Lecture 25

Thursday, March 21, 2024 11:34 AM

Parsing and the CKY algorithm

Given a CFG G in CNF form and a string x, determine if $x \in L(G)$.

Naive approach: because G is CNF, we only need to try all derivations of length at most $2|x|-1$ and see if x is generated.

check $\exists\ K \leq 2|x|-1$ s.t. $S \xrightarrow[G]{K} x$.

However, we would need to try exponentially many case in terms of $|x|$.

$S \to AB \mid BA \mid AC \mid BD \mid SS$

$A \to a$

$B \to b$

$C \to SB$

$D \to SA$

$L(G) = \{ x \in \{a,b\}^+ : \#a(x) = \#b(x) \geq 1 \}$

$D \rightarrow SA$

Heuristic: don't check braches that have already generated a "bad" string.

But this doesn't always work...

---

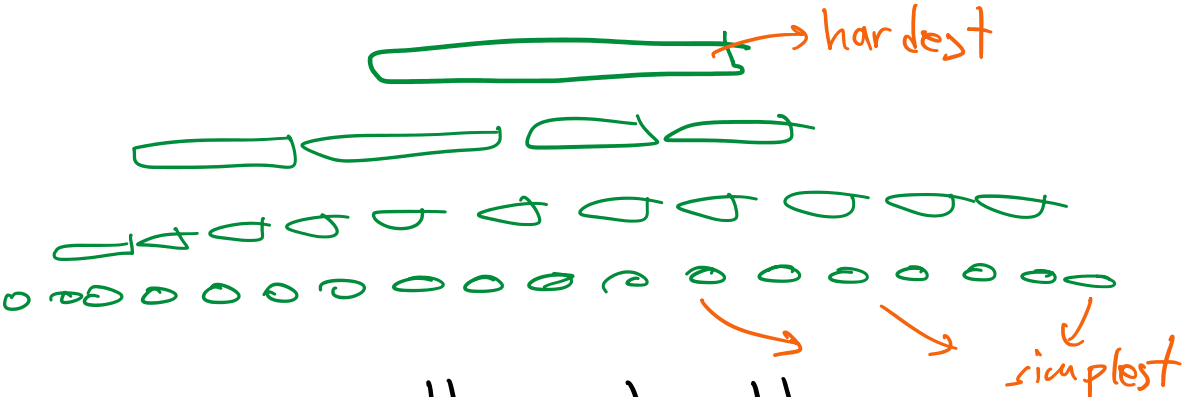
idea: use "dynamic programming".

Divide the problem into smaller problems.

Solve the easier/smaller problem first.

And use those answers to solve increasingly harder problems.

Avoid solving the same sub-problem multiple times (instead store the solutions and reuse them)



Defining the subproblem is the most important step!

---

notation: $x - abbaa$

Notation: $X = abbaa$

$X_{0:1} = a$ $\quad$ $X_{1:3} = bb$

Sub-problem: which non-terminals can generate $X_{i:j}$ (for each $j > i$)?

Call that set $T_{i:j} \subseteq N$.

Given $T_{i:j}$ for all $j > i$, how to tell if $X \in L(G)$?

$X \in L(G) \iff S \in T_{0:|X|}$

$X = aab$

$T_{0:1} = \{A\}$

$T_{1:2} = \{A\}$

$T_{2:3} = \{B\}$

$T_{0:3} = \emptyset \Rightarrow X \notin L(G)$

$T_{1:3} = \{S\}$

$T_{i:j}$

First solve $T_{i:i+1}$ $\quad \forall i \in \{0, \dots, |X|-1\}$

Then solve $T_{i:i+2}$ $\quad \forall i \in \{0, \dots, |X|-2\}$

$\vdots$ $\quad$ $\vdots$

solve $T_{0:|X|}$