Automatic Termination

Johannes Waldmann, HTWK Leipzig

Inference Seminar, FSU Jena, Oct. 2023

Termination: Definition

- string rewrite system (a.k.a. Turing machine, Markov algorithm, type 0 grammar) is set of rules $R \subseteq \Sigma^* \times \Sigma^*$, e.g., $R = \{(ab, ba)\}$ over $\Sigma = \{a, b\}$
- defines relation $\to_R$ on Σ^* by context closure $(\to_R) = \{(plq, prq) \mid p \in \Sigma^*, (l, r) \in R, q \in \Sigma^*\}$
  $\underset{p}{a}\,\underset{l}{ab}\,\underset{q}{b} \to_R \underset{p}{a}\,\underset{r}{ba}\,\underset{q}{b} \to_R abba \to_R baba \to_R bbaa$
- system R is terminating (strongly normalizing, SN(R)) iff no $w \in \Sigma^*$ starts an infinite $\to_R$-chain.
- Termination (= uniform halting problem) is not decidable. we want semi-algorithms for SN(R) and for $\neg$SN(R)
- this talk: on string rewriting, methods can be (and have been) generalized to term rewriting (functional programs)

Termination: History (brief and imcomplete)

- Alan M. Turing: *Checking a Large Routine*, 1949. "[using] a quantity which is asserted to decrease continually and vanish when the machine stops"
- Donald E. Knuth and Peter B. Bendix: *Simple Word Problems in Universal Algebras*, 1970. "a well-ordering on the set of all words such that each right-hand side of an [equation] represents a word smaller [...] that the left-hand side"
- Sam Kamin and Jean-Jacques Levy *Attempts for generalizing the recursive path orderings*, 1980. `http://perso.ens-lyon.fr/pierre.lescanne/not_accessible.html#termination`
- Workshop on Termination (St. Andrews 1993, ... Leipzig 2025), Termination Competition (2004 ...), `https://termination-portal.org/`

Termination: Basic Method: Counting

- $\{a \to b\}$: number of a is reduced
- $\{aa \to bbb, bb \to a\}$: $5|w|_a + 3|w|_b$ is reduced
- $\{aa \to aba\}$: number of blocks $\ldots aa \ldots$ is reduced
- $\{ab \to ba\}$: (bubble sort) number of inversions $(\ldots a \ldots b \ldots)$ is reduced
- $\{ab \to bba\}$? number of a stays put, of b goes up, exponentially: $\forall k \in \mathbb{N} : ab^k \to^k b^{2k} a$, $\forall k \in \mathbb{N} : a^k b \to^{2^k - 1} b^{2^k} a^k$,
- $\{aabb \to bbbaaa\}$? (Hans Zantema, 1990?) (solved by Alfons Geser 1993)
- $\{aa \to bc, bb \to ac, cc \to ab\}$? (Hans Zantema, 2003?) (solved by Dieter Hofbauer and J.W., 2005)

Termination: Basic Method: Interpretation

- ... into well-founded monotone algebra A
  - non-empty domain D_A, with well-founded relation $>_A$
  - for each letter $c \in \Sigma$, a function $c_A : (D_A \to D_A)$ such that $\forall x, y \in D_A : x >_A y \Rightarrow c_A(x) >_A c_A(y)$
- ... is *compatible with* R if $\forall (l, r) \in R, x \in D_A : l_A(x) >_A r_A(x)$ Thm. [folklore] such A exists $\Leftrightarrow$ SN(R).
- Ex. for $R = \{ab \to ba\}$, use $D_A = \mathbb{N}$ with usual $>$-relation, $a_A(x) = 2x$, $b_A(x) = x + 1$ $l_A(x) = a_A(b_A(x)) = 2(x + 1) > r_A(x) = b_A(a_A(x)) = 2x + 1$
- Ex. for $R = \{ab \to bba\}$, use $D_A = \mathbb{N}$ with usual $>$-relation, $a_A(x) = 3x$, $b_A(x) = x + 1$ gives exponential upper bound on derivation lengths
- $\{aa \to aba\}$? $\{aabb \to bbbaaa\}$?

Non-Termination

- cycle: $u \to_R^+ u$, ex. $R = \{a \to b, b \to a\}$ has cycle $a \to^2 a$,
- $\{0000 \to 0111, 1001 \to 0010\}$ (Andreas Gebhardt, 2006) has (shortest?) cycle of length 80, width 21 (Dieter Hofbauer: KnockedForLoops, 2010)
- loop: $u \to_R^+ puq$ ex. $R = \{ab \to bbaa\}$ has loop $abb \to bbaab \to bb\underline{abb}aa$ then $abb \to^2 bb\, abb\, aa \to^2 bb\, bb\, abb\, aa\, aa \to^2 \ldots$
- non-looping non-termination (must exist) $\{bc \to dc, bd \to db, ad \to abb\}$ (Nachum Dershowitz 1987) $ab^k c \to ab^{k-1} dc \to^* adb^{k-1} c \to abbb^{k-1} c = ab^{k+1} c \to^+ \ldots$ with two rules: Alfons Geser and Hans Zantema 1999, with one rule: open

Termination: Examples (Homework)

of these three systems

- $\{ba \to acb, bc \to abb\}$
- $\{ba \to acb, bc \to cbb\}$
- $\{ba \to aab, bc \to cbb\}$

can you tell which is

- non-terminating,
- terminating, with derivation lengths in $\exp(\exp(n))$
- ... multiply exponential ...

Example: $\{ba \to acb, bc \to abb\}$

- has a loop, and a nice method of describing it without writing down all steps:
- for the morphism $\phi : a \mapsto ac, b \mapsto b, c \mapsto ab$,
  - $\forall x \in \Sigma : bx \to^* \phi(x)b$,
  - iteration (D0L system): $\forall w \in \Sigma^* : b^k w \to^* \phi^k(w) b^k$
  - $a \overset{\phi}{\to} ac \overset{\phi}{\to} acab \overset{\phi}{\to} acabacb \overset{\phi}{\to} acabacbacabb \overset{\phi}{\to} \ldots$ number of occurences of b before rightmost a in $\phi^k(a)$ is $\geq \text{Fib}(k-1) \in 2^{\Omega(k)}$
  - exists k: $\phi^k(a) = va \ldots$ with $|v|_b \geq k$
  - $b^k a \to^* b^k \phi^k(a) = \ldots va \ldots \to^* \ldots b^k a \cdots \to \ldots$
- allows to compress "loops of super-exponential length" (Alfons Geser 2002) down to small (linear) certificate

Example: $\{ba \to aab, bc \to cbb\}$

- we have $b^k a \to^* a^{2^k} b^k$ (renaming of $ab \to bba$) and also $bc^k \to^* c^k b^{2^k}$ (rename and mirror image)
- combined: $bc^k a \to^* c^k b^{2^k} a \to^* c^k a^{2^{2^k}} b^{2^k}$ number of steps is $\Theta(2^{2^k})$ by comparing lengths
- this is one derivation, can it be worse? nonterminating? no compatible monotone linear interpretation on $\mathbb{N}$, since this gives singly exponential derivation lengths
- prove termination via two interpretations:
  - $a_1(x) = x, b_1(x) = 3x, c_1(x) = x + 1$
  - $a_2(x) = x + 1, b_2(x) = 3x, c_2(x) = x$

  then $w \mapsto (\overline{w}_1(0), w_2(0))$ is lexicographically decreasing
- this is an instance of *relative termination* (Geser 1990)

Example: $\{ba \to acb, bc \to cbb\}$

- as before, $bc^k \to^* c^k b^{2^k}$, also, $b^k a \to a(cb)^k \to^+ ac^k b^{2^{\Theta(k)}}$ then ba^k starts tower-of-exp length derivation
- termination proof: make blocks $\in \{b, c\}^*$ separated by a, linear interpretation in each block, combine lexicographically
- of course there must be terminating systems with (uncomputably) long derivations. Else, we could decide TM halting for fixed input.
- the observation here is that we can get long derivations from small systems already
- that's not too much of a surprise, cf. Busy Beaver TMs (survey: Heiner Marxen, Jürgen Buntrock, 1990)
- systematic enumeration of small (one-rule!) hard (for termination) string rewrite systems: Winfried Kurth 1990, Alfons Geser 2004, Mario Wenzel 2016

Matrix Interpretations (Motivation)

- recall wfmA for $R = \{ab \to ba\}$: $D_A = \mathbb{N}$ with usual $>$-relation, $a_A(x) = 2x$, $b_A(x) = x + 1$ $l_A(x) = a_A(b_A(x)) = 2(x + 1) > r_A(x) = b_A(a_A(x)) = 2x + 1$
- now write linear function $x \mapsto cx + d$ as matrix $\begin{pmatrix} c & d \\ 0 & 1 \end{pmatrix}$,

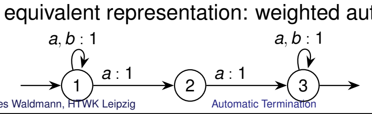
  $a_A = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}$ $b_A = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$ $[ab]_A = \begin{pmatrix} 2 & 2 \\ 0 & 1 \end{pmatrix}$ $[ba]_A = \begin{pmatrix} 2 & 1 \\ 0 & 1 \end{pmatrix}$
- matrices operate on domain $\mathbb{N}^2$ (column vectors) ordered by $\vec{x} > \vec{y}$ iff $x_1 > y_1 (\wedge x_2 = y_2)$
- generalize to larger dimensions! need suitable domain and order ($\Rightarrow$ monotonicity, compatibility) (Dieter Hofbauer, JW, Jörg Endrullis, Hans Zantema, 2006)

Matrix Interpretations (Realization)

- A d-dimensional matrix interpretation has domain $\mathbb{N}^{d-1} \times \mathbb{N}_+$, ordered by $\vec{x} > \vec{y}$ iff $x_1 > y_1 \wedge \forall i > 1 : x_i \geq y_i$ and for each letter $c \in \Sigma$, a square matrix $[c] \in \mathbb{N}^{d \times d}$ with $[c]_{1,1} \geq 1, [c]_{d,d} \geq 1$ (then $[c]$ is monotone) and is compatible with R if for each $(l, r) \in R$, $[l] \geq [r]$ (point-wise everywhere) and $[l]_{1,d} > [r]_{1,d}$ (top right)
- example: SN($aa \to aba$) by interpretation:

  $[a] = \begin{pmatrix} \boxed{1} & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & \boxed{1} \end{pmatrix}$ $[b] = \begin{pmatrix} \boxed{1} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \boxed{1} \end{pmatrix}$ $[aa] = \begin{pmatrix} 1 & 1 & \boxed{1} \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}$ $[aba] = \begin{pmatrix} 1 & 1 & \boxed{0} \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}$

  it counts number of aa blocks
- equivalent representation: weighted automaton

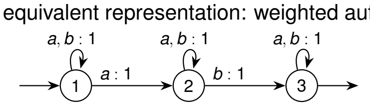


Matrix Interpretations (Another Example)

- ...domain $\mathbb{N}^{d-1} \times \mathbb{N}_+$, $\vec{x} > \vec{y}$ iff $x_1 > y_1 \wedge \forall i > 1 : x_i \geq y_i$, and for each letter $c \in \Sigma$, a square matrix $[c] \in \mathbb{N}^{d \times d}$ with $[c]_{1,1} \geq 1, [c]_{d,d} \geq 1$ (then $[c]$ is monotone) and is compatible with R if for each $(l, r) \in R$, $[l] \geq [r]$ (point-wise everywhere) and $[l]_{1,d} > [r]_{1,d}$ (top right)
- example: SN($ab \to ba$) by interpretation:

  $[a] = \begin{pmatrix} \boxed{1} & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \boxed{1} \end{pmatrix}$ $[b] = \begin{pmatrix} \boxed{1} & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & \boxed{1} \end{pmatrix}$ $[ab] = \begin{pmatrix} 1 & 1 & \boxed{1} \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$ $[ba] = \begin{pmatrix} 1 & 1 & \boxed{0} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

  it counts number of inversions $\ldots a \ldots b \ldots$
- equivalent representation: weighted automaton



Matrix Interpretations: The Killer Example

- SN($\{aa \to bc, bb \to ac, cc \to ab\}$)
-

```
a =           b =           c =
/ 1 2 0 0 0 \ / 1 0 0 1 0 \ / 1 1 0 0 0 \
| 0 0 0 0 0 | | 0 0 0 0 0 | | 0 0 0 1 0 |
| 0 2 0 1 1 | | 0 0 0 1 0 | | 0 2 0 1 2 |
| 0 1 1 0 0 | | 0 1 0 2 1 | | 0 1 0 0 0 |
\ 0 0 0 0 1 / \ 0 0 0 0 1 / \ 0 0 0 0 1 /
b b =         a c =
/ 1 1 0 3 1 \ / 1 1 0 2 0 \
| 0 0 0 0 0 | | 0 0 0 0 0 |
| 0 1 0 2 1 | | 0 1 0 2 1 |
| 0 2 0 4 3 | | 0 2 0 2 2 |
\ 0 0 0 0 1 / \ 0 0 0 0 1 /
```

- found in 2005, termination problem was open until then

Matrix Interpretations: Properties, Extensions

- exponential upper bound on derivation lengths. combination with other termination proof methods (e.g., lexicographic) can lift this bound
- for restricted shape (e.g., upper triangular): polynomial upper bound (JW. RTA 2010)
  - recall 3D-Int. for $ab \to ba$ (quadratic), $aa \to aba$ (linear)
  - challenge: find polynomially bounded matrices for $\{aa \to bc, bb \to ac, cc \to ab\}$ Sergei Adian gave manual proof for quadratic bound for derivation lengths
- extensions of matrix interpretations:
  - matrices over other domains (ordered semi-rings), e.g., arctic $(\{-\infty\} \cup \mathbb{N}, \max, +, -\infty, 0)$
  - weaker monotonicity

How We Find Matrices

- for dimension d, $[l] > [r]$ is system of d^2 inequalities between polynomials (in $|\Sigma| \cdot d^2$ unknown entries of $[c]$)
- solvability over $\mathbb{N}$ is undecidable (Hilbert 10), over $\mathbb{R}$ is hard (Tarski, QEPCAD)
- since the method is incomplete for termination anyway, we don't need a complete solver, but a powerful semi-algorithm
- D. Hofbauer, MultumNonMulta: incrementally add paths to automaton = increase entries of matrices: aha, gradient descent! — with a provision for vanishing gradient
- JW.: Matchbox: solve constraints by bit-blasting: fix (small) bit width b, represent numbers in binary, realize arithemtical operations as boolean circuits, use SAT solver

How We Find Matrices: Completion

- D. Hofbauer, MultumNonMulta:
  incrementally modify/add paths to automaton = increase entries of matrices: aha, gradient descent!
- when gradient vanishes: use higher derivatives (= increase weights along a longer path in the automaton)
- works best for: sparse matrices, can be large, e.g.,

```
Removing 1 rule by a matrix interpretation [Hofbauer/Waldmann, RTA 2006]
  of type E_J with J = {1,...,2} and dimension 14:

0 -> /                             \
     | 1 0 1 0 0 0 0 0 1 0 0 0 0 0 |
     | 0 1 0 0 0 0 0 0 0 0 0 0 0 0 |
     | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
     | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
     | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
     | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
...
```

`https://www.starexec.org/starexec/services/jobs/pairs/615286154/stdout/1?limit=-1`

How We Find Matrices: Bit Blasting

- JW.: Matchbox: solve constraints by (eager) *bit-blasting*:
  - fix (small) bit width b,
  - represent numbers in unary or binary,
  - realize arithemtical operations as boolean circuits,
  - Tseitin-transform to CNF (using `ersatz` eDSL/library, Edward Kmett et al. 2010–)
  - get satisfying assignment from *minisat* (Niklas Een, Niklas Sörensson, 2003–), *kissat* (Armin Biere, 2020–)
- for killer example: $d = 5$, $b = 3$, unary,
  CNF with 3.857 vars, 17.454 clauses,
  satisfying assignment found by kissat in < 1 second
- works best for: small matrices (can be dense), e.g.,
  `https://www.starexec.org/starexec/services/jobs/pairs/615266392/stdout/1?limit=-1`

Summary, Discussion, Challenges

- matrix intepretation: an instance of well-founded monotone algebras, have become a standard method in automated termination, shows that SAT solvers are highly useful
- . . . do we really want this? it assumes/supports the "first write a program, then guess why it works" amateur-hour style of programming— instead of "use a language with a type system that only allows total (terminating) programs"
- . . . like Agda, but even Agda has built-in automated termination ("smaller-subterm" criterion for recursive calls)
- we want *some* type inference (avoid writing trivial types)
- we want not just termination but bounds on derivation lengths (cost of computation) (could be part of the type)
- terminating? $\{0000 \to 1011, 1001 \to 0100\}$ (Gebhardt/20)
- is termination decidable for one-rule string rewriting?