計算理論　第２回

1. 有限オートマトン
   - 決定性有限オートマトン（DFA）
   - 非決定性有限オートマトン(NFA)
   - DFA と NFA の等価性，サブセット構成
   - 応用：テキスト検索
   - ϵ-NFA と NFA の等価性
2. 正則言語の性質
3. 文脈自由文法と言語
4. プッシュダウン・オートマトン
5. 文脈自由言語の性質
6. チューリングマシン

テキスト
2.3～2.5節

本日の学習目標

- 非決定性有限オートマトン，非決定動作，ϵ-動作を，例を用いて説明できる
- 非決定性有限オートマトンと決定性有限オートマトンの等価性を，その理由を示して説明できる
- ϵ-動作の有無が，非決定性オートマトンの言語のクラスに影響しないことを，その理由を示して説明できる
- 非決定性有限オートマトンが与えられたとき，その言語を説明できる
- 指定された言語を受理する非決定性有限オートマトンを設計でき，状態遷移図／状態遷移表で表現できる
- 決定性オートマトン，非決定性オートマトンそれぞれの他方に比べての利点を、例を用いて説明できる

2.3 非決定性有限オートマトン

- 非決定性有限オートマトン（NFA）　$A = (Q, \Sigma, \delta, q_0, F)$

  Q：状態の有限集合（$Q \neq \emptyset$）

  Σ：入力記号の有限集合（$\Sigma \neq \emptyset$）
  （入力テープ上の記号）

  δ：状態遷移関数　$Q \times \Sigma \to 2^Q$
  （2^Q：Q のべき集合（部分集合の集合））

  q_0：開始状態　$q_0 \in Q$

  F：受理状態（最終状態）の集合　$F \subseteq Q$

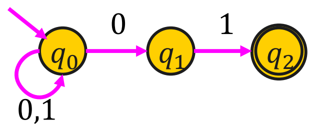
次状態が一意に定まらない
・複数の可能性
・ない場合（空集合）もある

開始状態 q_0 から列 w を読んで到達する状態の集合に
受理状態が一つでも含まれていると w を受理

Non-deterministic Finite Automaton

例2.6 非決定性有限オートマトンの例(1)

- 末尾が 01 の列を受理する NFA
  - 01，001，101，0001，0101，1001，1101，...



NFA　$A = (\{q_0, q_1, q_2\}, \{0, 1\}, \delta, q_0, \{q_2\})$

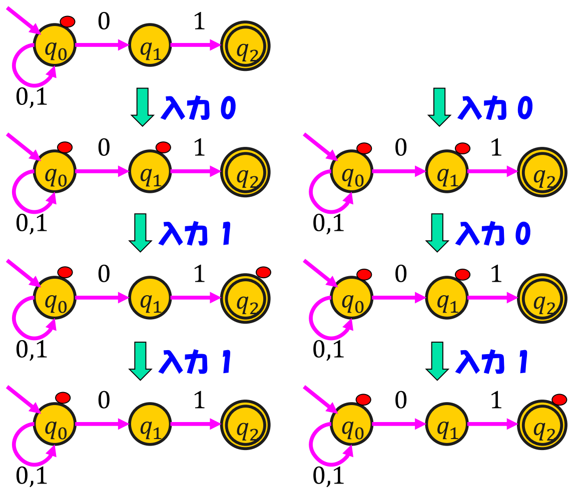
$\delta(q_0, 0) = \{q_0, q_1\}$
$\delta(q_0, 1) = \{q_0\}$
$\delta(q_1, 0) = \emptyset$
$\delta(q_1, 1) = \{q_2\}$
$\delta(q_2, 0) = \emptyset$
$\delta(q_2, 1) = \emptyset$

	0	1
$\to q_0$	$\{q_0, q_1\}$	$\{q_0\}$
q_1	$\emptyset$	$\{q_2\}$
$* q_2$	$\emptyset$	$\emptyset$

次状態が一意に定まらない
・複数の可能性
・ない場合（空集合）もある

例2.6 非決定性有限オートマトンの例(2)

- 末尾が 01 の列を受理する NFA の動作



実行例

列	状態
0	q_0, q_1
01	$q_0, \boldsymbol{q_2}$
011	q_0
0110	q_0, q_1
01100	q_0, q_1
011001	$q_0, \boldsymbol{q_2}$

2.3.3 遷移関数の拡張

- NFA $A = (Q, \Sigma, \delta, q_0, F)$
  - $\hat{\delta}$:状態遷移関数 δ の拡張
    - 長さ 0 以上の記号列を読んだときの状態遷移
    - $\hat{\delta}\ : Q \times \Sigma^* \to 2^Q$
  - 基礎：各 $q \in Q$ に対して，$\hat{\delta}(q, \epsilon) = \{q\}$
  - 再帰：各 $q \in Q, w = xa \in \Sigma^+ \ (x \in \Sigma^*, a \in \Sigma)$ に対して，

$$\hat{\delta}(q, w) = \{r_1, r_2, \ldots, r_m\}$$

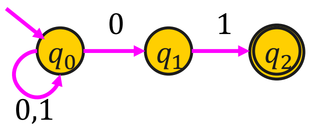
ここで，$\hat{\delta}(q, x) = \{p_1, p_2, \ldots, p_k\}$

$$\bigcup_{i=1}^{k} \delta(p_i, a) = \{r_1, r_2, \ldots, r_m\}$$

- $w \in \Sigma^*$ に対し，$\hat{\delta}(q_0, w) \cap F \neq \emptyset$ なら w を受理
  - q_0 から w を読んで到達する状態の集合に一つでも受理状態が含まれていると w を受理

例2.8 遷移関数の拡張

- 末尾が 01 の列を受理する NFA
  - 01, 001, 101, 0001, 0101, 1001, 1101, . . .



$\hat{\delta}(q_0, \epsilon) = \{q_0\}$
$\hat{\delta}(q_0, 0) = \{q_0, q_1\}$
$\hat{\delta}(q_0, 01) = \{q_0, \boldsymbol{q_2}\}$
$\hat{\delta}(q_0, 011) = \{q_0\}$
$\hat{\delta}(q_0, 0110) = \{q_0, q_1\}$
$\hat{\delta}(q_0, 01100) = \{q_0, q_1\}$
$\hat{\delta}(q_0, 011001) = \{q_0, \boldsymbol{q_2}\}$

実行例

列	状態
0	q_0, q_1
01	$q_0, \boldsymbol{q_2}$
011	q_0
0110	q_0, q_1
01100	q_0, q_1
011001	$q_0, \boldsymbol{q_2}$

2.3.4 NFAの言語

- NFA $A = (Q, \Sigma, \delta, q_0, F)$ の言語 $L(A)$
  - $L(A) = \{w \in \Sigma^* \mid \hat{\delta}(q_0, w) \cap F \neq \emptyset\}$
    - q_0 から w を読んで到達する状態の集合に一つでも受理状態が含まれていると w を受理

2.3.5 DFAとNFAの等価性

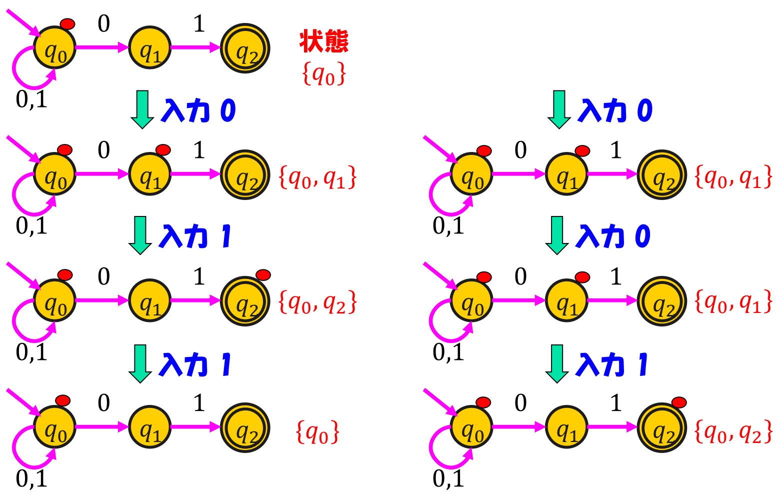
- $\mathcal{D}$：DFA のクラス（DFA すべての集合）
- $\mathcal{N}$：NFA のクラス（NFA すべての集合）
  - $\mathcal{D} \subset \mathcal{N}$　（定義から明らか）
    - 同じ言語を認識するオートマトン
      - **NFA は DFA より簡潔になる可能性がある**
- $\boldsymbol{L}(\mathcal{D})$:DFA の言語のクラス（DFA の言語すべての集合）
- $\boldsymbol{L}(\mathcal{N})$:NFA の言語のクラス（NFA の言語すべての集合）
  - $\boldsymbol{L}(\mathcal{D}) = \boldsymbol{L}(\mathcal{N})$
    - **非決定性は有限オートマトンの受理能力には影響を与えない**

DFAとNFAの等価性の示し方

- $\boldsymbol{L}(\mathcal{D}) = \boldsymbol{L}(\mathcal{N})$
  - $\boldsymbol{L}(\mathcal{D}) \subseteq \boldsymbol{L}(\mathcal{N})$　定義から明らか
  - $\boldsymbol{L}(\mathcal{D}) \supseteq \boldsymbol{L}(\mathcal{N})$ を示せばよい
- **サブセット構成**
  - NFA $N = (Q_N, \Sigma, \delta_N, q_0, F_N)$ が与えられたとき
    - $L(N) = L(D)$ を満たす DFA $D = (Q_D, \Sigma, \delta_D, \{q_0\}, F_D)$ を構成する方法
    - $\boldsymbol{L}(\mathcal{D}) \supseteq \boldsymbol{L}(\mathcal{N})$ を示せる
  - DFA D の状態を NFA N の状態の集合で表す
    - $Q_D \subseteq 2^{Q_N}$ が成り立つ

サブセット構成法のアイデア

- 例2.6：末尾が 01 の列を受理する NFA の動作



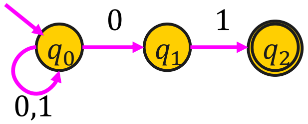
サブセット構成法

- NFA $N = (Q_N, \Sigma, \delta_N, q_0, F_N)$ が与えられたとき
  - $L(N) = L(D)$ を満たす DFA $D = (Q_D, \Sigma, \delta_D, \{q_0\}, F_D)$ を構成する方法

1. $Q_D = 2^{Q_N}$ とする
   （Q_D は Q_N の部分集合すべての集合）
2. $F_D = \{S \subseteq Q_N \mid S \cap F_N \neq \emptyset\}$ とする
   （F_D は Q_N の部分集合のうち，受理状態を含むものすべての集合）
3. 各 $S \subseteq Q_N$，各 $a \in \Sigma$ に対し，
   $\delta_D(S, a) = \bigcup_{p \in S} \delta_N(p, a)$ とする

例2.10 サブセット構成(1)

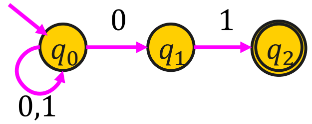
- 末尾が 01 の列を受理する NFA



- 末尾が 01 の列を受理する DFA
  $D=(Q_D,\Sigma,\delta_D,\{q_0\},F_D)$ を構成する
  1. $Q_D=2^{Q_N}$ とする
     - $Q_D=\{\emptyset,\{q_0\},\{q_1\},\{q_2\},\{q_0,q_1\},\{q_0,q_2\},\{q_1,q_2\},\{q_0,q_1,q_2\}\}$
  2. $F_D=\{S\subseteq Q_N \mid S\cap F_N\neq\emptyset\}$ とする
     - $F_D=\{\{\boldsymbol{q_2}\},\{q_0,\boldsymbol{q_2}\},\{q_1,\boldsymbol{q_2}\},\{q_0,q_1,\boldsymbol{q_2}\}\}$
  3. 各 $S\subseteq Q_N$, 各 $a\in\Sigma$ に対し,
     $\delta_D(S,a)=\bigcup_{p\in S}\delta_N(p,a)$ とする

例2.10 サブセット構成(2)

- 末尾が 01 の列を受理する NFA



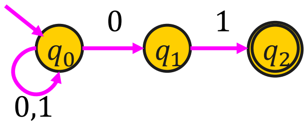
- 末尾が 01 の列を受理する DFA
  $D=(Q_D,\Sigma,\delta_D,\{q_0\},F_D)$ を構成する
  3. 各 $S\subseteq Q_N$, 各 $a\in\Sigma$ に対し,
     $\delta_D(S,a)=\bigcup_{p\in S}\delta\_N(p,a)$ とする

アイデア
非決定性の動作を
すべて同時に
シミュレートしている

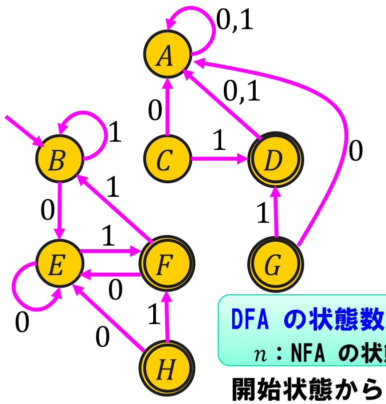
	0	1
$\emptyset$	$\emptyset$	$\emptyset$
$\rightarrow\{q_0\}$	$\{q_0,q_1\}$	$\{q_0\}$
$\{q_1\}$	$\emptyset$	$\{q_2\}$
$*\{q_2\}$	$\emptyset$	$\emptyset$
$\{q_0,q_1\}$	$\{q_0,q_1\}$	$\{q_0,q_2\}$
$*\{q_0,q_2\}$	$\{q_0,q_1\}$	$\{q_0\}$
$*\{q_1,q_2\}$	$\emptyset$	$\{q_2\}$
$*\{q_0,q_1,q_2\}$	$\{q_0,q_1\}$	$\{q_0,q_2\}$

例2.10 サブセット構成(3)

- 末尾が 01 の列を受理する NFA



- 末尾が 01 の列を受理する DFA



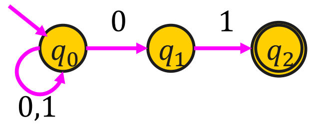
		0	1
A	$\emptyset$	$\emptyset$	$\emptyset$
B	$\rightarrow\{q_0\}$	$\{q_0,q_1\}$	$\{q_0\}$
C	$\{q_1\}$	$\emptyset$	$\{q_2\}$
D	$*\{q_2\}$	$\emptyset$	$\emptyset$
E	$\{q_0,q_1\}$	$\{q_0,q_1\}$	$\{q_0,q_2\}$
F	$*\{q_0,q_2\}$	$\{q_0,q_1\}$	$\{q_0\}$
G	$*\{q_1,q_2\}$	$\emptyset$	$\{q_2\}$
H	$*\{q_0,q_1,q_2\}$	$\{q_0,q_1\}$	$\{q_0,q_2\}$

DFA の状態数$=2^n$
n : NFA の状態数

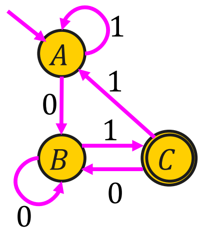
開始状態から到達できない状態も含まれる

例2.10 サブセット構成(4)

- 末尾が 01 の列を受理する NFA



- 末尾が 01 の列を受理する DFA
  - 開始状態から**到達可能な状態のみ**を順次作る
    - DFA 構成の手間を省けることがある



		0	1
A	$\rightarrow\{q_0\}$	$\{q_0,q_1\}$	$\{q_0\}$
B	$\{q_0,q_1\}$	$\{q_0,q_1\}$	$\{q_0,q_2\}$
C	$*\{q_0,q_2\}$	$\{q_0,q_1\}$	$\{q_0\}$

定理2.11, 定理2.12

テキストの証明は読んでおくこと

【定理2.11】

NFA $N=(Q_N,\Sigma,\delta_N,q_0,F_N)$ からサブセット構成によって
DFA $D=(Q_D,\Sigma,\delta_D,\{q_0\},F_D)$ が作られたとき,

$$L(D)=L(N)$$

が成り立つ

【定理2.12】

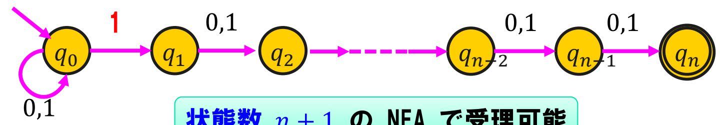
言語 L がある DFA で受理されるための必要十分条件は
L がある NFA で受理されることである

*DFA と NFA の受理能力は等価である

2.3.6 サブセット構成で状態数が増える場合

- NFA $N=(Q_N,\Sigma,\delta_N,q_0,F_N)$ に対し, $L(D)=L(N)$ を満たす状態数最小の DFA $D=(Q_D,\Sigma,\delta_D,\{q_0\},F_D)$ でも, DFAの状態数が NFA の状態数に比べ指数的に増えることがある

- 例2.13：最後から n 番目の記号が 1 である語すべて



状態数 $n+1$ の NFA で受理可能

DFA では直前の n 個の入力を覚えておく必要があり
状態数は 2^n 以上
（テキストの証明を読んでおくこと）

2.4 応用：テキスト検索

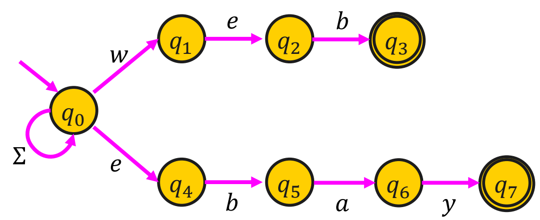
- テキストから特定のキーワードを検索する
  - 「データ構造とアルゴリズム」で学習した文字列照合問題
    - 1つのキーワードの検索
    - 力まかせ法, ラビン-カープ法, クヌース-モリス-プラッツ法, ボイヤー-ムーア法

- 複数のキーワードの検索に有限オートマトンを利用
  - $O(m)$ 時間で検索（m：テキスト長）
  - 検索キーワードに応じた前処理（DFAの構成）が必要

テキスト検索：NFA による方法

- 複数のキーワードの検索に NFA を利用
  - キーワードを発見すれば受理状態
  - 簡単に構成できる

- 例：web, ebay の検索



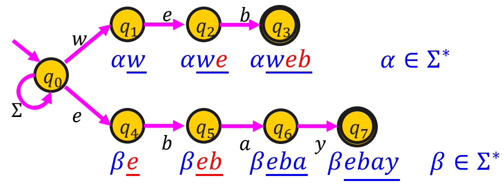
web, ebay を受理する NFA

テキスト検索：DFA による方法(1)

- 複数のキーワードの検索に DFA を利用
  - キーワードを発見すれば受理状態
  - NFA をサブセット構成で DFA に変換
    - （開始状態から到達可能な）状態数は増加しない
    - NFA：開始状態 q_0 から記号列 $w = a_1 a_2 \dots a_m$ で到達する状態 q
    - DFA：q に対応する状態 A_q
      - $q \in A_q$
      - $q_0 \in A_q$
      - q_0 から w の接尾辞（$a_i a_{i+1} \dots a_m$ という形の部分列）で到達する状態 $q' \in A_q$

テキスト検索：DFA による方法(2)

- 複数のキーワードの検索に DFA を利用
  - キーワードを発見すれば受理状態
  - NFA をサブセット構成で DFA に変換
    - （開始状態から到達可能な）状態数は増加しない



状態の対応関係

NFA　DFA

$q_2 \leftrightarrow \{q_0, q_2, q_4\}$

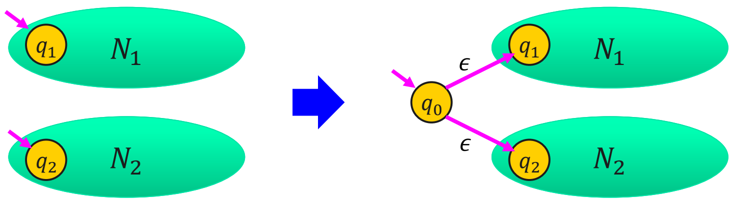
$q_3 \leftrightarrow \{q_0, q_3, q_5\}$

DFA の状態部分集合は共通接尾辞を持つ
NFA の最長の接尾辞を持つ状態に対応

構成される DFA は p.79 図2.17

2.5 ϵ-動作を含む有限オートマトン

- ϵ-動作
  - ϵ を読んで（何も読まずに）状態遷移する
  - 非決定性動作が発生
  - ϵ-動作を許すと，NFA を簡潔に表現できることがある
  - ϵ-動作は NFA の受理能力には影響を与えない
- 例：言語 $L = L_1 \cup L_2$ を受理する NFA N
  - N_1（L_1 を受理する NFA），N_2（L_2 を受理する NFA）から N を合成



2.5.2 ϵ-NFA の定義

- ϵ-動作を含む NFA（ϵ-NFA）　$A = (Q, \Sigma, \delta, q_0, F)$

  Q：状態の有限集合（$Q \neq \emptyset$）

  Σ：入力記号の有限集合（$\Sigma \neq \emptyset$）
  （入力テープ上の記号）

  ϵ-動作（ϵ-遷移）を許す

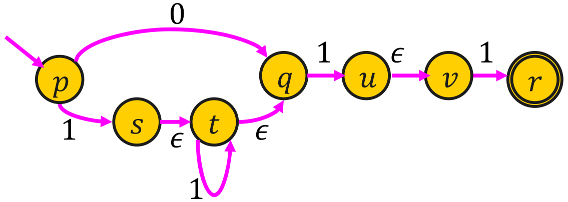
  δ：状態遷移関数 $Q \times (\Sigma \cup \{\epsilon\}) \to 2^Q$
  （2^Q：Q のべき集合（部分集合の集合））

  q_0：開始状態 $q_0 \in Q$

  F：受理状態（最終状態）の集合 $F \subseteq Q$

2.5.3 ϵ-閉包

- ϵ-動作は NFA の受理能力には影響を与えないことを示すための準備
- 状態 q の ϵ-閉包 ECLOSE(q)
  - q から ϵ-動作のみで到達可能な状態の集合（qを含む）



ECLOSE$(s) = \{s, t, q\}$，ECLOSE$(t) = \{t, q\}$，ECLOSE$(u) = \{u, v\}$
その他の状態 x については，ECLOSE$(x) = \{x\}$

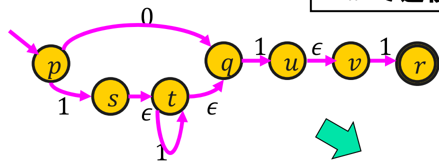
テキストとは少し異なる

2.5.5 ϵ-遷移の除去 (1)

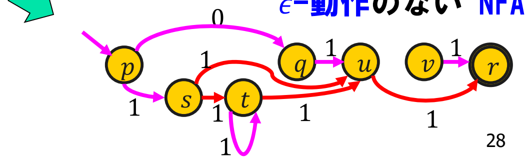
- ϵ-NFA $N = (Q, \Sigma, \delta_1, q_0, F_1)$ から，$L(N) = L(N')$ を満たす，ϵ-動作を含まない NFA $N' = (Q, \Sigma, \delta_2, q_0, F_2)$を構成する

1. 各状態 $x \in Q$，各入力記号 $a \in \Sigma$ に対し
   $\delta_2(x, a) = \{y \mid z \in$ECLOSE$(x) \wedge y \in \delta_1(z, a)\}$

y は，x から a で遷移可能，または，
x から ϵ 遷移（複数も可）の後に
a で遷移可能



ϵ-動作のない NFA



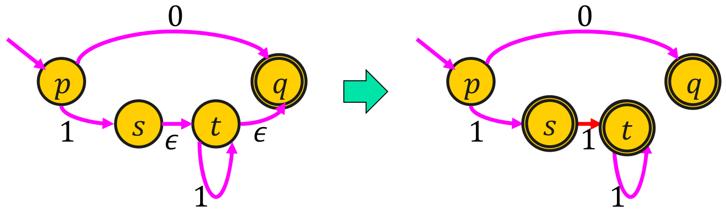
テキストとは少し異なる

2.5.5 ϵ-遷移の除去 (2)

- ϵ-NFA $N = (Q, \Sigma, \delta_1, q_0, F_1)$ から，$L(N) = L(N')$ を満たす，ϵ-動作を含まない NFA $N' = (Q, \Sigma, \delta_2, q_0, F_2)$を構成する

2. 受理状態集合 $F_2 = \{y \in Q \mid F_1 \cap$ECLOSE$(y) \neq \emptyset\}$

y から ϵ 遷移で受理状態に到達可能



テキストの手法

2.5.5 ϵ-遷移の除去 (1)

- ϵ-NFA $E = (Q_E, \Sigma, \delta_E, q_0, F_E)$ から，$L(E) = L(D)$ を満たす，ϵ-動作を含まない DFA $D = (Q_D, \Sigma, \delta_D, q_D, F_D)$を構成する

直前の手法では，より簡単な NFA の構成法を示した
NFA の DFA への変換法（サブセット構成）は既習なので

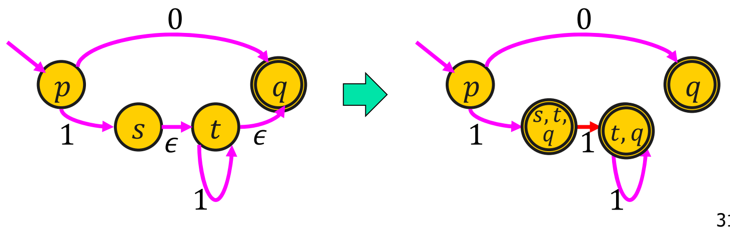
1. $Q_D = 2^{Q_E}$ （Q_D は Q_E のすべての部分集合の集合）
   - Q_D には不要な状態も含まれてしまう
     有用なのは，$S = \cup_{q \in S} ECLOSE(q)$ を満たす状態だけ
2. $q_D = ECLOSE(q_0)$
3. $F_D = \{S \in Q_D \mid S \cap F_E \neq \emptyset\}$

2.5.5 ϵ-遷移の除去 (2)

テキストの手法

- ϵ-NFA $E = (Q_E, \Sigma, \delta_E, q_0, F_E)$ から，$L(E) = L(D)$ を満たす，ϵ-動作を含まない DFA $D = (Q_D, \Sigma, \delta_D, q_D, F_D)$を構成する

4. $\boldsymbol{\delta_D(S, a) = \cup_{j=1}^{m} ECLOSE(r_j)}$
   - ここで，
     $S = \{p_1, p_2. \ldots, p_k\}$,
     $\cup_{i=1}^{k} \delta_E(p_i, a) = \{r_1, r_2, \ldots, r_m\}$



本日の講義のまとめ

1. 有限オートマトン
   - 決定性有限オートマトン（DFA）
   - 非決定性有限オートマトン（NFA）
   - DFA と NFA の等価性，サブセット構成
   - 応用：テキスト検索
   - ϵ-NFA と NFA の等価性
2. 正則言語の性質
3. 文脈自由文法と言語
4. プッシュダウン・オートマトン
5. 文脈自由言語の性質
6. チューリングマシン

テキスト
2.3～2.5節

本日の学習目標

目標を達成できたか
確認してみよう
（復習も含めて）

- 非決定性有限オートマトン，非決定動作，ϵ-動作を，例を用いて説明できる
- 非決定性有限オートマトンと決定性有限オートマトンの等価性を，その理由を示して説明できる
- ϵ-動作の有無が，非決定性オートマトンの言語のクラスに影響しないことを，その理由を示して説明できる
- 非決定性有限オートマトンが与えられたとき，その言語を説明できる
- 指定された言語を受理する非決定性有限オートマトンを設計でき，状態遷移図／状態遷移表で表現できる
- 決定性オートマトン，非決定性オートマトンそれぞれの他方に比べての利点を、例を用いて説明できる