

# **CS 228 : Logic in Computer Science**

Krishna. S

# So Far

---

- ▶  $\omega$ -automata with Büchi acceptance, also called Büchi automata
- ▶ Non-determinism versus determinism

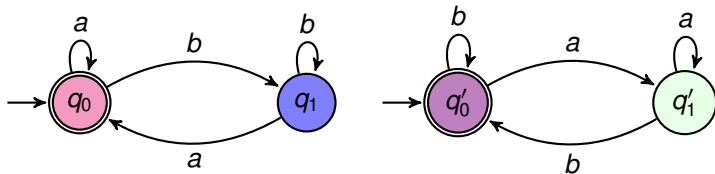
# Büchi Acceptance

---

A language  $L \subseteq \Sigma^\omega$  is called  $\omega$ -regular if there exists a NBA  $\mathcal{A}$  such that  $L = L(\mathcal{A})$ .

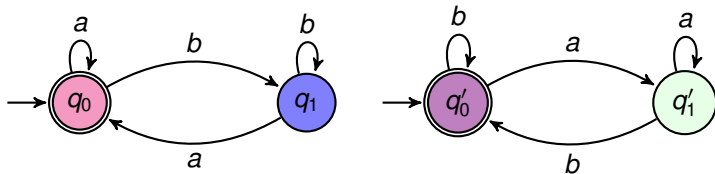
# Union and Intersection of NBA

---



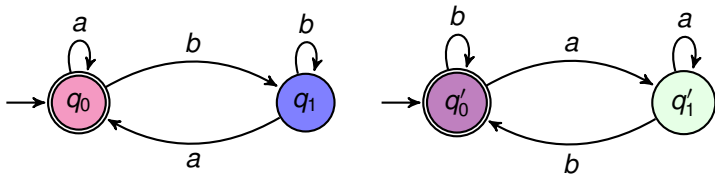
# Union and Intersection of NBA

---



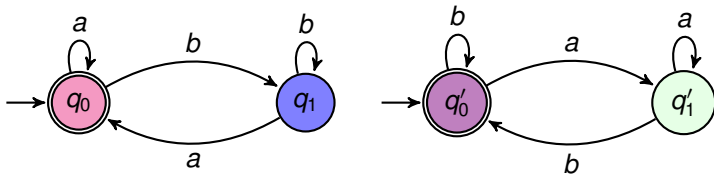
- States as  $Q_1 \times Q_2 \times \{1, 2\}$ , start state  $(q_0, q'_0, 1)$

# Union and Intersection of NBA



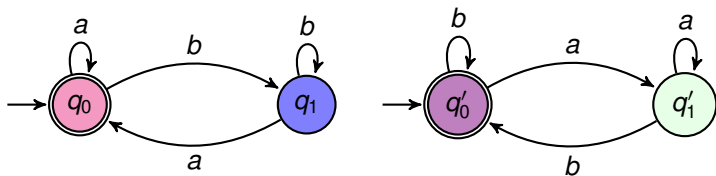
- ▶ States as  $Q_1 \times Q_2 \times \{1, 2\}$ , start state  $(q_0, q'_0, 1)$
- ▶  $(q_1, q_2, 1) \xrightarrow{a} (q'_1, q'_2, 1)$  if  $q_1 \xrightarrow{a} q'_1$  and  $q_2 \xrightarrow{a} q'_2$  and  $q_1 \notin G_1$
- ▶  $(q_1, q_2, 1) \xrightarrow{a} (q'_1, q'_2, 2)$  if  $q_1 \xrightarrow{a} q'_1$  and  $q_2 \xrightarrow{a} q'_2$  and  $q_1 \in G_1$

# Union and Intersection of NBA



- ▶ States as  $Q_1 \times Q_2 \times \{1, 2\}$ , start state  $(q_0, q'_0, 1)$
- ▶  $(q_1, q_2, 1) \xrightarrow{a} (q'_1, q'_2, 1)$  if  $q_1 \xrightarrow{a} q'_1$  and  $q_2 \xrightarrow{a} q'_2$  and  $q_1 \notin G_1$
- ▶  $(q_1, q_2, 1) \xrightarrow{a} (q'_1, q'_2, 2)$  if  $q_1 \xrightarrow{a} q'_1$  and  $q_2 \xrightarrow{a} q'_2$  and  $q_1 \in G_1$
- ▶  $(q_1, q_2, 2) \xrightarrow{a} (q'_1, q'_2, 2)$  if  $q_1 \xrightarrow{a} q'_1$  and  $q_2 \xrightarrow{a} q'_2$  and  $q_2 \notin G_2$
- ▶  $(q_1, q_2, 2) \xrightarrow{a} (q'_1, q'_2, 1)$  if  $q_1 \xrightarrow{a} q'_1$  and  $q_2 \xrightarrow{a} q'_2$  and  $q_2 \in G_2$

# Union and Intersection of NBA

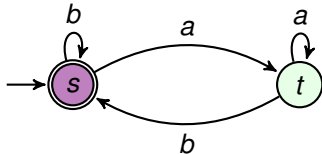
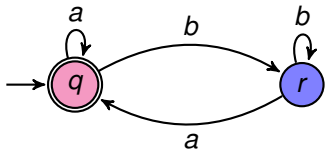


- ▶ States as  $Q_1 \times Q_2 \times \{1, 2\}$ , start state  $(q_0, q'_0, 1)$
- ▶  $(q_1, q_2, 1) \xrightarrow{a} (q'_1, q'_2, 1)$  if  $q_1 \xrightarrow{a} q'_1$  and  $q_2 \xrightarrow{a} q'_2$  and  $q_1 \notin G_1$
- ▶  $(q_1, q_2, 1) \xrightarrow{a} (q'_1, q'_2, 2)$  if  $q_1 \xrightarrow{a} q'_1$  and  $q_2 \xrightarrow{a} q'_2$  and  $q_1 \in G_1$
- ▶  $(q_1, q_2, 2) \xrightarrow{a} (q'_1, q'_2, 2)$  if  $q_1 \xrightarrow{a} q'_1$  and  $q_2 \xrightarrow{a} q'_2$  and  $q_2 \notin G_2$
- ▶  $(q_1, q_2, 2) \xrightarrow{a} (q'_1, q'_2, 1)$  if  $q_1 \xrightarrow{a} q'_1$  and  $q_2 \xrightarrow{a} q'_2$  and  $q_2 \in G_2$
- ▶ Good states =  $Q_1 \times G_2 \times \{2\}$  or  $G_1 \times Q_2 \times \{1\}$



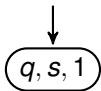
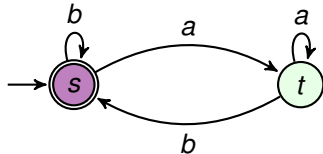
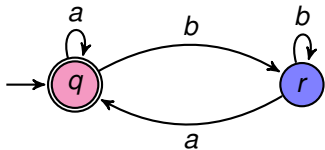
# Union and Intersection of NBA

---

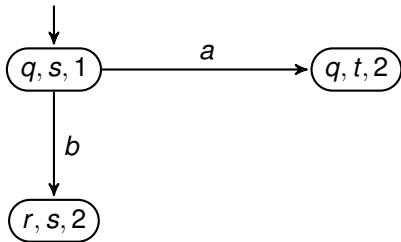
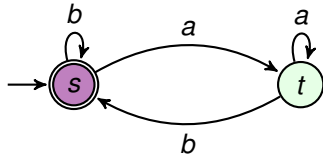
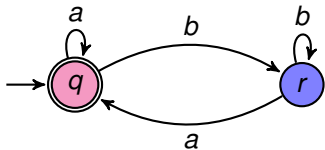


# Union and Intersection of NBA

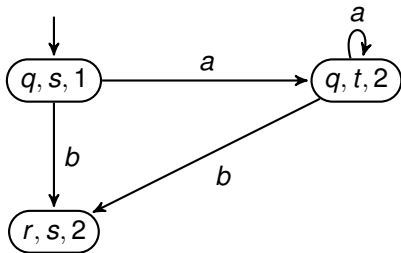
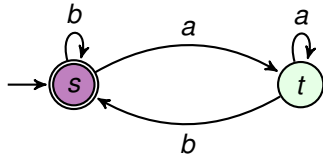
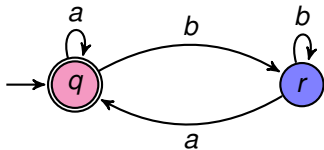
---



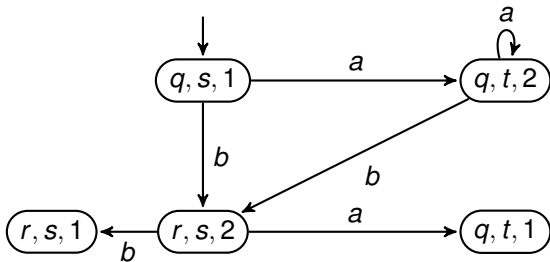
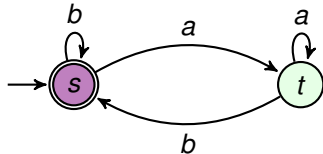
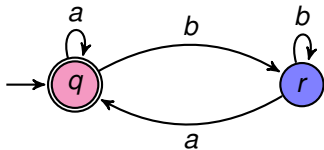
# Union and Intersection of NBA



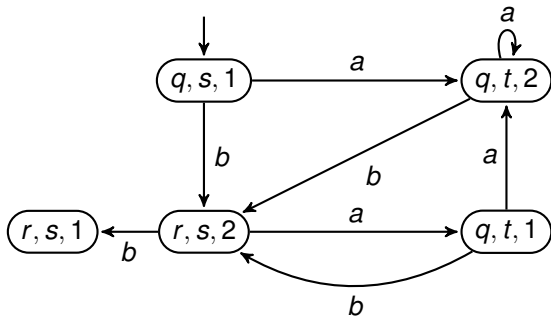
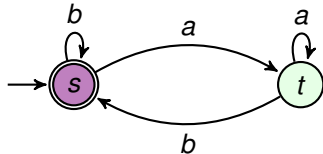
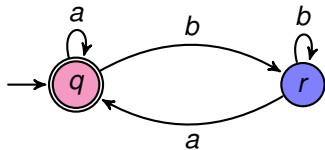
# Union and Intersection of NBA



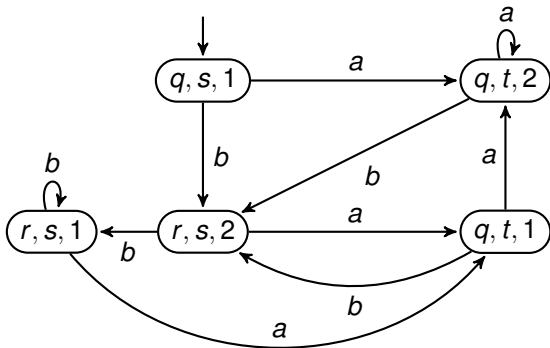
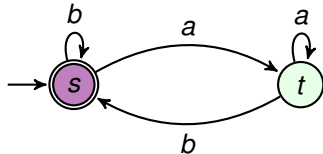
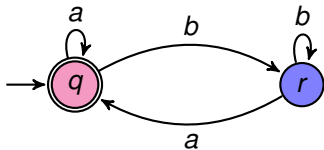
# Union and Intersection of NBA



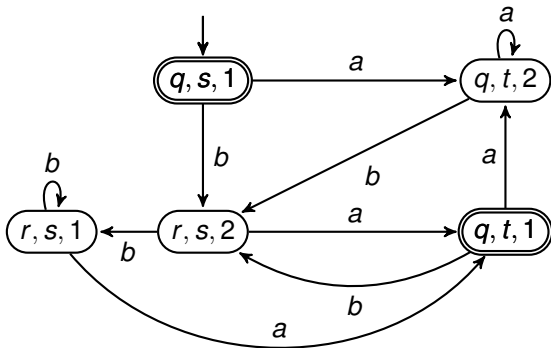
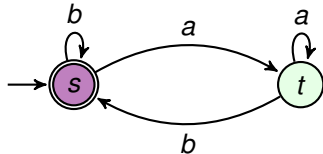
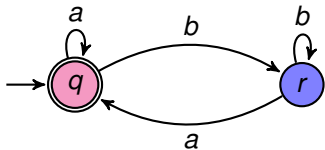
# Union and Intersection of NBA



# Union and Intersection of NBA



# Union and Intersection of NBA





# Emptiness

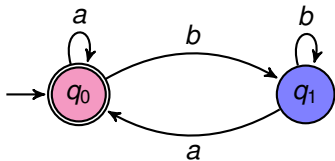
---

Given an NBA/DBA  $\mathcal{A}$ , how do you check if  $L(\mathcal{A}) = \emptyset$ ?

- ▶ Enumerate SCCs
- ▶ Check if there is an SCC containing a good state

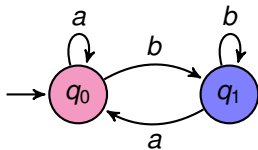
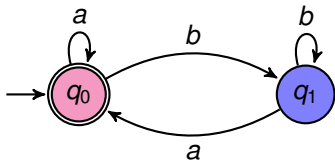
# Complementation of DBA

---



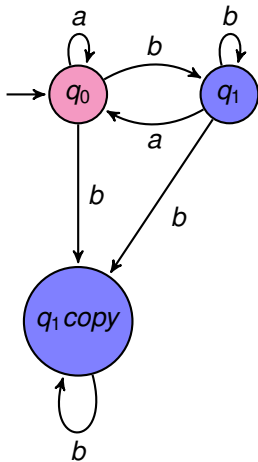
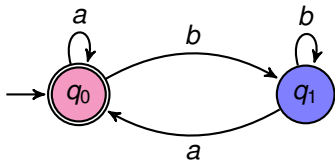
# Complementation of DBA

---



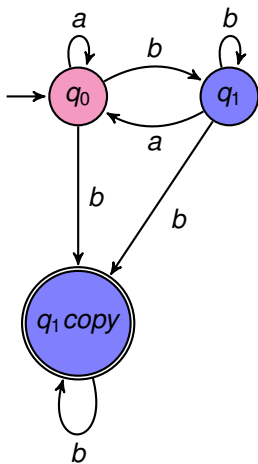
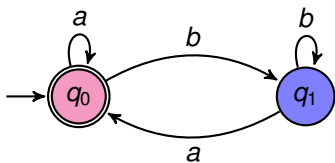
# Complementation of DBA

---



# Complementation of DBA

---



# Complementation of DBA

---

- ▶ Given  $\mathcal{A}$  is a DBA, and  $w \notin L(\mathcal{A})$ , then after some finite prefix, the unique run of  $w$  settles in bad states.
- ▶ Idea for complement: “copy” states of  $Q - G$ , once you enter this block, you stay there.
- ▶ View this as the set of good states, any word  $w$  that was rejected by  $\mathcal{A}$  has two possible runs in this automaton: the original run, and one another, that will settle in the  $Q - G$  copy, and will be accepted.
- ▶ What we get now is an NBA for  $\overline{L(\mathcal{A})}$ , not a DBA.

Complementing NBA non-trivial, can be done.

# Normal Form for $\omega$ -regular languages

An  $\omega$ -regular language  $L \subseteq \Sigma^\omega$  can be written as  $L = \bigcup_{i=1}^n U_i V_i^\omega$ , where  $U_i, V_i$  are regular languages.

One direction : Assume  $L$  is accepted by an NBA/DBA.

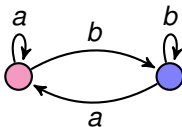
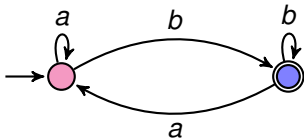
- ▶ Define  $U_g = \{w \in \Sigma^* \mid q_0 \xrightarrow{w} g\}$
- ▶ Define  $V_g = \{w \in \Sigma^* \mid g \xrightarrow{w} g\}$
- ▶ Then  $L = \bigcup_{g \in G} U_g V_g^\omega$ , where  $U_g, V_g$  are regular
- ▶ Show that  $U_g, V_g$  are regular.

# Normal Form for $\omega$ -regular languages

An  $\omega$ -regular language  $L \subseteq \Sigma^\omega$  can be written as  $L = \bigcup_{i=1}^n U_i V_i^\omega$ , where  $U_i, V_i$  are regular languages.

Other direction : Assume  $L = \bigcup_{i=1}^n U_i V_i^\omega$ . Show that  $L$  is accepted by an NBA/DBA.

1. If  $V$  is regular,  $V^\omega$  is  $\omega$ -regular



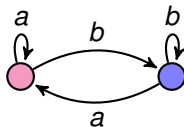
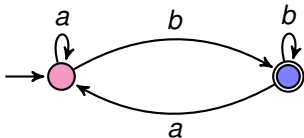


# Normal Form for $\omega$ -regular languages

An  $\omega$ -regular language  $L \subseteq \Sigma^\omega$  can be written as  $L = \bigcup_{i=1}^n U_i V_i^\omega$ , where  $U_i, V_i$  are regular languages.

Other direction : Assume  $L = \bigcup_{i=1}^n U_i V_i^\omega$ . Show that  $L$  is accepted by an NBA/DBA.

1. If  $V$  is regular,  $V^\omega$  is  $\omega$ -regular

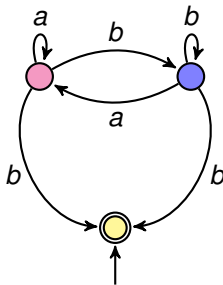
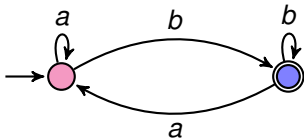


# Normal Form for $\omega$ -regular languages

An  $\omega$ -regular language  $L \subseteq \Sigma^\omega$  can be written as  $L = \bigcup_{i=1}^n U_i V_i^\omega$ , where  $U_i, V_i$  are regular languages.

Other direction : Assume  $L = \bigcup_{i=1}^n U_i V_i^\omega$ . Show that  $L$  is accepted by an NBA/DBA.

1. If  $V$  is regular,  $V^\omega$  is  $\omega$ -regular

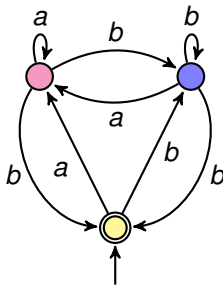
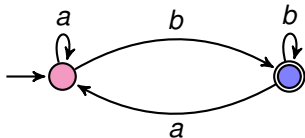


# Normal Form for $\omega$ -regular languages

An  $\omega$ -regular language  $L \subseteq \Sigma^\omega$  can be written as  $L = \bigcup_{i=1}^n U_i V_i^\omega$ , where  $U_i, V_i$  are regular languages.

Other direction : Assume  $L = \bigcup_{i=1}^n U_i V_i^\omega$ . Show that  $L$  is accepted by an NBA/DBA.

1. If  $V$  is regular,  $V^\omega$  is  $\omega$ -regular



# Normal Form for $\omega$ -regular languages

---

1. If  $V$  is regular,  $V^\omega$  is  $\omega$ -regular
  - ▶ Let  $D = (Q, \Sigma, q_0, \delta, F)$  be a DFA accepting  $V$
  - ▶ Construct NBA  $E = (Q \cup \{p_0\}, \Sigma, p_0, \Delta, G)$  such that  $G = \{p_0\}$ ,
  - ▶  $\Delta = \delta \cup \{p_0 \in \Delta(q, a) \mid \delta(q, a) \in F\} \cup \{\Delta(p_0, a) = s \mid \delta(q_0, a) = s\}$
2. Show that if  $U$  is regular and  $V^\omega$  is  $\omega$ -regular, then  $UV^\omega$  is  $\omega$ -regular
  - ▶  $D = (Q_1, \Sigma, q_0, \delta_1, F)$  be a DFA,  $L(D) = U$  and  $E = (Q_2, \Sigma, q'_0, \delta_2, G)$  be an NBA,  $L(E) = V^\omega$ .
  - ▶  $A = (Q_1 \cup Q_2, \Sigma, q_0, \delta', G)$  NBA such that  $\delta' = \delta_1 \cup \delta_2 \cup \{(q, a, q'_0) \mid \delta_1(q, a) \in F\}$