Unary finite automata

Below, H(n) represents the function $e^{\sqrt{nlog(n)}}$

These are some statements related to unary finite automata

- Each unary n-state 2dfa can be simulated by a 1dfa with O(H(n)) states.
- For each n there is a unary n-state 2dfa A such that each 1dfa recognizing L(A) requires Ω(H(n)) states.

Unary finite automata

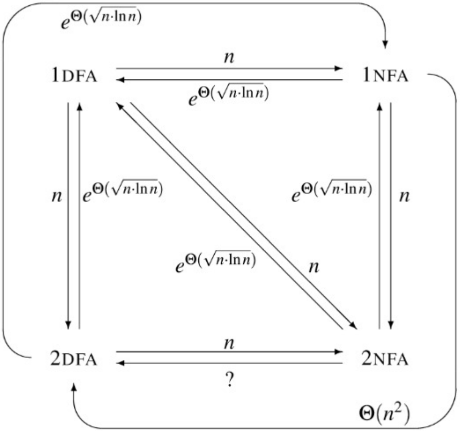
- For each n there is a unary n-state 2dfa A such that each 1nfa recognizing L(A) requires $\Omega(H(n))$
- Each unary n-state 1nfa A can be simulated by a 2dfa B with $O(n^2)$ states.
- For each n there is a unary n-state 1nfa A such that each 2dfa recognizing L(A) requires $\Omega(n^2)$ states.

Unary finite automata

Informally speaking we can see that 2dfa's are hard to simulate by ldfa's, even if we consider only unary languages.

Also, for unary languages,two-way motion is more powerful, in a sense, because we can simulate unary lnfa's by 2dfa's increasing the number of states only polynomially, which is not possible the other way round.

Unary finite automata



1nfa to 2dfa

Lemma A: If gcd(a,b)=1, then the greatest number such that the equation $ax + by = c$ has no solutions in natural numbers is (a-1)(b-1)-1.

Theorem:For each n there is a unary n-state 1nfa A such that each 2dfa recognizing L(A) requires $\Omega(n^2)$ states.

Proof: Let L=$\{x|x = nx_1$+(n-1)x_2 for x_1, x_2 $\in$N$\}$. L can be recognized by a Infa A with n states. A=(Q, q_0, E, F) is defined as follows: Q=q_0,..., q_{n-l}, E $= \{(q_i,\ q_{i+1})|i = 0, ..., n-1\} \cup (q_1, q_3)$ (the addition is mod n) and F = q_0. Let m = max($\mathbb{N}$-L). By Lemma A, m = O(n^2). Consider a 2dfa B recognizing L and its computation on m. Suppose that, in all passes on m, B enters a cycle and let y_1,...,y_k be the lengths of these cycles. Then B would reject also m'= m + lcm($y_1, .., y_k$), which contradicts the fact that m' $\in$ L. Therefore, there is a pass. of B on m without a cycle and the theorem follows.