$ for $\geq$, we found rather systematically

$$\tau[\Delta](a(t)) = (1+\Delta) \cdot \tau\left[\frac{\Delta}{1+\Delta}\right](t)$$
$$\tau[\Delta](b(t)) = 1 + \tau[\Delta](t)$$
$$\tau[\Delta](c) = 0$$

Example $abx \rightarrow bax$ (cont'd)

- Show $\tau[\Delta](abt) - \tau[\Delta](bat) \geq \Delta$ (exercise)
- E.g. $\tau[\Delta](a^n b^m c) = (1 + \Delta n)m$

Thus

$$\mathrm{dh}_R(a^n b^m c) \leq \inf_{\Delta > 0} \frac{\tau[\Delta](\dots)}{\Delta} = \inf_{\Delta > 0} \left(\frac{1}{\Delta} + n\right) m = n \cdot m$$

For this system,

$$\inf_{\Delta > 0} \frac{\tau[\Delta](t)}{\Delta} = \mathrm{dh}_R(t)$$

in fact holds *for every term t* (exercise): *exact bounds*

Example $(x \circ y) \circ z \to x \circ (y \circ z)$

Traditionally,

$$\circ_\tau(n_1, n_2) = 2n_1 + n_2 + 1$$

By the same *creative step* as above guess

$$\circ_\tau[\Delta](z_1, z_2) = (1 + \Delta)z_1 + z_2 + 1$$

Solving the Δ-monotonicity constraints yields

$$\tau[\Delta](s \circ t) = (1 + \Delta) \cdot \tau\left[\frac{\Delta}{1 + \Delta}\right](s) + \tau[\Delta](t) + 1$$

Remark: proof of $\tau[\Delta](\ell\gamma) - \tau[\Delta](r\gamma) \geq \Delta$ uses induction.

$(x \circ y) \circ z \longrightarrow x \circ (y \circ z)$ (cont'd)

Again *for every term* t (exercise)

$$\boxed{\inf_{\Delta > 0} \frac{\tau[\Delta](t)}{\Delta} = \mathrm{dh}_R(t)}$$

- Expl: For the *"left comb"* ℓ of depth n

$$\tau[\Delta](\ell) = n + \Delta n(n-1)/2$$

thus $\mathrm{dh}_R(\ell) \leq \inf_{\Delta>0} \tau[\Delta](\ell)/\Delta = \boxed{n(n-1)/2}$

- Expl: For the *"right comb"* r of depth n

$$\tau[\Delta](r) = n$$

thus $\mathrm{dh}_R(r) \leq \inf_{\Delta>0} \tau[\Delta](r)/\Delta = \boxed{0}$

Monotonicity revisited

Strict monotonicity

$$m > n \quad \text{implies} \quad f_{\mathcal{T}}(\ldots m \ldots) > f_{\mathcal{T}}(\ldots n \ldots)$$

is (over $\mathbb{N}$) equivalent to

$$m - n \geq 1 \quad \text{implies} \quad f_{\mathcal{T}}(\ldots m \ldots) - f_{\mathcal{T}}(\ldots n \ldots) \geq 1$$

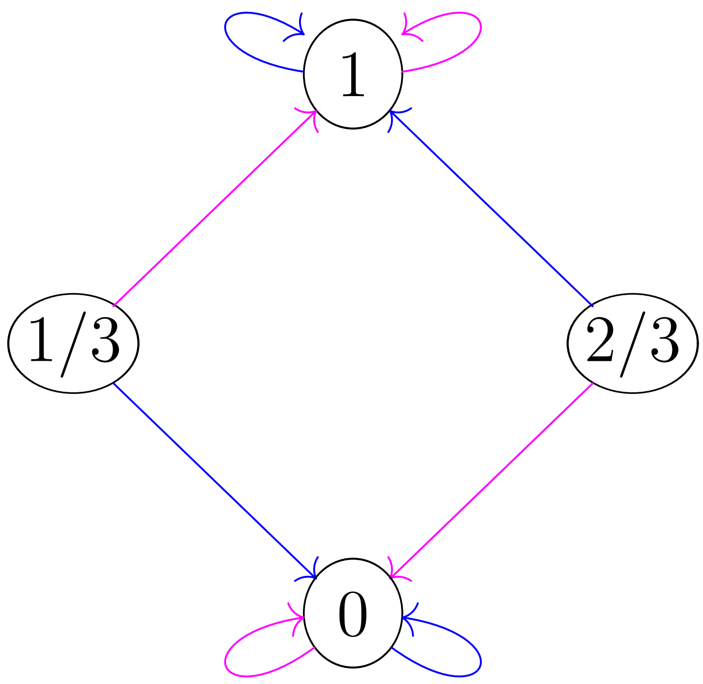
thus equivalent to strict monotonicity of $>_1$, where

$$\boxed{m >_1 n \quad \text{iff} \quad m - n \geq 1}$$

- $>_1$ is total on $\mathbb{N}$
- $>_1$ is not total on $\mathbb{Q}_0^+$ (but well-founded)

Expl $g(a) \to g(b),\ f(b) \to f(a)$

- No interpretation into $\mathbb{N}$ with $\tau(\ell) >_1 \tau(r)$ and strict monotonicity modulo $>_1$ exists (why?)
- It *does exist* into $(\mathbb{Q}_0^+, >_1)$, even into a finite subset:



$a_\tau = 1/3$
$b_\tau = 2/3$
f_τ
g_τ

- Exercise: verify $\tau(\ell) >_1 \tau(r)$; strict monotonicity of $>_1$

Expl $ffx \to fgf$

- Not simply terminating
- An interpretation into $(\mathbb{Q}_0^+, >_1)$ exists:

$$f_\mathcal{T}(z) = n + 1/2 \qquad \text{if } n - 1 < z \leq n$$

$$g_\mathcal{T}(z) = n \qquad \text{if } n - 1/2 < z \leq n + 1/2$$

- The resulting (linear) upper bound

$$\boxed{\mathrm{dh}(t) = \lfloor \tau(t) \rfloor}$$

is exact (exercise).

Context-dependent Int's: Remarks

- Even if *exact* bounds are not achievable, improved bounds can be derived.
- Proving that bounds are exact: typically needs knowledge about optimal / worst case rewrite strategies.
- Top-down propagation of Δ versus bottom-up term evaluation: two-phase transducer.
- Here: *weak* context-dependency. Only a non-local *strong* version would deserve to be called *context-sensitive*.
- Implementation
  - Non-trivial calculations $\leadsto$ computer algebra?
  - Inductive proofs $\leadsto$ theorem prover?
  - Work by [Schnabl/Moser]: `cdiprover3` $\leadsto$ demo

Relative Termination

Let $S = \{ab \to baa\}$, $R = \{cb \to bbc\}$.
Consider R-steps in $R \cup S$-derivations.

The interpretation $\Sigma \to (\mathbb{N} \to \mathbb{N})$ with

$$a \mapsto \lambda n.n \qquad b \mapsto \lambda n.n+1 \qquad c \mapsto \lambda n.3n$$

is $\boxed{\text{constant}}$ for S and $\boxed{\text{decreasing}}$ for R
$\Rightarrow$ number of R-steps is $2^{O(n)}$.

Relative Termination

Let $S = \{ab \to baa\}$, $R = \{cb \to bbc\}$.
Consider R-steps in $R \cup S$-derivations.

The interpretation $\Sigma \to (\mathbb{N} \to \mathbb{N})$ with

$$a \mapsto \lambda n.n \qquad b \mapsto \lambda n.n + 1 \qquad c \mapsto \lambda n.3n$$

is constant for S and decreasing for R
$\Rightarrow$ number of R-steps is $2^{O(n)}$.

Relative termination allows to remove rules successively $\rightsquigarrow$

- Modular termination proofs
- Automatic methods for proving relative termination are incorporated in all state of the art termination provers.
- $\rightsquigarrow$ Annual termination competition [WST]

The Problem

Let R and S be rewriting systems.
Assume termination of $R \cup S$ has been shown
by proving termination of R/S and termination of S.

- Give a bound on $\mathrm{dc}_{R \cup S}$ in terms of $\mathrm{dc}_{R/S}$ and dc_S.

Note: Proof methods for relative termination
can handle situations where S is not terminating.
Here we assume that S *is* terminating.

Basic Observation

Let $\Delta_R = \max\{|r| \dot{-} |\ell| \mid (\ell \to r) \in R\}$, and assume (for simplicity) that this implies $\max\{|x| \dot{-} |y| \mid x \to_R y\} \leq \Delta_R$.

- Note: $\Delta_R = 0$ in case R is not size-increasing.

Now consider an arbitrary finite derivation modulo $R \cup S$:

$$x_0 \to_S^* x'_0 \to_R x_1 \to_S^* x'_1 \to_R x_2 \to_S^* \cdots \to_S^* x'_{k-1} \to_R x_k \to_S^* x'_k$$

Define $\delta : \mathbb{N} \to \mathbb{N}$ by $\delta(n) = n + \Delta_S \cdot \mathrm{dc}_S(n) + \Delta_R$. Then

$$|x_{i+1}| \leq \delta(|x_i|).$$

Monotonicity of dc_S implies monotonicity of δ, thus

$$|x_{i+1}| \leq \delta^i(|x_0|).$$

The General Upper Bound

$$x_0 \to_S^* x_0' \to_R x_1 \to_S^* x_1' \to_R x_2 \to_S^* \cdots \to_S^* x_{k-1}' \to_R x_k \to_S^* x_k'$$

...thus the length of the above derivation is bounded by

$$\begin{aligned} \mathrm{dc}_{R\cup S}(|x_0|) &\leq \mathrm{dc}_{R/S}(|x_0|) + \sum_{i=0}^{k} \mathrm{dc}_S(|x_i|) \\ &\leq \mathrm{dc}_{R/S}(|x_0|) + \sum_{i=0}^{k} \mathrm{dc}_S\left(\delta^i(|x_0|)\right) \end{aligned}$$

We have $\delta^{i+1}(n) \geq \delta^i(n)$ by $\delta(n) \geq n$. Since $k \leq \mathrm{dc}_{R/S}(|x_0|)$,

$$\mathrm{dc}_{R\cup S}(n) \in O\Big(\mathrm{dc}_{R/S}(n) \cdot \mathrm{dc}_S\left(\delta^{\mathrm{dc}_{R/S}(n)}(n)\right)\Big)$$

Particular Cases

- R and S not size-increasing: $\delta(n) = n$

$$\boxed{\mathrm{dc}_{R \cup S}(n) \in O\big(\mathrm{dc}_{R/S}(n) \cdot \mathrm{dc}_{S}\,(n)\big)}$$

Multiplication

Particular Cases

- R and S not size-increasing: $\delta(n) = n$

  $$\boxed{\mathrm{dc}_{R\cup S}(n) \in O\big(\mathrm{dc}_{R/S}(n) \cdot \mathrm{dc}_S\,(n)\big)}$$

  Multiplication

- S not size-increasing:

  $\delta(n) = n + \Delta_R$, thus $\delta^i(n) = n + i \cdot \Delta_R$

  $$\boxed{\mathrm{dc}_{R\cup S}(n) \in O\big(\mathrm{dc}_{R/S}(n) \cdot \mathrm{dc}_S\,(n + \mathrm{dc}_{R/S}(n) \cdot \Delta_R)\big)}$$

  Composition

Particular Cases

- R and S not size-increasing: $\delta(n) = n$

$$\boxed{\mathrm{dc}_{R \cup S}(n) \in O\big(\mathrm{dc}_{R/S}(n) \cdot \mathrm{dc}_S\,(n)\big)}$$

Multiplication

- S not size-increasing:
  $\delta(n) = n + \Delta_R$, thus $\delta^i(n) = n + i \cdot \Delta_R$

$$\boxed{\mathrm{dc}_{R \cup S}(n) \in O\big(\mathrm{dc}_{R/S}(n) \cdot \mathrm{dc}_S\,(n + \mathrm{dc}_{R/S}(n) \cdot \Delta_R)\big)}$$

Composition

- S size-increasing: $\delta \in \Theta(\mathrm{dc}_S)$

$$\boxed{\mathrm{dc}_{R \cup S}(n) \in O\big(\mathrm{dc}_{R/S}(n) \cdot \mathrm{dc}_S^{\mathrm{dc}_{R/S}(n)+1}(n)\big)}$$

Iteration

Consequences

- Consider function classes with certain closure properties:
  - Closed under addition, multiplication, composition
    Example: polynomials
  - Closed under iteration
    Example: primitive recursive functions

Consequences

- Consider function classes with certain closure properties:
  - Closed under addition, multiplication, composition
    Example: polynomials
  - Closed under iteration
    Example: primitive recursive functions
- Can this general bound be improved?
  No, as the following generic construction reveals.
  (For string rewriting, therefore can be done in every sufficiently rich rewriting model.)

The Lower Bound Result

The general upper bound can be attained, even for string rewriting. Proof:

Take arbitrary string rewriting systems R_0 over Σ, S_0 over Γ (w.l.o.g. disjoint alphabets) and add new letters σ, γ. Define

$$R = \{l \to r\sigma \mid (l \to r) \in R_0\} \quad \text{(introduce marker)}$$

$$S = S_0 \cup \{\sigma a \to a\sigma \mid a \in \Sigma\} \quad \text{(move marker)}$$

$$\cup \{\sigma \to \gamma\} \quad \text{(switch markers)}$$

$$\cup \{\gamma b \to c\gamma \mid b, c \in \Gamma\} \quad \text{(nondeterministic reset)}$$

We have $\boxed{\mathrm{dc}_{R_0} \approx \mathrm{dc}_{R/S}}$, $\boxed{\mathrm{dc}_{S_0} + \Theta(n^2) \approx \mathrm{dc}_S}$ and

$\boxed{\mathrm{dc}_{R \cup S} = \Theta(\text{upper bound in terms of } \mathrm{dc}_{R/S} \text{ and } \mathrm{dc}_S).}$

So the construction shows optimality if $\mathrm{dc}_S \in \Omega(n^2)$.

Example: Polynomial Upper Bound

$$B_k = \{ki \to jk \mid k > i, j\}$$

$$R_d = B_2 \cup \cdots \cup B_d$$

over alphabet $\{1, 2, \ldots, d\}$. The bound $\boxed{\mathrm{dc}_{R_d} \in \Theta(n^d)}$ can be shown via some matrix interpretation of dimension $d + 1$.

Example: Polynomial Upper Bound

$$B_k = \{ki \to jk \mid k > i, j\}$$

$$R_d = B_2 \cup \cdots \cup B_d$$

over alphabet $\{1, 2, \ldots, d\}$. The bound $\boxed{\mathrm{dc}_{R_d} \in \Theta(n^d)}$ can be shown via some matrix interpretation of dimension $d + 1$.

A simpler proof via relative termination:

- Show $\mathrm{SN}(B_d / R_{d-1})$ via the interpretation
  $$\{1, \ldots, d-1\} \mapsto \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}, \qquad d \mapsto \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$
- $\mathrm{dc}_{B_d / R_{d-1}} \in O(n^2)$ (matrices are upper triangular)
- B_d and R_{d-1} are size-preserving, so the upper bound result implies (by induction) $\boxed{\mathrm{dc}_{R_d} \in O(n^{2(d-1)})}$.

Bound is overestimated, but nevertheless polynomial.
Termination proof much easier to find.

Discussion

- Can the general upper bound be reached for $\mathrm{dc}_S \in O(n)$?

Discussion

- Can the general upper bound be reached for $\mathrm{dc}_S \in O(n)$? Yes for term rewriting:

$$R = \{f(s(x), y, z) \to f(x, z, y) \mid x, y, z \geq 0\}$$

$$S = \{f(x, s(y), z) \to f(x, y, s(s(z))) \mid x, y, z \geq 0\}$$

Here, $\mathrm{dc}_{R/S} \in O(n)$ and $\mathrm{dc}_S \in O(n)$, but $\mathrm{dc}_{R \cup S}$ is exponential: $f(s^n(0), 1, 0) \to^* f(0, 0, s^{2^n}(0))$.

Discussion

- Can the general upper bound be reached for $\mathrm{dc}_S \in O(n)$? Yes for term rewriting:

$$R = \{f(s(x), y, z) \to f(x, z, y) \mid x, y, z \geq 0\}$$

$$S = \{f(x, s(y), z) \to f(x, y, s(s(z))) \mid x, y, z \geq 0\}$$

  Here, $\mathrm{dc}_{R/S} \in O(n)$ and $\mathrm{dc}_S \in O(n)$, but $\mathrm{dc}_{R \cup S}$ is exponential: $f(s^n(0), 1, 0) \to^* f(0, 0, s^{2^n}(0))$.

- Remark: Similarly with binary symbol f.
  Exercise: How about unary symbols only, i.e. for string rewriting?

Discussion

- Can the general upper bound be reached for $\mathrm{dc}_S \in O(n)$? Yes for term rewriting:

$$R = \{f(s(x), y, z) \to f(x, z, y) \mid x, y, z \geq 0\}$$
$$S = \{f(x, s(y), z) \to f(x, y, s(s(z))) \mid x, y, z \geq 0\}$$

  Here, $\mathrm{dc}_{R/S} \in O(n)$ and $\mathrm{dc}_S \in O(n)$, but $\mathrm{dc}_{R \cup S}$ is exponential: $f(s^n(0), 1, 0) \to^* f(0, 0, s^{2^n}(0))$.

- Remark: Similarly with binary symbol f.
  Exercise: How about unary symbols only,
  i.e. for string rewriting?

- Make the implicit notion of "abstract reduction system with size measure" explicit.

Acknowledgements

Thanks for generously sharing ideas, exercises, slides . . .
to Alfons Geser, Jörg Endrullis, Johannes Waldmann,
Hans Zantema, and many others

. . . and thanks for patiently listening.