Language equations:
the story of computational completeness

ALEXANDER OKHOTIN

Department of Mathematics, University of Turku;
Academy of Finland

Dagstuhl, 17 December 2010 A. D.

Language equations

System of equations:

$$\begin{cases} \varphi_1(X_1, \ldots, X_n) & = & \psi_1(X_1, \ldots, X_n) \\ & \vdots & \\ \varphi_m(X_1, \ldots, X_n) & = & \psi_m(X_1, \ldots, X_n) \end{cases}$$

Language equations

System of equations:

$$\begin{cases} \varphi_1(X_1, \ldots, X_n) & = & \psi_1(X_1, \ldots, X_n) \\ & \vdots & \\ \varphi_m(X_1, \ldots, X_n) & = & \psi_m(X_1, \ldots, X_n) \end{cases}$$

- Alphabet Σ.

Language equations

System of equations:

$$\begin{cases} \varphi_1(X_1, \ldots, X_n) & = & \psi_1(X_1, \ldots, X_n) \\ & \vdots & \\ \varphi_m(X_1, \ldots, X_n) & = & \psi_m(X_1, \ldots, X_n) \end{cases}$$

- Alphabet Σ.
- X_i: unknown formal languages.

Language equations

System of equations:

$$\begin{cases} \varphi_1(X_1, \ldots, X_n) & = & \psi_1(X_1, \ldots, X_n) \\ & \vdots & \\ \varphi_m(X_1, \ldots, X_n) & = & \psi_m(X_1, \ldots, X_n) \end{cases}$$

- Alphabet Σ.
- X_i: unknown formal languages.
- φ_i: variables, operations on languages, constant languages.

Language equations

System of equations:

$$\begin{cases} \varphi_1(X_1, \ldots, X_n) & = & \psi_1(X_1, \ldots, X_n) \\ & \vdots & \\ \varphi_m(X_1, \ldots, X_n) & = & \psi_m(X_1, \ldots, X_n) \end{cases}$$

- Alphabet Σ.
- X_i: unknown formal languages.
- φ_i: variables, operations on languages, constant languages.
- Solutions $X_i = L_i$:

Language equations

System of equations:

$$\begin{cases} \varphi_1(X_1, \ldots, X_n) & = & \psi_1(X_1, \ldots, X_n) \\ & \vdots & \\ \varphi_m(X_1, \ldots, X_n) & = & \psi_m(X_1, \ldots, X_n) \end{cases}$$

- Alphabet Σ.
- X_i: unknown formal languages.
- φ_i: variables, operations on languages, constant languages.
- Solutions $X_i = L_i$:
  - Unique solutions.

Language equations

System of equations:

$$\begin{cases} \varphi_1(X_1, \ldots, X_n) & = & \psi_1(X_1, \ldots, X_n) \\ & \vdots & \\ \varphi_m(X_1, \ldots, X_n) & = & \psi_m(X_1, \ldots, X_n) \end{cases}$$

- Alphabet Σ.
- X_i: unknown formal languages.
- φ_i: variables, operations on languages, constant languages.
- Solutions $X_i = L_i$:
  - Unique solutions.
  - Least/greatest wrt. partial order:

$$(L_1, \ldots, L_n) \sqsubseteq (L'_1, \ldots, L'_n) \quad \text{if} \quad L_i \subseteq L'_i \text{ for all } i$$

Two simple well-known cases

1. GINSBURG, RICE (1962): context-free grammars represented by

$$\begin{cases} X_1 = \varphi_1(X_1, \ldots, X_n) \\ \vdots \\ X_n = \varphi_n(X_1, \ldots, X_n) \end{cases}$$

with operations $\{\cup, \cdot\}$.

Two simple well-known cases

1. Ginsburg, Rice (1962): context-free grammars represented by

$$\begin{cases} X_1 = \varphi_1(X_1, \ldots, X_n) \\ \vdots \\ X_n = \varphi_n(X_1, \ldots, X_n) \end{cases}$$

with operations $\{\cup, \cdot\}$.

- Rules $\boxed{A \to \alpha_1 \mid \ldots \mid \alpha_n}$ yield equation $\boxed{A = \alpha_1 \cup \ldots \cup \alpha_n}$.

Two simple well-known cases

1. Ginsburg, Rice (1962): context-free grammars represented by

$$\begin{cases} X_1 = \varphi_1(X_1, \ldots, X_n) \\ \vdots \\ X_n = \varphi_n(X_1, \ldots, X_n) \end{cases}$$

with operations $\{\cup, \cdot\}$.

- Rules $\boxed{A \to \alpha_1 \mid \ldots \mid \alpha_n}$ yield equation $\boxed{A = \alpha_1 \cup \ldots \cup \alpha_n}$.
- Least solution.

Two simple well-known cases

1. GINSBURG, RICE (1962): context-free grammars represented by

$$\begin{cases} X_1 = \varphi_1(X_1, \ldots, X_n) \\ \vdots \\ X_n = \varphi_n(X_1, \ldots, X_n) \end{cases}$$

with operations $\{\cup, \cdot\}$.

- Rules $\boxed{A \to \alpha_1 \mid \ldots \mid \alpha_n}$ yield equation $\boxed{A = \alpha_1 \cup \ldots \cup \alpha_n}$.
- Least solution.
- With operations $\{\cup, \cap, \cdot\}$: conjunctive grammars (OKHOTIN, 2001).

Two simple well-known cases

1. GINSBURG, RICE (1962): context-free grammars represented by

$$\begin{cases} X_1 = \varphi_1(X_1, \ldots, X_n) \\ \vdots \\ X_n = \varphi_n(X_1, \ldots, X_n) \end{cases}$$

with operations $\{\cup, \cdot\}$.

   - Rules $\boxed{A \to \alpha_1 \mid \ldots \mid \alpha_n}$ yield equation $\boxed{A = \alpha_1 \cup \ldots \cup \alpha_n}$.
   - Least solution.
   - With operations $\{\cup, \cap, \cdot\}$: conjunctive grammars (OKHOTIN, 2001).

2. One-sided concatenation:
small fragment of MSO on infinite trees (RABIN, 1969).

Two simple well-known cases

1. GINSBURG, RICE (1962): context-free grammars represented by

$$\begin{cases} X_1 = \varphi_1(X_1, \ldots, X_n) \\ \vdots \\ X_n = \varphi_n(X_1, \ldots, X_n) \end{cases}$$

with operations $\{\cup, \cdot\}$.

- Rules $\boxed{A \to \alpha_1 \mid \ldots \mid \alpha_n}$ yield equation $\boxed{A = \alpha_1 \cup \ldots \cup \alpha_n}$.
- Least solution.
- With operations $\{\cup, \cap, \cdot\}$: conjunctive grammars (OKHOTIN, 2001).

2. One-sided concatenation:
small fragment of MSO on infinite trees (RABIN, 1969).
   - Solutions are regular. Everything is decidable.

Two simple well-known cases

1. GINSBURG, RICE (1962): context-free grammars represented by

$$\begin{cases} X_1 = \varphi_1(X_1, \ldots, X_n) \\ \vdots \\ X_n = \varphi_n(X_1, \ldots, X_n) \end{cases}$$

   with operations $\{\cup, \cdot\}$.

   - Rules $\boxed{A \to \alpha_1 \mid \ldots \mid \alpha_n}$ yield equation $\boxed{A = \alpha_1 \cup \ldots \cup \alpha_n}$.
   - Least solution.
   - With operations $\{\cup, \cap, \cdot\}$: conjunctive grammars (OKHOTIN, 2001).

2. One-sided concatenation:
   small fragment of MSO on infinite trees (RABIN, 1969).

   - Solutions are regular. Everything is decidable.
   - Simpler proof of regularity. Everything is EXPTIME-complete. (BAADER, OKHOTIN, 2006).

Undecidable properties

- "Given CFG $G = (\Sigma, N, P, S)$, is $L(G)$ equal to Σ^*?": undecidable.

Undecidable properties

- "Given CFG $G = (\Sigma, N, P, S)$, is $L(G)$ equal to Σ^*?": undecidable.
- Solution uniqueness for $X_i = \varphi_i(X_1, \ldots, X_n)$ is undecidable.

Undecidable properties

- "Given CFG $G = (\Sigma, N, P, S)$, is $L(G)$ equal to Σ^*?": undecidable.
- Solution uniqueness for $X_i = \varphi_i(X_1, \ldots, X_n)$ is undecidable.

$$S' = S' \cup S$$
$$S = \ldots$$

Undecidable properties

- "Given CFG $G = (\Sigma, N, P, S)$, is $L(G)$ equal to Σ^*?": undecidable.
- Solution uniqueness for $X_i = \varphi_i(X_1, \ldots, X_n)$ is undecidable.

$$S' = S' \cup S$$
$$S = \ldots$$

- General method: computation histories (Hartmanis, 1968).

Undecidable properties

- "Given CFG $G = (\Sigma, N, P, S)$, is $L(G)$ equal to Σ^*?": undecidable.
- Solution uniqueness for $X_i = \varphi_i(X_1, \ldots, X_n)$ is undecidable.

$$S' = S' \cup S$$
$$S = \ldots$$

- General method: computation histories (Hartmanis, 1968).
  - Turing machine T.

Undecidable properties

- "Given CFG $G = (\Sigma, N, P, S)$, is $L(G)$ equal to Σ^*?": undecidable.
- Solution uniqueness for $X_i = \varphi_i(X_1, \ldots, X_n)$ is undecidable.

$$S' = S' \cup S$$
$$S = \ldots$$

- General method: computation histories (Hartmanis, 1968).
  - Turing machine T.

$$\text{VALC}(T) = \{w\natural C_T(w) \mid w \in L(T)\}$$

Undecidable properties

- "Given CFG $G = (\Sigma, N, P, S)$, is $L(G)$ equal to Σ^*?": undecidable.
- Solution uniqueness for $X_i = \varphi_i(X_1, \ldots, X_n)$ is undecidable.

$$S' = S' \cup S$$
$$S = \ldots$$

- General method: computation histories (Hartmanis, 1968).
  - Turing machine T.

$$\mathrm{VALC}(T) = \{w \natural C_T(w) \mid w \in L(T)\}$$

  - Suitable $C_T : \Sigma^* \to \Gamma^*$ encodes computations.

Undecidable properties

- "Given CFG $G = (\Sigma, N, P, S)$, is $L(G)$ equal to Σ^*?": undecidable.
- Solution uniqueness for $X_i = \varphi_i(X_1, \ldots, X_n)$ is undecidable.

$$S' = S' \cup S$$
$$S = \ldots$$

- General method: computation histories (Hartmanis, 1968).
  - Turing machine T.

$$\text{VALC}(T) = \{w\natural C_T(w) \mid w \in L(T)\}$$

  - Suitable $C_T : \Sigma^* \to \Gamma^*$ encodes computations.
  - Complement of VALC(T) is CF.

Undecidable properties

- "Given CFG $G = (\Sigma, N, P, S)$, is $L(G)$ equal to Σ^*?": undecidable.
- Solution uniqueness for $X_i = \varphi_i(X_1, \ldots, X_n)$ is undecidable.

$$S' = S' \cup S$$
$$S = \ldots$$

- General method: computation histories (Hartmanis, 1968).
  - Turing machine T.

$$\text{VALC}(T) = \{w \natural C_T(w) \mid w \in L(T)\}$$

  - Suitable $C_T : \Sigma^* \to \Gamma^*$ encodes computations.
  - Complement of VALC(T) is CF.
  - $\text{VALC}(T) = L(G_1) \cap L(G_2)$ for context-free G_1, G_2.

Undecidable properties

- "Given CFG $G = (\Sigma, N, P, S)$, is $L(G)$ equal to Σ^*?": undecidable.
- Solution uniqueness for $X_i = \varphi_i(X_1, \ldots, X_n)$ is undecidable.

$$S' = S' \cup S$$
$$S = \ldots$$

- General method: computation histories (Hartmanis, 1968).
  - Turing machine T.

$$\mathrm{VALC}(T) = \{w\natural C_T(w) \mid w \in L(T)\}$$

  - Suitable $C_T : \Sigma^* \to \Gamma^*$ encodes computations.
  - Complement of VALC(T) is CF.
  - $\mathrm{VALC}(T) = L(G_1) \cap L(G_2)$ for context-free G_1, G_2.
- Solution existence for $\varphi_i(X_1, \ldots, X_n) = \psi_i(X_1, \ldots, X_n)$ is undecidable (Parikh et al., 1985; Charatonik, 1994).

Computationally universal solutions

Theorem (Okhotin, ICALP 2003)

$L \subseteq \Sigma^*$ with $|\Sigma| \geqslant 2$ is given by a least solution of a system $\varphi = \psi$ with $\{\cup, \cap, \cdot\}$

if and only if

L is r.e.

Computationally universal solutions

Theorem (Okhotin, ICALP 2003)

$L \subseteq \Sigma^*$ with $|\Sigma| \geqslant 2$ is given by a least solution of a system $\varphi = \psi$ with $\{\cup, \cap, \cdot\}$

if and only if

L is r.e.

- $\mathrm{VALC}(T) = \{w \natural C_T(w) \mid w \in L(T)\}$: accepting computations.

Computationally universal solutions

Theorem (Okhotin, ICALP 2003)

$L \subseteq \Sigma^*$ with $|\Sigma| \geqslant 2$ is given by a least solution of a system $\varphi = \psi$ with $\{\cup, \cap, \cdot\}$

if and only if

L is r.e.

- $\text{VALC}(T) = \{w\natural C_T(w) \mid w \in L(T)\}$: accepting computations.
- $L(T)$ is a least solution of

$$\text{VALC}(T) \subseteq X\natural\Sigma^*$$

Computationally universal solutions

Theorem (Okhotin, ICALP 2003)

$L \subseteq \Sigma^*$ with $|\Sigma| \geqslant 2$ is given by a least solution of a system $\varphi = \psi$ with $\{\cup, \cap, \cdot\}$

if and only if

L is r.e.

- $\mathrm{VALC}(T) = \{w \natural C_T(w) \mid w \in L(T)\}$: accepting computations.
- $L(T)$ is a least solution of

$$\mathrm{VALC}(T) \subseteq X \natural \Sigma^*$$

  - $\mathrm{VALC}(T) \cup X \natural \Sigma^* = X \natural \Sigma^*$.

Computationally universal solutions

Theorem (Okhotin, ICALP 2003)

$L \subseteq \Sigma^*$ with $|\Sigma| \geqslant 2$ is given by a least (greatest, unique) solution of a system $\varphi = \psi$ with $\{\cup, \cap, \cdot\}$

if and only if

L is r.e. (co-r.e., recursive).

- $\mathrm{VALC}(T) = \{w\natural C_T(w) \mid w \in L(T)\}$: accepting computations.
- $L(T)$ is a least solution of

$$\mathrm{VALC}(T) \subseteq X\natural\Sigma^*$$

  - $\mathrm{VALC}(T) \cup X\natural\Sigma^* = X\natural\Sigma^*$.
- Greatest and unique solutions.

Computationally universal solutions

Theorem (Okhotin, ICALP 2003)

$L \subseteq \Sigma^*$ with $|\Sigma| \geqslant 2$ is given by a least (greatest, unique) solution of a system $\varphi = \psi$ with $\{\cup, \cap, \cdot\}$

if and only if

L is r.e. (co-r.e., recursive).

- $\mathrm{VALC}(T) = \{w\natural C_T(w) \mid w \in L(T)\}$: accepting computations.
- $L(T)$ is a least solution of

$$\mathrm{VALC}(T) \subseteq X\natural\Sigma^*$$

  - $\mathrm{VALC}(T) \cup X\natural\Sigma^* = X\natural\Sigma^*$.
- Greatest and unique solutions.
- Matching upper bounds.

Fewer operations

- Systems $\varphi = \psi$ with $\{\cup, \cdot\}$ or $\{\cap, \cdot\}$ (Okhotin, 2005).

Fewer operations

- Systems $\varphi = \psi$ with $\{\cup, \cdot\}$ or $\{\cap, \cdot\}$ (Okhotin, 2005).
- Systems $\varphi = \psi$ with $\{\Delta, \cdot\}$ (Okhotin, 2006).

Fewer operations

- Systems $\varphi = \psi$ with $\{\cup, \cdot\}$ or $\{\cap, \cdot\}$ (Okhotin, 2005).
- Systems $\varphi = \psi$ with $\{\Delta, \cdot\}$ (Okhotin, 2006).
- $LX = XL$ (Kunc, 2005). Finite constant $L \subseteq \{a, b\}^*$:

Fewer operations

- Systems $\varphi = \psi$ with $\{\cup, \cdot\}$ or $\{\cap, \cdot\}$ (Okhotin, 2005).
- Systems $\varphi = \psi$ with $\{\Delta, \cdot\}$ (Okhotin, 2006).
- $LX = XL$ (Kunc, 2005). Finite constant $L \subseteq \{a, b\}^*$: greatest X is co-r.e.-complete.

Fewer operations

- Systems $\varphi = \psi$ with $\{\cup, \cdot\}$ or $\{\cap, \cdot\}$ (Okhotin, 2005).
- Systems $\varphi = \psi$ with $\{\Delta, \cdot\}$ (Okhotin, 2006).
- $LX = XL$ (Kunc, 2005). Finite constant $L \subseteq \{a, b\}^*$: greatest X is co-r.e.-complete.
- Multiple-letter alphabet in all cases.

The one-letter case: $\Sigma = \{a\}$

Example (Leiss, 1995)

$$X = a\overline{\overline{\overline{X}^2}^2}^2$$

has a unique solution $\{n \mid \exists i \geqslant 0 : \; 2^{3i} \leqslant n < 2^{3i+2}\}$.

The one-letter case: $\Sigma = \{a\}$

Example (Leiss, 1995)

$$X = a\overline{\overline{\overline{X}^2}^2}^2$$

has a unique solution $\{n \mid \exists i \geqslant 0 : \ 2^{3i} \leqslant n < 2^{3i+2}\}$.

- $= \{n \mid$ base-8 notation of n begins with 1, 2 or 3$\}$.

The one-letter case: $\Sigma = \{a\}$

Example (Leiss, 1995)

$$X = a\overline{\overline{\overline{X}^2}^2}^2$$

has a unique solution $\{n \mid \exists i \geqslant 0 : \ 2^{3i} \leqslant n < 2^{3i+2}\}$.

- $= \{n \mid$ base-8 notation of n begins with 1, 2 or 3$\}$.

Example (Jeż, 2007)

$$\begin{aligned} X_1 &= (X_1X_3 \cap X_2X_2) \ \cup \ \{a\} \\ X_2 &= (X_1X_1 \cap X_2X_6) \ \cup \ \{aa\} \\ X_3 &= (X_1X_2 \cap X_6X_6) \ \cup \ \{aaa\} \\ X_6 &= (X_1X_2 \cap X_3X_3) \end{aligned}$$

has a least solution $X_i = \{a^{i \cdot 4^n} \mid n \geqslant 0\}$.

The one-letter case: $\Sigma = \{a\}$

Example (Leiss, 1995)

$$X = a\overline{\overline{\overline{X}^2}^2}^2$$

has a unique solution $\{n \mid \exists i \geqslant 0 : \ 2^{3i} \leqslant n < 2^{3i+2}\}$.

- $= \{n \mid$ base-8 notation of n begins with 1, 2 or 3$\}$.

Example (Jeż, 2007)

$$\begin{aligned}
X_1 &= (X_1X_3 \cap X_2X_2) \cup \{a\} \\
X_2 &= (X_1X_1 \cap X_2X_6) \cup \{aa\} \\
X_3 &= (X_1X_2 \cap X_6X_6) \cup \{aaa\} \\
X_6 &= (X_1X_2 \cap X_3X_3)
\end{aligned}$$

has a least solution $X_i = \{a^{i \cdot 4^n} \mid n \geqslant 0\}$.

- In base-4: $(10^*)_4$, $(20^*)_4$, $(30^*)_4$, $(120^*)_4$.

Base-k notation

Theorem (Jeż, Okhotin, 2007)

Assume $L \subseteq \{0, 1, \ldots, k-1\}$ is recognized by a one-way real-time cellular automaton. Then $(L)_k = \{a^n \mid \text{base-}k \text{ notation of } n \text{ is in } L\}$ is generated by a conjunctive grammar.

Base-k notation

Theorem (Jeż, Okhotin, 2007)

Assume $L \subseteq \{0, 1, \dots, k-1\}$ is recognized by a one-way real-time cellular automaton. Then $(L)_k = \{a^n \mid \text{base-}k \text{ notation of } n \text{ is in } L\}$ is generated by a conjunctive grammar.

- That is, by unique solutions of systems

$$\begin{cases} X_1 = \varphi_1(X_1, \dots, X_n) \\ \vdots \\ X_n = \varphi_n(X_1, \dots, X_n) \end{cases}$$

with $\{\cup, \cap, \cdot\}$.

Base-k notation

Theorem (Jeż, Okhotin, 2007)

Assume $L \subseteq \{0, 1, \ldots, k-1\}$ is recognized by a one-way real-time cellular automaton. Then $(L)_k = \{a^n \mid \text{base-}k \text{ notation of } n \text{ is in } L\}$ is generated by a conjunctive grammar.

- That is, by unique solutions of systems

$$\begin{cases} X_1 = \varphi_1(X_1, \ldots, X_n) \\ \vdots \\ X_n = \varphi_n(X_1, \ldots, X_n) \end{cases}$$

  with $\{\cup, \cap, \cdot\}$.

- VALC(T) recognized by such CA.

Base-k notation

Theorem (JEŻ, OKHOTIN, 2007)

Assume $L \subseteq \{0, 1, \ldots, k-1\}$ is recognized by a one-way real-time cellular automaton. Then $(L)_k = \{a^n \mid \text{base-}k \text{ notation of } n \text{ is in } L\}$ is generated by a conjunctive grammar.

- That is, by unique solutions of systems

$$\begin{cases} X_1 &= \varphi_1(X_1, \ldots, X_n) \\ &\vdots \\ X_n &= \varphi_n(X_1, \ldots, X_n) \end{cases}$$

  with $\{\cup, \cap, \cdot\}$.
- VALC(T) recognized by such CA.
- Consider VALC(T) over $\Sigma = \{0, 1, \ldots, k-1\}$.

Base-k notation

Theorem (Jeż, Okhotin, 2007)

Assume $L \subseteq \{0, 1, \ldots, k-1\}$ is recognized by a one-way real-time cellular automaton. Then $(L)_k = \{a^n \mid \text{base-}k \text{ notation of } n \text{ is in } L\}$ is generated by a conjunctive grammar.

- That is, by unique solutions of systems

$$\begin{cases} X_1 = \varphi_1(X_1, \ldots, X_n) \\ \vdots \\ X_n = \varphi_n(X_1, \ldots, X_n) \end{cases}$$

  with $\{\cup, \cap, \cdot\}$.
- VALC(T) recognized by such CA.
- Consider VALC(T) over $\Sigma = \{0, 1, \ldots, k-1\}$.
- $(\text{VALC}(T))_k$ represented by language equations.

Extracting $L(T)$ from $(\text{VALC}(T))_k$

- Cf: $\boxed{\text{VALC}(T) \subseteq X\sharp\Sigma^*}$ for larger alphabets.

Extracting $L(T)$ from $(\mathrm{VALC}(T))_k$

- Cf: $\boxed{\mathrm{VALC}(T) \subseteq X\sharp\Sigma^*}$ for larger alphabets.
- Here: same general idea, more difficult.

Extracting $L(T)$ from $(\text{VALC}(T))_k$

- Cf: $\boxed{\text{VALC}(T) \subseteq X\sharp\Sigma^*}$ for larger alphabets.
- Here: same general idea, more difficult.
- Extract number $(w)_k$ from number $(C_T(w)\natural w)_k$.

Extracting $L(T)$ from $(\mathrm{VALC}(T))_k$

- Cf: $\boxed{\mathrm{VALC}(T) \subseteq X\sharp\Sigma^*}$ for larger alphabets.
- Here: same general idea, more difficult.
- Extract number $(w)_k$ from number $(C_T(w)\natural w)_k$.
- A special encoding of VALC(T) over base-6 alphabet.

Extracting $L(T)$ from $(\mathrm{VALC}(T))_k$

- Cf: $\boxed{\mathrm{VALC}(T) \subseteq X\sharp\Sigma^*}$ for larger alphabets.
- Here: same general idea, more difficult.
- Extract number $(w)_k$ from number $(C_T(w)\natural w)_k$.
- A special encoding of $\mathrm{VALC}(T)$ over base-6 alphabet.

Theorem (Jeż, Okhotin, ICALP 2008)

$L \subseteq a^*$ is given by unique (least, greatest) solution of a system $\varphi = \psi$ with operations $\{\cup, \cap, \cdot\}$

if and only if

L is recursive (r.e., co-r.e.).

Extracting $L(T)$ from $(\mathrm{VALC}(T))_k$

- Cf: $\boxed{\mathrm{VALC}(T) \subseteq X\sharp\Sigma^*}$ for larger alphabets.
- Here: same general idea, more difficult.
- Extract number $(w)_k$ from number $(C_T(w)\natural w)_k$.
- A special encoding of VALC(T) over base-6 alphabet.

Theorem (Jeż, Okhotin, ICALP 2008)

$L \subseteq a^*$ is given by unique (least, greatest) solution of a system $\varphi = \psi$ with operations $\{\cup, \cdot\}$

if and only if

L is recursive (r.e., co-r.e.).

- Intersection not needed! (reconstruction from scratch)

Unary alphabet, only concatenation

- Can anything be represented with concatenation only?

Unary alphabet, only concatenation

- Can anything be represented with concatenation only?
- Denote $\tau_i(L) = \{a^{16n+i} \mid a^n \in L\}$: track i.

Unary alphabet, only concatenation

- Can anything be represented with concatenation only?
- Denote $\tau_i(L) = \{a^{16n+i} \mid a^n \in L\}$: track i.
- Simulate equations with $\{\cup, \cdot\}$ using only $\{\cdot\}$.

Unary alphabet, only concatenation

- Can anything be represented with concatenation only?
- Denote $\tau_i(L) = \{a^{16n+i} \mid a^n \in L\}$: track i.
- Simulate equations with $\{\cup, \cdot\}$ using only $\{\cdot\}$.

Theorem (Jeż, Okhotin, STACS 2009)

For every recursive (r.e., co-r.e.) language $L \subseteq a^*$, the set

$$\sigma(L) = \{\varepsilon\} \cup \tau_6(\mathbb{N}) \cup \tau_8(\mathbb{N}) \cup \tau_9(\mathbb{N}) \cup \tau_{12}(\mathbb{N}) \cup \tau_{13}(L).$$

is representable by unique (least, greatest) solution of a system with $\{\cdot\}$.

Unary alphabet, only concatenation

- Can anything be represented with concatenation only?
- Denote $\tau_i(L) = \{a^{16n+i} \mid a^n \in L\}$: track i.
- Simulate equations with $\{\cup, \cdot\}$ using only $\{\cdot\}$.

Theorem (Jeż, Okhotin, STACS 2009)

For every recursive (r.e., co-r.e.) language $L \subseteq a^*$, the set

$$\sigma(L) = \{\varepsilon\} \cup \tau_6(\mathbb{N}) \cup \tau_8(\mathbb{N}) \cup \tau_9(\mathbb{N}) \cup \tau_{12}(\mathbb{N}) \cup \tau_{13}(L).$$

is representable by unique (least, greatest) solution of a system with $\{\cdot\}$.

- Encoding checked by
$X \cdot \{\varepsilon, a^4, a^{11}\} = \bigcup_{i \in \{0,4,6,8,9,10,12,13\}} \tau_i(a^*) \cup \bigcup_{i \in \{1,3,7\}} \tau_i(a^+) \cup \{a^{11}\}$.

Unary alphabet, only concatenation

- Can anything be represented with concatenation only?
- Denote $\tau_i(L) = \{a^{16n+i} \mid a^n \in L\}$: track i.
- Simulate equations with $\{\cup, \cdot\}$ using only $\{\cdot\}$.

Theorem (Jeż, Okhotin, STACS 2009)

For every recursive (r.e., co-r.e.) language $L \subseteq a^*$, the set

$$\sigma(L) = \{\varepsilon\} \cup \tau_6(\mathbb{N}) \cup \tau_8(\mathbb{N}) \cup \tau_9(\mathbb{N}) \cup \tau_{12}(\mathbb{N}) \cup \tau_{13}(L).$$

is representable by unique (least, greatest) solution of a system with $\{\cdot\}$.

- Encoding checked by
$X \cdot \{\varepsilon, a^4, a^{11}\} = \bigcup_{i \in \{0,4,6,8,9,10,12,13\}} \tau_i(a^*) \cup \bigcup_{i \in \{1,3,7\}} \tau_i(a^+) \cup \{a^{11}\}$.
- $KL = MN$ checked by $\sigma(K)\sigma(L)\{\varepsilon, a\} = \sigma(M)\sigma(N)\{\varepsilon, a\}$.

Unary alphabet, only concatenation

- Can anything be represented with concatenation only?
- Denote $\tau_i(L) = \{a^{16n+i} \mid a^n \in L\}$: track i.
- Simulate equations with $\{\cup, \cdot\}$ using only $\{\cdot\}$.

Theorem (Jeż, Okhotin, STACS 2009)

For every recursive (r.e., co-r.e.) language $L \subseteq a^*$, the set

$$\sigma(L) = \{\varepsilon\} \cup \tau_6(\mathbb{N}) \cup \tau_8(\mathbb{N}) \cup \tau_9(\mathbb{N}) \cup \tau_{12}(\mathbb{N}) \cup \tau_{13}(L).$$

is representable by unique (least, greatest) solution of a system with $\{\cdot\}$.

- Encoding checked by
  $X \cdot \{\varepsilon, a^4, a^{11}\} = \bigcup_{i \in \{0,4,6,8,9,10,12,13\}} \tau_i(a^*) \cup \bigcup_{i \in \{1,3,7\}} \tau_i(a^+) \cup \{a^{11}\}$.
- $KL = MN$ checked by $\sigma(K)\sigma(L)\{\varepsilon, a\} = \sigma(M)\sigma(N)\{\varepsilon, a\}$.
- $K \cup L = M \cup N$ checked by $\sigma(K)\sigma(L)\{\varepsilon, a^2\} = \sigma(M)\sigma(N)\{\varepsilon, a^2\}$.

One variable, two equations

- Encode all variables into one by $\pi : \left(2^{a^*}\right)^n \to 2^{a^*}$.

One variable, two equations

- Encode all variables into one by $\pi : \left(2^{a^*}\right)^n \to 2^{a^*}$.

Theorem (Lehtinen, Okhotin, 2010)

For every recursive (r.e., co-r.e.) language $L_0 \subseteq a^*$ there exists a system

$$\begin{cases} XXK = XL \\ XM = N \end{cases}$$

with a unique (least, greatest, respectively) solution $X = L_0'$, such that $a^n \in L_0$ if and only if $a^{pn+d} \in L_0'$.

One variable, two equations

- Encode all variables into one by $\pi : \left(2^{a^*}\right)^n \to 2^{a^*}$.

Theorem (Lehtinen, Okhotin, 2010)

For every recursive (r.e., co-r.e.) language $L_0 \subseteq a^*$ there exists a system

$$\begin{cases} XXK = XL \\ XM = N \end{cases}$$

with a unique (least, greatest, respectively) solution $X = L_0'$, such that $a^n \in L_0$ if and only if $a^{pn+d} \in L_0'$.

- Apparently the simplest form of language equations.

One variable, two equations

- Encode all variables into one by $\pi : \left(2^{a^*}\right)^n \to 2^{a^*}$.

Theorem (Lehtinen, Okhotin, 2010)

For every recursive (r.e., co-r.e.) language $L_0 \subseteq a^*$ there exists a system

$$\begin{cases} XXK = XL \\ XM = N \end{cases}$$

with a unique (least, greatest, respectively) solution $X = L_0'$, such that $a^n \in L_0$ if and only if $a^{pn+d} \in L_0'$.

- Apparently the simplest form of language equations.
- Decision problems for such systems:

One variable, two equations

- Encode all variables into one by $\pi : \left(2^{a^*}\right)^n \to 2^{a^*}$.

Theorem (Lehtinen, Okhotin, 2010)

For every recursive (r.e., co-r.e.) language $L_0 \subseteq a^*$ there exists a system

$$\begin{cases} XXK = XL \\ XM = N \end{cases}$$

with a unique (least, greatest, respectively) solution $X = L_0'$, such that $a^n \in L_0$ if and only if $a^{pn+d} \in L_0'$.

- Apparently the simplest form of language equations.
- Decision problems for such systems:
  - Existence of solution: Π_1^0-complete;

One variable, two equations

- Encode all variables into one by $\pi : \left(2^{a^*}\right)^n \to 2^{a^*}$.

Theorem (Lehtinen, Okhotin, 2010)

For every recursive (r.e., co-r.e.) language $L_0 \subseteq a^*$ there exists a system

$$\begin{cases} XXK = XL \\ XM = N \end{cases}$$

with a unique (least, greatest, respectively) solution $X = L_0'$, such that $a^n \in L_0$ if and only if $a^{pn+d} \in L_0'$.

- Apparently the simplest form of language equations.
- Decision problems for such systems:
  - Existence of solution: Π_1^0-complete;
  - Uniqueness of solution: Π_2^0-complete;

One variable, two equations

- Encode all variables into one by $\pi : \left(2^{a^*}\right)^n \to 2^{a^*}$.

Theorem (Lehtinen, Okhotin, 2010)

For every recursive (r.e., co-r.e.) language $L_0 \subseteq a^*$ there exists a system

$$\begin{cases} XXK = XL \\ XM = N \end{cases}$$

with a unique (least, greatest, respectively) solution $X = L_0'$, such that $a^n \in L_0$ if and only if $a^{pn+d} \in L_0'$.

- Apparently the simplest form of language equations.
- Decision problems for such systems:
  - Existence of solution: Π_1^0-complete;
  - Uniqueness of solution: Π_2^0-complete;
  - Having finitely many solutions: Σ_3^0-complete;

Any future work?

- Computational completeness in language equations:

Any future work?

- Computational completeness in language equations:
  - All or almost all rec/r.e./co-r.e. sets by unique/least/greatest sol.

Any future work?

- Computational completeness in language equations:
  - All or almost all rec/r.e./co-r.e. sets by unique/least/greatest sol.
  - Upper bound: recursive constants, computable operations.

Any future work?

- Computational completeness in language equations:
  - All or almost all rec/r.e./co-r.e. sets by unique/least/greatest sol.
  - Upper bound: recursive constants, computable operations.
  - Lower bound: $XXK = XL$, $XM = N$, all unary.

Any future work?

- Computational completeness in language equations:
  - All or almost all rec/r.e./co-r.e. sets by unique/least/greatest sol.
  - Upper bound: recursive constants, computable operations.
  - Lower bound: $XXK = XL$, $XM = N$, all unary.
  - No more room for improvement.

Any future work?

- Computational completeness in language equations:
  - All or almost all rec/r.e./co-r.e. sets by unique/least/greatest sol.
  - Upper bound: recursive constants, computable operations.
  - Lower bound: $XXK = XL$, $XM = N$, all unary.
  - No more room for improvement.
- Any further properties?

Any future work?

- Computational completeness in language equations:
  - All or almost all rec/r.e./co-r.e. sets by unique/least/greatest sol.
  - Upper bound: recursive constants, computable operations.
  - Lower bound: $XXK = XL$, $XM = N$, all unary.
  - No more room for improvement.
- Any further properties?
  - "Does a given system have countably many solutions?"

Any future work?

- Computational completeness in language equations:
  - All or almost all rec/r.e./co-r.e. sets by unique/least/greatest sol.
  - Upper bound: recursive constants, computable operations.
  - Lower bound: $XXK = XL$, $XM = N$, all unary.
  - No more room for improvement.
- Any further properties?
  - "Does a given system have countably many solutions?"
  - More to consider.

Any future work?

- Computational completeness in language equations:
  - All or almost all rec/r.e./co-r.e. sets by unique/least/greatest sol.
  - Upper bound: recursive constants, computable operations.
  - Lower bound: $XXK = XL$, $XM = N$, all unary.
  - No more room for improvement.
- Any further properties?
  - "Does a given system have countably many solutions?"
  - More to consider.
- Any more powerful models?

Any future work?

- Computational completeness in language equations:
  - All or almost all rec/r.e./co-r.e. sets by unique/least/greatest sol.
  - Upper bound: recursive constants, computable operations.
  - Lower bound: $XXK = XL$, $XM = N$, all unary.
  - No more room for improvement.
- Any further properties?
  - "Does a given system have countably many solutions?"
  - More to consider.
- Any more powerful models?
  - Equations over sets of integers (Jeż, Okhotin, 2009–2010): reach up to Σ^1_1 sets.

Any future work?

- Computational completeness in language equations:
  - All or almost all rec/r.e./co-r.e. sets by unique/least/greatest sol.
  - Upper bound: recursive constants, computable operations.
  - Lower bound: $XXK = XL$, $XM = N$, all unary.
  - No more room for improvement.
- Any further properties?
  - "Does a given system have countably many solutions?"
  - More to consider.
- Any more powerful models?
  - Equations over sets of integers (JEŻ, OKHOTIN, 2009–2010): reach up to Σ^1_1 sets.
- Any weaker models?

Any future work?

- Computational completeness in language equations:
  - All or almost all rec/r.e./co-r.e. sets by unique/least/greatest sol.
  - Upper bound: recursive constants, computable operations.
  - Lower bound: $XXK = XL$, $XM = N$, all unary.
  - No more room for improvement.
- Any further properties?
  - "Does a given system have countably many solutions?"
  - More to consider.
- Any more powerful models?
  - Equations over sets of integers (Jeż, Okhotin, 2009–2010): reach up to Σ_1^1 sets.
- Any weaker models?
  - Yes, formal grammars.

Any future work?

- Computational completeness in language equations:
  - All or almost all rec/r.e./co-r.e. sets by unique/least/greatest sol.
  - Upper bound: recursive constants, computable operations.
  - Lower bound: $XXK = XL$, $XM = N$, all unary.
  - No more room for improvement.
- Any further properties?
  - "Does a given system have countably many solutions?"
  - More to consider.
- Any more powerful models?
  - Equations over sets of integers (Jeż, Okhotin, 2009–2010): reach up to Σ^1_1 sets.
- Any weaker models?
  - Yes, formal grammars.
  - Problems for conjunctive and Boolean grammars.

Any future work?

- Computational completeness in language equations:
  - All or almost all rec/r.e./co-r.e. sets by unique/least/greatest sol.
  - Upper bound: recursive constants, computable operations.
  - Lower bound: $XXK = XL$, $XM = N$, all unary.
  - No more room for improvement.
- Any further properties?
  - "Does a given system have countably many solutions?"
  - More to consider.
- Any more powerful models?
  - Equations over sets of integers (Jeż, Okhotin, 2009–2010): reach up to Σ^1_1 sets.
- Any weaker models?
  - Yes, formal grammars.
  - Problems for conjunctive and Boolean grammars.
  - Perhaps, more general models.