Pumping

Makoto Kanazawa
National Institute of Informatics
Pumping

$u_0 v_1^n u_1 v_2^n u_2 \in L(G)$ for all $n \geq 0$

The case of CFG.
The case of 2-MCFGs.
Is this the general picture?
Difficulty with Pumping

All but finitely many derivation trees contain a pump.

All sufficiently large derivation trees contain a part that can be iterated.
A derivation tree containing this pump yields a 4–pumpable string.
Difficultly with Pumping

Rather complex pattern.
The original string (n=1) cannot be pumped, but the string obtained by iterating the pump twice is 2-pumpable.
**Theorem** (Seki et al. 1991).
$L \in m$-MCFL $\Rightarrow$ $L$ is weakly $2m$-iterative.

**Myth.** $L \in m$-MCFL $\Rightarrow$ $L$ is $2m$-iterative.


L is weakly $k$–iterative if it contains an infinite subset that is $k$–iterative.
Many people erroneously believed that Seki et al. proved a stronger result.
Three Infinite Hierarchies

$\text{MCFL} = \bigcup_{m \geq 1} m\text{-MCFL}$

$\gamma\text{CFT}_{sp} = \bigcup_{m \geq 1} \gamma\text{CFT}_{sp} (m - 1)$

$\text{MCFL}_{wn} = \bigcup_{m \geq 1} m\text{-MCFL}_{wn}$

$\mathbf{C} = \bigcup_{k \geq 1} \mathbf{C}_k$

Pumping lemmas in the usual form hold for the two subhierarchies of the MCFLs.
Pumping Lemma for PDA

\[ \neg (\text{All but finitely many accepting computations reach stack height } |Q|^2) \]

\[ \{ w \mid w \text{ has an accepting computation that doesn’t reach stack height } |Q|^2 \} \text{ is regular} \]

The proof in each case is somewhat similar to the proof of the pumping lemma for CFLs using PDA, rather than CFG.
Pumping Lemma for $C_k$

$$CT(L_1, L_2) = L_1 \cap (L_2)^*$$

- No long spine $\Rightarrow$ element of a regular set

The set of trees without long spines are the Kleene star of a finite set.
Pumping Lemma for $C_k$

$$CT(L_1, L_2) = L_1 \cap (L_2)^{*,c}$$

When some spine is long enough to be pumpable, ...
Pumping Lemma for $C_k$

$$CT(L_1, L_2) = L_1 \cap (L_2)^{*,c}$$

When a tree has a spine that is $m$-pumpable, the yield of the tree is $2m$-pumpable.
Pumping Lemma for $\mathbf{C}_k$

$$\text{CT}(L_1, L_2) = L_1 \cap (L_2)^*, c$$

$L_1 \in \text{LOC}$ and $L_2$ is $m$-iterative

$\Rightarrow y\text{CT}(L_1, L_2)$ is $2m$-iterative

**Theorem** (Palis and Shende 1995).

$L \in \mathbf{C}_k \Rightarrow L$ is $2^k$-iterative.
Pumping Lemma for $m$-MCFL$_{wn}$

**Theorem** (Kanazawa 2009).

$L \in m$-MCFL$_{wn} \Rightarrow L$ is $2m$-iterative.

The proof of the Pumping Lemma for $m$-MCFL$_{wn}$ is more complex.
• If $G$ is a well-nested $m$-MCFG,

\[
\{ T \mid T \text{ is a derivation tree of } G \text{ without even } m\text{-pumps} \}
\]

may not be finite.

• But there is a well-nested $(m-1)$-MCFG generating

\[
\{ \text{yield}(T) \mid T \text{ is a derivation tree of } G \text{ without even } m\text{-pumps} \}.
\]

If the derivation tree contains an even $m$-pump, the string is $2m$-pumpable. Otherwise, the string is in the language of some w.n. $(m-1)$-MCFG, and therefore is $2(m-1)$-pumpable (disregarding finitely many exceptions).

Proof by induction on $m$. 

\[
(B v_1 x_1 v_2, \ldots, v_{2m-1} x_m v_{2m})
\]

“even $m$-pump”
The proof of this claim is by successive transformations on the grammar.
A rule is m–proper if the head nonterminal is m–ary and there is an m–ary nonterminal on the right–hand side, each of whose arguments appear in the corresponding argument of the head nonterminal.

Unfold until there is no m–proper rule. This procedure terminates because the grammar does not allow an even m–pump.
\[ \Pi_1: \ S(\mathbf{x}_1 \mathbf{x}_2) \leftarrow B(\mathbf{x}_1, \mathbf{x}_2) \]
\[ \Pi_5: \ B(aa \mathbf{x}_1 b \mathbf{x}_2 cb, cdd) \leftarrow A(\mathbf{x}_1, \mathbf{x}_2) \quad \text{m-degree} = 1 \]
\[ \Pi_6: \ B(ab, cd) \leftarrow \]
\[ \Pi_3: \ A(\mathbf{a} \mathbf{x}_1 \mathbf{b} \mathbf{x}_2 c, d) \leftarrow A(\mathbf{x}_1, \mathbf{x}_2) \quad \text{m-degree} = 1 \]
\[ \Pi_4: \ A(\varepsilon, \varepsilon) \leftarrow \]

\[ \downarrow \quad \text{unfolding}^{-1} \]

\[ \Pi_1: \ S(\mathbf{x}_1 \mathbf{x}_2) \leftarrow B(\mathbf{x}_1, \mathbf{x}_2) \]
\[ \Pi_{5.1}: \ B(aa \mathbf{x} \mathbf{c}b, cdd) \leftarrow C(\mathbf{x}) \]
\[ \Pi_{5.2}: \ C(\mathbf{x}_1 \mathbf{b} \mathbf{x}_2) \leftarrow A(\mathbf{x}_1, \mathbf{x}_2) \]
\[ \Pi_2: \ B(ab, cd) \leftarrow \]
\[ \Pi_{3.1}: \ A(\mathbf{a} \mathbf{x} \mathbf{c}, d) \leftarrow D(\mathbf{x}) \]
\[ \Pi_{3.2}: \ D(\mathbf{x}_1 \mathbf{b} \mathbf{x}_2) \leftarrow A(\mathbf{x}_1, \mathbf{x}_2) \]
\[ \Pi_4: \ A(\varepsilon, \varepsilon) \leftarrow \]

\[ \Pi_5 = \Pi_{5.1} \circ \Pi_{5.2} \]
\[ \Pi_3 = \Pi_{3.1} \circ \Pi_{3.2} \]

The m-degree of a rule is 0 if the arity of the head nonterminal is < m; otherwise it’s the number of m-ary nonterminals on the right-hand side. Do the converse of unfolding.
\[ \pi_1: S(x_1x_2) \leftarrow B(x_1, x_2) \]

\[ \pi_{5.1}: B(aaxcb, cdd) \leftarrow C(x) \]

\[ \pi_{5.2}: C(x_1bx_2) \leftarrow A(x_1, x_2) \]

\[ \pi_2: B(ab, cd) \leftarrow \]

\[ \pi_{3.1}: A(axc, d) \leftarrow D(x) \]

\[ \pi_{3.2}: D(x_1bx_2) \leftarrow A(x_1, x_2) \]

\[ \pi_4: A(\varepsilon, \varepsilon) \leftarrow \]

\[ \downarrow \text{unfolding} \]

\[ \pi_1 \circ \pi_{5.1}: S(aaxcbcdd) \leftarrow C(x) \]

\[ \pi_1 \circ \pi_2: S(abcd) \leftarrow \]

\[ \pi_{5.2} \circ \pi_{3.1}: C(axcbd) \leftarrow D(x) \]

\[ \pi_{5.2} \circ \pi_4: C(b) \leftarrow \]

\[ \pi_{3.2} \circ \pi_{3.1}: D(axcbd) \leftarrow D(x) \]

\[ \pi_{3.2} \circ \pi_4: D(b) \leftarrow \]

Now each rule contains m-ary nonterminals only on one side of the rule, if any. Unfolding eliminates all m-ary nonterminals.
Program Transformation

$m$-MCFG\textsubscript{wn} with no even $m$-pumps

unfolding

no $m$-proper rules

unfolding\textsuperscript{-1}

total $m$-degree = 0

unfolding

$(m-1)$-MCFG\textsubscript{wn}
Reduction of $m$-degrees

\[ B(x_1 y_1 z_1, z_2 y_2 a y_3 b, c x_2 d) \leftarrow A(x_1, x_2), B(y_1, y_2, y_3), C(z_1, z_2) \]

\[ \downarrow \text{unfolding}^{-1} \]

\[ B(x_1 w_1, w_2 b, c x_2 d) \leftarrow A(x_1, x_2), D(w_1, w_2) \]
\[ D(y_1 z_1, z_2 y_2 a y_3) \leftarrow B(y_1, y_2, y_3), C(z_1, z_2) \]

The well-nestedness assumption is necessary in the second step. Here’s a case of a well-nested rule.
Reduction of $m$-degrees

If a rule is non-well-nested, the procedure does not work.
The proof shows that a 2-MCFL is 4-iterative.
$H(x_2) \leftarrow G(x_1, x_2, x_3)$
$G(ax_1, y_1cx_2cdy_2dx_3, y_3b) \leftarrow G(x_1, x_2, x_3), G(y_1, y_2, y_3)$
$G(a, \varepsilon, b) \leftarrow$

$$a^{n+1}c \ldots b^m \overline{c}d a^n \ldots \overline{d} b^{m+1}$$

$\nu_0 = \varepsilon$
$\nu_{n+1} = a^{n+1}c \nu_n \overline{c}d \nu_n \overline{d} b^{n+1}$

Pumping fails for $m$–MCFLs for $(m > 2)$.
Here’s an example of a 3–MCFL that is not $k$–iterative for any $k$. 
Since every language in C is k–iterative for some k, this language separates MCFL from C.