Hidden Markov Models

Eklavya SARKAR
Biometrics Security and Privacy, Idiap

Table of Contents

Introduction

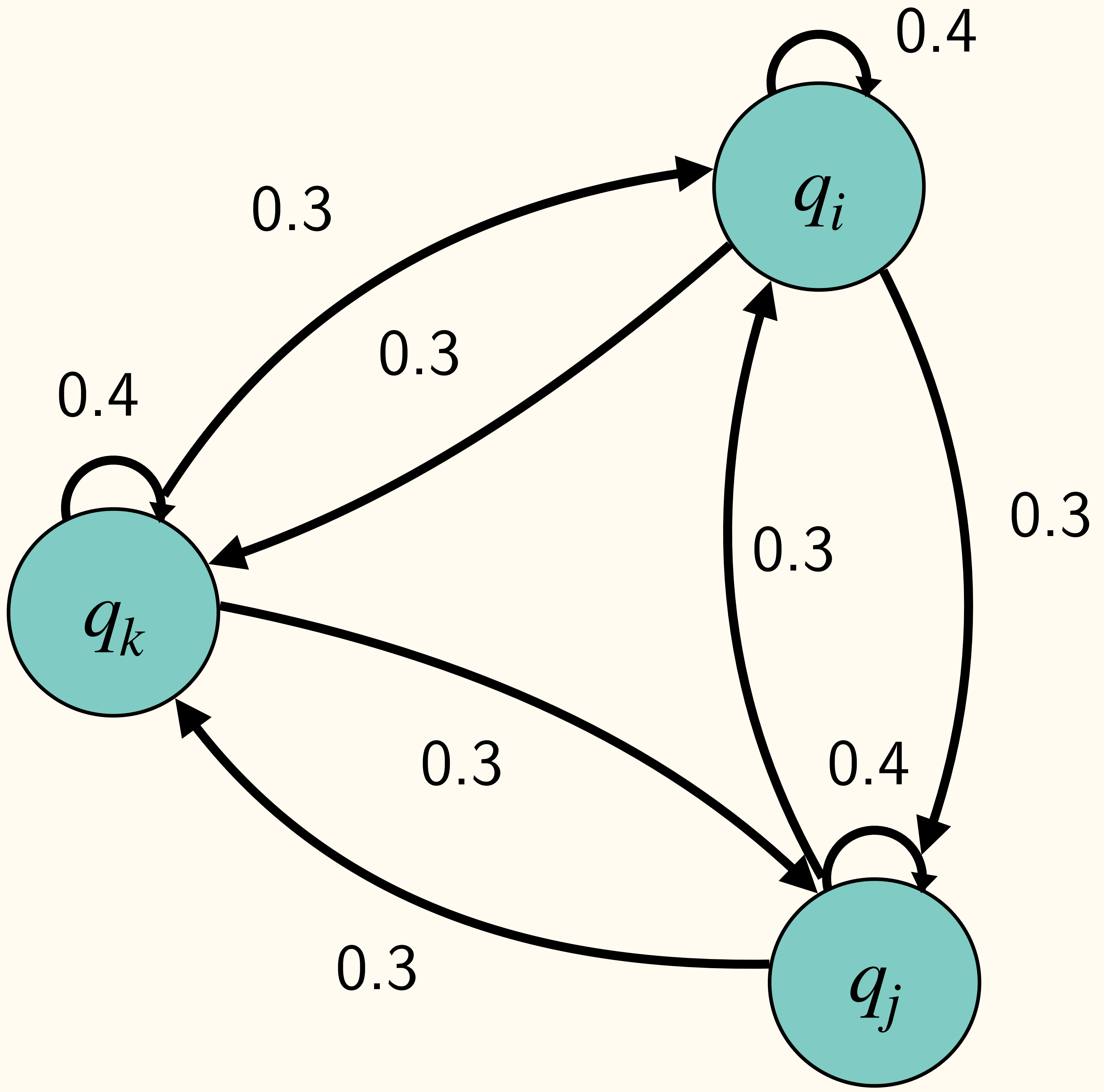
L. R. Rabiner

- Sequence processing:
  - Input: sequence X
  - Goal: estimate a sequence of outputs M
  - $P(M|X)$

- Tool: Hidden Markov Models (HMMs)
  - Introduced and studied in 1960-70s
  - Lawrence R. Rabiner. *A tutorial on Hidden Markov Models and selected applications in speech recognition.*

Discrete Markov Models (DMMs)

- Model M_k
- Composed of states $Q = \{q_1, \ldots, q_k, \ldots, q_K\}$
  - q_j^t denotes state q_j at time t
- Transition probabilities:
  - $$A = \{a_{ij}\} = \frac{C(i \to j)}{\sum_k C(i \to k)}$$
- First-order Markov Models
- Time independent
- $X = \{\}$

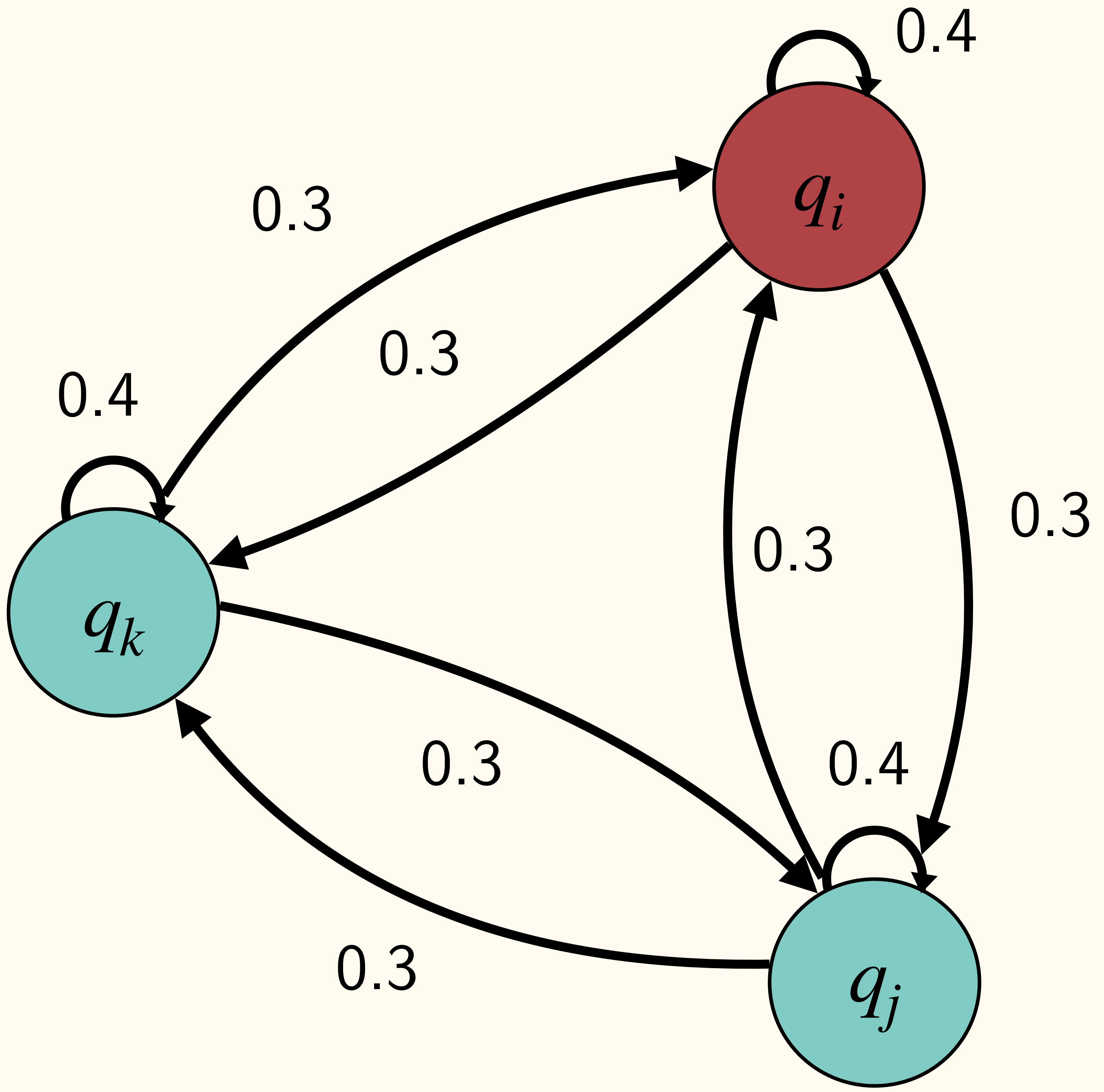


Discrete Markov Models (DMMs)

- Model M_k
- Composed of states $Q = \{q_1, \ldots, q_k, \ldots, q_K\}$
  - q_j^t denotes state q_j at time t
- Transition probabilities $A = \{a_{ij}\}$

- First-order Markov Models
- Time independent

- $X = \{q_i\}$

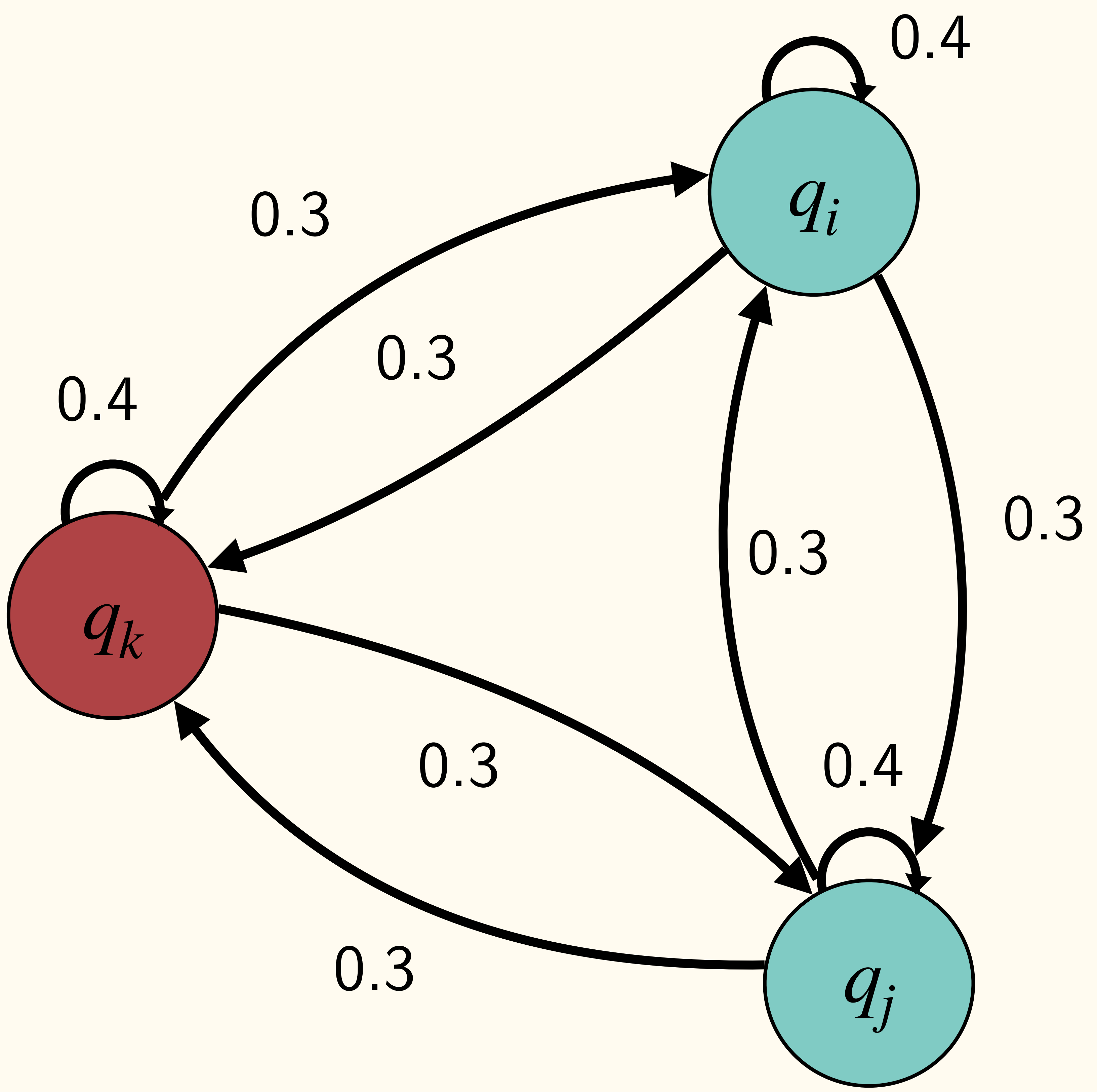


Discrete Markov Models (DMMs)

- Model M_k
- Composed of states $Q = \{q_1, \ldots, q_k, \ldots, q_K\}$
  - q_j^t denotes state q_j at time t
- Transition probabilities $A = \{a_{ij}\}$

- First-order Markov Models
- Time independent

- $X = \{q_i, q_k\}$

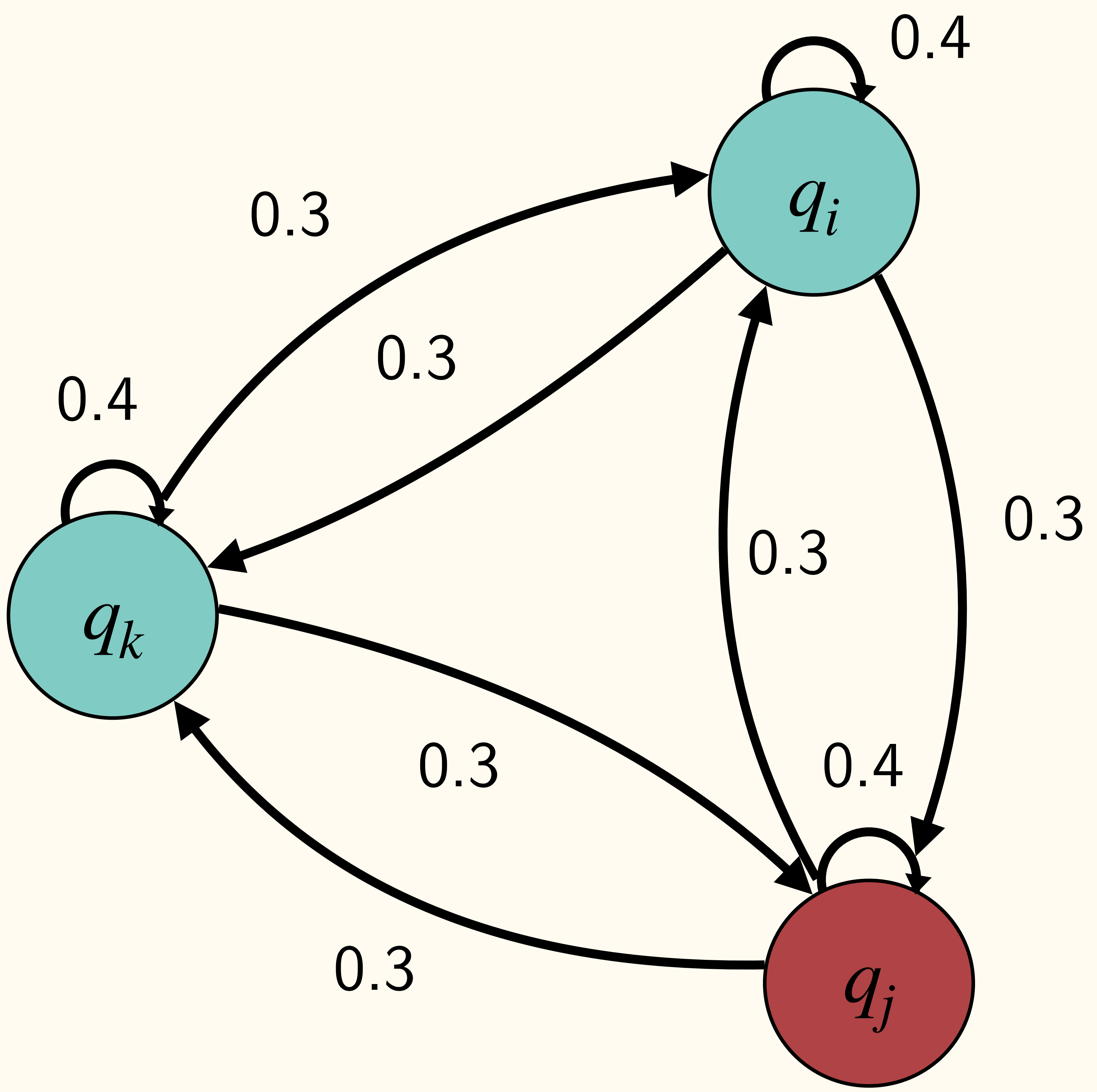


Discrete Markov Models (DMMs)

- Model M_k
- Composed of states $Q = \{q_1, \ldots, q_k, \ldots, q_K\}$
  - q_j^t denotes state q_j at time t
- Transition probabilities $A = \{a_{ij}\}$

- First-order Markov Models
- Time independent

- $X = \{q_i, q_k, q_j\}$

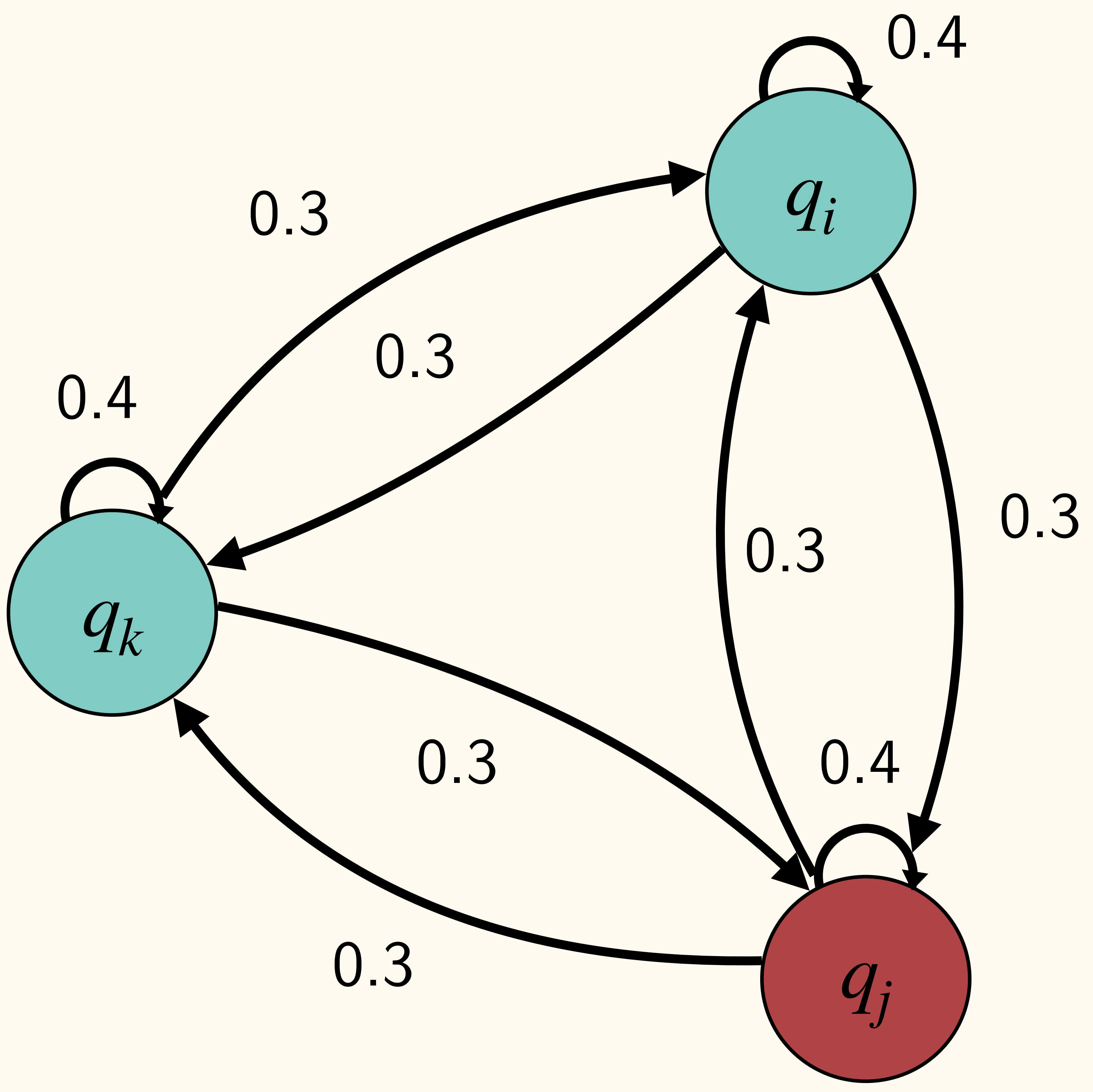


Discrete Markov Models (DMMs)

- Model M_k
- Composed of states $Q = \{q_1, \ldots, q_k, \ldots, q_K\}$
  - q_j^t denotes state q_j at time t
- Transition probabilities $A = \{a_{ij}\}$

- First-order Markov Models
- Time independent

- $X = \{q_i, q_k, q_j, q_j\}$

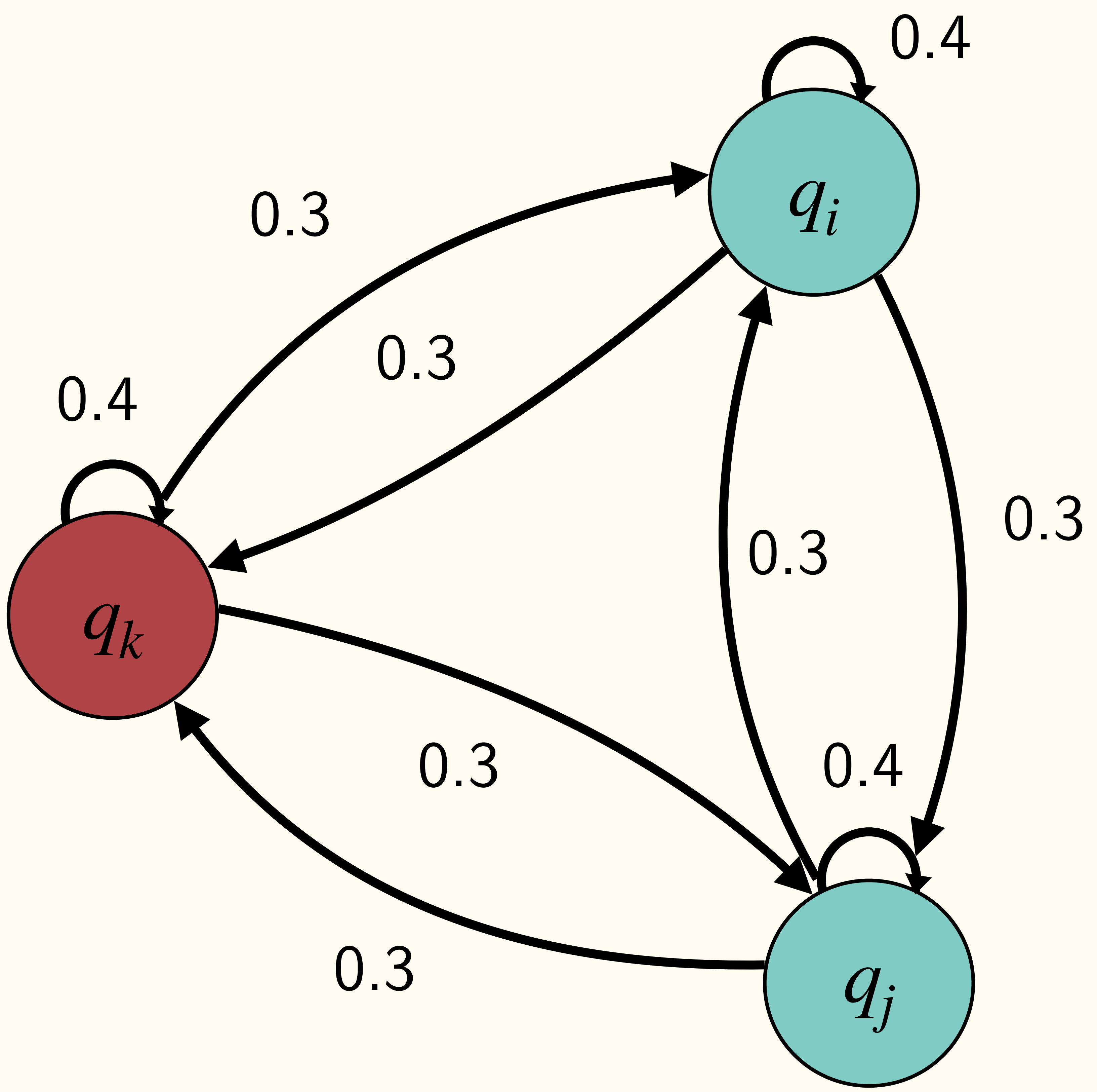


Discrete Markov Models (DMMs)

- Model M_k
- Composed of states $Q = \{q_1, \ldots, q_k, \ldots, q_K\}$
  - q_j^t denotes state q_j at time t
- Transition probabilities $A = \{a_{ij}\}$

- First-order Markov Models
- Time independent

- $X = \{q_i, q_k, q_j, q_j, q_k\}$

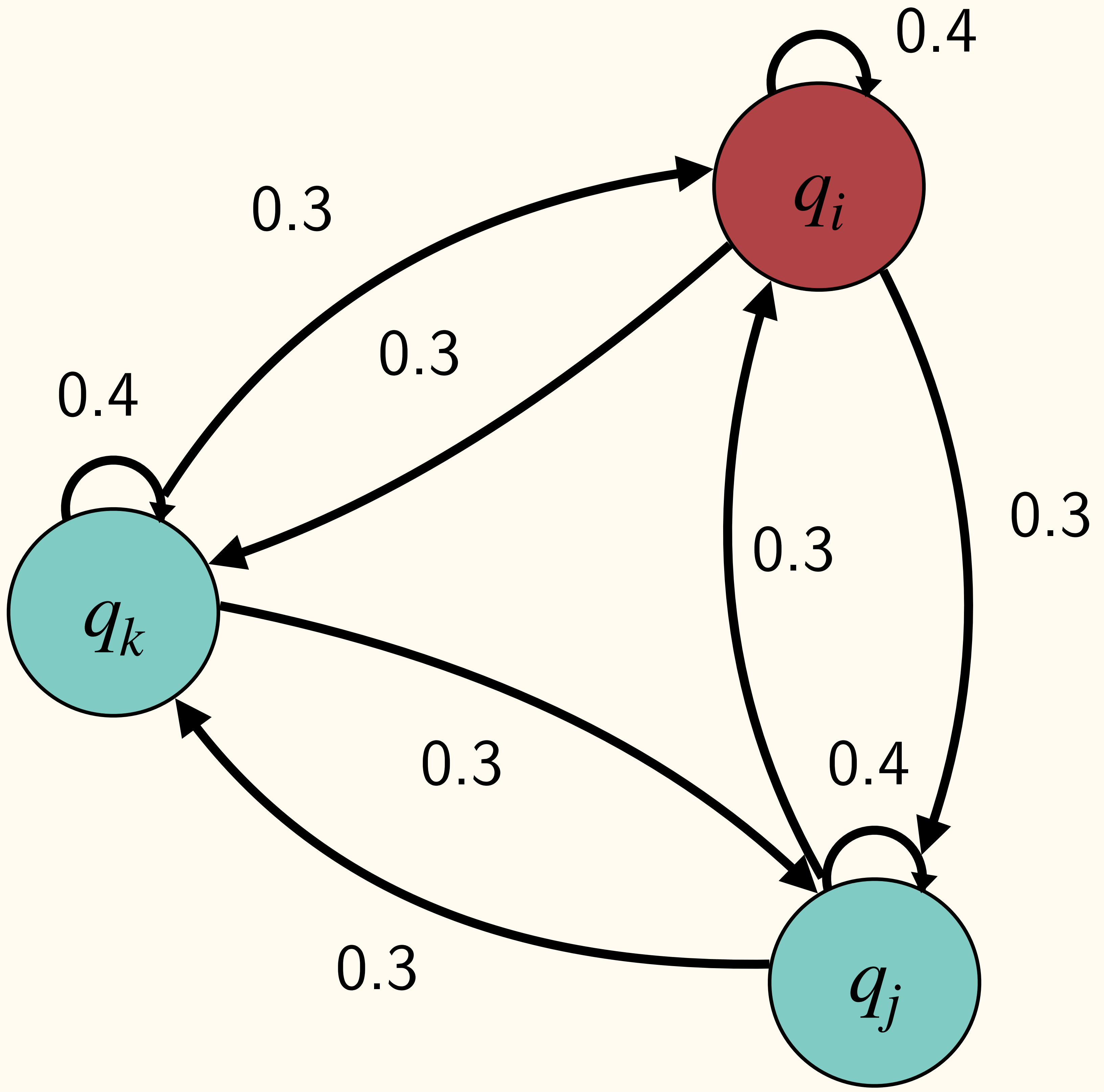


Discrete Markov Models (DMMs)

- Model M_k
- Composed of states $Q = \{q_1, \ldots, q_k, \ldots, q_K\}$
  - q_j^t denotes state q_j at time t
- Transition probabilities $A = \{a_{ij}\}$

- First-order Markov Models
- Time independent

- $X = \{q_i, q_k, q_j, q_j, q_k, q_i\}$



DMMs - Sequence Probability

$X = \{q_i, q_k, q_j, q_j, q_k, q_i\}$

$$
\begin{aligned}
&P(X \mid M) \\
&= P(q_i, q_k, q_j, q_j, q_k, q_i \mid M) \\
&= P(q_i \mid q_i, q_k, q_j, q_j, q_k) \cdot P(q_i, q_k, q_j, q_j, q_k) \\
&= P(q_i \mid q_i, q_k, q_j, q_j, q_k) \cdot P(q_k \mid q_i, q_k, q_j, q_j) \cdot P(q_i, q_k, q_j, q_j) \\
&= P(q_i \mid q_i, q_k, q_j, q_j, q_k) \cdot P(q_k \mid q_i, q_k, q_j, q_j) \cdot P(q_j \mid q_i, q_k, q_j) \cdot P(q_i, q_k, q_j) \\
&= P(q_i \mid q_i, q_k, q_j, q_j, q_k) \cdot P(q_k \mid q_i, q_k, q_j, q_j) \cdot P(q_j \mid q_i, q_k, q_j) \cdot P(q_j \mid q_i, q_k) \cdot P(q_i, q_k) \\
&= P(q_i \mid q_i, q_k, q_j, q_j, q_k) \cdot P(q_k \mid q_i, q_k, q_j, q_j) \cdot P(q_j \mid q_i, q_k, q_j) \cdot P(q_j \mid q_i, q_k) \cdot P(q_k \mid q_i) \cdot P(q_i)
\end{aligned}
$$

DMMs - Sequence Probability

$X = \{q_i, q_k, q_j, q_j, q_k, q_i\}$

$$
\begin{aligned}
&P(X \mid M) \\
&= P(q_i, q_k, q_j, q_j, q_k, q_i \mid M) \\
&= P(q_i \mid q_i, q_k, q_j, q_j, q_k) \cdot P(q_i, q_k, q_j, q_j, q_k) \\
&= P(q_i \mid q_i, q_k, q_j, q_j, q_k) \cdot P(q_k \mid q_i, q_k, q_j, q_j) \cdot P(q_i, q_k, q_j, q_j) \\
&= P(q_i \mid q_i, q_k, q_j, q_j, q_k) \cdot P(q_k \mid q_i, q_k, q_j, q_j) \cdot P(q_j \mid q_i, q_k, q_j) \cdot P(q_i, q_k, q_j) \\
&= P(q_i \mid q_i, q_k, q_j, q_j, q_k) \cdot P(q_k \mid q_i, q_k, q_j, q_j) \cdot P(q_j \mid q_i, q_k, q_j) \cdot P(q_j \mid q_i, q_k) \cdot P(q_i, q_k) \\
&= P(q_i \mid q_i, q_k, q_j, q_j, q_k) \cdot P(q_k \mid q_i, q_k, q_j, q_j) \cdot P(q_j \mid q_i, q_k, q_j) \cdot P(q_j \mid q_i, q_k) \cdot P(q_k \mid q_i) \cdot P(q_i)
\end{aligned}
$$

DMMs - Sequence Probability

$$P(q_i \mid \cancel{q_i, q_k, q_j, q_j}, q_k) \cdot P(q_k \mid \cancel{q_i, q_k, q_j}, q_j) \cdot P(q_j \mid \cancel{q_i, q_k}, q_j) \cdot P(q_j \mid \cancel{q_i}, q_k) \cdot P(q_k \mid q_i) \cdot P(q_i)$$

First-Order Markov Property

$X = \{q_i, q_k, q_j, q_j, q_k, q_i\}$

$$\Rightarrow P(q_i \mid q_i) \cdot P(q_k \mid q_j) \cdot P(q_j \mid q_j) \cdot P(q_j \mid q_k) \cdot P(q_k \mid q_i) \cdot P(q_i)$$

Only need transition probabilities

$$= a_{ii} \cdot a_{kj} \cdot a_{jj} \cdot a_{jk} \cdot a_{ki} \cdot \pi_{q_i}$$

$$= 0.4 \cdot 0.3 \cdot 0.4 \cdot 0.3 \cdot 0.3 \cdot 1$$

$$= 0.00432$$

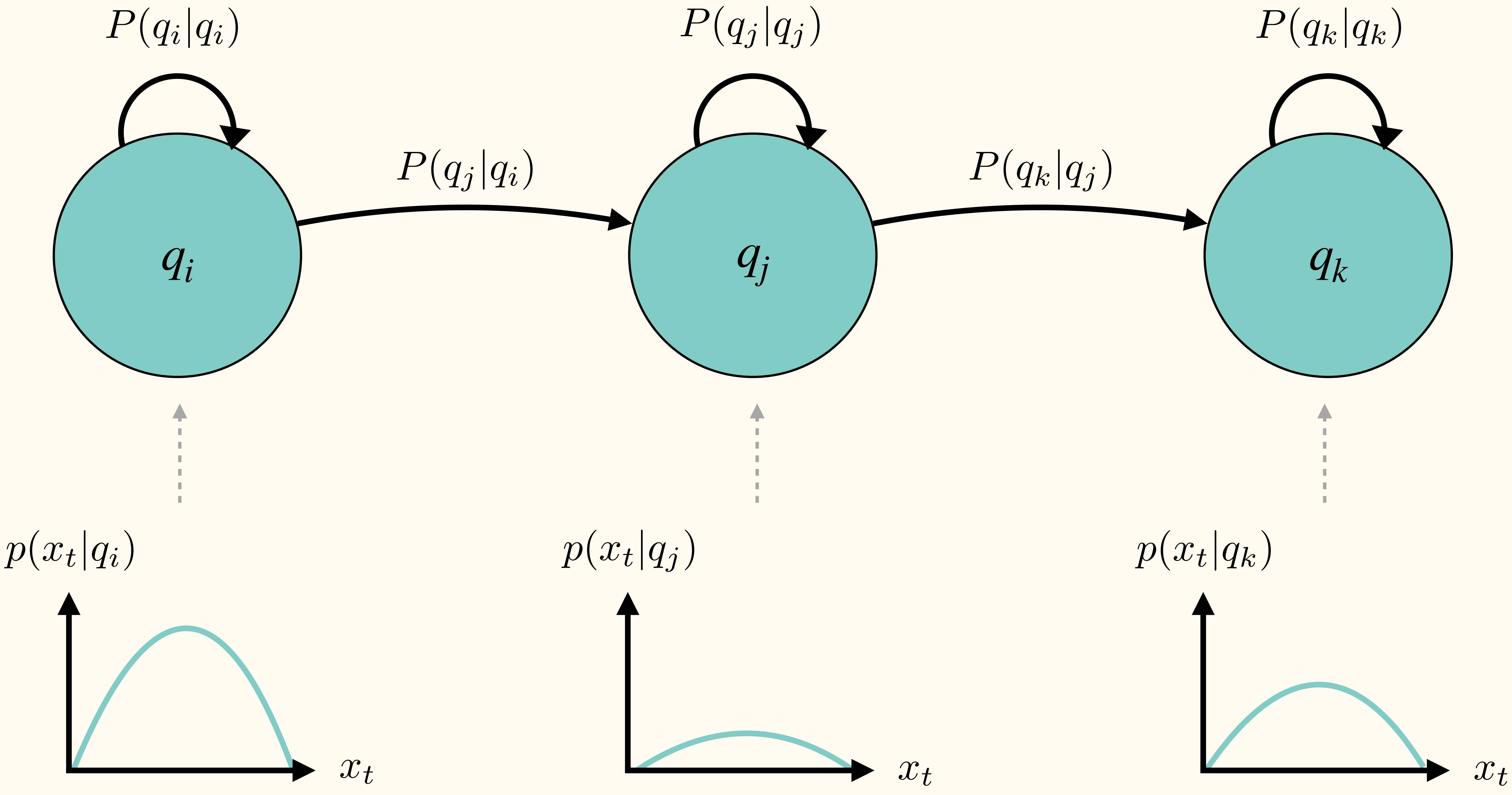
$$A = \{a_{ij}\} = \begin{bmatrix} 0.4 & 0.3 & 0.3 \\ 0.3 & 0.4 & 0.3 \\ 0.4 & 0.3 & 0.4 \end{bmatrix}$$

DMMs - Consecutive Sequence Probability

Given the model M is in a known state, what is the probability it stays in the same state for exactly d days ?

- $X = \{q_i^1, q_j^2, q_j^3, \ldots, q_i^d, q_j^{d+1} \neq q_i\}$
- Discrete probability density function of duration d in state i:
  - $P(X|M, q^1 = q_i) = (a_{ii})^{d-1} \cdot (1 - a_{ii}) = p_i(d)$
- Expected number of observations (duration) in a state:
  - $\bar{d} = \sum_{d=1}^{\infty} d p_i(d) = \sum_{d=1}^{\infty} d \cdot (a_{ii})^{d-1} \cdot (1 - a_{ii}) = \dfrac{1}{1 - a_{ii}}$

Hidden Markov Models (HMMs)



HMMs

- Sequence of observations: $X = \{x_1, \ldots, x_t, \ldots, x_T\}$
- Sequence of states: $Q = \{q_1, \ldots, q_k, \ldots, q_K\}$, q_j^t is state a q_j at time t
- Transition probabilities: $A = \{a_{ij}\} : a_{ij} = P(q_j^{t+1} | q_i^t), \qquad 1 \leq i, j \leq K$
- Emission probability distribution: $B = \{b_j(k)\} : b_j(k) = p(v_k^t | q_j^t), \qquad 1 \leq j \leq K$
- Initial state distribution: $\pi = \{\pi_i\} : \pi_i = P(q_i^1), \qquad 1 \leq j \leq K$

$$\Theta = \{\pi, A, B\}$$

- Observations now also described by emission probabilities, characterized by different stochastic distributions for each state q_i, $i \in [1, \ldots, K]$.
    - Discrete, Gaussians, GMMs, ANNs (MLPs, or RNNs).

HMMs - Steps

1. Choose an initial state $q_1 = q_i$ according to initial state distribution π.
2. Set $t = 1$.
3. Choose $X_t = v_k$ according to emission probability distribution in state q_i i.e. $b_i(k)$.
4. Transit to a new state q_j^{t+1} according to state transition probabilities i.e. a_{ij}.
5. Set $t = t + 1$
   - If $t < T$:
     - Return to step 3)
   - Else:
     - Terminate.

HMM-based Pattern Classification

Bayes Theorem

$$P(M|X,\Theta) = \frac{p(X|M,\Theta)\ P(M|\Theta)}{p(X|\Theta)}$$

- M: Sequential (sentence) model
- Θ: Model Parameters

- $P(X, M\,|\,\Theta)$: HMM (acoustic model)
- $P(X\,|\,\Theta)$: Assumed constant
- $P(M\,|\,\Theta)$: Prior knowledge (language model). $P(M|\Theta) \Rightarrow P(M|\Theta^*)$

Three HMM Problems

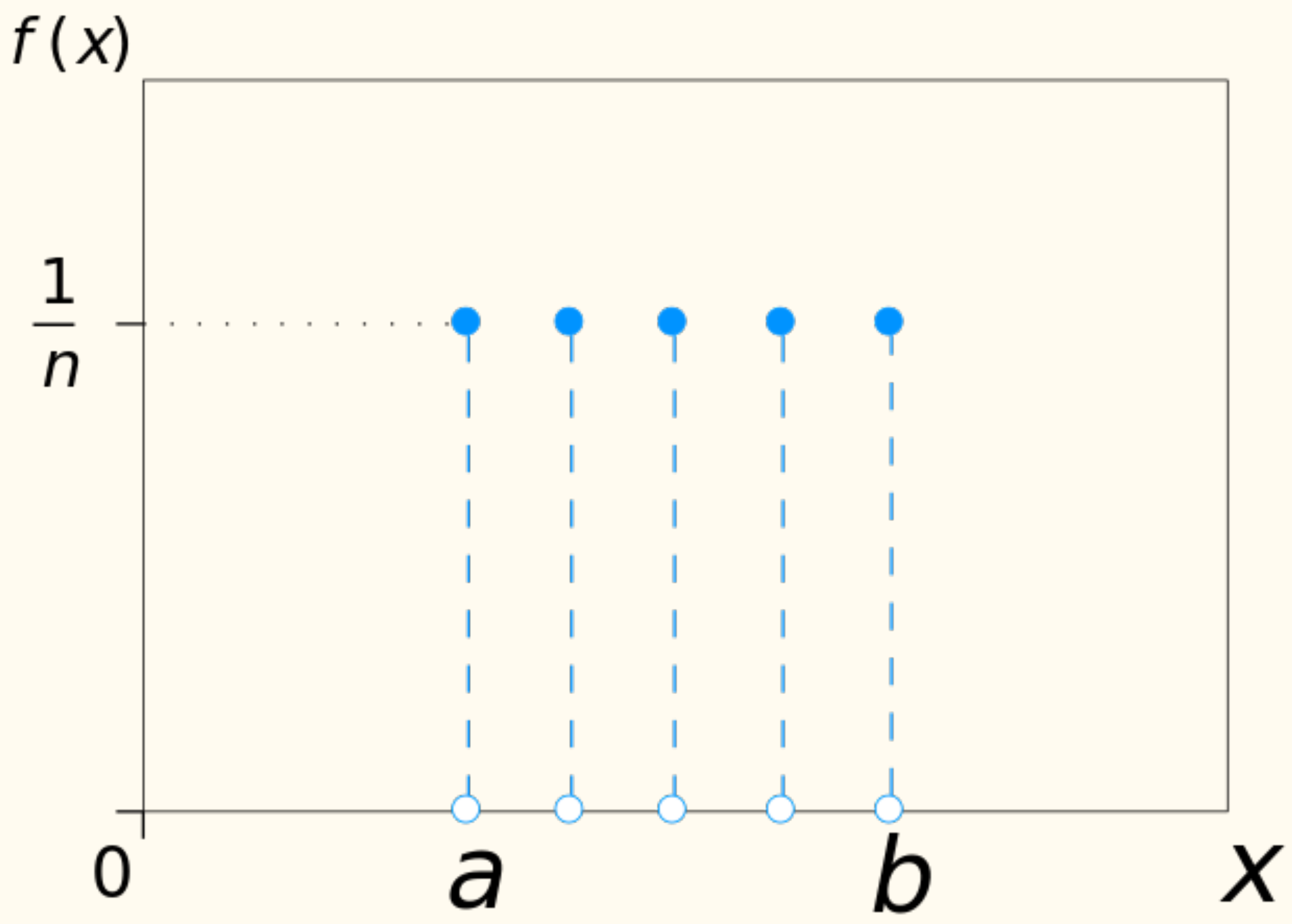
1. Definition and estimation of transition a_{ij} and emission $b_i(x)$ probabilities:
   - Computing likelihood $P(X|M,\Theta)$ for a given M_k and fixed Θ
2. Training a HMM:
   - Estimating Θ such that: $\mathrm{argmax}_\Theta \prod_{j=1}^{J} P(X_j|M_j,\Theta)$
3. Classification (decoding) of an observed sequence X:
   - $X \in M_j \quad \text{if} \quad M_j = \mathrm{argmax}_{M_j} P(X|M_k,\Theta)P(M_k)$

Training Problem

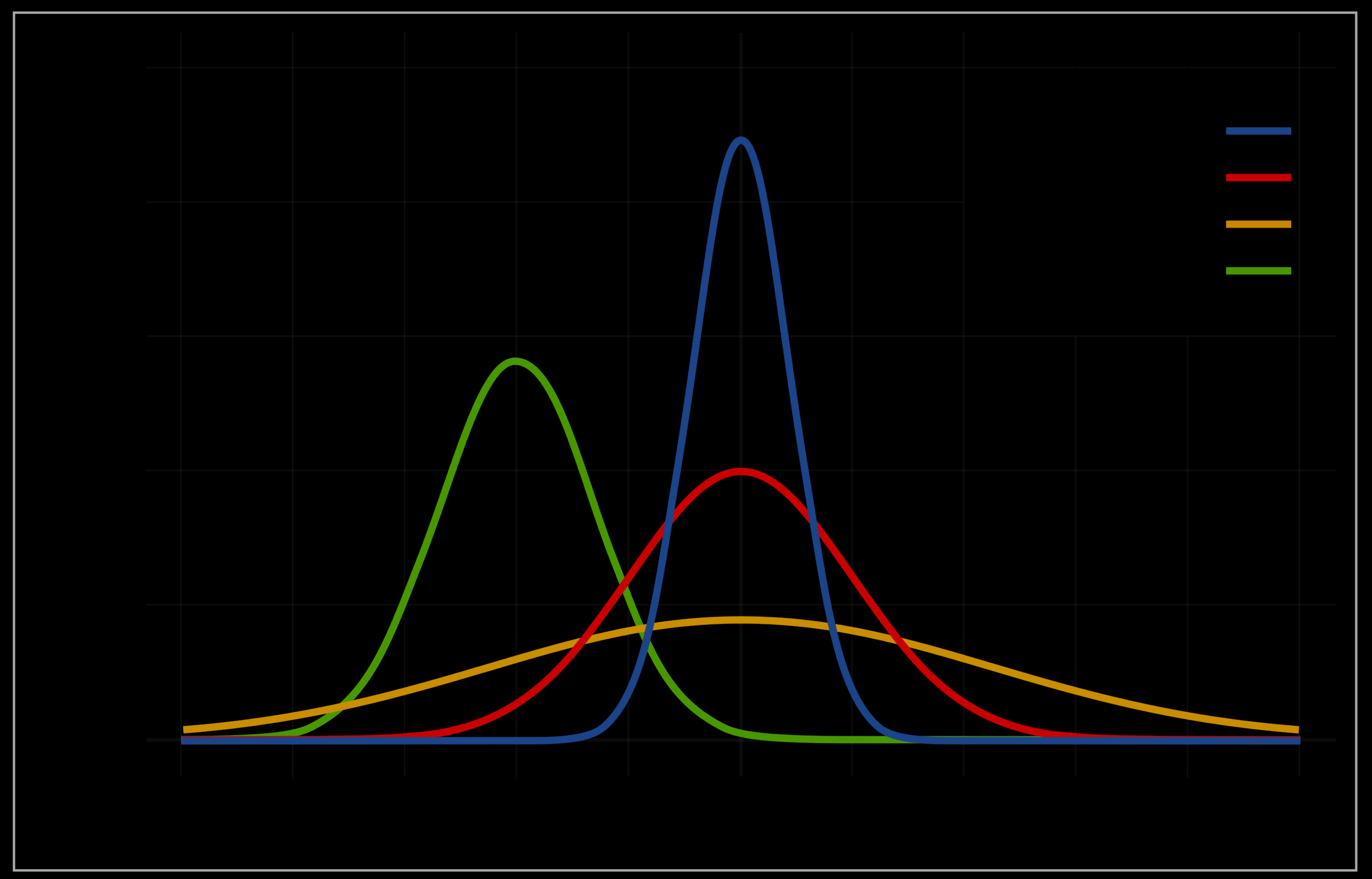
HMM Training Problem

- We want accurate parameters Θ from the observations sequence.
- States in HMM are hidden $\rightarrow$ no closed-form equation for estimating parameters.
- We need to estimate the parameters $\Theta = \{\pi, A, B\}$ with a maximum likelihood framework on $p(X|\Theta)$.
    - Transition matrix A
    - Emission's underlying PDF B (discrete or continuous)

Discrete



Continuous



HMM Training Problem

- We estimate these parameters such that $\operatorname{argmax}_{\Theta} \prod_{j=1}^{J} P(X_j | M_j, \Theta)$
- We use the Forward-Backward algorithm
  - Iterative procedure of re-estimations
  - Efficient:
    - $\mathcal{O}(TK^T) \rightarrow \mathcal{O}(TK^2)$
    - Greatly reduces computation of the likelihood of a sequence given parameters.
    - Stores intermediate values that lead to a given state at a given time.
- Can also use *embedded* Viterbi approximation.

I. Forward-Backward Training

Forward-Backward Training

- Algorithms and variables:
  - Forward algorithm and variable $\alpha_t(i)$
  - Backward algorithm variable $\beta_t(i)$
  - Sequence of events $\xi_t(i,j)$
  - Gamma variable $\gamma_t(i)$

Forward Recurrence

We define the following variable:

- $\alpha_t(i) = p(x_1, \ldots, x_t, q^t = q_i \mid \Theta)$

i.e. the probability of having observed the partial sequence $\{x_1, \ldots, x_t\}$ and being at state i at time t, given the parameters Θ.

- Requires π, A, B
- Complexity: $\mathcal{O}(TK^2)$

1. Initialization:
   - $$\alpha_1(i) = \pi_i \, b_i(x_1), \quad 1 \le i \le K$$

   Join probability of state q_i and initial observation x_i.

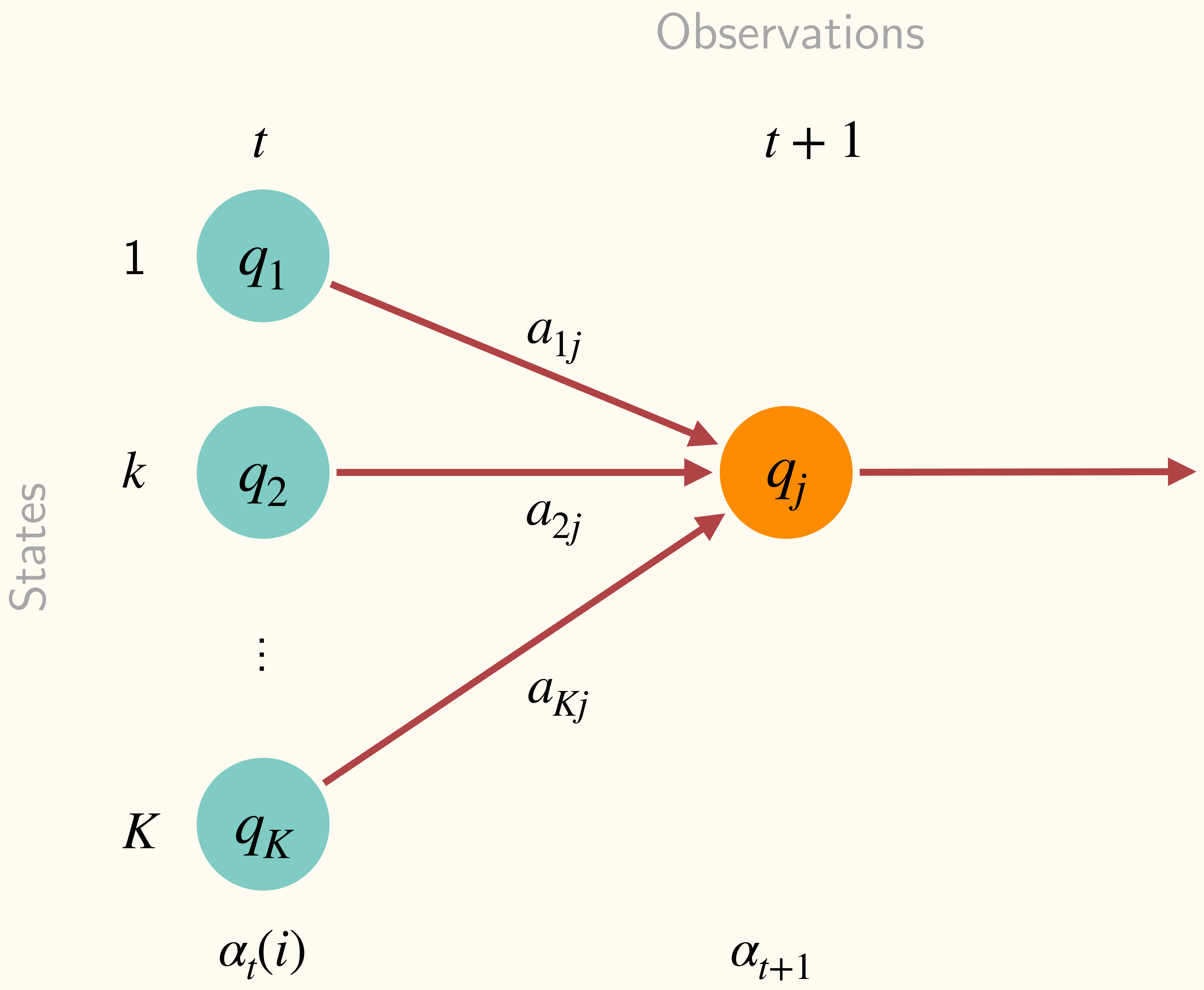
2. Recursion:
   - $$\alpha_{t+1}(j) = [\sum_{i=1}^{K} \alpha_t(i)\, a_{ij}]\, b_j(x_{t+1})$$

   All possible ways to reach j · probability to generate x_{t+1}

3. Termination:
   - $$P(X \mid \Theta) = \sum_{i=1}^{K} \alpha_T(i)$$

   Sum over all possible states one could've ended up in.

Forward Algorithm - Recursion



Variable:

$$\underbrace{\alpha_t(i)} = p(x_1, \ldots, x_t, q^t = q_i \,|\, \Theta)$$

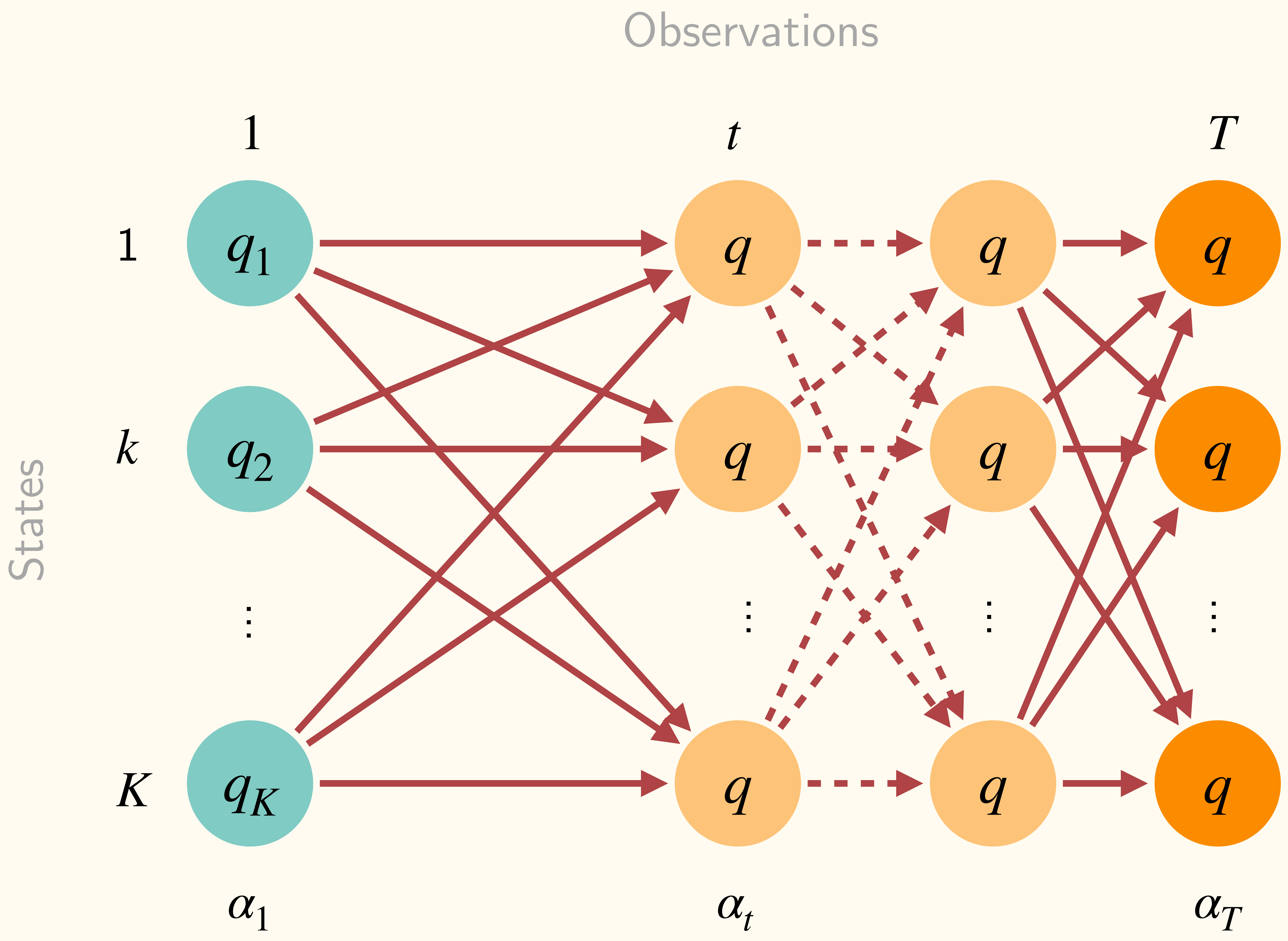
Probability of joint event that X is observed and the state at time t is q_i.

Recursion step:

$$\alpha_{t+1}(j) = [\sum_{i=1}^{K} \underbrace{\alpha_t(i)\, a_{ij}}]\, \overbrace{b_j(x_{t+1})}^{\text{Generate observation}}$$

Probability of joint event that X is observed and q_j is reached at time $t+1$ via q_i at time t.

Forward Algorithm - Termination



Variable:

$$\underbrace{\alpha_t(i)} = p(x_1, \ldots, x_t, q^t = q_i \,|\, \Theta)$$

Probability of joint event that X is observed and the state at time t is q_i.

Termination step:

$$P(X \,|\, \Theta) = \sum_{i=1}^{K} \alpha_T(i)$$

Backward Algorithm

We define the following variable:

- $\beta_t(i) = p(x_{t+1}, \ldots, x_T | q^t = q_i, \Theta)$

i.e. the probability of having observed the partial sequence $\{x_{t+1}, \ldots, x_T\}$, given the state i at time t and the parameters Θ.

- Requires π, A, B
- Complexity: $\mathcal{O}(TK^2)$

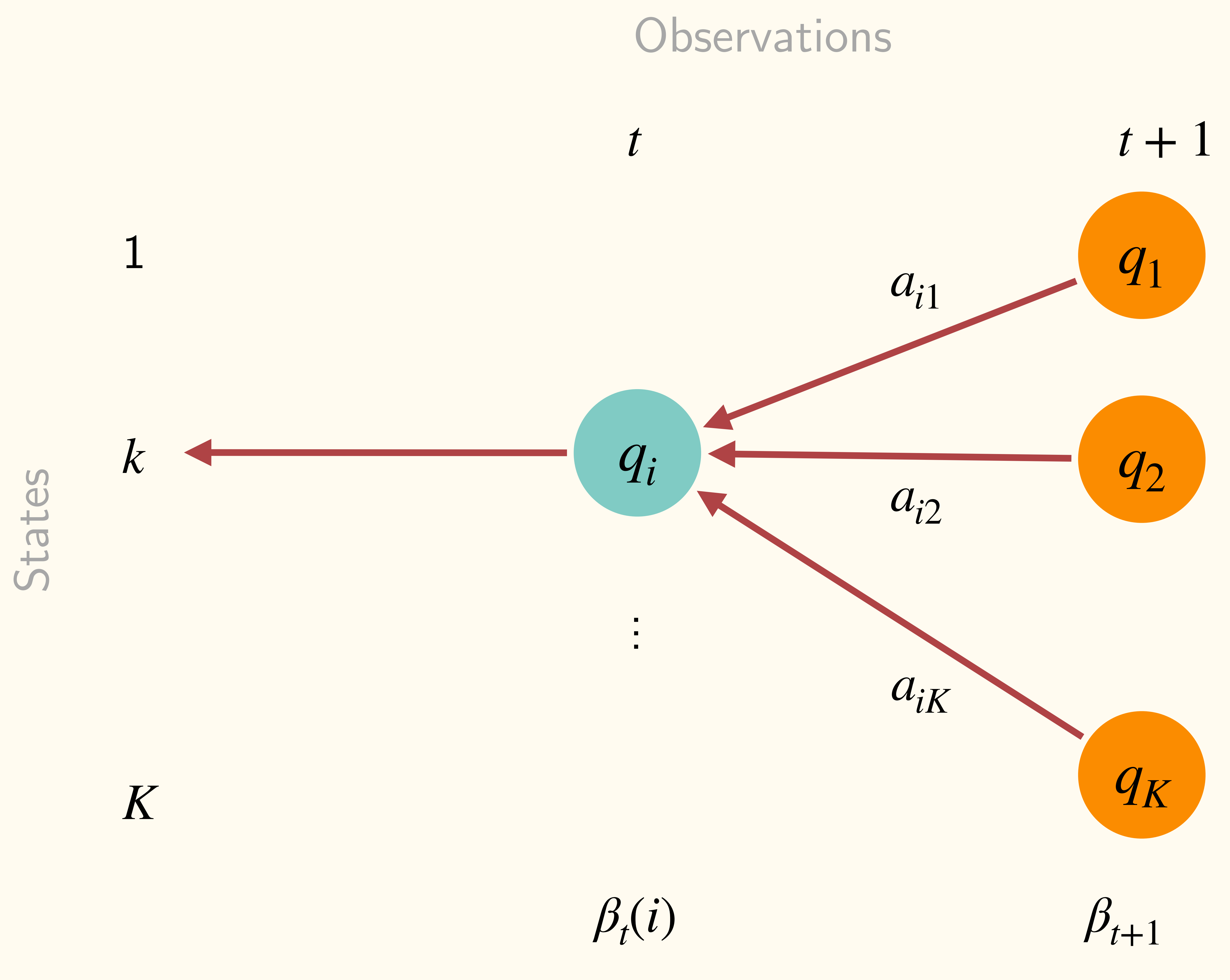
1. Initialization:
   - $\beta_T(i) = 1$

Arbitrarily defined to be 1 for all i.

2. Recursion:
   - $$\beta_t(j) = [\sum_{i=1}^{K} \beta_{t+1}(i)\, a_{ij}]\ b_j\ (x_{t+1})$$

3. Termination:
   - $$\beta_0 = P(X | \Theta) = \sum_{i=1}^{K} \pi_i\, b_i(x_1)\, \beta_1(i)$$

Backward Algorithm - Recursion



q_j can be reached at time $t+1$ from the K possible states.

Variable:

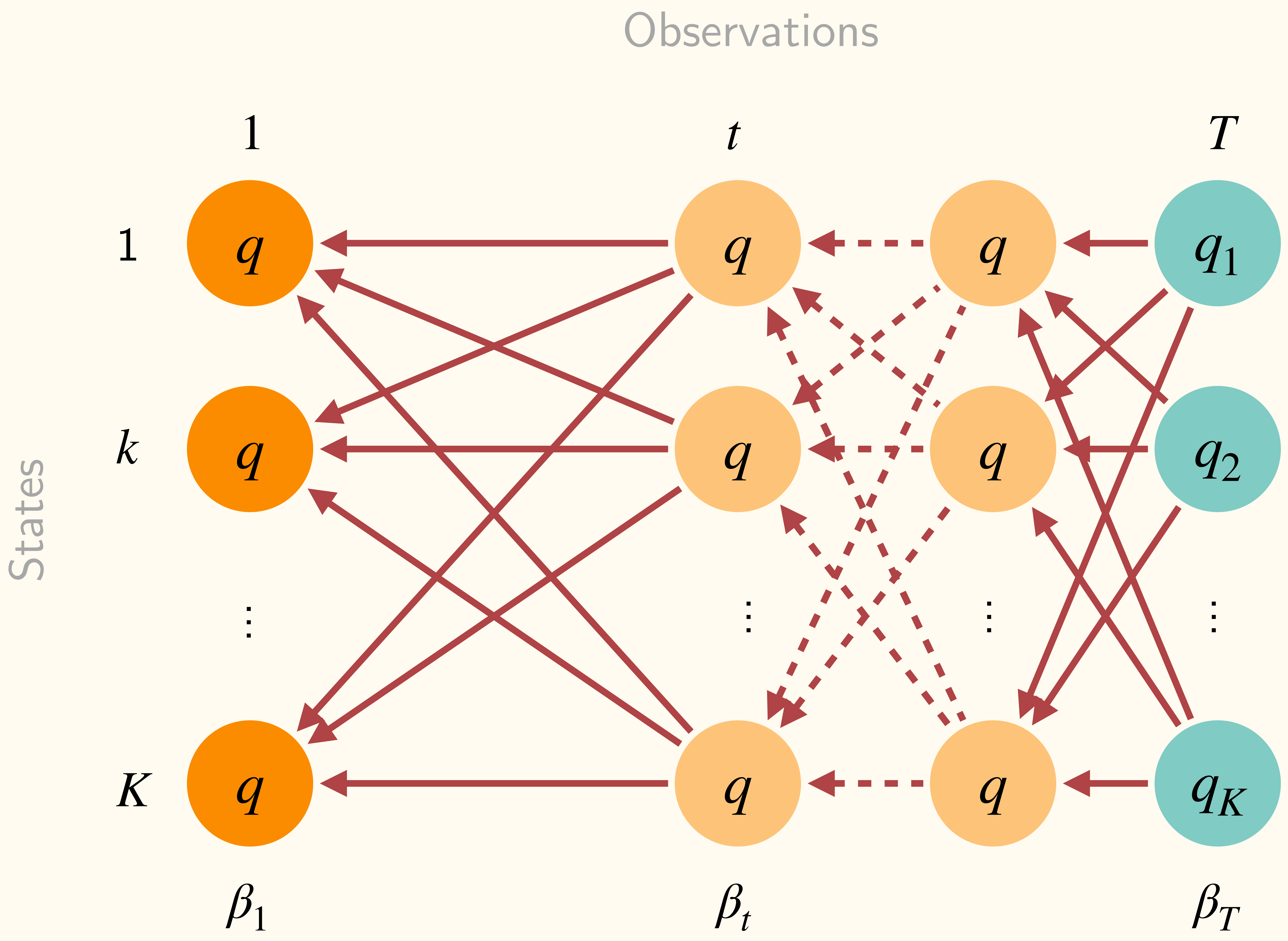
$$\underbrace{\beta_t(i)} = p(x_{t+1}, \ldots, x_T \,|\, q^t = q_i, \Theta)$$

Probability that X is observed given the state q_i at time t and model parameters Θ.

Recursion step:

$$\beta_t(j) = [\sum_{i=1}^{K} \beta_{t+1}(i)\, a_{ij}]\; b_j\,(x_{t+1})$$

Backward Algorithm - Termination



Variable:

$$\underbrace{\beta_t(i)} = p(x_{t+1}, \ldots, x_T \,|\, q^t = q_i, \Theta)$$

Probability that X is observed given the state q_i at time t and model parameters Θ.

Termination step:

$$\beta_0 = P(X \,|\, \Theta) = \sum_{i=1}^{K} \pi_i \, b_i(x_1) \, \beta_1(i)$$

Transition Probabilities Re-Estimation

- Forward and backward algorithms used to isolate states within HMM
- These variables let us estimate:
  - Transition probabilities between states
  - Emission probability distribution

- Start with re-estimation of A:
  - $$\overline{a_{ij}} = \frac{\text{Expected number of times from state } q_i \text{ to } q_j}{\text{Expected number of transitions from } q_i}$$
  - Need ξ

Sequence of Events

Forward Backward

We define the following variable:

- $\xi_t(i,j) = P(q^t = q_i, q^{t+1} = q_j \,|\, X, \Theta)$

i.e. the probability of being in state i at time t and in state j at time $t+1$, given the observations and parameters Θ.

Can be expressed in terms of both forward and backward variables as:

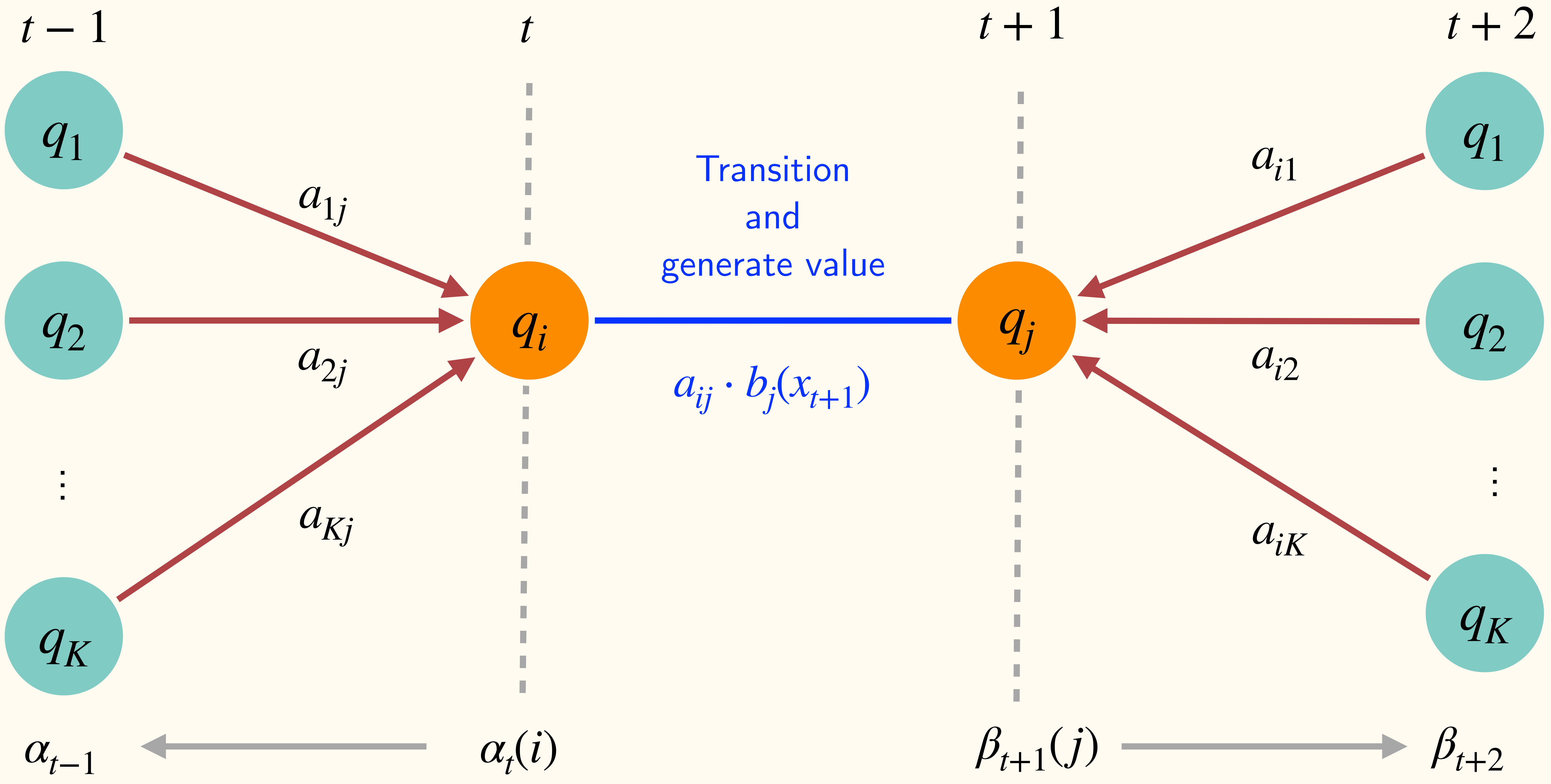
$$\xi_t(i,j) = \frac{P(q_i^t, q_j^{t+1} \,|\, X, \Theta)}{\underbrace{P(X \,|\, \Theta)}_{\text{Normalization factor}}}$$

$$= \frac{\alpha_t(i)\, a_{ij}\, b_j(x_{t+1})\, \beta_{t+1}(j)}{\sum_{i=1}^{K} \alpha_t(i)\beta_t(i)}$$

$$= \frac{\alpha_t(i)\, a_{ij}\, b_j(x_{t+1})\, \beta_{t+1}(j)}{\sum_{i=1}^{K} \sum_{j=1}^{K} \alpha_t(i) a_{ij}\, b_j(x_{t+1})\beta_{t+1}(j)}$$

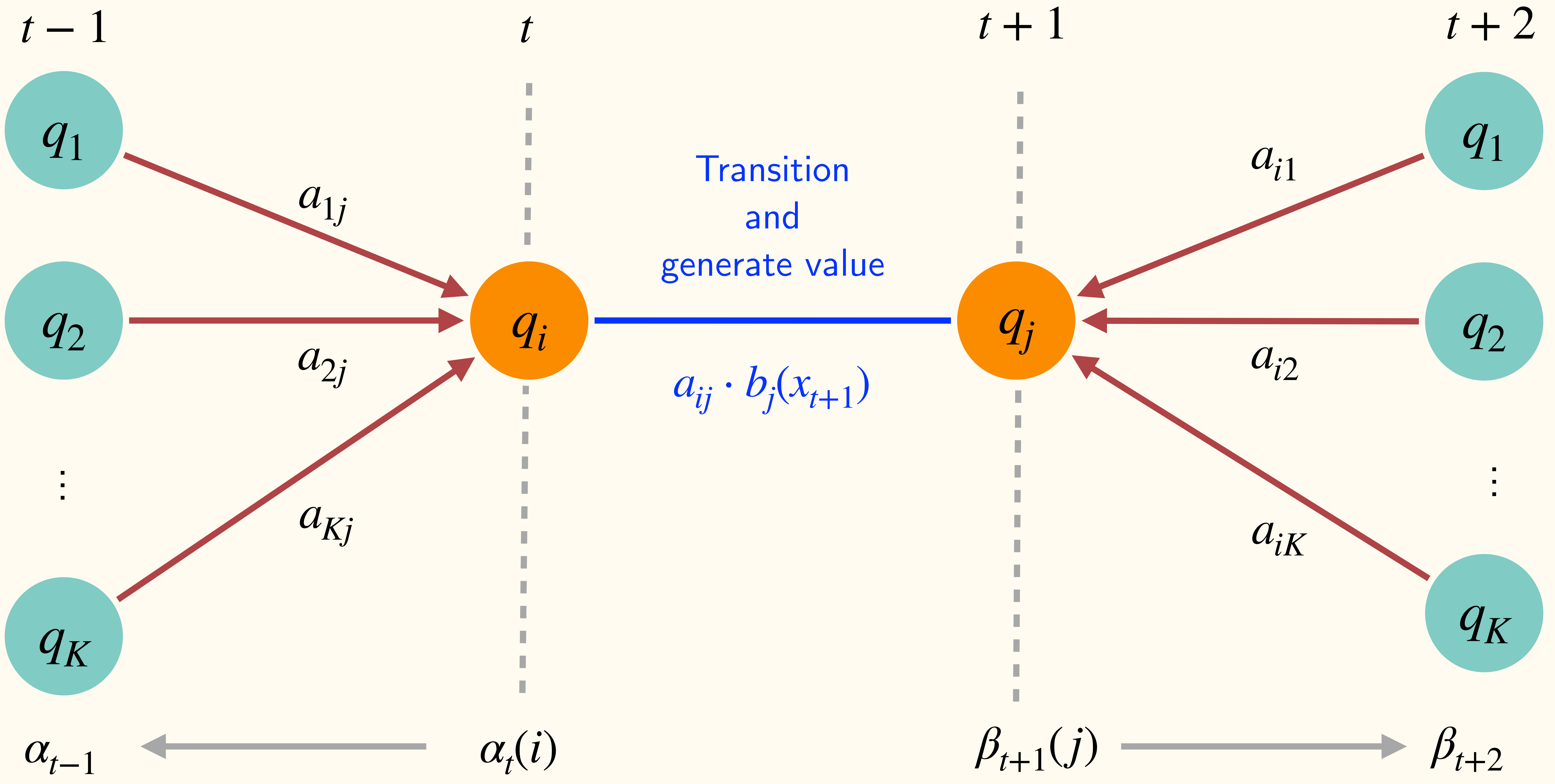
Sequence of Events

$$\xi_t(i,j) = P(q^t = q_i, q^{t+1} = q_j \,|\, X, \Theta)$$



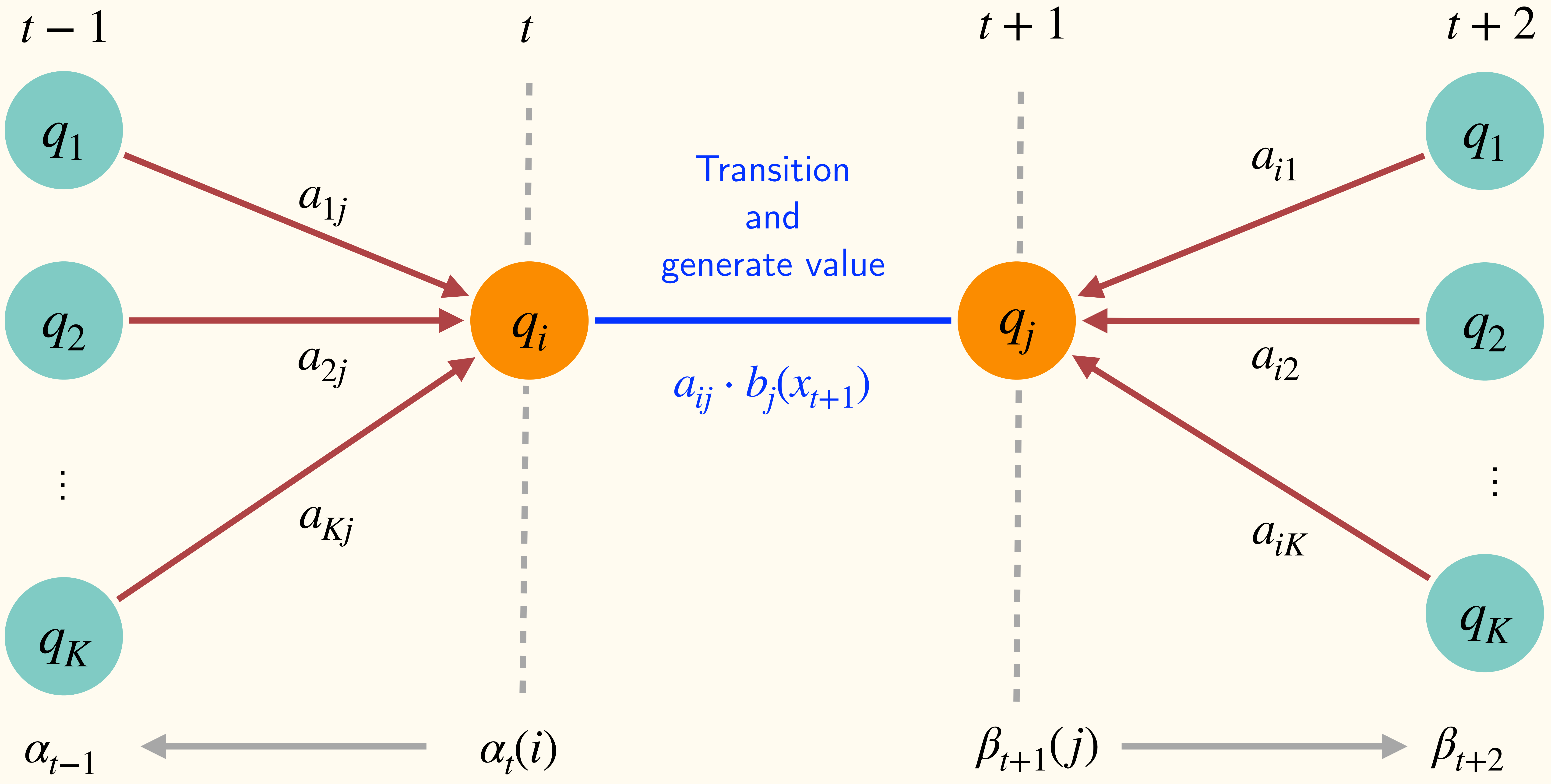
Sequence of Events

$$\xi_t(i,j) = \frac{\alpha_t(i)\, a_{ij}\, b_j(x_{t+1})\, \beta_{t+1}(j)}{P(X\,|\,\Theta)}$$



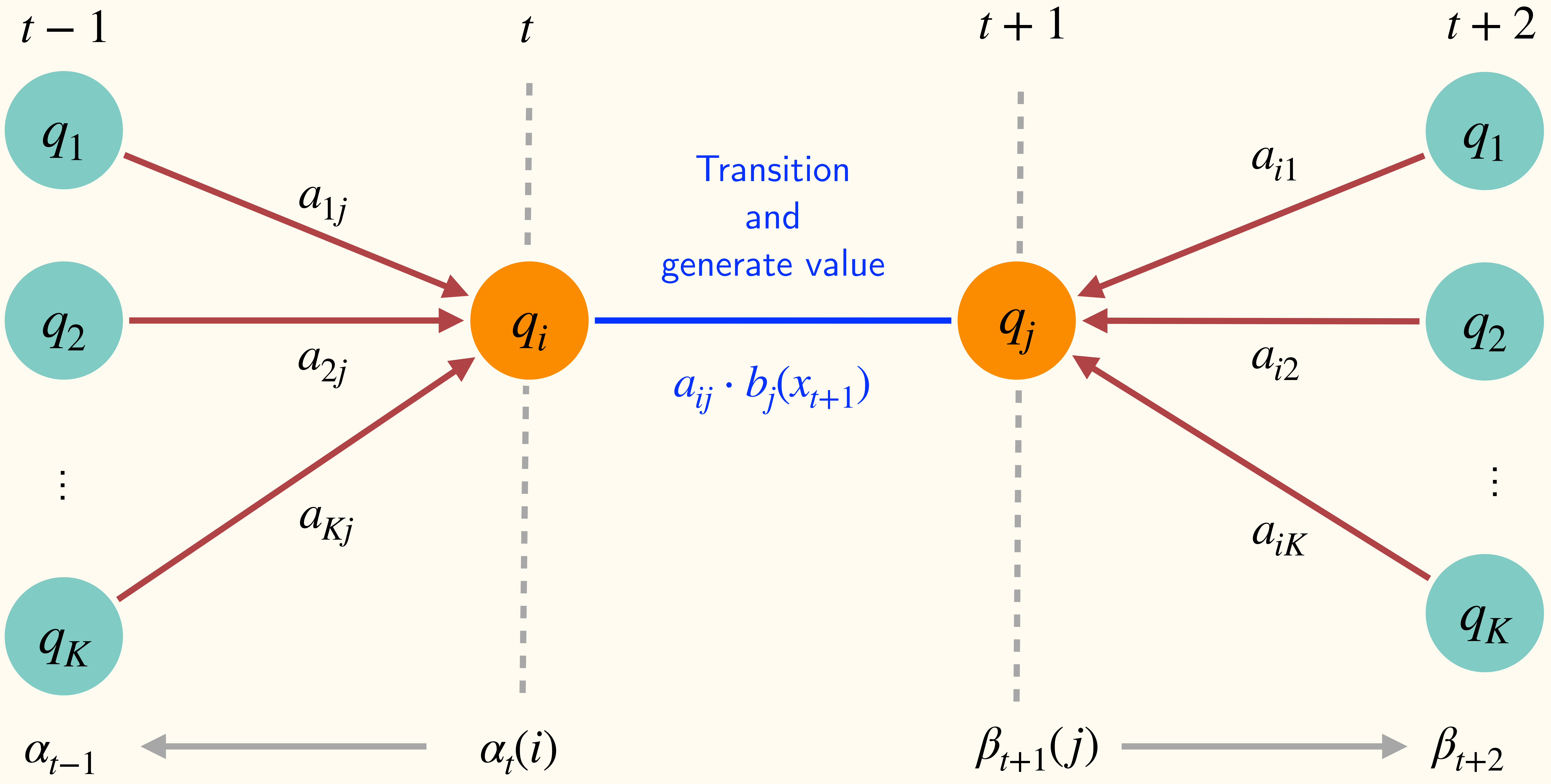
Sequence of Events

$$\xi_t(i,j) = \frac{\alpha_t(i)\, a_{ij}\, b_j(x_{t+1})\, \beta_{t+1}(j)}{\sum_{j=1}^{K} \alpha_t(i) \beta_t(j)}$$



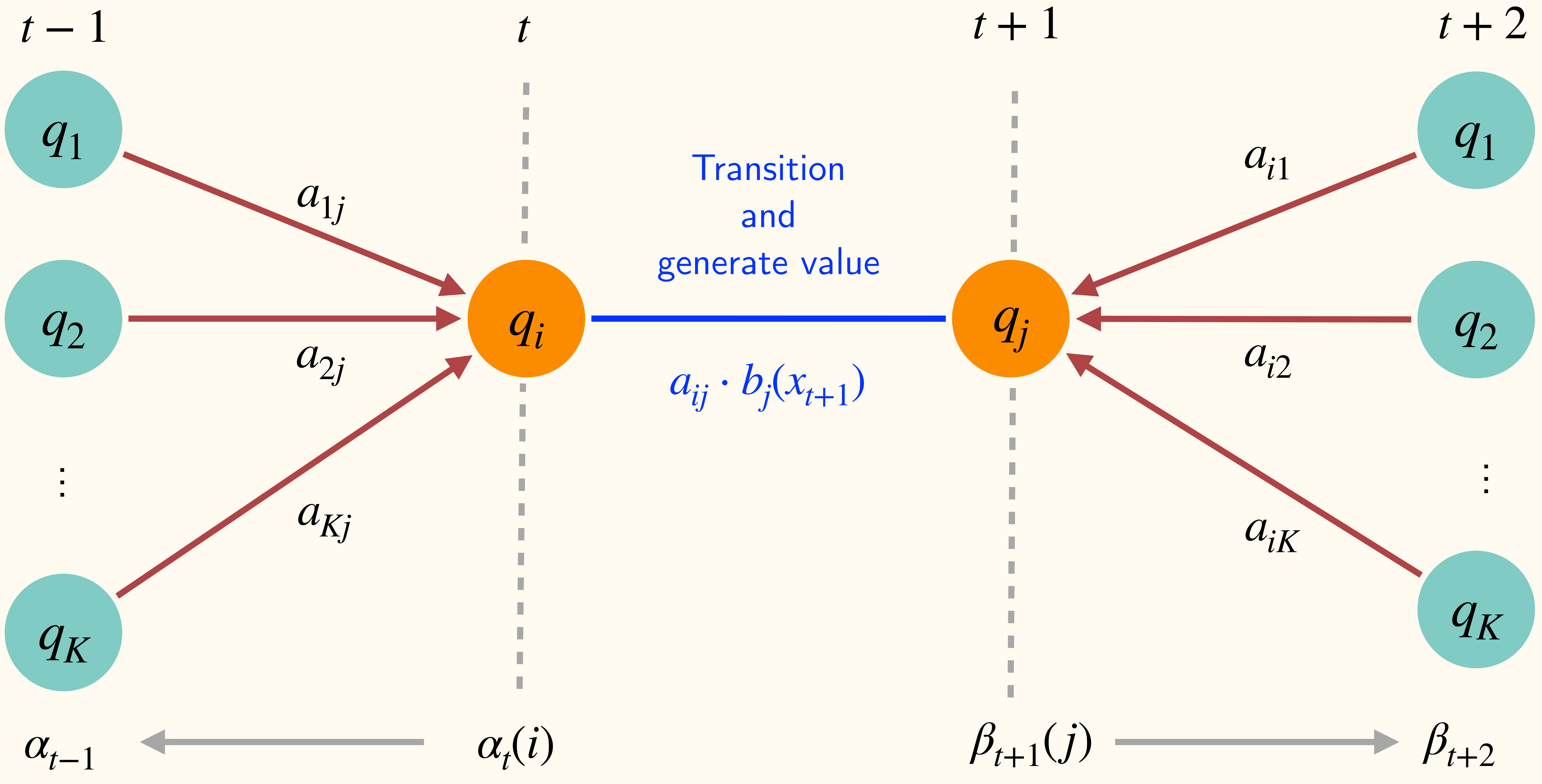
Sequence of Events

$$\xi_t(i,j) = \frac{\alpha_t(i)\, a_{ij}\, b_j(x_{t+1})\, \beta_{t+1}(j)}{\sum_{i=1}^{K} \sum_{j=1}^{K} \alpha_t(i) a_{ij}\; b_j(x_{t+1}) \beta_{t+1}(j)}$$



Sequence of Events

$$\sum_{t=1}^{T-1} \xi_t(i,j) \ ?$$



Transition Matrix Re-Estimation

- $$\overline{a_{ij}} = \frac{\text{Expected number of times from state } q_i \text{ to } q_j}{\text{Expected number of transitions from } q_i} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \sum_{k=1}^{K} \xi_t(i,k)}$$

- Compute for all pairs (i,j): $\bar{A} = \begin{bmatrix} \bar{a}_{11} & \bar{a}_{12} & \bar{a}_{13} \\ \bar{a}_{21} & \bar{a}_{22} & \bar{a}_{23} \\ \bar{a}_{31} & \bar{a}_{32} & \bar{a}_{33} \end{bmatrix}$

Emission Probability Distribution Re-Estimation (Discrete)

- At each state q, we have an observation x which is a discrete value in the 'observation vocabulary' V.

  - $$\overline{b_j(v_k)} = \frac{\text{Expected number of times in state } q_j \text{ and observing } v_k}{\text{Expected number of times in state } q_j}$$

  - Need γ

Gamma Variable

Forward Backward

We define the following variable:

- $\gamma_t(i) = P(q^t = q_i | X, \Theta)$

i.e. the probability of being in state i at time t, given the observations and parameters Θ.

Can be expressed in terms of both forward and backward variables as:

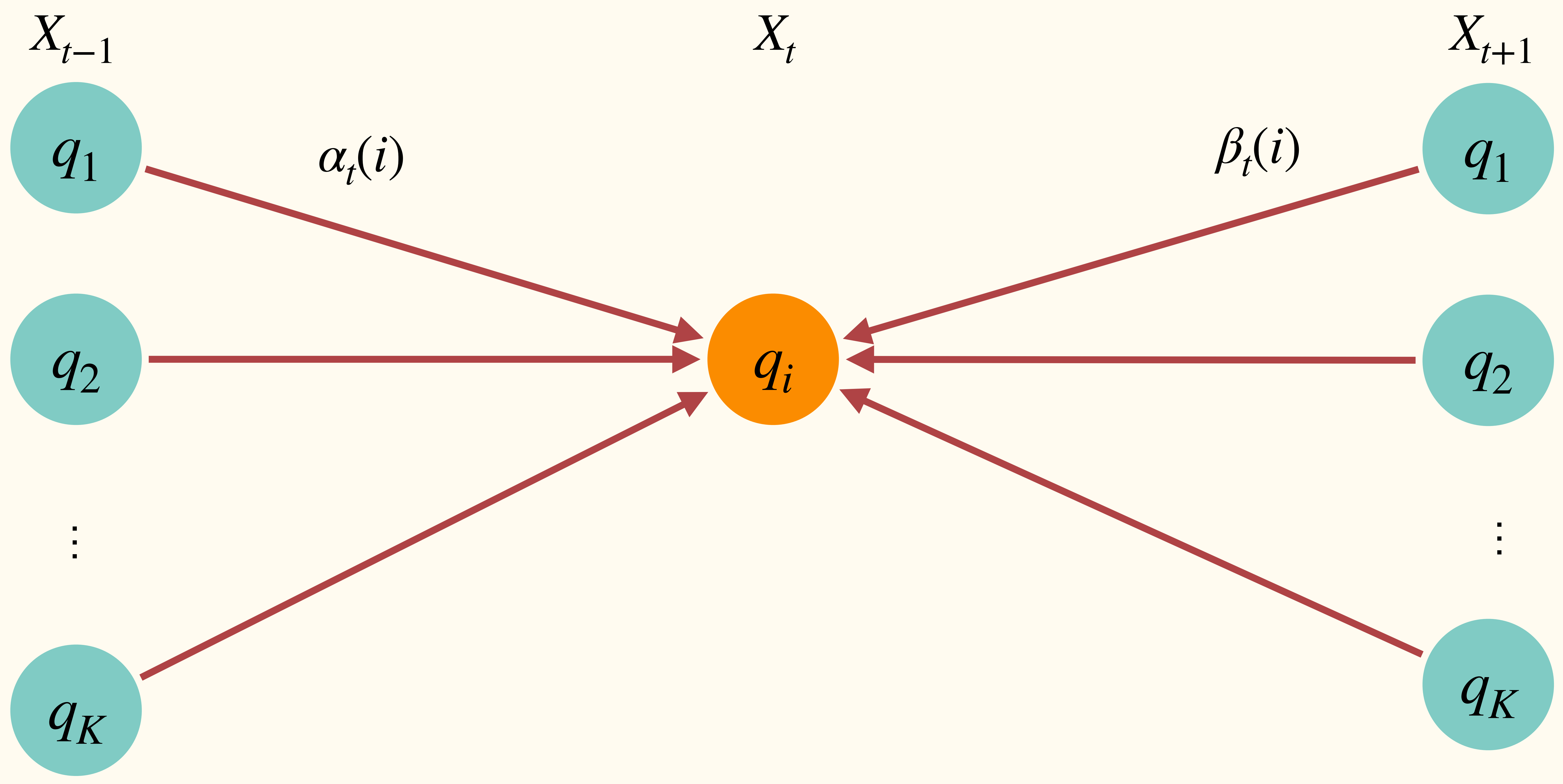
$$\gamma_t(i) = \frac{P(q_i^t, X | \Theta)}{P(X | \Theta)} = \frac{\alpha_t(i)\,\beta_t(i)}{P(X | \Theta)}$$

We can relate $\gamma_t(i)$ to $\xi(i,j)$:

$$\gamma_t(i) = \sum_{j=1}^{K} \xi(i,j)$$

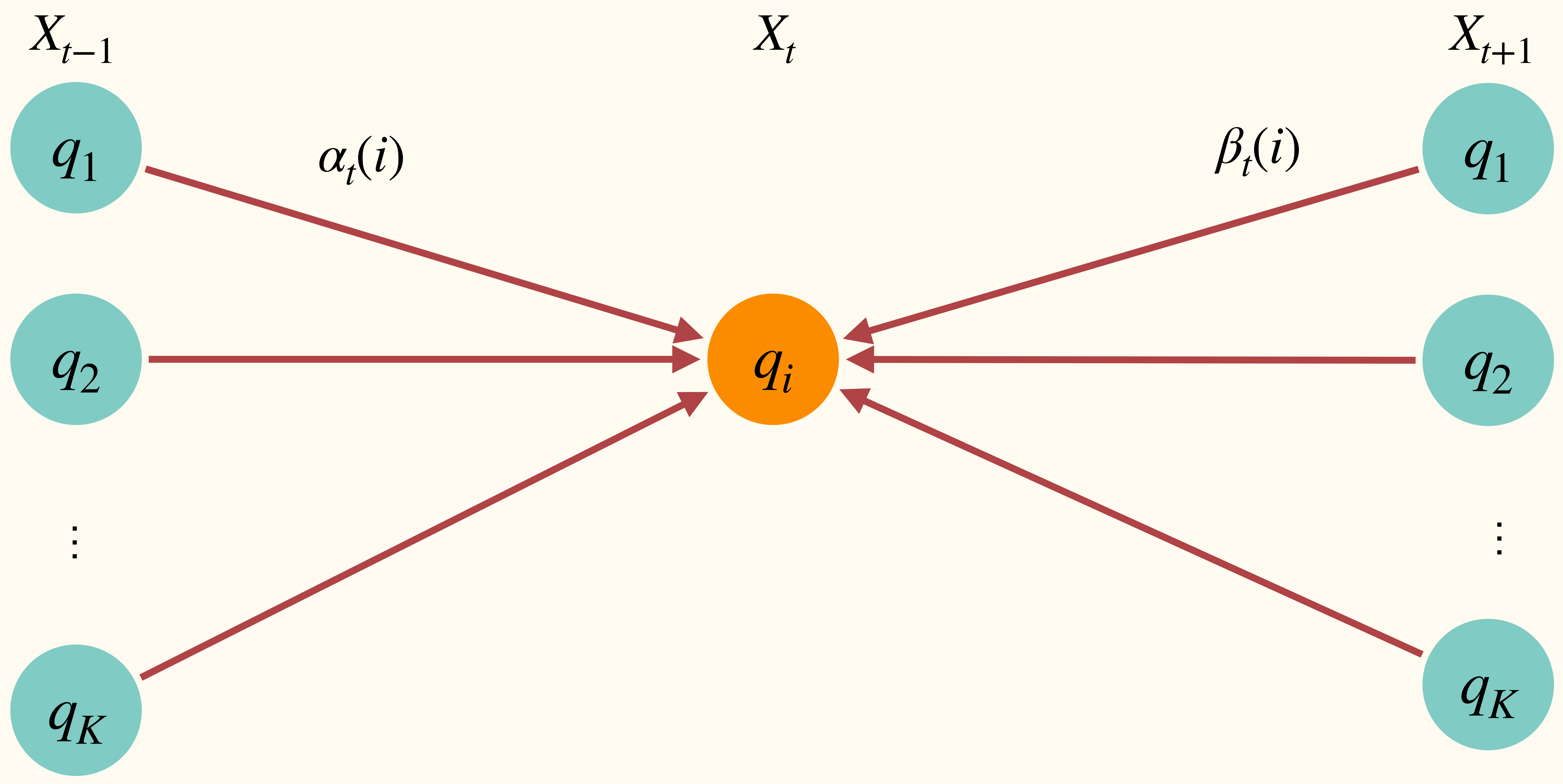
Gamma Variable

$$\gamma_t(i) = P(q^t = q_i \,|\, X, \Theta)$$



Gamma Variable

$$\gamma_t(i) = \frac{\alpha_t(i)\,\beta_t(i)}{P(X\,|\,\Theta)}$$



Emission Probability Distribution Re-Estimation (Discrete)

$$\overline{b_j(v_k)} = \frac{\text{Expected number of times in state } q_j \text{ and observing } v_k}{\text{Expected number of times in state } q_j} = \frac{\sum_{t=1 \& x_t = v_k}^{T} \gamma_t(i)}{\sum_{t=1}^{T} \gamma_t(i)}$$

Gamma Variable

We can express $\gamma_t(i)$ in 2 ways.

The expected number of times q_i is visited:

$$\sum_{t=1}^{T-1} \gamma_t(i)$$

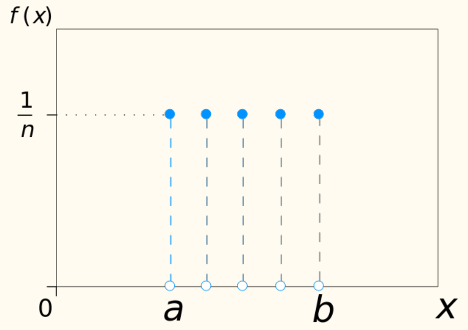
Useful for re-estimating transition probabilities.

The expected number of times transitions are made from q_i:

$$\sum_{t=1}^{T} \gamma_t(i)$$

Useful for re-estimating emission probability distribution.

Parameters Re-Estimation (Discrete)



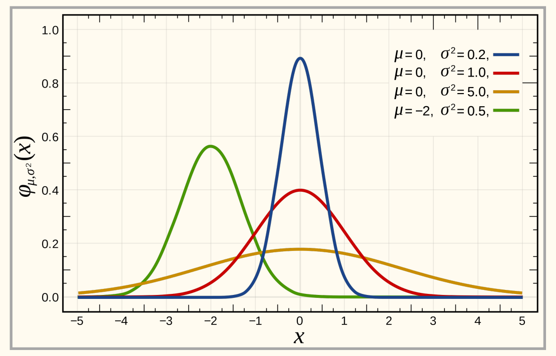
We define the following formulas, as estimators for the:

- Initial state: $\overline{\pi_i} = \gamma_1(i)$ ← Expected frequency in state q_i at time $t = 1$

- Transition probabilities: $\overline{a_{ij}} = \dfrac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$
  - Numerator ← Expected number of transitions from state q_i to q_j
  - Denominator ← Expected number of transitions from state q_i

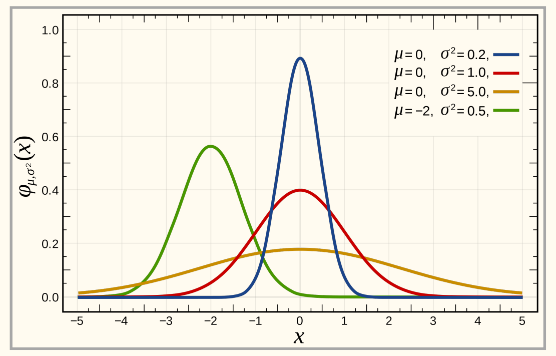
- Emission PDF: $\overline{b_j(v_k)} = \dfrac{\sum_{t=1 \& x_t = v_k}^{T} \gamma_t(i)}{\sum_{t=1}^{T} \gamma_t(i)}$
  - Numerator ← Expected number of times in state q_j and observing v_k
  - Denominator ← Expected number of times in state q_j

Parameters Re-Estimation (Continuous)



- The re-estimate for the transition probabilities $\bar{a}_{ij}$ are the same.
- We are not interested in the emission probabilities, but the *parameters* that describe its distribution, e.g.: $\mathcal{N}(\mu, \sigma^2)$ or $\mathcal{N}(\mu, \Sigma)$
  - Emission PDF: $\overline{b_j} = \{\bar{\mu}, \bar{\sigma}^2\}$ or $\overline{b_j} = \{\bar{\mu}, \overline{\Sigma}\}$ for a Gaussian distribution.
  - $$\overline{\mu_{jk}} = \frac{\Sigma_{t=1}^{T} \gamma_t(j,k) \cdot X_t}{\Sigma_{t=1}^{T} \gamma_t(j,k)}$$
  - $$\bar{\sigma}_{jk}^2 = \frac{\Sigma_{t=1}^{T} \gamma_t(j,k) \cdot (X_t - \bar{\mu}_{jk})^2}{\Sigma_{t=1}^{T} \gamma_t(j,k)}$$ or $$\overline{\Sigma}_{jk} = \frac{\Sigma_{t=1}^{T} \gamma_t(j,k) \cdot (X_t - \bar{\mu}_{jk}) \cdot (X_t - \bar{\mu}_{jk})^T}{\Sigma_{t=1}^{T} \gamma_t(j,k)}$$

Parameters Re-Estimation (Continuous)



Probability of being in state j at time t with k-th mixture component accounting for X_t :

- $\overline{b_j} = \{\bar{c}, \bar{\mu}, \overline{\Sigma}\}$

- $\bar{c}_{jk} = \dfrac{\Sigma_{t=1}^{T} \gamma_t(j,k)}{\Sigma_{t=1}^{T} \Sigma_{k=1}^{M} \gamma_t(j,k)}$

- $\gamma_t(j,k) = \dfrac{\alpha_t(j)\beta_t(j)}{\sum_{j=1}^{K} \alpha_T(j)\beta_t(j)} \cdot \dfrac{c_{jk}\mathcal{N}(X_t, \mu_{jk}, \Sigma_{jk})}{\sum_{m=1}^{M} c_{jm}\mathcal{N}(X_t, \mu_{jm}, \Sigma_{jm})}$

Baum-Welch EM Algorithm

- $\overline{a_{ij}}$ and $\overline{b_j(v_k)}$
  - Re-compute α_t, β_t, γ_t, ξ_t
    - New values $\overline{a_{ij}}$ and $\overline{b_j(v_k)}$
      - …

E-Step:

$\gamma_t(i), \xi_t(i,j)$

M-Step:

$\overline{a_{ij}}, \overline{b_j(v_k)}$

Iterative Training

1. Estimate $p(X \mid \Theta)$ and $p(X \mid \bar{\Theta})$.
2. If $p(X \mid \bar{\Theta}) \geq p(X \mid \Theta)$:
   - Replace Θ with new estimate of parameters $\bar{\Theta}$.
   - Repeat the **E** and **M** steps of EM algorithm.
   - Go to step 1.
3. Else:
   - Terminate with $\bar{\Theta}$ as trained parameters (**convergence**).

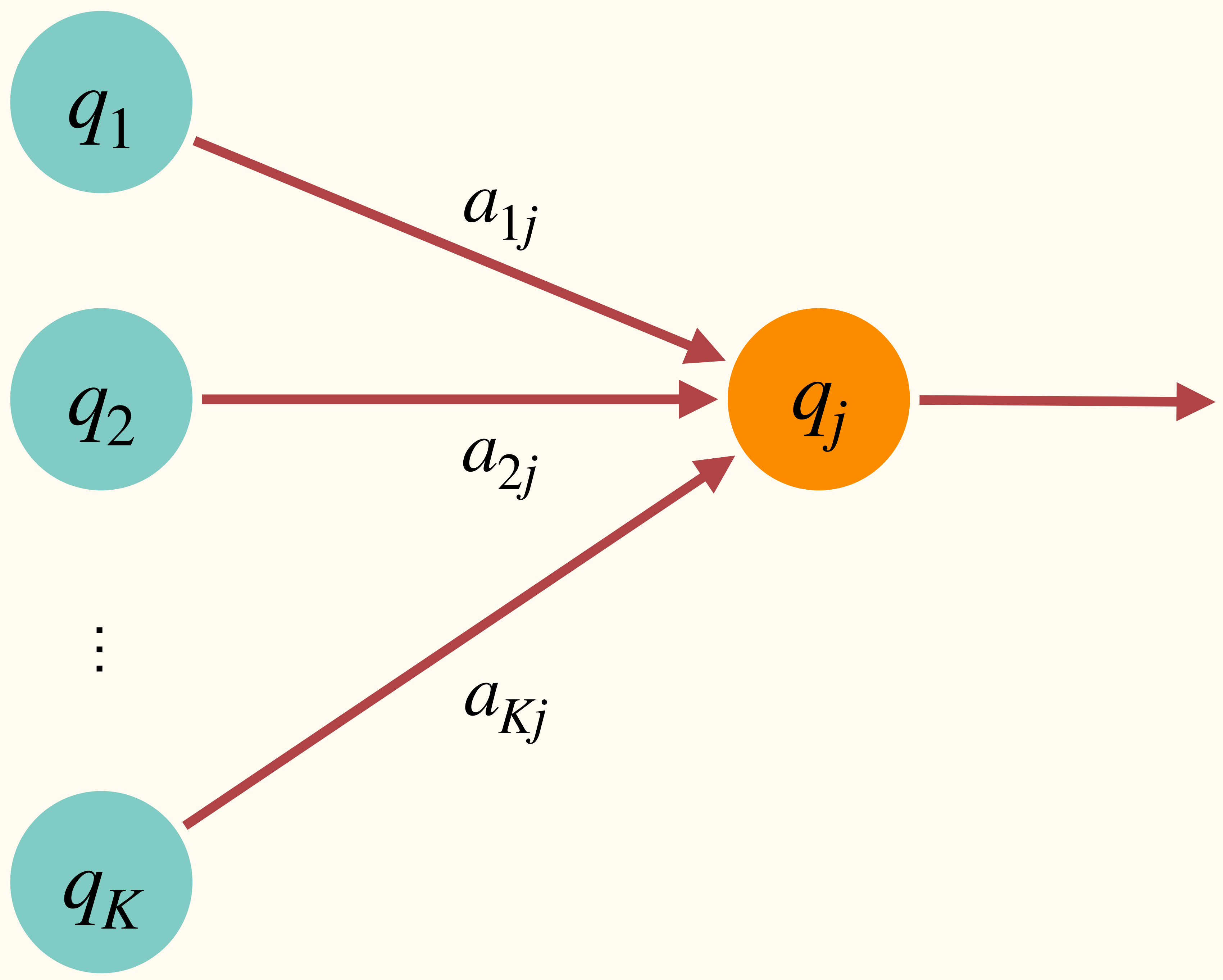
E-Step:
$\gamma_t(i), \xi_t(i,j)$

M-Step:
$\overline{a_{ij}}, \overline{b_j(v_k)}$

II. Embedded Viterbi Training

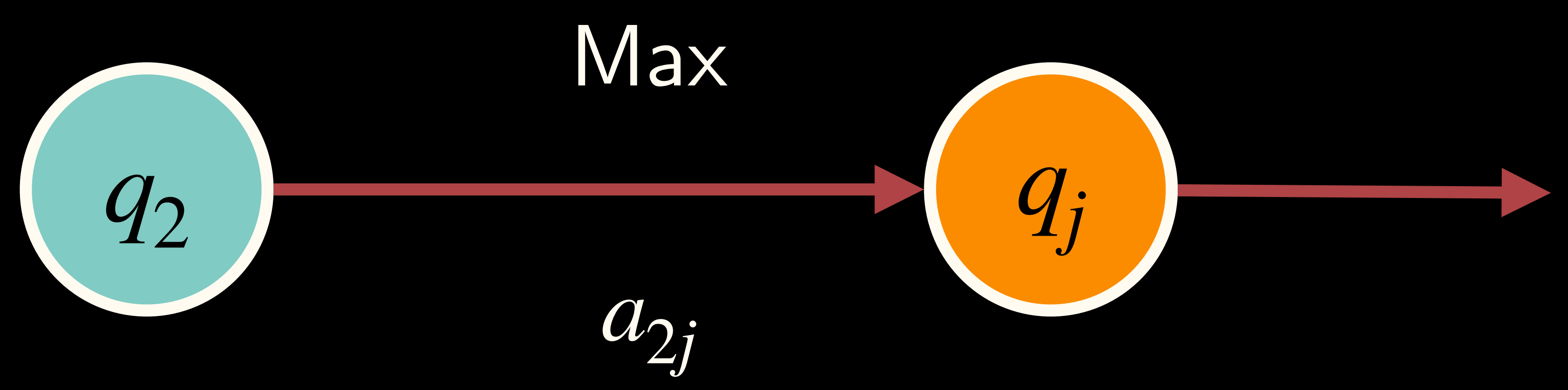
Forward-Backward

$$\alpha_t(i) = p(x_1, \ldots, x_t, q^t = q_i \mid \Theta)$$



Viterbi

$$\delta_t(i) = \max p(q^1, \ldots, q_i^t, x^1, \ldots, x^t \mid \Theta)$$



Viterbi Algorithm

We define 2 variables:

1. $\delta_t(i)$: highest likelihood along a side path among all paths ending in state q_i at time t:
   - $\delta_t(i) = \max p(q^1, \ldots, q_i^t, x^1, \ldots, x^t \,|\, \Theta)$

2. $\psi_t(i)$: variable to keep track of 'best path' ending in state q_i at time t:
   - $\psi_t(i) = \text{argmax } \; p(q^1, \ldots, q_i^t, x^1, \ldots, x^t \,|\, \Theta)$

Viterbi Algorithm

1. Initialization:
   - $\delta_1(i) = \pi_i \, b_i(x_1)$
   - $\psi_1(i) = 0$
2. Recursion:
   - $\delta_t(j) = \max\limits_{1 \leq i \leq K} [\delta_{t-1}(i) \, a_{ij}] \, b_j(x_t)$
   - $\psi_t(j) = \text{argmax}_{1 \leq i \leq K}[\delta_{t-1}(i) \, a_{ij}]$
3. Termination:
   - $P^*(X \,|\, \Theta) = \max\limits_{1 \leq i \leq K} \delta_T(i)$
   - $q^{T*} = \text{argmax}_{1 \leq i \leq K}[\delta_T(i)]$
4. Backtracking:
   - $q^{t*} = \psi_{t+1}(q^{t+1*})$

Embedded Viterbi Approximation

E-Step:

$\gamma_t(i), \xi_t(i,j)$

M-Step:

$\overline{a_{ij}}, \overline{b_j(v_k)}$

1. Estimate $p(X \mid \Theta)$ and $p(X \mid \bar{\Theta})$.
2. If $p(X \mid \bar{\Theta}) \geq p(X \mid \Theta)$:
   - Replace Θ with new estimate of parameters $\bar{\Theta}$.
   - Repeat the **E** and **M** steps of EM algorithm.
     - Obtain optimal state sequence.
     - γ_t and ξ_t are either 0 or 1.
3. Else:
   - Terminate with $\bar{\Theta}$ as trained parameters (**convergence**).

- Faster than BW, as computational cost is less.

Solved !

Summary

Pros:

- Flexible topology.
- Rich mathematical framework.
- Wide range of applications.
- Powerful learning and decoding methods.
- Good abstraction for sequences, temporal aspects.

Cons:

- A priori selection of model topology and statistical distributions.
- First order Markov model for state transition.
- Lack of contextual information as correlation between successive acoustic vectors is ignored.
- Assumption of independence for computational efficiency.

Thank you !

Room 207-2, Idiap Research Institute

www.idiap.ch/~esarkar/

+41 78 82 50 754

eklavya.sarkar@idiap.ch

Parameters Re-Estimation

- Transition probabilities: $\overline{a_{ij}} = \dfrac{C(i \rightarrow j)}{\sum_k C(i \rightarrow k)}$
- Emission PDF:
  - $\overline{\mu_j} = \dfrac{\Sigma_{x \in Z_j} x}{|Z_i|}$
  - $\overline{\Sigma_j} = \dfrac{\Sigma_{x \in Z_j} (X_t - \bar{\mu}_j) \cdot (X_t - \bar{\mu_{jk}})^T}{|Z_j|}$
  - Z_j: Set of observed features assigned to q_j

Old Slides

Likelihood Problem

Likelihood Estimation Problem

$$P(M|X,\Theta) = \frac{p(X|M,\Theta)\ P(M|\Theta)}{p(X|\Theta)}$$

- Computing $P(X|M,\Theta)$
- Fixed Θ
- Likelihood of a sequence of observations w.r.t. a HMM:

- Complexity: $\mathcal{O}(TK^T)$
  - Infeasible !

$$\begin{aligned}
P(X|M) &= \sum_{Q\in M} P(X,Q|M) \\
&\overset{\text{Bayes Theorem}}{=} \sum_{Q\in M} P(X|Q,M)P(Q|M) \\
&= \sum_{Q\in M} \prod_{t=1}^{T} p(x_t|q^t) \prod p_{q^{t-1},q^t} \\
&= \sum_{Q\in M} \prod_{t=1}^{T} p(x_t|q^t)\, p_{q^{t-1},q^t}
\end{aligned}$$

Forward Recurrence - Log Space

1. Initialization:
   - $\alpha_1(i) = \pi_i\, b_i(x_1), \quad 1 \leq i \leq K$

2. Recursion:
   - $$\alpha_{t+1}(j) = [\sum_{i=1}^{K} \alpha_t(i)\, a_{ij}]\, b_j(x_{t+1})$$

3.Termination:
   - $$P(X\,|\,M) = \sum_{i=1}^{K} \alpha_T(i)$$

1. Initialization:
   - $\alpha_1^{(\log)}(i) = \log \pi_i + \log b_i(x_1)$

3. Recursion:
   - $\alpha_{t+1}^{(\log)}(j) = [\text{logsum}_{i=1}^{K}(\alpha_t^{(\log)}(i) + \log a_{ij})] + \log b_j(x_{t+1})$

6.Termination:
   - $\log P(X\,|\,M) = [\text{logsum}_{i=1}^{K} \alpha_T^{(\log)}(i)]$

Decoding Problem

Decoding Problem

- Estimating an optimal sequence of states given a sequence of observations and the parameters of a model.
  - Viterbi algorithm

Viterbi Algorithm - Log Space

1. Initialization:
   - $\delta_1^{(\log)}(i) = \log \pi_i + \log b_i(x_1)$
   - $\psi_1(i) = 0$
2. Recursion:
   - $\delta_t^{(\log)}(i) = \max\limits_{1 \leq i \leq K} [\delta_{t-1}^{(\log)}(i) + \log a_{ij}] + \log b_j(x_t)$
   - $\psi_t(j) = \text{argmax}_{1 \leq i \leq K}[\delta_{t-1}^{(\log)}(i) + \log a_{ij}]$
3. Termination:
   - $\log P^*(X \mid \Theta) = \max\limits_{1 \leq i \leq K} \delta_T^{(\log)}(i)$
   - $q_T^* = \text{argmax}_{1 \leq i \leq K}[\delta_T^{(\log)}(i)]$
4. Backtracking:
   - $q^{t^*} = \psi_{t+1}(q^{t+1^*})$

Viterbi Algorithm

In summary, given a:

- Sequence of observations $X = \{x_1, \ldots, x_n, \ldots x_T\}$
- Parameters Θ

The Viterbi algorithm returns the:

- Optimal path $Q* = \{q_1^*, \ldots, q_T^*\}$
- Likelihood along the best path $P*(X|\Theta)$