CAP 4800/5805 Computer Simulation Concepts

Lecture 14 and 17

Markov model

- Markov models are state machines like finite state automata, but the major difference is that Markov machines are nondeterministic
- Each arc in the Markov graph model is labeled with a probability of taking that arc when transitioning to another state
- Formal definition
    1. MARKOV = $<P, O, Q, \delta, \lambda>$
    2. $P = [0.0, 1.0]$
    3. $\delta : Q \times P \rightarrow Q$
    4. $\lambda : Q \rightarrow O$

    - P is the set of probabilities, Q is the state set, δ is the state transition function, and λ is the output function

Markov matrix

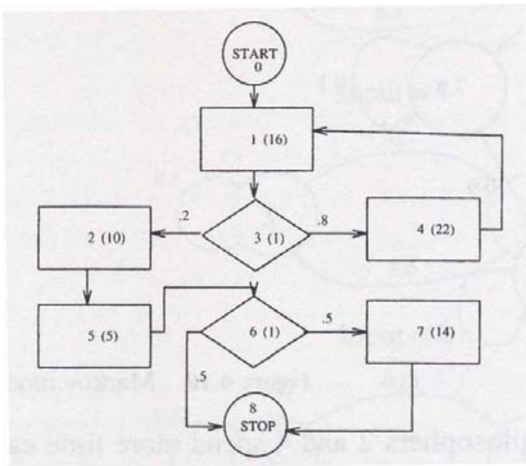
$P\{X_{n+1} = j \mid X_n = i\} = P(i, j)$

$P(i, j)$ means the probability of transitioning from state i to state j

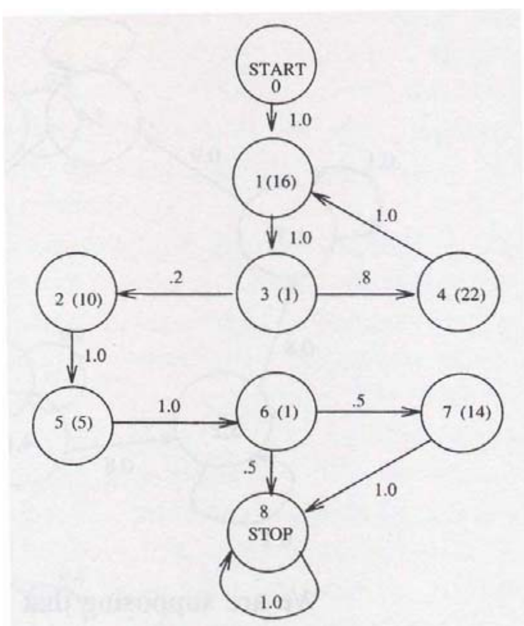
$$\begin{bmatrix} P(0,0) & P(0,1) & P(0,2) & \ldots \\ P(1,0) & P(1,1) & P(1,2) & \ldots \\ P(2,0) & P(2,1) & P(2,2) & \ldots \\ \cdot & \cdot & \cdot & \ldots \\ \cdot & \cdot & \cdot & \ldots \\ \cdot & \cdot & \cdot & \ldots \end{bmatrix}$$

Example

- **Process flow chart**



- **Markov model**

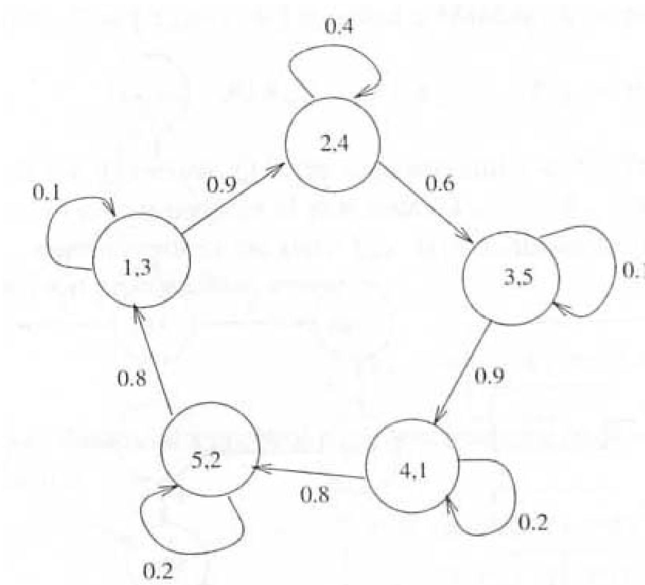


- **Probability matrix**

$$\begin{bmatrix} 0.0 & 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 1.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.2 & 0.0 & 0.8 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 1.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.5 & 0.5 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 1.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix}$$

Markov model for DP

- **Markov model**



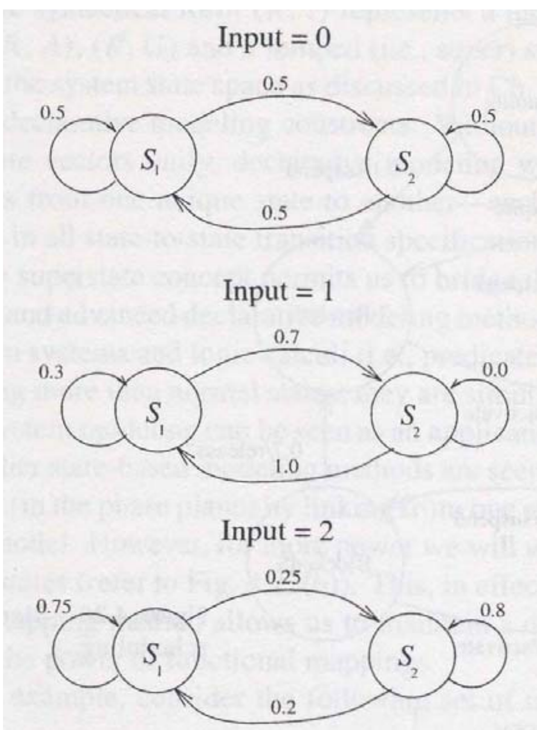
- **Formal definition**

1. MARKOV = <*P*, *O*, *Q*, δ, λ>
2. *P* = [0, 1]
3. *O* = {(1, 3), (2, 4), (3, 5), (4, 1), (5, 2)}
4. $Q = \{q_0, \ldots, q_4\}$

5. $\delta : Q \times P \rightarrow Q$
6. $\lambda : Q \rightarrow O$

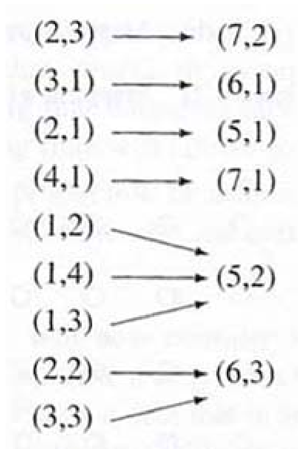
Markov model with input

Markov models may also be defined with input. To provide for input as well as the influence of probability over a change in state, we specify a Markov model for each value that an input can assume. Figure 4.19 shows a two state/three input Markov model. The two states are $S1$ and $S2$ and the three inputs are 0, 1 and 2. This system is controlled by the input but is not determinate. Supposing our initial state is $S1$ and our input string is 0211. We have a 50% chance of staying in state $S1$ or continuing to $S2$—we'll assume that we stay in $S1$. Now, since the input changes from 0 to 2, we refer to the third model in Fig. 4.19. After sampling from a probability mass function (pmf) we move to $S2$ with a probability of 0.25. Then, we switch to the second Markov model since our last two input values are both 1. We now move back to state $S1$ with probability of 1.0. The FSA with input is a special case of the Markov model with input where the probabilities are either 0 or 1. For instance, in Fig. 4.19, if we replace all of the probabilities with those of 0 or 1, we now have a determinate system that is more appropriately drawn as a single FSA model.



Production-based models

- Superstate: a subset (rather than an element) of state space
- Declarative modeling would be limited to models delineating transitions from one unique state to another
- Superstate concept permits us to bridge the gap between the state space systems approach and advanced declarative modeling methods
  → mapping a group of states to another group of states
- **Mapping example**
  - **Mapping**



  - **Production rules**

1. $(2, 3) \rightarrow (7, 2)$
2. $(?x, 1), x \in \{2, 3, 4\} \rightarrow (x + 3, 1)$
3. $(1, ?x), x \in \{2, 3, 4\} \rightarrow (5, 2)$
4. $(?x, x), x \in \{2, 3\} \rightarrow (6, 3)$

- ? means "match any number" and ?*x* means match any number and store the number in the variable *x*