A Logical Characterization of Constant-Depth Arithmetic Circuits over the Reals

Timon Barlag and Heribert Vollmer

Introduction

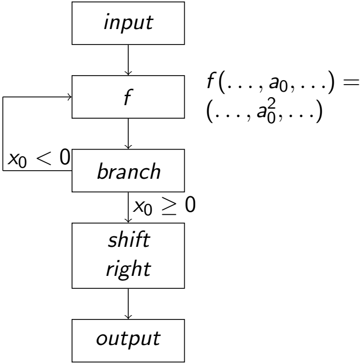
- Real computation
  - $\mathbb{R}$-machines
- Parallel computation
  - Arithmetic circuits (over $\mathbb{R}$)
    - $AC^0_{(\mathbb{R})}$
- Logics & Descriptive complexity
  - $AC^0 = FO$ [Im89]

$\mathbb{R}$-machines

	x_{-1}	x_0	x_1	x_2	x_3	x_4	x_5	
$\cdots$	0	π	$-e$	5	2.3	-9	0	$\cdots$

$\mathbb{R}$-machines

- unbounded tape of $\mathbb{R}$-registers
- associated graph with node types:
  - input
  - output
  - computation
  - branch
  - shift

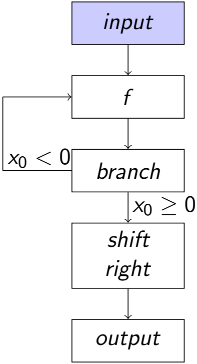


$\mathbb{R}$-machines

$\cdots$	x_{-1}	x_0	x_1	x_2	x_3	x_4	x_5	$\cdots$
$\cdots$	0	π	$-e$	5	2.3	-9	0	$\cdots$

$\mathbb{R}$-machines

- unbounded tape of $\mathbb{R}$-registers
- associated graph with node types:
  - input
  - output
  - computation
  - branch
  - shift



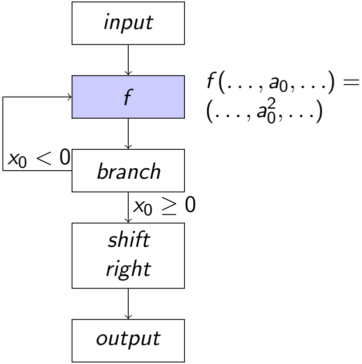
$f(\ldots, a_0, \ldots) = (\ldots, a_0^2, \ldots)$

$\mathbb{R}$-machines

	x_{-1}	x_0	x_1	x_2	x_3	x_4	x_5	
$\cdots$	0	π	$-e$	5	2.3	-9	0	$\cdots$

$\mathbb{R}$-machines

- unbounded tape of $\mathbb{R}$-registers
- associated graph with node types:
  - input
  - output
  - computation
  - branch
  - shift

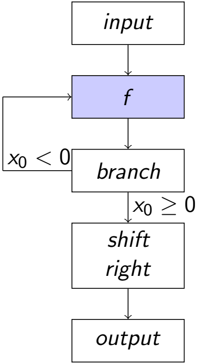


$\mathbb{R}$-machines

	x_{-1}	x_0	x_1	x_2	x_3	x_4	x_5	
$\cdots$	0	π^2	$-e$	5	2.3	-9	0	$\cdots$

$\mathbb{R}$-machines

- unbounded tape of $\mathbb{R}$-registers
- associated graph with node types:
  - input
  - output
  - computation
  - branch
  - shift



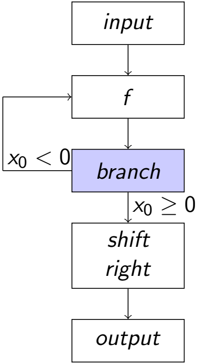
$f(\ldots, a_0, \ldots) = (\ldots, a_0^2, \ldots)$

$\mathbb{R}$-machines

	x_{-1}	x_0	x_1	x_2	x_3	x_4	x_5	
$\cdots$	0	π^2	$-e$	5	2.3	-9	0	$\cdots$

$\mathbb{R}$-machines

- unbounded tape of $\mathbb{R}$-registers
- associated graph with node types:
  - input
  - output
  - computation
  - branch
  - shift



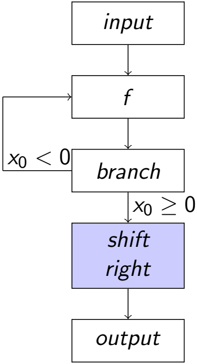
$f(\ldots, a_0, \ldots) = (\ldots, a_0^2, \ldots)$

$\mathbb{R}$-machines

	x_{-1}	x_0	x_1	x_2	x_3	x_4	x_5	
$\cdots$	0	π^2	$-e$	5	2.3	-9	0	$\cdots$

$\mathbb{R}$-machines

- unbounded tape of $\mathbb{R}$-registers
- associated graph with node types:
  - input
  - output
  - computation
  - branch
  - shift



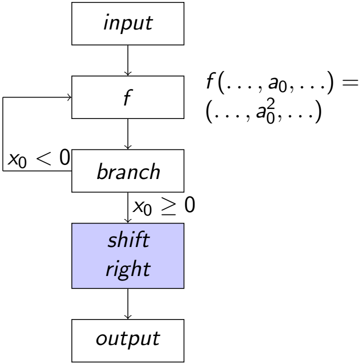
$f(\ldots, a_0, \ldots) = (\ldots, a_0^2, \ldots)$

ℝ-machines

	x_{-1}	x_0	x_1	x_2	x_3	x_4	x_5	
$\cdots$	0	0	π^2	$-e$	5	2.3	-9	$\cdots$

ℝ-machines

- unbounded tape of ℝ-registers
- associated graph with node types:
  - input
  - output
  - computation
  - branch
  - shift

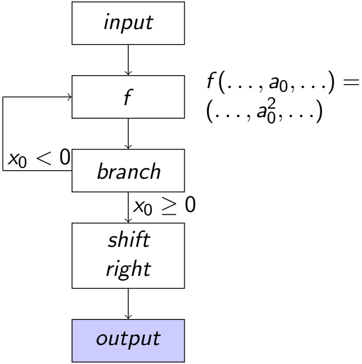


$\mathbb{R}$-machines

	x_{-1}	x_0	x_1	x_2	x_3	x_4	x_5	
$\cdots$	0	0	π^2	$-e$	5	2.3	-9	$\cdots$

$\mathbb{R}$-machines

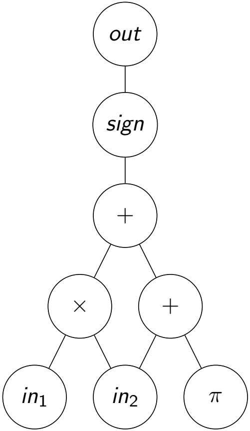
- unbounded tape of $\mathbb{R}$-registers
- associated graph with node types:
  - input
  - output
  - computation
  - branch
  - shift



Arithmetic Circuits over $\mathbb{R}$

Arithmetic Circuits over $\mathbb{R}$

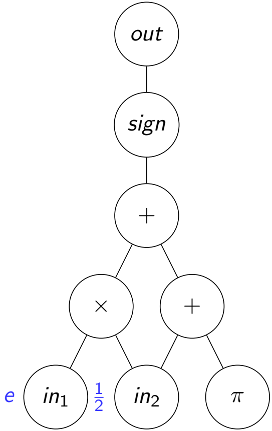
- DAG with node types:
  - input (fan-in 0)
  - constant (fan-in 0)
  - arithmetic (fan-in ≥ 0)
  - sign (fan-in 1)
  - output (fan-in 1)



Arithmetic Circuits over $\mathbb{R}$

Arithmetic Circuits over $\mathbb{R}$

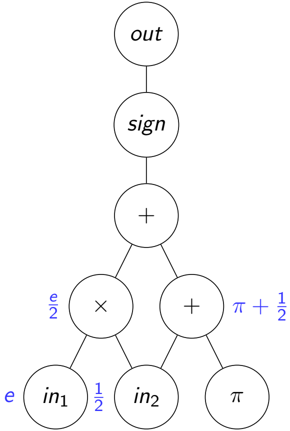
- DAG with node types:
  - input (fan-in 0)
  - constant (fan-in 0)
  - arithmetic (fan-in ≥ 0)
  - sign (fan-in 1)
  - output (fan-in 1)



Arithmetic Circuits over $\mathbb{R}$

Arithmetic Circuits over $\mathbb{R}$

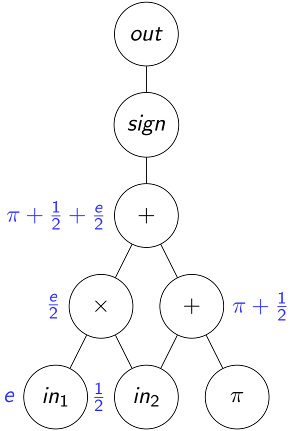
- DAG with node types:
  - input (fan-in 0)
  - constant (fan-in 0)
  - arithmetic (fan-in ≥ 0)
  - sign (fan-in 1)
  - output (fan-in 1)



Arithmetic Circuits over $\mathbb{R}$

Arithmetic Circuits over $\mathbb{R}$

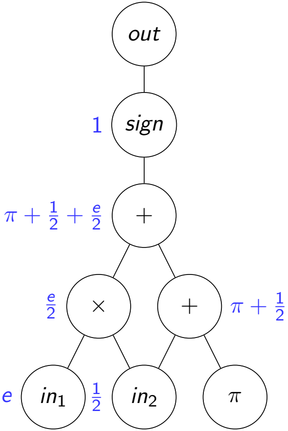
- DAG with node types:
  - input (fan-in 0)
  - constant (fan-in 0)
  - arithmetic (fan-in ≥ 0)
  - sign (fan-in 1)
  - output (fan-in 1)



Arithmetic Circuits over $\mathbb{R}$

Arithmetic Circuits over $\mathbb{R}$

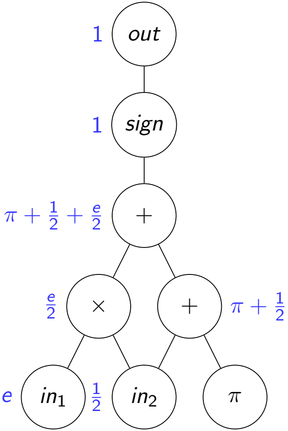
- DAG with node types:
  - input (fan-in 0)
  - constant (fan-in 0)
  - arithmetic (fan-in $\geq$ 0)
  - sign (fan-in 1)
  - output (fan-in 1)



Arithmetic Circuits over $\mathbb{R}$

Arithmetic Circuits over $\mathbb{R}$

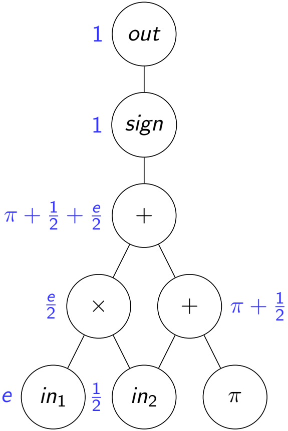
- DAG with node types:
  - input (fan-in 0)
  - constant (fan-in 0)
  - arithmetic (fan-in $\geq$ 0)
  - sign (fan-in 1)
  - output (fan-in 1)



Arithmetic Circuits over $\mathbb{R}$

Arithmetic Circuits over $\mathbb{R}$

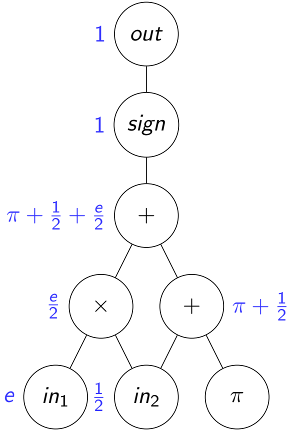
- DAG with node types:
  - input (fan-in 0)
  - constant (fan-in 0)
  - arithmetic (fan-in ≥ 0)
  - sign (fan-in 1)
  - output (fan-in 1)
- *size*: number of gates
- *depth*: longest path from input to output



Arithmetic Circuits over $\mathbb{R}$

Arithmetic Circuits over $\mathbb{R}$

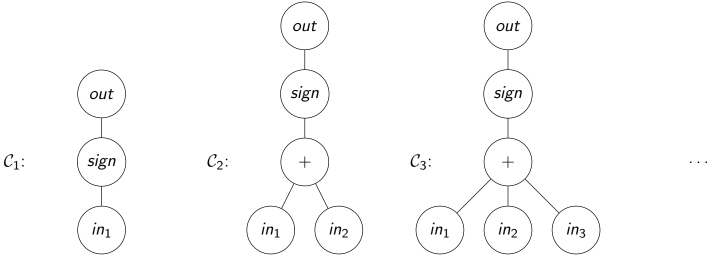
- DAG with node types:
  - input (fan-in 0)
  - constant (fan-in 0)
  - arithmetic (fan-in ≥ 0)
  - sign (fan-in 1)
  - output (fan-in 1)
- *size*: number of gates
- *depth*: longest path from input to output
- non-uniform!



Circuit families

Circuit families

- Sequences $\mathcal{C} = (\mathcal{C}_1, C_2, \ldots)$ of circuits where $\mathcal{C}_i$ has i input gates
- If $\mathcal{C}_i$ computes f^i for all $i \in \mathbb{N}$, then $\mathcal{C}$ computes $f_{\mathcal{C}}(w) = f^{|w|}(w)$.



Circuit families

Circuit families deciding sets

A circuit family $\mathcal{C}$ decides a set $S \subseteq \bigcup_{n\in\mathbb{N}} \mathbb{R}^n$ iff $\mathcal{C}$ computes the characteristic function of S.

Circuit classes

- $\mathrm{NC}^i_{\mathbb{R}}$: sets decided by bounded fan-in circuits families of size $\mathcal{O}(n^{\mathcal{O}(1)})$ and depth in $\mathcal{O}(\log(n)^i)$
- $\mathrm{AC}^i_{\mathbb{R}}$: sets decided by unbounded fan-in circuit families of size $\mathcal{O}(n^{\mathcal{O}(1)})$ and depth in $\mathcal{O}(\log(n)^i)$

Uniformity

Uniform circuit families

- There is an $\mathbb{R}$-machine M producing the circuit family.
  - i.e. M computes the function $n \mapsto \mathcal{C}_n$
- M works in polynomial time $\rightarrow$ P-uniform
  - U_{P}-C is the subclass of C decided by P-uniform families
- M works in logarithmic time $\rightarrow$ L-uniform
  - U_{LT}-C is the subclass of C decided by L-uniform families

First-order Logic over $\mathbb{R}$

$$\varphi \triangleq \exists x_1 \exists x_2 \ \mathit{val}(x_1) = \mathit{val}(x_2) + \pi \wedge x_1 = \mathit{succ}(x_2)$$

First-order Logic over $\mathbb{R}$

$$\varphi \triangleq \exists x_1 \exists x_2 \; val(x_1) = val(x_2) + \pi \wedge x_1 = succ(x_2)$$

metafinite $\mathbb{R}$-structures

- $\mathbb{R}$-structure $\mathcal{D} = (\mathcal{A}, \mathcal{F})$ of signature $\sigma = (L_s, L_f)$
  - $\mathcal{A}$: finite structure of L_s with universe A
    - the *skeleton* of $\mathcal{D}$
  - $\mathcal{F}$: finite set of functions $X : A^k \to \mathbb{R}$ interpreting symbols in L_f
    - the *arithmetic part* of $\mathcal{D}$

First-order Logic over $\mathbb{R}$

$$\sigma = (\{val\}, \{succ\})$$

$$\varphi \triangleq \exists x_1 \exists x_2 \ val(x_1) = val(x_2) + \pi \wedge x_1 = succ(x_2)$$

metafinite $\mathbb{R}$-structures

- $\mathbb{R}$-structure $\mathcal{D} = (\mathcal{A}, \mathcal{F})$ of signature $\sigma = (L_s, L_f)$
  - $\mathcal{A}$: finite structure of L_s with universe A
    - the *skeleton* of $\mathcal{D}$
  - $\mathcal{F}$: finite set of functions $X : A^k \to \mathbb{R}$ interpreting symbols in L_f
    - the *arithmetic part* of $\mathcal{D}$

First-order Logic over $\mathbb{R}$

$$\sigma = (\{val\}, \{succ\})$$

$$\varphi \triangleq \exists x_1 \exists x_2 \ val(x_1) = val(x_2) + \pi \wedge x_1 = succ(x_2)$$

$FO_{\mathbb{R}}$

- formulas and terms over signature $\sigma = (L_s, L_f)$ for variables $x_1, x_2, \ldots$
  - *index terms*: variables, functions $f \in L_s$
  - *number terms*: real numbers, functions $g \in L_f$, $t_1 + t_2$, $t_1 \times t_2$, $sign(t_1)$
  - formulas
    - atomic: $t_1 = t_2$, $t_1 \leq t_2$, predicates $P \in L_s$
    - non-atomic: closure of atomic formulas under Boolean connectives and quantification $(\exists, \forall)$

First-order Logic over $\mathbb{R}$ - Example

$$\sigma = (\{val\}, \{succ\})$$

$$\varphi \triangleq \exists x_1 \exists x_2 \; val(x_1) = val(x_2) + \pi \land x_1 = succ(x_2)$$

First-order Logic over $\mathbb{R}$ - Example

$$\sigma = (\{val\}, \{succ\})$$

$$\varphi \triangleq \exists x_1 \exists x_2 \; val(x_1) = val(x_2) + \pi \land x_1 = succ(x_2)$$

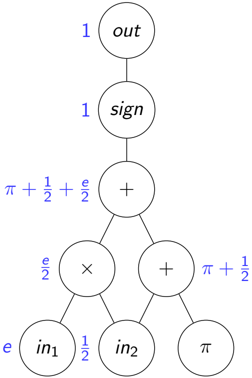
Structure $\mathcal{D}$:

$$A = \{v \mid v \in C\},$$

$$val(v) = \text{the value of } v \text{ on inputs } e \text{ and } \frac{1}{2},$$

$$succ(v) = \text{the successor gate of } v$$

Circuit C:



First-order Logic over $\mathbb{R}$ - Example

$$\sigma = (\{val\}, \{succ\})$$

$$\varphi \triangleq \exists x_1 \exists x_2 \; val(x_1) = val(x_2) + \pi \wedge x_1 = succ(x_2)$$

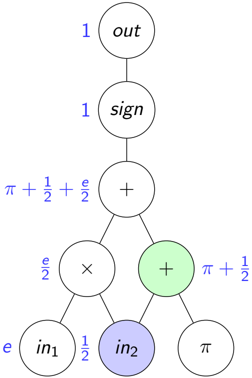
Structure $\mathcal{D}$:

$$A = \{v \mid v \in C\},$$

$$val(v) = \text{the value of } v \text{ on inputs } e \text{ and } \frac{1}{2},$$

$$succ(v) = \text{the successor gate of } v$$

Circuit C:



Extensions to $FO_{\mathbb{R}}$

Additional functions / relations

- $FO_{\mathbb{R}}[R]$ for a set R of functions and relations

Additional constructions

- the *sum*, *product* and *maximization* rules for creating number terms
  - to use $\sum_{i \in A} t(i)$, $\prod_{i \in A} t(i)$ and $\max_{i \in A}(t(i))$ in formulas

$$\varphi = \exists v_1 \; \underset{v_2}{sum}(g(v_2)) > g(v_1) \times 2$$

$$A = \{\diamondsuit, \spadesuit\}, \; g = \{\diamondsuit \mapsto \pi, \spadesuit \mapsto 42\}$$

Extensions to $FO_{\mathbb{R}}$

Additional functions / relations

- $FO_{\mathbb{R}}[R]$ for a set R of functions and relations

Additional constructions

- the *sum*, *product* and *maximization* rules for creating number terms
  - to use $\sum\limits_{i \in A} t(i)$, $\prod\limits_{i \in A} t(i)$ and $\max\limits_{i \in A}(t(i))$ in formulas

$$\varphi = \exists v_1 \; \overbrace{\underset{v_2}{sum}(g(v_2))}^{42+\pi} > g(v_1) \times 2$$

$$A = \{\diamondsuit, \spadesuit\}, \;\; g = \{\diamondsuit \mapsto \pi, \spadesuit \mapsto 42\}$$

Logical Characterizations

Cucker and Meer, 1999:

$$NC^i_{\mathbb{R}} = FP_{\mathbb{R}}[O(\log^i n)].$$ (Fixed point logic)

„The expressive power of first-order logic is not too big."

$$FO_{\mathbb{R}} \subsetneq NC^1_{\mathbb{R}}.$$

Exact characterization?

Logical Characterizations

Non-uniform $\mathrm{AC}^0_{\mathbb{R}}$

$$\mathrm{AC}^0_{\mathbb{R}} = \mathsf{FO}_{\mathbb{R}}[\mathsf{Arb}]$$

Polynomial-time uniform $\mathrm{AC}^0_{\mathbb{R}}$

$$\mathrm{U}_{\mathrm{P}}\text{-}\mathrm{AC}^0_{\mathbb{R}} = \mathsf{FO}_{\mathbb{R}}[\mathsf{FTIME}_{\mathbb{R}}(n^{\mathcal{O}(1)})]$$

Logarithmic-time uniform $\mathrm{AC}^0_{\mathbb{R}}$

$$\mathrm{U}_{\mathrm{LT}}\text{-}\mathrm{AC}^0_{\mathbb{R}} = \mathsf{FO}_{\mathbb{R}}[\mathsf{FTIME}_{\mathbb{R}}(\log(n))] + \mathsf{SUM}_{\mathbb{R}} + \mathsf{PROD}_{\mathbb{R}}$$

Logical Characterizations

Non-uniform $\mathrm{AC}^0_{\mathbb{R}}$

$$\mathrm{AC}^0_{\mathbb{R}} = \mathsf{FO}_{\mathbb{R}}[\mathsf{Arb}]$$

Polynomial-time uniform $\mathrm{AC}^0_{\mathbb{R}}$

$$\mathrm{U_P}\text{-}\mathrm{AC}^0_{\mathbb{R}} = \mathsf{FO}_{\mathbb{R}}[\mathsf{FTIME}_{\mathbb{R}}(n^{\mathcal{O}(1)})]$$

Logarithmic-time uniform $\mathrm{AC}^0_{\mathbb{R}}$

$$\mathrm{U_{LT}}\text{-}\mathrm{AC}^0_{\mathbb{R}} = \mathsf{FO}_{\mathbb{R}}[\mathsf{FTIME}_{\mathbb{R}}(\log(n))] + \mathsf{SUM}_{\mathbb{R}} + \mathsf{PROD}_{\mathbb{R}}$$

One Generalization

For $f(n) \geq \log(n)$:

$$\mathrm{U}_f\text{-}\mathrm{AC}^0_{\mathbb{R}} = \mathsf{FO}_{\mathbb{R}}[\mathsf{FTIME}_{\mathbb{R}}(f(n))] + \mathsf{SUM}_{\mathbb{R}} + \mathsf{PROD}_{\mathbb{R}}$$

Future/Current Research

- Find logical characterizations for more unbounded/semi-unbounded fan-in classes
  - e.g. AC^1, SAC^1 and the respective hierarchies
- contextualize these classes
  - separate $\mathrm{AC}^0_{\mathbb{R}}$ and $\mathrm{NC}^1_{\mathbb{R}}$
- find meaningful real analogue of TC
  - investigate then in particular the question, whether $\mathrm{TC}^0_{\mathbb{R}} = \mathrm{NC}^1_{\mathbb{R}}$

Future/Current Research

- Find logical characterizations for more unbounded/semi-unbounded fan-in classes
  - e.g. AC^1, SAC^1 and the respective hierarchies
- contextualize these classes
  - separate $\mathrm{AC}^0_{\mathbb{R}}$ and $\mathrm{NC}^1_{\mathbb{R}}$
- find meaningful real analogue of TC
  - investigate then in particular the question, whether $\mathrm{TC}^0_{\mathbb{R}} = \mathrm{NC}^1_{\mathbb{R}}$

Thank you!

Sources

[Im89] Neil Immerman. Expressibility and Parallel Complexity. SIAM J. Comput. 18(3), 625-638 (1989)

[CM99] Felipe Cucker and Klaus Meer. Logics which capture complexity classes over the reals. J. Symb. Log., 64(1):363-390, 1999.

[BCSS98] Lenore Blum, Felipe Cucker, Michael Shub, and Steve Smale. Complexity and Real Computation. Springer-Verlag, New York, 1998.