

بسم الله الرحمن الرحيم

دانشگاه صنعتی اصفهان – دانشکده مهندسی برق و کامپیوتر
(نیم سال تحصیلی ۴۰۲۱)

نظریه زبان‌ها و ماشین‌ها

حسین فلسفین

اهم مطالبی که در این اسلاید به آنها خواهیم پرداخت:

👉 تعریف رسمی مفهوم پذیرش برای یک *NFA*

👉 اثبات قضیه هم‌ارزی *NFA* ها و *DFA* ها

👉 بیان مثال‌ها

👉 بازگشت به مبحث خواص بستاری زبان‌های منظم و اثبات بسته بودن کلاس

زبان‌های منظم تحت عملگرهای \cup ، \circ ، و $*$ با بهره‌گیری از مفهوم عدم قطعیت

The formal definition of computation for an NFA: Let $N = (Q, \Sigma, \delta, q_0, F)$ be an NFA and w a string over the alphabet Σ . Then we say that N accepts w if we can write w as $w = y_1 y_2 \cdots y_m$, where each y_i is a member of Σ_ϵ and a sequence of states r_0, r_1, \dots, r_m exists in Q with three conditions:

1. $r_0 = q_0$,
2. $r_{i+1} \in \delta(r_i, y_{i+1})$, for $i = 0, 1, \dots, m - 1$, and
3. $r_m \in F$.

Condition 1 says that the machine starts out in the start state. **Condition 2** says that state r_{i+1} is one of the allowable next states when N is in state r_i and reading y_{i+1} . Observe that $\delta(r_i, y_{i+1})$ is the set of allowable next states and so we say that r_{i+1} is a member of that set. Finally, **condition 3** says that the machine accepts its input if the last state is an accept state.

The Extended Transition Function δ^* for an NFA, and the Definition of Acceptance

Let $N = (Q, \Sigma, \delta, q_0, F)$ be an NFA. We define the extended transition function $\delta^* : Q \times \Sigma^* \mapsto 2^Q$ as follows:

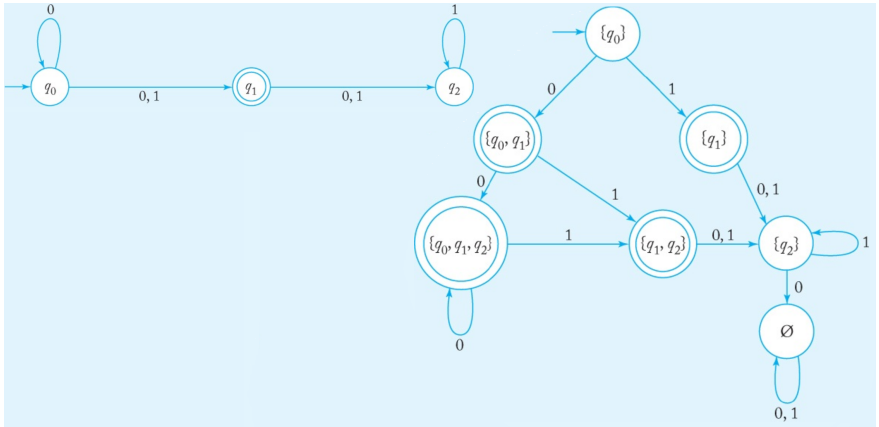
1. For every $q \in Q$, $\delta^*(q, \varepsilon) = E(\{q\})$.
2. For every $q \in Q$, every $y \in \Sigma^*$, and every $\sigma \in \Sigma$,

$$\delta^*(q, y\sigma) = E\left(\bigcup \{\delta(p, \sigma) \mid p \in \delta^*(q, y)\}\right).$$

A string $x \in \Sigma^*$ is accepted by N if $\delta^*(q_0, x) \cap F \neq \emptyset$. The language $L(N)$ accepted by N is the set of all strings accepted by N .

هم تعریف این اسلاید و هم تعریف اسلاید قبل، هر دو معتبر هستند.

An NFA and its equivalent DFA: Example 1



Equivalence of DFAs & NFAs

Every NFA has an equivalent DFA. This means that, for every language $A \subseteq \Sigma^*$ accepted by an NFA $N = (Q, \Sigma, \delta, q_0, F)$, there is a DFA $M = (Q', \Sigma, \delta', q'_0, F')$ that also accepts A .

Proof: Let $N = (Q, \Sigma, \delta, q_0, F)$ be the NFA recognizing some language A . We construct a DFA $M = (Q', \Sigma, \delta', q'_0, F')$ recognizing A .

✎ Before doing the full construction, let's first consider the easier case wherein N has no ε arrows. Later we take the ε arrows into account.

1. $Q' = \mathcal{P}(Q)$. Every state of M is a set of states of N . Recall that $\mathcal{P}(Q)$ is the set of subsets of Q .
2. For $R \in Q'$ and $a \in \Sigma$, let

$$\delta'(R, a) = \{q \in Q \mid q \in \delta(r, a) \text{ for some } r \in R\}.$$

If R is a state of M , it is also a set of states of N . When M reads

a symbol a in state R , it shows where a takes each state in R . Because each state may go to a set of states, we take the union of all these sets. Another way to write this expression is

$$\delta'(R, a) = \bigcup_{r \in R} \delta(r, a).$$

3. $q'_0 = \{q_0\}$. M starts in the state corresponding to the collection containing just the start state of N .

4. $F' = \{R \in Q' \mid R \text{ contains an accept state of } N\}$. The machine M accepts if one of the possible states that N could be in at this point is an accept state.

Now we need to consider the ε arrows.

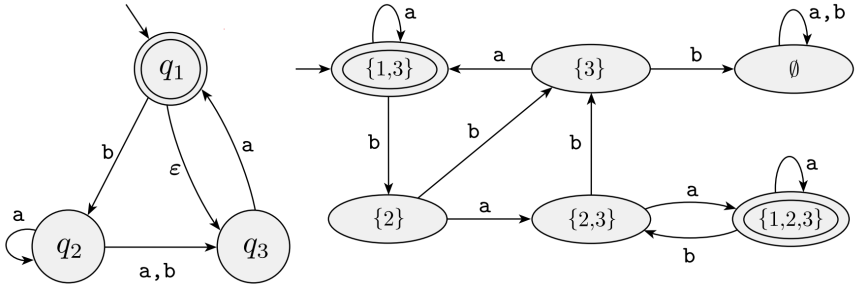
☞ We modify the transition function of M to place additional fingers on all states that can be reached by going along ε arrows after every step. Replacing $\delta(r, a)$ by $E(\delta(r, a))$ achieves this effect. Thus

$$\delta'(R, a) = \{q \in Q \mid q \in E(\delta(r, a)) \text{ for some } r \in R\} = \bigcup_{r \in R} E(\delta(r, a)).$$

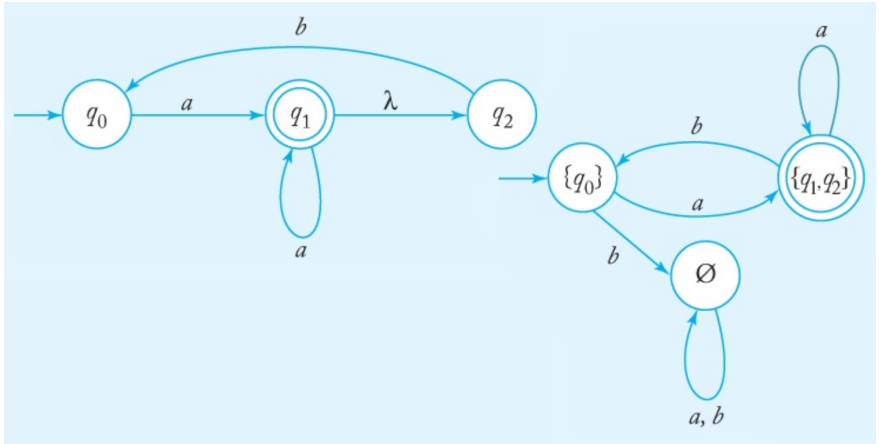
☞ Additionally, we need to modify the start state of M to move the fingers initially to all possible states that can be reached from the start state of N along the ε arrows. Changing q'_0 to be $E(\{q_0\})$ achieves this effect.

☞ We have now completed the construction of the DFA M that simulates the NFA N . The construction of M obviously works correctly. At every step in the computation of M on an input, it clearly enters a state that corresponds to the subset of states that N could be in at that point. **Thus our proof is complete.**

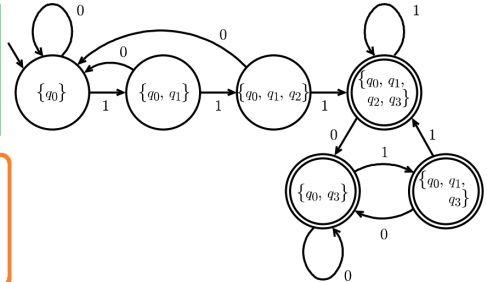
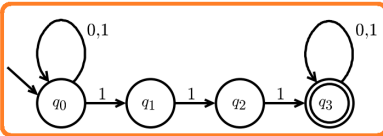
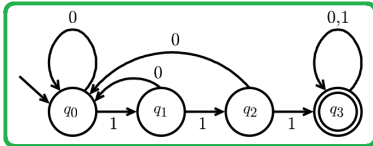
An NFA and its equivalent DFA: Example 2



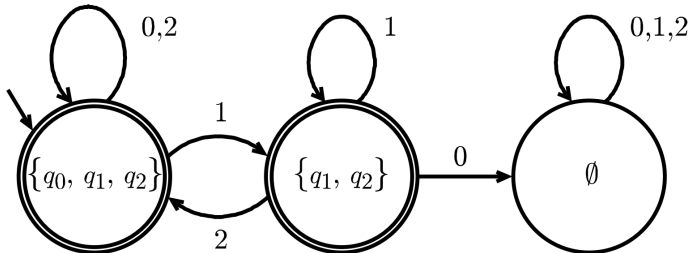
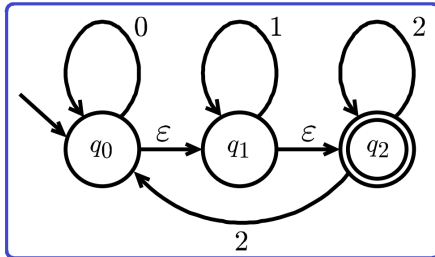
An NFA and its equivalent DFA: Example 3



An NFA and its equivalent DFA: Example 4



An NFA and its equivalent DFA: Example 5

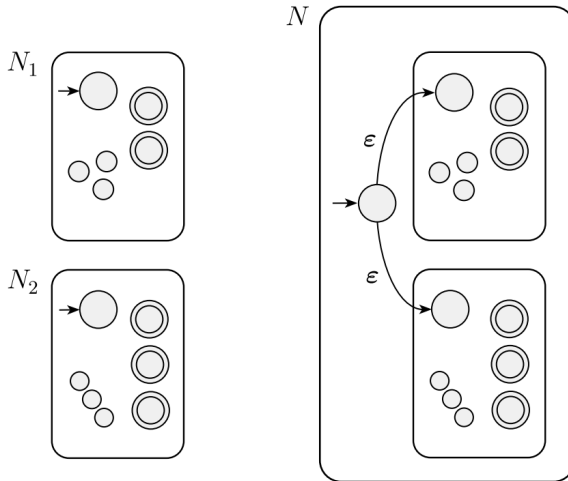


Now we return to the closure of the class of regular languages under the operations \cup , \circ , and $$. Our aim is to prove that the union, concatenation, and star of regular languages are still regular.*

The use of nondeterminism makes the proofs much easier.

*First, let's consider again closure under **union**. Earlier we proved closure under union by simulating deterministically both machines simultaneously via **a Cartesian product construction**. We now give a new proof to illustrate the technique of nondeterminism. Reviewing the first proof may be worthwhile to see how much easier and more intuitive the new proof is.*

The class of regular languages is closed under the union operation.



Let $N_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ recognize A_1 , and
 $N_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ recognize A_2 .

Construct $N = (Q, \Sigma, \delta, q_0, F)$ to recognize $A_1 \cup A_2$.

1. $Q = \{q_0\} \cup Q_1 \cup Q_2$.

The states of N are all the states of N_1 and N_2 , with the addition of a new start state q_0 .

2. The state q_0 is the start state of N .

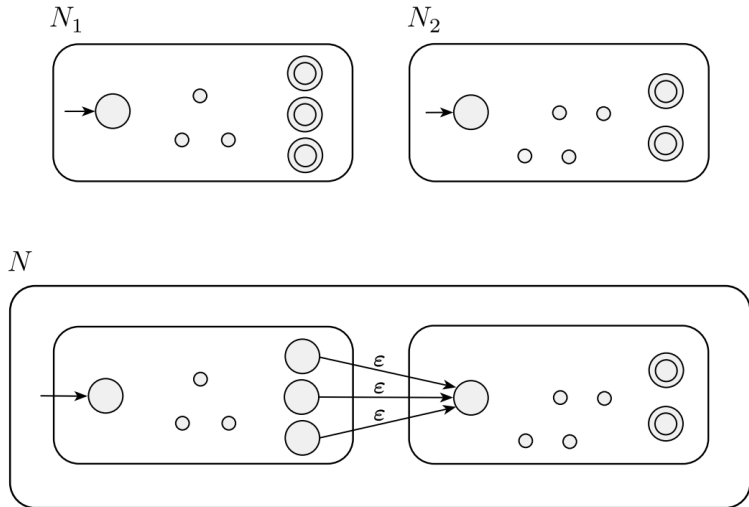
3. The set of accept states $F = F_1 \cup F_2$.

The accept states of N are all the accept states of N_1 and N_2 . That way, N accepts if either N_1 accepts or N_2 accepts.

4. Define δ so that for any $q \in Q$ and any $a \in \Sigma_\epsilon$,

$$\delta(q, a) = \begin{cases} \delta_1(q, a) & q \in Q_1 \\ \delta_2(q, a) & q \in Q_2 \\ \{q_1, q_2\} & q = q_0 \text{ and } a = \epsilon \\ \emptyset & q = q_0 \text{ and } a \neq \epsilon. \end{cases}$$

The class of regular languages is closed under the concatenation operation.



Let $N_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ recognize A_1 , and
 $N_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ recognize A_2 .

Construct $N = (Q, \Sigma, \delta, q_1, F_2)$ to recognize $A_1 \circ A_2$.

1. $Q = Q_1 \cup Q_2$.

The states of N are all the states of N_1 and N_2 .

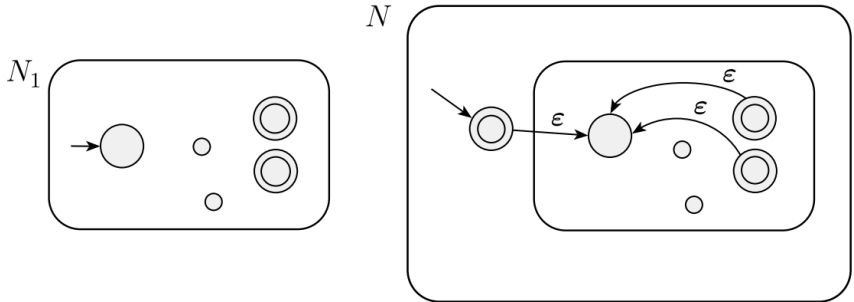
2. The state q_1 is the same as the start state of N_1 .

3. The accept states F_2 are the same as the accept states of N_2 .

4. Define δ so that for any $q \in Q$ and any $a \in \Sigma_\varepsilon$,

$$\delta(q, a) = \begin{cases} \delta_1(q, a) & q \in Q_1 \text{ and } q \notin F_1 \\ \delta_1(q, a) & q \in F_1 \text{ and } a \neq \varepsilon \\ \delta_1(q, a) \cup \{q_2\} & q \in F_1 \text{ and } a = \varepsilon \\ \delta_2(q, a) & q \in Q_2. \end{cases}$$

The class of regular languages is closed under the star operation.



PROOF Let $N_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ recognize A_1 .

Construct $N = (Q, \Sigma, \delta, q_0, F)$ to recognize A_1^* .

1. $Q = \{q_0\} \cup Q_1$.

The states of N are the states of N_1 plus a new start state.

2. The state q_0 is the new start state.

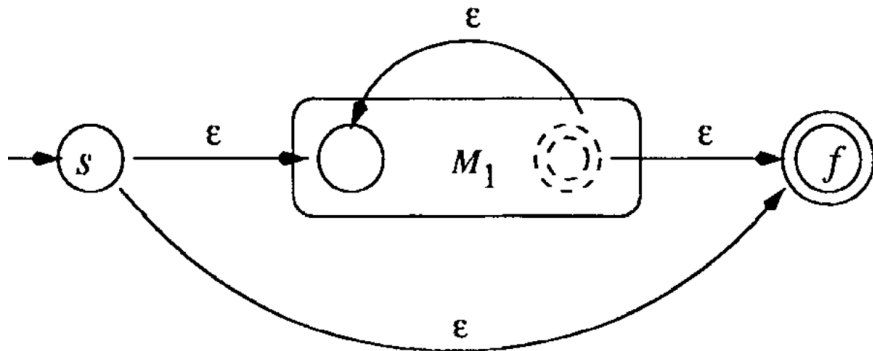
3. $F = \{q_0\} \cup F_1$.

The accept states are the old accept states plus the new start state.

4. Define δ so that for any $q \in Q$ and any $a \in \Sigma_\epsilon$,

$$\delta(q, a) = \begin{cases} \delta_1(q, a) & q \in Q_1 \text{ and } q \notin F_1 \\ \delta_1(q, a) & q \in F_1 \text{ and } a \neq \epsilon \\ \delta_1(q, a) \cup \{q_1\} & q \in F_1 \text{ and } a = \epsilon \\ \{q_1\} & q = q_0 \text{ and } a = \epsilon \\ \emptyset & q = q_0 \text{ and } a \neq \epsilon. \end{cases}$$

یک شیوه ساخت معتبر دیگر:



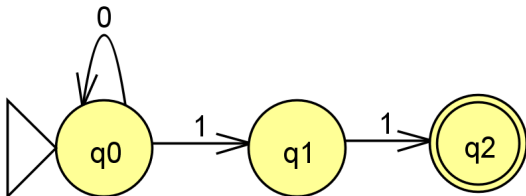
تمرین ۱.۱۵ کتاب: درباره یک ایده ساخت بد

1.15 Give a counterexample to show that the following construction fails to prove Theorem 1.49, the closure of the class of regular languages under the star operation.⁷ Let $N_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ recognize A_1 . Construct $N = (Q_1, \Sigma, \delta, q_1, F)$ as follows. N is supposed to recognize A_1^* .

- a. The states of N are the states of N_1 .
- b. The start state of N is the same as the start state of N_1 .
- c. $F = \{q_1\} \cup F_1$.
The accept states F are the old accept states plus its start state.
- d. Define δ so that for any $q \in Q_1$ and any $a \in \Sigma_\varepsilon$,

$$\delta(q, a) = \begin{cases} \delta_1(q, a) & q \notin F_1 \text{ or } a \neq \varepsilon \\ \delta_1(q, a) \cup \{q_1\} & q \in F_1 \text{ and } a = \varepsilon. \end{cases}$$

(Suggestion: Show this construction graphically, as in Figure 1.50.)



A bad idea (Exercise 1.15):

