CS250: Discrete Math for Computer Science

L33: Kleene's Theorem

Thm: For every n-state NFA, $N = (Q, \Sigma, \Delta, s, F)$, there is an equivalent 2^n-state DFA.

Thm: For every n-state NFA, $N = (Q, \Sigma, \Delta, s, F)$, there is an equivalent 2^n-state DFA.

proof: We may assume that N has no ϵ moves. Let

$$\begin{aligned} D &= (\wp(Q), \Sigma, \delta, \{s\}, \{T \subseteq Q \mid T \cap F \neq \emptyset\}) \\ \delta(T, a) &= \bigcup_{q \in T} \Delta(q, a) \end{aligned}$$

Thm: For every n-state NFA, $N = (Q, \Sigma, \Delta, s, F)$, there is an equivalent 2^n-state DFA.

proof: We may assume that N has no ϵ moves. Let

$$D = (\wp(Q), \Sigma, \delta, \{s\}, \{T \subseteq Q \mid T \cap F \neq \emptyset\})$$

$$\delta(T, a) = \bigcup_{q \in T} \Delta(q, a)$$

Claim: $\forall w \in \Sigma^\star \; (\delta^\star(\{s\}, w) = \Delta^\star(s, w))$

state D is in after w = **the set of states *N* can be in after w**

Thm: For every n-state NFA, $N = (Q, \Sigma, \Delta, s, F)$, there is an equivalent 2^n-state DFA.

proof: We may assume that N has no ϵ moves. Let

$$D = (\wp(Q), \Sigma, \delta, \{s\}, \{T \subseteq Q \mid T \cap F \neq \emptyset\})$$

$$\delta(T, a) = \bigcup_{q \in T} \Delta(q, a)$$

Claim: $\forall w \in \Sigma^\star \ (\delta^\star(\{s\}, w) = \Delta^\star(s, w))$

state D is in after w = **the set of states *N* can be in after w**

pf: by induction on $|w|$ **base case:** $\delta^\star(\{s\}, \epsilon) = \{s\} = \Delta^\star(s, \epsilon)$

Thm: For every n-state NFA, $N = (Q, \Sigma, \Delta, s, F)$, there is an equivalent 2^n-state DFA.

proof: We may assume that N has no ϵ moves. Let

$$D = (\wp(Q), \Sigma, \delta, \{s\}, \{T \subseteq Q \mid T \cap F \neq \emptyset\})$$

$$\delta(T, a) = \bigcup_{q \in T} \Delta(q, a)$$

Claim: $\forall w \in \Sigma^\star \, (\delta^\star(\{s\}, w) = \Delta^\star(s, w))$

state D is in after w = **the set of states *N* can be in after w**

pf: by induction on $|w|$ **base case:** $\delta^\star(\{s\}, \epsilon) = \{s\} = \Delta^\star(s, \epsilon)$

inductive case: indHyp: for $|w| = n_0$, $\delta^\star(\{s\}, w) = \Delta^\star(s, w)$.

Thm: For every n-state NFA, $N = (Q, \Sigma, \Delta, s, F)$, there is an equivalent 2^n-state DFA.

proof: We may assume that N has no ϵ moves. Let

$$D = (\wp(Q), \Sigma, \delta, \{s\}, \{T \subseteq Q \mid T \cap F \neq \emptyset\})$$

$$\delta(T, a) = \bigcup_{q \in T} \Delta(q, a)$$

Claim: $\forall w \in \Sigma^\star \, (\delta^\star(\{s\}, w) = \Delta^\star(s, w))$

state D is in after w = **the set of states N can be in after w**

pf: by induction on $|w|$ **base case:** $\delta^\star(\{s\}, \epsilon) = \{s\} = \Delta^\star(s, \epsilon)$

inductive case: indHyp: for $|w| = n_0$, $\delta^\star(\{s\}, w) = \Delta^\star(s, w)$.

Let $|x| = n_0 + 1$. Then $x = wa$ for some $w \in \Sigma^{n_0}$, $a \in \Sigma$

Thm: For every n-state NFA, $N = (Q, \Sigma, \Delta, s, F)$, there is an equivalent 2^n-state DFA.

proof: We may assume that N has no ϵ moves. Let

$$D = (\wp(Q), \Sigma, \delta, \{s\}, \{T \subseteq Q \mid T \cap F \neq \emptyset\})$$

$$\delta(T, a) = \bigcup_{q \in T} \Delta(q, a)$$

Claim: $\forall w \in \Sigma^\star \; (\delta^\star(\{s\}, w) = \Delta^\star(s, w))$

state D is in after w = **the set of states N can be in after w**

pf: by induction on $|w|$ **base case:** $\delta^\star(\{s\}, \epsilon) = \{s\} = \Delta^\star(s, \epsilon)$

inductive case: indHyp: for $|w| = n_0$, $\delta^\star(\{s\}, w) = \Delta^\star(s, w)$.

Let $|x| = n_0 + 1$. Then $x = wa$ for some $w \in \Sigma^{n_0}$, $a \in \Sigma$

$$\delta^\star(\{s\}, wa) = \delta(\delta^\star(\{s\}, w), a) = \delta(\Delta^\star(s, w), a)$$

Thm: For every n-state NFA, $N = (Q, \Sigma, \Delta, s, F)$, there is an equivalent 2^n-state DFA.

proof: We may assume that N has no ϵ moves. Let

$$\begin{aligned} D &= (\wp(Q), \Sigma, \delta, \{s\}, \{T \subseteq Q \mid T \cap F \neq \emptyset\}) \\ \delta(T, a) &= \bigcup_{q \in T} \Delta(q, a) \end{aligned}$$

Claim: $\forall w \in \Sigma^\star \ (\delta^\star(\{s\}, w) = \Delta^\star(s, w))$

state D is in after w = **the set of states *N* can be in after w**

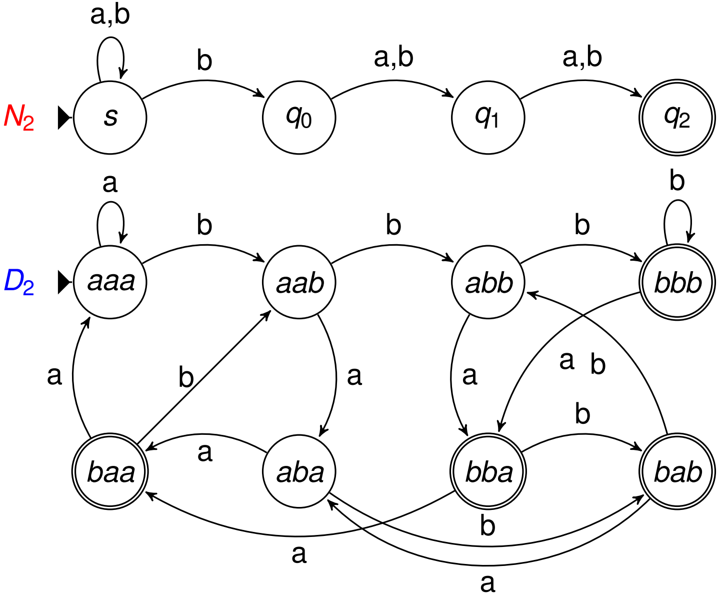
pf: by induction on $|w|$ **base case:** $\delta^\star(\{s\}, \epsilon) = \{s\} = \Delta^\star(s, \epsilon)$

inductive case: indHyp: for $|w| = n_0$, $\delta^\star(\{s\}, w) = \Delta^\star(s, w)$.

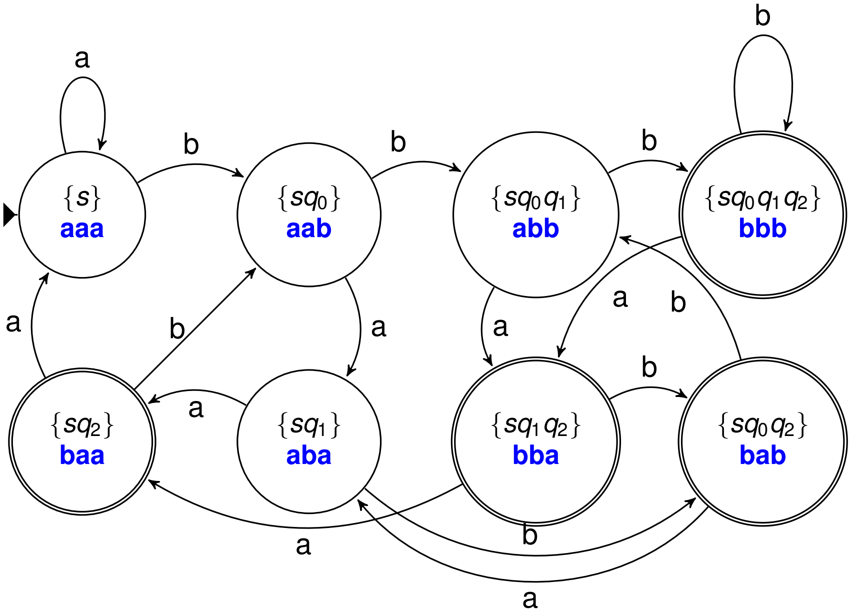
Let $|x| = n_0 + 1$. Then $x = wa$ for some $w \in \Sigma^{n_0}$, $a \in \Sigma$

$$\begin{aligned} \delta^\star(\{s\}, wa) &= \delta(\delta^\star(\{s\}, w), a) = \delta(\Delta^\star(s, w), a) \\ &= \bigcup_{q \in \Delta^\star(s,w)} \Delta(q, a) = \Delta^\star(s, wa) \qquad \square \end{aligned}$$

$\mathcal{L}(N_2) = \mathcal{L}(D_2)$



$\mathcal{L}(N_2) = \mathcal{L}(D_2)$



Kleene's Theorem Let $A \subseteq \Sigma^\star$ be any language. Then the following are equivalent:

1. $A = \mathcal{L}(D)$, for some DFA D.
2. $A = \mathcal{L}(N)$, for some NFA N wo ϵ transitions.
3. $A = \mathcal{L}(N)$, for some NFA N.
4. $A = \mathcal{L}(e)$, for some regular expression e.
5. A is regular.

Kleene's Theorem Let $A \subseteq \Sigma^\star$ be any language. Then the following are equivalent:

1. $A = \mathcal{L}(D)$, for some DFA D.
2. $A = \mathcal{L}(N)$, for some NFA N wo ϵ transitions.
3. $A = \mathcal{L}(N)$, for some NFA N.
4. $A = \mathcal{L}(e)$, for some regular expression e.
5. A is regular.

Already shown: (1) $\Leftrightarrow$ (2) $\Leftrightarrow$ (3)

Kleene's Theorem Let $A \subseteq \Sigma^{\star}$ be any language. Then the following are equivalent:

1. $A = \mathcal{L}(D)$, for some DFA D.
2. $A = \mathcal{L}(N)$, for some NFA N wo ϵ transitions.
3. $A = \mathcal{L}(N)$, for some NFA N.
4. $A = \mathcal{L}(e)$, for some regular expression e.
5. A is regular.

Already shown: $(1) \Leftrightarrow (2) \Leftrightarrow (3)$

True by definition: $(4) \Leftrightarrow (5)$

Kleene's Theorem Let $A \subseteq \Sigma^\star$ be any language. Then the following are equivalent:

1. $A = \mathcal{L}(D)$, for some DFA D.
2. $A = \mathcal{L}(N)$, for some NFA N wo ϵ transitions.
3. $A = \mathcal{L}(N)$, for some NFA N.
4. $A = \mathcal{L}(e)$, for some regular expression e.
5. A is regular.

Already shown: $(1) \Leftrightarrow (2) \Leftrightarrow (3)$

True by definition: $(4) \Leftrightarrow (5)$

Now, we will prove: $(4) \Rightarrow (3) \ \wedge \ (1) \Rightarrow (4)$

(4) $\Rightarrow$ (3): $\forall e \in \mathrm{regexp}(\Sigma)\ \exists$ NFA N $(\mathcal{L}(e) = \mathcal{L}(N))$.

(4) $\Rightarrow$ (3): $\forall e \in \mathrm{regexp}(\Sigma)\ \exists$ NFA N $(\mathcal{L}(e) = \mathcal{L}(N))$.

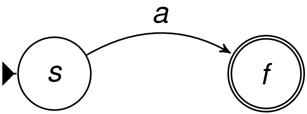
proof: by induction on e.

(4) $\Rightarrow$ (3): $\forall e \in \mathrm{regexp}(\Sigma)\ \exists$ NFA N $(\mathcal{L}(e) = \mathcal{L}(N))$.

proof: by induction on e.

base cases:

$$e = a$$

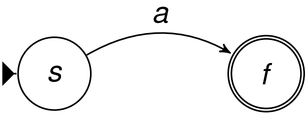


(4) $\Rightarrow$ (3): $\forall e \in \mathrm{regexp}(\Sigma)\ \exists$ NFA N $(\mathcal{L}(e) = \mathcal{L}(N))$.

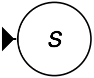
proof: by induction on e.

base cases:

$e = a$



$e = \emptyset$



inductive cases: Assume **indHyp:** $\mathcal{L}(N_i) = \mathcal{L}(e_i),\ i = 1, 2$

$$N_i \quad = \quad (Q_i, \Sigma, \Delta_i, s_i, F_i) \qquad Q_1 \cap Q_2 = \emptyset$$

inductive cases: Assume **indHyp:** $\mathcal{L}(N_i) = \mathcal{L}(e_i),\ i = 1, 2$

$$N_i \quad = \quad (Q_i, \Sigma, \Delta_i, s_i, F_i) \qquad Q_1 \cap Q_2 = \emptyset$$

$e \ = \ e_1 | e_2$

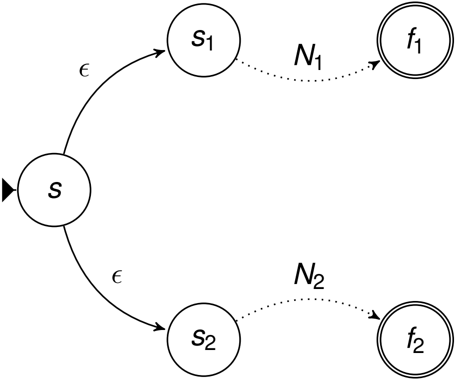
inductive cases: Assume **indHyp:** $\mathcal{L}(N_i) = \mathcal{L}(e_i),\ i = 1, 2$

$$N_i \quad = \quad (Q_i, \Sigma, \Delta_i, s_i, F_i) \qquad Q_1 \cap Q_2 = \emptyset$$

$$e \ = \ e_1 | e_2$$

$$N_{e_1|e_2} \quad = \quad (\{s\} \cup Q_1 \cup Q_2, \Sigma, \{(s, \epsilon, \{s_1, s_2\})\} \cup \Delta_1 \cup \Delta_2, s, F_1 \cup F_2)$$

$N_{e_1|e_2}$

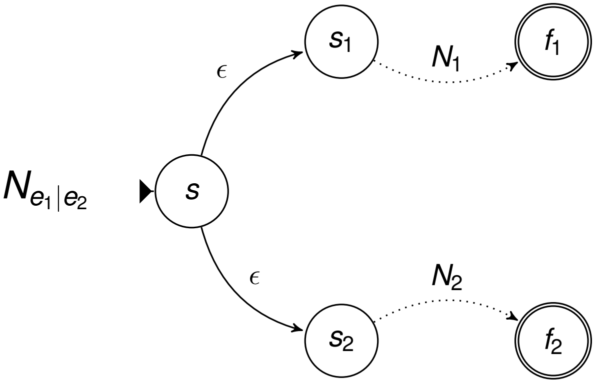


inductive cases: Assume **indHyp:** $\mathcal{L}(N_i) = \mathcal{L}(e_i),\ i = 1, 2$

$$N_i = (Q_i, \Sigma, \Delta_i, s_i, F_i) \qquad Q_1 \cap Q_2 = \emptyset$$

$$e = e_1|e_2 \qquad \mathcal{L}(N_{e_1|e_2}) = \mathcal{L}(N_1) \cup \mathcal{L}(N_2) = \mathcal{L}(e)$$

$$N_{e_1|e_2} = (\{s\} \cup Q_1 \cup Q_2, \Sigma, \{(s, \epsilon, \{s_1, s_2\})\} \cup \Delta_1 \cup \Delta_2, s, F_1 \cup F_2)$$



inductive cases: Assume **indHyp:** $\mathcal{L}(N_i) = \mathcal{L}(e_i), \; i = 1, 2$

$$N_i \quad = \quad (Q_i, \Sigma, \Delta_i, s_i, F_i) \qquad Q_1 \cap Q_2 = \emptyset$$

inductive cases: Assume **indHyp:** $\mathcal{L}(N_i) = \mathcal{L}(e_i),\ i = 1, 2$

$$N_i \quad = \quad (Q_i, \Sigma, \Delta_i, s_i, F_i) \qquad Q_1 \cap Q_2 = \emptyset$$

$e \ = \ e_1 e_2$

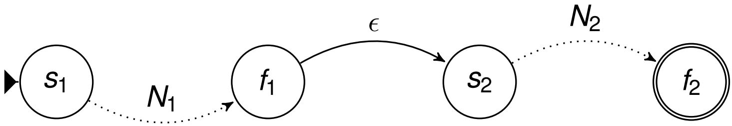
inductive cases: Assume **indHyp:** $\mathcal{L}(N_i) = \mathcal{L}(e_i),\ i = 1, 2$

$$N_i \quad = \quad (Q_i, \Sigma, \Delta_i, s_i, F_i) \qquad Q_1 \cap Q_2 = \emptyset$$

$e \ = \ e_1 e_2$

$$N_{e_1 e_2} \ = \ (Q_1 \cup Q_2, \Sigma, \{(f_1, \epsilon, \{s_2\}) \mid f_1 \in F_1\} \cup \Delta_1 \cup \Delta_2, s_1, F_2)$$

$N_{e_1 e_2}$



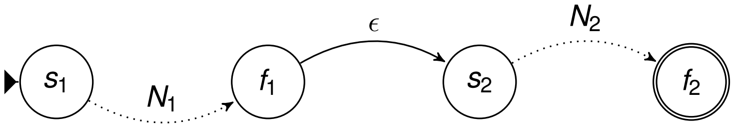
inductive cases: Assume **indHyp:** $\mathcal{L}(N_i) = \mathcal{L}(e_i),\ i = 1, 2$

$$N_i \quad = \quad (Q_i, \Sigma, \Delta_i, s_i, F_i) \qquad Q_1 \cap Q_2 = \emptyset$$

$$e \ = \ e_1 e_2 \qquad \mathcal{L}(N_{e_1 e_2}) \ = \ \mathcal{L}(N_1) \cdot \mathcal{L}(N_2) \ = \ \mathcal{L}(e)$$

$$N_{e_1 e_2} \ = \ (Q_1 \cup Q_2, \Sigma, \big\{(f_1, \epsilon, \{s_2\}) \ \big| \ f_1 \in F_1\big\} \cup \Delta_1 \cup \Delta_2, s_1, F_2)$$

$N_{e_1 e_2}$



inductive cases: Assume **indHyp:** $\mathcal{L}(N_i) = \mathcal{L}(e_i),\ i = 1, 2$

$$N_i \quad = \quad (Q_i, \Sigma, \Delta_i, s_i, F_i)$$

inductive cases: Assume **indHyp:** $\mathcal{L}(N_i) = \mathcal{L}(e_i),\ i = 1, 2$

$$N_i \quad = \quad (Q_i, \Sigma, \Delta_i, s_i, F_i)$$

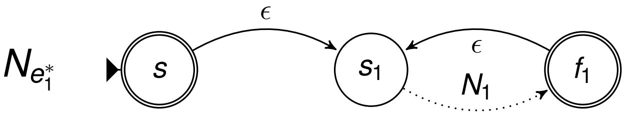
$e \ = \ e_1^\star$

inductive cases: Assume **indHyp:** $\mathcal{L}(N_i) = \mathcal{L}(e_i), \ i = 1, 2$

$$N_i \quad = \quad (Q_i, \Sigma, \Delta_i, s_i, F_i)$$

$$e \ = \ e_1^\star$$

$$N_{e_1^*} = (\{s\} \cup Q_1, \Sigma, \{(s, \epsilon, \{s_1\})\} \cup \Big\{(f_1, \epsilon, \{s_1\}) \ \Big| \ f_1 \in F_1\Big\} \cup \Delta_1, s, \{s\} \cup F_1)$$



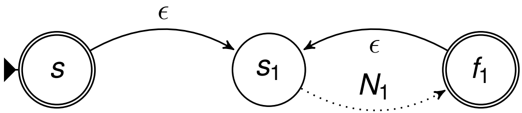
inductive cases: Assume **indHyp:** $\mathcal{L}(N_i) = \mathcal{L}(e_i),\ i = 1, 2$

$$N_i \quad = \quad (Q_i, \Sigma, \Delta_i, s_i, F_i)$$

$e \;=\; e_1^\star$ $$\mathcal{L}(N_{e_1^*}) \;=\; \mathcal{L}(N_1)^\star \;=\; \mathcal{L}(e)$$

$$N_{e_1^*} = (\{s\} \cup Q_1, \Sigma, \{(s, \epsilon, \{s_1\})\} \cup \big\{(f_1, \epsilon, \{s_1\}) \;\big|\; f_1 \in F_1\big\} \cup \Delta_1, s, \{s\} \cup F_1)$$

$N_{e_1^*}$



This completes the proof of $(4) \Rightarrow (3)$**.**

(1) $\Rightarrow$ (4): **Assume:** $A = \mathcal{L}(D)$, $D = (Q, \Sigma, \delta, s, F)$ where $Q = \{1, \ldots, n\}$, $F = \{f_1, \ldots, f_r\}$.

(1) $\Rightarrow$ (4): **Assume:** $A = \mathcal{L}(D)$, $D = (Q, \Sigma, \delta, s, F)$ where $Q = \{1, \ldots, n\}$, $F = \{f_1, \ldots, f_r\}$.

Use **dynamic programming algorithm** to construct $e \in \mathrm{regexp}(\Sigma)$ s.t. $\mathcal{L}(e) = \mathcal{L}(D)$. Build regexps e_{ij}^{k} s.t.

(1) $\Rightarrow$ (4): **Assume:** $A = \mathcal{L}(D)$, $D = (Q, \Sigma, \delta, s, F)$ where $Q = \{1, \ldots, n\}$, $F = \{f_1, \ldots, f_r\}$.

Use **dynamic programming algorithm** to construct $e \in \mathrm{regexp}(\Sigma)$ s.t. $\mathcal{L}(e) = \mathcal{L}(D)$. Build regexps e_{ij}^k s.t.

Goal : $\mathcal{L}(e_{ij}^k) = \{w \mid i \overset{\star}{\underset{w}{\to}} j;$ no intermediate state $\# > k\}$

(1) $\Rightarrow$ (4): **Assume:** $A = \mathcal{L}(D)$, $D = (Q, \Sigma, \delta, s, F)$ where $Q = \{1, \ldots, n\}$, $F = \{f_1, \ldots, f_r\}$.

Use **dynamic programming algorithm** to construct $e \in \mathrm{regexp}(\Sigma)$ s.t. $\mathcal{L}(e) = \mathcal{L}(D)$. Build regexps e_{ij}^k s.t.

Goal : $\mathcal{L}(e_{ij}^k) = \{ w \mid i \overset{\star}{\underset{w}{\to}} j;\ \text{no intermediate state} \# > k \}$

base case $e_{ij}^0 := \{ a \mid j = \delta(i, a) \} \mid \{ \epsilon \mid i = j \} \mid \emptyset$

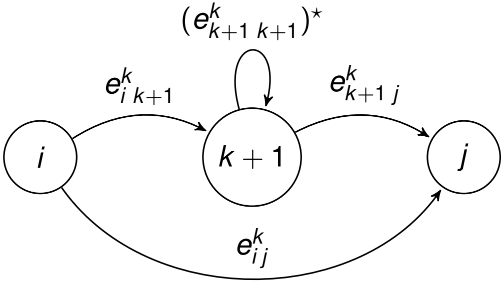
(1) $\Rightarrow$ (4): **Assume:** $A = \mathcal{L}(D)$, $D = (Q, \Sigma, \delta, s, F)$ where $Q = \{1, \ldots, n\}$, $F = \{f_1, \ldots, f_r\}$.

Use **dynamic programming algorithm** to construct $e \in \mathrm{regexp}(\Sigma)$ s.t. $\mathcal{L}(e) = \mathcal{L}(D)$. Build regexps e_{ij}^k s.t.

Goal : $\mathcal{L}(e_{ij}^k) = \{w \mid i \overset{\star}{\underset{w}{\rightarrow}} j;$ no intermediate state $\# > k\}$

base case $e_{ij}^0 := \{a \mid j = \delta(i, a)\} \mid \{\epsilon \mid i = j\} \mid \emptyset$

inductive case $e_{ij}^{k+1} := e_{ij}^k \mid e_{i\,k+1}^k \, (e_{k+1\,k+1}^k)^\star \, e_{k+1\,j}^k$



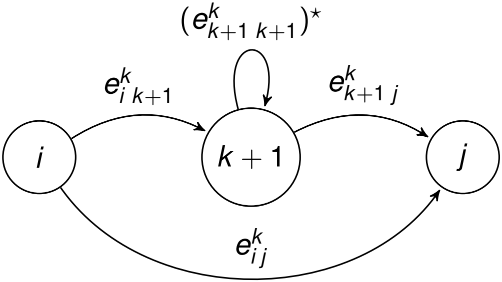
(1) $\Rightarrow$ (4): **Assume:** $A = \mathcal{L}(D)$, $D = (Q, \Sigma, \delta, s, F)$ where $Q = \{1, \ldots, n\}$, $F = \{f_1, \ldots, f_r\}$.

Use **dynamic programming algorithm** to construct $e \in \mathrm{regexp}(\Sigma)$ s.t. $\mathcal{L}(e) = \mathcal{L}(D)$. Build regexps e_{ij}^k s.t.

Goal : $\mathcal{L}(e_{ij}^k) = \{ w \mid i \xrightarrow[w]{\star} j;$ no intermediate state $\# > k \}$

base case $e_{ij}^0 := \{ a \mid j = \delta(i, a) \} \mid \{ \epsilon \mid i = j \} \mid \emptyset$

inductive case $e_{ij}^{k+1} := e_{ij}^k \mid e_{i\,k+1}^k \, (e_{k+1\,k+1}^k)^\star \, e_{k+1\,j}^k$



$e = e_{s\,f_1}^n | e_{s\,f_2}^n | \cdots | e_{s\,f_r}^n$

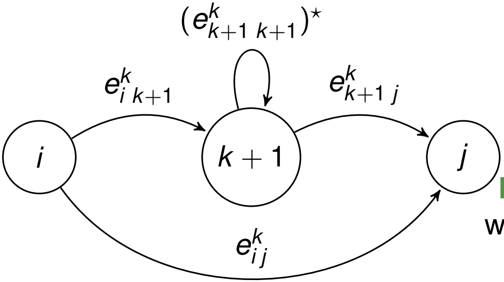
(1) $\Rightarrow$ (4): **Assume:** $A = \mathcal{L}(D)$, $D = (Q, \Sigma, \delta, s, F)$ where $Q = \{1, \ldots, n\}$, $F = \{f_1, \ldots, f_r\}$.

Use **dynamic programming algorithm** to construct $e \in \mathrm{regexp}(\Sigma)$ s.t. $\mathcal{L}(e) = \mathcal{L}(D)$. Build regexps e_{ij}^k s.t.

Goal : $\mathcal{L}(e_{ij}^k) = \{w \mid i \xrightarrow[w]{\star} j;$ no intermediate state $\# > k\}$

base case $e_{ij}^0 := \{a \mid j = \delta(i, a)\} \mid \{\epsilon \mid i = j\} \mid \emptyset$

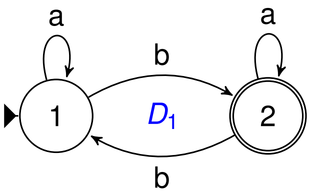
inductive case $e_{ij}^{k+1} := e_{ij}^k \mid e_{i\,k+1}^k \, (e_{k+1\,k+1}^k)^\star \, e_{k+1\,j}^k$



$e = e_{s\,f_1}^n | e_{s\,f_2}^n | \cdots | e_{s\,f_r}^n$

Note: this construction works for NFAs as well.

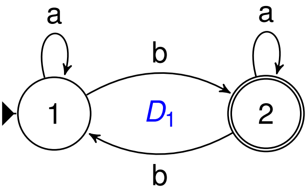
Goal :	$\mathcal{L}(e_{ij}^k)$	$=$	$\{ w \mid i \xrightarrow[w]{\star} j;\ \text{no intermediate state } \# > k \}$
base case	e_{ij}^0	$:=$	$\{ a \mid j = \delta(i,a) \} \mid \{ \epsilon \mid i = j \} \mid \emptyset$
inductive case	e_{ij}^{k+1}	$:=$	$e_{ij}^k \mid e_{i\,k+1}^k \,(e_{k+1\,k+1}^k)^\star\, e_{k+1\,j}^k$



Goal : $\mathcal{L}(e_{ij}^k) = \{ w \mid i \xrightarrow[w]{\star} j;\ \text{no intermediate state } \# > k \}$

base case $e_{ij}^0 := \{ a \mid j = \delta(i, a) \} \mid \{ \epsilon \mid i = j \} \mid \emptyset$

inductive case $e_{ij}^{k+1} := e_{ij}^k \mid e_{i\,k+1}^k \,(e_{k+1\,k+1}^k)^\star\, e_{k+1\,j}^k$



iClicker 33.1

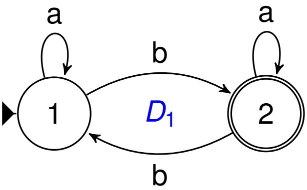
What is e_{12}^0 ?

A: a **B:** b **C:** ϵ **D:** $\emptyset$

Goal : $\mathcal{L}(e^k_{ij}) = \{ w \mid i \xrightarrow[w]{\star} j;\ \text{no intermediate state } \# > k \}$

base case $e^0_{ij} := \{ a \mid j = \delta(i,a) \} \mid \{ \epsilon \mid i = j \} \mid \emptyset$

inductive case $e^{k+1}_{ij} := e^k_{ij} \mid e^k_{i\,k+1}\ (e^k_{k+1\,k+1})^\star\ e^k_{k+1\,j}$

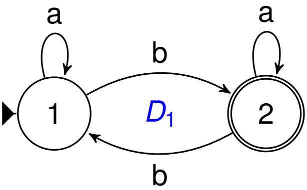


$$e^0 = \begin{pmatrix} a|\epsilon & b \\ b & a|\epsilon \end{pmatrix}$$

Goal : $\mathcal{L}(e_{ij}^k) = \{ w \mid i \xrightarrow[w]{\star} j;$ no intermediate state $\# > k \}$

base case $e_{ij}^0 := \{ a \mid j = \delta(i,a) \} \mid \{ \epsilon \mid i = j \} \mid \emptyset$

inductive case $e_{ij}^{k+1} := e_{ij}^k \mid e_{i\,k+1}^k \, (e_{k+1\,k+1}^k)^\star \, e_{k+1\,j}^k$



$$e^0 = \begin{pmatrix} a|\epsilon & b \\ b & a|\epsilon \end{pmatrix}$$

iClicker 33.2

What is e_{12}^1 ?

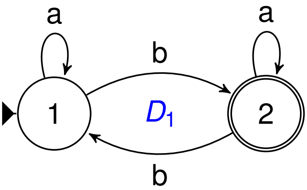
A: a **B:** b

C: $a^* b$ **D:** $a^* b a^*$

Goal : $\mathcal{L}(e^k_{ij}) = \{w \mid i \overset{\star}{\underset{w}{\to}} j;\ \text{no intermediate state } \# > k\}$

base case $e^0_{ij} := \{a \mid j = \delta(i,a)\} \mid \{\epsilon \mid i = j\} \mid \emptyset$

inductive case $e^{k+1}_{ij} := e^k_{ij} \mid e^k_{i\,k+1}\,(e^k_{k+1\,k+1})^\star\, e^k_{k+1\,j}$



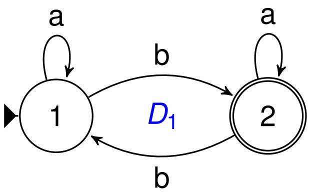
$$e^0 = \begin{pmatrix} a|\epsilon & b \\ b & a|\epsilon \end{pmatrix}$$

$$e^1 = \begin{pmatrix} a^* & a^*b \\ ba^* & a|\epsilon|ba^*b \end{pmatrix}$$

Goal : $\mathcal{L}(e^k_{ij}) \;=\; \{ w \mid i \overset{\star}{\underset{w}{\to}} j;\ \text{no intermediate state} \# > k \}$

base case $e^0_{ij} \;:=\; \{ a \mid j = \delta(i,a) \} \mid \{ \epsilon \mid i = j \} \mid \emptyset$

inductive case $e^{k+1}_{ij} \;:=\; e^k_{ij} \mid e^k_{i\,k+1}\,(e^k_{k+1\,k+1})^\star\, e^k_{k+1\,j}$



$$e^0 = \begin{pmatrix} a|\epsilon & b \\ b & a|\epsilon \end{pmatrix}$$

$$e^1 = \begin{pmatrix} a^* & a^*b \\ ba^* & a|\epsilon|ba^*b \end{pmatrix}$$

iClicker 33.3

What is e^2_{12} ?

A: a^*ba^*

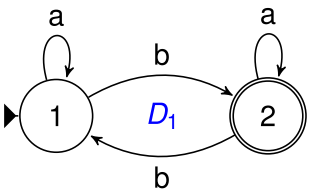
B: $a^*b(a|ba^*b)^*$

C: $a^*b|(a^*ba^*ba^*)^*$

Goal : $\mathcal{L}(e_{ij}^k) = \{w \mid i \xrightarrow[w]{\star} j;\ \text{no intermediate state } \# > k\}$

base case $e_{ij}^0 := \{a \mid j = \delta(i,a)\} \mid \{\epsilon \mid i = j\} \mid \emptyset$

inductive case $e_{ij}^{k+1} := e_{ij}^k \mid e_{i\,k+1}^k\ (e_{k+1\,k+1}^k)^\star\ e_{k+1\,j}^k$



$$e^0 = \begin{pmatrix} a|\epsilon & b \\ b & a|\epsilon \end{pmatrix}$$

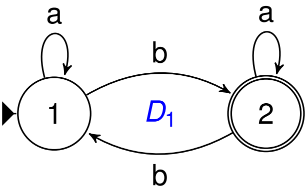
$$e^1 = \begin{pmatrix} a^* & a^*b \\ ba^* & a|\epsilon|ba^*b \end{pmatrix}$$

$$e^2 = \begin{pmatrix} (a|ba^*b)^* & a^*b(a|ba^*b)^* \\ a^*b(a|ba^*b)^* & (a|ba^*b)^* \end{pmatrix}$$

Goal : $\mathcal{L}(e_{ij}^{k}) = \{w \mid i \xrightarrow[w]{\star} j;\ \text{no intermediate state } \# > k\}$

base case $e_{ij}^{0} := \{a \mid j = \delta(i,a)\} \mid \{\epsilon \mid i = j\} \mid \emptyset$

inductive case $e_{ij}^{k+1} := e_{ij}^{k} \mid e_{i\,k+1}^{k}\ (e_{k+1\,k+1}^{k})^{\star}\ e_{k+1\,j}^{k}$



$$e^0 = \begin{pmatrix} a|\epsilon & b \\ b & a|\epsilon \end{pmatrix}$$

$$e^1 = \begin{pmatrix} a^* & a^*b \\ ba^* & a|\epsilon|ba^*b \end{pmatrix}$$

$$e^2 = \begin{pmatrix} (a|ba^*b)^* & a^*b(a|ba^*b)^* \\ a^*b(a|ba^*b)^* & (a|ba^*b)^* \end{pmatrix}$$

$$\mathcal{L}(D_1) = \mathcal{L}(e_{12}^2) = \mathcal{L}(a^*b(a|ba^*b)^*)$$

Kleene's Theorem Let $A \subseteq \Sigma^\star$ be any language. Then the following are equivalent:

1. $A = \mathcal{L}(D)$, for some DFA D.
2. $A = \mathcal{L}(N)$, for some NFA N wo ϵ transitions.
3. $A = \mathcal{L}(N)$, for some NFA N.
4. $A = \mathcal{L}(e)$, for some regular expression e.
5. A is regular.

We have completed the proof of Kleene's Theorem! $\square$