Properties and Pumping Lemma of Context Free Languages

Closure, decision problems, and limits of context-free languages.

Constructive proofs • Pumping arguments • Exam practice

Learning outcomes

Closure

Identify operations that preserve context-free languages and construct grammars.

Decision properties

Distinguish decidable membership, emptiness and finiteness from undecidable problems.

Non-context-free languages

Apply the pumping lemma with correct quantifiers and exhaustive case analysis.

Closure: the big picture

Closed under

Union, concatenation, Kleene star, reversal, homomorphism, inverse homomorphism and intersection with regular languages.

Not closed under

General intersection, complement and difference of two context-free languages.

Meaning

Closure means every pair of languages in the class yields another language in the class under the specified operation.

Union, concatenation and star

Start with CFGs G_1 and G_2

Rename variables so the grammars have disjoint variable sets. Let their start symbols be S_1 and S_2.

Add a fresh start symbol S

Union: $S \to S_1 \mid S_2$ Concatenation: $S \to S_1 S_2$

Kleene star of L_1

$S \to S_1 S \mid \varepsilon$. Keep the original productions of G_1. The empty string is always included.

Reversal and homomorphisms

Reversal

Reverse every production right-hand side. Induction on parse trees proves that the new grammar generates L^R.

Homomorphism

Replace each terminal a in every rule by h(a), allowing h(a)=ε. The grammar generates h(L).

Inverse homomorphism

An NPDA can simulate reading h(a) whenever it consumes a; finite control stores the pending suffix of h(a). Thus $h^{-1}(L)$ is context-free.

Intersection with a regular language

Construction

Run an NPDA for L and a DFA for R in parallel. The control states are pairs (p,q); the stack belongs to the NPDA.

Transition rule

On an input symbol a, update both controls; on an ε-move, update only the NPDA control.

Acceptance

Use final-state acceptance after consuming the entire input. Both component states must be accepting. Therefore L ∩ R is context-free.

Why general intersection fails

Two context-free languages

$L_1 = \{a^i b^i c^j : i,j \geq 0\}$; grammar $S \to AC$, $A \to aAb \mid \varepsilon$, $C \to cC \mid \varepsilon$.

Second language

$L_2 = \{a^i b^j c^j : i,j \geq 0\}$; grammar $S \to AB$, $A \to aA \mid \varepsilon$, $B \to bBc \mid \varepsilon$.

Intersection

$L_1 \cap L_2 = \{a^n b^n c^n : n \geq 0\}$, which is not context-free (proof follows).

Complement and difference

Complement

If CFLs were closed under complement, De Morgan’s law and union closure would imply intersection closure. Contradiction.

Difference

If CFLs were closed under difference, $\Sigma^* - L$ would always be a CFL. This would imply complement closure.

Useful exception

If L is context-free and R is regular, then $L - R = L \cap (\Sigma^* - R)$ is context-free.

Decision properties

Decidable

Membership: CYK after CNF conversion (handle ε separately). Emptiness: mark generating variables and test the start symbol.

Finiteness

Convert to CNF, remove useless variables; a cycle in the variable dependency graph implies infinitely many words. Otherwise the language is finite.

Undecidable in general

Equivalence, inclusion, universality and whether the intersection of two CFG languages is empty.

Pumping lemma: formal statement

If L is context-free

There exists $p \geq 1$ such that every $w \in L$ with $|w| \geq p$ has a decomposition $w = uvxyz$ satisfying:

Three conditions

$|vxy| \leq p$; $|vy| \geq 1$; $uv^ixy^iz \in L$ for every integer $i \geq 0$.

Interpretation

Two substrings, v and y, are pumped together. Either may be empty, but not both.

Why two substrings can be pumped

$S \Rightarrow^* u\ A\ z$

$A \Rightarrow^* v\ A\ y$

$A \Rightarrow^* x$

Repeat the middle derivation i times: $S \Rightarrow^* uv^ixy^iz$.

Proof idea using CNF

Choose a pumping length

For a CNF grammar with m variables, take $p = 2^m$. A parse tree for a word of length $\geq p$ has a path with at least m+1 variable occurrences.

Find a repeated variable

On a longest path, take the lowest m+1 variable occurrences. A variable A repeats; the upper subtree yields at most p terminals.

Pump the repeated section

$A \Rightarrow^* vAy$ and $A \Rightarrow^* x$. Binary branching ensures at least one of v,y is nonempty. Repeating or removing the section preserves derivability.

How to prove a language is not context-free

1. Assume L is context-free

Let p be the pumping length. Choose a witness $w \in L$, $|w| \geq p$, depending on p.

2. Consider every valid decomposition

Let $w = uvxyz$ with $|vxy| \leq p$ and $|vy| \geq 1$. Do not select v and y yourself.

3. Derive a contradiction

For each valid split, find $i \geq 0$ for which $uv^ixy^iz \notin L$. The assumption must be false.

Solved example 1: three equal blocks

Language and witness

$L = \{a^n b^n c^n : n \geq 0\}$. Assume L is context-free; choose $w = a^p b^p c^p$.

Locality of pumping

Since $|vxy| \leq p$, this region cannot contain both an a and a c. It touches at most two adjacent blocks.

Pump down: i = 0

At least one symbol count decreases ($|vy| \geq 1$), while at least one block count stays p. The three counts cannot remain equal. Contradiction.

Solved example 2: equal counts in any order

Language

$L = \{w \in \{a,b,c\}^* : \#a(w) = \#b(w) = \#c(w)\}$.

Direct pumping proof

Choose $w = a^p b^p c^p$. The same window argument shows pumping down changes at least one count but leaves another unchanged.

Alternative closure proof

Intersect L with the regular language $a^*b^*c^*$. The result is $\{a^n b^n c^n\}$, not context-free. Therefore L is not context-free.

Solved example 3: unary powers of two

Language and witness

L = {a^(2^n) : n ≥ 0}. Choose $N = 2^k > p$ and $w = a^N$.

Pump up: i = 2

Let d = |v| + |y|. Then $1 \le d \le p < N$. The pumped word has length N+d.

Contradiction

$N < N+d < 2N$. No power of two lies strictly between consecutive powers N and 2N. Thus the pumped word is not in L.

Common errors and classroom exercises

Avoid these errors

The lemma gives a necessary condition, not a sufficient condition. Do not choose the split, and do not pump v without y.

Exercise 1

Show $\{a^n b^n c^{2n} : n \geq 0\}$ is not context-free.

Exercise 2

Is the intersection of a CFL with a regular language always a CFL? Is the complement of a CFL always a CFL? Justify.

Solutions and recap

Exercise 1

Choose $a^p b^p c^{2p}$. Pump down: at least one block decreases and one is unchanged. The equalities #a=#b and #c=2#a cannot both hold.

Exercise 2

Yes: product of NPDA and DFA. No: complement plus union would imply general intersection closure, which fails.

Takeaway

Use closure to build or reduce languages; use pumping to disprove context-freeness. Next: Turing Machines and Their Variants.