2. 有限オートマトン(1) :
(テキスト2.1~2.3.4)

2.1. 直感的説明

- 有限オートマトン(DFA; Deterministic Finite Automata) とは「状態を持つ機械」のモデル
  - 例 : 船による運搬問題
    - 川の左岸に狼(W)、羊(G)、キャベツ(C)を持った運搬人(M)がいる。
    - MがいないとWはGを、GはCを食べてしまう。
    - 船にはM以外には高々1つしか乗せられない。
    - 川の右岸に運搬する方法を求めよ。

2. Finite Automaton (1)：(Text 2.1~2.3.4)

2.1. Preliminaries

- Deterministic Finite Automata (DFA) is a formal model of ‘machines with finite states.’
  - Ex：Problem for a transporter
    - On the left side of a river, a transporter M has wolf W, goat G, and cabbage C.
    - If M is not present, W eats G, and G eats C.
    - On a boat, M can only carry one of W, G, and C.
    - How can M carry all of them to the right side of the river?

2.1. 直感的説明

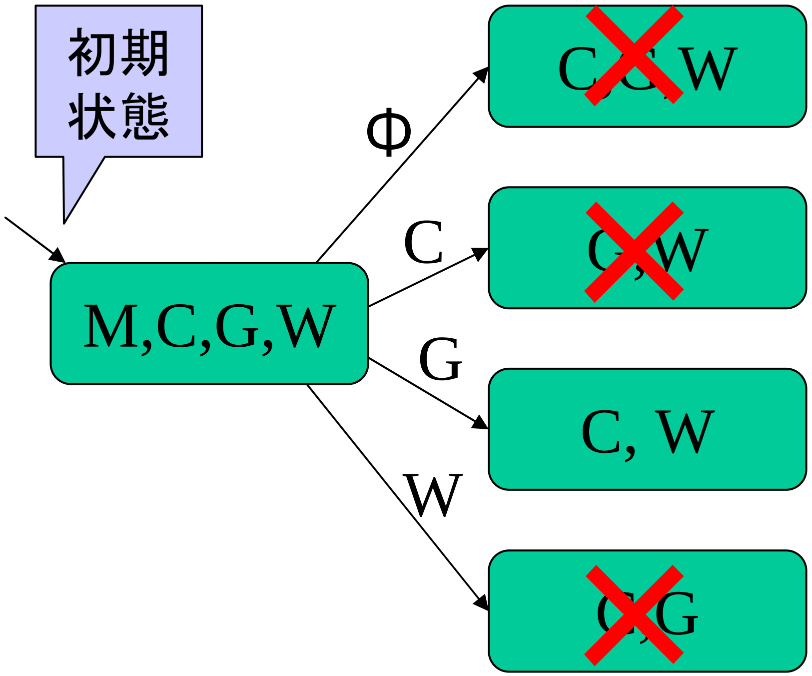
- DFA＝「状態を持つ機械」
  - 船による運搬問題
    - 状態: 左岸にいるものの集合
    - 入力: 船で人間が運ぶもの
      - 初期状態は{M,C,G,W}, 受理状態は{Φ}

2.1. Preliminaries

- DFA＝ ‘A machine with some states’
  - Problem for a transporter
    - States: the set of objects on the left side of the river
    - Inputs: the thing what M carries by the boat
      - Initial state is {M,C,G,W}, Final state is {Φ}

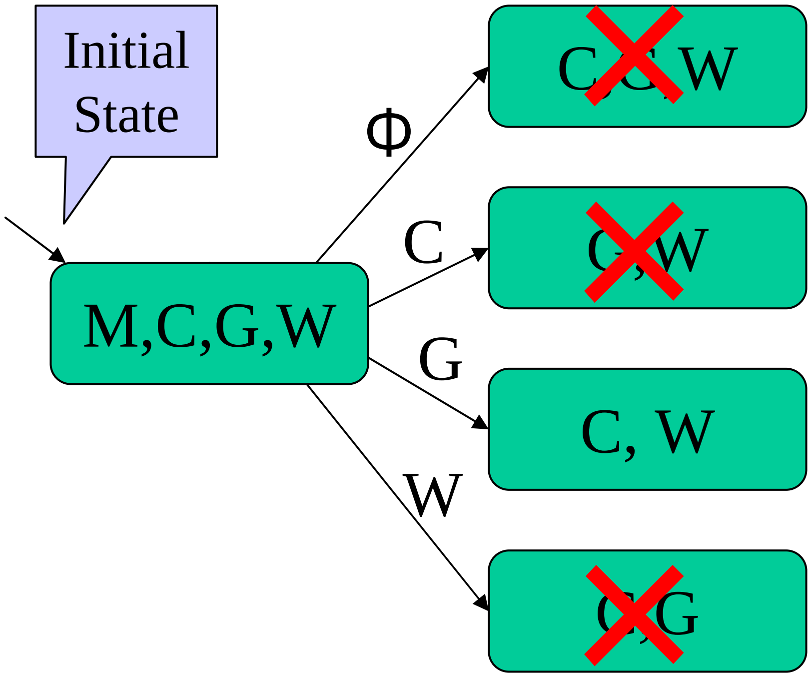
2.1. 直感的説明

- 船による運搬問題の状態遷移図



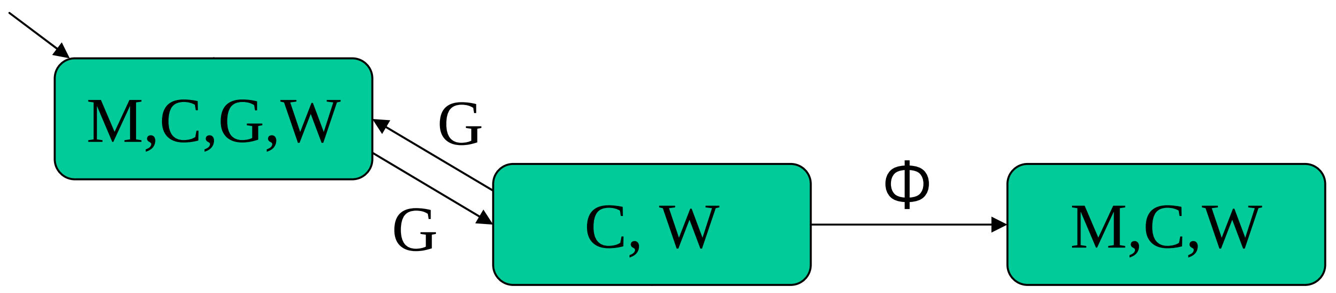
2.1. Preliminaries

- A state transition diagram of the problem for a transporter



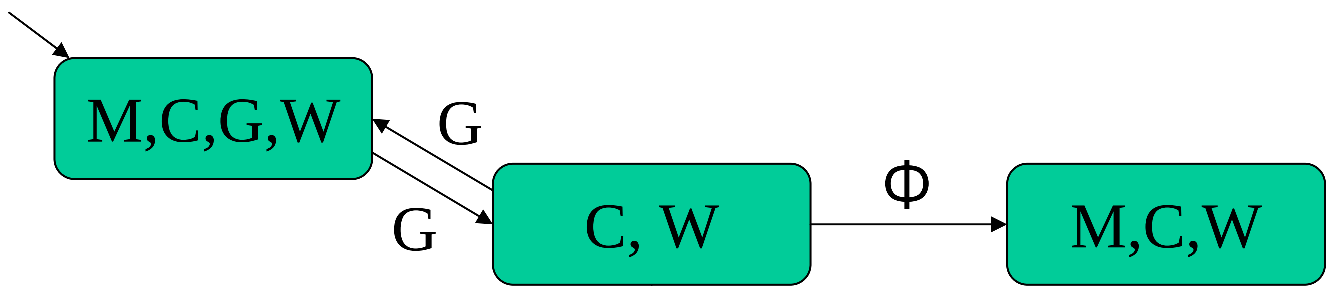
2.1. 直感的説明

- 船による運搬問題の状態遷移図



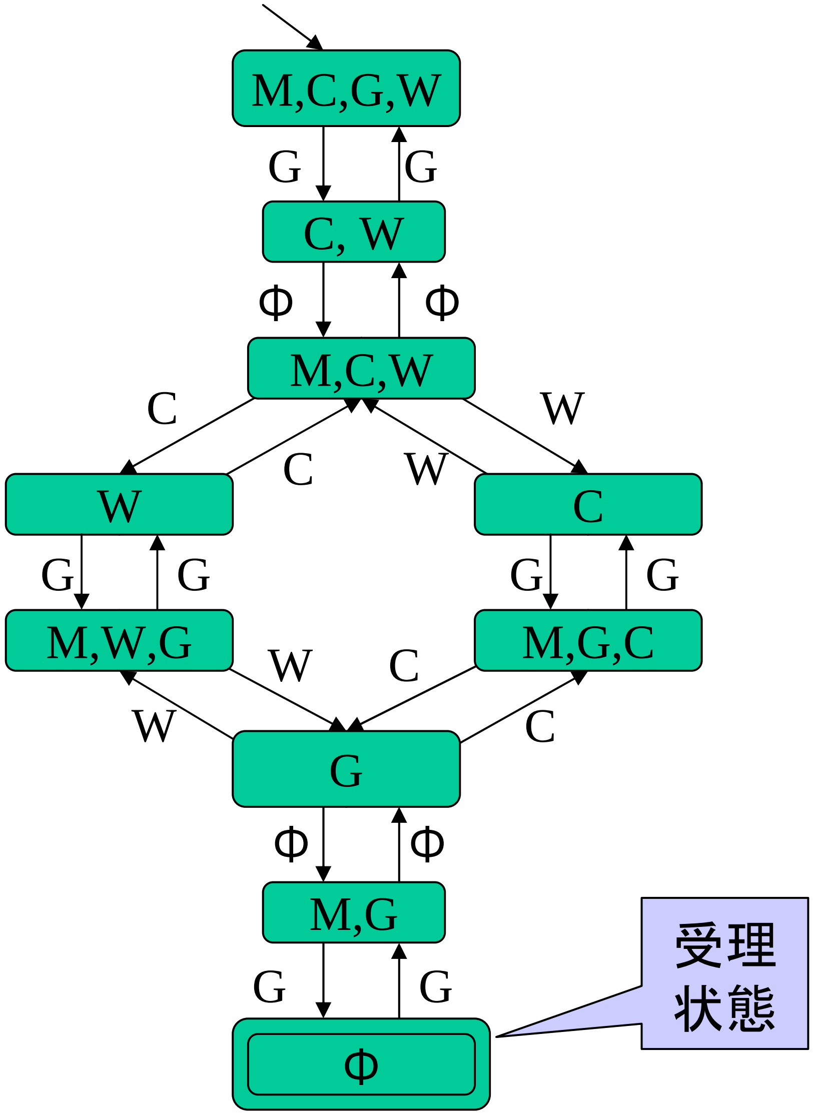
2.1. Preliminaries

- A state transition diagram of the problem for a transporter



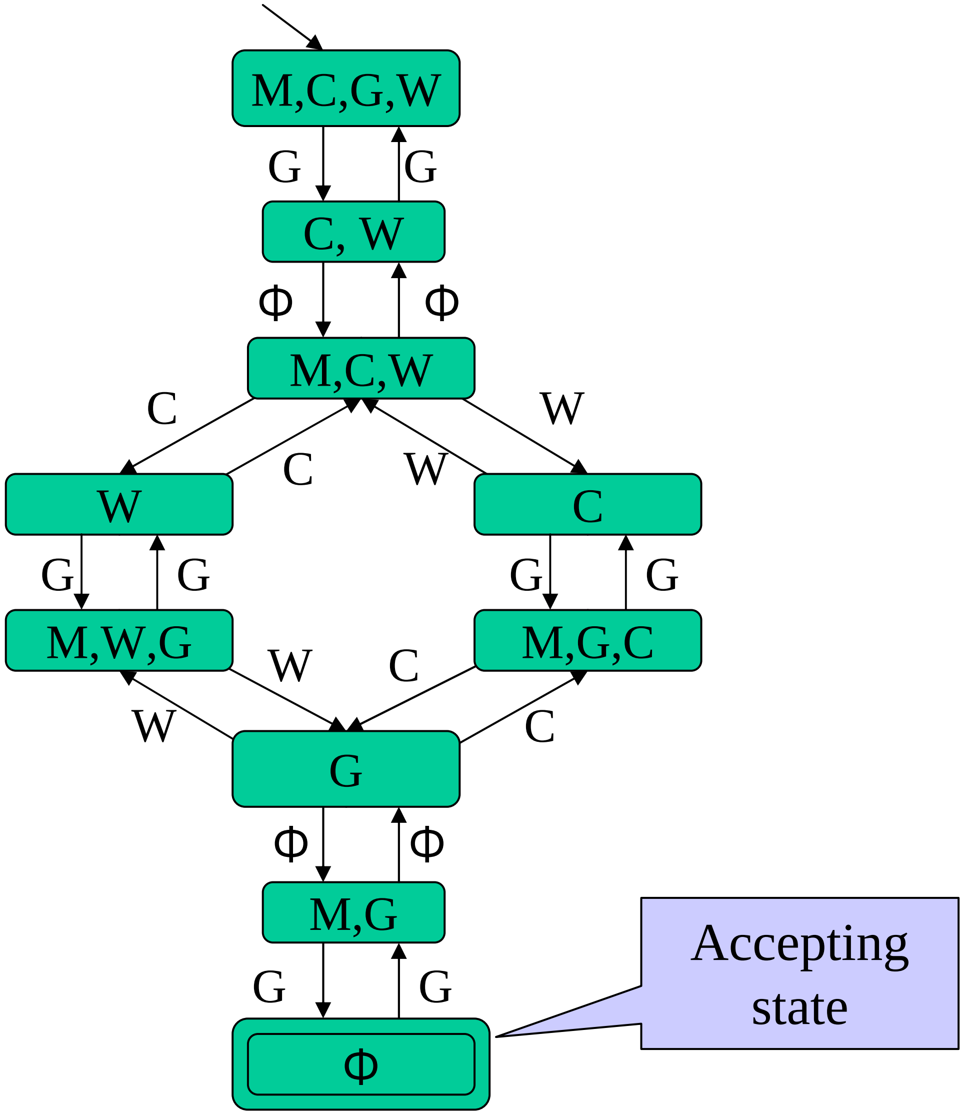
2.1. 直感的説明

- 船による運搬問題の状態遷移図



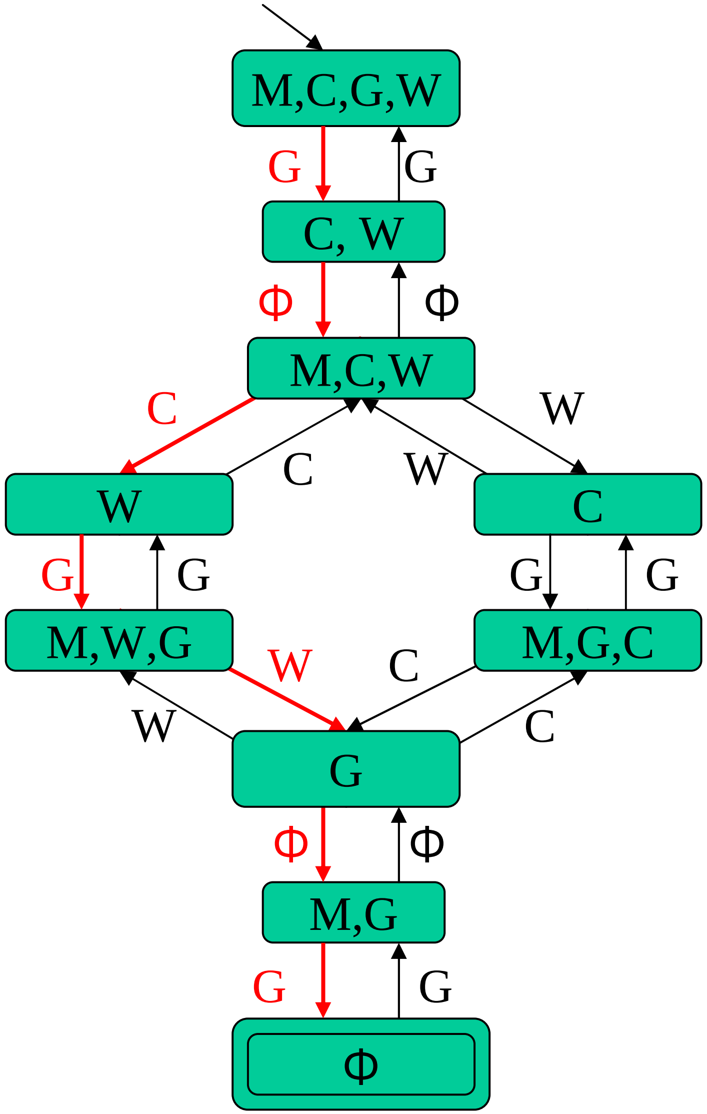
2.1. Preliminaries

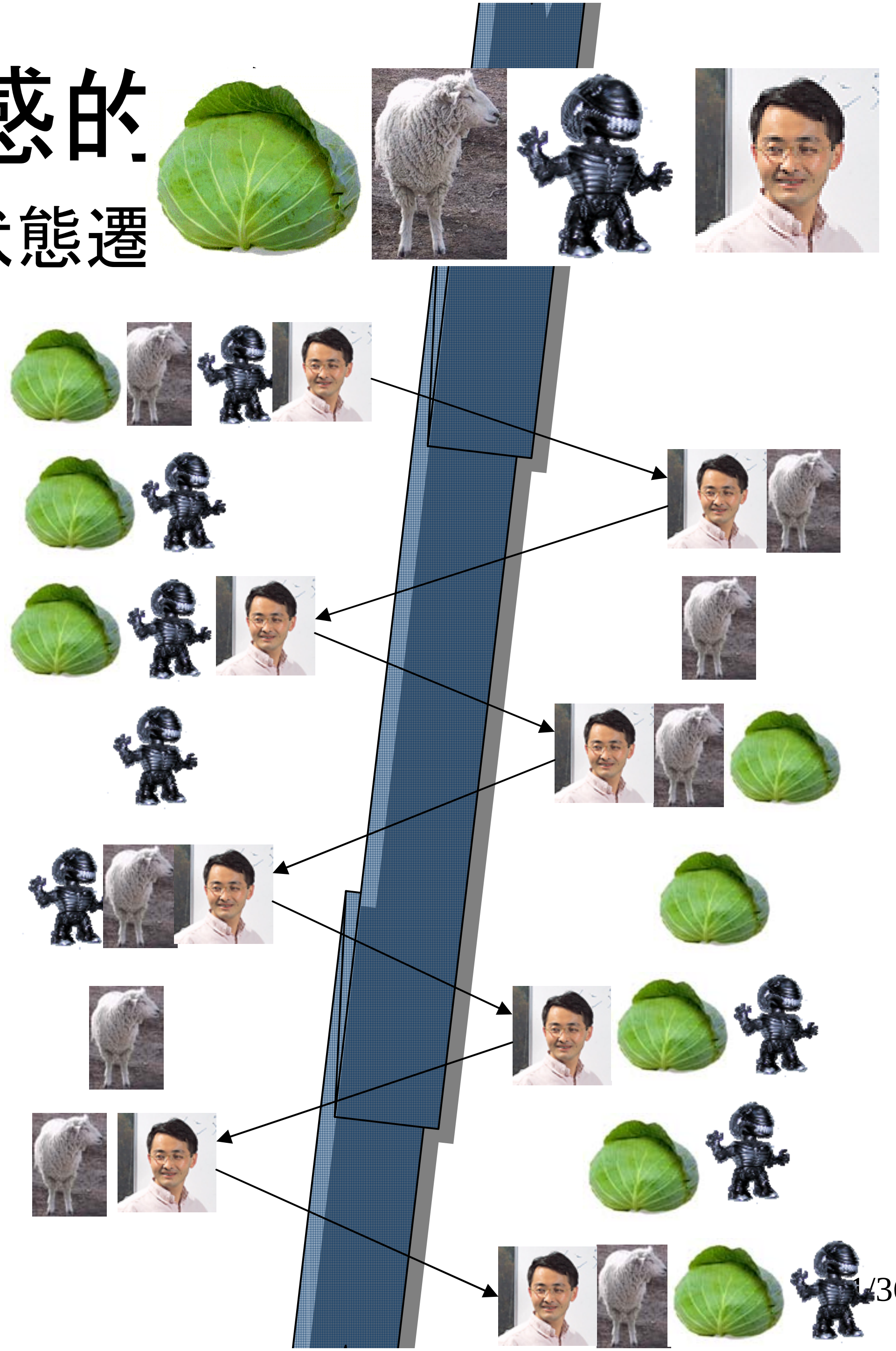
- A state transition diagram of the problem for a transporter



2.1. 直感的

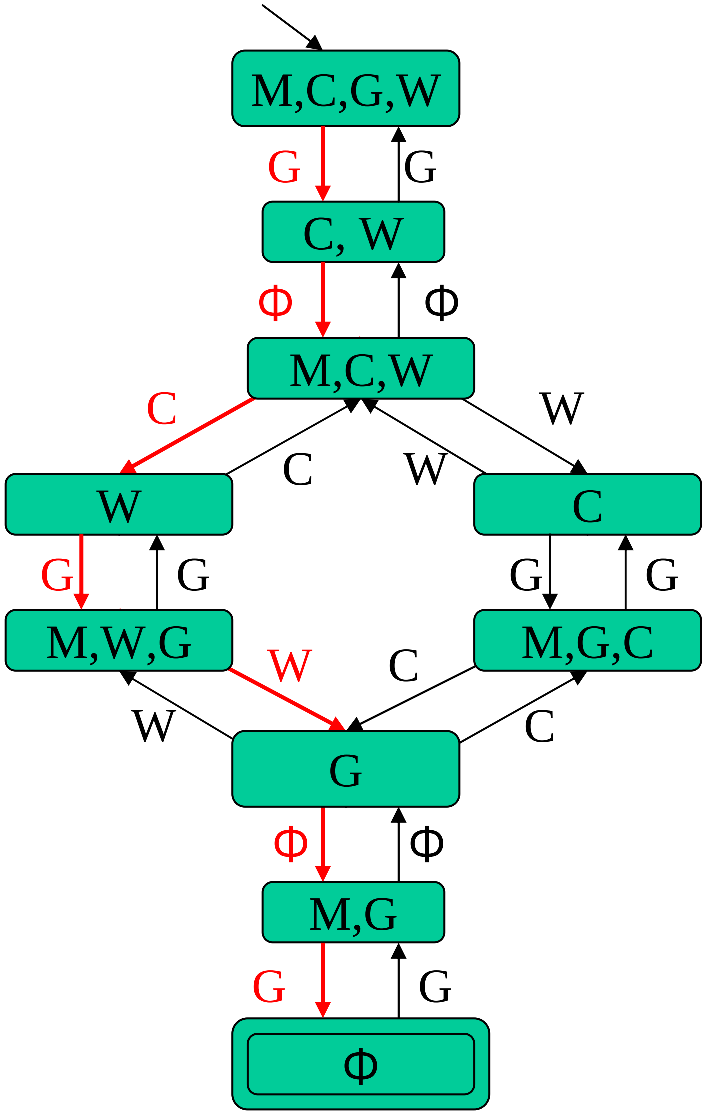
- 船による運搬問題の状態遷

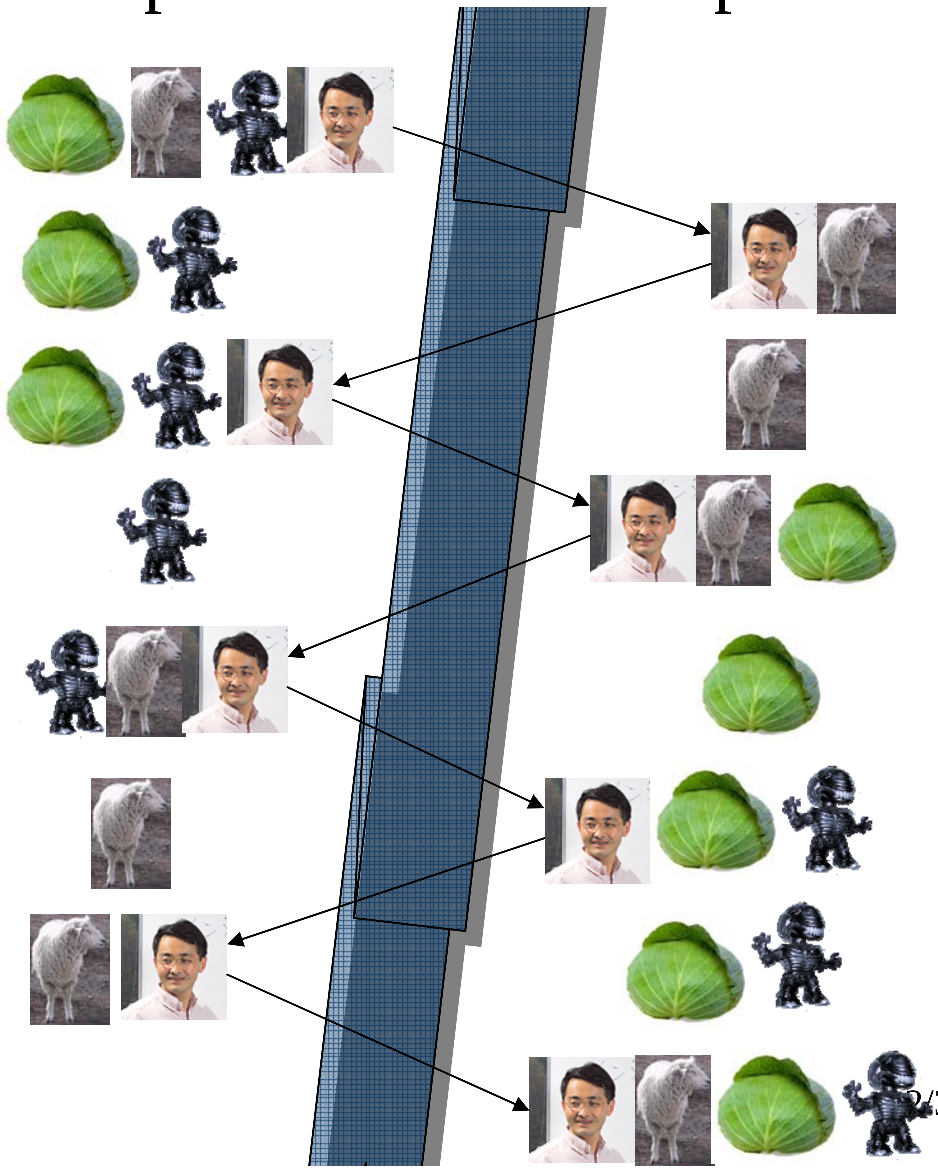




2.1. Prelimi

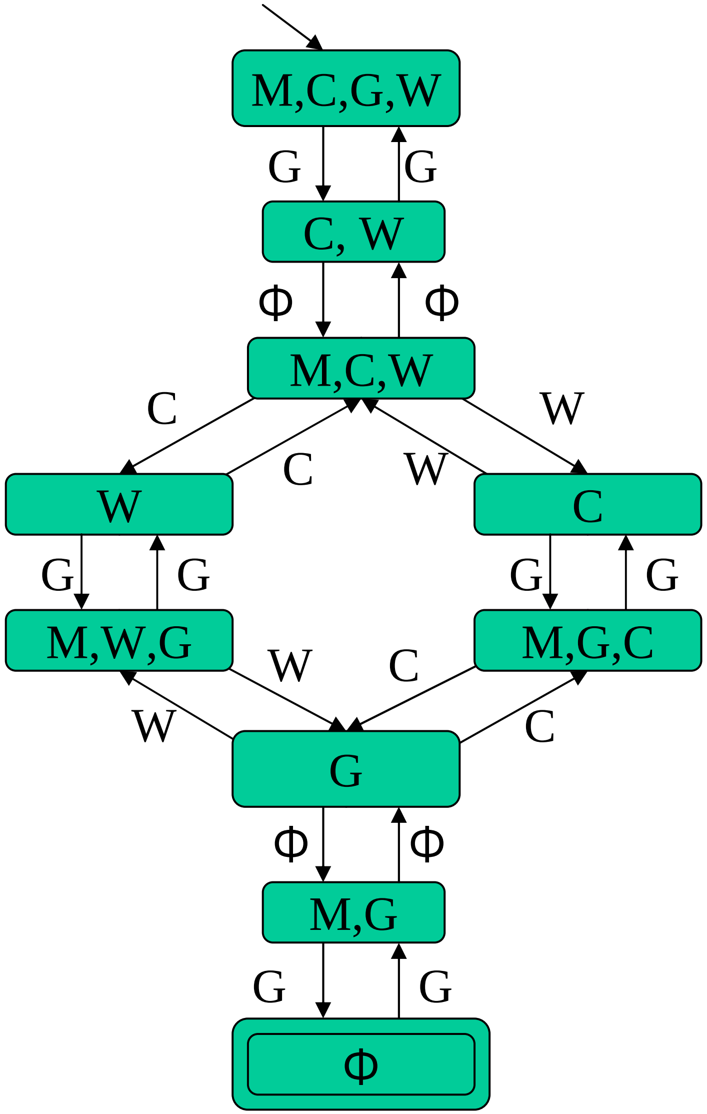
- A state transition diagram of the problem for a transporter





2.1. 直感的説明

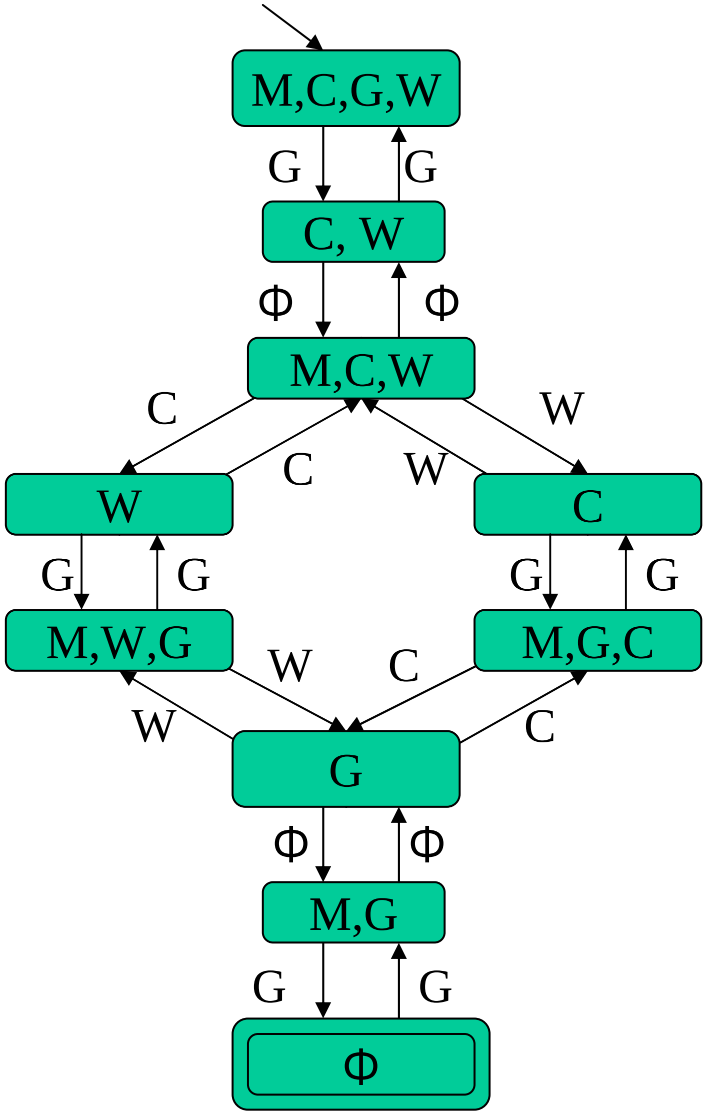
- 船による運搬問題の状態遷移図



- 「解」は「初期状態」から「受理状態」へたどりつく任意の路
- 無限に解がある
- 以下の二つを理論的に保証できる(手数＝船に乗る回数)
  1. 手数が7の解が存在する
  2. 手数が7未満の解は存在しない

2.1. Preliminaries

- A state transition diagram of the problem for a transporter



- A 'Solution' is given by <u>any path</u> from the initial state to the accepting state

- There are infinite solutions

- We can guarantee the followings theoretically (turn = # of riding)
  1. There <u>is</u> a solution of 7 turns.
  2. <u>No solutions</u> less than 7 turns.

2.2. 決定性有限オートマトンの形式的定義

- 決定性有限オートマトン(DFA)の定義
  1. 状態(state)の有限集合Q
  2. 入力記号(input symbols)の有限集合 Σ
  3. 遷移関数(transition function) δ
     - 入力は(状態,入力記号)のペア；今の状態と、それへの入力
     - 出力は状態；次の状態
  4. 初期状態(または開始状態)q ($q \in Q$)
  5. 受理状態(または最終状態)F ($F \subseteq Q$)

– DFA Aは$A=(Q, \Sigma, \delta, q, F)$の5つ組で表現される。

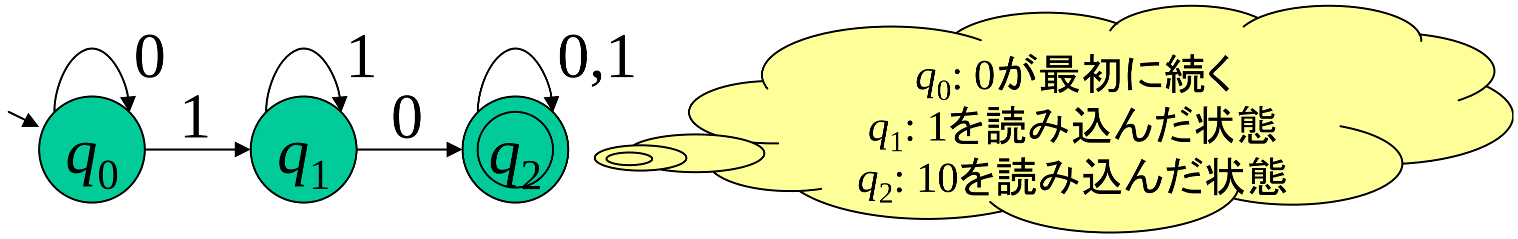
2.2. Formal definition of a DFA

- Definition of a DFA
  1. Q: a finite set of states
  2. Σ : a finite set of input symbols
  3. δ : a transition function
     - Input is (state, symbol), which means the current state and given input
     - Output is a state, which means the next state
  4. $q \in Q$: initial state (or start state)
  5. $F \subseteq Q$: accepting state (or final state)

– DFA A is defined by a 5-tuple $A=(Q, \Sigma, \delta, q, F)$

2.2. 決定性有限オートマトンの形式的定義

例:「0,1からなる文字列で、文字列10を含む」文字列



– 上記の言語を受理するDFA $A=(Q, \Sigma, \delta, q_0, F)$は次の通り:
  – $Q = \{q_0, q_1, q_2\}$
  – $\Sigma=\{0,1\}$
  – δは右の表
  – $F=\{q_2\}$

	q_0	q_1	q_2
0	q_0	q_2	q_2
1	q_1	q_1	q_2

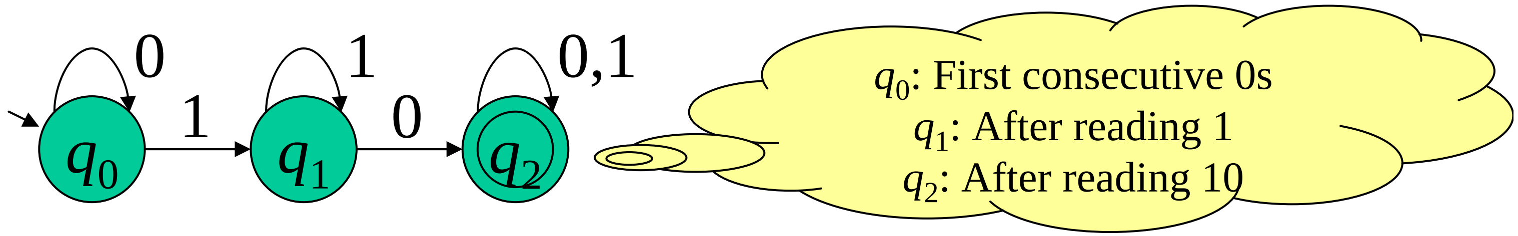
例: $\delta(q_1,0)=q_2$

– 形式的定義は
  – 論文など、厳密性を要求される文章を書くとき
  – 機械的・一般的に処理したいとき

に必要になる。

2.2. Formal definition of a DFA

Ex : Words over '0,1' contain substring 10



- The language above is accepted by the following DFA $A=(Q, \Sigma, \delta, q_0, F)$:
  - $Q = \{q_0, q_1, q_2\}$
  - $\Sigma = \{0,1\}$
  - δ is given by the right table
  - $F=\{q_2\}$

	q_0	q_1	q_2
0	q_0	q_2	q_2
1	q_1	q_1	q_2

Ex: $\delta(q_1,0)=q_2$

- **Formal definitions** are required when
  - Writing a formal document, like a paper,
  - They will be dealt machinery.

2.2. 決定性有限オートマトンの形式的定義

関数 δ は定義域は[Qの要素とΣの要素のペア]で、値域はQの要素

- 遷移関数 δ は
  - $\delta: Q\times \Sigma \rightarrow Q$

  を満たす関数。これを自然に拡張した
  - $\hat{\delta}: Q\times \Sigma^* \rightarrow Q$

  を次のように定義する。

本当は②は冗長

① $\hat{\delta}(q, \varepsilon) = q$ for any $q \in Q$

② $\hat{\delta}(q, a) = \delta(q, a)$ for any $a \in \Sigma$

③ $\hat{\delta}(q, w) = \delta(\hat{\delta}(q, w'), a)$ for $w=w'a \in \Sigma^+$

- DFA A の言語(より正確には DFA A によって受理される言語) $L(A)$ とは, $A=(Q, \Sigma, \delta, q_0, F)$に対し次のように定義される。

$$L(A) = \{ w \mid \hat{\delta}(q_0, w) \in F\}$$

2.2. Formal definition of a DFA

- Transition function δ is
  – $\delta : Q \times \Sigma \rightarrow Q$

Function δ has domain [the pair of an element in Q and an element in Σ], and range [an element in Q]

That can be extended naturally as follows
  – $\hat{\delta} : Q \times \Sigma^* \rightarrow Q$

② is redundant

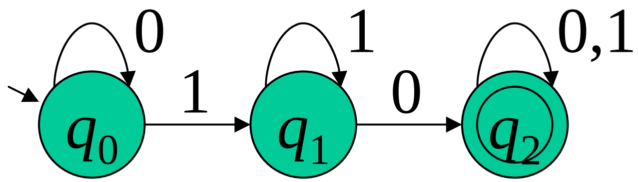
① $\hat{\delta}(q, \varepsilon) = q$ for any $q \in Q$
② $\hat{\delta}(q, a) = \delta(q, a)$ for any $a \in \Sigma$
③ $\hat{\delta}(q, w) = \delta(\hat{\delta}(q, w'), a)$ for $w = w'a \in \Sigma^+$

- The language $L(A)$ accepted by the DFA A is defined by

$$L(A) = \{ w \mid \hat{\delta}(q_0, w) \in F \}$$

2.2. 決定性有限オートマトンの形式的定義

例:「0,1からなる文字列で、文字列10を含む」文字列



– 上記の言語を受理するDFA $A=(Q, \Sigma, \delta, q_0, F)$は:

– $Q = \{q_0, q_1, q_2\}$

– $\Sigma=\{0,1\}$

– δは右の表

– $F=\{q_2\}$

	q_0	q_1	q_2
0	q_0	q_2	q_2
1	q_1	q_1	q_2

1. 入力 0100 に対する動作例:

$$\hat{\delta}(q_0, 0100) = \delta(\hat{\delta}(q_0, 010), 0) = \delta(\delta(\hat{\delta}(q_0, 01), 0, 0) = \delta(\delta(\delta(\hat{\delta}(q_0, 0), 1), 0), 0)$$

$$= \delta(\delta(\delta(\underline{\delta(q_0, 0)}, 1), 0), 0) = \delta(\delta(\underline{\delta(q_0, 1)}, 0), 0) = \delta(\underline{\delta(q_1, 0)}, 0) = \delta(q_2, 0) = q_2 \in F$$

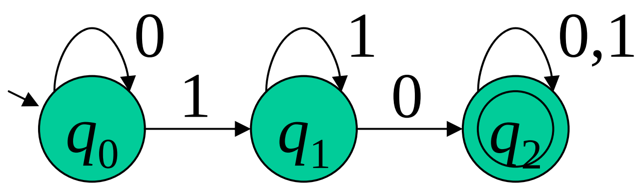
2. 入力 0011 に対する動作例:

$$\hat{\delta}(q_0, 0011) = \delta(\hat{\delta}(q_0, 001), 1) = \delta(\delta(\hat{\delta}(q_0, 00), 1), 1) = \delta(\delta(\delta(\hat{\delta}(q_0, 0), 0), 1), 1)$$

$$= \delta(\delta(\delta(\underline{\delta(q_0, 0)}, 0), 1), 1) = \delta(\delta(\underline{\delta(q_0, 0)}, 1), 1) = \delta(\underline{\delta(q_0, 1)}, 1) = \delta(q_1, 1) = q_1 \notin F$$

2.2. Formal definition of a DFA

Ex : Words over '0,1' contain substring 10



– The language above is accepted by DFA $A=(Q, \Sigma, \delta, q_0, F)$:
  – $Q = \{q_0, q_1, q_2\}$
  – $\Sigma = \{0,1\}$
  – δ is given by the right table
  – $F=\{q_2\}$

	q_0	q_1	q_2
0	q_0	q_2	q_2
1	q_1	q_1	q_2

1. Transition for the input 0100:

$$\hat{\delta}(q_0, 0100) = \delta(\hat{\delta}(q_0, 010), 0) = \delta(\delta(\hat{\delta}(q_0, 01), 0, 0) = \delta(\delta(\delta(\hat{\delta}(q_0, 0), 1), 0), 0)$$

$$= \delta(\delta(\delta(\underline{\delta(q_0, 0)}, 1), 0), 0) = \delta(\delta(\underline{\delta(q_0, 1)}, 0), 0) = \delta(\underline{\delta(q_1, 0)}, 0) = \delta(q_2, 0) = q_2 \in F$$

2. Transition for the input 0011:

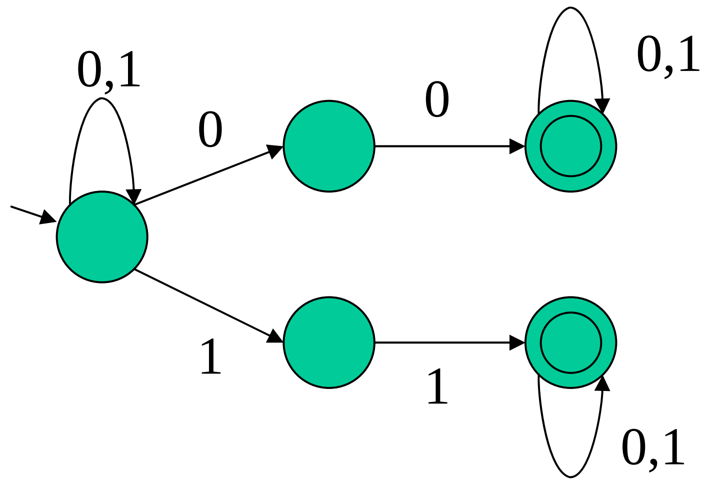
$$\hat{\delta}(q_0, 0011) = \delta(\hat{\delta}(q_0, 001), 1) = \delta(\delta(\hat{\delta}(q_0, 00), 1), 1) = \delta(\delta(\delta(\hat{\delta}(q_0, 0), 0), 1), 1)$$

$$= \delta(\delta(\delta(\underline{\delta(q_0, 0)}, 0), 1), 1) = \delta(\delta(\underline{\delta(q_0, 0)}, 1), 1) = \delta(\underline{\delta(q_0, 1)}, 1) = \delta(q_1, 1) = q_1 \notin F$$

2.3. 非決定性有限オートマトン

- 例: Σ={0,1}上の文字列で、'00'または '11' を含むもの

自然に思いつくオートマトン(?):



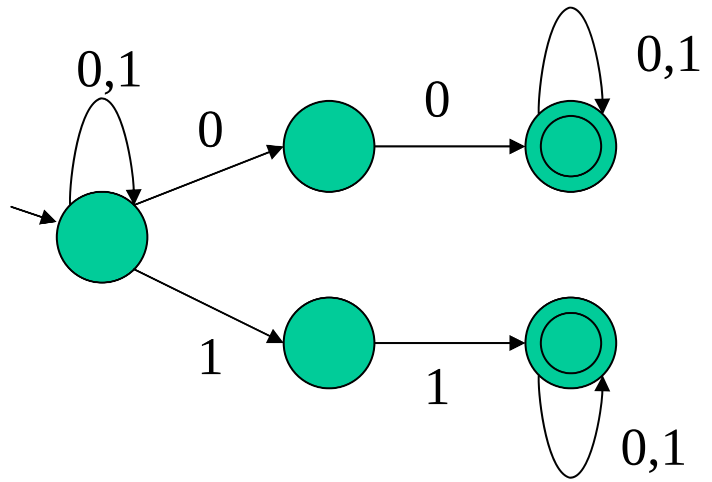
★入力に対する遷移先が1つではない

⇒非決定性有限オートマトン(NFA; Nondeterministic Finite Automaton)

2.3. Nondeterministic Finite Automaton

Ex: Words contains '00' or '11' as a substring

Natural idea of the automaton for the language (?):



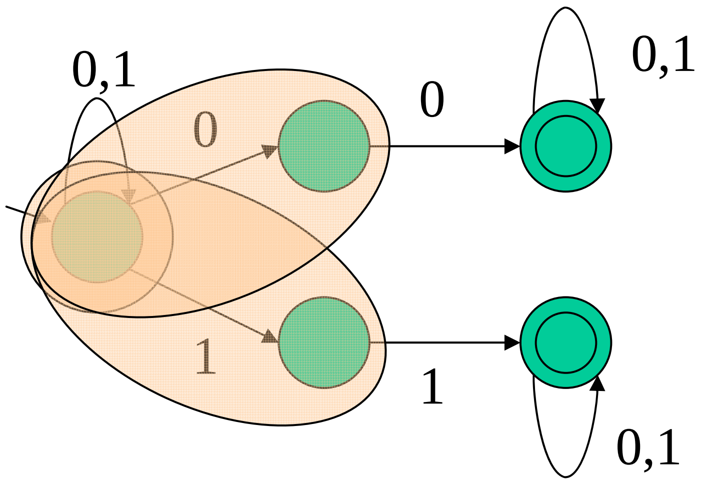
★Transition for an input is not determined uniquely

⇒Nondeterministic Finite Automaton (NFA)

2.3. 非決定性有限オートマトン

- 例: $\Sigma=\{0,1\}$上の文字列で、'00'または '11' を含むものを受理する非決定性有限オートマトン

入力 10101 に対する動作例

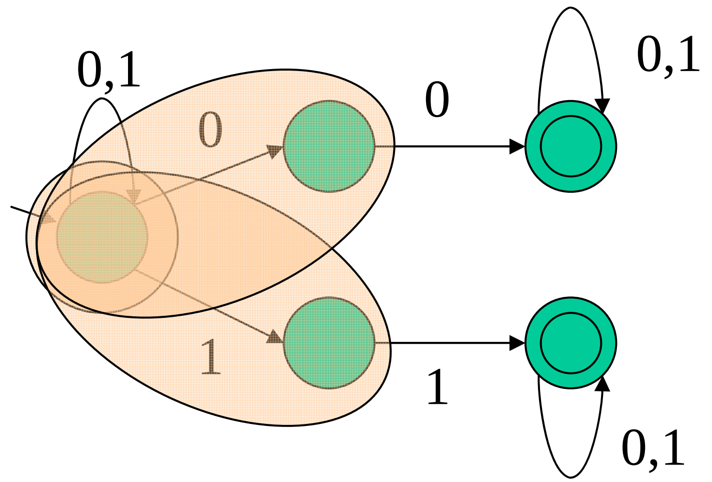


1 0 1 0 1

2.3. Nondeterministic Finite Automaton

Ex: Nondeterministic Finite Automaton that accepts words contains ’00’ or ’11’ as a substring

Transitions for the input 10101

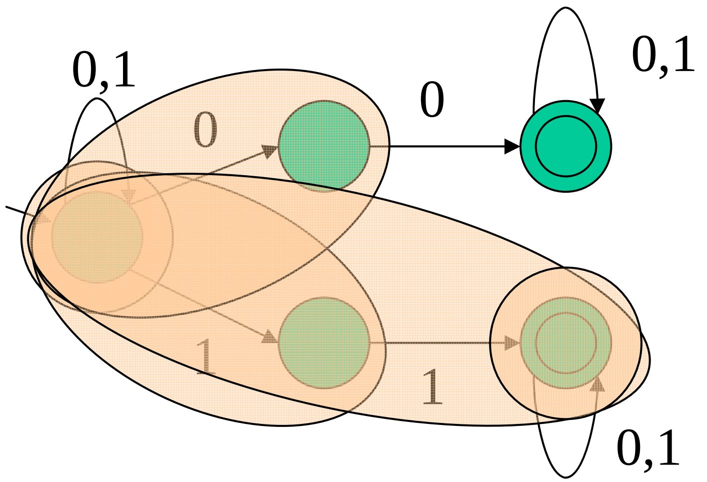


1 0 1 0 1

2.3. 非決定性有限オートマトン

- 例: $\Sigma=\{0,1\}$上の文字列で、'00'または '11' を含むものを受理する非決定性有限オートマトン

入力 10110 に対する動作例

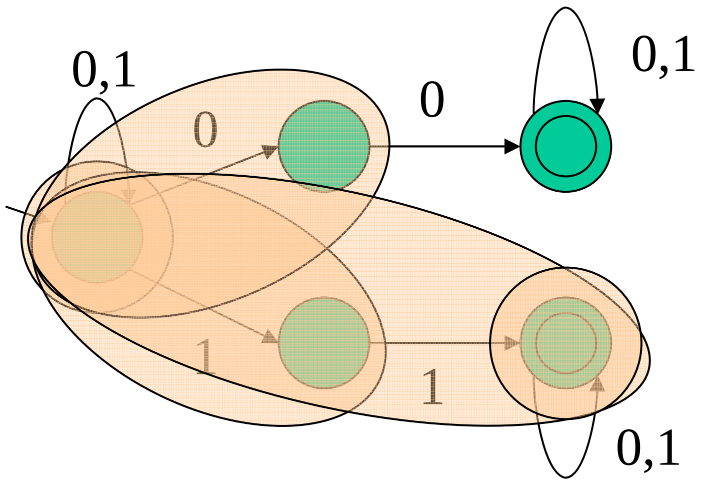


1 0 1 1 0

2.3. Nondeterministic Finite Automaton

Ex: Nondeterministic Finite Automaton that accepts words contains ’00’ or ’11’ as a substring

Transitions for an input 10110

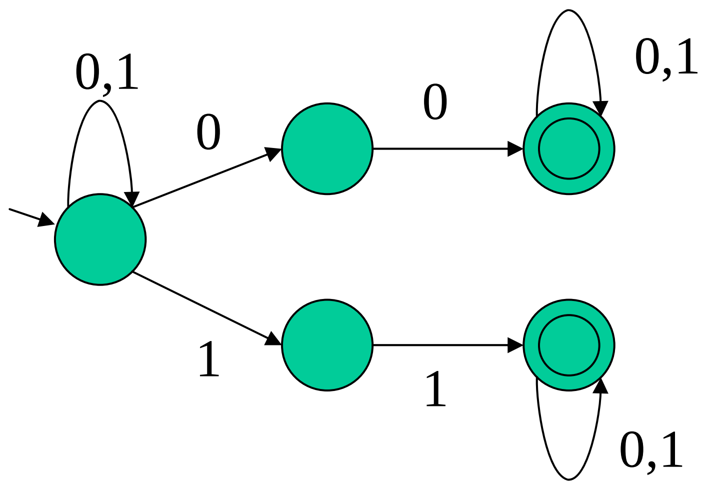


1 0 1 1 0

2.3. 非決定性有限オートマトン

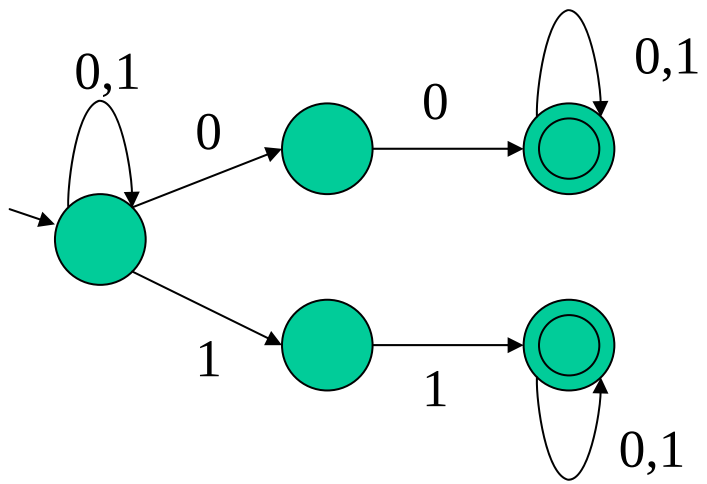
- 非決定性有限オートマトン
  - 特定の入力に対する遷移先が複数あってもよい
  - 遷移先は ‘遷移可能なすべての状態の集合’
  - 受理の条件は ‘遷移した状態集合と受理状態が共通部分を持つ’

という3点が決定性有限オートマトンと違う。



2.3. Nondeterministic Finite Automaton

- Nondeterministic Finite Automaton differs from DFA on the following points:
  - The transition can be many states for an input
  - Next 'state' is the set of all possible states
  - The NFA accepts the input if the set of all possible states after reading the input <u>contains at least one accepting state</u>



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の形式的定義

NFA $A=(Q, \Sigma, \delta, q_0, F)$

– Q, Σ, q_0, F は決定性と同じ

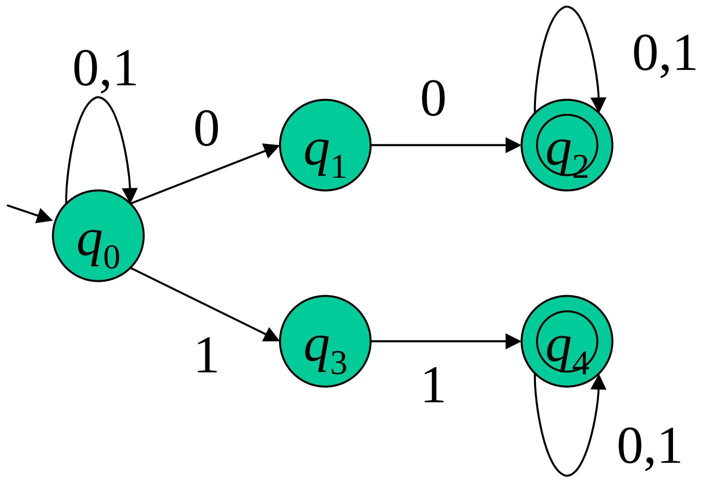
✓ δは

$\delta: Q\times \Sigma \rightarrow 2^Q$

2^S: 集合Sのすべての部分集合の集合
Ex.: $2^{\{0,1\}}=\{\Phi,\{0\},\{1\},\{0,1\}\}$

✓ 受理の条件は ‘遷移した状態集合と受理状態が共通部分を持つ’

例: $A=(\{q_0,q_1,q_2,q_3,q_4\}, \{0,1\}, \delta, q_0, \{q_2,q_4\})$



2.3. Nondeterministic Finite Automaton

- Formal definition of a NFA

NFA $A=(Q, \Sigma, \delta, q_0, F)$

– Q, Σ, q_0, F are the same as DFA

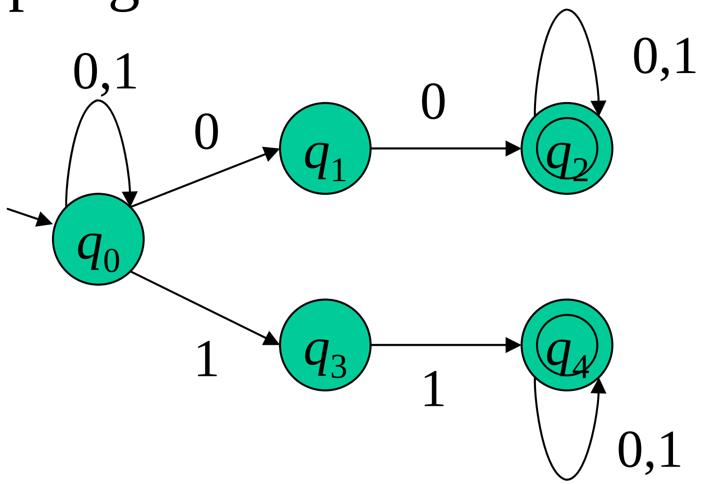
✓ δ is defined

$\delta : Q \times \Sigma \rightarrow 2^Q$

2^S: the set of all subsets of S
Ex.: $2^{\{0,1\}}=\{\Phi,\{0\},\{1\},\{0,1\}\}$

✓ It accepts if the set of states for the input has an intersection with the accepting states F

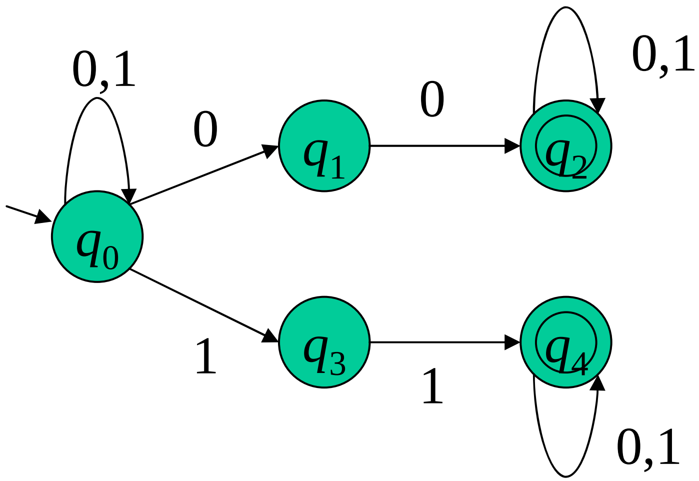
Ex: $A=(\{q_0,q_1,q_2,q_3,q_4\},$
$\{0,1\}, \delta, q_0, \{q_2,q_4\})$



2.3. 非決定性有限オートマトン

例: $A=(\{q_0,q_1,q_2,q_3,q_4\},\{0,1\},\delta,q_0,\{q_2,q_4\})$

δ	0	1
q_0	$\{q_0,q_1\}$	$\{q_0,q_3\}$
q_1	$\{q_2\}$	Φ
q_2	$\{q_2\}$	$\{q_2\}$
q_3	Φ	$\{q_4\}$
q_4	$\{q_4\}$	$\{q_4\}$



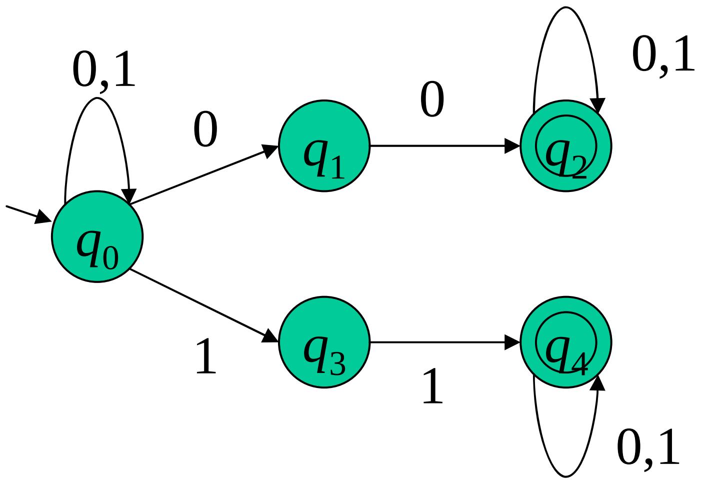
[状態,入力]から
[状態集合]への関数として

遷移関数 δの自然な拡張 $\hat{\delta}$も同様に定義できる。

2.3. Nondeterministic Finite Automaton

Ex: $A=(\{q_0,q_1,q_2,q_3,q_4\},\{0,1\}, \delta,q_0,\{q_2,q_4\})$

δ	0	1
q_0	$\{q_0,q_1\}$	$\{q_0,q_3\}$
q_1	$\{q_2\}$	Φ
q_2	$\{q_2\}$	$\{q_2\}$
q_3	Φ	$\{q_4\}$
q_4	$\{q_4\}$	$\{q_4\}$



As a function from [state,input] to [set of states]

The natural extension $\hat{\delta}$ of the transition function δ can be defined similarly.

2.3. 非決定性有限オートマトン

- NFA *A* の言語(より正確には NFA *A* によって受理される言語) $L(A)$ とは, $A=(Q, \Sigma, \delta, q_0, F)$に対し次のように定義される。

  $$L(A) = \{\, w \mid \hat{\delta}(q_0, w) \cap F \neq \Phi \}$$

C.f. DFAの場合は $L(A) = \{\, w \mid \underline{\hat{\delta}(q_0, w) \in F} \}$ であった。

2.3. Nondeterministic Finite Automaton

- The language $L(A)$ accepted by the NFA $A=(Q,\Sigma,\delta,q_0,F)$ is defined as follows:

  $L(A) = \{ w \mid \hat{\delta}(q_0,w) \cap F \neq \Phi \}$

C.f. In the case of DFA, $L(A) = \{ w \mid \underline{\hat{\delta}(q_0,w) \in F} \}$.