CS 440
Introduction to Artificial Intelligence

Lecture 26:

Monte Carlo Markov Chains

April 16, 2020

Markov Processes

- **Markovian assumption**
  - **Future state only depend on current state**
  - **Only matters where cars currently are on road**
    - **Doesn't matter what maneuvers they took to get there**
  - **Assumption makes solving problems a lot easier**
    - **Only need to keep track of current state**
      - **As opposed to history of previous states**
    - **Only need to reason over current state**

Types of Markov Processes

- **Discrete vs continuous**
- **Passive vs active**
  - **Active process: The agent's actions influence process**
- **Observable state vs partially observable state**
  - **Observable state: agent can observe state directly**
  - **Partially observable state:**
    - **Example: Wumpus world**

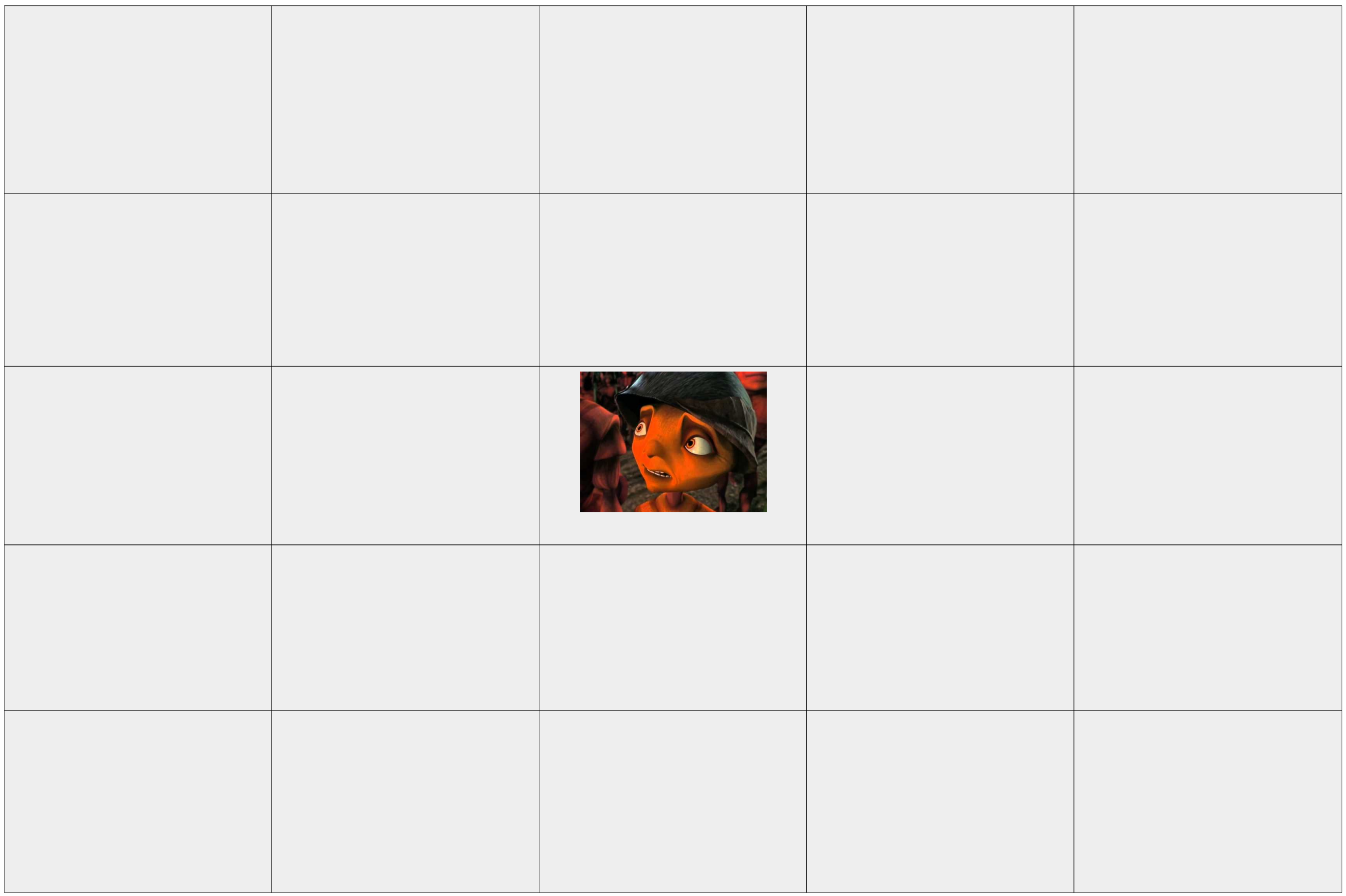
	Observable State	Hidden State
Passive	Markov Chain	Hidden Markov Model
Active	Markov Decision Process (MPD)	Partially Observable Markov Decision Process (POMPD)

Solutions to Markov Chain

- Given an environment in a known state
  - What could the environment look like after n steps?
  - Probability of being in each state
- Solve inductively
  - Assume we know the probability you are in each state s at step i
    - $p_{s,i}$ for all $s \in S$
  - Compute probability for each state at step i+1
    - $p_{s',i+1}$ for all $s' \in S$
  - Probability we will transition from state s to s' at step i+1 equal to probability we are in state s at step i times the transition probability from s to s'
    - $p_{s',i+1} = p(s'|s) * p_{s,i}$
  - Probability we will be in state s' at step i+1
    - $p_{s',i+1} = \sum_{s' \in S} p(s'|s) * p_{s,i}$

Markov Chains Example

- **And moves up/down/left/right with probability p**
  - **stays in same cell with probability 1-p**
- **We don't know what move the ant will make**

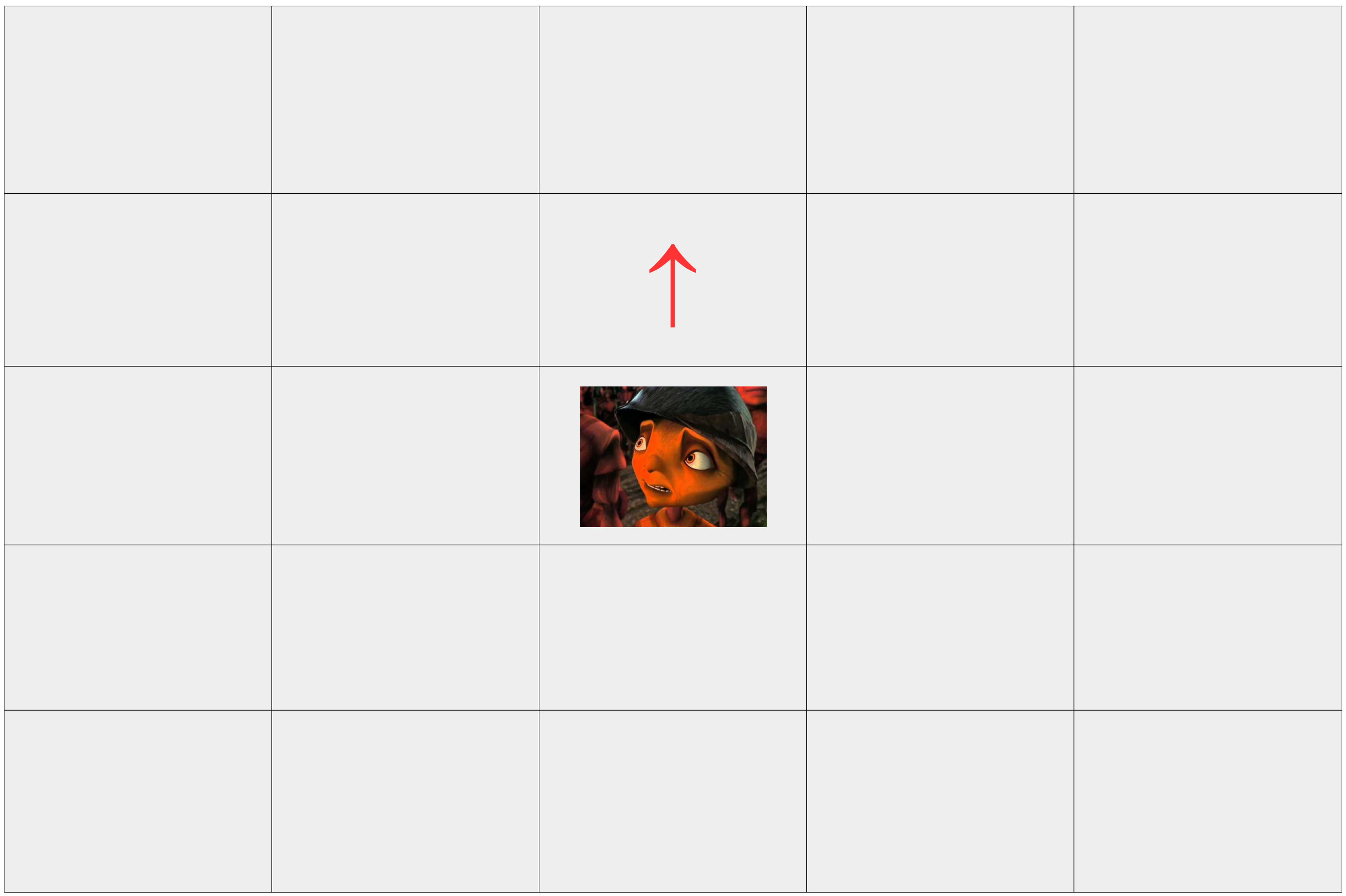


Monte Carlo Sampling

- **Sample**
  - **Start at initial state**
  - **At each step randomly select transition according to probability distribution**
- **Take many samples**
  - **Approximate aggregate behavior of system**

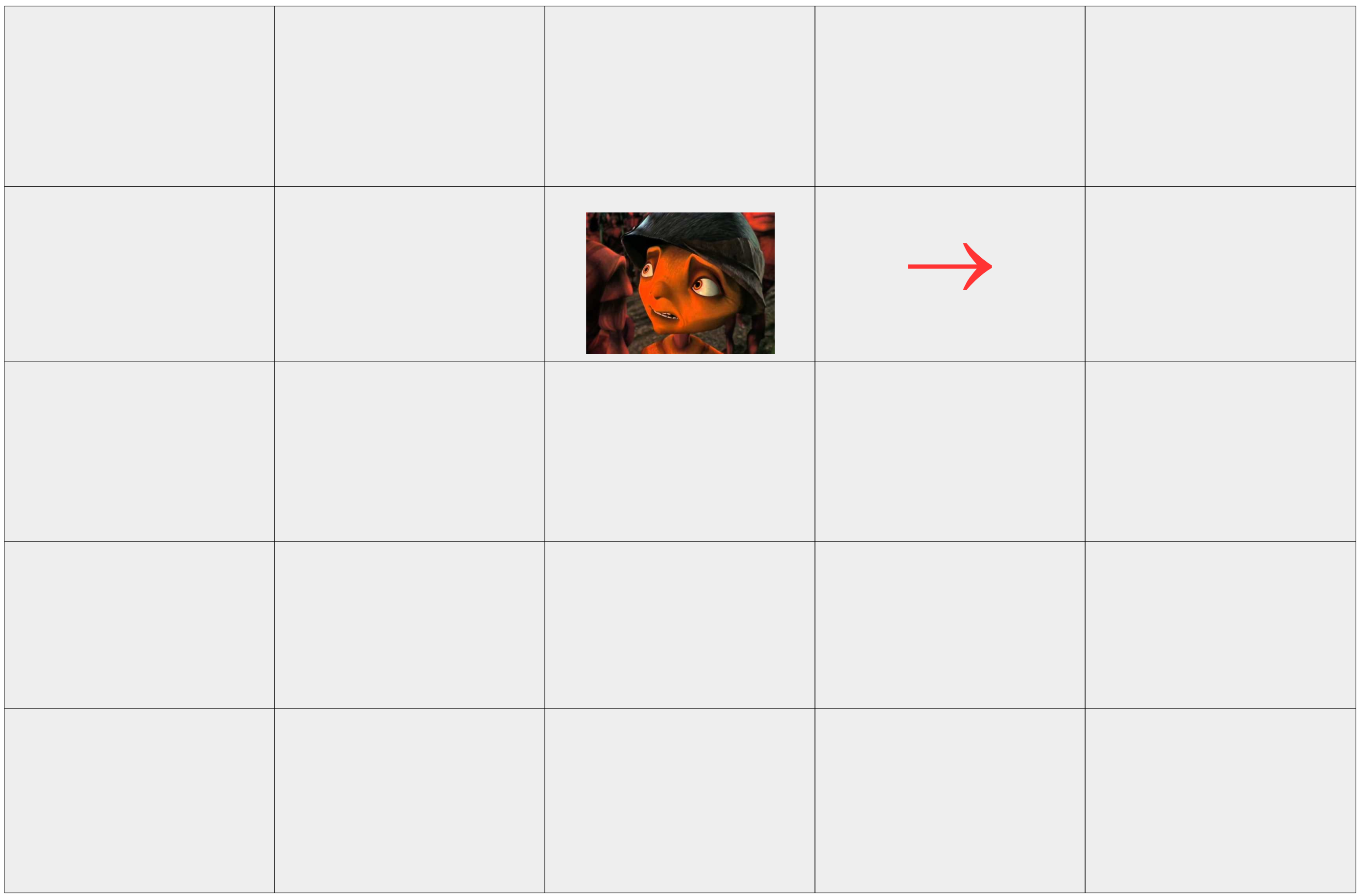
MCMC Example

- **First sample**
  - **First move**
    - **Roll the dice and get UP**



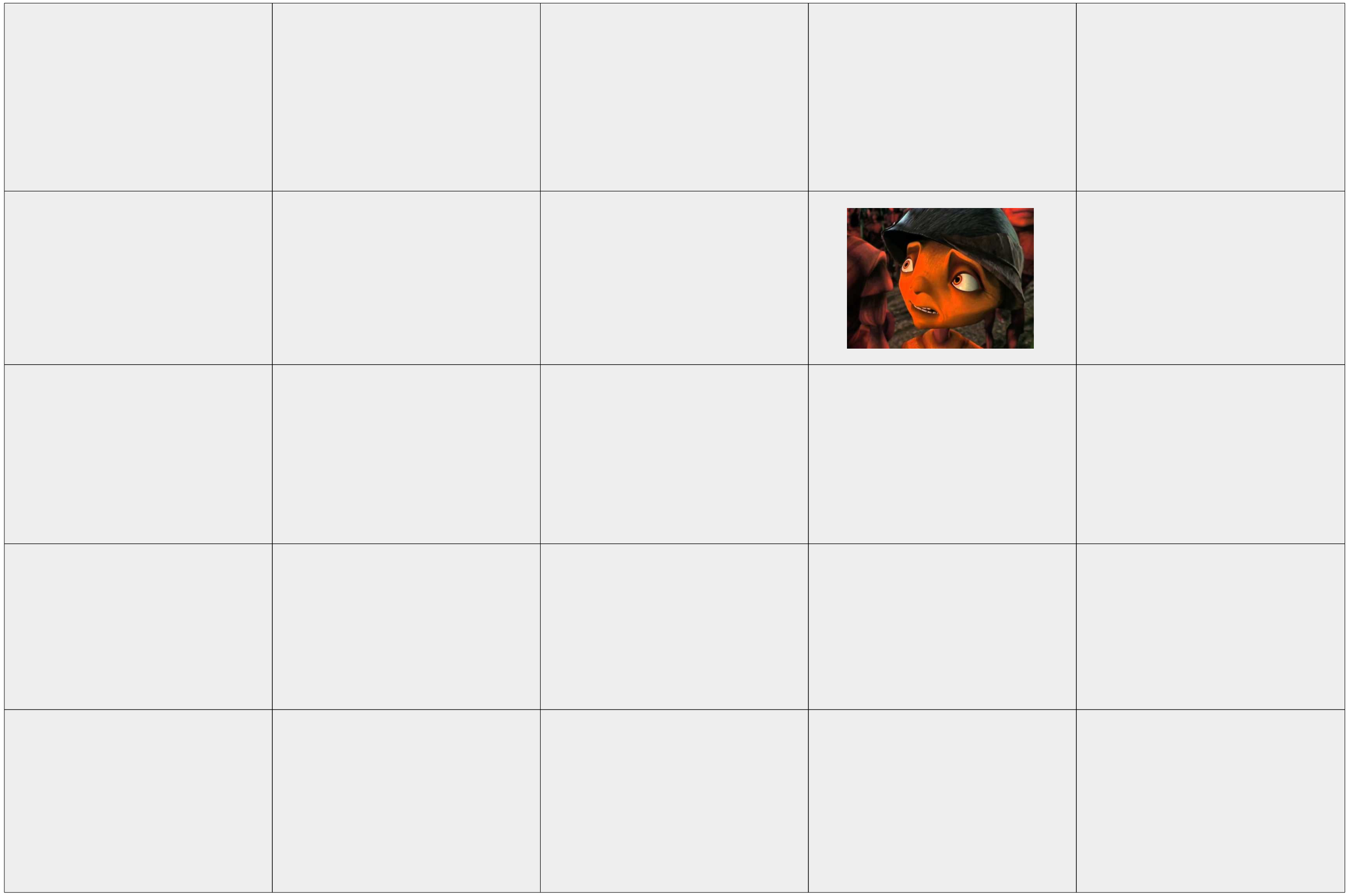
MCMC Example

- **First sample**
  - **Second move**
    - **Roll the dice and get Right**



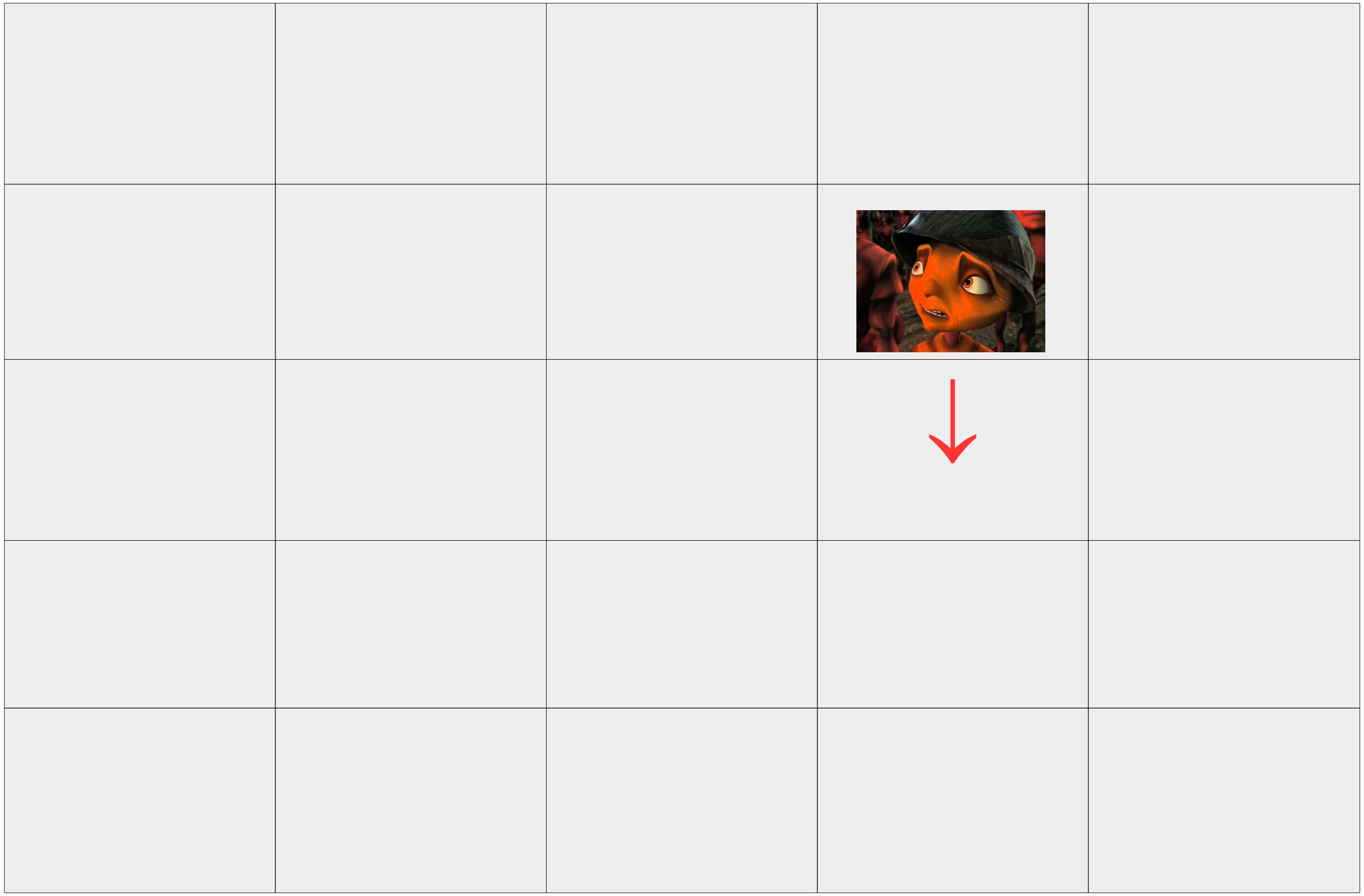
MCMC Example

- **First sample**
  - **third move**
    - **Roll the dice and get stay in same place**



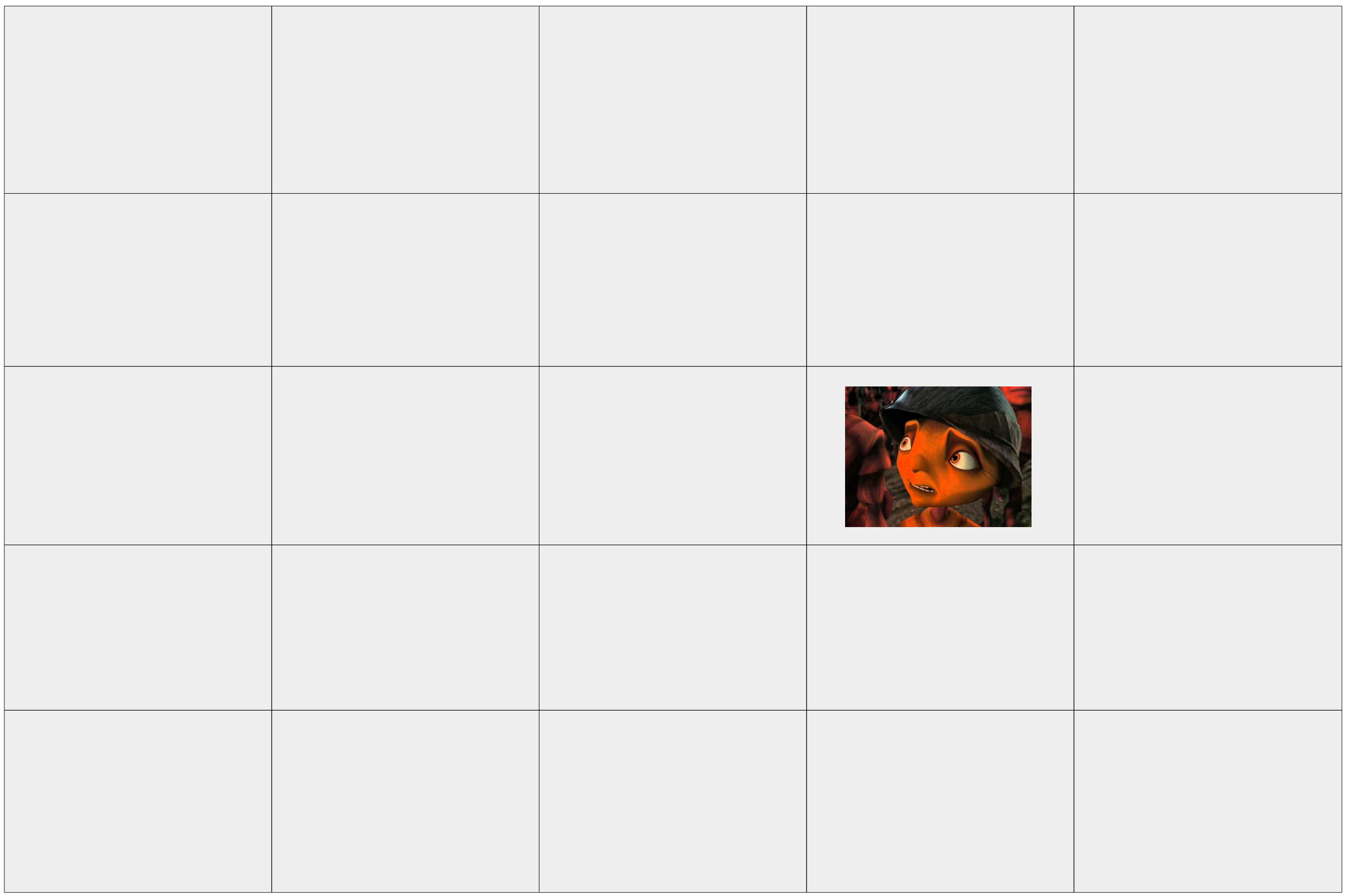
MCMC Example

- **First sample**
  - **forth move**
    - **Roll the dice and get down**



MCMC Example

- First sample = {Up,Right,Stay,Down}
- Repeat process to get many samples
- Proportion of samples that end in each cell approximates probability ant will be in that cell after 4 moves
  - As number of samples goes to , converges to actual probability
    - Actual probability = probability obtained from computing directly



Why use MCMC

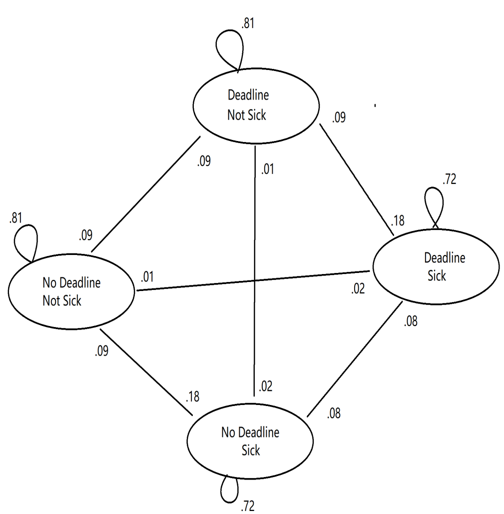
- **Simpler then computing exact solution**
- **Anytime algorithm**
  - **Can stop algorithm after specified time and use samples that have been generated**
  - **Useful for problems where there is a fixed time window**

Application to Hidden Markov Models

- Sample
  - Initialize hidden variables based on probability distribution
  - Each step
    - Randomly select observable variables according to current setting of hidden variables
    - Transition hidden variables according to transition probability
- Approximate probability of x given y
  - Take all samples for which y is true
  - Compute proportion of those sample for which x is true
- Example: Compute probability of hidden variable value given a sequence of observations
  - Take all samples with that sequence of observations
  - Find portion of those samples with hidden variable value
- Compute next move given a sequence of moves
  - Given moves 0 through i, compute move i+1
  - Take all samples with sequence where moves 0 through i are the given sequence of moves
  - Distribution of move i+1 on those samples

Hidden Markov Model: Example

- Consider the following model
  - If Bob has a paper deadline at time step i there is a .9 probability he will have a paper deadline at step i+1
  - If Bob has a no deadline at time step i there is a .1 probability he will have a paper deadline at step i+1
  - If Bob is sick at time step i there is a .8 probability he will be sick at step i+1
  - If Bob is not at time step i there is a .1 probability he will be sick at step i+1
- If we know the values of deadline and sick this is a Markov Chain



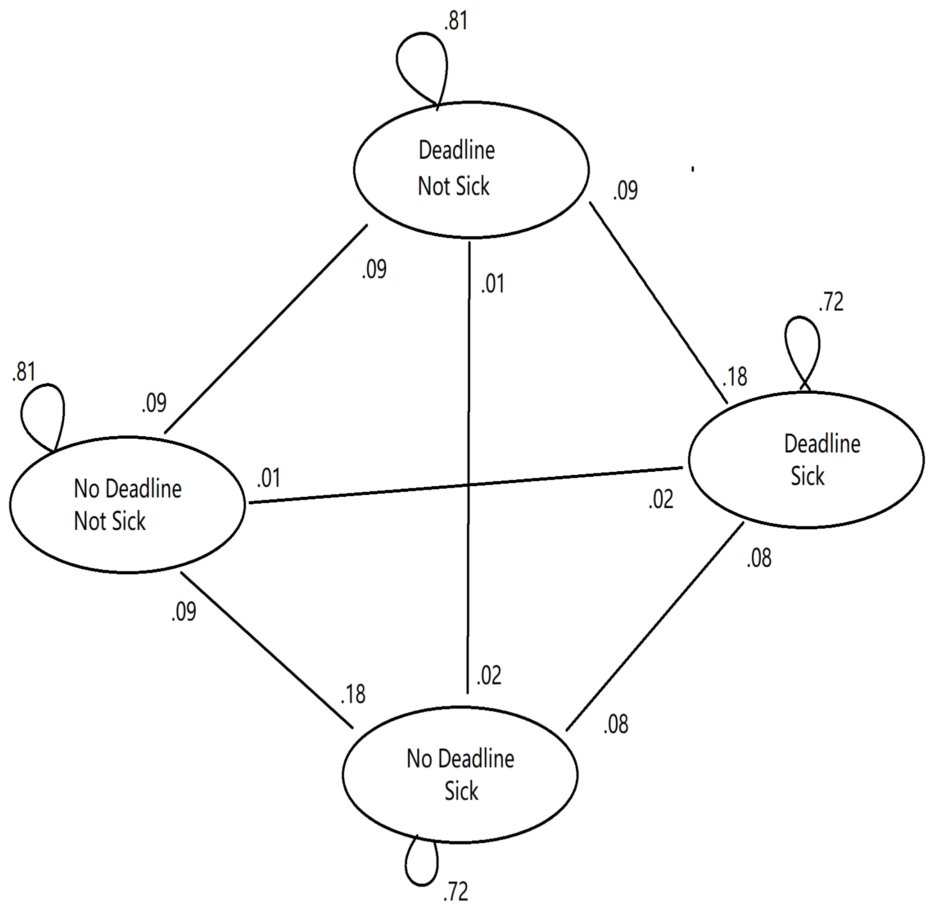
Hidden Markov Model: Example

	Not Sick	Sick
Deadline	.9	.6
No Deadline	.7	.1

- One an ordinary day Bob has a .7 probability of being in his office
- If Bob is busy working on a paper deadline he has a .9 probability of being in his office
- If he is sick and there is no paper deadline he has a .1 probability of being in his office
- If he is sick but there is a paper deadline he has a .6 probability of being in his office
- John does not know if Bob has a paper deadline of if he is sick
  - He only knows if Bob is in his office

Hidden Markov Model: Example

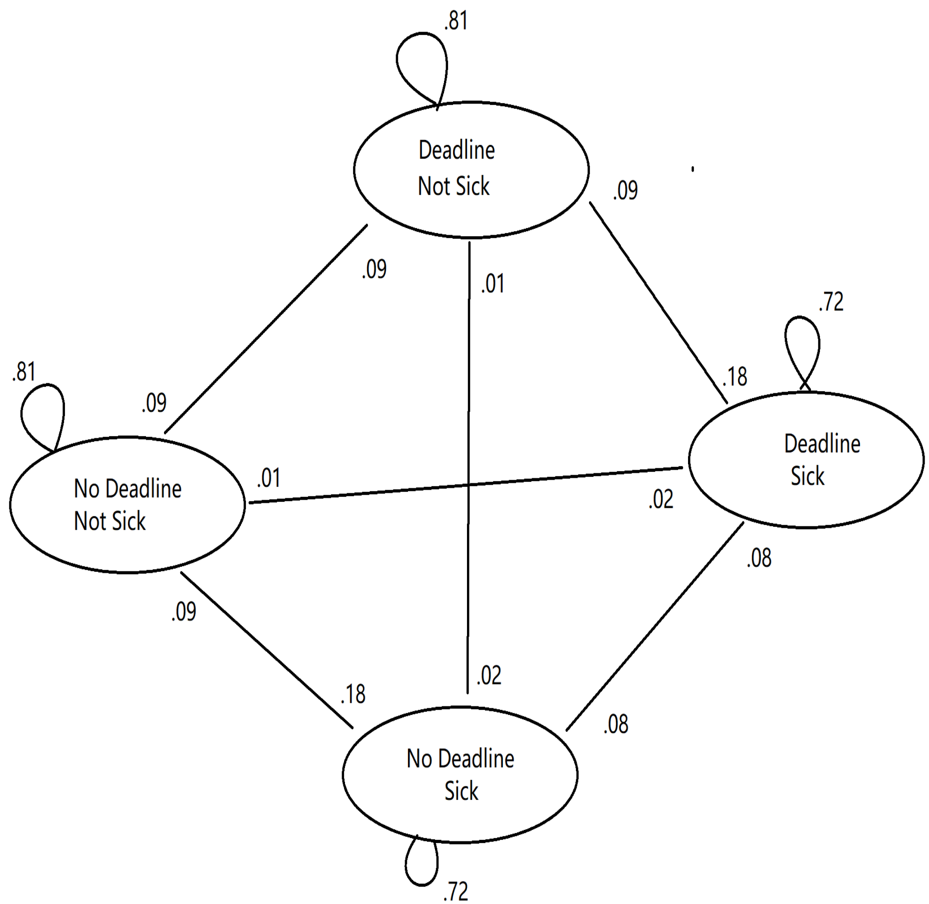
- Sample 1
  - Initial settings
    - Roll the dice
    - Get bob is sick and has no deadline
  - Step 1
    - Roll dice and get Bob not in office
    - Roll dice and transition to sick and no deadline
  - Step 2
    - Roll dice and get Bob in office
    - Roll dice and transition to not sick and no deadline
  - Step 3
    - Roll dice and get Bob in office
    - Roll dice and transition to not sick and no deadline



	Not Sick	Sick
Deadline	.9	.6
No Deadline	.7	.1

Hidden Markov Model: Example

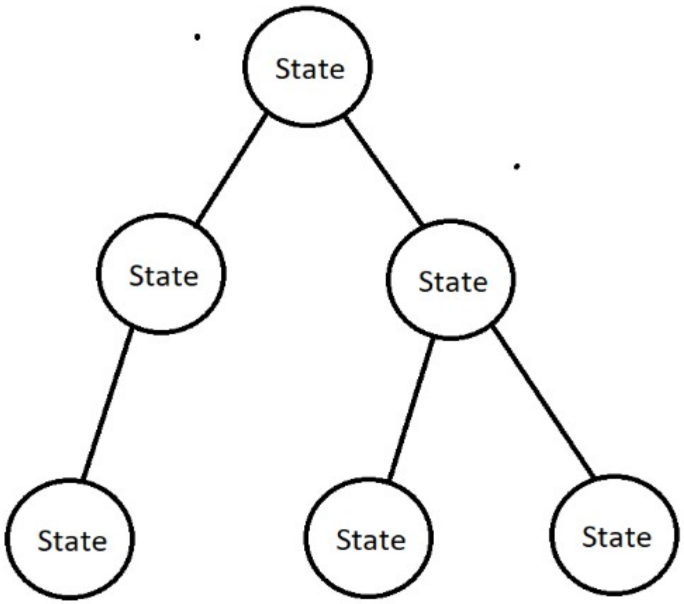
- Compute many samples
- Use to approximate probability of system
- Example: Compute probability Bob is sick if Bob is not in office for first 3 steps
  - Take all samples where Bob is not in office for first 3 steps
  - Compute portion of those samples where bob is sick
- Example: If Bob is in office for first three steps what is probability he will be in office at fourth step
  - Take all samples where Bob is in office for first 3 steps
  - Compute portion of those samples where Bob is in office for fourth step



	Not Sick	Sick
Deadline	.9	.6
No Deadline	.7	.1

Sample Trees

- Each Sample can be seen as a sequence of states
  - $s_0, s_1, s_2, \ldots$
- Sample tree
  - Nodes correspond to states
  - Nodes at each level correspond to state at that step
    - Nodes at ith level correspond to states at ith step
  - Samples correspond to paths in tree
    - Path formed by sequence of states
    - Samples with same sequence correspond to same path
    - Paths for samples diverge at node corresponding to first state where sequences differ
  - Edge weights = number of samples that pass through edge
    - Weight of edges out of state proportional to transition probability



Application to MDPs/POMDPs

- Can sample transitions from probability distribution
- For a given sequence of actions we can compute sample
- Want to know expected future reward
  - States' value equal to expected value of best action
  - Actions' value equal to weighted average of possible resulting states
    - $\sum$ value(s')*p(s'|a,s)

State/Action Trees

- Each Sample can be seen as a sequence of alternating states and actions
  - $s_0, a_0\ s_1, a_1, s_2, a_2, \ldots$
- Sample tree
  - Nodes correspond to states
  - Nodes at each level correspond to state at that step
    - Nodes at level 2i correspond to states at ith step
    - Nodes at level 2i+1corospond to action at ith step
  - Compute value of states by backtracking
    - Action nodes: Value = weighted sum of children + any reward for taking that action, R(a)
      - Weighted by number of samples that go through edge to child
    - State nodes: Value = max value over children + any reward associated with state, R(s)
      - Action with highest value