Descriptional Complexity of Some Regulated Rewriting Grammars

Dr. Lakshmanan Kuppusamy
School of Comp. Sci. & Engg., VIT Vellore, INDIA.

email: klakshma@vit.ac.in

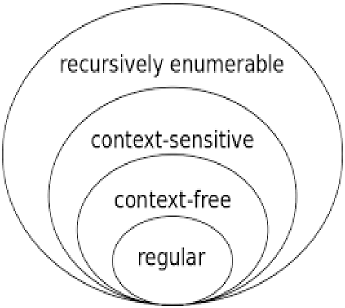
18th Nov 2019

Outline of the Talk

Chomsky hierarchy

Class	Grammars	Languages	Automaton
Type-0	Unrestricted	Recursive Enumerable	Turing Machine
Type-1	Context Sensitive	Context Sensitive	Linear-Bound
Type-2	Context Free	Context Free	Pushdown
Type-3	Regular	Regular	Finite

Chomsky Hierarchy



Courtesy: Google Images

Motivation and Objective

- Can we describe a type-0 language using type-2 grammar? (i.e., context-free grammars) Obviously NO
- Can we generate recursively enumerable languages (RE) using context-free rules along with some tools Ans: Yes.
- What additional tool(s) can be used to achieve the above?

Motivation and Objective

- Can we describe a type-0 language using type-2 grammar? (i.e., context-free grammars) Obviously NO
- Can we generate recursively enumerable languages (RE) using context-free rules along with some tools Ans: Yes.
- What additional tool(s) can be used to achieve the above?
- One is context-based restriction and the other is rule based restriction.
  - Semi-conditional grammars, generalized forbidding grammars
  - Graph-controlled grammars, Matrix grammars, etc.

Motivation and Objective

- Can we describe a type-0 language using type-2 grammar? (i.e., context-free grammars) Obviously NO
- Can we generate recursively enumerable languages (RE) using context-free rules along with some tools Ans: Yes.
- What additional tool(s) can be used to achieve the above?
- One is context-based restriction and the other is rule based restriction.
  - Semi-conditional grammars, generalized forbidding grammars
  - Graph-controlled grammars, Matrix grammars, etc.
- Question: What and how much resources are required for grammars to generate RE? Is that optimal/succinct?
- resources meant the component size that require to describe the system, thus, called descriptional complexity measures.

Semi-conditional grammars

A semi-conditional grammar of degree (i, j) is $G = (N, T, S, P)$, where P is a finite set of rules of the form $(A \to x, \alpha, \beta)$, where

- $A \to x$ is a context-free rule,
- $\alpha, \beta = \phi$ or $\alpha, \beta \in (N \cup T)^*$ and
- $|\alpha| \leq i$, $|\beta| \leq j$.

Semi-conditional grammars

A semi-conditional grammar of degree (i, j) is $G = (N, T, S, P)$, where P is a finite set of rules of the form $(A \to x, \alpha, \beta)$, where

- $A \to x$ is a context-free rule,
- $\alpha, \beta = \phi$ or $\alpha, \beta \in (N \cup T)^*$ and
- $|\alpha| \leq i$, $|\beta| \leq j$.

A rule $(A \to x, \alpha, \beta)$ can be applied to a string w if and only if

- α (when $\alpha \neq \phi$) is a substring of w (permitting context) and
- β (when $\beta \neq \phi$) is not a substring of w (forbidding context).
- If $\alpha = \phi$, $\beta = \phi$, the rule is applied without any restriction.

Semi-conditional grammars

A semi-conditional grammar of degree (i, j) is $G = (N, T, S, P)$, where P is a finite set of rules of the form $(A \to x, \alpha, \beta)$, where

- $A \to x$ is a context-free rule,
- $\alpha, \beta = \phi$ or $\alpha, \beta \in (N \cup T)^*$ and
- $|\alpha| \leq i$, $|\beta| \leq j$.

A rule $(A \to x, \alpha, \beta)$ can be applied to a string w if and only if

- α (when $\alpha \neq \phi$) is a substring of w (permitting context) and
- β (when $\beta \neq \phi$) is not a substring of w (forbidding context).
- If $\alpha = \phi$, $\beta = \phi$, the rule is applied without any restriction.
- A rule is applied based on the presence of the permitting string and the absence of the forbidden string in the current sentencial form.
- As usual, $w \in T^*$ is collected for languages.

Variants of Semi-conditional grammars

A semi-conditional grammar is called

- Random Context Grammar: if degree $(i, j) = (1, 1)$.
- Simple: If either $\alpha = \phi$ or $\beta = \phi$ in every rule of P.
- Permitting Grammar: if degree $= (i, 0)$
  Here $\beta = \phi$ in every rule of P.
- Forbidding Grammar: if degree $= (0, j)$
  Here $\alpha = \phi$ in every rule of P.

Variants of Semi-conditional grammars

A semi-conditional grammar is called

- Random Context Grammar: if degree $(i, j) = (1, 1)$.
- Simple: If either $\alpha = \phi$ or $\beta = \phi$ in every rule of P.
- Permitting Grammar: if degree $= (i, 0)$
  Here $\beta = \phi$ in every rule of P.
- Forbidding Grammar: if degree $= (0, j)$
  Here $\alpha = \phi$ in every rule of P.

A Forbidding Rule: $(A \rightarrow x, \beta)$

- $A \rightarrow x$ is a context-free rule,
- $\beta = \phi$ or $\beta \in (N \cup T)^*$ [β is a string]

An Example of Simple Random Context Grammar

Example

$G = (\{S, A, X, C, Y, a, b, c\}, \{a, b, c\}, S, P)$ where P is :
1. $(S \to AC, \phi, \phi)$, 2. $(C \to Y, A, \phi)$, 3. $(A \to aXb, Y, \phi)$,
4. $(Y \to Cc, \phi, A)$, 5. $(X \to A, C, \phi)$ 6. $(A \to ab, Y, \phi)$,
7. $(Y \to c, \phi, A)$.

An Example of Simple Random Context Grammar

Example

$G = (\{S, A, X, C, Y, a, b, c\}, \{a, b, c\}, S, P)$ where P is :
1. $(S \rightarrow AC, \phi, \phi)$, 2. $(C \rightarrow Y, A, \phi)$, 3. $(A \rightarrow aXb, Y, \phi)$,
4. $(Y \rightarrow Cc, \phi, A)$, 5. $(X \rightarrow A, C, \phi)$ 6. $(A \rightarrow ab, Y, \phi)$,
7. $(Y \rightarrow c, \phi, A)$.

$$S \Rightarrow_1 AC \Rightarrow_2 AY \Rightarrow_3 aXbY \Rightarrow_4 aXbCc \Rightarrow_5 aAbCc$$

An Example of Simple Random Context Grammar

Example

$G = (\{S, A, X, C, Y, a, b, c\}, \{a, b, c\}, S, P)$ where P is :
1. $(S \rightarrow AC, \phi, \phi)$, 2. $(C \rightarrow Y, A, \phi)$, 3. $(A \rightarrow aXb, Y, \phi)$,
4. $(Y \rightarrow Cc, \phi, A)$, 5. $(X \rightarrow A, C, \phi)$ 6. $(A \rightarrow ab, Y, \phi)$,
7. $(Y \rightarrow c, \phi, A)$.

$$S \Rightarrow_1 AC \Rightarrow_2 AY \Rightarrow_3 aXbY \Rightarrow_4 aXbCc \Rightarrow_5 aAbCc \Rightarrow^{n-2}_{2,3,4,5}$$

An Example of Simple Random Context Grammar

Example

$G = (\{S, A, X, C, Y, a, b, c\}, \{a, b, c\}, S, P)$ where P is :
1. $(S \to AC, \phi, \phi)$, 2. $(C \to Y, A, \phi)$, 3. $(A \to aXb, Y, \phi)$,
4. $(Y \to Cc, \phi, A)$, 5. $(X \to A, C, \phi)$ 6. $(A \to ab, Y, \phi)$,
7. $(Y \to c, \phi, A)$.

$S \Rightarrow_1 AC \Rightarrow_2 AY \Rightarrow_3 aXbY \Rightarrow_4 aXbCc \Rightarrow_5 aAbCc \Rightarrow_{2,3,4,5}^{n-2}$
$a^{n-1}Ab^{n-1}Cc^{n-1} \Rightarrow_2 a^nAb^nYc^{n-1} \Rightarrow_6 a^nb^nYc^{n-1} \Rightarrow_7 a^nb^nc^n$.

An Example of Simple Random Context Grammar

Example

$G = (\{S, A, X, C, Y, a, b, c\}, \{a, b, c\}, S, P)$ where P is :
1. $(S \to AC, \phi, \phi)$, 2. $(C \to Y, A, \phi)$, 3. $(A \to aXb, Y, \phi)$,
4. $(Y \to Cc, \phi, A)$, 5. $(X \to A, C, \phi)$ 6. $(A \to ab, Y, \phi)$,
7. $(Y \to c, \phi, A)$.

$S \Rightarrow_1 AC \Rightarrow_2 AY \Rightarrow_3 aXbY \Rightarrow_4 aXbCc \Rightarrow_5 aAbCc \Rightarrow^{n-2}_{2,3,4,5}$
$a^{n-1}Ab^{n-1}Cc^{n-1} \Rightarrow_2 a^nAb^nYc^{n-1} \Rightarrow_6 a^nb^nYc^{n-1} \Rightarrow_7 a^nb^nc^n$.

Language and Degree

- $L(G) = \{a^nb^nc^n \mid n \geq 1\}$: a context-sensitive language.
- #Conditional Productions =6.
- #Non-terminals=5, Degree = $(1, 1)$.

Semi-Conditional Grammars and RE

Degree (i, j)	# Non-Terminals n	# Conditional Productions c	References
$(2, 1)$	7	6	Okubo, IPL, 2009
	6	$7 + \|P_{cf}\|$	We @ CiE 2018
$(2, 2)$	6	7	We @ CiE 2018
$(3, 1)$	6	13	We @ CiE 2018

Usual ambition: How small the four parameters (i, j, n, c) could be for a semi-conditional grammar to describe RE?

Semi-Conditional Grammars and RE

Degree (i, j)	# Non-Terminals n	# Conditional Productions c	References
$(2, 1)$	7	6	Okubo, IPL, 2009
	6	$7 + \|P_{cf}\|$	We @ CiE 2018
$(2, 2)$	6	7	We @ CiE 2018
$(3, 1)$	6	13	We @ CiE 2018

Usual ambition: How small the four parameters (i, j, n, c) could be for a semi-conditional grammar to describe RE?
Usual Technique to show $SC(i, j; n; c)$=RE: Consider a type-0 grammar for RE in one of the variants of Geffert Normal Form; produce SCG rules of prescribed size to simulate the assumed GNF.

Geffert Normal Form: $(5, 2)$-GNF

A type-0 grammar G is said to be in Geffert Normal Form if all of its production rules are of the form

- $S \to uSa, S \to uSv,$
  $\mathrm{S} \to uv,$
- $AB \to \lambda$, $CD \to \lambda$.

where S is the initial nonterminal and A, B, C, D are nonterminals and $u \in \{A, C\}$, $v \in \{B, D\}^*$.
Only 5 nonterminals are used
But no control on the length of the RHS of S.

Geffert Normal Form: $(5, 2)$-GNF

A type-0 grammar G is said to be in Geffert Normal Form if all of its production rules are of the form

- $S \rightarrow uSa, S \rightarrow uSv$, S $\rightarrow uv$,
- $AB \rightarrow \lambda$, $CD \rightarrow \lambda$.

where S is the initial nonterminal and A, B, C, D are nonterminals and $u \in \{A, C\}$, $v \in \{B, D\}^*$.
Only 5 nonterminals are used
But no control on the length of the RHS of S.

A type-0 grammar G is said to be in Special Geffert Normal Form if all of its production rules are of the form

- $X \rightarrow bY$, $X \rightarrow Yb$, $S' \rightarrow \lambda$,
- $AB \rightarrow \lambda$, $CD \rightarrow \lambda$.

where $N = N_1 \cup N_2$, $\{X, Y, S, S'\} \subseteq N_1$, $\{A, B, C, D\} = N_2$ and $b \in N_2 \cup T$.
control on the length of the RHS of CF rule; length is 2 only.

Variants of GNF($S \to v$, $AB \to \lambda$, $CD \to \lambda$)

$(4, 1)$-GNF

If $\phi_1(A) = AB$, $\phi_1(B) = C$, $\phi_1(C) = A$, $\phi_1(D) = BC$, then

- $S \to v$,
- $ABC \to \lambda$.

$(4, 2)$-GNF

If $\phi_2(A) = CAA$, $\phi_2(B) = BBC$, $\phi_2(C) = CA$, $\phi_2(D) = BC$, then

- $S \to v$,
- $AB \to \lambda$, $CC \to \lambda$.

Variants of GNF($S \to v$, $AB \to \lambda$, $CD \to \lambda$)

(4, 1)-GNF

If $\phi_1(A) = AB$, $\phi_1(B) = C$, $\phi_1(C) = A$, $\phi_1(D) = BC$, then

- $S \to v$,
- $ABC \to \lambda$.

(4, 2)-GNF

If $\phi_2(A) = CAA$, $\phi_2(B) = BBC$, $\phi_2(C) = CA$, $\phi_2(D) = BC$, then

- $S \to v$,
- $AB \to \lambda$, $CC \to \lambda$.

(3, 1)-GNF

If $\phi_3(A) = ABB$, $\phi_3(B) = BA$, $\phi_3(C) = AB$, $\phi_3(D) = BBA$, then

- $S \to v$,
- $ABBBA \to \lambda$.

(3, 2)-GNF

- $S \to v$,
- $AA \to \lambda$, $BBB \to \lambda$.

SC(2, 1; 7, 6)=RE : Okubo, IPL, 2009

$S \to w$ is simulated by
$S \to w, 0, 0$

SC(2, 1; 7, 6)=RE : Okubo, IPL, 2009

$S \rightarrow w$ is simulated by
$S \rightarrow w, 0, 0$

1. $A \rightarrow \$, AB, \$$
2. $B \rightarrow \#, \$B, \#$
3. $C \rightarrow \$, CD, \$$
4. $D \rightarrow \#, \$D, \#$
5. $\$ \rightarrow \lambda, \$\#, \lambda$
6. $\# \rightarrow \lambda, 0, \$$

N=$\{S, A, B, C, D, \$, \#\}$
Normal Form is (5,2)-GNF
$(AB \rightarrow \lambda, CD \rightarrow \lambda)$

SC(2, 1; 7, 6)=RE : Okubo, IPL, 2009

$S \rightarrow w$ is simulated by
$S \rightarrow w, 0, 0$

1. $A \rightarrow \$, AB, \$$
2. $B \rightarrow \#, \$B, \#$
3. $C \rightarrow \$, CD, \$$
4. $D \rightarrow \#, \$D, \#$
5. $\$ \rightarrow \lambda, \$\#, \lambda$
6. $\# \rightarrow \lambda, 0, \$$

N=$\{S, A, B, C, D, \$, \#\}$
Normal Form is (5,2)-GNF
$(AB \rightarrow \lambda, CD \rightarrow \lambda)$

Sample Simulation for $AB \rightarrow \lambda$

$AB \Rightarrow_1 \$B \Rightarrow_2 \$\# \Rightarrow_5 \# \Rightarrow_6 \lambda$

SC(2, 1; 7, 6)=RE : Okubo, IPL, 2009

$S \to w$ is simulated by
$S \to w, 0, 0$

1. $A \to \$, AB, \$$
2. $B \to \#, \$B, \#$
3. $C \to \$, CD, \$$
4. $D \to \#, \$D, \#$
5. $\$ \to \lambda, \$\#, \lambda$
6. $\# \to \lambda, 0, \$$

N=$\{S, A, B, C, D, \$, \#\}$
Normal Form is (5,2)-GNF
$(AB \to \lambda, CD \to \lambda)$

Sample Simulation for $AB \to \lambda$

$AB \Rightarrow_1 \$B \Rightarrow_2 \$\# \Rightarrow_5 \# \Rightarrow_6 \lambda$

Sample Simulation for $CD \to \lambda$

$CD \Rightarrow_3 \$D \Rightarrow_4 \$\# \Rightarrow_5 \# \Rightarrow_6 \lambda$

$SC(2,1;6,7+|P_{cf}|)$=RE : CiE 2018

$S \rightarrow w$ is simulated by

$P_{cf} : S \rightarrow w, 0, \$$ plus

1. $A \rightarrow \$S,\ AB, \$$
2. $B \rightarrow \#,\ \$S,\ \#$
3. $S \rightarrow \$,\ S\#,\ 0$
4. $C \rightarrow \$\$,\ CC,\ \$$
5. $C \rightarrow \#,\ \$\$,\ \#$
6. $\$ \rightarrow \lambda,\ \$\#,\ 0$
7. $\# \rightarrow \lambda,\ 0,\ \$$

N=$\{S, A, B, C, \$, \#\}$

NF: $AB \rightarrow \lambda,\ \ CC \rightarrow \lambda$

SC$(2,1;6,7+|P_{cf}|)$=RE : CiE 2018

$S \rightarrow w$ is simulated by

P_{cf} : $S \rightarrow w, 0, \$$ plus

1. $A \rightarrow \$S,\ AB, \$$
2. $B \rightarrow \#,\ \$S,\ \#$
3. $S \rightarrow \$,\ S\#,\ 0$
4. $C \rightarrow \$\$,\ CC,\ \$$
5. $C \rightarrow \#,\ \$\$,\ \#$
6. $\$ \rightarrow \lambda,\ \$\#,\ 0$
7. $\# \rightarrow \lambda,\ 0,\ \$$

N=$\{S, A, B, C, \$, \#\}$

NF: $AB \rightarrow \lambda,\ CC \rightarrow \lambda$

Sample Simulation for $AB \rightarrow \lambda$

$AB \Rightarrow_1 \$SB \Rightarrow_2 \$S\# \Rightarrow_3$
$\$\$\# \Rightarrow_6 \$\# \Rightarrow_6 \# \Rightarrow_7 \lambda$

SC(2, 1; 6, 7 + $|P_{cf}|$)=RE : CiE 2018

$S \to w$ is simulated by
$P_{cf}: S \to w, 0, \$$ plus

1. $A \to \$S,\ AB, \$$
2. $B \to \#,\ \$S,\ \#$
3. $S \to \$,\ S\#,\ 0$
4. $C \to \$\$,\ CC,\ \$$
5. $C \to \#,\ \$\$,\ \#$
6. $\$ \to \lambda,\ \$\#,\ 0$
7. $\# \to \lambda,\ 0,\ \$$

N=$\{S, A, B, C, \$, \#\}$
NF: $AB \to \lambda,\ CC \to \lambda$

Sample Simulation for $AB \to \lambda$

$AB \Rightarrow_1 \$SB \Rightarrow_2 \$S\# \Rightarrow_3$
$\$\$\# \Rightarrow_6 \$\# \Rightarrow_6 \# \Rightarrow_7 \lambda$

Sample Simulation for $CC \to \lambda$

$CC \Rightarrow_4 \$\$C \Rightarrow_5 \$\$\# \Rightarrow_6$
$\$\# \Rightarrow_6 \# \Rightarrow_7 \lambda$

SC(2, 2; 6, 7)=RE : CiE 2018

$S \to w$ is simulated by
$S \to w, 0, 0$

1. $A \to \$, AB, \$$
2. $\$ \to \$\$,\ \$B,\ C\#$
3. $B \to \#,\ \$\$,\ \#$
4. $C \to \#\$,\ CC,\ \#$
5. $C \to \#\#,\ \$C,\ \#\#$
6. $\$ \to \lambda,\ \$\#,\ AB$
7. $\# \to \lambda,\ 0,\ \$$

N=$\{S, A, B, C, \$, \#\}$
NF is (4,2)-GNF:
$AB \to \lambda, CC \to \lambda$

SC(2, 2; 6, 7)=RE : CiE 2018

$S \rightarrow w$ is simulated by
$S \rightarrow w, 0, 0$

1. $A \rightarrow \$, AB, \$$
2. $\$ \rightarrow \$\$,\ \$B,\ C\#$
3. $B \rightarrow \#,\ \$\$,\ \#$
4. $C \rightarrow \#\$,\ CC,\ \#$
5. $C \rightarrow \#\#,\ \$C,\ \#\#$
6. $\$ \rightarrow \lambda,\ \$\#,\ AB$
7. $\# \rightarrow \lambda,\ 0,\ \$$

N=$\{S, A, B, C, \$, \#\}$
NF is (4,2)-GNF:
$AB \rightarrow \lambda, CC \rightarrow \lambda$

Sample Simulation for $AB \rightarrow \lambda$

$AB \Rightarrow_1 \$B \Rightarrow_2 \$\$B \Rightarrow_3 \$\$\# \Rightarrow_6 \$\# \Rightarrow_6 \# \Rightarrow_7 \lambda$

SC(2, 2; 6, 7)=RE : CiE 2018

$S \to w$ is simulated by
$S \to w, 0, 0$

1. $A \to \$, AB, \$$
2. $\$ \to \$\$,\ \$B,\ C\#$
3. $B \to \#,\ \$\$,\ \#$
4. $C \to \#\$,\ CC,\ \#$
5. $C \to \#\#,\ \$C,\ \#\#$
6. $\$ \to \lambda,\ \$\#,\ AB$
7. $\# \to \lambda,\ 0,\ \$$

$N = \{S, A, B, C, \$, \#\}$
NF is (4,2)-GNF:
$AB \to \lambda, CC \to \lambda$

Sample Simulation for $AB \to \lambda$

$AB \Rightarrow_1 \$B \Rightarrow_2 \$\$B \Rightarrow_3 \$\$\# \Rightarrow_6 \$\# \Rightarrow_6 \# \Rightarrow_7 \lambda$

Sample Simulation for $CC \to \lambda$

$CC \Rightarrow_4 \#\$C \Rightarrow_5 \#\$\#\# \Rightarrow_6 \#\#\# \Rightarrow_7^3 \lambda$

Simple Semi-Conditional Grammars and RE

Degree (i, j)	# Non-Terminals n	# Conditional Productions c	References
$(2, 1)$	10	9	T. Masopust, 2007
	9	9	MCU'18
$(3, 1)$	9	8	Okubo, IPL, 2009
	7	7	MCU'18, submitted to FI
$(4, 1)$	7	6	MCU'18, submitted to FI
	6	8	We, submitted to FI

SSC(2, 1; 9, 9)=RE : MCU 2018

$S \to w$ is simulated by

$S \to w, 0, 0$

1. $A \to \#\$A'$,0,$A'$
2. $B \to B'$, 0, B'
3. $C \to C'\$\#$, 0, $\#$
4. $A' \to B'$, $A'B'$, 0
5. $C' \to \lambda$,$B'B'$,0
6. $B' \to \lambda$, $B'\$$,0
7. $\$ \to \#$, $\$\$$,0
8. $\$ \to \#$, $\#\#$,0
9. $\# \to \lambda$, 0, $\$$

N={$S, A, B, C, A', B', C', \$, \#$}

SSC(2, 1; 9, 9)=RE : MCU 2018

$S \to w$ is simulated by

$S \to w, 0, 0$

1. $A \to \#\$A'$,0,$A'$
2. $B \to B'$, 0, B'
3. $C \to C'\$\#$, 0, $\#$
4. $A' \to B'$, $A'B'$, 0
5. $C' \to \lambda$,$B'B'$,0
6. $B' \to \lambda$, $B'\$$,0
7. $\$ \to \#$, $\$\$$,0
8. $\$ \to \#$, $\#\#$,0
9. $\# \to \lambda$, 0, $\$$

N={$S, A, B, C, A', B', C', \$, \#$}

Sample Simulation for $ABC \to \lambda$

$ABC \Rightarrow_1 \#\$A'BC \Rightarrow_2$
$\$\#\$A'B'C \Rightarrow_3 \#\$A'B'C'\$\#$
$\Rightarrow_4 \#\$B'B'C'\$\# \Rightarrow_5$
$\#\$B'B'B'\$\# \Rightarrow_6^3 \#\$\$\#$
$\Rightarrow_7 \#\$\#\# \Rightarrow_8 \#^4 \Rightarrow_9^4 \lambda$

SSC(3, 1; 9, 8)=RE : Okubo, IPL, 2009

Okubo's rules

1. $A \rightarrow A'$,0,A'
2. $B \rightarrow B'$,0, B'
3. $C \rightarrow C'$, 0, C'
4. $A' \rightarrow \$$, $A'B'C'$,0
5. $C' \rightarrow \#$, $\$B'C'$, 0
6. $B' \rightarrow \lambda$, $\$B'\#$, 0
7. $\$ \rightarrow \lambda$, $\$\#$, 0
8. $\# \rightarrow \lambda$, 0, $\#$

NF: (4,1)-GNF

SSC(3, 1; 9, 8)=RE : Okubo, IPL, 2009

Okubo's rules

1. $A \rightarrow A'$,0,A'
2. $B \rightarrow B'$,0, B'
3. $C \rightarrow C'$, 0, C'
4. $A' \rightarrow \$$, $A'B'C'$,0
5. $C' \rightarrow \#$, $\$B'C'$, 0
6. $B' \rightarrow \lambda$, $\$B'\#$, 0
7. $\$ \rightarrow \lambda$, $\$\#$, 0
8. $\# \rightarrow \lambda$, 0, $\#$

NF: (4,1)-GNF

Aim: to avoid $\#, \$$

1. $A \rightarrow A'$,0,A'
2. $B \rightarrow B'$,0, B'
3. $C \rightarrow C'$, 0, C'
4. $B' \rightarrow \lambda$, $A'B'C'$, 0
5. $A' \rightarrow \lambda$, $A'C'$, 0
6. $C' \rightarrow \lambda$, 0,A'

SSC(3, 1; 9, 8)=RE : Okubo, IPL, 2009

Okubo's rules

1. $A \rightarrow A'$,0,A'
2. $B \rightarrow B'$,0, B'
3. $C \rightarrow C'$, 0, C'
4. $A' \rightarrow \$$, $A'B'C'$,0
5. $C' \rightarrow \#$, $\$B'C'$, 0
6. $B' \rightarrow \lambda$, $\$B'\#$, 0
7. $\$ \rightarrow \lambda$, $\$\#$, 0
8. $\# \rightarrow \lambda$, 0, $\#$

NF: (4,1)-GNF

Aim: to avoid $\#, \$$

1. $A \rightarrow A'$,0,A'
2. $B \rightarrow B'$,0, B'
3. $C \rightarrow C'$, 0, C'
4. $B' \rightarrow \lambda$, $A'B'C'$, 0
5. $A' \rightarrow \lambda$, $A'C'$, 0
6. $C' \rightarrow \lambda$, 0,A'

Unintended Simulation for $AC \rightarrow \lambda$

$AC \Rightarrow_{1,3} A'C' \Rightarrow_5 C' \Rightarrow_6 \lambda$

SSC(3, 1; 7, 7)=RE

Recall Okubo's rules

1. $A \to A'$,0,A'
2. $B \to B'$,0, B'
3. $C \to C'$, 0, C'
4. $A' \to \$$, $A'B'C'$,0
5. $C' \to \#$, $\$B'C'$, 0
6. $B' \to \lambda$, $\$B'\#$, 0
7. $\$ \to \lambda$, $\$\#$, 0
8. $\# \to \lambda$, 0, $\#$

Avoiding $B', \$$ (MCU'18)

1. $A \to \#A'$,0,$\#$
2. $B \to A'$,0, A'
3. $C \to C'$, 0, C'
4. $C' \to \#$, $A'A'C'$, 0
5. $A' \to \lambda$, $A'A'\#$,0
6. $A' \to \lambda$, $\#A'\#$, 0
7. $\# \to \lambda$, 0, A'

SSC(3, 1; 7, 7)=RE

Recall Okubo's rules

1. $A \to A'$,0,A'
2. $B \to B'$,0, B'
3. $C \to C'$, 0, C'
4. $A' \to \$$, $A'B'C'$,0
5. $C' \to \#$, $\$B'C'$, 0
6. $B' \to \lambda$, $\$B'\#$, 0
7. $\$ \to \lambda$, $\$\#$, 0
8. $\# \to \lambda$, 0, $\#$

Avoiding $B', \$$ (MCU'18)

1. $A \to \#A'$,0,$\#$
2. $B \to A'$,0, A'
3. $C \to C'$, 0, C'
4. $C' \to \#$, $A'A'C'$, 0
5. $A' \to \lambda$, $A'A'\#$,0
6. $A' \to \lambda$, $\#A'\#$, 0
7. $\# \to \lambda$, 0, A'

Intended Simulation for $ABC \to \lambda$

$ABC \Rightarrow_2 AA'C \Rightarrow_{1,3} \#A'A'C' \Rightarrow_4$

Generalized Forbidding Grammar

- Introduced by Meduna in 1990. Its a context-free rule, where each rule is regulated with finitely many forbidding strings.
- A nonterminal is rewritten by a rule only if none of the forbidding strings of the rule occur in the sentential form.
- GFG is a quadruple $G = (V, T, P, S)$ where
  - V is the total alphabet, $T \subset V$ is the terminal alphabet,
  - $S \in V \setminus T$ is the start symbol and P is a set of rules
  - Rule Form: $(A \to x, F)$, where $A \in V \setminus T$, $x \in V^*$, $F \subseteq (N \cup T)^+ \to$ a finite set of forbidding words
- Every RE language can be generated by some GF grammar whose forbidding strings have length (degree) at most two but not one.
- i.e., GF(2)=RE but, GF(1) = Fordibben random context gr. $= \subsetneq$ RE.

GF(d, i, n, c): The resources

The language family $\mathrm{GF}(d, i, n, c)$ is defined as follows:
$L \in \mathrm{GF}(d, i, n, c)$ iff there is a GFG, $G = (V, T, P, S)$ such that:

1. $L = L(G)$,
2. $d \geq d(G) := \max_{(A \to x, F) \in P} \max_{f \in F} |f|$,
   Informally: d is maximum length of the strings in a forbidding set, called degree
3. $i \geq i(G) := \max_{(A \to x, F) \in P} |F|$,
   Informally: i is maximum number of elements of a forbidding set, called index
4. $n \geq |V \setminus T|$, the number or nonterminals in G,
5. $c \geq |P_c|$, the number of conditional rules in G.

Describing RE by GF(d, i, n, c)

Generalized Forbidding Grammars and RE

(d, i)	# Non-Terminals n	# Conditional Productions c	References
$(2, 6)$	9	10	Masopust, Meduna, 2007
	8	6	We, submitted to DAM
$(2, 5)$	8	8	We, @ CALDAM 2019
	9	7	We, submitted to DAM
$(2, 4)$	10	11	Masopust, Meduna, 2007
	9	9	We, @ CALDAM 2019
	8	8	We, submitted to DAM
$(2, 3)$	20	18	We, submitted to DAM

Simulation technique

- Consider a type-0 grammar (in some GNF) starting with S.
- Consider a new starting variable S' (in G') such that $S' \to \sigma S \sigma$
- $S \to g(x)$ whenever $S \to x$ (unconditional rules).
- $g(\gamma) = \begin{cases} \sigma\gamma\sigma \text{ if } \gamma \in T \\ \gamma \text{ if } \gamma \in V \end{cases}$
- By induction, $S' \Rightarrow^* \sigma u \alpha v \sigma t \sigma$ where $t \in (T \cup \{\sigma\})^*$ and
- If the grammar is in $(4,1)$-GNF, then
  - $u \in \{A, AB\}^*$, $v \in \{BC, C\}^*$,
  - $\alpha \in \{AC, ABBC, ABC\}$ (the central part),
- If the grammar is in $(5,2)$-GNF, then
  - $u \in \{A, C\}^*$, $v \in \{B, D\}^*$,
  - $\alpha \in \{AB, CD, AD, CB\}$ (the central part).

GF(2, 6, 8, 6)=RE

Simulating rules for $ABC \to \lambda$

1: $(B \to \$\ ,\ \{S, AC, BB, \$, \#\})$
2: $(A \to \#\$,\ \{S, AC, BB, \#\})$
3: $(C \to \$\#,\ \{S, AC, BB, \$\#, C\#, \#C\})$
4: $(\$ \to \lambda\ ,\ \{A\$, \$C, \$\sigma, \sigma\$, B\$, \$B\})$
5: $(\# \to \lambda,\ \{\$, \#A, C\#\})$
6: $(\sigma \to \lambda\ ,\ \{A, B, C, \#, \$, S\})$

GF(2, 6, 8, 6)=RE

Simulating rules for $ABC \to \lambda$

1: $(B \to \$\ \ ,\ \{S, AC, BB, \$, \#\})$
2: $(A \to \#\$,\ \{S, AC, BB, \#\})$
3: $(C \to \$\#,\ \{S, AC, BB, \$\#, C\#, \#C\})$
4: $(\$ \to \lambda\ ,\ \{A\$, \$C, \$\sigma, \sigma\$, B\$, \$B\})$
5: $(\# \to \lambda,\ \{\$, \#A, C\#\})$
6: $(\sigma \to \lambda\ ,\ \{A, B, C, \#, \$, S\})$

Simulating $ABC \to \lambda$

$$\sigma u ABC v\sigma t\sigma \Rightarrow_1 \sigma u A\$C v\sigma t\sigma \Rightarrow_2 \sigma u\#\$\$C v\sigma t\sigma \Rightarrow_3$$
$$\sigma u\#\$\$\$\# v\sigma t\sigma \Rightarrow_4^3 \sigma u\#\# v\sigma t\sigma \Rightarrow_5^2 \sigma uv\sigma t\sigma.$$

- $ABA \Rightarrow_1 A\$A \Rightarrow_2 \#\$\$A \Rightarrow_4^2 \#A \not\Rightarrow_5$
- $ABB \not\Rightarrow_{1,2,3}$
- $BBC \not\Rightarrow_{1,2,3}$
- $CBC \Rightarrow_1 C\$C \Rightarrow_3 \$\#\$C \not\Rightarrow_4$

GF(2, 5, 9, 7)=RE

Recalling simulating rules of GF(2, 6, 8, 6)=RE

1: $(B \rightarrow \$ \;, \{S, AC, BB, \$, \#\})$
2: $(A \rightarrow \#\$, \{S, AC, BB, \#\})$
3: $(C \rightarrow \$\#, \{S, AC, BB, \$\#, C\#, \#C\})$
4: $(\$ \rightarrow \lambda \;, \{A\$, \$C, \$\sigma, \sigma\$, B\$, \$B\})$
5: $(\# \rightarrow \lambda, \{\$, \#A, C\#\})$
6: $(\sigma \rightarrow \lambda \;, \{A, B, C, \#, \$, S\})$

Simulating rules of GF(2, 5, 9, 7)=RE

1: $(B \rightarrow \dagger \;, \{S, AC, BB, \dagger, \#\})$
2: $(A \rightarrow \#\$, \{S, AC, BB, \#\})$
3: $(C \rightarrow \$\#, \{S, AC, BB, \$\#, \#C\})$
4: $(\dagger \rightarrow \lambda \;, \{A\dagger, \dagger C, C\dagger, \dagger\sigma, \sigma\dagger\})$
5: $(\$ \rightarrow \lambda \;, \{\dagger, \$A, \$B, B\$, C\$\})$
6: $(\# \rightarrow \lambda, \{\$, \dagger, \#\sigma, \sigma\#\})$
7: $(\sigma \rightarrow \lambda \;, \{A, B, C, \#, S\})$

GF(2, 5, 9, 7)=RE

Recalling simulating rules of GF(2, 6, 8, 6)=RE

1: $(B \rightarrow \$ \;\;, \{S, AC, BB, \$, \#\})$
2: $(A \rightarrow \#\$, \{S, AC, BB, \#\})$
3: $(C \rightarrow \$\#, \{S, AC, BB, \$\#, C\#, \#C\})$
4: $(\$ \rightarrow \lambda \;, \{A\$, \$C, \$\sigma, \sigma\$, B\$, \$B\})$
5: $(\# \rightarrow \lambda, \{\$, \#A, C\#\})$
6: $(\sigma \rightarrow \lambda \;, \{A, B, C, \#, \$, S\})$

Simulating rules of GF(2, 5, 9, 7)=RE

1: $(B \rightarrow \dagger \;\;, \{S, AC, BB, \dagger, \#\})$
2: $(A \rightarrow \#\$, \{S, AC, BB, \#\})$
3: $(C \rightarrow \$\#, \{S, AC, BB, \$\#, \#C\})$
4: $(\dagger \rightarrow \lambda \;\;, \{A\dagger, \dagger C, C\dagger, \dagger\sigma, \sigma\dagger\})$
5: $(\$ \rightarrow \lambda \;, \{\dagger, \$A, \$B, B\$, C\$\})$
6: $(\# \rightarrow \lambda, \{\$, \dagger, \#\sigma, \sigma\#\})$
7: $(\sigma \rightarrow \lambda \;, \{A, B, C, \#, S\})$

Simulating $ABC \rightarrow \lambda$

$\sigma u ABC v\sigma t\sigma \Rightarrow_{1,2,3} \sigma u \#\$ \dagger \$\# v\sigma t\sigma \Rightarrow_4 \sigma u \#\$\$\# v\sigma t\sigma \Rightarrow_5^2$
$\sigma u \#\# v\sigma t\sigma \Rightarrow^2 \sigma u v\sigma t\sigma$

GF(2, 5, 9, 7)=RE

A dooming set of rules of GF(2, 5, 9, 7)=RE

1: $(B \to \$ \,,\ \{AC, BB, \$, \#, \dagger\})$
2: $(A \to \#,\ \{S, AC, BB, \#, A\dagger\})$
3: $(C \to \dagger \,,\ \{S, AC, BB, \dagger, \sigma C\})$
4: $(\$ \to \lambda \,,\ \{S, A\$, \$C, \sigma\$, \$\sigma\})$
5: $(\# \to \lambda,\ \{\$, \#\sigma, \#B, \#A\})$
6: $(\dagger \to \lambda \,,\ \{\$, \#, B\dagger, C\dagger\})$
7: $(\sigma \to \lambda \,,\ \{A, C, \dagger, \#, S\})$

(Not) simulating $ABC \to \lambda$ but simulating BC $\to B$

$\sigma u ABC v\sigma t\sigma \Rightarrow_{1,2,3} \sigma u \#\$\dagger v\sigma t\sigma \Rightarrow_4 \sigma u \# \dagger v\sigma t\sigma \Rightarrow_5 \sigma u \dagger v\sigma t\sigma =$
$\sigma u' B \dagger v\sigma t\sigma$ (if u = u'B) $\not\Rightarrow_6$ since $B\dagger$ is forbidden.
If $B\dagger$ is not there in rule 6, then
$\sigma BC\sigma t\sigma \Rightarrow_3 \sigma B \dagger \sigma t\sigma \Rightarrow_6 \sigma B\sigma t\sigma$.

Masopust, Meduna Normal Form

Let G be a $(5,2) - GNF$.
Let $h : \{A, B, C, D\}^* \to \{0,1\}^*$ be a homomorphism defined by
$h(A) = 00$, $h(B) = 00$,
$h(C) = 01$, and $h(D) = 10$,
then the unconditional rules are

- $S \to h(u)Sa$, if $S \to uSa$,
- $S \to h(u)Sh(v)$, if $S \to uSv$,
- $S \to h(u)\$h(v)$, if $S \to uv$,

and the non-context-free rules are of the form

- $0\$0 \to \$$, $1\$1 \to \$$ and
- the context-free rule $\$ \to \lambda$.

Future Research

- Mathematical: Investigate further possibilities to shrink the resources
- We have given the upper bound and finding lower bound is open. Would it be $GF(2, 2, *, *) \neq RE$? Recall, $GF(2) = RE$
- Impose the regulation of forbidding sets on
  1. P systems (membrane Computing)
  2. Lindenmayer systems
  3. Insertion-deletion system

  and study the computational completeness of these systems.
- If not RE, then can we at least simulate regular grammars, or context-free grammars? especially, with $d = 1$?

THANK YOU (Děkuji)

Questions are welcome.