Computational Completeness Resulting from Scattered Context Grammars Working Under Various Derivation Modes

Alexander Meduna & Ondřej Soukup
Brno University of Technology, Faculty of Information Technology
Božetěchova 2, 612 00 Brno, CZ
www.fit.vutbr.cz/~isoukup

MEMICS 2014
Recent results


• Introduction
• Motivation
  • Praxis × Theory
• Derivation modes for SCGs
  • Definition
  • Examples
  • Turing-completeness
• Conclusion
Modern architecture
Modern architecture

- Parallel and often distributed
### Modern architecture

- Parallel and often distributed
- Complex instructions
  - Access multiple memory cells simultaneously
  - Consistent with restrictions preventing collisions
Modern architecture

- Parallel and often distributed
- Complex instructions
  - Access multiple memory cells simultaneously
  - Consistent with restrictions preventing collisions
How to model various modern parallel architectures?
How to model various modern parallel architectures?

Ideal model
How to model various modern parallel architectures?

**Ideal model**

- **Strong enough**
  - Turing-complete
  - Suitable for modelling of parallelism
How to model various modern parallel architectures?

Ideal model

- **Strong enough**
  - Turing-complete
  - Suitable for modelling of parallelism
- **Simple**
How to model various modern parallel architectures?

**Ideal model**

- **Strong enough**
  - Turing-complete
  - Suitable for modelling of parallelism
- **Simple**
- **Flexible** to model various approaches to memory access
How to model various modern parallel architectures?

**Ideal model**

- **Strong enough**
  - Turing-complete
  - Suitable for modelling of parallelism
- **Simple**
- **Flexible** to model various approaches to memory access

Ideals do not exist ...
How to model various modern parallel architectures?

Ideal model

- **Strong enough**
  - Turing-complete
  - Suitable for modelling of parallelism
- **Simple**
- **Flexible** to model various approaches to memory access

Ideals do not exist ... however ...
Scattered Context Grammar

G = (V, T, P, S), N = V − T, S ∈ N | P ⊆ ∞ \bigcup_{m=1}^{\infty} (N_1 \times N_2 \times \cdots \times N_m \times V^*_1 \times V^*_2 \times \cdots \times V^*_m)

V_i = V, N_i = N, 1 ≤ i ≤ m

(A_1, A_2, ..., A_n) \rightarrow (x_1, x_2, ..., x_n)

Example

G = ({S, A, B, C, a, b, c}, {a, b, c}, P, S)

P = {1: (S) → (ABC), 2: (A, B, C) → (aA, bB, cC), 3: (A, B, C) → (a, b, c)}


L(G) = {x | x = a^n b^n c^n, n ≥ 1}

Computational Completeness Resulting from Scattered Context Grammars Working Under Various Derivation Modes
Scattered Context Grammar

- \( G = (V, T, P, S), N = V - T, S \in N \)
Scattered Context Grammar

- $G = (V, T, P, S), N = V - T, S \in N$
- $P \subseteq \bigcup_{m=1}^{\infty} \left( N_1 \times N_2 \times \cdots \times N_m \times V_1^* \times V_2^* \times \cdots \times V_m^* \right)$
  - $V_i = V, N_i = N, 1 \leq i \leq m$
- $(A_1, A_2, \ldots, A_n) \rightarrow (x_1, x_2, \ldots, x_n)$
Idea

Scattered Context Grammar

- \( G = (V, T, P, S), N = V \setminus T, S \in N \)
- \( P \subseteq \bigcup_{m=1}^{\infty} \left( N_1 \times N_2 \times \cdots \times N_m \times V_1^* \times V_2^* \times \cdots \times V_m^* \right) \)
  - \( V_i = V, N_i = N, 1 \leq i \leq m \)
  - \((A_1, A_2, \ldots, A_n) \rightarrow (x_1, x_2, \ldots, x_n)\)

Example

- \( G = (\{S, A, B, C, a, b, c\}, \{a, b, c\}, P, S) \)
  - \( P = \{ 1: (S) \rightarrow (ABC), 2: (A, B, C) \rightarrow (aA, bB, cC), 3: (A, B, C) \rightarrow (a, b, c) \} \)
- \( L(G) = \{ x \mid x = a^n b^n c^n, n \geq 1 \} \)
Idea

Scattered Context Grammars

Computational Completeness Resulting from Scattered Context Grammars Working Under Various Derivation Modes
Scattered Context Grammars

- Strong enough
Scattered Context Grammars

- Strong enough
- Simple
### Idea

**Scattered Context Grammars**

- Strong enough
- Simple
- Flexible

---

**Computational Completeness Resulting from Scattered Context Grammars Working Under Various Derivation Modes**
Idea

Scattered Context Grammars

- Strong enough
- Simple
- Flexible

Computational Completeness Resulting from Scattered Context Grammars Working Under Various Derivation Modes
Idea

Scattered Context Grammars

- Strong enough
- Simple
- Flexible

What about to break the usual concept of rewriting?
Idea

Scattered Context Grammars

- Strong enough
- Simple
- Flexible

What about to break the usual concept of rewriting?
Derivation modes
Derivation modes

- Do not change the definition of the model
Derivation modes

- Do not change the definition of the model
- Change the way the model works
Derivation modes

- Do not change the definition of the model
- Change the way the model works
- Flexibility
Derivation modes

- Do not change the definition of the model
- Change the way the model works
- Flexibility

Scattered Context Grammars + Derivation modes

- Strong enough
- Simple
- Flexible
(A, B, C) → (X, Y, Z)

aaaAbbbBcccCddd
Mode 1

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaAbbbBcccCddd\)

\[
aaaAbbbBcccCddd \Rightarrow aaaXbbbYcccZddd
\]
Mode 2

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaAbbbBcccCddd\)
Mode 2

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaAbbbBcccCddd\)

\[
aaaAbbbBcccCddd \Rightarrow aaaXbbbYcccZddd
\]
Mode 2

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \textit{aaa}A\textit{bb}bB\textit{ccc}C\textit{ddd}

\[
\text{aaaA}b\textit{bb}bB\textit{ccc}C\textit{ddd} \Rightarrow \text{aaaX}b\textit{bb}bY\textit{ccc}Z\textit{ddd}
\]
\[
\text{aaaA}b\textit{bb}bB\textit{ccc}C\textit{ddd} \Rightarrow \text{aa}abbXbYcZ\textit{cc}ddd
\]
Mode 2

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaA bbbB cccC ddd\)

\[
\begin{align*}
aaaA bbbB cccC ddd & \Rightarrow aaaX bbbY cccZ ddd \\
naaaA bbbB cccC ddd & \Rightarrow aabbbX bY cZ cccddd \\
naaaA bbbB cccC ddd & \Rightarrow aabbbcccXYZ ddd
\end{align*}
\]
Mode 3

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaAbbBcccCddd\)
Mode 3

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaAbbbBcccCddd\)

\[aaaAbbbBcccCddd \Rightarrow aaaXbbbYcccZddd\]
Mode 3

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaA bbbB cccC ddd\)

\[
\begin{align*}
aaaA bbbB cccC ddd & \Rightarrow aaaX bbbY cccZ ddd \\
aaaA bbbB cccC ddd & \Rightarrow X aaabbbY cccddddZ
\end{align*}
\]
Mode 3

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaAbbbBcccCddd\)

\[
\begin{align*}
aaaAbbbBcccCddd & \Rightarrow aaaXbbbYcccZddd \\
aaaAbbbBcccCddd & \Rightarrow XaaabbbYcccddZ \\
naaAbbbBcccCddd & \Rightarrow aXYaabbbcccZddd
\end{align*}
\]
Defined derivation modes has no influence on the generative power of scattered context grammars.

SCGs are still Turing-complete.

Mode 1 - trivial to prove ...

Mode 2 - ...
• Defined derivation modes has no influence on the generative power of scattered context grammars
  • SCGs are still Turing-complete
• Defined derivation modes has no influence on the generative power of scattered context grammars
  • SCGs are still Turing-complete
  • Mode 1
• Defined derivation modes has no influence on the generative power of scattered context grammars
  • SCGs are still Turing-complete
  • Mode 1 - trivial to prove ...
• Defined derivation modes has no influence on the generative power of scattered context grammars
  • SCGs are still Turing-complete
  • Mode 1 - trivial to prove ...
  • Mode 2 | 3 ...
Consider any $L \in \text{RE}$, $L \subseteq \Sigma^*$ then $L(G) = L_1$ and $L(G') = \text{reversal}(L_2)$.
• Consider any $L \in RE, L \subseteq \Sigma^*$
Consider any \( L \in RE, L \subseteq \Sigma^* \)

\[ L = h(L_1 \cap L_2), \text{ where } L_1, L_2 \in CF, h : T^* \rightarrow \Sigma^* \]
Consider any $L \in RE, L \subseteq \Sigma^*$

$L = h(L_1 \cap L_2)$, where $L_1, L_2 \in CF, h : T^* \rightarrow \Sigma^*$

then $G_i = (N_i, T, P_i, S_i), i = 1, 2$

- $L(G_1) = L_1$ and $L(G_2) = \text{reversal}(L_2)$
Mode 3 - generative power

- $(A, B, C) \rightarrow (X, Y, Z)$
- $aaaAabbbBcccCddd$
Mode 3 - generative power

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaAabbBcccCddd\)
- \((A) \rightarrow (X)\) ?
Mode 3 - generative power

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaA bbbB cccC ddd\)

- \((A) \rightarrow (X)\)?
- Context-free rules are not influenced
Mode 3 - generative power

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaAbbbBcccCddd\)

- \((A) \rightarrow (X) ?\)
- Context-free rules are not influenced

- We can simulate \(G_1\) and \(G_2\) by context free rules
• $(A, B, C) \rightarrow (X, Y, Z)$
• $aaaAAbbbbBcccCddd$

• $(A) \rightarrow (X)$?
• Context-free rules are not influenced

• We can simulate $G_1$ and $G_2$ by context free rules
  • $S \Rightarrow^* w_1 $$$w_2$
  • $\ldots 1001 $$$1001 \ldots$
• \((A, B, C) \rightarrow (X, Y, Z)\)
• \(aaaA bbbB cccC ddd\)

• \((A) \rightarrow (X) \?\)
• Context-free rules are not influenced

• We can simulate \(G_1\) and \(G_2\) by context free rules
  • \(S \Rightarrow^* w_1$$w_2\)
  • \(\ldots 1001$$1001\ldots\)

• \((0, $$, $$, 0) \rightarrow ($$, \varepsilon, \varepsilon, $$)\) and \((1, $$, $$, 1) \rightarrow ($$, \varepsilon, \varepsilon, $$)\)
Mode 3 - generative power

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaAbbbBcccCddd\)

- \((A) \rightarrow (X) ?\)
- Context-free rules are not influenced

- We can simulate \(G_1\) and \(G_2\) by context free rules
  - \(S \Rightarrow^* w_1$$w_2\)
  - \(\ldots1001$$1001\ldots\)

- \((0, $, $, 0) \rightarrow ($, \varepsilon, \varepsilon, $)\) and \((1, $, $, 1) \rightarrow ($, \varepsilon, \varepsilon, $)
  - \(\ldots1001$$1001\ldots\)
Mode 3 - generative power

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaAbbbBcccCddd\)

- \((A) \rightarrow (X)\) ?
- Context-free rules are not influenced

- We can simulate \(G_1\) and \(G_2\) by context free rules
  - \(S \Rightarrow^* w_1$$w_2\)
  - \(\ldots 1001$$1001\ldots\)

- \((0, $$, $$, 0) \rightarrow ($$, $$, $$, $$)\) and \((1, $$, $$, 1) \rightarrow ($$, $$, $$, $$)\)
  - \(\ldots 1001$$1001\ldots \Rightarrow \ldots 100$$001\ldots\)
Mode 3 - generative power

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaAbbbBcccCddd\)

- \((A) \rightarrow (X)\) ?
- Context-free rules are not influenced

- We can simulate \(G_1\) and \(G_2\) by context free rules
  - \(S \Rightarrow^* w_1$$w_2\)
  - \(\ldots 1001$$1001\ldots\)

- \((0, $, $, 0) \rightarrow ($, $, $, $)\) and \((1, $, $, 1) \rightarrow ($, $, $, $)\)
  - \(\ldots 1001$$1001\ldots \Rightarrow \ldots 100$$001\ldots :)\)
Mode 3 - generative power

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaA bbbB cccC ddd\)
- \((A) \rightarrow (X) \) ?
- Context-free rules are not influenced

- We can simulate \(G_1\) and \(G_2\) by context free rules
  - \(S \Rightarrow^{*} w_1$$w_2\)
  - \(\ldots 1001$$1001\ldots\)

- \((0, $$, $$, 0) \rightarrow ($$, $$, $$, $$)\) and \((1, $$, $$, 1) \rightarrow ($$, $$, $$, $$)\)
  - \(\ldots 1001$$1001\ldots \Rightarrow \ldots 100$$001\ldots :\) or \(\ldots 100$$0$$01\ldots\)
Mode 3 - generative power

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaAbbbBcccCddd\)

- \((A) \rightarrow (X)\) ?
- Context-free rules are not influenced

- We can simulate \(G_1\) and \(G_2\) by context free rules
  - \(S \Rightarrow^* w_1$$w_2\)
  - \(\ldots 1001$$1001\ldots\)

- \((0, $$, $$, 0) \rightarrow ($$, ε, ε, $$)\) and \((1, $$, $$, 1) \rightarrow ($$, ε, ε, $$)\)
  - \(\ldots 1001$$1001\ldots \Rightarrow \ldots 100$$001\ldots :)\) or \(\ldots 100$$01\ldots :)\)
Mode 3 - generative power

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaAbbbBcccCddd\)

- \((A) \rightarrow (X) ?\)
- Context-free rules are not influenced

- We can simulate \(G_1\) and \(G_2\) by context free rules
  - \(S \Rightarrow \ast w_1$$w_2\)
  - \(\ldots 1001$$1001\ldots\)

- \((0, $, $, 0) \rightarrow ($, \varepsilon, \varepsilon, $)\) and \((1, $, $, 1) \rightarrow ($, \varepsilon, \varepsilon, $)\)
  - \(\ldots 1001$$1001\ldots \Rightarrow \ldots 100$$$001\ldots :)\) or \(\ldots 100$$0$$01\ldots :)\)
  - \(\ldots 1001$$1001\ldots\)
Mode 3 - generative power

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaA bbbB cccC ddd\)
- \((A) \rightarrow (X) ?\)
- Context-free rules are not influenced

- We can simulate \(G_1\) and \(G_2\) by context free rules
  - \(S \Rightarrow^* w_1\$$w_2\)
  - \(...1001\$$1001...\)

- \((0, $, $, 0) \rightarrow ($, \(\varepsilon, \varepsilon, $\)) and \((1, $, $, 1) \rightarrow ($, \(\varepsilon, \varepsilon, $\))
  - \(...1001\$$1001... \Rightarrow ...100\$$001... :) or ...100$0$01... :(
  - \(...1001\$$1001... \Rightarrow ...10\$$11$01...\)
Mode 3 - generative power

- \((A, B, C) \rightarrow (X, Y, Z)\)
- \(aaaAabbbBccccCddd\)

- \((A) \rightarrow (X)\)
- Context-free rules are not influenced

- We can simulate \(G_1\) and \(G_2\) by context free rules
  - \(S \Rightarrow^* w_1$$w_2\)
  - \(\ldots 1001$$1001\ldots\)

- \((0, $$, $$, 0) \rightarrow ($$, $$, $$, $$)\) and \((1, $$, $$, 1) \rightarrow ($$, $$, $$, $$)\)
  - \(\ldots 1001$$1001\ldots \Rightarrow \ldots 100$$001\ldots :)\) or \(\ldots 100$$0$$01\ldots :)\)
  - \(\ldots 1001$$1001\ldots \Rightarrow \ldots 10$$11$$01\ldots :)\)
Conclusion

Scattered Context Grammars + Derivation modes

- Strong enough
- Simple
- Flexible
Conclusion

Scattered Context Grammars + Derivation modes

- Strong enough
- Simple
- Flexible
- Applicable to model parallel architectures

... and calling for the next research :)
Conclusion

Scattered Context Grammars + Derivation modes

- Strong enough
- Simple
- Flexible
- Applicable to model parallel architectures
- ... and calling for the next research :)

Computational Completeness Resulting from Scattered Context Grammars Working Under Various Derivation Modes


Thank you for your attention!