Jumping Finite Automata: New Results

Part Two: New Models

Radim Kocman and Zbyněk Křivka

Faculty of Information Technology
Brno University of Technology
Božetěchova 2, Brno Czech Republic

{ikocman,krivka}@fit.vutbr.cz

BRNO UNIVERSITY OF TECHNOLOGY FACULTY OF INFORMATION TECHNOLOGY

LTA 2016 (December 5, 2016)

Table of Contents

Motivation

Why study other models?

Motivation – Jumping Finite Automata

Possible Advantages

- completely discontinuous reading
- can accept some CF and CS languages

Motivation – Jumping Finite Automata

Possible Advantages

- completely discontinuous reading
- can accept some CF and CS languages

Possible Disadvantages

- cannot guarantee any specific reading order
- therefore it cannot accept languages like $a^* b^*$
- heavily nondeterministic behavior

Motivation – Right and Left Jumps

Definition

A GJFA makes a right jump from $wpyxz$ to $wxqz$ by $py \rightarrow q$:

$$wpyxz \; {}_{r}\curvearrowright \; wxqz$$

where $w, x, y, z \in \Sigma^*$.

Motivation – Right and Left Jumps

Definition

A GJFA makes a right jump from $wpyxz$ to $wxqz$ by $py \rightarrow q$:

$$wpyxz \; {}_r\!\curvearrowright \; wxqz$$

where $w, x, y, z \in \Sigma^*$.

Definition

A GJFA makes a left jump from $wxpyz$ to $wqxz$ by $py \rightarrow q$:

$$wxpyz \; {}_l\!\curvearrowright \; wqxz$$

where $w, x, y, z \in \Sigma^*$.

Properties of Right Jumps

- consider the configuration upv, where $p \in Q$, $u, v \in \Sigma^*$
- the automaton will get stuck for any $|u| > 0$
- result: the same power as FAs

Motivation – Right and Left Jumps

Properties of Right Jumps

- consider the configuration upv, where $p \in Q$, $u, v \in \Sigma^*$
- the automaton will get stuck for any $|u| > 0$
- result: the same power as FAs

Properties of Left Jumps

- open problem
- can define some non-regular languages

Motivation – Overview

Motivation for New Models

- partially discontinuous reading
- partially continuous reading
- explore new possibilities
- more deterministic behavior

n-Parallel Jumping Finite Automata

Based on

Radim Kocman and Alexander Meduna
On Parallel Versions of Jumping Finite Automata
Proceedings of SDOT 2015

n-Parallel JFAs – Parallelism

- heavily used in formal grammars
  (n-parallel grammars, simple matrix grammars, . . .)

- heavily used in formal grammars
  (n-parallel grammars, simple matrix grammars, ...)

Example Derivations

$S \Rightarrow ABC \Rightarrow aAbBcC \Rightarrow aaAbbBccC \Rightarrow aaabbbccc$

- heavily used in formal grammars
  (n-parallel grammars, simple matrix grammars, ...)

Example Derivations

$S \Rightarrow ABC \Rightarrow aAbBcC \Rightarrow aaAbbBccC \Rightarrow aaabbbccc$

- rarely used in classical automata
  (multiple tapes, more heads reading the same input, ...)

- heavily used in formal grammars
  (n-parallel grammars, simple matrix grammars, ...)

Example Derivations

$S \Rightarrow ABC \Rightarrow aAbBcC \Rightarrow aaAbbBccC \Rightarrow aaabbbccc$

- rarely used in classical automata
  (multiple tapes, more heads reading the same input, ...)

- What if the parallelism is combined with the jumping?

Definition

An n-parallel general jumping finite automaton (n-PGJFA) is a quintuple

$$M = (Q, \Sigma, R, S, F)$$

where

- Q is a finite set of states;
- Σ is an input alphabet, $Q \cap \Sigma = \emptyset$;
- R is a finite set of rules: $py \to q$, where $p, q \in Q$, $y \in \Sigma^*$;
- S is a set of start state strings, $S \subseteq Q^n$;
- F is a set of final states.

n-Parallel JFAs – Reading

- arbitrary splits the input into n parts
- steps of all heads are synchronized
- different types of the jumping:
  - unrestricted jumps – each part is processed as in JFA
  - right jumps – each part is processed as in FA

n-Parallel JFAs – Example

Example

Consider the 3-PGJFA

$$M = (\{s, r, p\}, \Sigma, R, \{srp\}, \{s, r, p\}),$$

where $\Sigma = \{a, b, c\}$ and R consists of the rules

$$sa \rightarrow s, \quad rb \rightarrow r, \quad pc \rightarrow p.$$

$L(M, 3{-}r) = \{a^n b^n c^n \mid n \geq 0\}$

Example

Consider the 3-PGJFA

$$M = (\{s, r, p\}, \Sigma, R, \{srp\}, \{s, r, p\}),$$

where $\Sigma = \{a, b, c\}$ and R consists of the rules

$$sa \to s, \quad rb \to r, \quad pc \to p.$$

$L(M, 3{-}r) = \{a^n b^n c^n \mid n \geq 0\}$

Example Steps (for $n = 3$ with only right jumps)

$|aaa|bbb|ccc \curvearrowright |aa|bb|cc \curvearrowright |a|b|c \curvearrowright |||$

Theorem

For every n-PRLG $G = (N_1, \ldots, N_n, T, S1, P)$, there is an n-PGJFA using only right n-jumps $M = (Q, \Sigma, R, S2, F)$, such that $L(M, n\text{–}r) = L(G)$.

Theorem

For every n-PGJFA using only right n-jumps $M = (Q, \Sigma, R, S2, F)$, there is an n-PRLG $G = (N_1, \ldots, N_n, T, S1, P)$, such that $L(G) = L(M, n\text{–}r)$.

Theorem

$_r$1-**PGJFA** $=$ $_r$**GJFA** $=$ **REG**.

Theorem

$_r$2-**PGJFA** $\subset$ **CF**.

Theorem

$_r n$-**PGJFA** $\subset$ **CS** and there exist non-context-free languages in $_r n$-**PGJFA** for all $n > 2$.

Theorem

For all $n \in \mathbb{N}$, $_r n$-**PGJFA** $\subset$ $_r(n+1)$-**PGJFA**.

Double-Jumping Finite Automata

Based on

Radim Kocman, Zbyněk Křivka and Alexander Meduna
On Double-Jumping Finite Automata
Proceedings of NCMA 2016

Definition

A general jumping finite automaton (GJFA) is a quintuple

$$M = (Q, \Sigma, R, s, F)$$

where

- Q is a finite set of states;
- Σ is the input alphabet;
- R is a finite set of rules of the form

$$py \rightarrow q \qquad (p, q \in Q, y \in \Sigma^*)$$

- s is the start state;
- F is a set of final states.

Used Symbols

$\blacktriangleright$ – Right Jump $\blacktriangleleft$ – Left Jump $\blacklozenge$ – Both Directions

Studied Modes

$\blacklozenge\blacklozenge \curvearrowright$ – Unrestricted 2-Jumps

$\blacktriangleright\blacktriangleleft \curvearrowright$ – Right-Left 2-Jumps

$\blacktriangleleft\blacktriangleright \curvearrowright$ – Left-Right 2-Jumps

$\blacktriangleright\blacktriangleright \curvearrowright$ – Right-Right 2-Jumps

$\blacktriangleleft\blacktriangleleft \curvearrowright$ – Left-Left 2-Jumps

Used Symbols

$\blacktriangleright$ – Right Jump $\blacktriangleleft$ – Left Jump $\blacklozenge$ – Both Directions

Studied Modes

$_{\blacklozenge\blacklozenge}\curvearrowright$ – Unrestricted 2-Jumps

$_{\blacktriangleright\blacktriangleleft}\curvearrowright$ – Right-Left 2-Jumps $_{\blacktriangleleft\blacktriangleright}\curvearrowright$ – Left-Right 2-Jumps

$_{\blacktriangleright\blacktriangleright}\curvearrowright$ – Right-Right 2-Jumps $_{\blacktriangleleft\blacktriangleleft}\curvearrowright$ – Left-Left 2-Jumps

Example

$$L(M_{\blacktriangleleft\blacktriangleright\curvearrowright}) = \{uvw \mid u, v, w \in \Sigma^*,\ usvsw\ {}_{\blacktriangleleft\blacktriangleright}\curvearrowright^*\ ff,\ f \in F\}.$$

Conditions for 2-Jumps

- both jumps follow the same rule
- the jumps cannot ever cross each other

Double-JFAs – Conditions

Conditions for 2-Jumps

- both jumps follow the same rule
- the jumps cannot ever cross each other

Example with $_{\blacktriangleleft\blacktriangleright}\curvearrowright$

- configuration: $uu'apvpaw'w$, where $a, u, u', v, w, w' \in \Sigma^*, p \in Q$
- rule: (p, a, q)
- 2-jump: $uu'apvpaw'w \;{}_{\blacktriangleleft\blacktriangleright}\curvearrowright\; uqu'vw'qw$

Properties

- required initial configuration: sxs, where $x \in \Sigma^*$
- cannot jump over any symbols
- every $x \in L(M_{\blacktriangleright\blacktriangleleft\curvearrowright})$ can be written as $x = u_1 u_2 \dots u_n u_n \dots u_2 u_1$, where $n \in \mathbb{N}$, and $u_i \in \Sigma^*$, $1 \leq i \leq n$
- accept string palindromes of even length

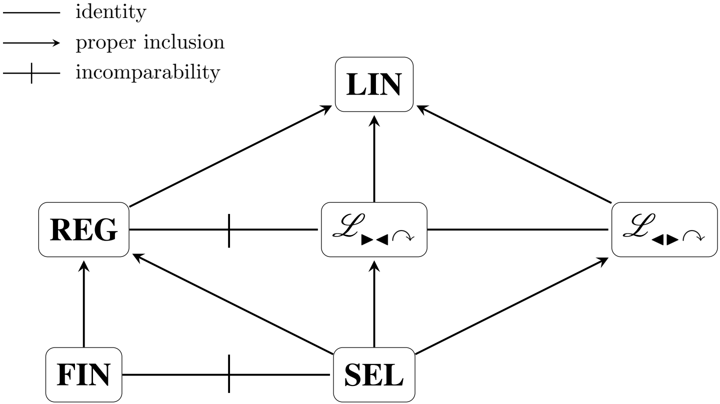
Properties

- required initial configuration: sxs, where $x \in \Sigma^*$
- cannot jump over any symbols
- every $x \in L(M_{\blacktriangleright\blacktriangleleft\curvearrowright})$ can be written as $x = u_1 u_2 \ldots u_n u_n \ldots u_2 u_1$, where $n \in \mathbb{N}$, and $u_i \in \Sigma^*$, $1 \leq i \leq n$
- accept string palindromes of even length

Language Family ($\mathscr{L}_{\blacktriangleright\blacktriangleleft\curvearrowright}$)

- a subfamily of the family of linear languages
- the same as $\mathscr{L}_{\blacktriangleleft\blacktriangleright\curvearrowright}$

Double-JFAs – Comparison



Properties

- the first jump should not skip symbols
- the second jump can skip symbols

Properties

- the first jump should not skip symbols
- the second jump can skip symbols

Example Behavior

- $u_1 u'_1 u_2 u'_2 \dots u_n u'_n$, where $n \in \mathbb{N}$, $u_i, u'_i \in \Sigma$, $u_i = u'_i$, $1 \leq i \leq n$
- red symbols can be also shifted to the right over blue symbols

Properties

- the first jump should not skip symbols
- the second jump can skip symbols

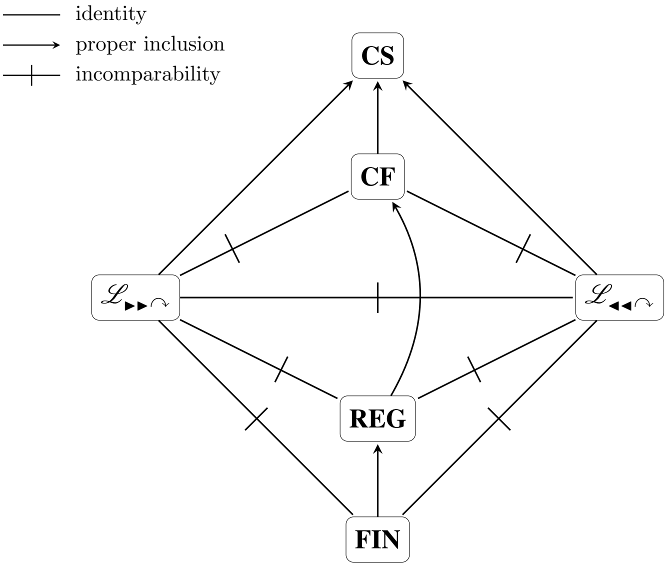
Example Behavior

- $u_1 u'_1 u_2 u'_2 \dots u_n u'_n$, where $n \in \mathbb{N}$, $u_i, u'_i \in \Sigma$, $u_i = u'_i$, $1 \leq i \leq n$
- red symbols can be also shifted to the right over blue symbols

Language Family ($\mathscr{L}_{\blacktriangleright\blacktriangleright\curvearrowright}$)

- a subfamily of the family of context-sensitive languages
- not the same as $\mathscr{L}_{\blacktriangleleft\blacktriangleleft\curvearrowright}$

Double-JFAs – Comparison



Double-JFAs – Closure Properties*

	$\mathscr{L}_{\blacktriangleright\blacktriangleleft\curvearrowright}$, $\mathscr{L}_{\blacktriangleleft\blacktriangleright\curvearrowright}$	$\mathscr{L}_{\blacktriangleright\blacktriangleright\curvearrowright}$	$\mathscr{L}_{\blacktriangleleft\blacktriangleleft\curvearrowright}$
endmarking (both sides)	− (+)	− (−)	− (−)
concatenation	−	−	−
square (L^2)	−	−	−
shuffle	−	−	−
union	+	+	+
complement	−	−	−
intersection	+	−	−
int. with regular languages	+	−	−
mirror image	+	−	−
finite substitution	−	−	−
homomorphism	+	−	−
ε-free homomorphism	+	−	−
inverse homomorphism	−	−	−

One-Way Jumping Finite Automata

Based on

- Hiroyuki Chigahara, Szilárd Zsolt Fazekas and Akihiro Yamamura
  One-way Jumping Finite Automata
  Int. J. Found. Comput. Sci. 27, 391 (2016)

- Szilárd Zsolt Fazekas and Akihiro Yamamura
  On Regular Languages accepted by One-Way Jumping Finite Automata
  Short Papers of NCMA 2016

Definition

A right one-way jumping finite automaton (ROWJFA) is a quintuple $M = (Q, \Sigma, R, s, F)$, where Q, Σ, R, s and F are defined as in a DFA.

Definition

A right one-way jumping finite automaton (ROWJFA) is a quintuple $M = (Q, \Sigma, R, s, F)$, where Q, Σ, R, s and F are defined as in a DFA.

Definition

The right one-way jumping relation, symbolically denoted by $\circlearrowright$, over $Q\Sigma^*$, is defined as follows. Suppose that x and y belong to Σ^*, a belongs to Σ, p and q are states in Q and $pa \rightarrow q \in R$. Then the ROWJFA M makes a jump from the configuration $pxay$ to the configuration qyx, written as

$$pxay \circlearrowright qyx$$

if x belongs to $\{\Sigma \setminus \Sigma_p\}^*$ where $\Sigma_p = \{b \in \Sigma \mid (p, b, q) \in R \text{ for some } q \in Q\}$.

Definition

The language accepted by M, denoted by $L(M)$, is defined as

$$L(M) = \{w \in \Sigma^* \mid sw \circlearrowright^* f, f \in F\}$$

Definition

The language accepted by M, denoted by $L(M)$, is defined as

$$L(M) = \{w \in \Sigma^* \mid sw \circlearrowright^* f, f \in F\}$$

- fully deterministic behavior
- There is also a similar definition for the left one-way jumping finite automaton.

Example 1

Let M_1 be a ROWJFA given by

$$M_1 = (\{q_0, q_1, q_2\}, \Sigma, R, q_0, \{q_0\}),$$

where $\Sigma = \{a, b, c\}$ and R consists of the rules

$$q_0 a \rightarrow q_1, \quad q_1 b \rightarrow q_2, \quad q_2 c \rightarrow q_0.$$

$L(M_1) = \{w \in \Sigma \mid |w|_a = |w|_b = |w|_c\}$

Example 1

Let M_1 be a ROWJFA given by

$$M_1 = (\{q_0, q_1, q_2\}, \Sigma, R, q_0, \{q_0\}),$$

where $\Sigma = \{a, b, c\}$ and R consists of the rules

$$q_0 a \rightarrow q_1, \quad q_1 b \rightarrow q_2, \quad q_2 c \rightarrow q_0.$$

$L(M_1) = \{w \in \Sigma \mid |w|_a = |w|_b = |w|_c\}$

Example 2

Let M_2 be a ROWJFA given by

$$M_2 = (\{q_0, q_1\}, \Sigma, R, q_0, \{q_0, q_1\}),$$

where $\Sigma = \{a, b\}$ and R consists of the rules

$$q_0 a \rightarrow q_0, \quad q_0 b \rightarrow q_1, \quad q_1 b \rightarrow q_1.$$

$L(M_2) = a^* b^*$

Theorem

ROWJ properly includes **REG**.

Theorem

ROWJ and **LOWJ** are incomparable.

Theorem

ROWJ $\not\subset$ **JFA**.

Theorem

CF and **ROWJ** are incomparable.

Theorem

The class **ROWJ** is not closed under

- intersection,
- concatenation,
- reversal,
- intersection with regular languages,
- concatenation with regular languages,
- substitution,
- Kleene star,
- Kleene plus.

Theorem

Let M be a ROWJFA. If there exists a constant k, such that for any word $w \in L(M)$ the number of sweeps needed by M to process w is at most k, then the language $L(M)$ is regular.

Overall Conclusion

n-Parallel Jumping Finite Automata
– combination of the parallel and jumping behavior

Double-Jumping Finite Automata
– parallel combination of different jumping modes

One-Way Jumping Finite Automata
– fully deterministic behavior

Thank you for your attention!