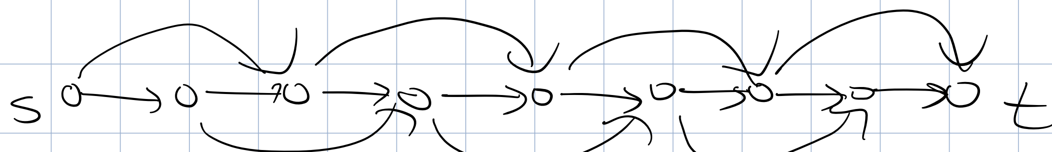
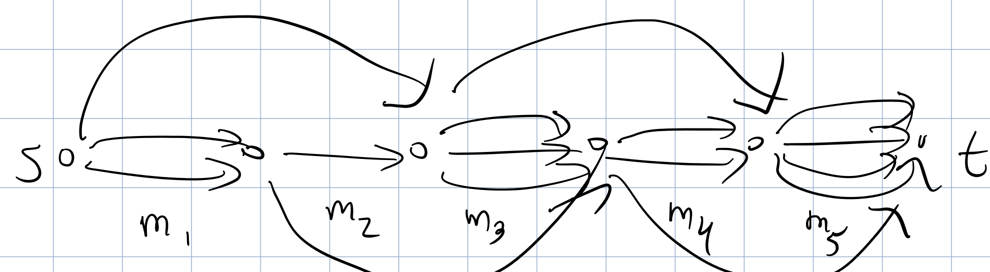
22 Sep 2021

Determinants and Perfect Matchings in NC^2

<u>Recall:</u> Multiplying $n \times n$ matrices with b-bit entries takes
$O(\log(nb))$ time
$O(\mathrm{poly}(n,b))$ work
on a parallel computer.



Q: If there are n nodes, how many s-t paths?
A: F_n, the nth Fibonacci number.



s-t paths is $f(m_1, m_2, \ldots, m_8)$.

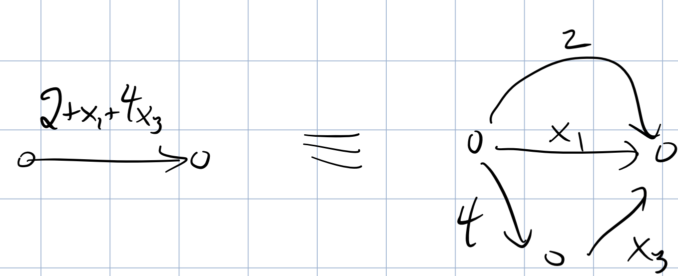
$f(m_1, \ldots, m_n)$ is a polynomial with F_{n+1} (Fibonacci #) monomials.

It can be evaluated in $O(n)$ arithmetic ops by dynamic programming.

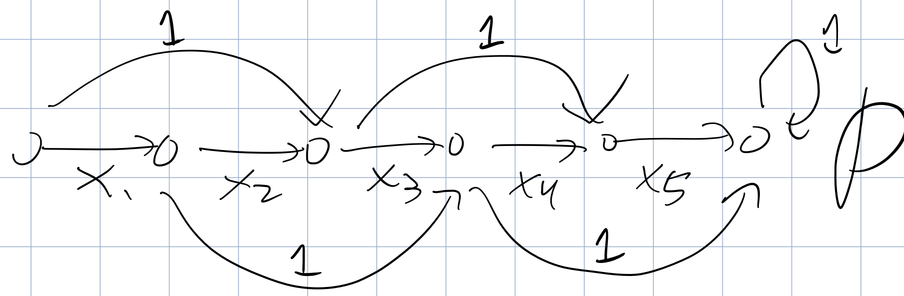
<u>Def.</u> An algebraic branching program (ABP) is a DAG with source s, sink t, and edges labeled with (degree-1 polynomial funcs of) formal variables $x_1, \ldots, x_m$.

The polynomial represented by branching program P is

$$f^P(x_1, \ldots, x_m) \triangleq \sum_{s\text{-}t \text{ path } P} \prod_{e \in E(P)} \mathrm{label}(e).$$

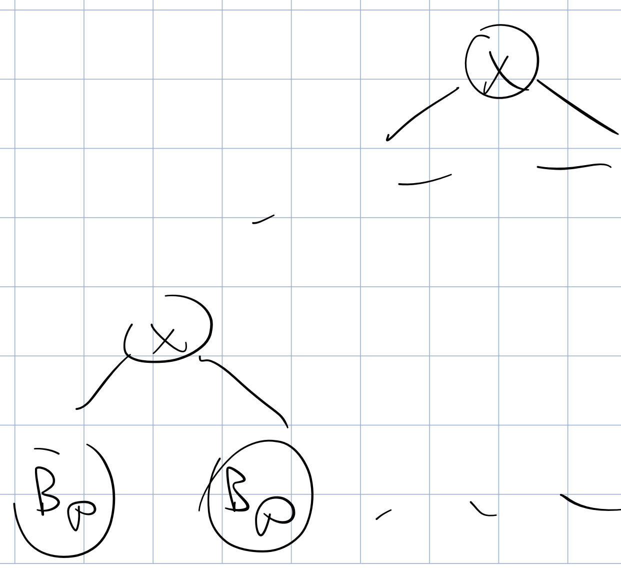


Evaluating ABP using matrix mult.

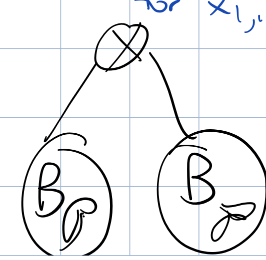


$$A_\rho = \begin{pmatrix} 0 & x_1 & 1 & 0 & 0 & 0 \\ & 0 & x_2 & 1 & 0 & 0 \\ & & 0 & x_3 & 1 & 0 \\ & 0 & & 0 & x_4 & 1 \\ & & & & 0 & x_5 \\ & & & & & 1 \end{pmatrix}$$

$$f^\rho(x_1, \ldots, x_m) = \left(A_\rho^n\right)_{1,n}.$$



$B_\rho = A_\rho$ with specified scalars substituting for $x_1, \ldots, x_m$



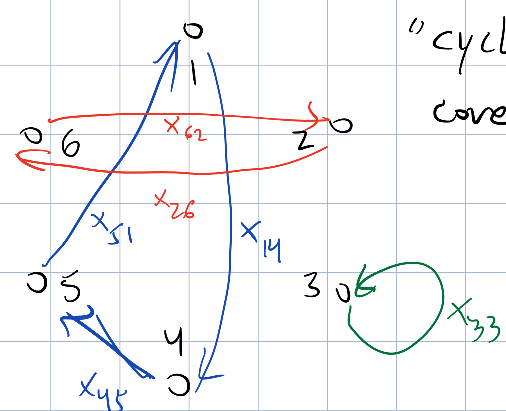
Each ⓧ in the diagram corresponds to an operation that takes $O(\log(nb))$ time in parallel.

So in total computing B_ρ^n takes $O(\log(n)\cdot\log(nb))$.

Reducing determinant to ABP.

Thinking about permutations using cycle diagrams.

$1 \mapsto 4$
$2 \mapsto 6$
$3 \mapsto 3$
$4 \mapsto 5$
$5 \mapsto 1$
$6 \mapsto 2$



"cycle cover"

$$\{\text{permutations of } [n]\} \longleftrightarrow \{\text{cycle covers of } K_n\}$$

monomial corresp. to σ in the determ. poly. $\longmapsto (-1)^{n-(\#\text{ cycles})} \cdot (\text{product of edge labels})$