UNIVERSITY OF LIVERPOOL

Department of Computer Science
Sebastian Wild

Date: 2021-03-03
Version: 2021-04-06 14:42

Tutorial 4 for COMP 526 – Applied Algorithmics, Spring 2021

Problem 1 (Parallel And)

We consider the problem of computing the logical *and* of an array $B[0..n-1]$ of n Boolean values (n bits), i.e., the result should be *true* if and only if all n entries are true. (We assume here that each bit is stored as a full word.)

a) Design a CREW-PRAM parallel algorithm for computing the "logical and" of $B[0..n-1]$. Your algorithm should have $\mathcal{O}(\log n)$ time (span) and $\mathcal{O}(n \log n)$ work.

b) Can you make the algorithm work-efficient?

c) Now consider a CR**C**W-PRAM; you can choose a write-conflict resolution rule that is convenient for your purposes. Design a *constant-time* parallel algorithm for computing the logical and.

Problem 2 (Fibonacci language and failure function)

The sequence of Fibonacci words $(w_i)_{i\in\mathbb{N}_0}$ is defined recursively:

$$\begin{aligned} w_0 &= \texttt{a} \\ w_1 &= \texttt{b} \\ w_n &= w_{n-1} \cdot w_{n-2} \qquad (n \ge 2) \end{aligned}$$

Unfolding the recursion yields $w_2 = \texttt{ba}$, $w_3 = \texttt{bab}$, $w_4 = \texttt{babba}$, an so on.

(Note that the lengths $|w_0|, |w_1|, |w_2|, \ldots$ are *Fibonacci numbers*, hence the name. More precisely, we have $|w_n| = F_{n+1}$, with the Fibonacci numbers defined as $F_0 = 0$, $F_1 = 1$, and $F_n = F_{n-1} + F_{n-2}$, for $n \ge 2$.)

a) Construct the transition function δ of the string-matching automaton for w_6 and draw the string-matching automaton.

b) Construct the prefix function F and the draw the KMP automaton with failure links for w_6.