Daily Problem: 26–Jan–2026

Problem Statement

Prove that the language recognized by any unary NFA (i.e., an NFA over the one-letter alphabet $\Sigma = \{a\}$) is a finite union of arithmetic progressions. In other words, show that for any unary NFA N the language $L(N)$ is ultimately periodic.

Proof

Let $N = (Q, \{a\}, \delta, I, F)$ be a unary NFA, where $|Q| = n$. Since the alphabet contains only the symbol a, every input string has the form a^k for some $k \geq 0$. Hence the language recognized by N is determined entirely by which values of k yield acceptance.

Observation: Unary NFAs Correspond to Directed Graphs

The transition structure of N can be viewed as a directed graph $G = (Q, E)$ where $(p, q) \in E$ iff $q \in \delta(p, a)$. Reading the symbol a corresponds to moving along a directed edge, so every computation on input a^k corresponds to a path of length k in G from some initial state to some accepting state.

Thus we have:

$$a^k \in L(N) \iff \exists \text{ a path of length } k \text{ from some } i \in I \text{ to some } f \in F.$$

Decomposition into Strongly Connected Components

Partition the graph G into its strongly connected components (SCCs). Every SCC is either:

- *trivial*: a single node with no self-loop, or
- *nontrivial*: contains a directed cycle.

If there is no path from any initial state to any accepting state passing through a cycle, then only finitely many lengths k admit valid paths, so $L(N)$ is finite, and hence a finite union of trivial arithmetic progressions.

Otherwise, consider any SCC C reachable from an initial state and co-reachable to an accepting state. Let C contain $m \geq 1$ states and let:

$$P = \{\ell \geq 1 : \exists \text{ a cycle in } C \text{ of length } \ell\}.$$

Since C is finite, P is finite and nonempty. Let $d = \gcd(P)$. Standard graph-theoretic results imply:

> For any sufficiently large integer k, there exists a path of length k from any node in C to any other node in C if and only if $k \equiv r \pmod{d}$ for some residue class r.

Applying to Accepted Lengths

Consider any accepting computation that enters some such SCC C. Let:

$$\alpha = \text{minimum length of a path from some } i \in I \text{ to some } c \in C,$$

$$\beta = \text{minimum length of a path from some } c \in C \text{ to some } f \in F.$$

Inside the component C, cycles allow us to pump the length by multiples of d. Hence for sufficiently large input lengths k, acceptance holds exactly when:

$$k = \alpha + t + \beta \quad \text{with} \quad t \in \{t_0, t_0 + d, t_0 + 2d, \dots\}$$

for some residue class $t_0 \bmod d$.

Therefore, the set:

$$L(N) = \{k \in \mathbb{N} : a^k \in L(N)\}$$

is ultimately a finite union of arithmetic progressions of modulus d, determined by the SCCs that lie on paths from initial to accepting states.

Finite Union of Progressions

Since there are finitely many SCCs and finitely many entrance/exit choices, the language splits into:

$$L(N) = F \cup \bigcup_{j=1}^{m} \{k \geq N_j : k \equiv r_j \pmod{d_j}\},$$

where F is a finite set, each d_j is a positive integer, and each $r_j \in \{0, 1, \ldots, d_j - 1\}$.

This shows that $L(N)$ is ultimately periodic.

Conclusion

We have shown that for any unary NFA N:

- acceptance depends solely on input length k,
- the set of accepting lengths is controlled by cycles in SCCs,
- beyond some bound, membership is periodic with period d,
- only finitely many residue classes arise.

Hence the language of a unary NFA is always a finite union of arithmetic progressions, i.e., it is ultimately periodic.