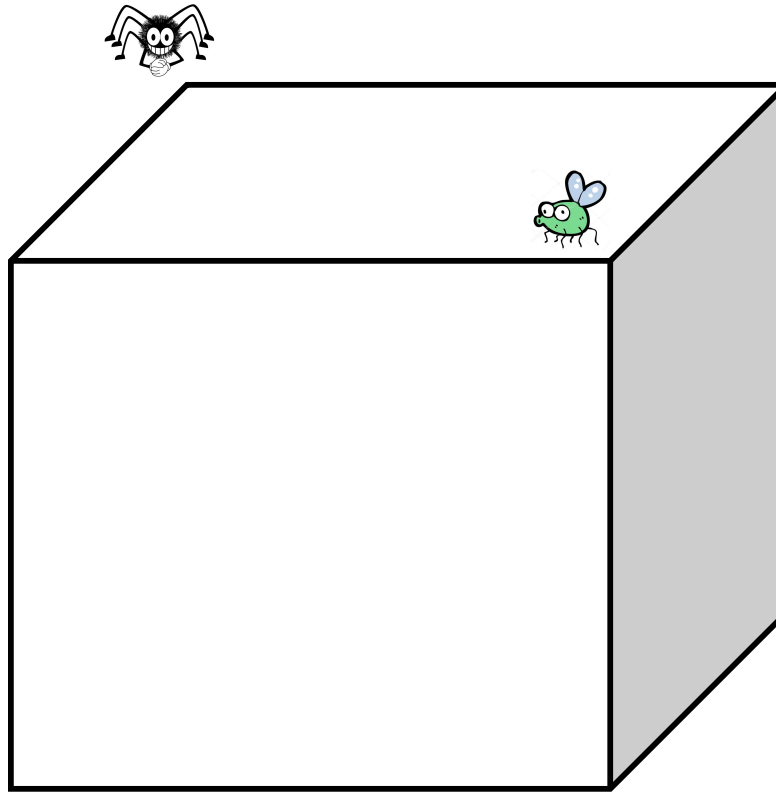


# Machine Learning for Signal Processing

## Hidden Markov Models

Bhiksha Raj

# A quick intro to Markov Chains..




- The case of flider and spy..

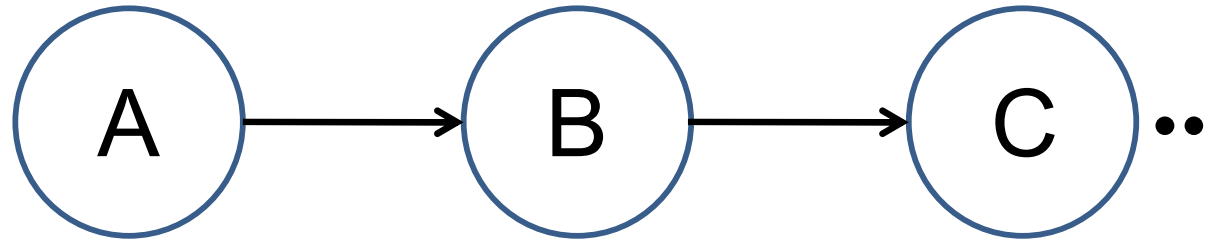
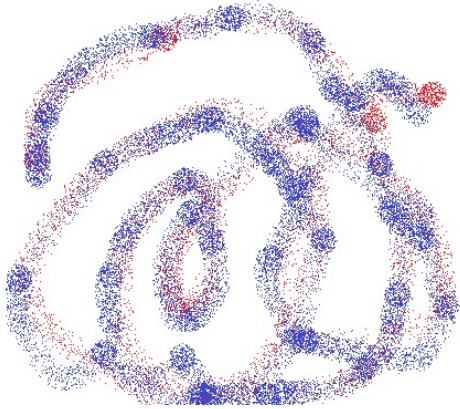
# Prediction : a holy grail

- Physical trajectories
  - Automobiles, rockets, heavenly bodies
- Natural phenomena
  - Weather
- Financial data
  - Stock market
- World affairs
  - Who is going to win the next election?
- Signals
  - Audio, video..

# The wind and the target

- Aim: measure wind velocity accurately
    - For some important task
  - Using a noisy wind speed sensor
    - E.g. arrows shot at a target
- 
- Situation:
    - Wind speed at time  $t$  depends on speed at  $t-1$ 
      - $S_t = S_{t-1} + \epsilon_t$
    - Arrow position at time  $t$  depends on wind speed at time  $t$ 
      - $Y_t = AS_t + \gamma_t$
  - Challenge: Given sequence of observation  $Y_1, Y_2, \dots, Y_t$ 
    - Estimate current wind speed  $S_t$
    - Predict wind speed and arrow position at  $t + 1$ :  $S_{t+1}$  and  $Y_{t+1}$

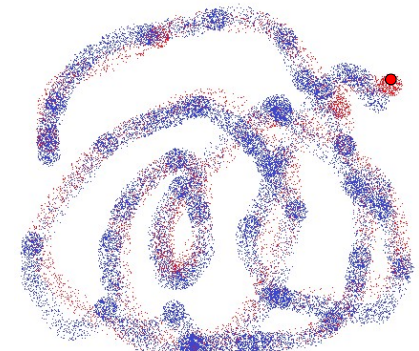
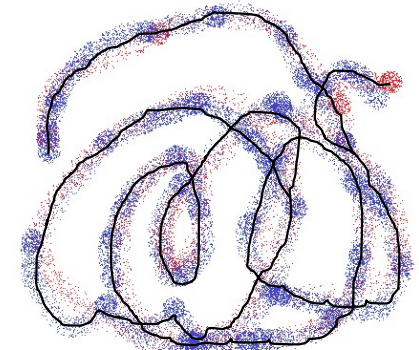
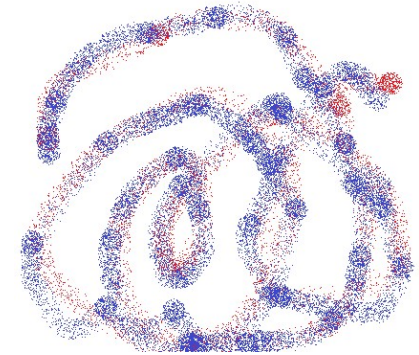
# A Common Trait



- *Series data with trends*
- Stochastic functions of stochastic functions (of stochastic functions of ...)
- An underlying process that progresses (seemingly) randomly
  - E.g. wind speed
  - E.g. Current position of a vehicle
  - E.g. current sentiment in stock market
- Random expressions of underlying process
  - E.g. Wind speed sensor measurement
  - E.g. what you see from the vehicle
  - E.g. current stock prices of various stock

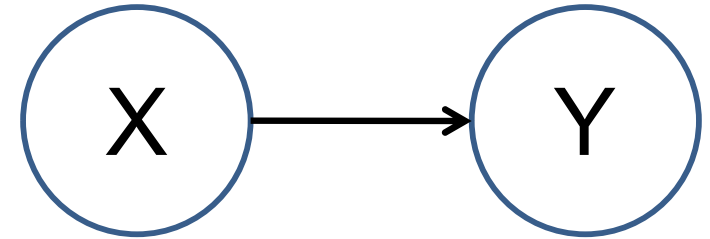
# What a sensible agent must do

- *Learn* about the process
  - From whatever they know
    - E.g. learn the wind-speed function and the arrow-to-wind function
  - Basic requirement for other procedures
- *Track* underlying processes
  - Track the wind speed
- Predict future values



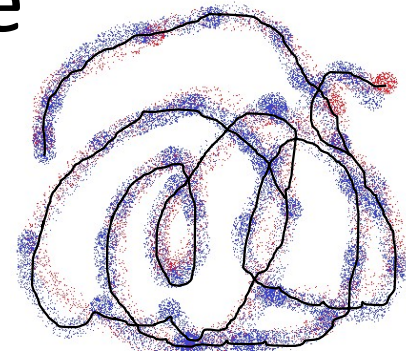
# A Specific Form of Process..

- Doubly stochastic processes



- One random process generates a “state” variable  $X$

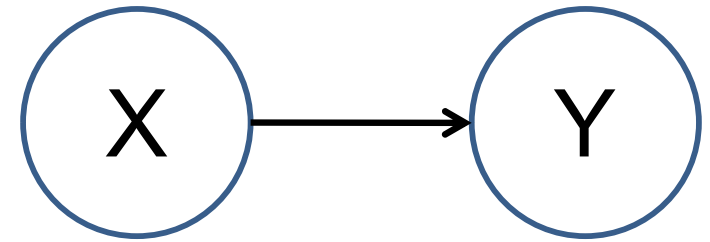
– Random process  $X \rightarrow P(X; \Theta)$



- Second-level process generates observations as a function of state  $X$
- Random process  $Y \rightarrow P(Y; f(X, \Lambda))$

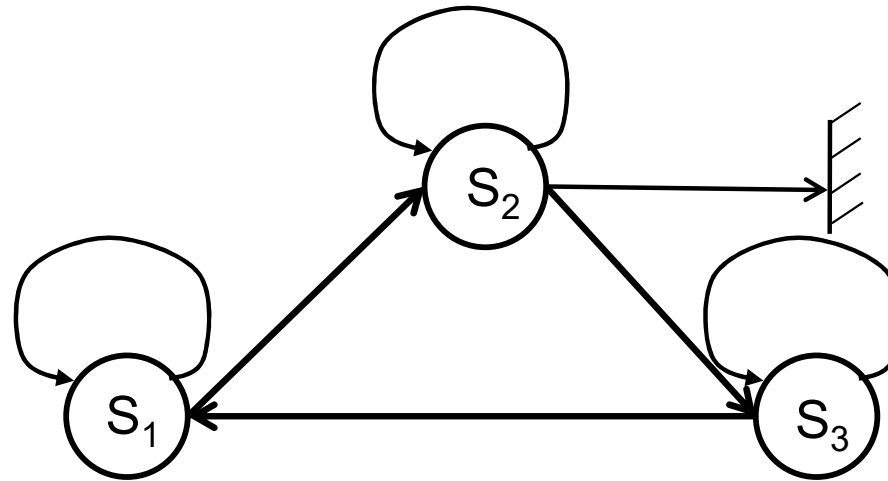
# Doubly Stochastic Process *Model*

- Doubly stochastic processes are *models*
  - May not be a *true* representation of process underlying actual data
- First level variable may be a *quantifiable* variable
  - Position/state of vehicle
  - Second level variable is a stochastic function of position
- First level variable may *not* have meaning
  - “Sentiment” of a stock market
  - “Configuration” of vocal tract





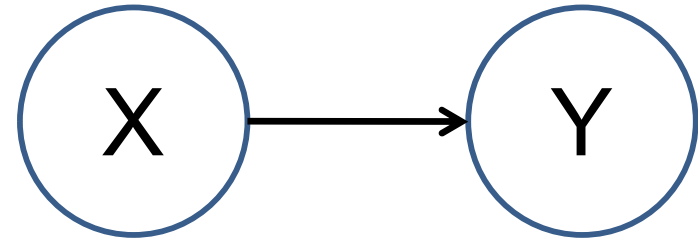
# Markov Chain



- Process can go through a number of states
  - Random walk, Brownian motion..
- From each state, it can go to any other state with a probability
  - Which only depends on the current state
- Walk goes on forever
  - Or until it hits an “absorbing wall”
- Output of the process – a sequence of states the process went through

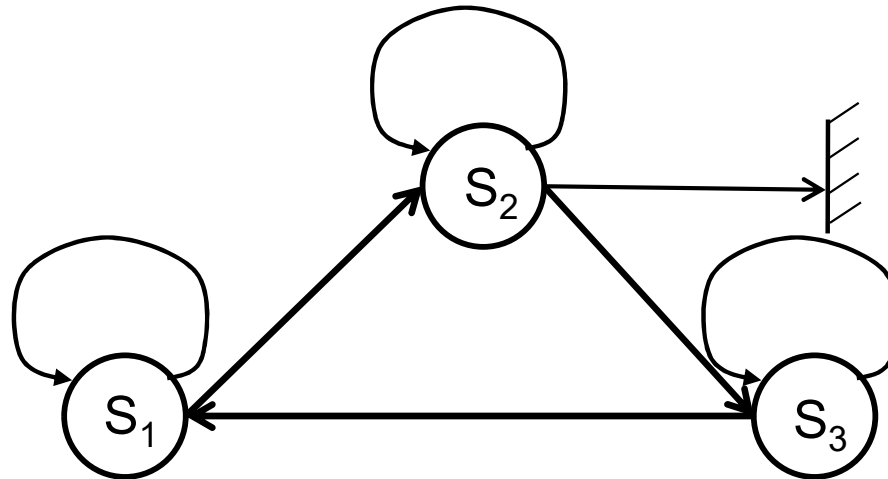
# Stochastic Function of a Markov Chain

- First-level variable is *usually* abstract



- The first level variable assumed to be the output of a Markov Chain
- The second level variable is a random variable whose distribution is a function of the output of the Markov Chain
- Also called an HMM
- Another variant – stochastic function of Markov *process*
  - *Kalman Filtering..*

# Stochastic Function of a Markov Chain



- Output:
  - $Y == Y_1 Y_2 \dots$
  - $Y_i \sim P(Y_i ; f(s_i))$ 
    - Probability distribution is a function of the state

# Poll 1

# A little parable

You've been kidnapped



# A little parable

You've been kidnapped



And blindfolded

# A little parable

You've been kidnapped



And blindfolded

You can only *hear* the car

You must find your way back home from wherever they drop you off

# Kidnapped



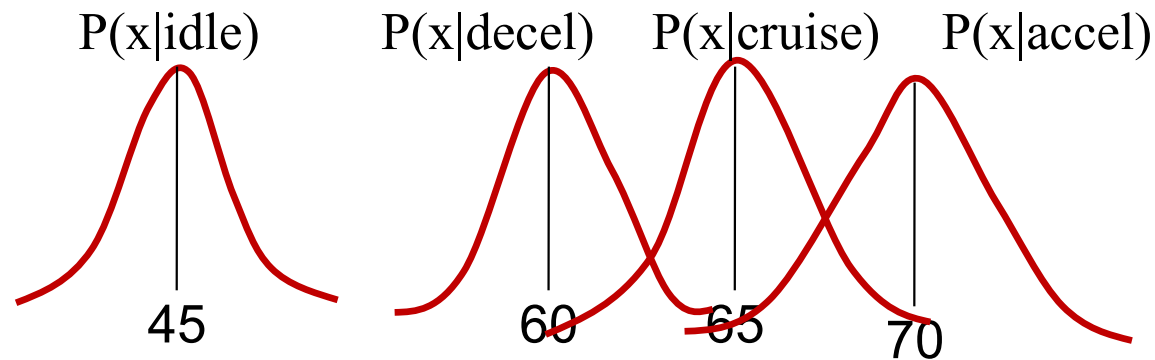
- Determine automatically, by only *listening* to a running automobile, if it is:
  - Idling; or
  - Travelling at constant velocity; or
  - Accelerating; or
  - Decelerating
- You are super acoustically sensitive and can determine sound pressure level (SPL)
  - The SPL is measured once per second



# What you know

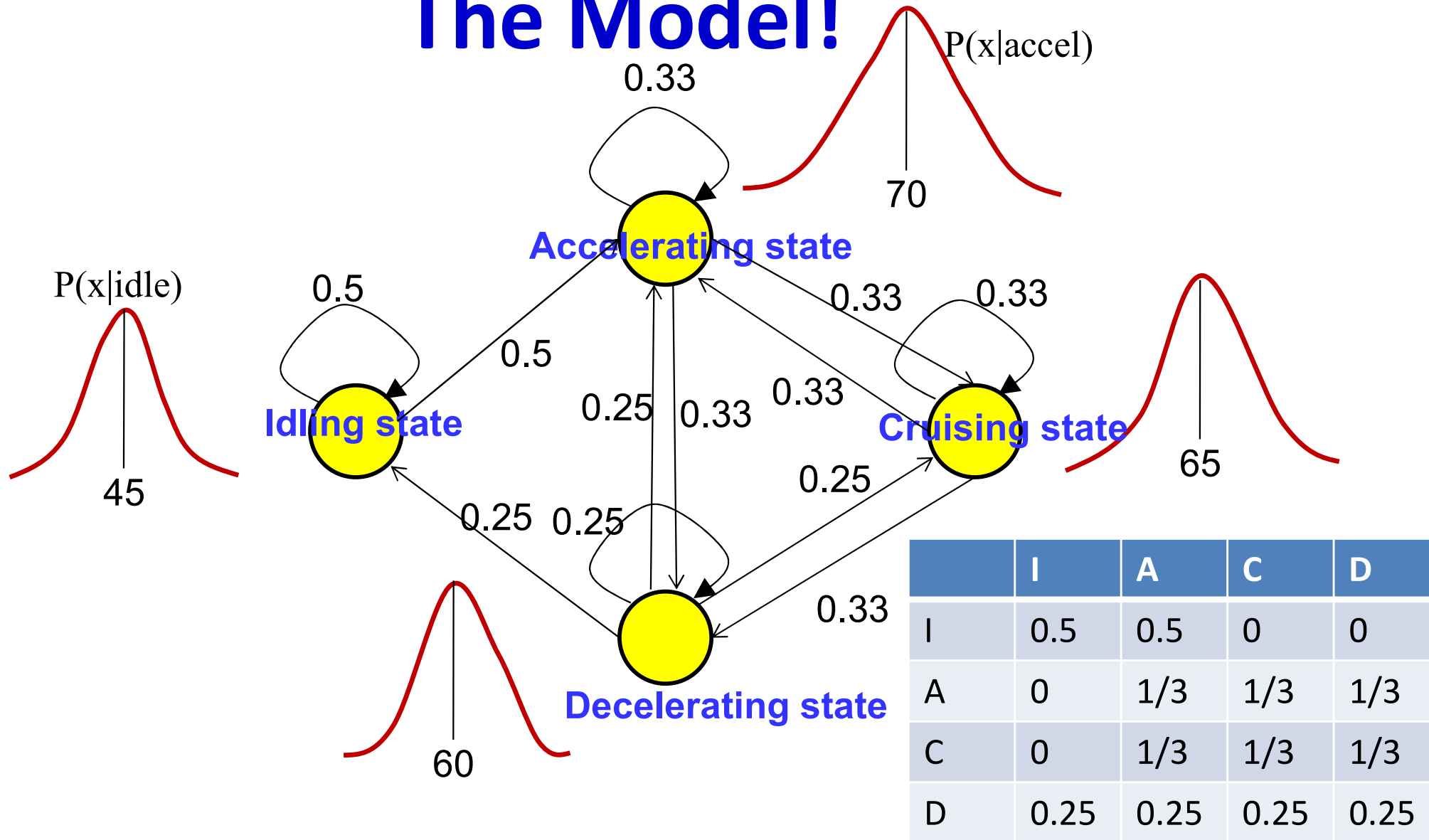
- An automobile that is at rest can accelerate, or continue to stay at rest
- An accelerating automobile can hit a steady-state velocity, continue to accelerate, or decelerate
- A decelerating automobile can continue to decelerate, come to rest, cruise, or accelerate
- An automobile at a steady-state velocity can stay in steady state, accelerate or decelerate

# What else you know



- The probability distribution of the SPL of the sound is different in the various conditions
  - As shown in figure
    - In reality, depends on the car
- The distributions for the different conditions overlap
  - Simply knowing the current sound level is not enough to know the state of the car

# The Model!

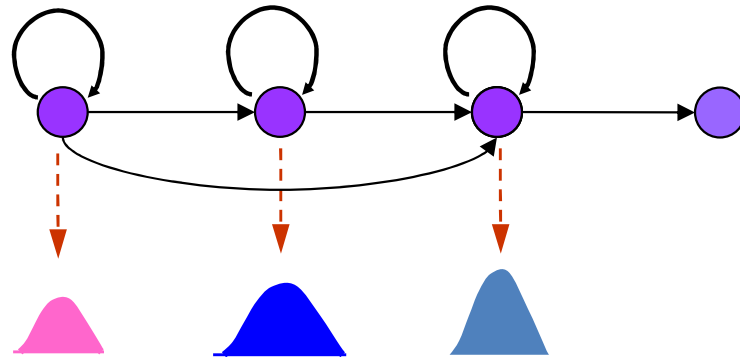


- The state-space model
  - Assuming all transitions from a state are equally probable
  - We will help you find your way back home in the next class

# What is an HMM

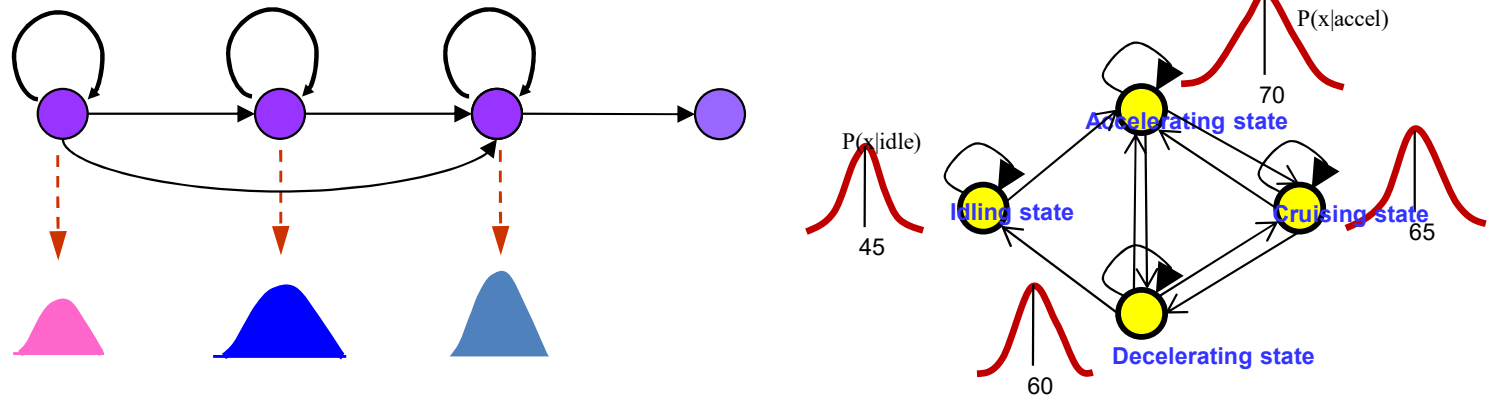
- The model assumes that the process can be in one of a number of states at any time instant
- The state of the process at any time instant depends only on the state at the previous instant (causality, Markovian)
- At each instant the process generates an observation from a probability distribution that is specific to the current state
- The generated observations are all that we get to see
  - the actual state of the process is not directly observable
    - Hence the qualifier hidden

# What is an HMM

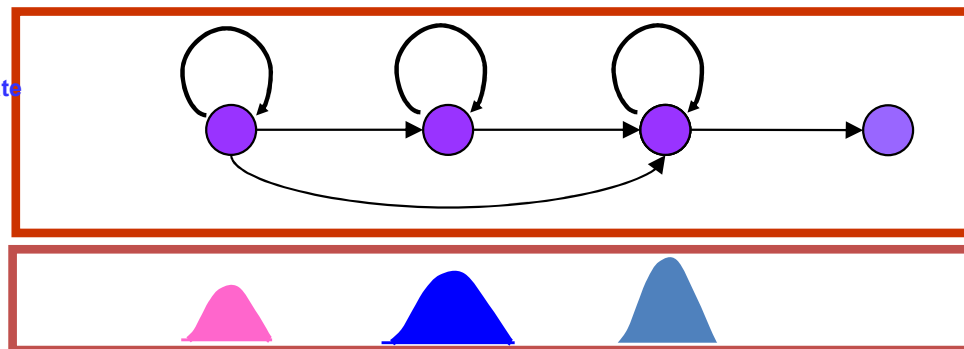
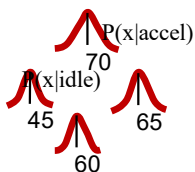
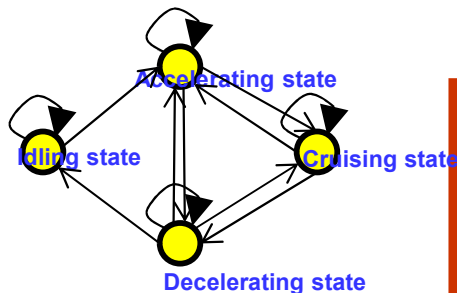


- “Probabilistic function of a markov chain”
- Models a dynamical system
- System goes through a number of states
  - Following a Markov chain model
- On arriving at any state it generates observations according to a state-specific probability distribution

# Hidden Markov Models



- A Hidden Markov Model consists of two components
  - A state/transition backbone that specifies how many states there are, and how they can follow one another
  - A set of probability distributions, one for each state, which specifies the distribution of all vectors in that state

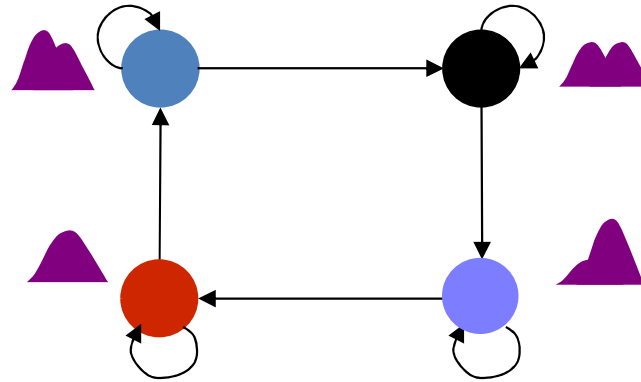


Markov chain

Data distributions

# How an HMM models a process

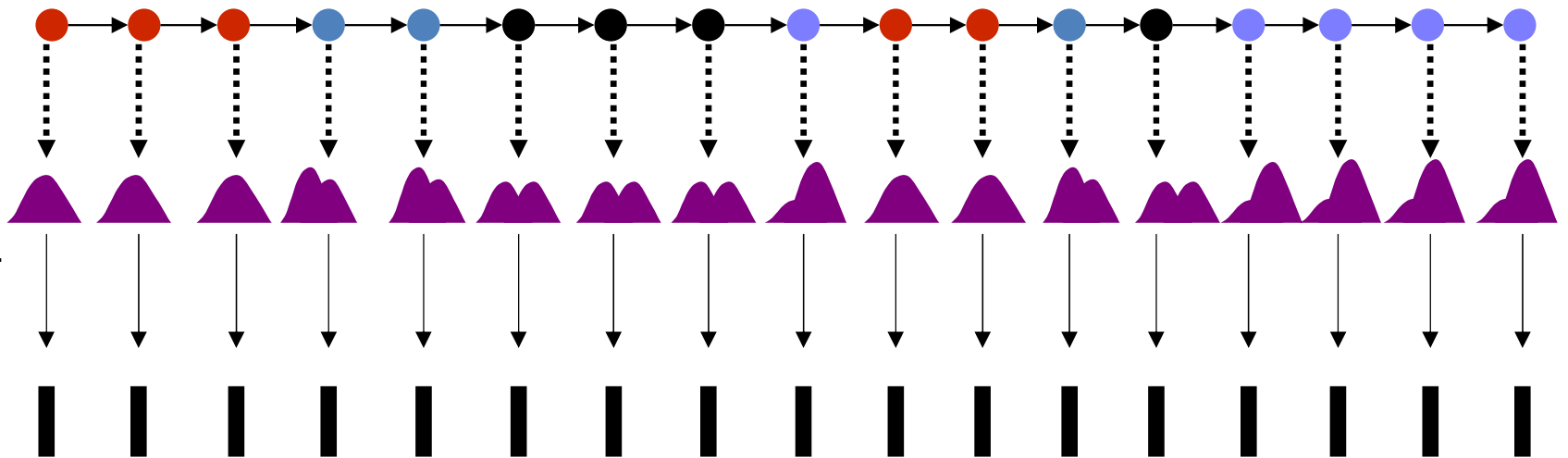
HMM assumed to be  
generating data



state  
sequence

state  
distributions

observation  
sequence

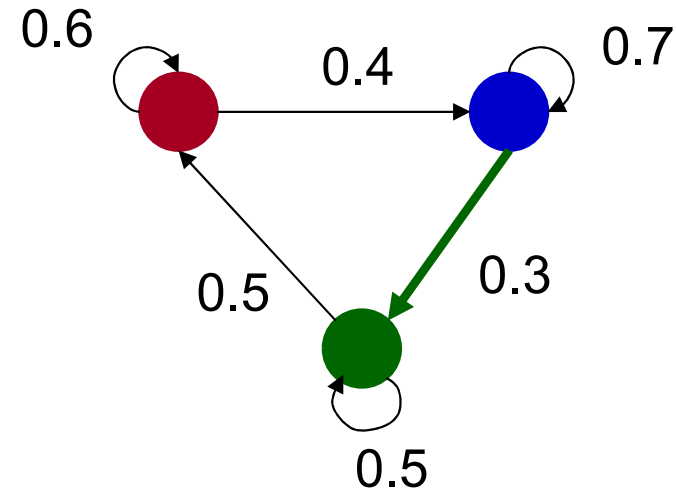


# Poll 2

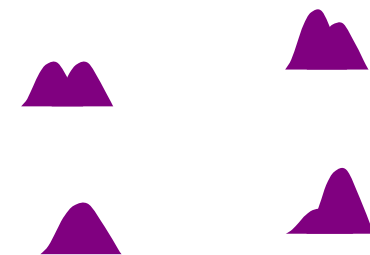


# HMM Parameters

- The *topology* of the HMM
  - Number of states and allowed transitions
  - E.g. here we have 3 states and cannot go from the blue state to the red
- The transition probabilities
  - Often represented as a matrix as here
  - $T_{ij}$  is the probability that when in state  $i$ , the process will move to  $j$
- The probability  $\pi_i$  of beginning at any state  $s_i$ 
  - The complete set is represented as  $\pi$
- The *state output distributions*



$$T = \begin{pmatrix} .6 & .4 & 0 \\ 0 & .7 & .3 \\ .5 & 0 & .5 \end{pmatrix}$$



# HMM state output distributions

- The state output distribution is the distribution of data produced from any state
- Typically modelled as Gaussian

$$P(x | s_i) = \text{Gaussian}(x; \mu_i, \Theta_i) = \frac{1}{\sqrt{(2\pi)^d |\Theta_i|}} e^{-0.5(x - \mu_i)^T \Theta_i^{-1} (x - \mu_i)}$$

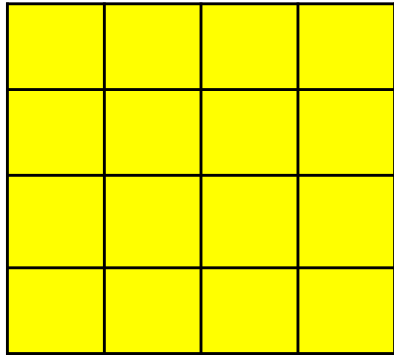
- The parameters are  $\mu_i$  and  $\Theta_i$
- More typically, modelled as Gaussian mixtures

$$P(x | s_i) = \sum_{j=0}^{K-1} w_{i,j} \text{Gaussian}(x; \mu_{i,j}, \Theta_{i,j})$$

- Other distributions may also be used
- E.g. histograms for discrete observations

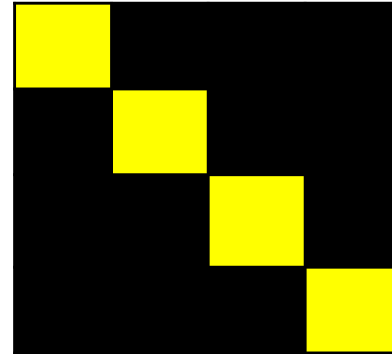
# The Diagonal Covariance Matrix

Full covariance:  
all elements are  
non-zero



$$-0.5(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Theta}^{-1}(\mathbf{x}-\boldsymbol{\mu})$$

Diagonal covariance:  
off-diagonal elements  
are zero



$$-\sum_i (x_i - \mu_i)^2 / 2\sigma_i^2$$

- For GMMs it is frequently assumed that the feature vector dimensions are all *independent* of each other
- *Result:* The covariance matrix is reduced to a diagonal form
  - The determinant of the diagonal  $\boldsymbol{\Theta}$  matrix is easy to compute

# Three Basic HMM Problems

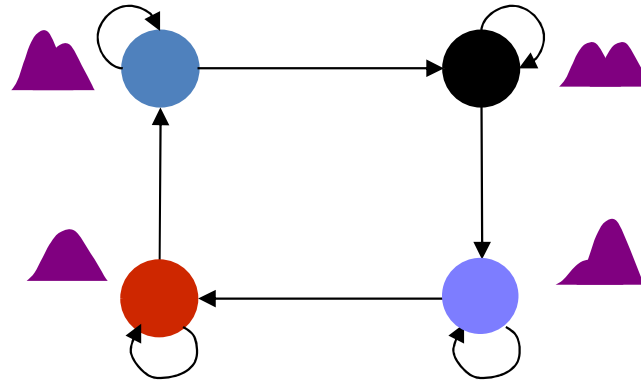
- What is the probability that it will generate a specific observation sequence
- Given an observation sequence, how do we determine which observation was generated from which state
  - The state segmentation problem
- How do we *learn* the parameters of the HMM from observation sequences

# Computing the Probability of an Observation Sequence

- Two aspects to producing the observation:
  - Progressing through a sequence of states
  - Producing observations from these states

# Progressing through states

HMM assumed to be  
generating data



state  
sequence



- The process begins at some state (red) here
- From that state, it makes an allowed transition
  - To arrive at the same or any other state
- From that state it makes another allowed transition
  - And so on

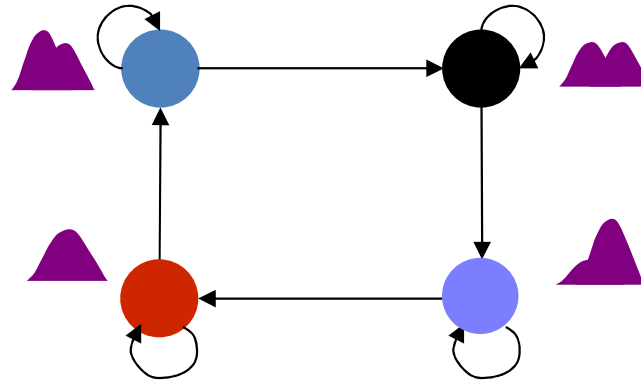
# Probability that the HMM will follow a particular state sequence

$$P(s_1, s_2, s_3, \dots) = P(s_1)P(s_2|s_1)P(s_3|s_2)\dots$$

- $P(s_1)$  is the probability that the process will initially be in state  $s_1$
- $P(s_i / s_i)$  is the transition probability of moving to state  $s_i$  at the next time instant when the system is currently in  $s_i$ 
  - Also denoted by  $T_{ij}$  earlier

# Generating Observations from States

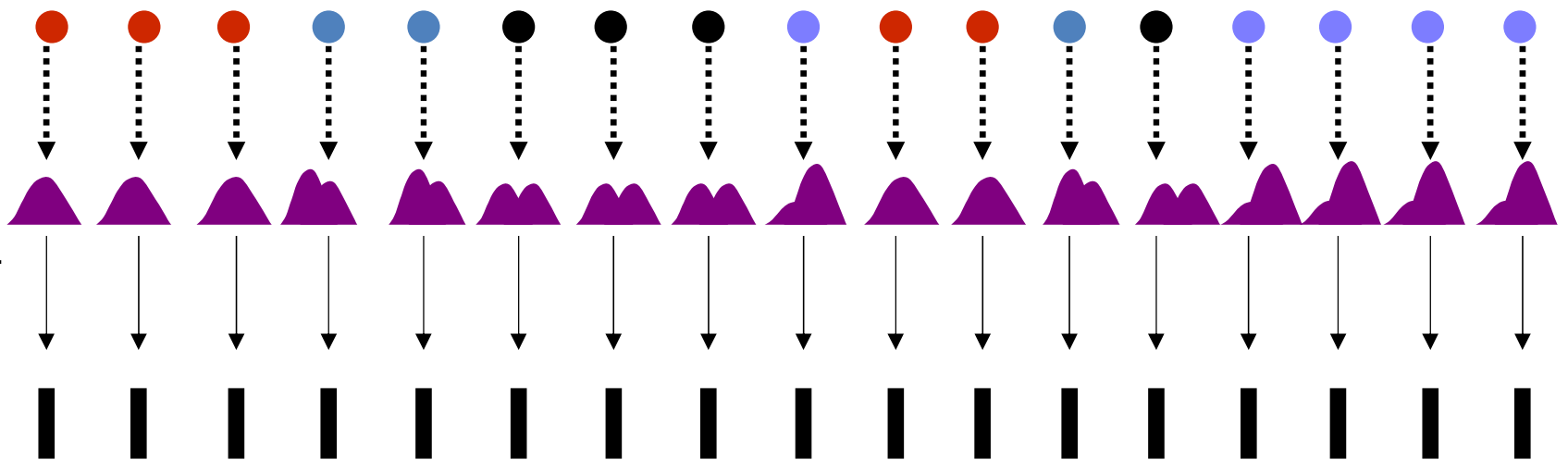
HMM assumed to be  
generating data



state  
sequence

state  
distributions

observation  
sequence



- At each time it generates an observation from the state it is in at that time



# Probability that the HMM will generate a particular observation sequence given a state sequence (state sequence known)

$$P(o_1, o_2, o_3, \dots | s_1, s_2, s_3, \dots) = P(o_1 | s_1) P(o_2 | s_2) P(o_3 | s_3) \dots$$

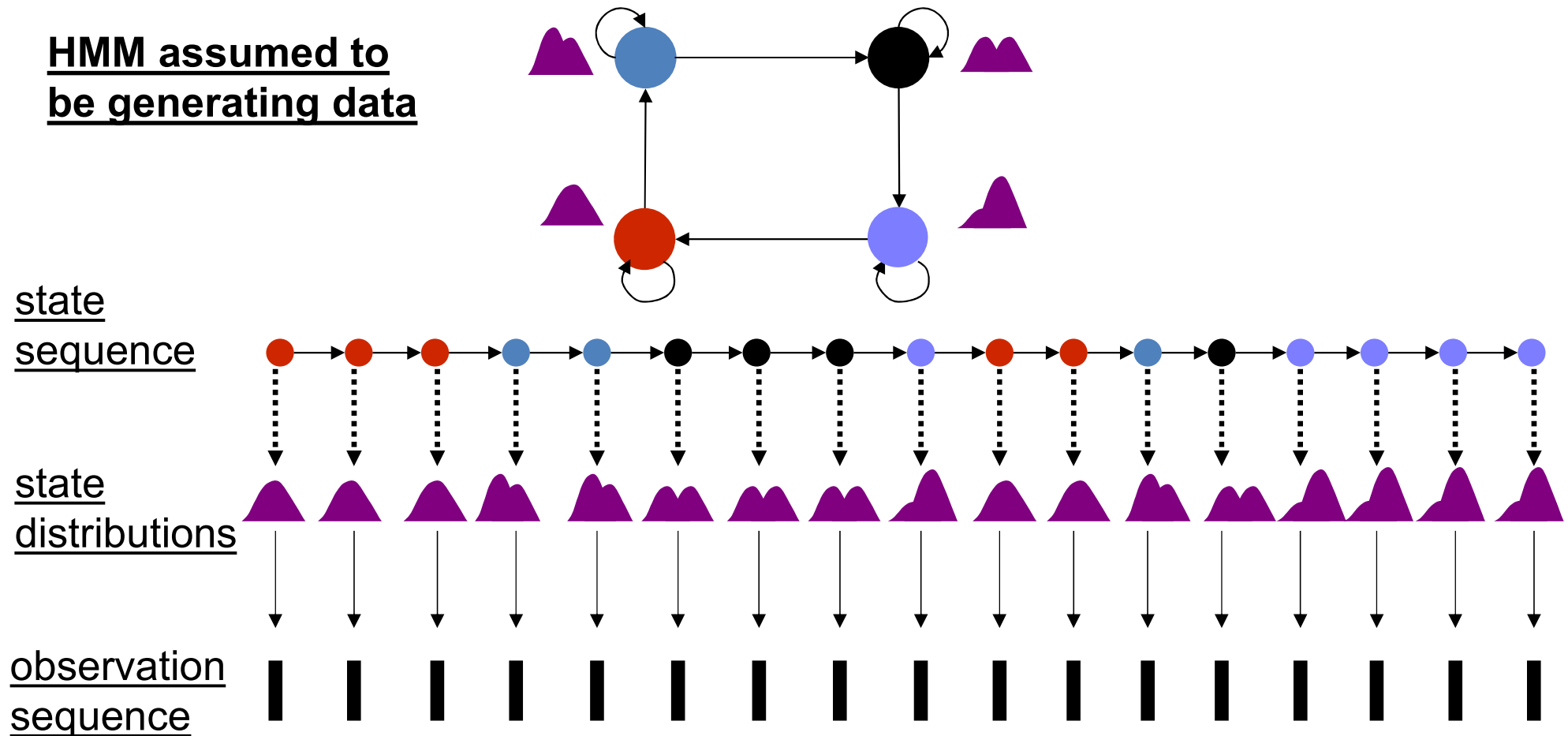


Computed from the Gaussian or Gaussian mixture for state  $s_1$

- $P(o_i | s_i)$  is the probability of generating observation  $o_i$  when the system is in state  $s_i$

# Proceeding through States and Producing Observations

HMM assumed to  
be generating data



- At each time it produces an observation and makes a transition

**Probability that the HMM will generate  
a particular state sequence and from it,  
a particular observation sequence**

$$P(o_1, o_2, o_3, \dots, s_1, s_2, s_3, \dots) =$$

$$P(o_1, o_2, o_3, \dots | s_1, s_2, s_3, \dots) P(s_1, s_2, s_3, \dots) =$$

$$P(o_1 | s_1) P(o_2 | s_2) P(o_3 | s_3) \dots P(s_1) P(s_2 | s_1) P(s_3 | s_2) \dots$$

# Probability of Generating an Observation Sequence

- The precise state sequence is not known
- All possible state sequences must be considered

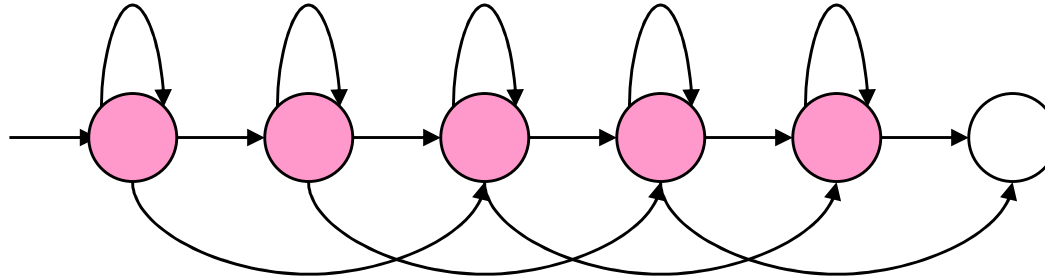
$$P(o_1, o_2, o_3, \dots) = \sum_{\substack{\text{all possible} \\ \text{state sequences}}} P(o_1, o_2, o_3, \dots, s_1, s_2, s_3, \dots) =$$

$$\sum_{\substack{\text{all possible} \\ \text{state sequences}}} P(o_1|s_1)P(o_2|s_2)P(o_3|s_3)\dots P(s_1)P(s_2|s_1)P(s_3|s_2)\dots$$

# Computing it Efficiently

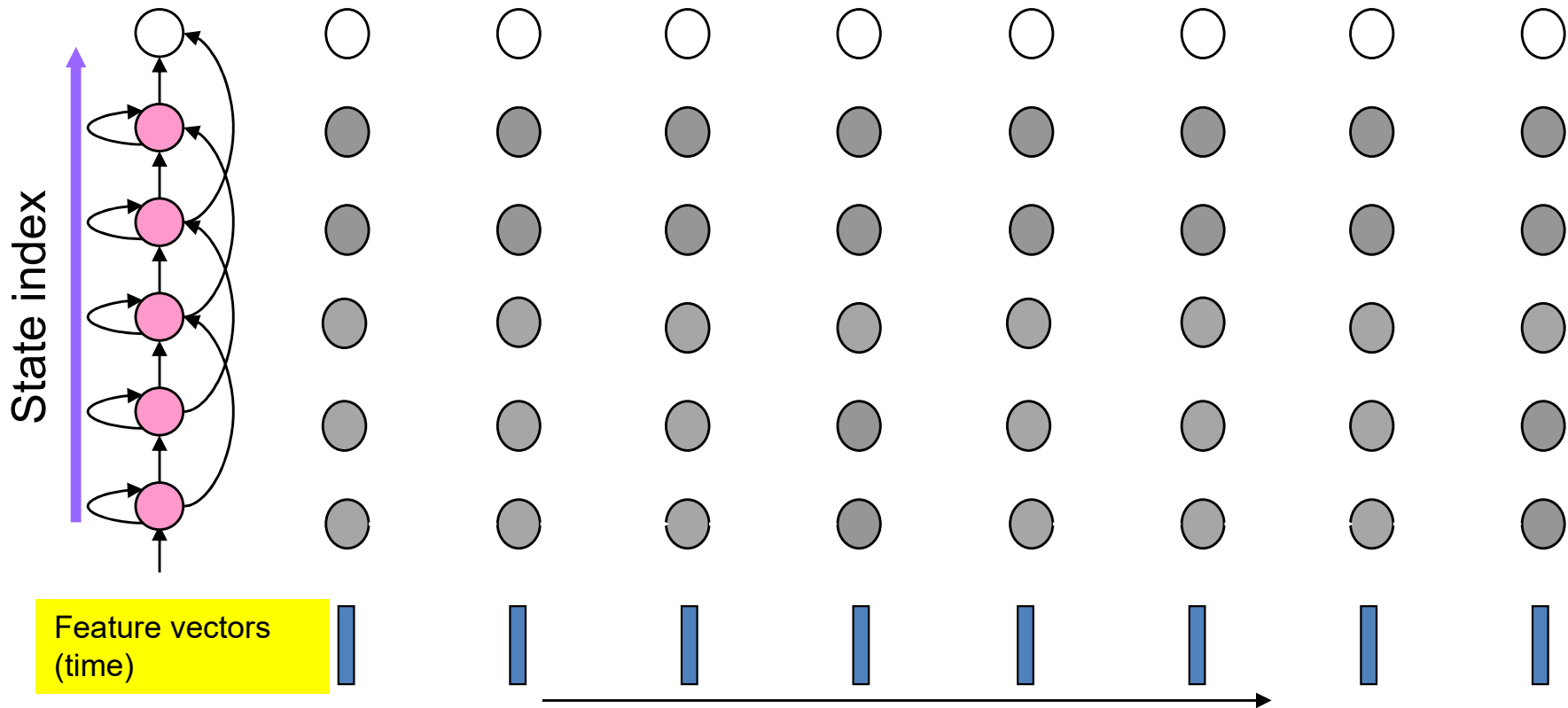
- Explicit summing over all state sequences is not tractable
  - A very large number of possible state sequences
- Instead we use the forward algorithm
- A dynamic programming technique.

# Illustrative Example



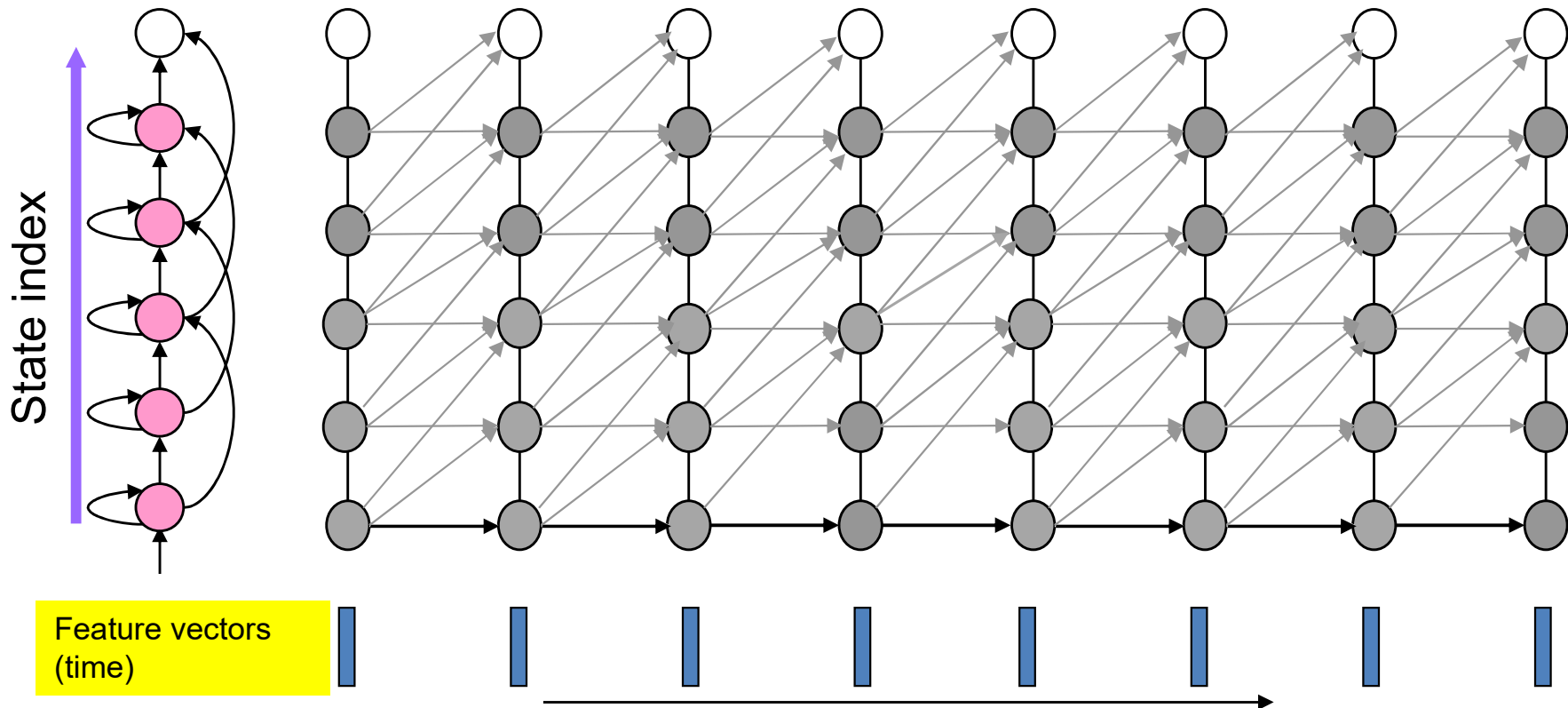
- Example: a generic HMM with 5 states and a “terminating state”.
  - Left to right topology
    - $P(s_i) = 1$  for state 1 and 0 for others
  - The arrows represent transition for which the probability is not 0

# States and times...



- The process can be at any of the 6 states at each time
- Every node represents the event of a particular observation being generated from a particular state

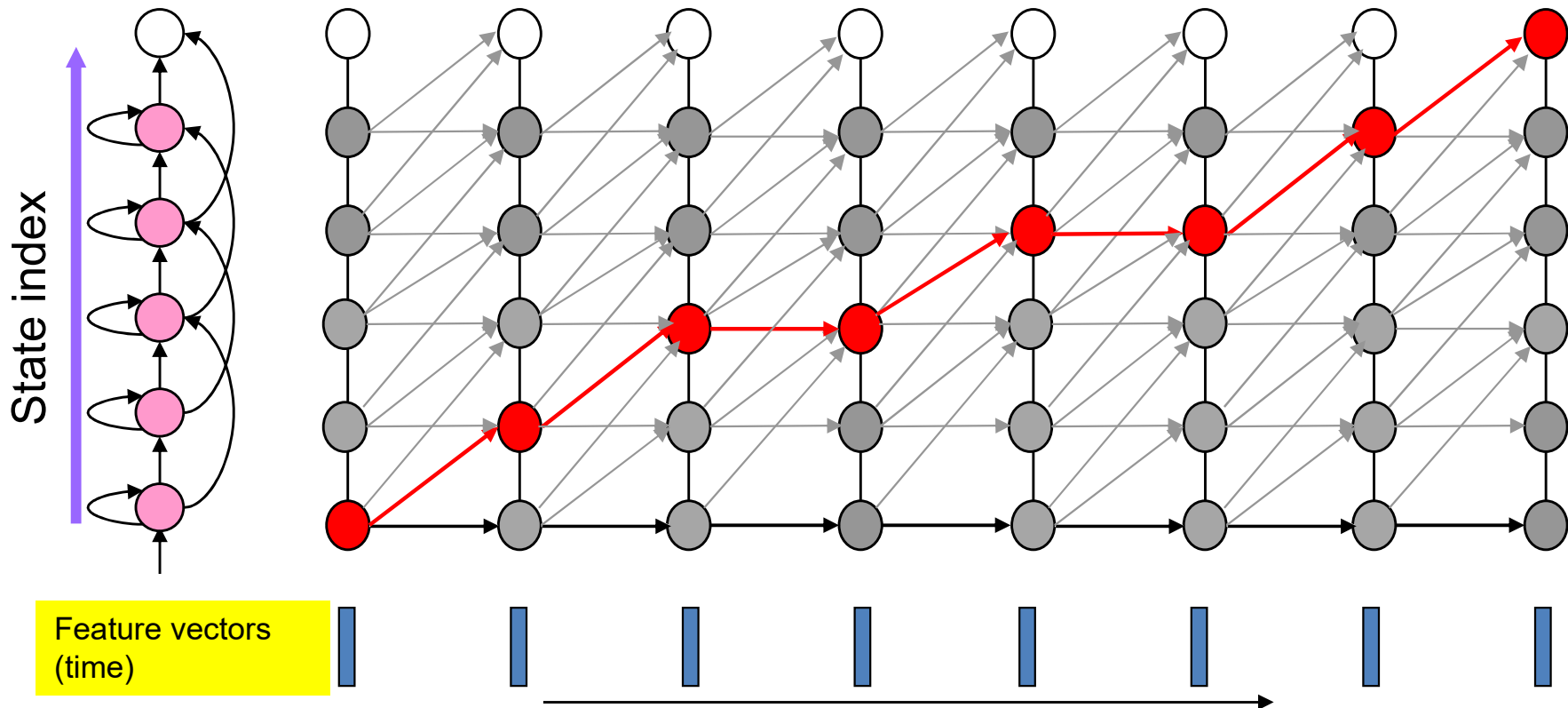
# The Trellis



- The trellis is a graphical representation of all possible paths through the HMM to produce a given observation
- The Y-axis represents HMM states, X axis represents observations
- Every edge in the graph represents a valid transition in the HMM over a single time step
  - Each edge carries the state transition probability between the source and destination states
- Every node represents the event of a particular observation being generated from a particular state
  - Each node for state  $s$  at time  $t$  carries the probability  $P(O_t | s)$

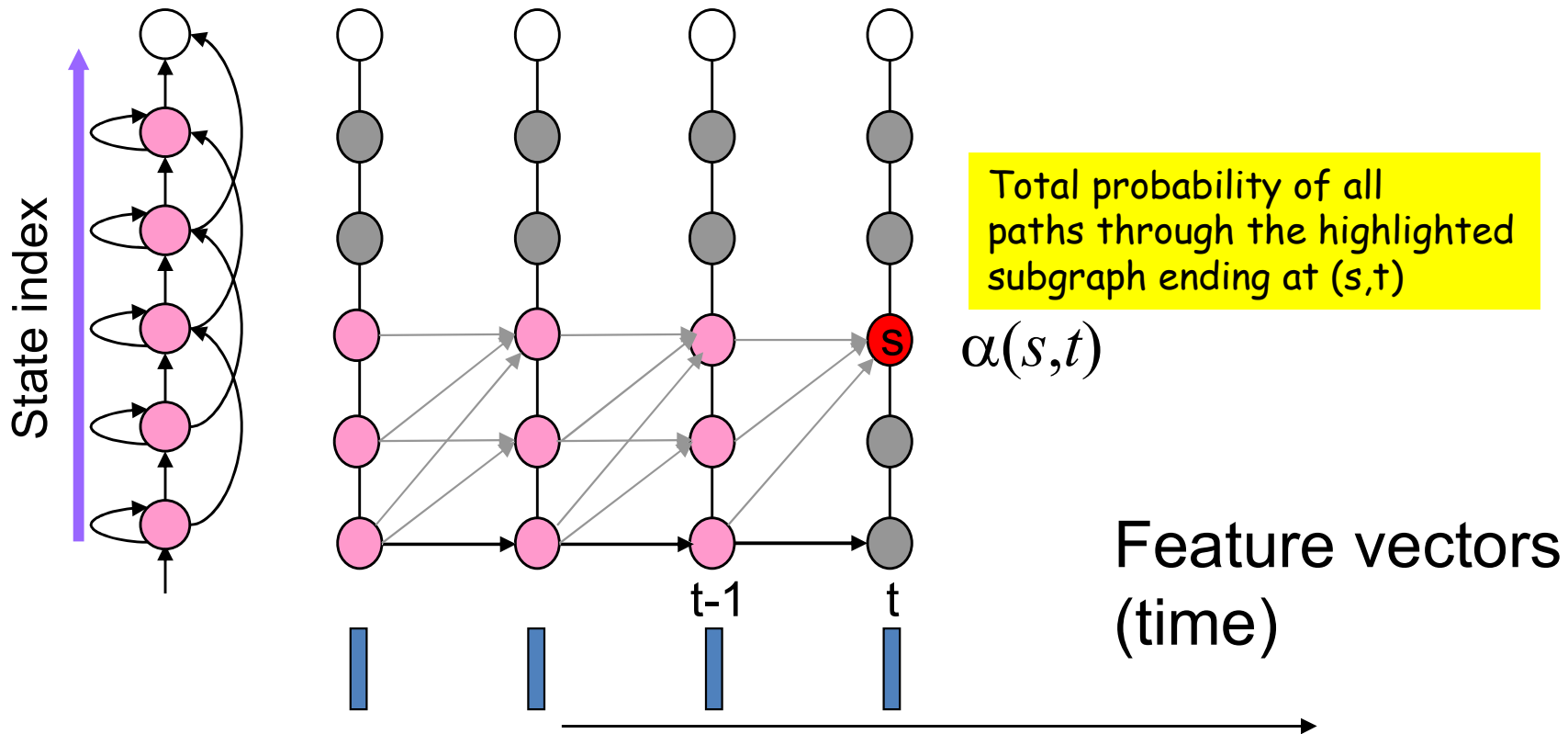


# The Trellis



- Any path through the trellis is a sequence of states that the processes has traversed in generating the observations
- The probability of the path is the product of all the edge and node probabilities on the path
  - $P(s_0)P(O_0|s_0) \prod_t P(s_t|s_{t-1})P(O_t|s_t)$

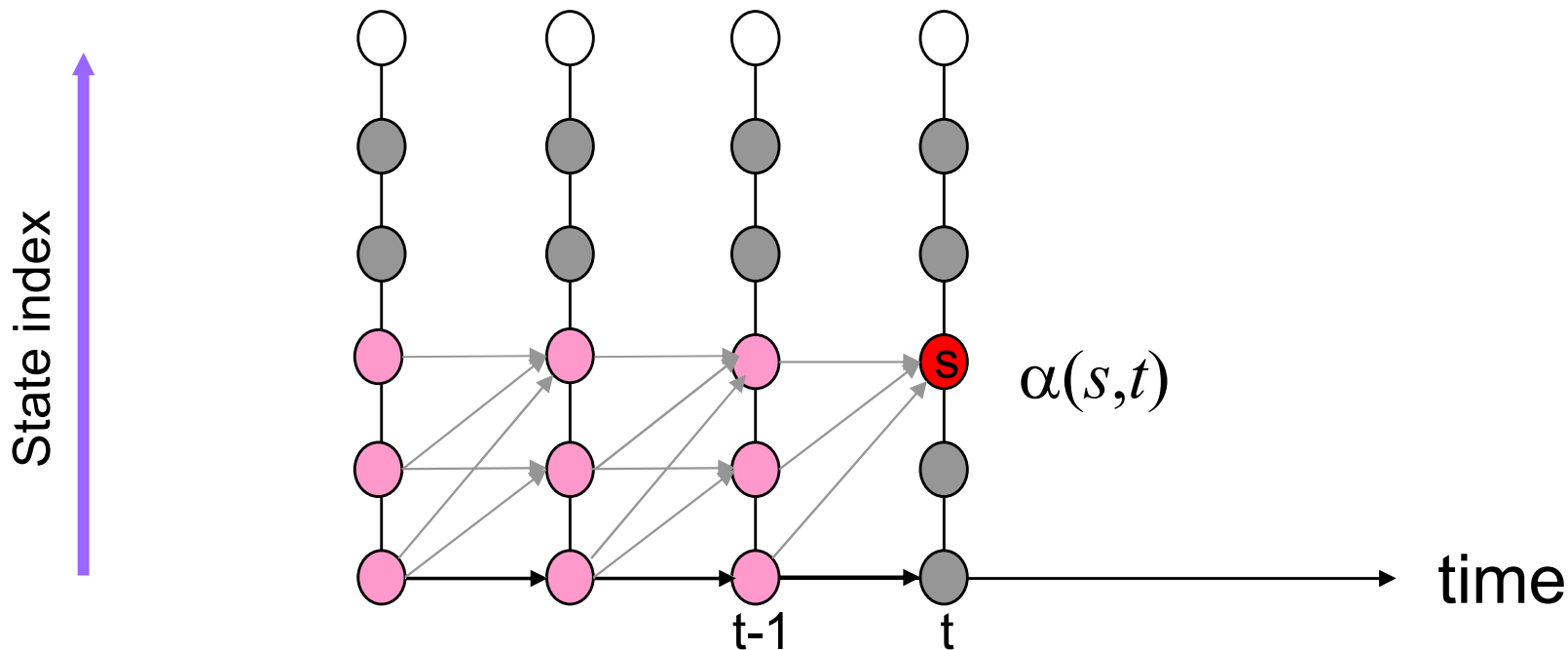
# Diversion: The Trellis



- The trellis is a graphical representation of all possible paths through the HMM to produce a given observation
- The Y-axis represents HMM states, X axis represents observations
- Every edge in the graph represents a valid transition in the HMM over a single time step
- Every node represents the event of a particular observation being generated from a particular state

# The Forward Algorithm

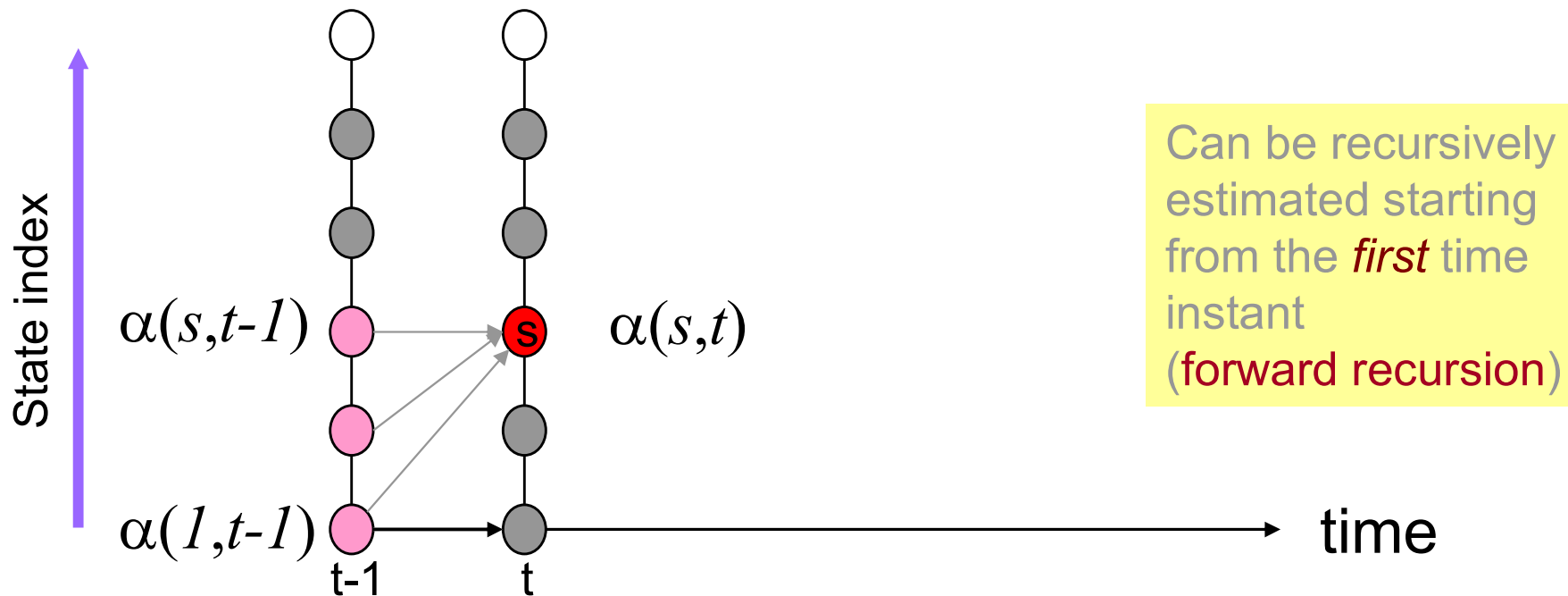
$$\alpha(s, t) = P(x_1, x_2, \dots, x_t, \text{state}(t) = s)$$



- $\alpha(s, t)$  is the total probability of ALL state sequences that end at state  $s$  at time  $t$ , and all observations until  $x_t$

# The Forward Algorithm

$$\alpha(s, t) = P(x_1, x_2, \dots, x_t, \text{state}(t) = s)$$

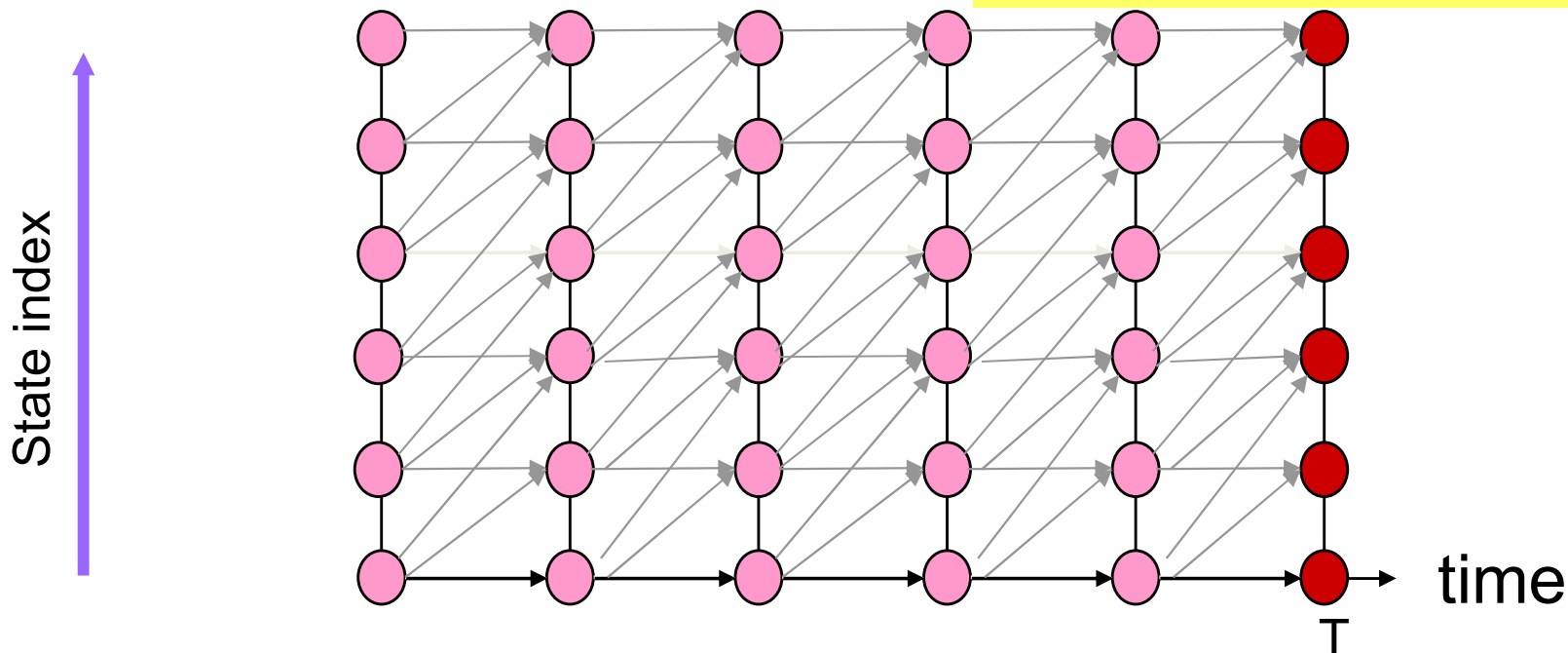


$$\alpha(s, t) = \sum_{s'} \alpha(s', t-1) P(s | s') P(x_t | s)$$

- $\alpha(s, t)$  can be recursively computed in terms of  $\alpha(s', t')$ , the forward probabilities at time t-1

# The Forward Algorithm

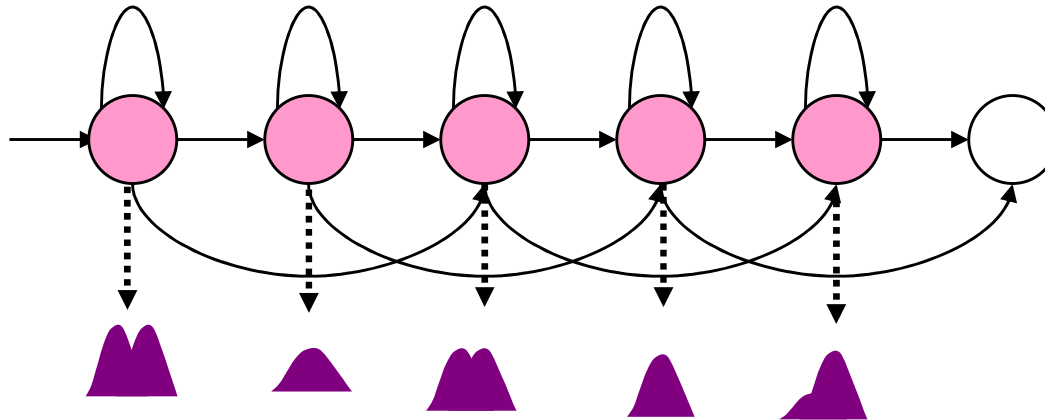
$$Totalprob = \sum_s \alpha(s, T)$$



- In the final observation the alpha at each state gives the probability of all state sequences ending at that state
- General model: The total probability of the observation is the sum of the alpha values at all states

# Poll 3

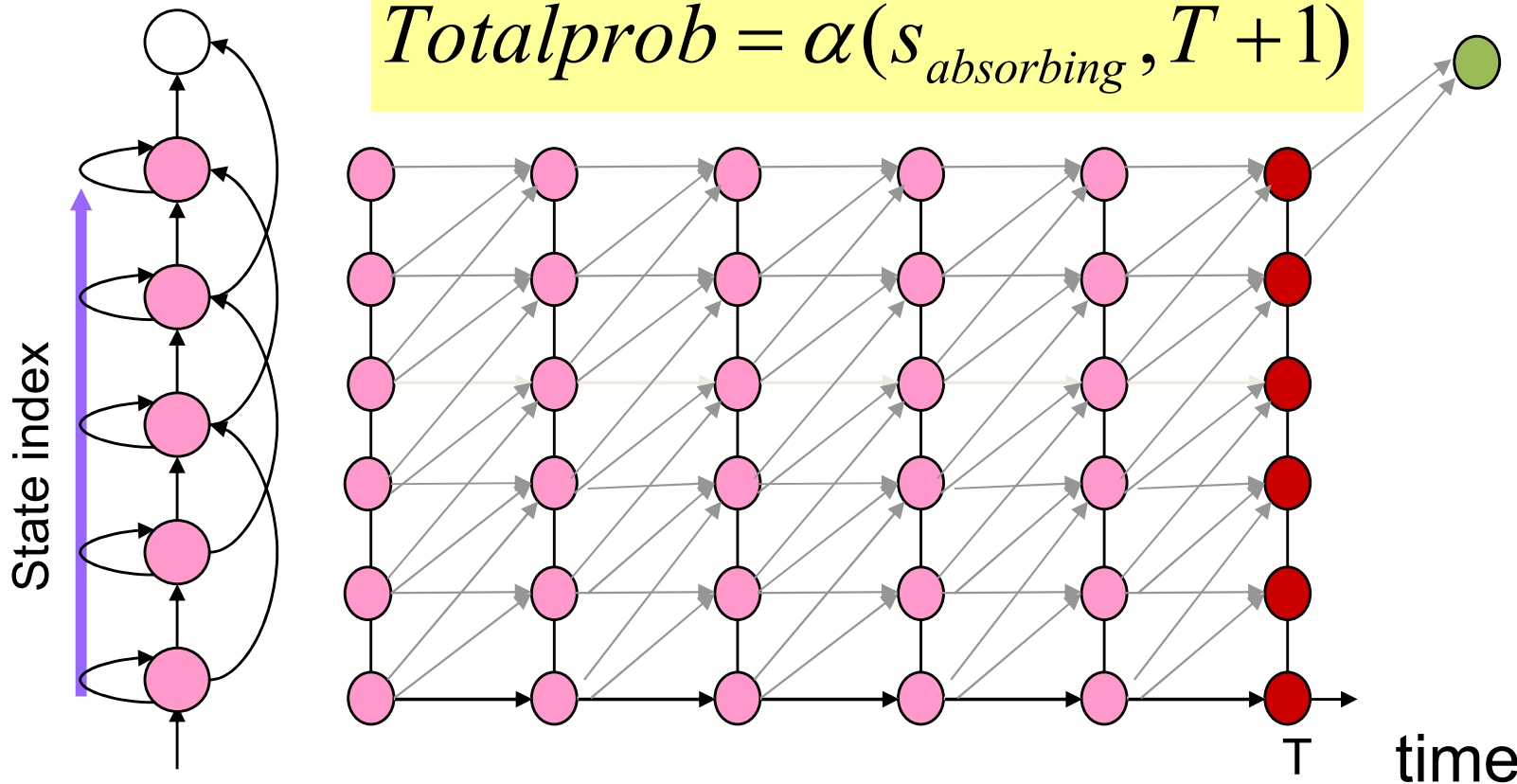
# The absorbing state



- Observation sequences are assumed to end only when the process arrives at an absorbing state
  - No observations are produced from the absorbing state

# The Forward Algorithm

$$Totalprob = \alpha(s_{absorbing}, T + 1)$$



$$\alpha(s_{absorbing}, T + 1) = \sum_{s'} \alpha(s', T) P(s_{absorbing} | s')$$

- Absorbing state model: The total probability is the alpha computed at the absorbing state after the final observation

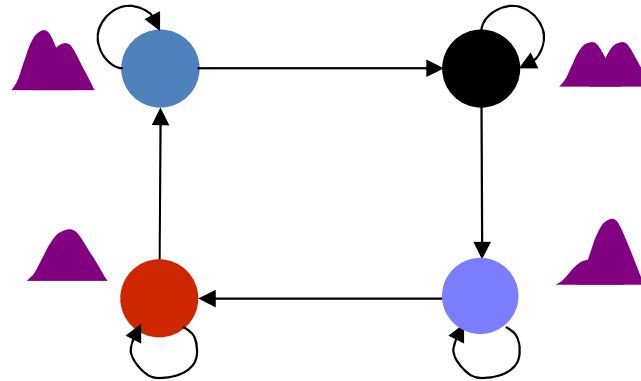


## Problem 2: State segmentation

- Given only a sequence of observations, how do we determine which sequence of states was followed in producing it?

# The HMM as a generator

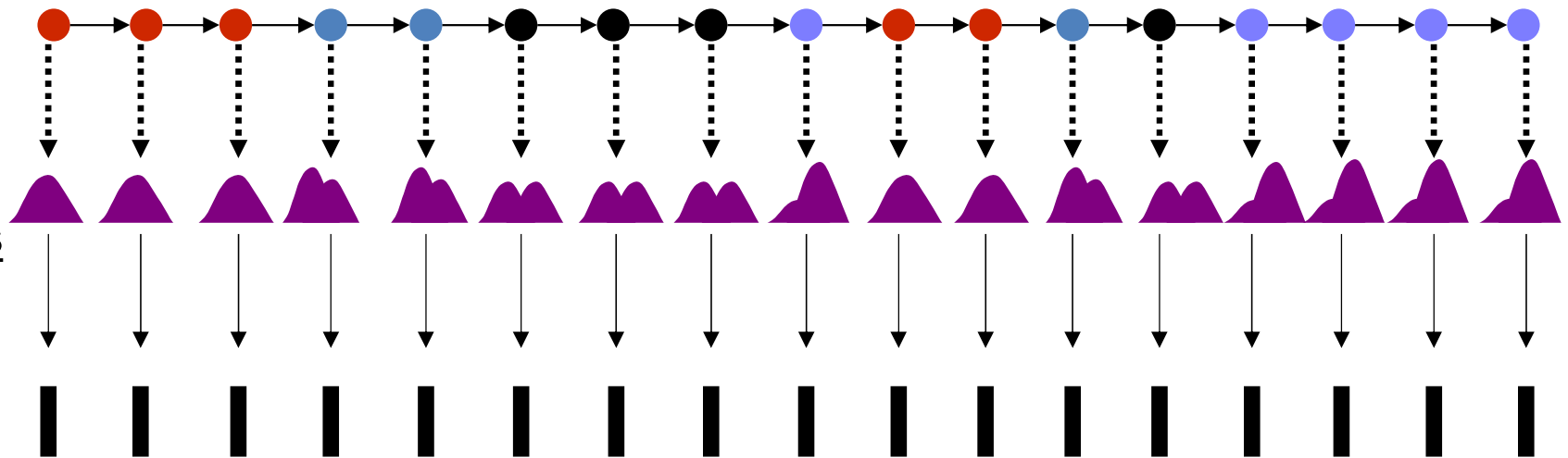
HMM assumed to be  
generating data



state  
sequence

state  
distributions

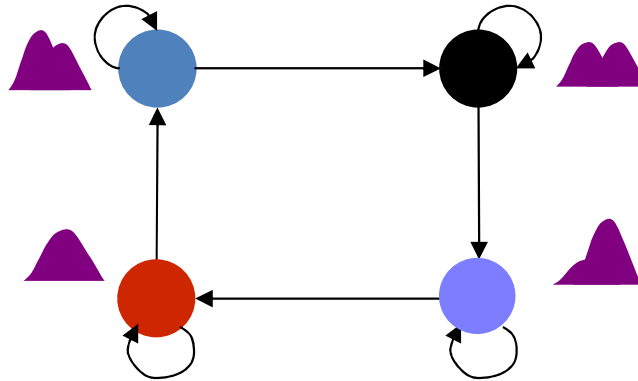
observation  
sequence



- The process goes through a series of states and produces observations from them

# States are hidden

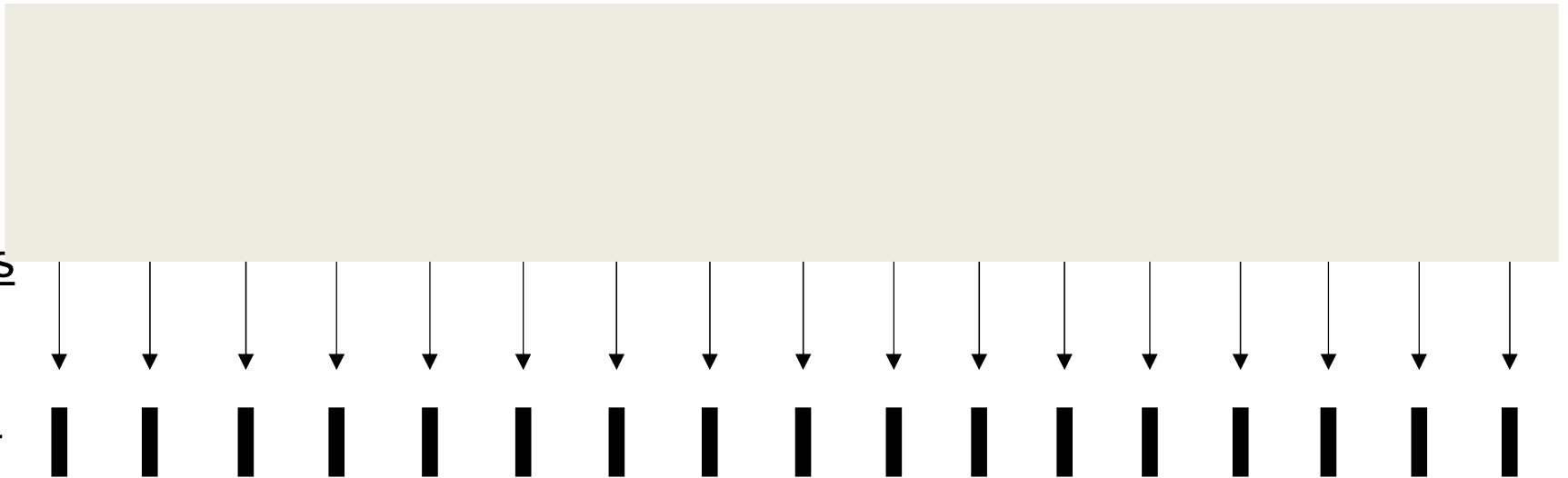
HMM assumed to be  
generating data



state  
sequence

state  
distributions

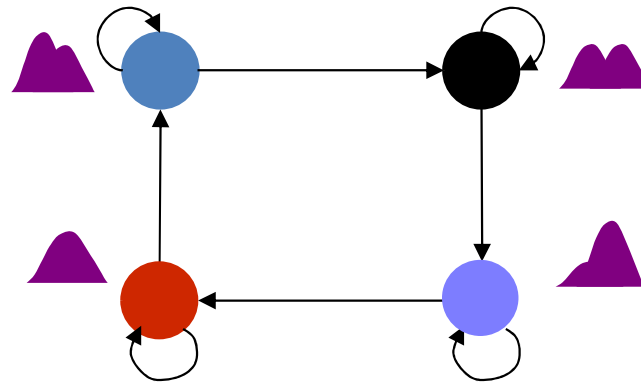
observation  
sequence



- The observations do not reveal the underlying state

# The state segmentation problem

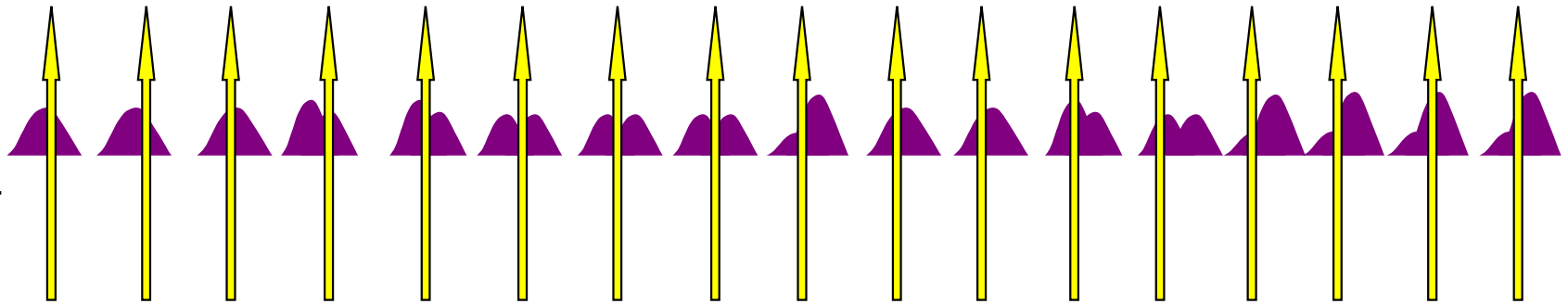
HMM assumed to be generating data



state sequence



state distributions



observation sequence



- State segmentation: Estimate state sequence given observations

# Estimating the State Sequence

- Many different state sequences are capable of producing the observation
- Solution: Identify the most *probable* state sequence
  - The state sequence for which the probability of progressing through that sequence and generating the observation sequence is maximum
  - i.e  $P(o_1, o_2, o_3, \dots, s_1, s_2, s_3, \dots)$  is maximum

# Estimating the state sequence

- Once again, exhaustive evaluation is impossibly expensive
- But once again a simple dynamic-programming solution is available

$$P(o_1, o_2, o_3, \dots, s_1, s_2, s_3, \dots) =$$

$$\underbrace{P(o_1 | s_1) P(o_2 | s_2) P(o_3 | s_3) \dots}_{\text{Observation sequence}} \underbrace{P(s_1) P(s_2 | s_1) P(s_3 | s_2) \dots}_{\text{State sequence}}$$

- Needed:

$$\arg \max_{s_1, s_2, s_3, \dots} P(o_1 | s_1) P(s_1) P(o_2 | s_2) P(s_2 | s_1) P(o_3 | s_3) P(s_3 | s_2)$$

# Estimating the state sequence

- Once again, exhaustive evaluation is impossibly expensive
- But once again a simple dynamic-programming solution is available

$$P(o_1, o_2, o_3, \dots, s_1, s_2, s_3, \dots) =$$

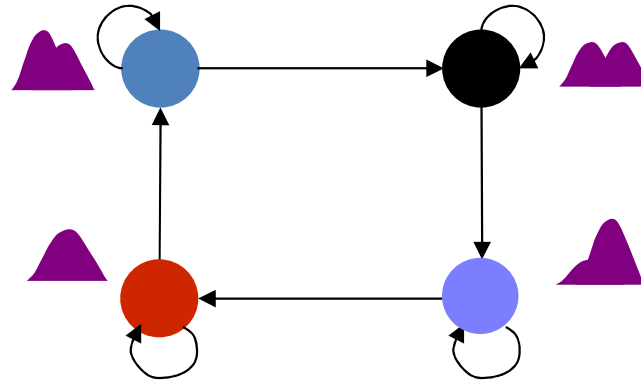
$$P(o_1 | s_1) P(o_2 | s_2) P(o_3 | s_3) \dots P(s_1) P(s_2 | s_1) P(s_3 | s_2) \dots$$

- Needed:

$$\arg \max_{s_1, s_2, s_3, \dots} P(o_1 | s_1) P(s_1) P(o_2 | s_2) P(s_2 | s_1) P(o_3 | s_3) P(s_3 | s_2)$$

# The HMM as a generator

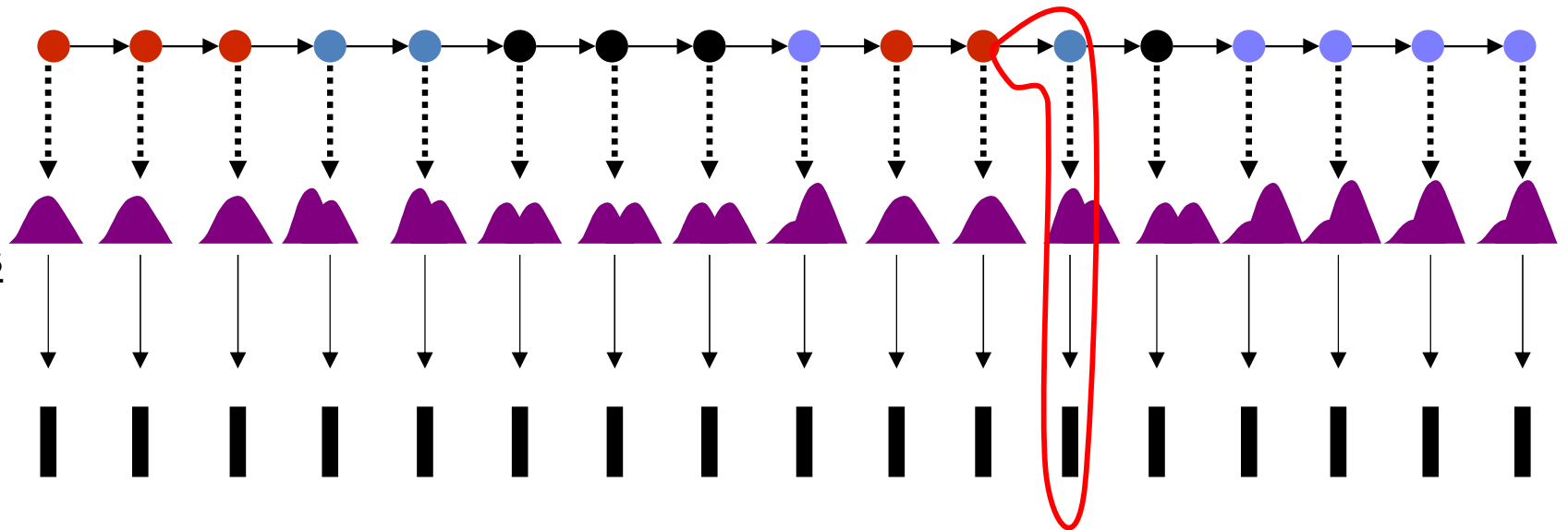
HMM assumed to be  
generating data



state  
sequence

state  
distributions

observation  
sequence



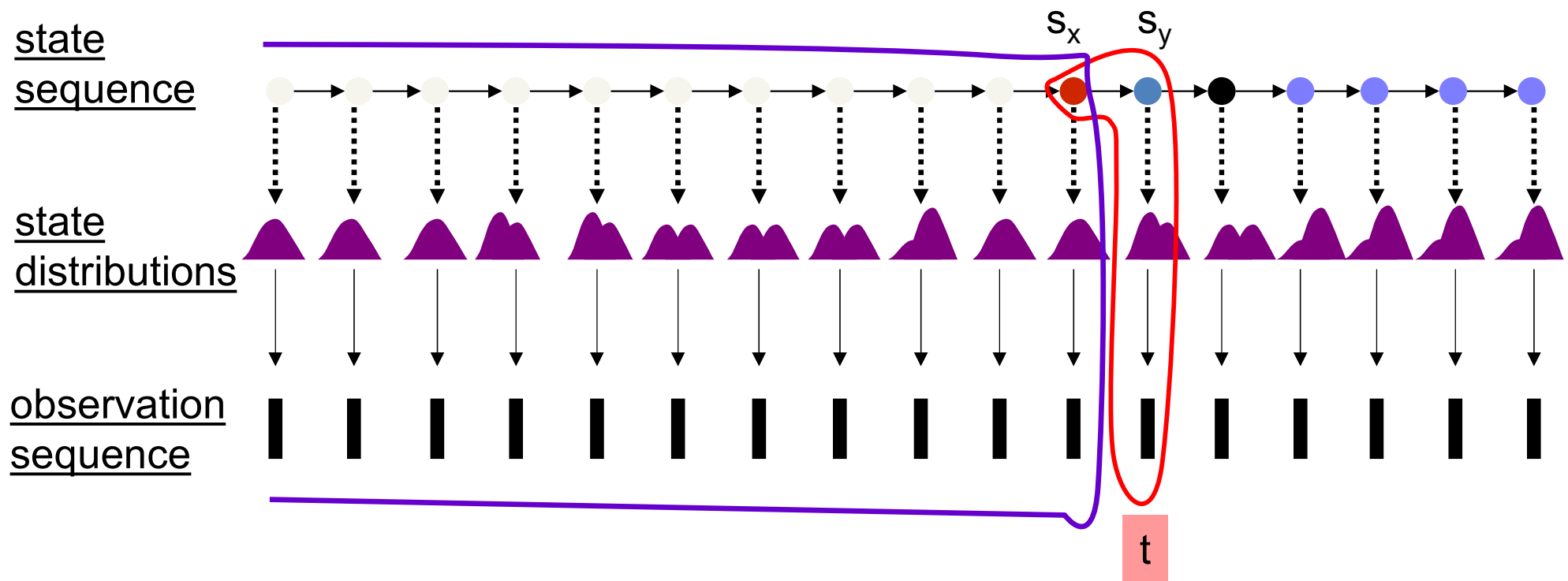
- Each enclosed term represents one forward transition and a subsequent emission



# The state sequence

- The probability of a state sequence  $?, ?, ?, ?, s_x, s_y$  ending at time  $t$ , and producing all observations until  $o_t$ 
  - $P(o_{1..t-1}, ?, ?, ?, ?, s_x, o_t, s_y) = P(\underline{o_{1..t-1}, ?, ?, ?, ?, s_x}) P(o_t | s_y) P(s_y | s_x)$
- The *best* state sequence that ends with  $s_x, s_y$  at  $t$  will have a probability equal to the probability of the best state sequence ending at  $t-1$  at  $s_x$  times  $P(o_t | s_y) P(s_y | s_x)$

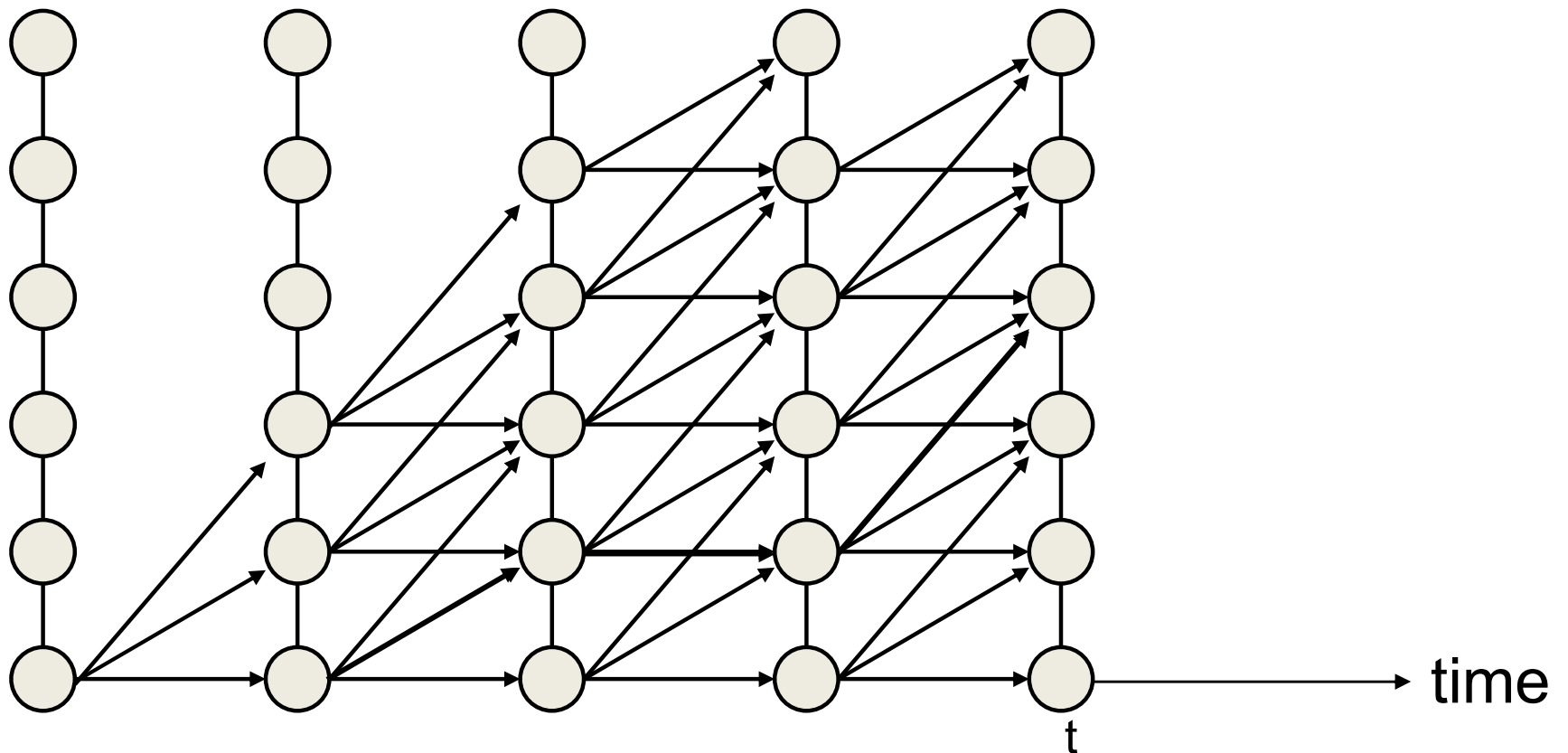
# Extending the state sequence



- The probability of a state sequence  $?, ?, ?, ?, s_x, s_y$  ending at time  $t$  and producing observations until  $o_t$ 
  - $P(o_{1..t-1}, o_t, ?, ?, ?, ?, s_x, s_y) = P(o_{1..t-1}, ?, ?, ?, ?, s_x) P(o_t | s_y) P(s_y | s_x)$

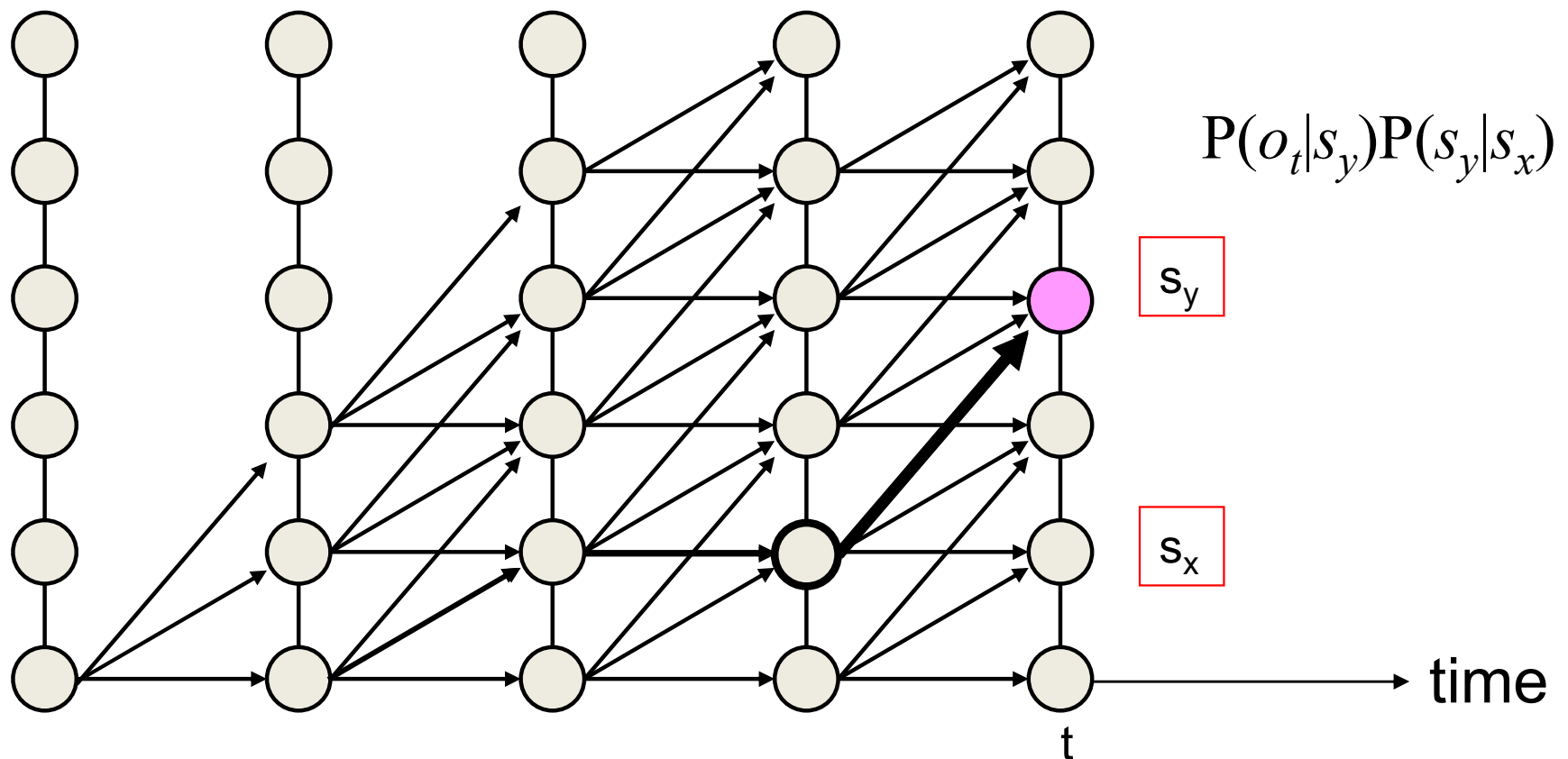
# Trellis

- The graph below shows the set of all possible state sequences through this HMM in five time instants



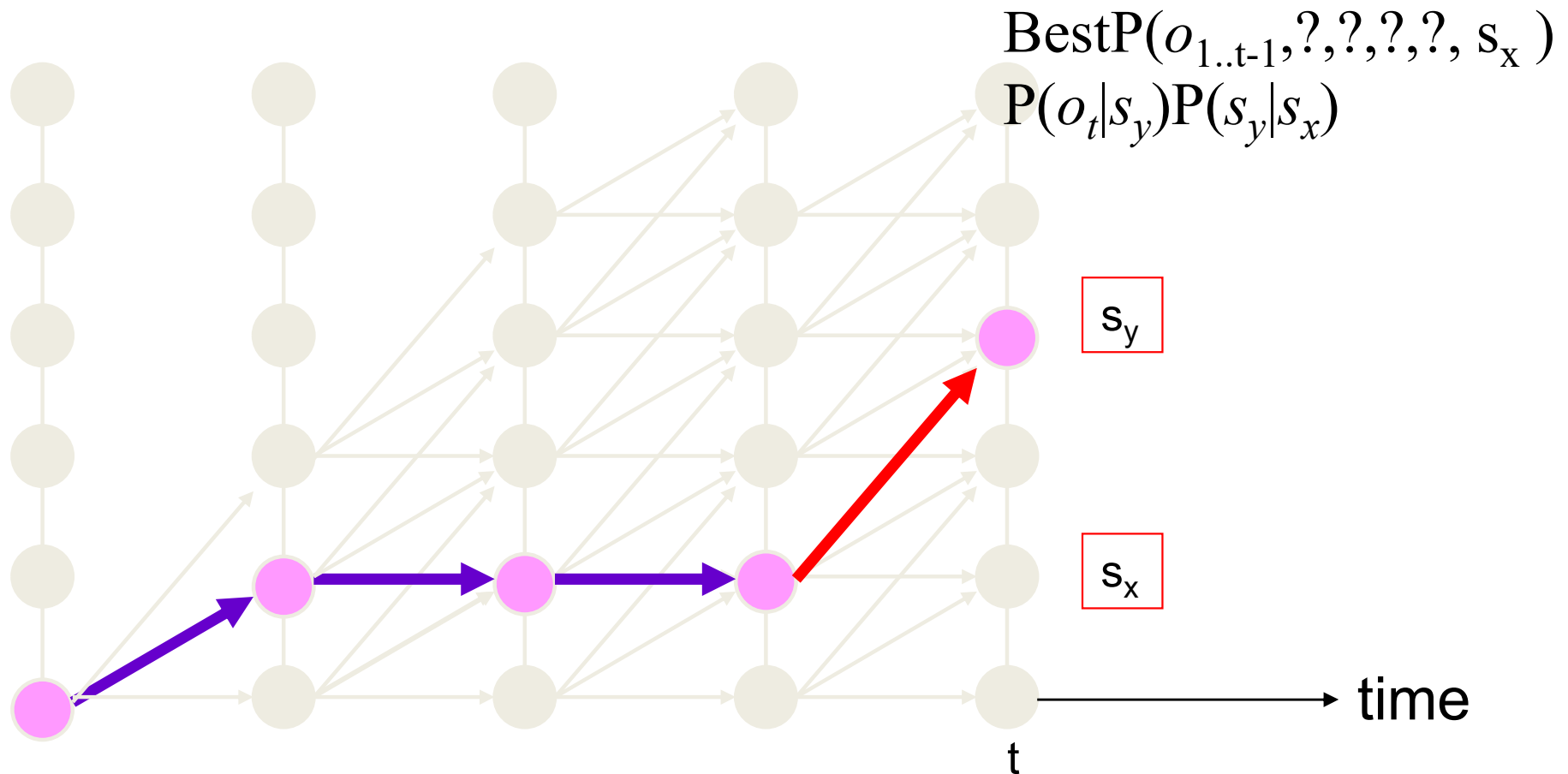
# The cost of extending a state sequence

- The cost of *extending* a state sequence ending at  $s_x$  is only dependent on the transition from  $s_x$  to  $s_y$ , and the observation probability at  $s_y$



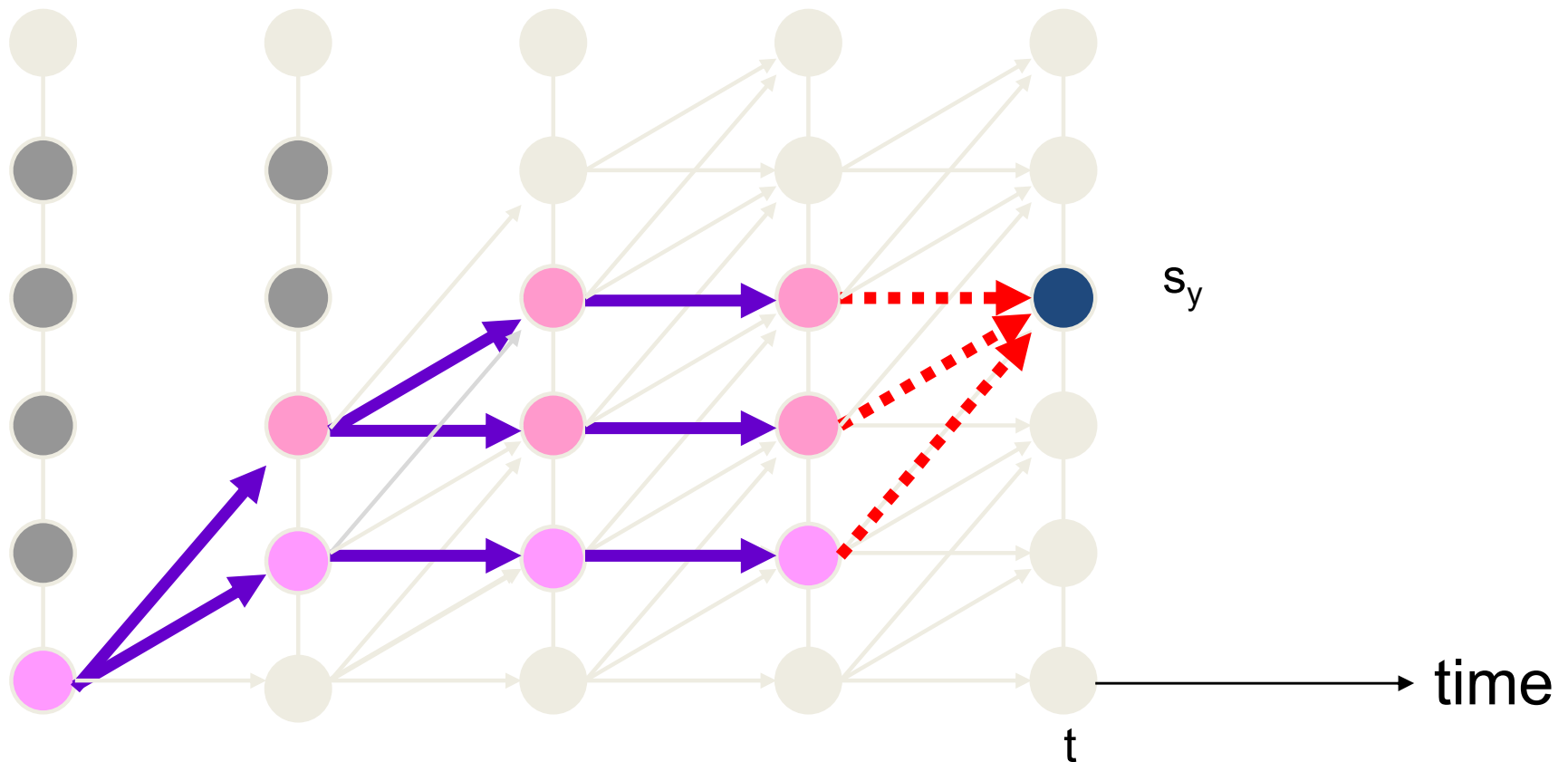
# The cost of extending a state sequence

- The best path to  $s_y$  through  $s_x$  is simply an extension of the best path to  $s_x$



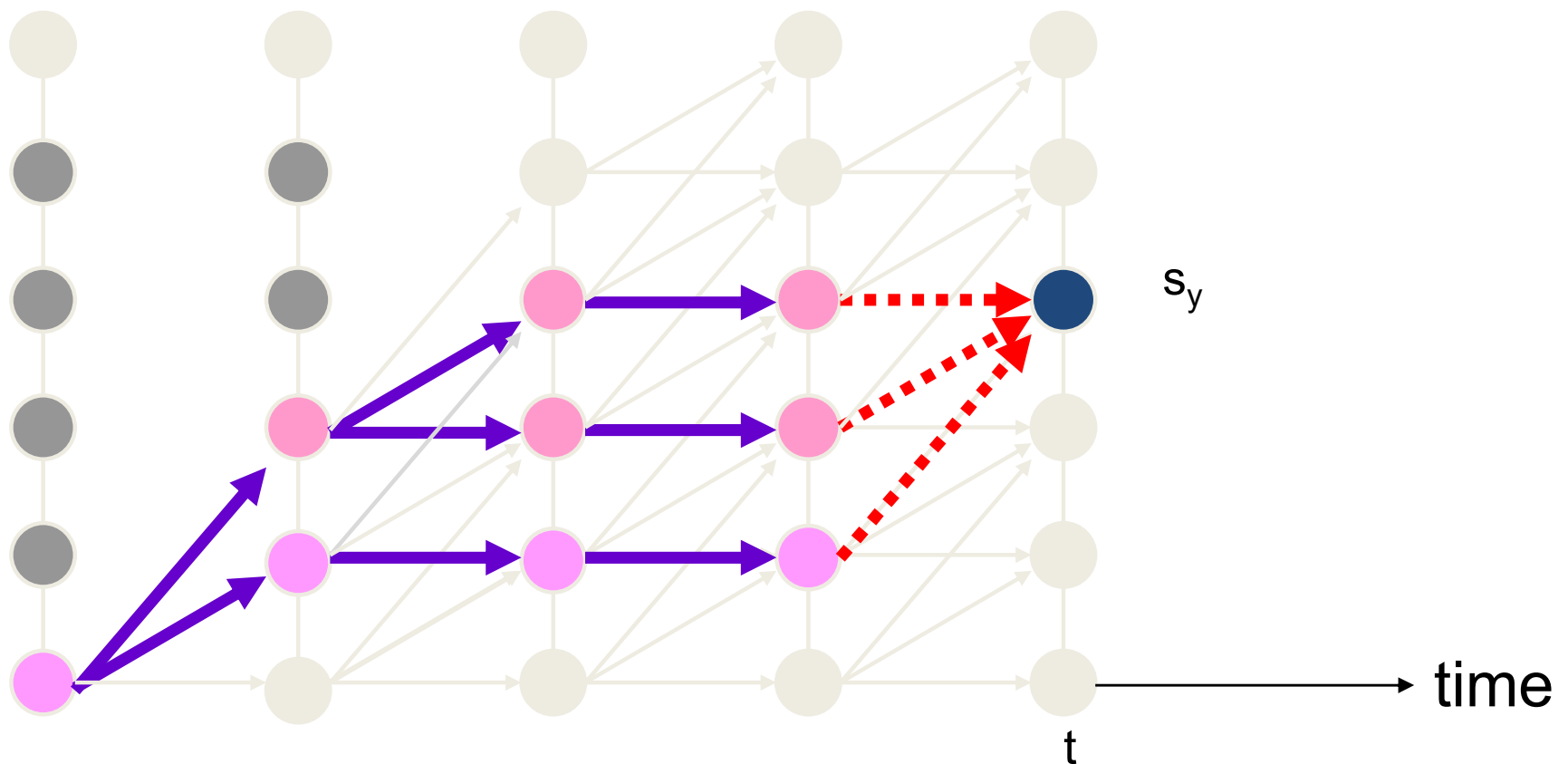
# The Recursion

- The overall best path to  $s_y$  is an extension of the best path to one of the states at the previous time



# The Recursion

- Prob. of best path to  $s_y$  =  
 $\text{Max}_{s_x} \text{BestP}(o_{1..t-1}, ?, ?, ?, s_x) P(o_t | s_y) P(s_y | s_x)$

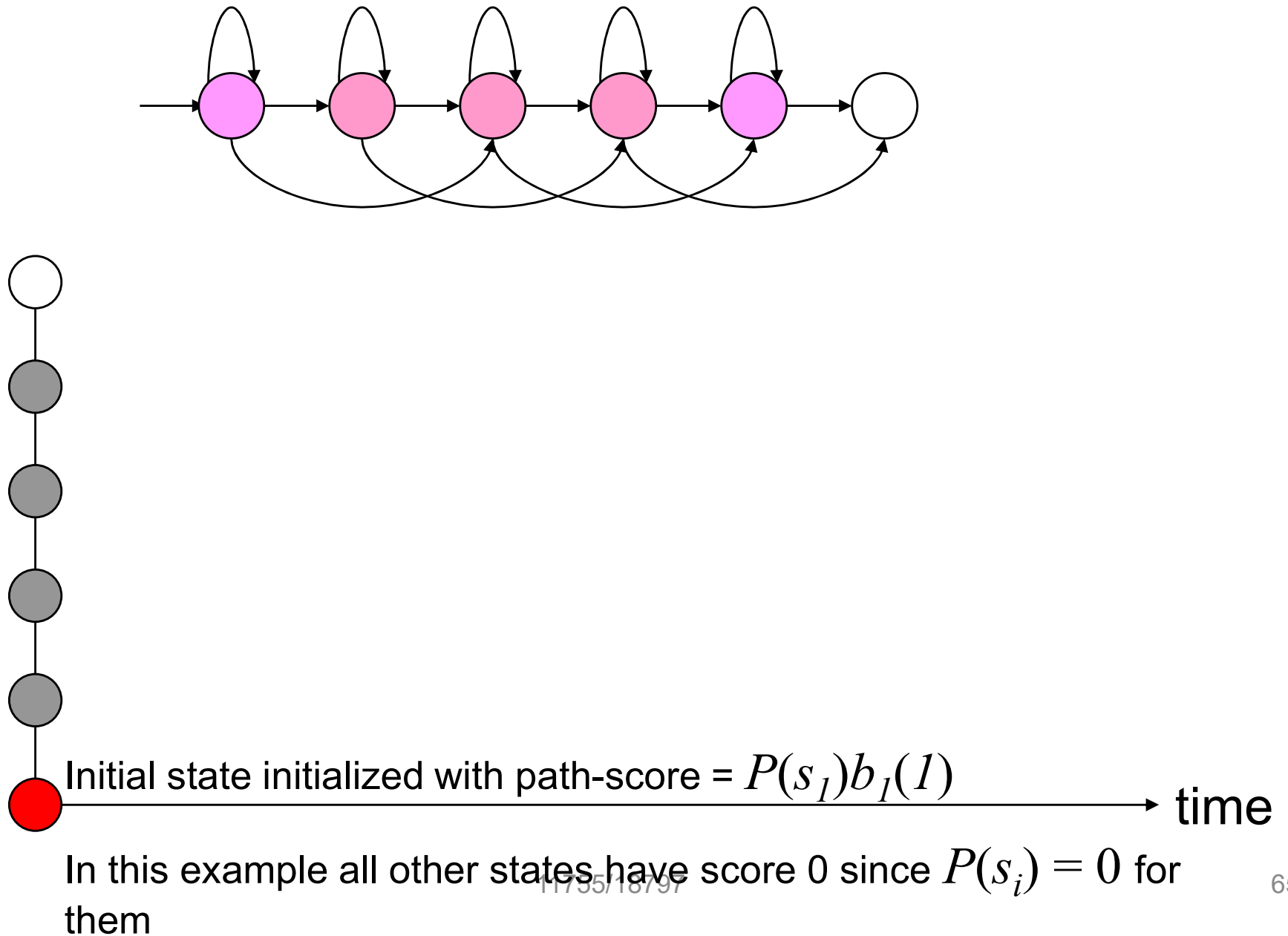


# Finding the best state sequence

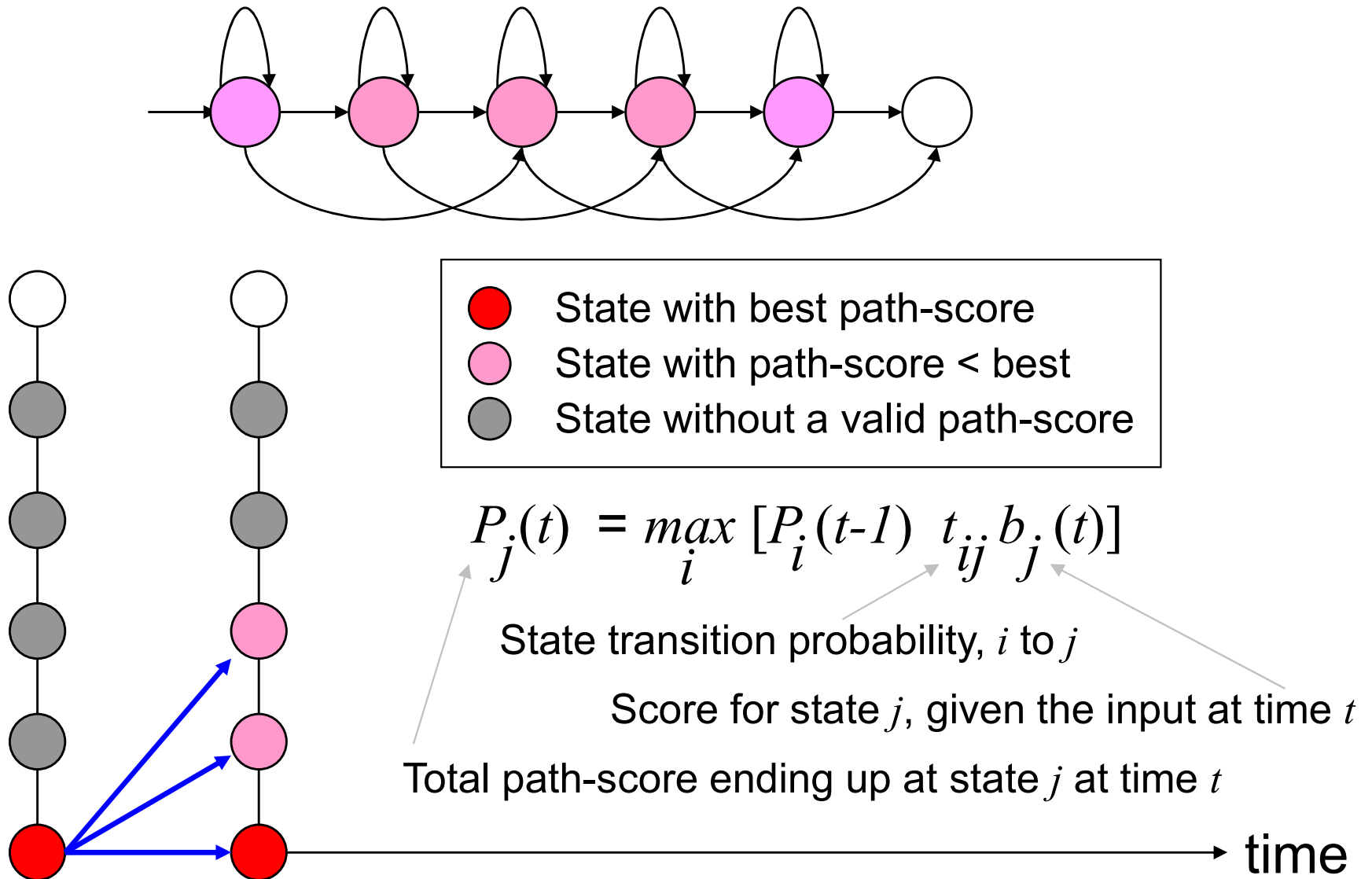
- The simple algorithm just presented is called the VITERBI algorithm in the literature
  - After A.J.Viterbi, who invented this dynamic programming algorithm for a completely different purpose: decoding error correction codes!



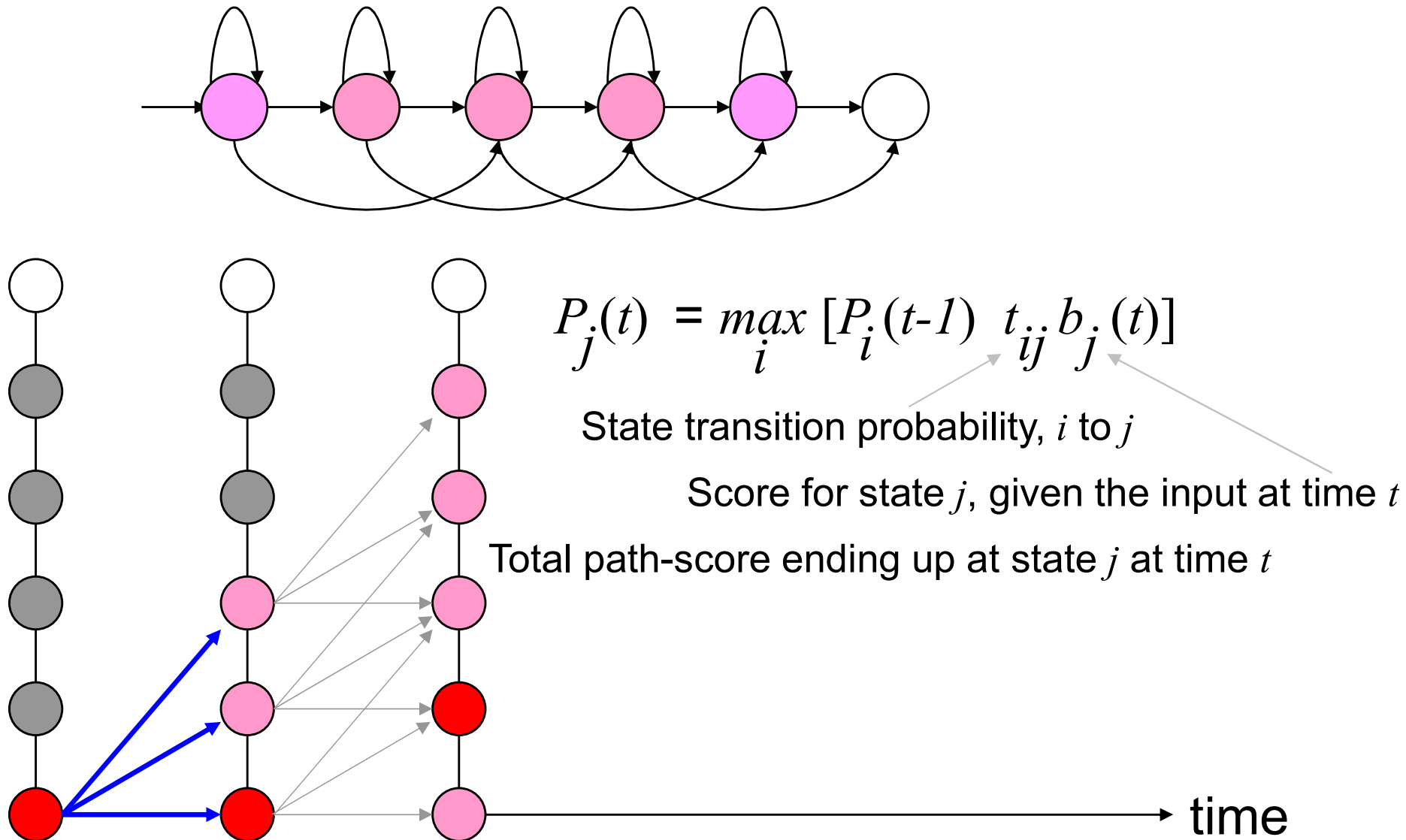
# Viterbi Search (contd.)



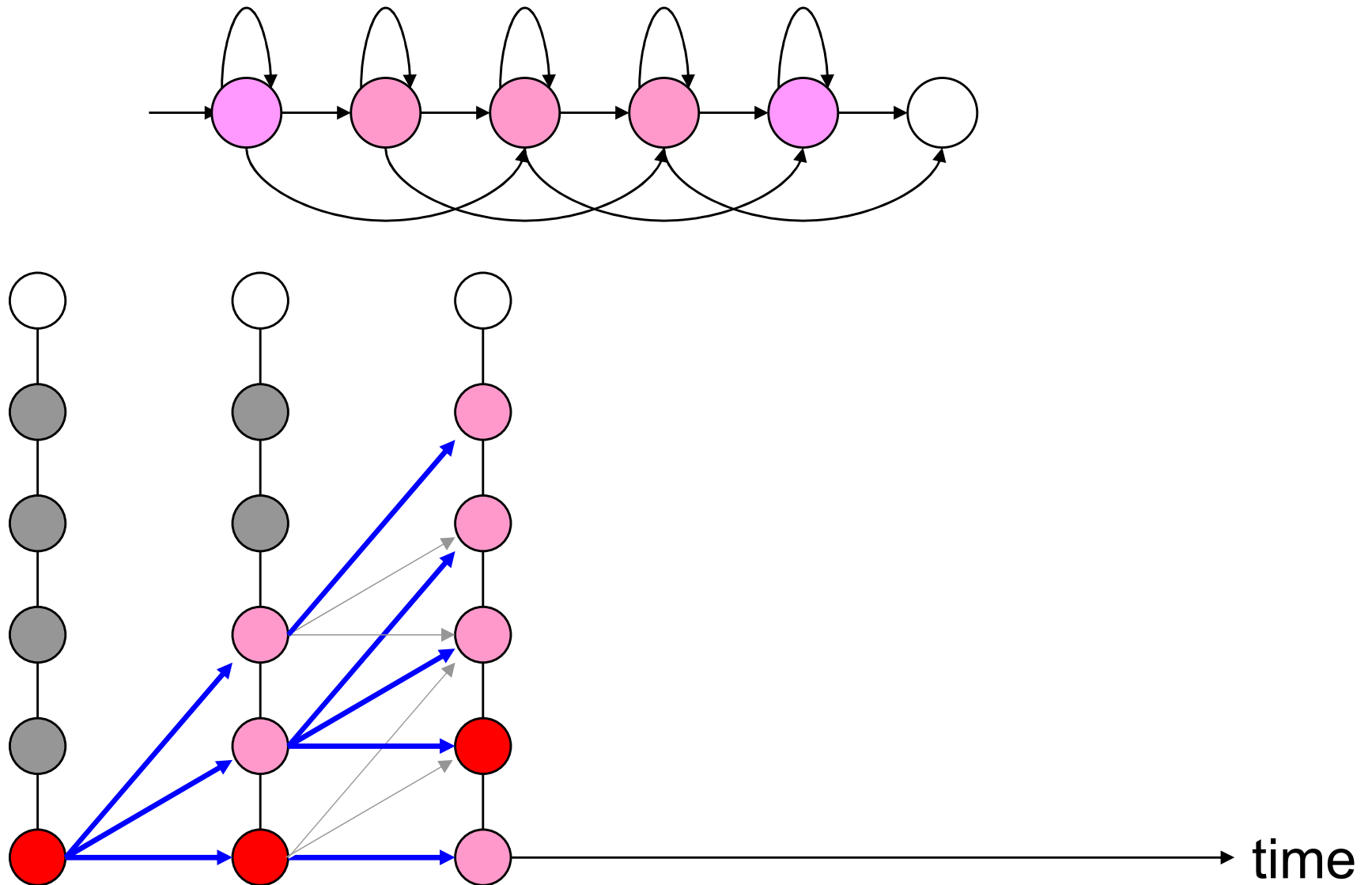
# Viterbi Search (contd.)



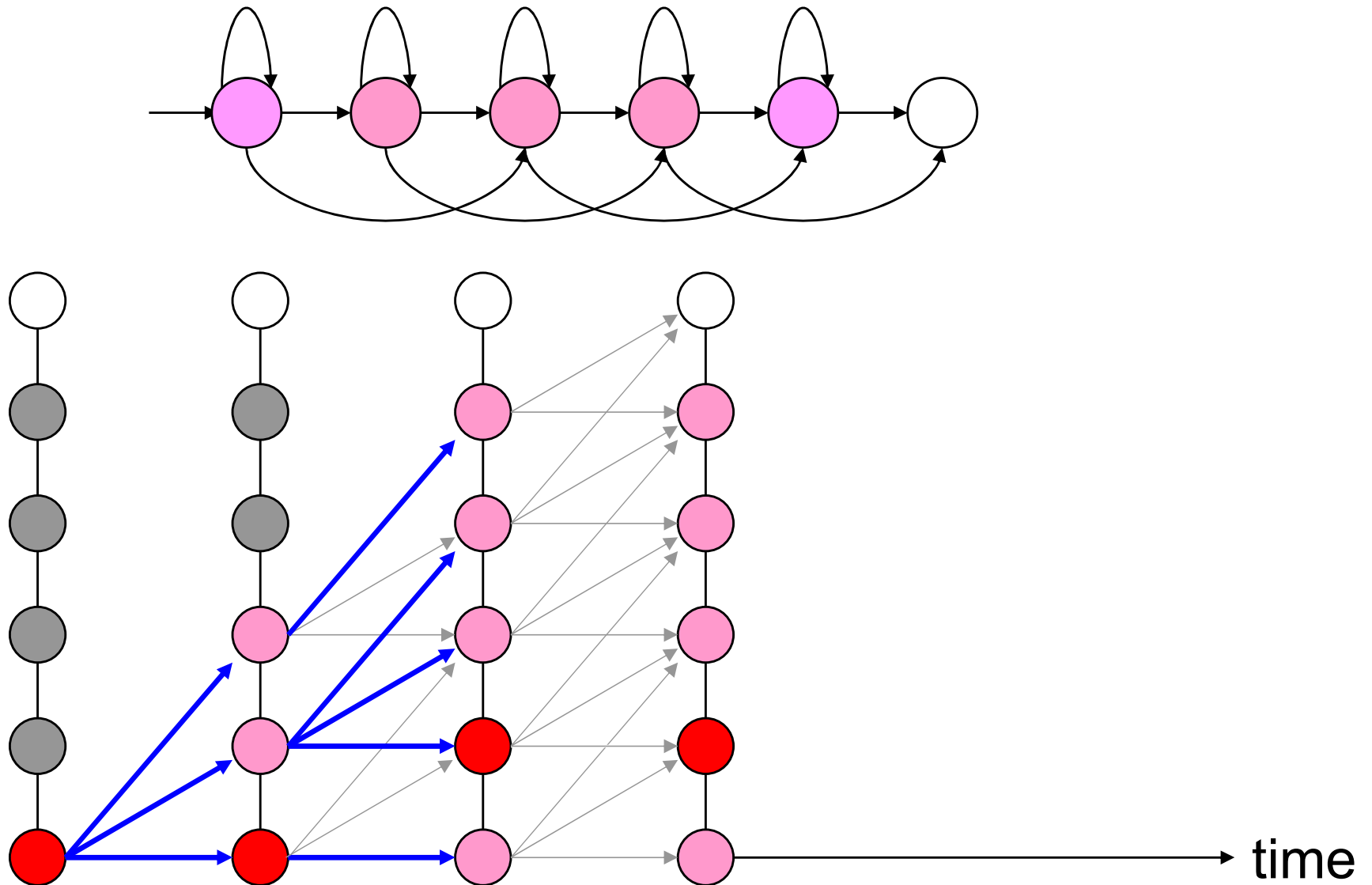
# Viterbi Search (contd.)



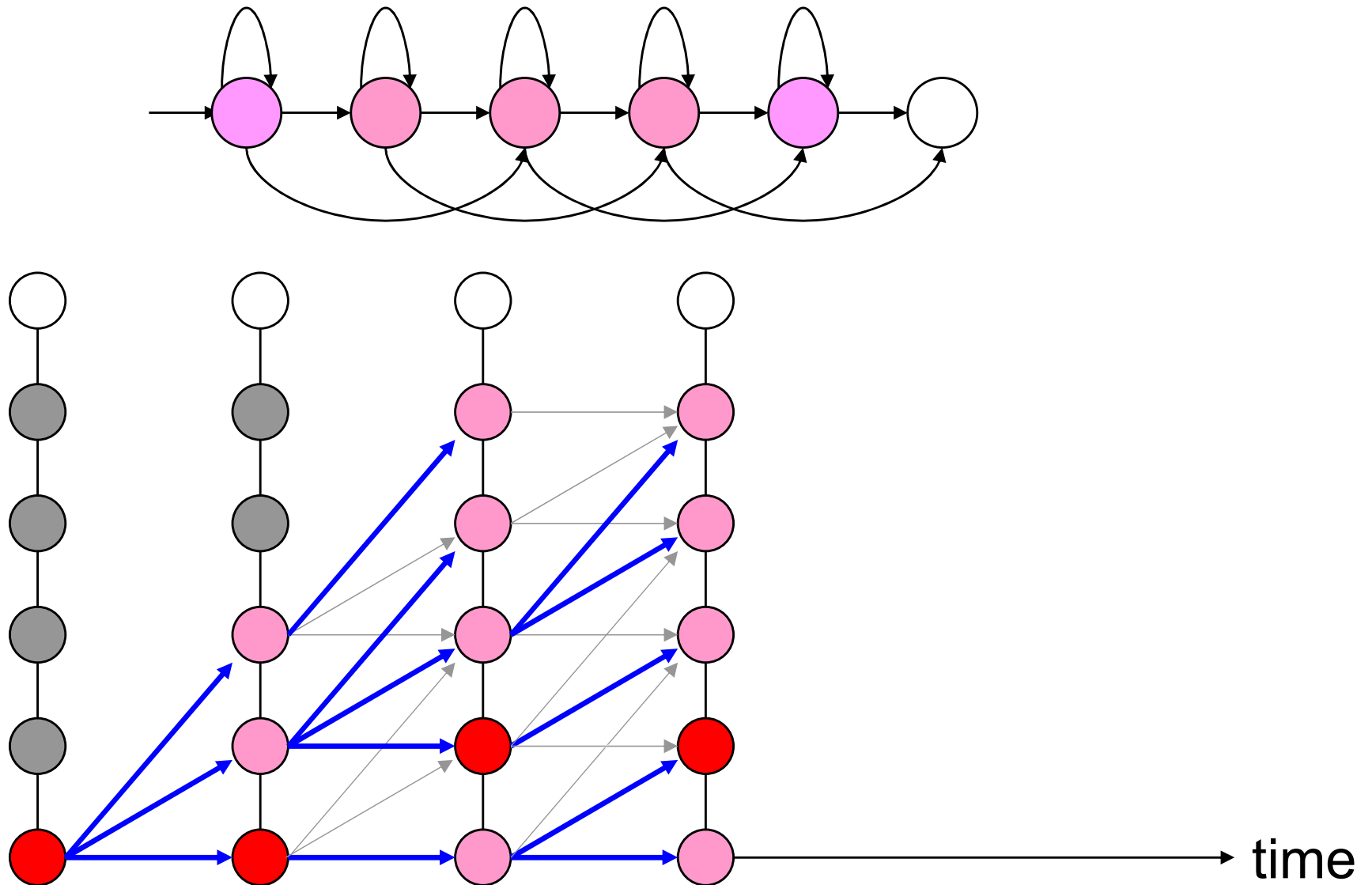
# Viterbi Search (contd.)



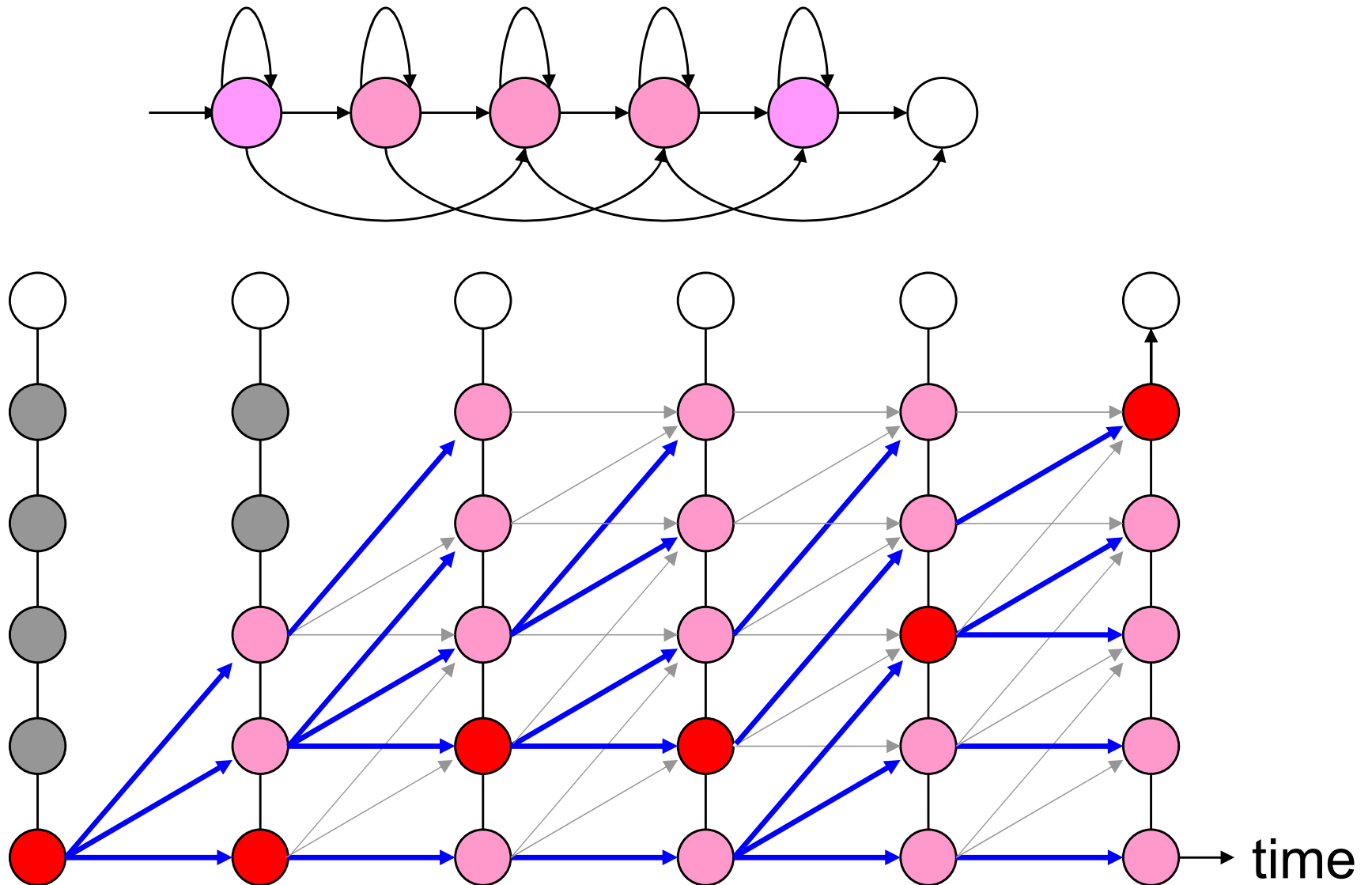
# Viterbi Search (contd.)



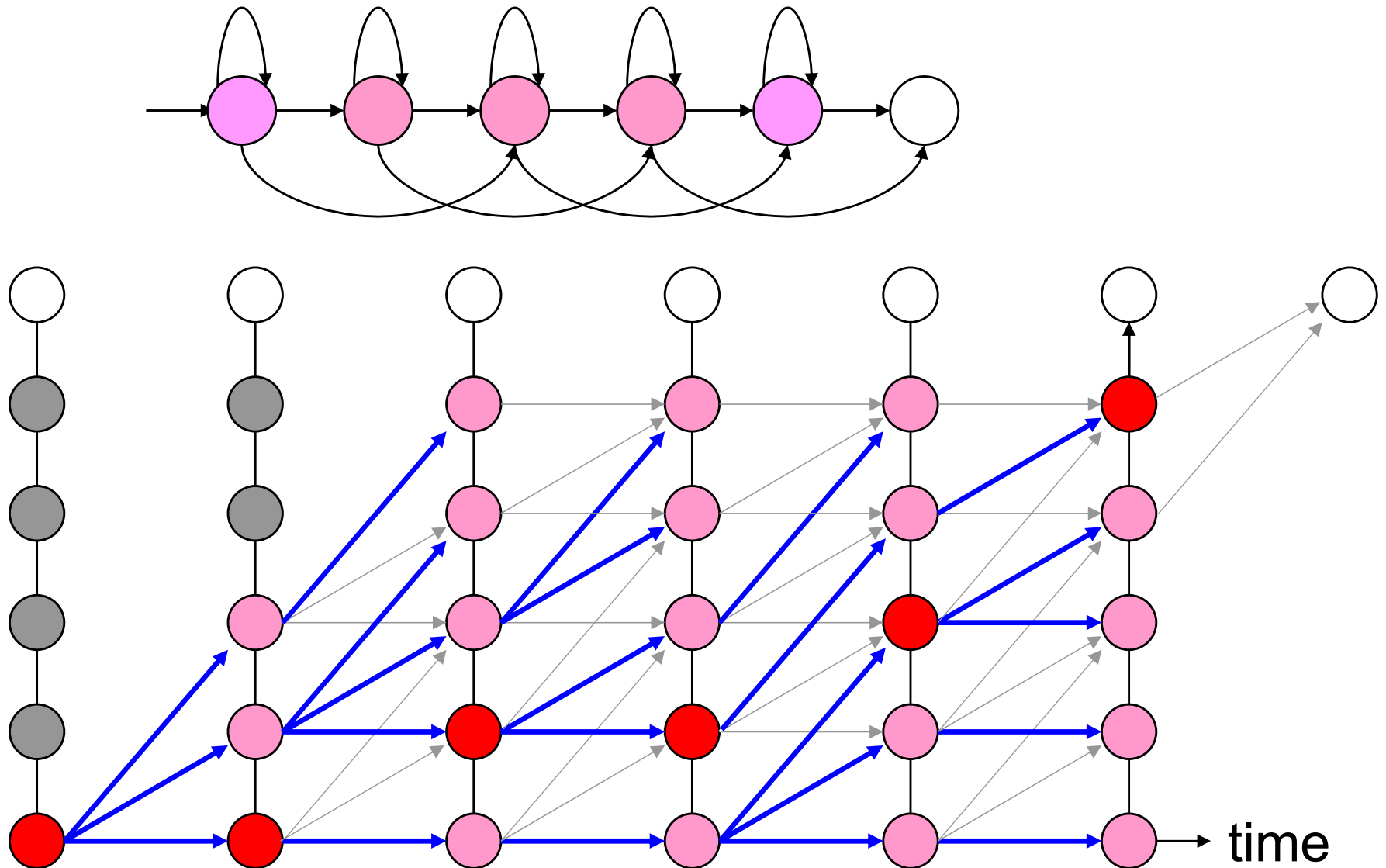
# Viterbi Search (contd.)



# Viterbi Search (contd.)

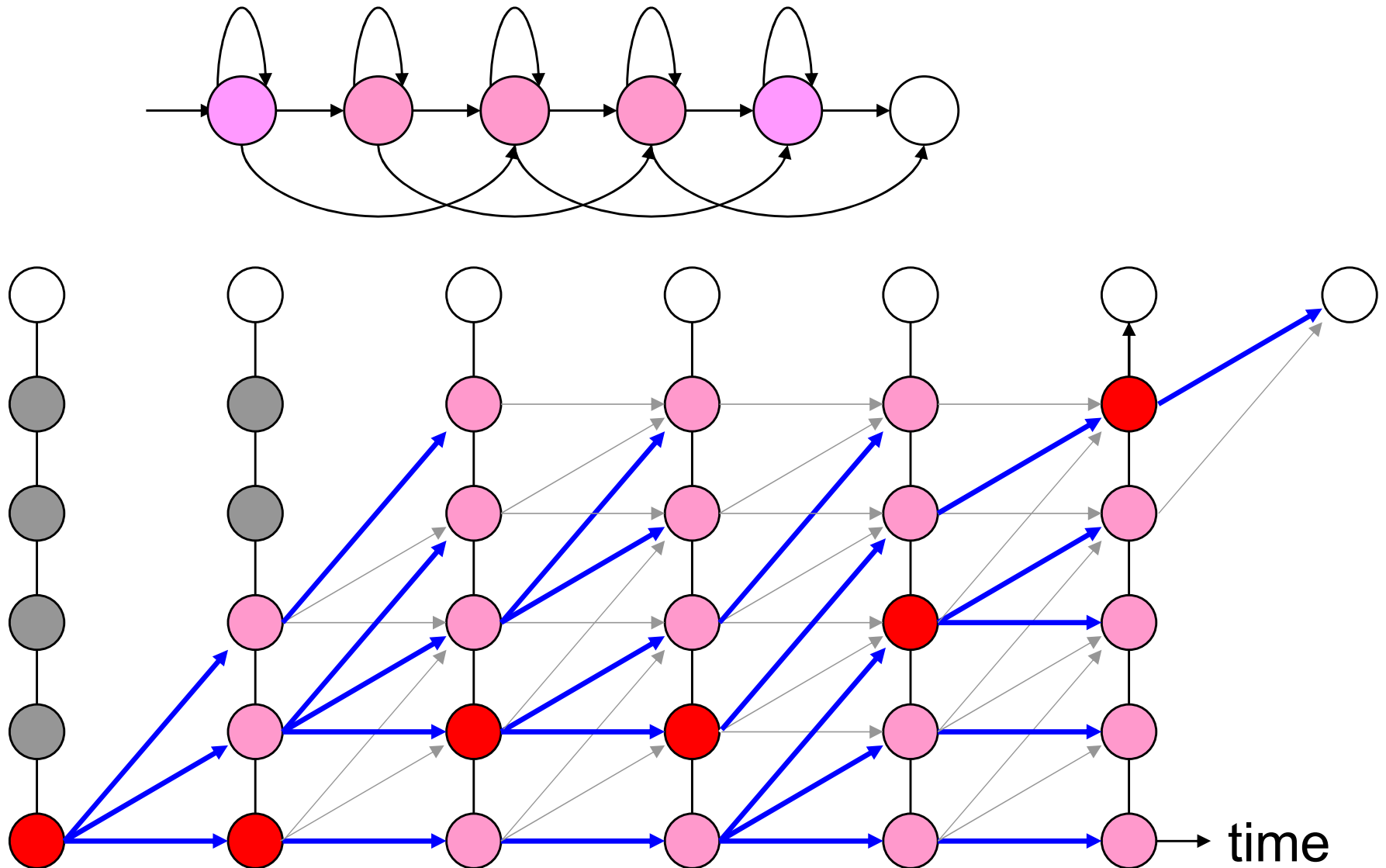


# Viterbi Search (contd.)

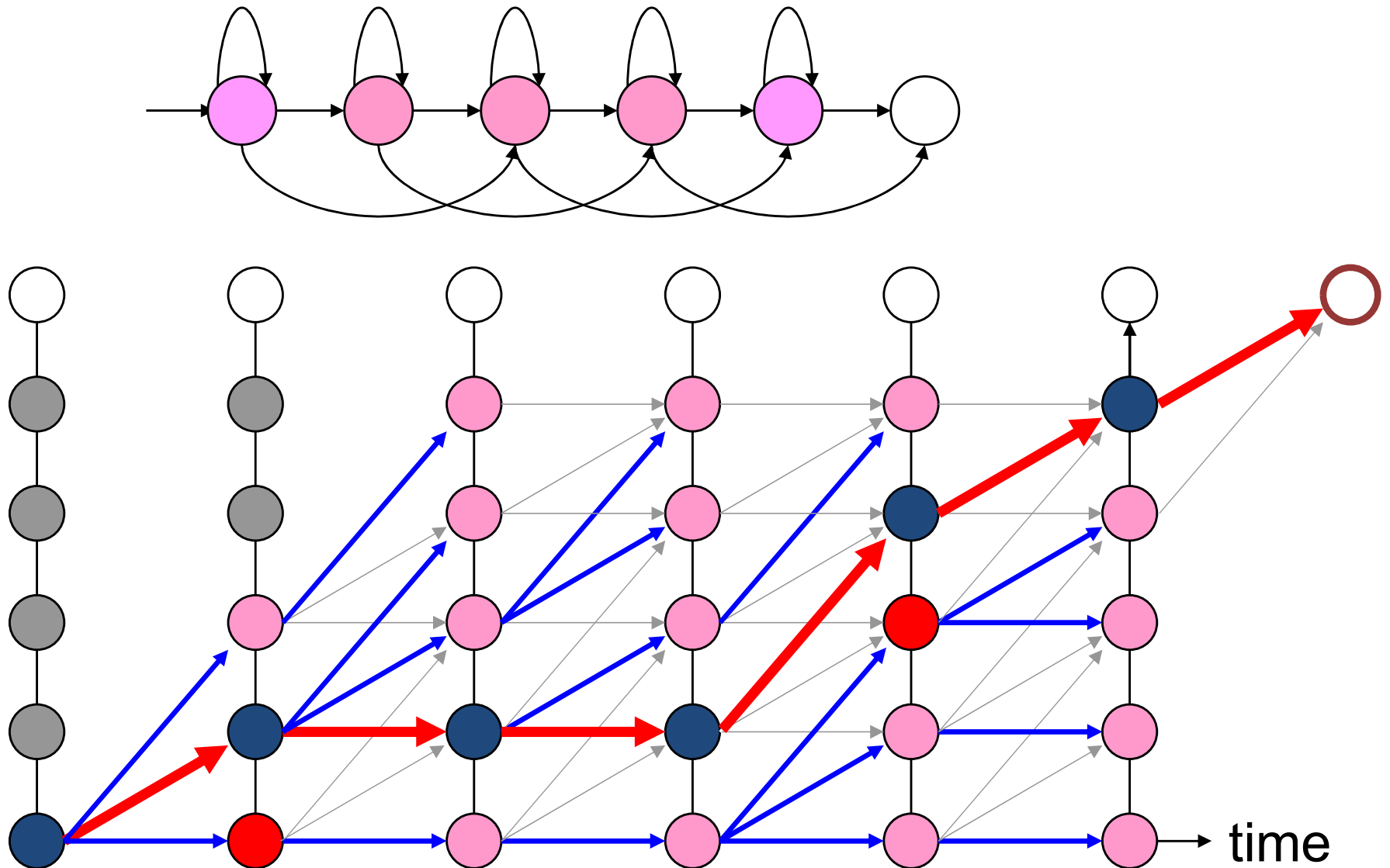




# Viterbi Search (contd.)

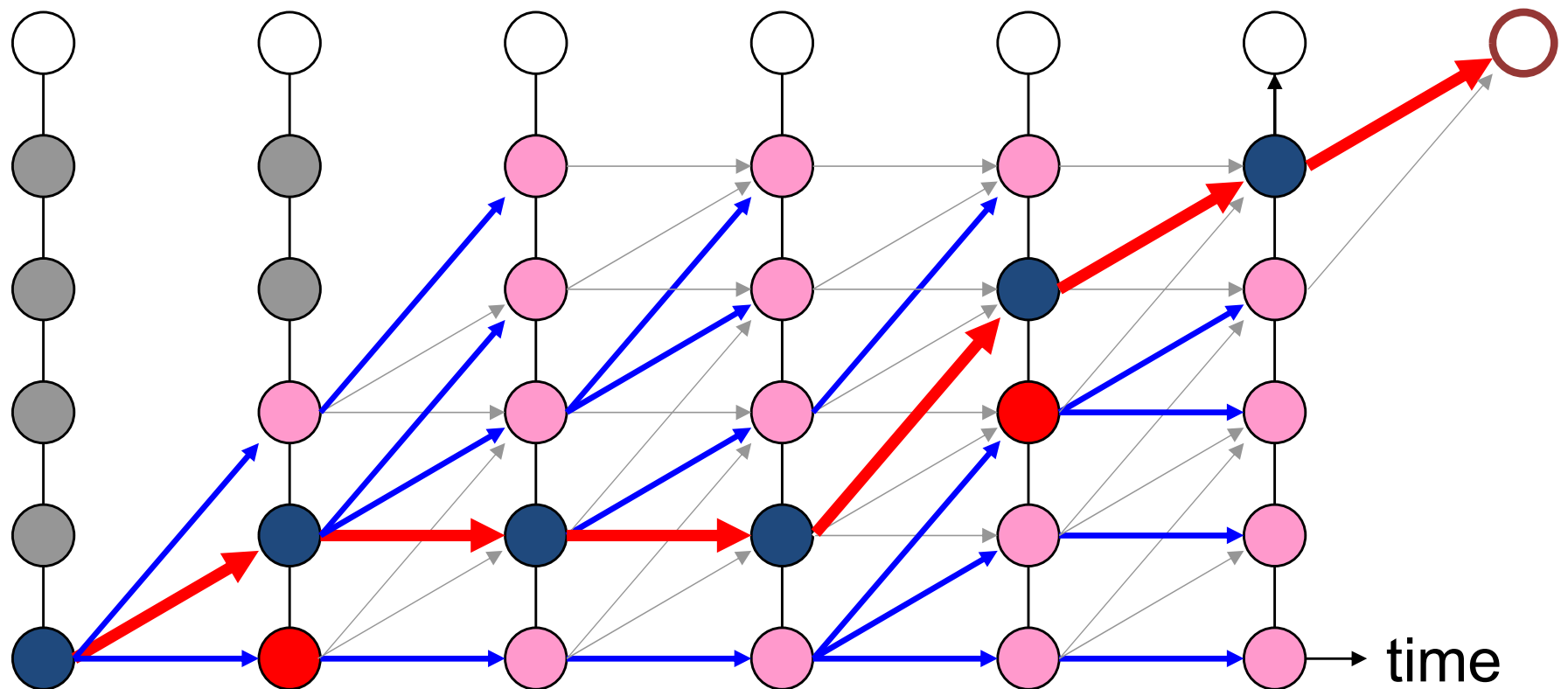


# Viterbi Search (contd.)



# Viterbi Search (contd.)

THE BEST STATE SEQUENCE IS THE ESTIMATE OF THE STATE SEQUENCE FOLLOWED IN GENERATING THE OBSERVATION



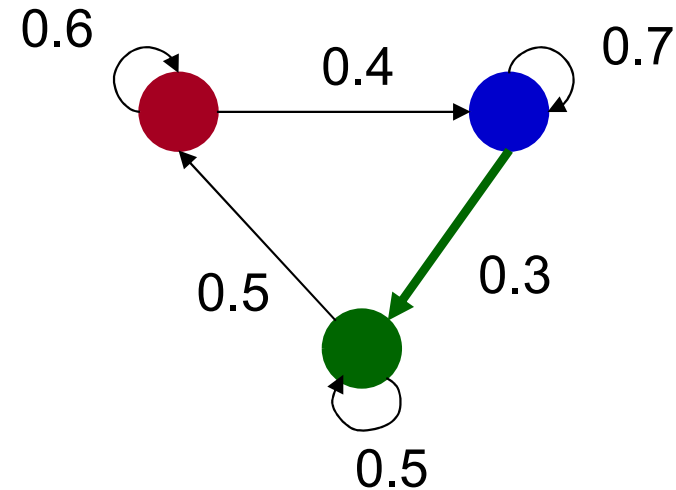
# Poll 4

# Problem3: Training HMM parameters

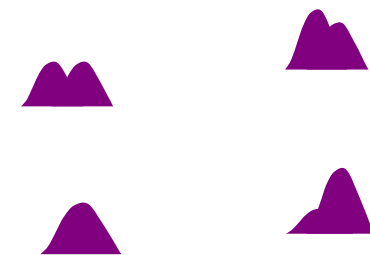
- We can compute the probability of an observation, and the best state sequence given an observation, using the HMM's parameters
- But where do the HMM parameters come from?
- They must be learned from a collection of observation sequences

# HMM Parameters

- The transition probabilities
  - Often represented as a matrix as here
  - $T_{ij}$  is the probability that when in state  $i$ , the process will move to  $j$
- The probability  $\pi_i$  of beginning at any state  $s_i$ 
  - The complete set is represented as  $\pi$
- The *state output distributions*
  - Typically histograms, Gaussians, or Gaussian mixtures
  - Assuming Gaussian
    - Parameters are mean and variance



$$T = \begin{pmatrix} .6 & .4 & 0 \\ 0 & .7 & .3 \\ .5 & 0 & .5 \end{pmatrix}$$

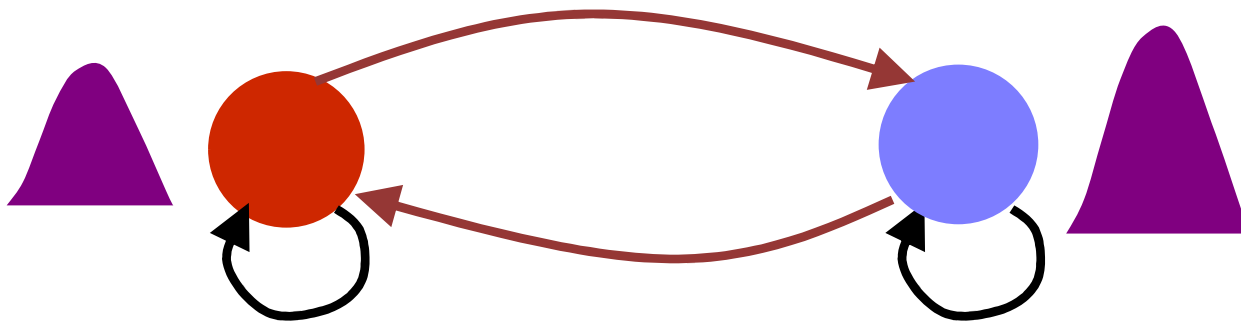


# Learning HMM parameters: Simple procedure – counting

- Given a set of training instances
- Iteratively:
  1. Initialize HMM parameters
  2. Segment all training instances
  3. Estimate transition probabilities and state output probability parameters by counting

# Learning by counting example

- Explanation by example in next few slides
- 2-state HMM, Gaussian PDF at states, 3 observation sequences
- Example shows ONE iteration
  - How to count after state sequences are obtained





# Example: Learning HMM Parameters

- We have an HMM with two states  $s_1$  and  $s_2$ .
- Observations are vectors  $x_{ij}$ 
  - $i$ -th sequence,  $j$ -th vector
- We are given the following three observation sequences
  - And have already estimated state sequences



Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	$X_{a1}$	$X_{a2}$	$X_{a3}$	$X_{a4}$	$X_{a5}$	$X_{a6}$	$X_{a7}$	$X_{a8}$	$X_{a9}$	$X_{a10}$

Observation 2

Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	$X_{b1}$	$X_{b2}$	$X_{b3}$	$X_{b4}$	$X_{b5}$	$X_{b6}$	$X_{b7}$	$X_{b8}$	$X_{b9}$

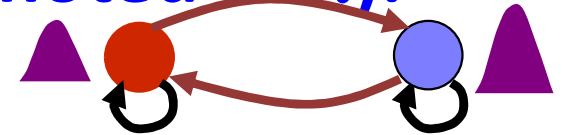
Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	$X_{c1}$	$X_{c2}$	$X_{c3}$	$X_{c4}$	$X_{c5}$	$X_{c6}$	$X_{c7}$	$X_{c8}$

# Example: Learning HMM Parameters

- Initial state probabilities (usually denoted as  $\pi$ ):

- We have 3 observations
- 2 of these begin with S1, and one with S2
- $\pi(S1) = 2/3, \pi(S2) = 1/3$



Observation 1

Time	1	2	3	4	5	6	7	8	9	10
stat	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	$X_{a1}$	$X_{a2}$	$X_{a3}$	$X_{a4}$	$X_{a5}$	$X_{a6}$	$X_{a7}$	$X_{a8}$	$X_{a9}$	$X_{a10}$

Observation 2

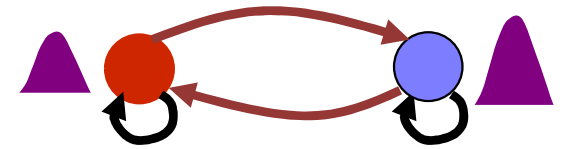
Time	1	2	3	4	5	6	7	8	9
stat	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	$X_{b1}$	$X_{b2}$	$X_{b3}$	$X_{b4}$	$X_{b5}$	$X_{b6}$	$X_{b7}$	$X_{b8}$	$X_{b9}$

Observation 3

Time	1	2	3	4	5	6	7	8
stat	S1	S2	S1	S1	S1	S2	S2	S2
Obs	$X_{c1}$	$X_{c2}$	$X_{c3}$	$X_{c4}$	$X_{c5}$	$X_{c6}$	$X_{c7}$	$X_{c8}$

# Example: Learning HMM Parameters

- Transition probabilities:
  - State S1 occurs 11 times in **non-terminal** locations



Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	$X_{a1}$	$X_{a2}$	$X_{a3}$	$X_{a4}$	$X_{a5}$	$X_{a6}$	$X_{a7}$	$X_{a8}$	$X_{a9}$	$X_{a10}$

Observation 2

Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	$X_{b1}$	$X_{b2}$	$X_{b3}$	$X_{b4}$	$X_{b5}$	$X_{b6}$	$X_{b7}$	$X_{b8}$	$X_{b9}$

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	$X_{c1}$	$X_{c2}$	$X_{c3}$	$X_{c4}$	$X_{c5}$	$X_{c6}$	$X_{c7}$	$X_{c8}$

# Example: Learning HMM Parameters



- Transition probabilities:

- State S1 occurs 11 times in non-terminal locations
- Of these, it is followed immediately by S1 6 times

Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	$x_{a1}$	$x_{a2}$	$x_{a3}$	$x_{a4}$	$x_{a5}$	$x_{a6}$	$x_{a7}$	$x_{a8}$	$x_{a9}$	$x_{a10}$

Observation 2

Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	$x_{b1}$	$x_{b2}$	$x_{b3}$	$x_{b4}$	$x_{b5}$	$x_{b6}$	$x_{b7}$	$x_{b8}$	$x_{b9}$

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	$x_{c1}$	$x_{c2}$	$x_{c3}$	$x_{c4}$	$x_{c5}$	$x_{c6}$	$x_{c7}$	$x_{c8}$

# Example: Learning HMM Parameters



- **Transition probabilities:**
  - State S1 occurs 11 times in non-terminal locations
  - Of these, it is followed immediately by S1 6 times
  - It is followed immediately by S2 5 times

Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	$X_{a1}$	$X_{a2}$	$X_{a3}$	$X_{a4}$	$X_{a5}$	$X_{a6}$	$X_{a7}$	$X_{a8}$	$X_{a9}$	$X_{a10}$

Observation 2

Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	$X_{b1}$	$X_{b2}$	$X_{b3}$	$X_{b4}$	$X_{b5}$	$X_{b6}$	$X_{b7}$	$X_{b8}$	$X_{b9}$

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	$X_{c1}$	$X_{c2}$	$X_{c3}$	$X_{c4}$	$X_{c5}$	$X_{c6}$	$X_{c7}$	$X_{c8}$

# Example: Learning HMM Parameters



- **Transition probabilities:**
  - State S1 occurs 11 times in non-terminal locations
  - Of these, it is followed immediately by S1 6 times
  - It is followed immediately by S2 5 times
  - $P(S1 \mid \mathbf{S1}) = 6 / 11$ ;  $P(S2 \mid \mathbf{S1}) = 5 / 11$

Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	$\mathbf{X}_{a1}$	$\mathbf{X}_{a2}$	$\mathbf{X}_{a3}$	$\mathbf{X}_{a4}$	$\mathbf{X}_{a5}$	$\mathbf{X}_{a6}$	$\mathbf{X}_{a7}$	$\mathbf{X}_{a8}$	$\mathbf{X}_{a9}$	$\mathbf{X}_{a10}$

Observation 2

Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	$\mathbf{X}_{b1}$	$\mathbf{X}_{b2}$	$\mathbf{X}_{b3}$	$\mathbf{X}_{b4}$	$\mathbf{X}_{b5}$	$\mathbf{X}_{b6}$	$\mathbf{X}_{b7}$	$\mathbf{X}_{b8}$	$\mathbf{X}_{b9}$

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	$\mathbf{X}_{c1}$	$\mathbf{X}_{c2}$	$\mathbf{X}_{c3}$	$\mathbf{X}_{c4}$	$\mathbf{X}_{c5}$	$\mathbf{X}_{c6}$	$\mathbf{X}_{c7}$	$\mathbf{X}_{c8}$

# Example: Learning HMM Parameters



- **Transition probabilities:**
  - State S2 occurs 13 times in non-terminal locations

Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs.	$X_{a1}$	$X_{a2}$	$X_{a3}$	$X_{a4}$	$X_{a5}$	$X_{a6}$	$X_{a7}$	$X_{a8}$	$X_{a9}$	$X_{a10}$

Observation 2

Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	$X_{b1}$	$X_{b2}$	$X_{b3}$	$X_{b4}$	$X_{b5}$	$X_{b6}$	$X_{b7}$	$X_{b8}$	$X_{b9}$

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	$X_{c1}$	$X_{c2}$	$X_{c3}$	$X_{c4}$	$X_{c5}$	$X_{c6}$	$X_{c7}$	$X_{c8}$

# Example: Learning HMM Parameters



- **Transition probabilities:**
  - State S2 occurs 13 times in non-terminal locations
  - Of these, it is followed immediately by S1 5 times

Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	$X_{a1}$	$X_{a2}$	$X_{a3}$	$X_{a4}$	$X_{a5}$	$X_{a6}$	$X_{a7}$	$X_{a8}$	$X_{a9}$	$X_{a10}$

Observation 2

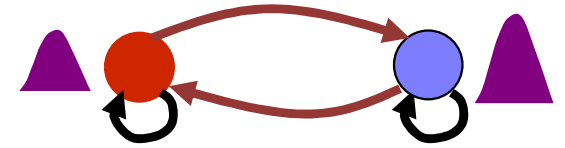
Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	$X_{b1}$	$X_{b2}$	$X_{b3}$	$X_{b4}$	$X_{b5}$	$X_{b6}$	$X_{b7}$	$X_{b8}$	$X_{b9}$

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	$X_{c1}$	$X_{c2}$	$X_{c3}$	$X_{c4}$	$X_{c5}$	$X_{c6}$	$X_{c7}$	$X_{c8}$



# Example: Learning HMM Parameters



- **Transition probabilities:**
  - State S2 occurs 13 times in non-terminal locations
  - Of these, it is followed immediately by S1 5 times
  - It is followed immediately by S2 8 times

Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	$X_{a1}$	$X_{a2}$	$X_{a3}$	$X_{a4}$	$X_{a5}$	$X_{a6}$	$X_{a7}$	$X_{a8}$	$X_{a9}$	$X_{a10}$

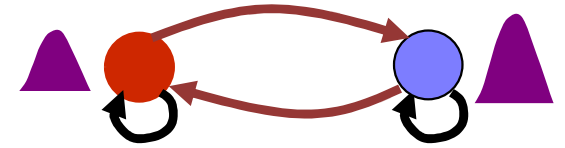
Observation 2

Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	$X_{b1}$	$X_{b2}$	$X_{b3}$	$X_{b4}$	$X_{b5}$	$X_{b6}$	$X_{b7}$	$X_{b8}$	$X_{b9}$

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	$X_{c1}$	$X_{c2}$	$X_{c3}$	$X_{c4}$	$X_{c5}$	$X_{c6}$	$X_{c7}$	$X_{c8}$

# Example: Learning HMM Parameters



- **Transition probabilities:**
  - State S2 occurs 13 times in non-terminal locations
  - Of these, it is followed immediately by S1 5 times
  - It is followed immediately by S2 8 times
  - $P(S1 \mid \mathbf{S2}) = 5 / 13$ ;  $P(S2 \mid \mathbf{S2}) = 8 / 13$

Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	$\mathbf{X}_{a1}$	$\mathbf{X}_{a2}$	$\mathbf{X}_{a3}$	$\mathbf{X}_{a4}$	$\mathbf{X}_{a5}$	$\mathbf{X}_{a6}$	$\mathbf{X}_{a7}$	$\mathbf{X}_{a8}$	$\mathbf{X}_{a9}$	$\mathbf{X}_{a10}$

Observation 2

Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	$\mathbf{X}_{b1}$	$\mathbf{X}_{b2}$	$\mathbf{X}_{b3}$	$\mathbf{X}_{b4}$	$\mathbf{X}_{b5}$	$\mathbf{X}_{b6}$	$\mathbf{X}_{b7}$	$\mathbf{X}_{b8}$	$\mathbf{X}_{b9}$

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	$\mathbf{X}_{c1}$	$\mathbf{X}_{c2}$	$\mathbf{X}_{c3}$	$\mathbf{X}_{c4}$	$\mathbf{X}_{c5}$	$\mathbf{X}_{c6}$	$\mathbf{X}_{c7}$	$\mathbf{X}_{c8}$

# Parameters learnt so far

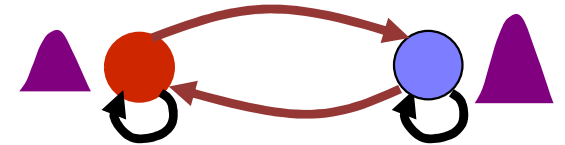
- State initial probabilities, often denoted as  $\pi$ 
  - $\pi(S1) = 2/3 = 0.66$
  - $\pi(S2) = 1/3 = 0.33$
- State transition probabilities
  - $P(S1 | S1) = 6/11 = 0.545$ ;  $P(S2 | S1) = 5/11 = 0.455$
  - $P(S1 | S2) = 5/13 = 0.385$ ;  $P(S2 | S2) = 8/13 = 0.615$
  - Represented as a transition matrix

$$A = \begin{pmatrix} P(S1 | S1) & P(S2 | S1) \\ P(S1 | S2) & P(S2 | S2) \end{pmatrix} = \begin{pmatrix} 0.545 & 0.455 \\ 0.385 & 0.615 \end{pmatrix}$$

Each row of this matrix must sum to 1.0

# Example: Learning HMM Parameters

- State output probability for S1
  - There are 13 observations in S1



Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	$X_{a1}$	$X_{a2}$	$X_{a3}$	$X_{a4}$	$X_{a5}$	$X_{a6}$	$X_{a7}$	$X_{a8}$	$X_{a9}$	$X_{a10}$

Observation 2

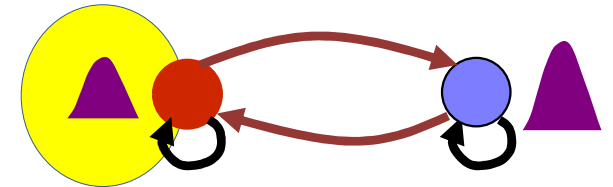
Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	$X_{b1}$	$X_{b2}$	$X_{b3}$	$X_{b4}$	$X_{b5}$	$X_{b6}$	$X_{b7}$	$X_{b8}$	$X_{b9}$

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	$X_{c1}$	$X_{c2}$	$X_{c3}$	$X_{c4}$	$X_{c5}$	$X_{c6}$	$X_{c7}$	$X_{c8}$

# Example: Learning HMM Parameters

- State output probability for S1
  - There are 13 observations in S1
  - Segregate them out and count
    - Compute parameters (mean and variance) of Gaussian output density for state S1



Time	1	2	6	7	9	10
state	S1	S1	S1	S1	S1	S1
Obs	$X_{a1}$	$X_{a2}$	$X_{a6}$	$X_{a7}$	$X_{a9}$	$X_{a10}$

$$P(X | S_1) = \frac{1}{\sqrt{(2\pi)^d |\Theta_1|}} \exp(-0.5(X - \mu_1)^T \Theta_1^{-1} (X - \mu_1))$$

Time	3	4	9
state	S1	S1	S1
Obs	$X_{b3}$	$X_{b4}$	$X_{b9}$

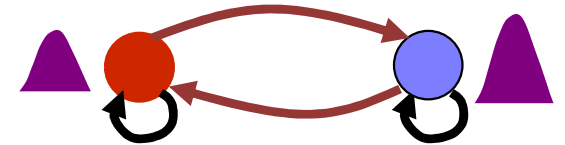
$$\mu_1 = \frac{1}{13} \begin{pmatrix} X_{a1} + X_{a2} + X_{a6} + X_{a7} + X_{a9} + X_{a10} + X_{b3} + \\ X_{b4} + X_{b9} + X_{c1} + X_{c2} + X_{c4} + X_{c5} \end{pmatrix}$$

Time	1	3	4	5
state	S1	S1	S1	S1
Obs	$X_{c1}$	$X_{c2}$	$X_{c4}$	$X_{c5}$

$$\Theta_1 = \frac{1}{13} \begin{pmatrix} (X_{a1} - \mu_1)(X_{a1} - \mu_1)^T + (X_{a2} - \mu_1)(X_{a2} - \mu_1)^T + \dots \\ (X_{b3} - \mu_1)(X_{b3} - \mu_1)^T + (X_{b4} - \mu_1)(X_{b4} - \mu_1)^T + \dots \\ (X_{c1} - \mu_1)(X_{c1} - \mu_1)^T + (X_{c2} - \mu_1)(X_{c2} - \mu_1)^T + \dots \end{pmatrix}$$

# Example: Learning HMM Parameters

- State output probability for S2
  - There are 14 observations in S2



Observation 1

Time	1	2	3	4	5	6	7	8	9	10
state	S1	S1	S2	S2	S2	S1	S1	S2	S1	S1
Obs	$X_{a1}$	$X_{a2}$	$X_{a3}$	$X_{a4}$	$X_{a5}$	$X_{a6}$	$X_{a7}$	$X_{a8}$	$X_{a9}$	$X_{a10}$

Observation 2

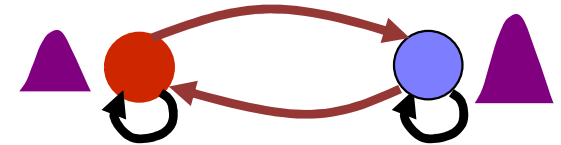
Time	1	2	3	4	5	6	7	8	9
state	S2	S2	S1	S1	S2	S2	S2	S2	S1
Obs	$X_{b1}$	$X_{b2}$	$X_{b3}$	$X_{b4}$	$X_{b5}$	$X_{b6}$	$X_{b7}$	$X_{b8}$	$X_{b9}$

Observation 3

Time	1	2	3	4	5	6	7	8
state	S1	S2	S1	S1	S1	S2	S2	S2
Obs	$X_{c1}$	$X_{c2}$	$X_{c3}$	$X_{c4}$	$X_{c5}$	$X_{c6}$	$X_{c7}$	$X_{c8}$

# Example: Learning HMM Parameters

- State output probability for S2
  - There are 14 observations in S2
  - Segregate them out and count
    - Compute parameters (mean and variance) of Gaussian output density for state S2



Time	3	4	5	8
state	S2	S2	S2	S2
Obs	$X_{a3}$	$X_{a4}$	$X_{a5}$	$X_{a8}$

$$P(X | S_2) = \frac{1}{\sqrt{(2\pi)^d |\Theta_2|}} \exp(-0.5(X - \mu_2)^T \Theta_2^{-1} (X - \mu_2))$$

Time	1	2	5	6	7	8
state	S2	S2	S2	S2	S2	S2
Obs	$X_{b1}$	$X_{b2}$	$X_{b5}$	$X_{b6}$	$X_{b7}$	$X_{b8}$

Time	2	6	7	8
state	S2	S2	S2	S2
Obs	$X_{c2}$	$X_{c6}$	$X_{c7}$	$X_{c8}$

$$\mu_2 = \frac{1}{14} \begin{pmatrix} X_{a3} + X_{a4} + X_{a5} + X_{a8} + X_{b1} + X_{b2} + X_{b5} + \\ X_{b6} + X_{b7} + X_{b8} + X_{c2} + X_{c6} + X_{c7} + X_{c8} \end{pmatrix}$$

$$\Theta_1 = \frac{1}{14} ((X_{a3} - \mu_2)(X_{a3} - \mu_2)^T + \dots)$$

# We have learnt all the HMM parameters

- State initial probabilities, often denoted as  $\pi$ 
  - $\pi(S1) = 0.66$        $\pi(S2) = 1/3 = 0.33$
- State transition probabilities

$$A = \begin{pmatrix} 0.545 & 0.455 \\ 0.385 & 0.615 \end{pmatrix}$$

- State output probabilities

State output probability for S1

$$P(X | S_1) = \frac{1}{\sqrt{(2\pi)^d |\Theta_1|}} \exp\left(-0.5(X - \mu_1)^T \Theta_1^{-1} (X - \mu_1)\right)$$

State output probability for S2

$$P(X | S_2) = \frac{1}{\sqrt{(2\pi)^d |\Theta_2|}} \exp\left(-0.5(X - \mu_2)^T \Theta_2^{-1} (X - \mu_2)\right)$$



# Update rules at each iteration

$$\pi(s_i) = \frac{\sum_{obs} \delta_{t=1}(s_i)}{N_{obs}}$$

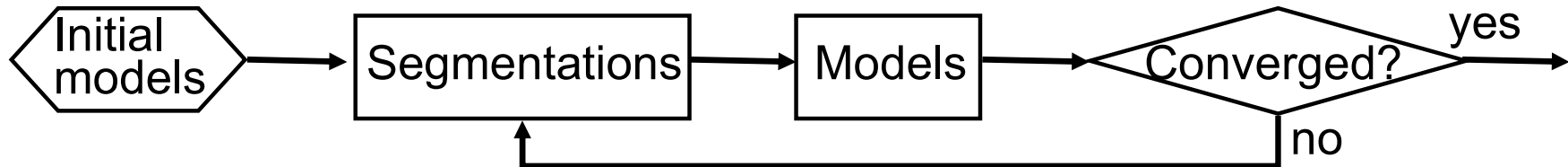
$$P(s_j | s_i) = \frac{\sum_{obs} \sum_{t=1}^{T-1} \delta_{t,t+1}(s_j | s_i)}{\sum_{obs} \sum_{t=1}^{T-1} \delta_t(s_i)}$$

$$\mu_i = \frac{\sum_{obs} \sum_{t=1}^T \delta_t(s_i) X_{obs}(t)}{\sum_{obs} \sum_{t=1}^T \delta_t(s_i)}$$

$$\Theta_i = \frac{\sum_{obs} \sum_{t=1}^T \delta_t(s_i) (X_{obs}(t) - \mu_i)(X_{obs}(t) - \mu_i)^T}{\sum_{obs} \sum_{t=1}^T \delta_t(s_i)}$$

- Assumes state output PDF = Gaussian
  - For GMMs, estimate GMM parameters from collection of observations at any state

# Training by segmentation: Viterbi training



- Initialize all HMM parameters
- Segment all training observation sequences into states using the Viterbi algorithm with the current models
- Using estimated state sequences and training observation sequences, reestimate the HMM parameters
- This method is also called a “segmental k-means” learning procedure

# Poll 1

# Alternative to counting: SOFT counting

- Expectation maximization
- *Every* observation contributes to every state

# Update rules at each iteration

$$\pi(s_i) = \frac{\sum_{obs} \delta_{t=1}(s_i)}{N_{obs}}$$

$$P(s_j | s_i) = \frac{\sum_{obs} \sum_{t=1}^{T-1} \delta_{t,t+1}(s_j | s_i)}{\sum_{obs} \sum_{t=1}^{T-1} \delta_t(s_i)}$$

$$\mu_i = \frac{\sum_{obs} \sum_{t=1}^T \delta_t(s_i) X_{obs}(t)}{\sum_{obs} \sum_{t=1}^T \delta_t(s_i)}$$

$$\Theta_i = \frac{\sum_{obs} \sum_{t=1}^T \delta_t(s_i) (X_{obs}(t) - \mu_i)(X_{obs}(t) - \mu_i)^T}{\sum_{obs} \sum_{t=1}^T \delta_t(s_i)}$$

- Assumes state output PDF = Gaussian
  - For GMMs, estimate GMM parameters from collection of observations at any state

# Update rules at each iteration

$$\pi(s_i) = \frac{\sum_{Obs} P(state(t=1) = s_i | Obs)}{\text{Total no. of observation sequences}}$$

$$P(s_j | s_i) = \frac{\sum_{Obs} \sum_t P(state(t) = s_i, state(t+1) = s_j | Obs)}{\sum_{Obs} \sum_t P(state(t) = s_i | Obs)}$$

$$\mu_i = \frac{\sum_{Obs} \sum_t P(state(t) = s_i | Obs) X_{Obs,t}}{\sum_{Obs} \sum_t P(state(t) = s_i | Obs)}$$

$$\Theta_i = \frac{\sum_{Obs} \sum_t P(state(t) = s_i | Obs) (X_{Obs,t} - \mu_i)(X_{Obs,t} - \mu_i)^T}{\sum_{Obs} \sum_t P(state(t) = s_i | Obs)}$$

- Every observation contributes to every state

# Poll 2

# Update rules at each iteration

$$\pi(s_i) = \frac{\sum_{Obs} P(state(t=1) = s_i | Obs)}{\text{Total no. of observation sequences}}$$

$$P(s_j | s_i) = \frac{\sum_{Obs} \sum_t P(state(t) = s_i, state(t+1) = s_j | Obs)}{\sum_{Obs} \sum_t P(state(t) = s_i | Obs)}$$

$$\mu_i = \frac{\sum_{Obs} \sum_t P(state(t) = s_i | Obs) X_{Obs,t}}{\sum_{Obs} \sum_t P(state(t) = s_i | Obs)}$$

$$\Theta_i = \frac{\sum_{Obs} \sum_t P(state(t) = s_i | Obs) (X_{Obs,t} - \mu_i)(X_{Obs,t} - \mu_i)^T}{\sum_{Obs} \sum_t P(state(t) = s_i | Obs)}$$

- Where did these terms come from?



$$P(state(t) = s \mid Obs)$$

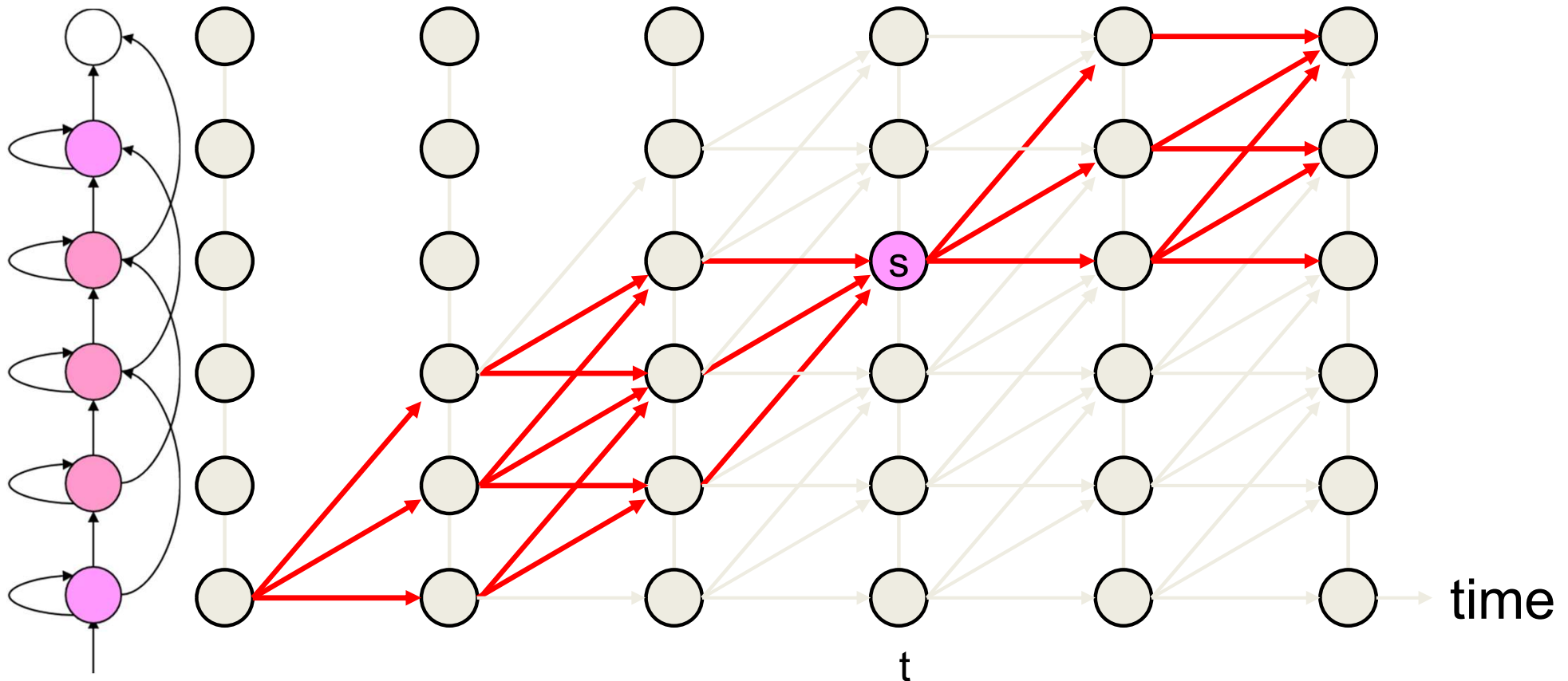
- The probability that the process was at  $s$  when it generated  $X_t$  given the entire observation
  - Dropping the “Obs” subscript for brevity

$$P(state(t) = s \mid X_1, X_2, \dots, X_T) \propto P(state(t) = s, X_1, X_2, \dots, X_T)$$

- We will compute  $P(state(t) = s_i, x_1, x_2, \dots, x_T)$  first
  - This is the probability that the process visited  $s$  at time  $t$  while producing the entire observation

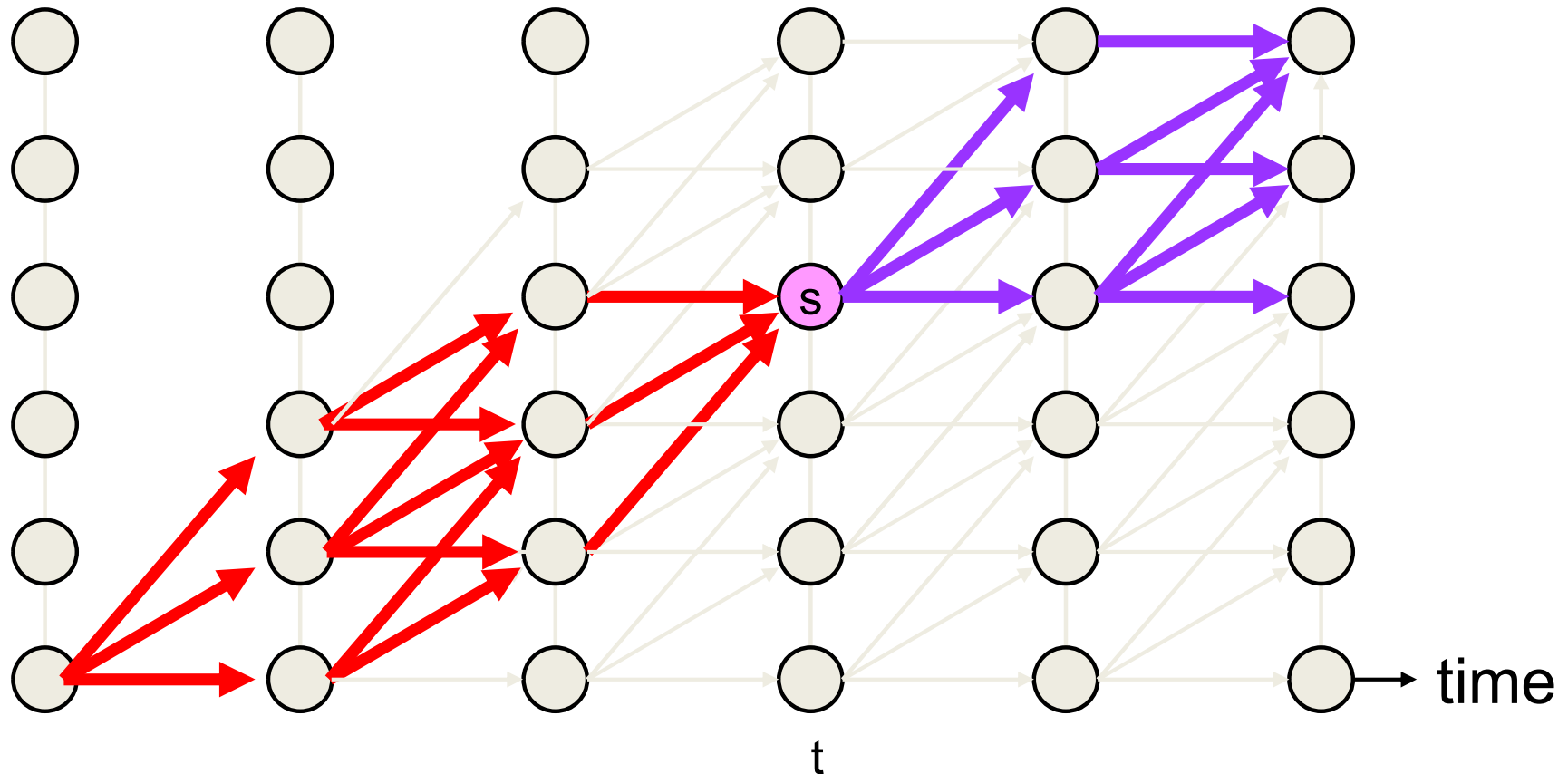
$$P(\text{state}(t) = s, x_1, x_2, \dots, x_T)$$

- The probability that the HMM was in a particular state  $s$  when generating the observation sequence is the probability that it followed a state sequence that passed through  $s$  at time  $t$



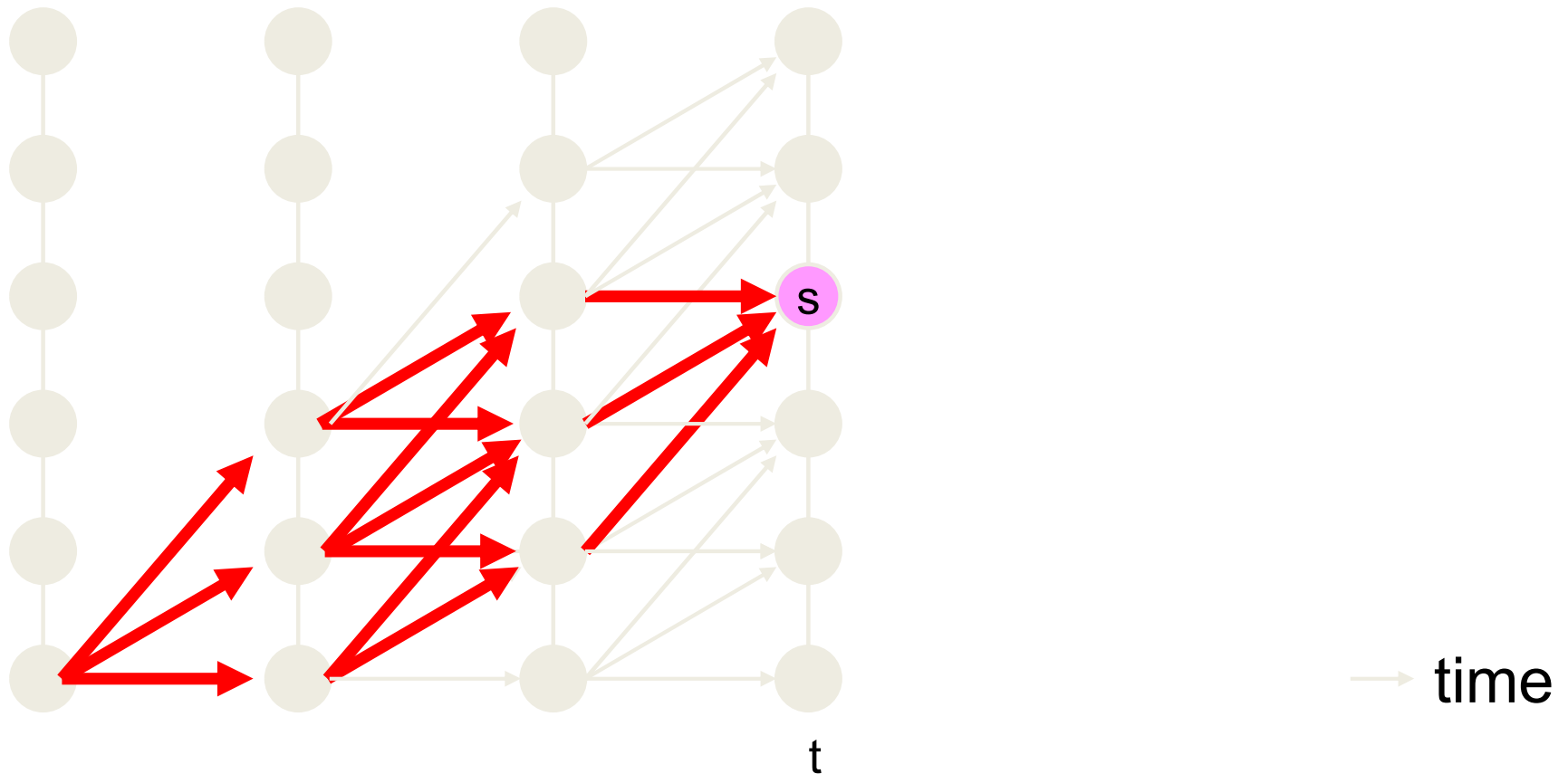
$$P(\text{state}(t) = s, x_1, x_2, \dots, x_T)$$

- This can be decomposed into two multiplicative sections
  - The section of the lattice leading into state  $s$  at time  $t$  and the section leading out of it



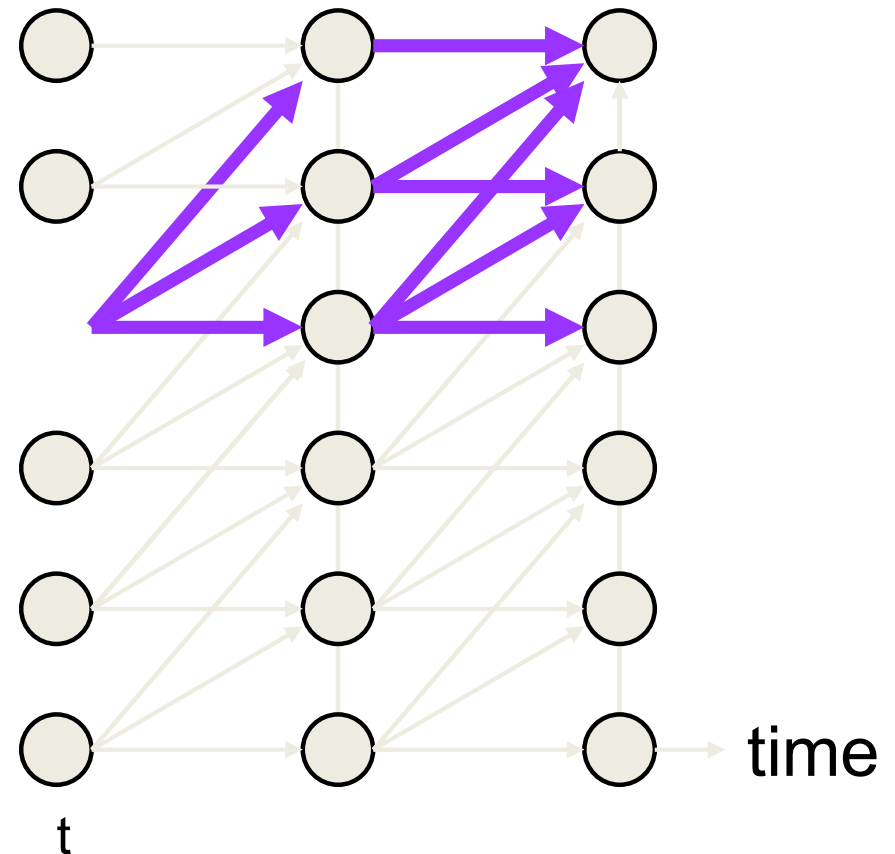
# The Forward Paths

- The probability of the red section is the total probability of all state sequences ending at state  $s$  at time  $t$ 
  - This is simply  $\alpha(s, t)$
  - Can be computed using the forward algorithm



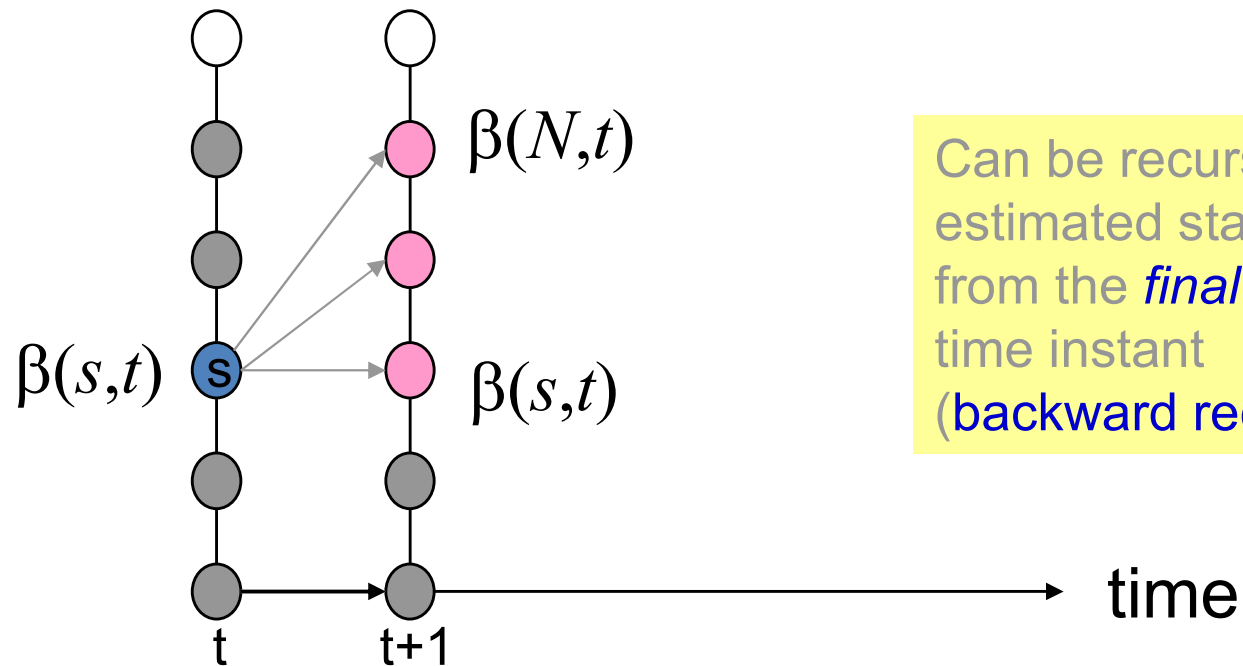
# The Backward Paths

- The blue portion represents the probability of all state sequences that began at state  $s$  at time  $t$ 
  - Like the red portion it can be computed using a *backward recursion*



# The Backward Recursion

$$\beta(s, t) = P(x_{t+1}, x_{t+2}, \dots, x_T \mid \text{state}(t) = s)$$

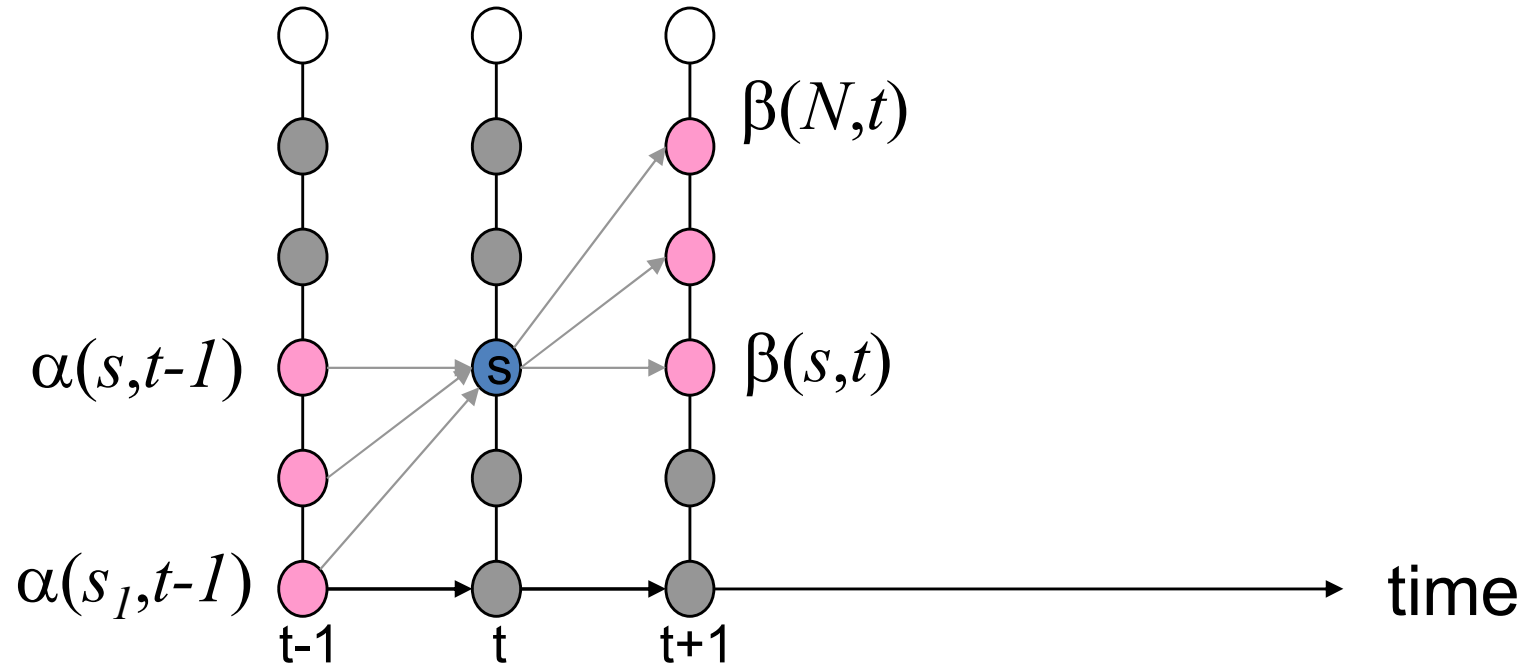


$$\beta(s, t) = \sum_{s'} \beta(s', t+1) P(s' \mid s) P(x_{t+1} \mid s')$$

- $\beta(s, t)$  is the total probability of ALL state sequences that depart from  $s$  at time  $t$ , and all observations after  $x_t$ 
  - $\beta(s, T) = 1$  at the final time instant for all valid final states

# The complete probability

$$\alpha(s, t) \beta(s, t) = P(x_{t+1}, x_{t+2}, \dots, x_T, \text{state}(t) = s)$$



# Poll 3



# Posterior probability of a state

- The probability that the process was in state  $s$  at time  $t$ , given that we have observed the data is obtained by simple normalization

$$P(state(t) = s | Obs) = \frac{P(state(t) = s, x_1, x_2, \dots, x_T)}{\sum_{s'} P(state(t) = s, x_1, x_2, \dots, x_T)} = \frac{\alpha(s, t) \beta(s, t)}{\sum_{s'} \alpha(s', t) \beta(s', t)}$$

- This term is often referred to as the gamma term and denoted by  $\gamma_{s,t}$

# Update rules at each iteration

$$\pi(s_i) = \frac{\sum_{Obs} P(state(t=1) = s_i | Obs)}{\text{Total no. of observation sequences}}$$

$$P(s_j | s_i) = \frac{\sum_{Obs} \sum_t P(state(t) = s_i, state(t+1) = s_j | Obs)}{\sum_{Obs} \sum_t P(state(t) = s_i | Obs)}$$

$$\mu_i = \frac{\sum_{Obs} \sum_t P(state(t) = s_i | Obs) X_{Obs,t}}{\sum_{Obs} \sum_t P(state(t) = s_i | Obs)}$$

$$\Theta_i = \frac{\sum_{Obs} \sum_t P(state(t) = s_i | Obs) (X_{Obs,t} - \mu_i)(X_{Obs,t} - \mu_i)^T}{\sum_{Obs} \sum_t P(state(t) = s_i | Obs)}$$

- These have been found

# Update rules at each iteration

$$\pi(s_i) = \frac{\sum_{Obs} P(state(t=1) = s_i | Obs)}{\text{Total no. of observation sequences}}$$

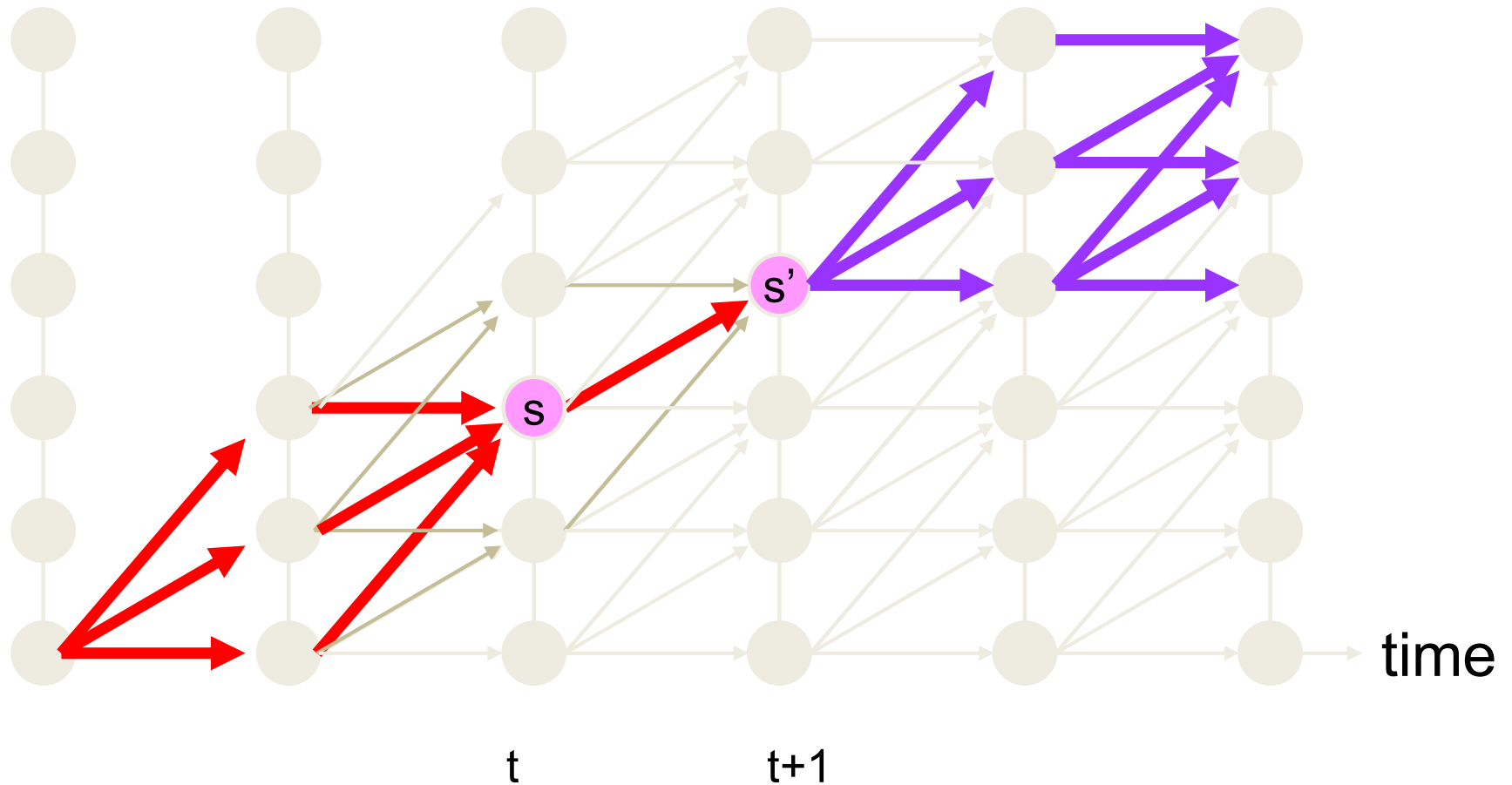
$$P(s_j | s_i) = \frac{\sum_{Obs} \sum_t P(state(t) = s_i, state(t+1) = s_j | Obs)}{\sum_{Obs} \sum_t P(state(t) = s_i | Obs)}$$

$$\mu_i = \frac{\sum_{Obs} \sum_t P(state(t) = s_i | Obs) X_{Obs,t}}{\sum_{Obs} \sum_t P(state(t) = s_i | Obs)}$$

$$\Theta_i = \frac{\sum_{Obs} \sum_t P(state(t) = s_i | Obs) (X_{Obs,t} - \mu_i)(X_{Obs,t} - \mu_i)^T}{\sum_{Obs} \sum_t P(state(t) = s_i | Obs)}$$

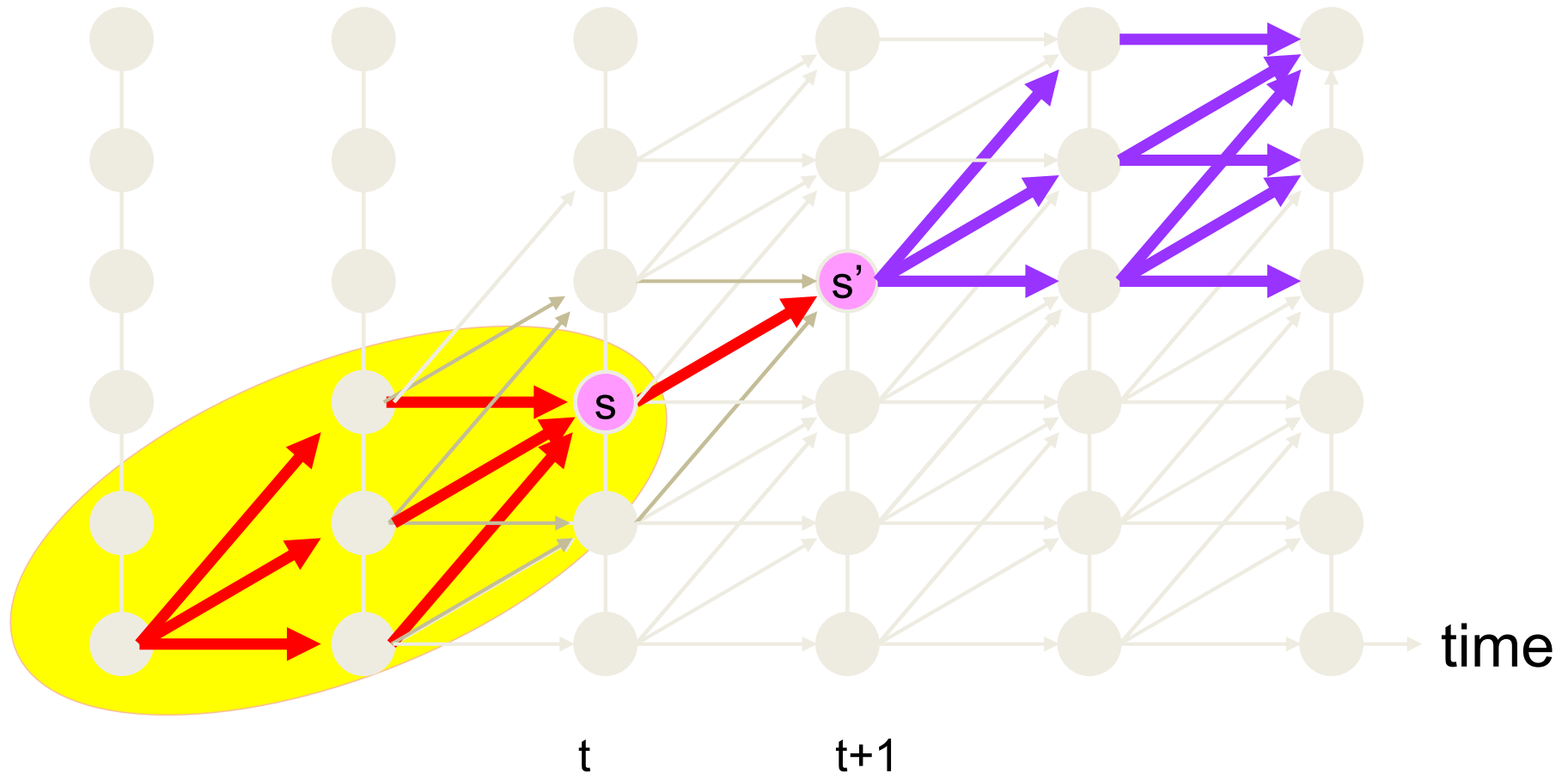
- Where did these terms come from?

$$P(\text{state}(t) = s, \text{state}(t+1) = s', x_1, x_2, \dots, x_T)$$



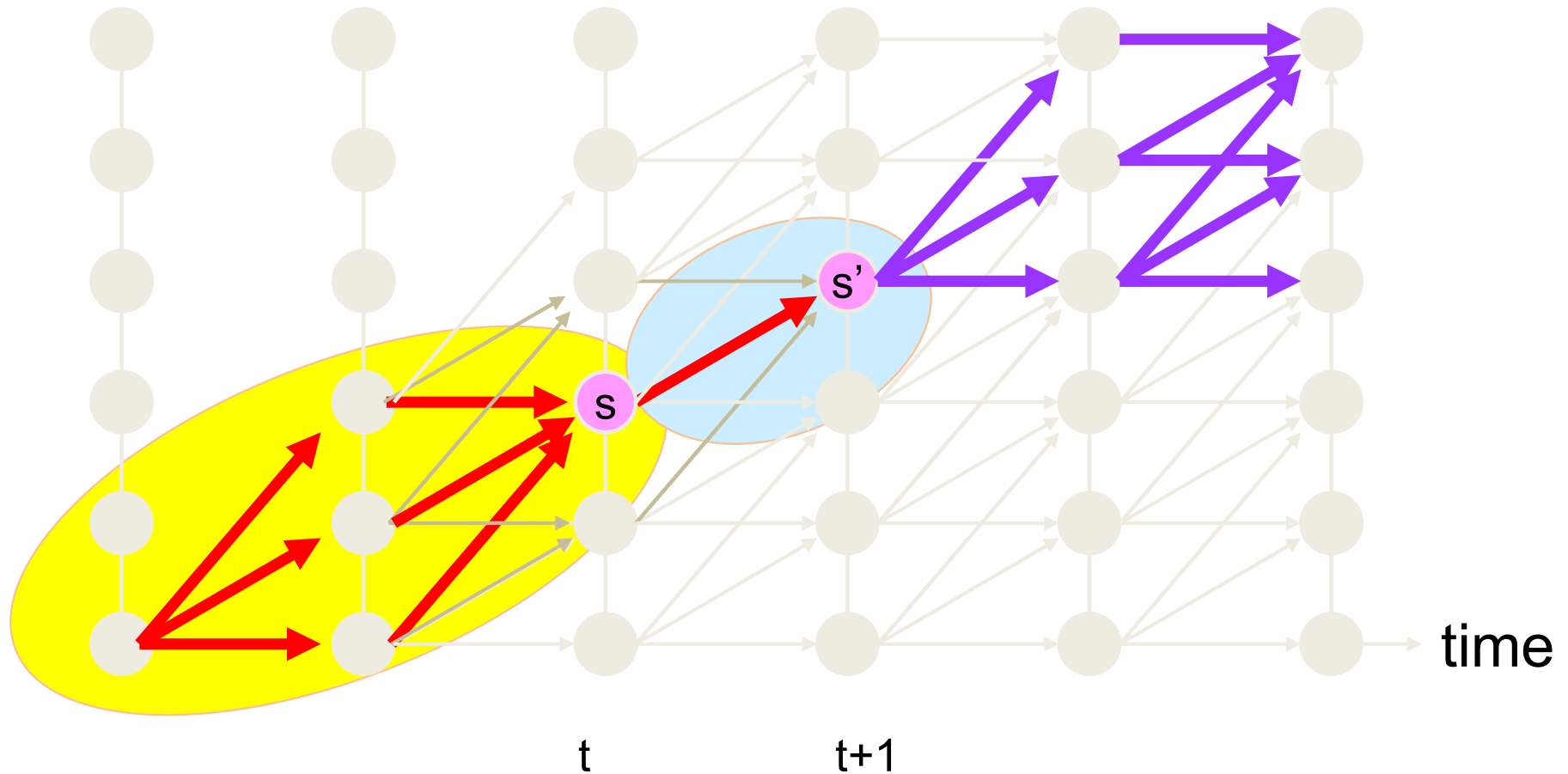
$$P(\text{state}(t) = s, \text{state}(t+1) = s', x_1, x_2, \dots, x_T)$$

$$\alpha(s, t)$$



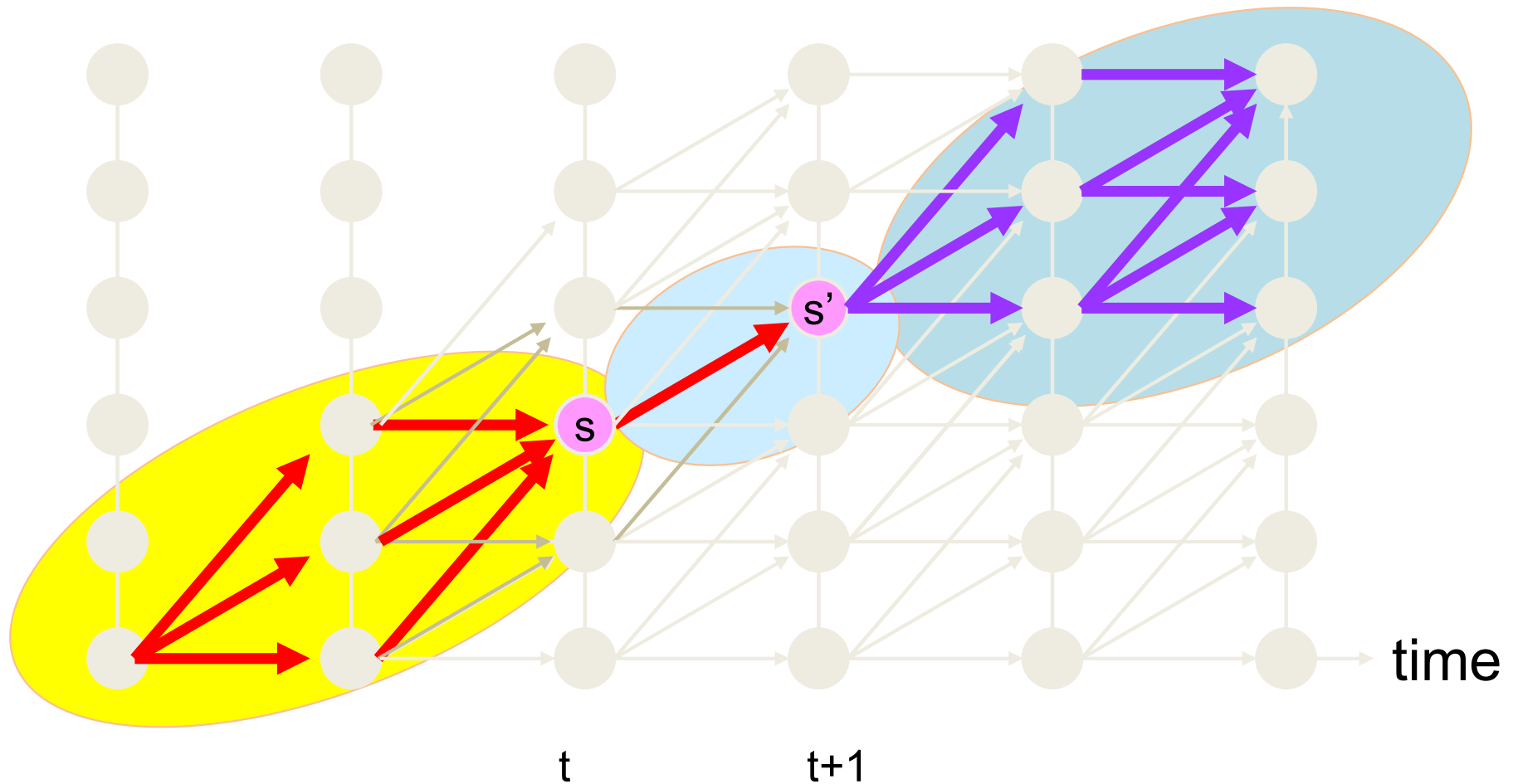
$$P(\text{state}(t) = s, \text{state}(t+1) = s', x_1, x_2, \dots, x_T)$$

$$\alpha(s, t) P(s' | s) P(x_{t+1} | s')$$



$$P(\text{state}(t) = s, \text{state}(t+1) = s', x_1, x_2, \dots, x_T)$$

$$\alpha(s, t) P(s' | s) P(x_{t+1} | s') \beta(s', t+1)$$



# The a posteriori probability of transition

$$P(state(t) = s, state(t+1) = s' | Obs) = \frac{\alpha(s, t) P(s' | s) P(x_{t+1} | s') \beta(s', t+1)}{\sum_{s_1} \sum_{s_2} \alpha(s_1, t) P(s_2 | s_1) P(x_{t+1} | s_2) \beta(s_2, t+1)}$$

- The a posteriori probability of a transition given an observation



# Update rules at each iteration

$$\pi(s_i) = \frac{\sum_{Obs} P(state(t=1) = s_i | Obs)}{\text{Total no. of observation sequences}}$$

$$P(s_j | s_i) = \frac{\sum_{Obs} \sum_t P(state(t) = s_i, state(t+1) = s_j | Obs)}{\sum_{Obs} \sum_t P(state(t) = s_i | Obs)}$$

$$\mu_i = \frac{\sum_{Obs} \sum_t P(state(t) = s_i | Obs) X_{Obs,t}}{\sum_{Obs} \sum_t P(state(t) = s_i | Obs)}$$

$$\Theta_i = \frac{\sum_{Obs} \sum_t P(state(t) = s_i | Obs) (X_{Obs,t} - \mu_i)(X_{Obs,t} - \mu_i)^T}{\sum_{Obs} \sum_t P(state(t) = s_i | Obs)}$$

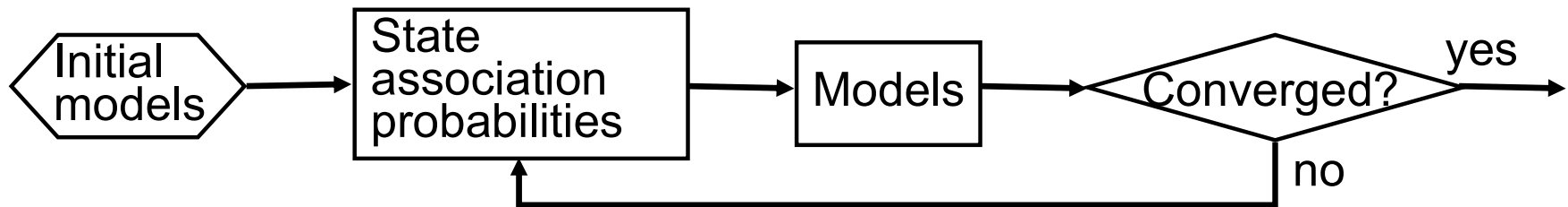
- These have been found

# Poll 4

# Training without explicit segmentation: MLSP

## Baum-Welch training

- Every feature vector associated with every state of every HMM with a probability



- Probabilities computed using the forward-backward algorithm
- Soft decisions taken at the level of HMM state
- In practice, the segmentation based Viterbi training is much easier to implement and is much faster
- The difference in performance between the two is small, especially if we have lots of training data

# HMM Issues

- How to find the best state sequence: Covered
- How to learn HMM parameters: Covered
- How to compute the probability of an observation sequence: Covered

# Magic numbers

- How many states:
  - No nice automatic technique to learn this
  - You choose
    - For speech, HMM topology is usually left to right (no backward transitions)
    - For other cyclic processes, topology must reflect nature of process
    - No. of states – 3 per phoneme in speech
    - For other processes, depends on estimated no. of distinct states in process

# Applications of HMMs

- Classification:
  - Learn HMMs for the various classes of time series from training data
  - Compute probability of test time series using the HMMs for each class
  - Use in a Bayesian classifier
  - Speech recognition, vision, gene sequencing, character recognition, text mining...
- Prediction
- Tracking

# Applications of HMMs

- Segmentation:
  - Given HMMs for various events, find event boundaries
    - Simply find the best state sequence and the locations where state identities change
- Automatic speech segmentation, text segmentation by topic, genome segmentation, ...