

ソフトウェアモデル論(2013年度)  
第5回・2013/10/24

桑原 寛明  
情報理工学部 情報システム学科

**正規表現から有限オートマトンへの変換** (復習)

1. 正規表現  $\epsilon$ ,  $\emptyset$ ,  $a$  が表す言語を受理する有限オートマトンを作る
2. 正規表現  $(P+Q)$ ,  $(P \cdot Q)$ ,  $(P^*)$  が表す言語を受理する有限オートマトンは、 $P$  や  $Q$  が表す言語を受理する有限オートマトンを組合せて作る

ソフトウェアモデル論(2013/10/24) 2

**{ $\epsilon$ }** を受理する有限オートマトン (復習)

ソフトウェアモデル論(2013/10/24) 3

**$\emptyset$**  を受理する有限オートマトン (復習)

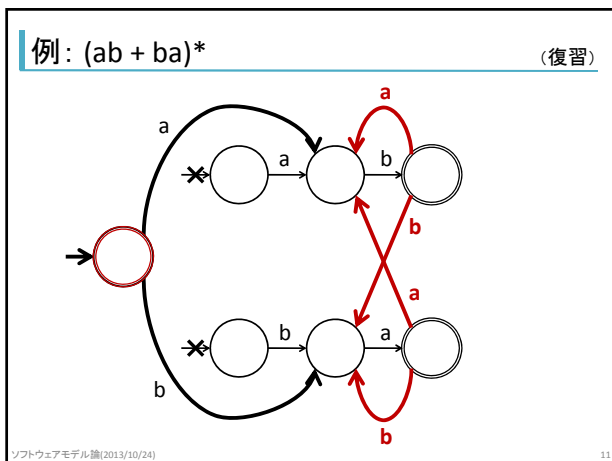
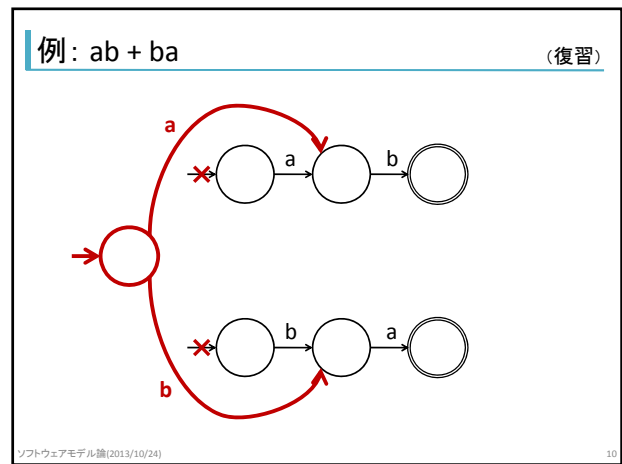
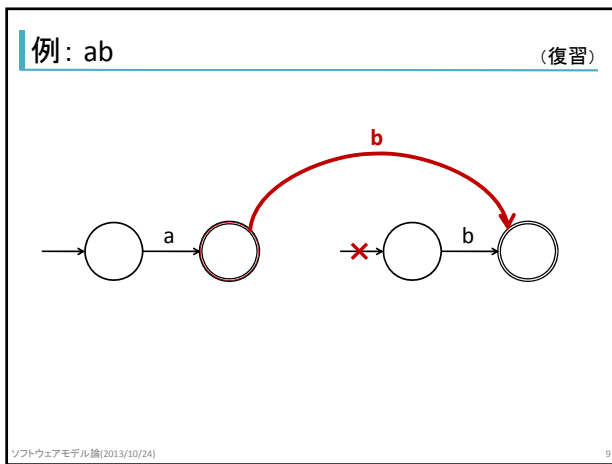
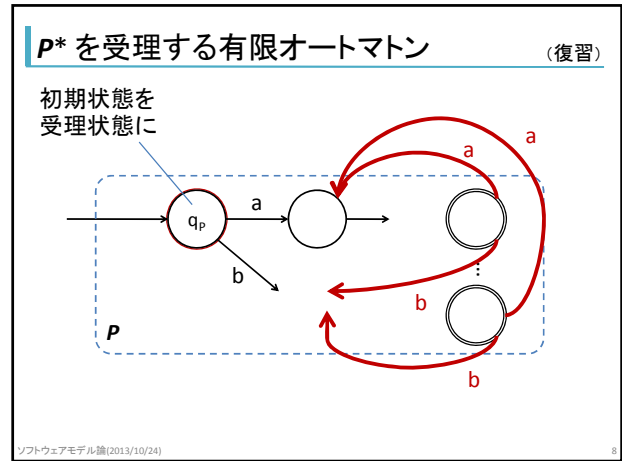
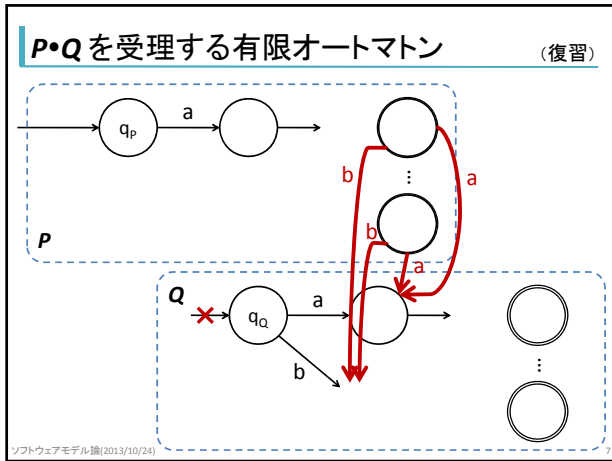
ソフトウェアモデル論(2013/10/24) 4

**{ $a$ }** を受理する有限オートマトン (復習)

ソフトウェアモデル論(2013/10/24) 5

**$P+Q$**  を受理する有限オートマトン (復習)

ソフトウェアモデル論(2013/10/24) 6



### 有限オートマトンから正規表現への変換 (復習)

- 有限オートマトン  $M = (\{1, \dots, n\}, \Sigma, \delta, 1, F)$
- $r_{ij}^k$  を求める
  - 状態  $i$  から状態  $j \rightarrow k$  以下の状態のみを通過して到達する記号列の正規表現
    - $i$  や  $j$  は  $k$  より大きくてもよい
  - $r_{ij}^0$  から順に帰納的に
- $r_{1f1}^n + \dots + r_{1f1}^1$  が初期状態から受理状態へ到達する記号列の正規表現
  - $\{f1, \dots, f1\}$  は受理状態の集合

ソフトウェアモデル論(2013/10/24) 12

### $r_{ij}^0$ を求める (復習)

- 状態  $i$  から状態  $j$  へ直接到達
- $\delta(i,a) = j$  を満たす  $a$  の集合を  $\{a_1, \dots, a_l\}$  とする  
 - そのような  $a$  がなければ  $\emptyset$
- $i \neq j$  ならば  $r_{ij}^0 = a_1 + \dots + a_l$
- $i = j$  ならば  $r_{ij}^0 = a_1 + \dots + a_l + \epsilon$

ソフトウェアモデル論(2013/10/24) 13

### $r_{ij}^k$ を求める (復習)

- $r_{ij}^k = r_{ik}^{k-1} (r_{kk}^{k-1})^* r_{kj}^{k-1} + r_{ij}^{k-1}$
- 状態  $k$  を通過するかしないかの2択
  - 通過する
    1.  $i$  から  $k-1$  以下を通過して初めて  $k$  に到達
    2.  $k$  から  $k-1$  以下を通過してまた  $k$  に到達(を繰り返す)
    3.  $k$  から  $k-1$  以下を通過して  $j$  に到達
  - 通過しない
    - $i$  から  $k-1$  以下の状態のみ通過して  $j$  に到達

ソフトウェアモデル論(2013/10/24) 14

### $r_{ij}^k$ を求める

ソフトウェアモデル論(2013/10/24) 15

### 例

$\begin{aligned} r_{12}^1 &= r_{11}^0(r_{11}^0)^*r_{12}^0 + r_{12}^0 \\ &= \epsilon\epsilon^*a + a \\ &= a \end{aligned}$	$\begin{aligned} r_{23}^1 &= r_{21}^0(r_{11}^0)^*r_{13}^0 + r_{23}^0 \\ &= \emptyset\epsilon^*\emptyset + b \\ &= b \end{aligned}$	$\begin{aligned} r_{13}^2 &= r_{12}^1(r_{22}^1)^*r_{23}^1 + r_{13}^1 \\ &= a\epsilon^*b + \emptyset \\ &= ab \end{aligned}$
$\begin{aligned} r_{11}^{-1} &= r_{11}^0(r_{11}^0)^*r_{13}^0 + r_{13}^0 \\ &= \epsilon\epsilon^*a + \emptyset \\ &= \emptyset \end{aligned}$	$\begin{aligned} r_{32}^{-1} &= r_{31}^0(r_{11}^0)^*r_{12}^0 + r_{32}^0 \\ &= \emptyset\epsilon^*a + \emptyset \\ &= \emptyset \end{aligned}$	$\begin{aligned} r_{33}^1 &= r_{32}^1(r_{22}^1)^*r_{23}^1 + r_{33}^1 \\ &= \emptyset\epsilon^*b + \epsilon \\ &= \epsilon \end{aligned}$
$\begin{aligned} r_{22}^{-1} &= r_{21}^0(r_{11}^0)^*r_{12}^0 + r_{22}^0 \\ &= \emptyset\epsilon^*a + \epsilon \\ &= \epsilon \end{aligned}$	$\begin{aligned} r_{33}^1 &= r_{31}^0(r_{11}^0)^*r_{13}^0 + r_{33}^0 \\ &= \emptyset\epsilon^*\emptyset + \epsilon \\ &= \epsilon \end{aligned}$	$\begin{aligned} r_{13}^3 &= r_{12}^1(r_{22}^1)^*r_{23}^1 + r_{13}^2 \\ &= a\epsilon^*\epsilon + ab \\ &= ab \end{aligned}$

ソフトウェアモデル論(2013/10/24) 16

### 正規言語

- 正規文法が生成する言語
- 有限オートマトンが受理できる言語
- 正規表現で表現できる言語

ある言語が正規言語か否か判定するにはどうすればよいか

ソフトウェアモデル論(2013/10/24) 17

### 有限オートマトンが長い語を受理する場合

- 初期状態から受理状態に到達する間にループが存在する
  - 同じ状態を2度以上通過する 部屋割り原理  
鳩ノ巣原理
- 状態数が  $n$  で語の長さが  $n$  以上だと...

ソフトウェアモデル論(2013/10/24) 18

### 反復補題

- 正規言語  $L$  に対して  $n$  が存在して、 $|z| \geq n$  なる任意の  $z \in L$  について以下を満たすように  $z$  を  $uvw$  に分解できる
  - $|uv| \leq n$
  - $|v| \geq 1$
  - 0 以上の任意の  $i$  について  $uv^i w \in L$
- 条件を満たす分解が1つでもあればよい

ソフトウェアモデル論(2013/10/24)

19

### $L = \{ a^n b^n \mid n \text{ は } 0 \text{ 以上の整数} \}$ は非正規言語

- $L$  が正規言語と仮定する
- 適当な  $n$  に対して  $|a^n b^n| \geq n$  となる  $l$  を選ぶ
- $L$  は正規言語なので  $a^l b^l = uvw$  と分解できる
- さらに、 $uv^2 w \in L$
- $a^l b^l$  の分解方法は以下の3通り
  - $uvw = a^j a^i a^{k_1} b^{k_2}$  ( $j \geq 1, k_1 \geq 0, i + j + k_1 = k_2 = l$ )
  - $uvw = a^i a^{j_1} b^{j_2} b^k$  ( $j_1 \geq 1, j_2 \geq 1, i + j_1 = j_2 + k = l$ )
  - $uvw = a^{i_1} b^{i_2} b^k$  ( $j \geq 1, i_2 \geq 0, i_1 = i_2 + j + k = l$ )

ソフトウェアモデル論(2013/10/24)

20

### $L = \{ a^n b^n \mid n \text{ は } 0 \text{ 以上の整数} \}$ は非正規言語

- 1. の場合、 $uv^2 w = a^j a^{2i} a^{k_1} b^{k_2}$  であり、 $j \geq 1$  と  $i + j + k_1 = k_2$  より  $i + 2j + k_1 \neq k_2$  なので  $uv^2 w \notin L$
- 3. の場合も同様
- 2. の場合、 $uv^2 w = a^i a^{j_1} b^{j_2} a^{j_1} b^{j_2} b^k$  であり、明らかに  $L$  の語ではない
- 以上より、 $L$  は正規言語ではない
  - 反復補題の  $uv^i w \in L$  を満たさない

ソフトウェアモデル論(2013/10/24)

21

### 反復補題は必要条件

- 反復補題は正規言語の必要条件
  - 反復補題が成り立たなければ正規言語ではない
- 反復補題を満たす正規言語でない言語も存在する

ソフトウェアモデル論(2013/10/24)

22