EE 5345
Biomedical Instrumentation
Lecture 21: slides 396-416

Carlos E. Davila, Electrical Engineering Dept.
Southern Methodist University
slides can be viewed at:
http:// www.seas.smu.edu/~cd/ee5345.html

Example (cont.)

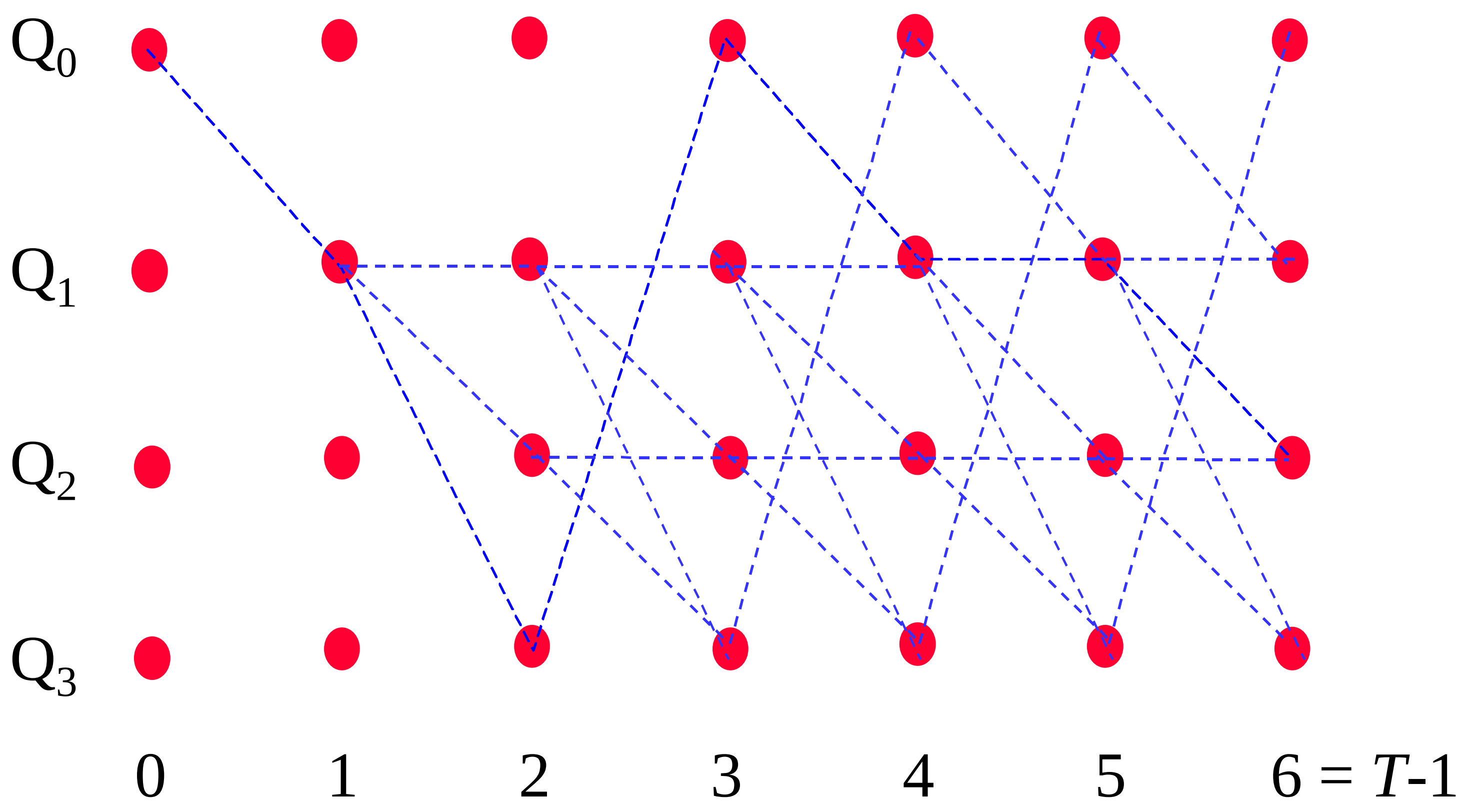
$\Pr(\, I \,) = 0.03$
$\Pr(\, O|I \,) = \; 0.0003024$

$$\Rightarrow \Pr(O \cap I) = 0.03 \times 0.0003024 = 9.072 \times 10^{-6}$$

ex) How many possible state sequences are there?

- in general, there are on the order of N^T possible state sequences, (for Example 1, that's $4^7 = 16{,}384$).
- Since some of the transition probabilities are zero, this number decreases to only 30.
- Let each state sequence be denoted by I_i, $i = 1, \ldots,$ $R=O(N^T)$.

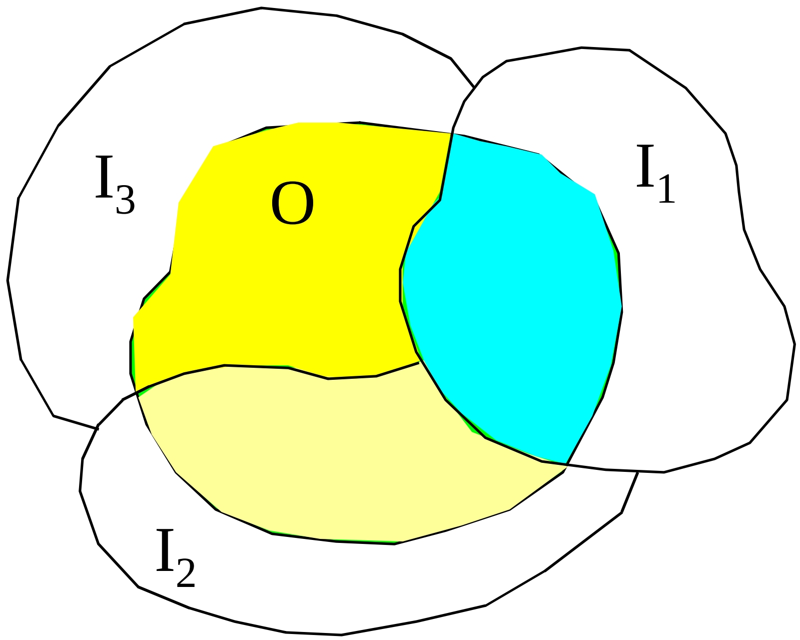
Total Number of Possible State Sequences: 30



Distributive-Type Property

since $I_i,\ i = 1, \ldots R \equiv O\left(N^T\right)$ are disjoint events:

R = 3



$$\Pr\left(\sum_{i=1}^{R} O \cap I_i\right) = \sum_{i=1}^{R} \Pr\left(O \cap I_i\right) = \Pr\left(O\right)$$

(by axiom 2)

since *R* is so large, this is not a practical solution to Problem 1

Solution to Problem 1: Forward-Backward Algorithm

We seek $\Pr(O | l)$

Forward variable:

$$a_t(i) = \Pr(O_0, O_1, \ldots, O_t, i_t = Q_i | l)$$

- this is the probability that we observe the partial observation sequence, $O_0, O_1, \ldots, O_t$ and arrive at state Q_i at time t (given the model λ).
- In the forward-backward algorithm the forward variable is updated recursively.
- Note that the events $O_0, O_1, \ldots, O_t, i_t = Q_i$ are disjoint for each Q_i.

Forward-Backward Algorithm (cont.)

$$a_0(i) = p_i b_i(O_0), \quad 0 \le i \le N-1$$

- for $t = 0, 1, \ldots, T\text{-}2,\ 0 \le j \le N\text{-}1$

$$a_{t+1}(j) = \left[\sum_{i=0}^{N-1} a_t(i) a_{ij}\right] b_j(O_{t+1})$$

- then,

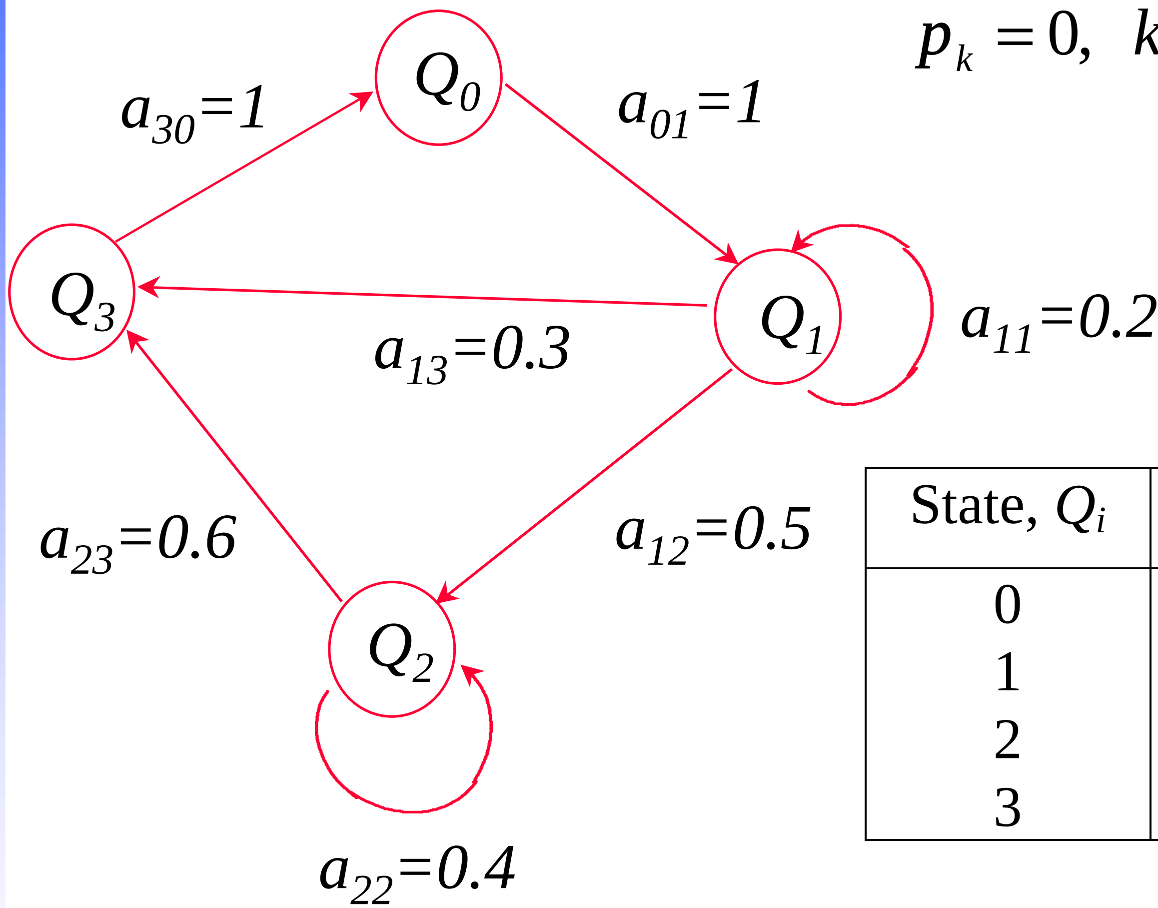
$$\Pr(O|\,l\,) = \sum_{i=0}^{N-1} a_{T-1}(i)$$

the algorithm can be easily implmented via arithmetic involving the matrices A, B, and Π.

Application of Forward-Backward Algorithm to Example 1

$$p_0 = 1$$

$$p_k = 0, \quad k \neq 0$$



State, Q_i	$b_i(R)$	$b_i(B)$	$b_i(Y)$
0	0.3	0.2	0.5
1	0.7	0.2	0.1
2	0.9	0	0.1
3	0.2	0.8	0

- observed output sequence: *R*, *Y*, *B*, *B*, *R*, *Y*, *R*
- we don't know the state sequence

Application of Forward-Backward Algorithm to Example 1 (cont.)

$a_t(j)$

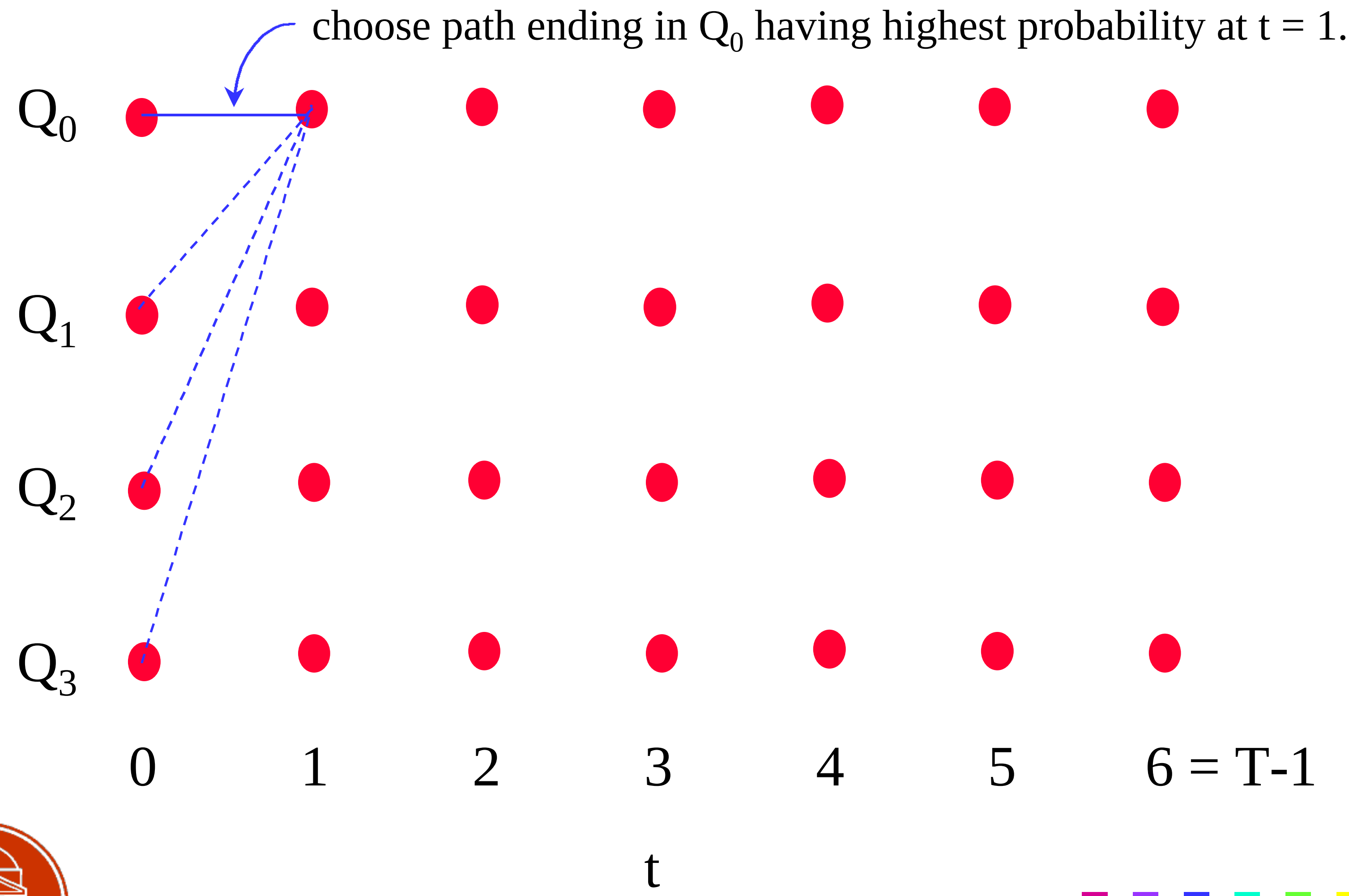
t \ j	0	1	2	3
0	0.3	0	0	0
1	0	0.03	0	0
2	0	1.2E-2	0	7.2E-3
3	1.44E-3	4.8E-5	0	2.88E-4
4	8.64E-5	1.0147E-3	2.16E-5	2.88E-6
5	1.44E-6	2.8934E-5	5.15E-5	0
6	0	5.0588E-6	3.1596E-5	7.9281E-6

$$\Pr(O|\, l\,) = 4.4582\text{E} - 5$$

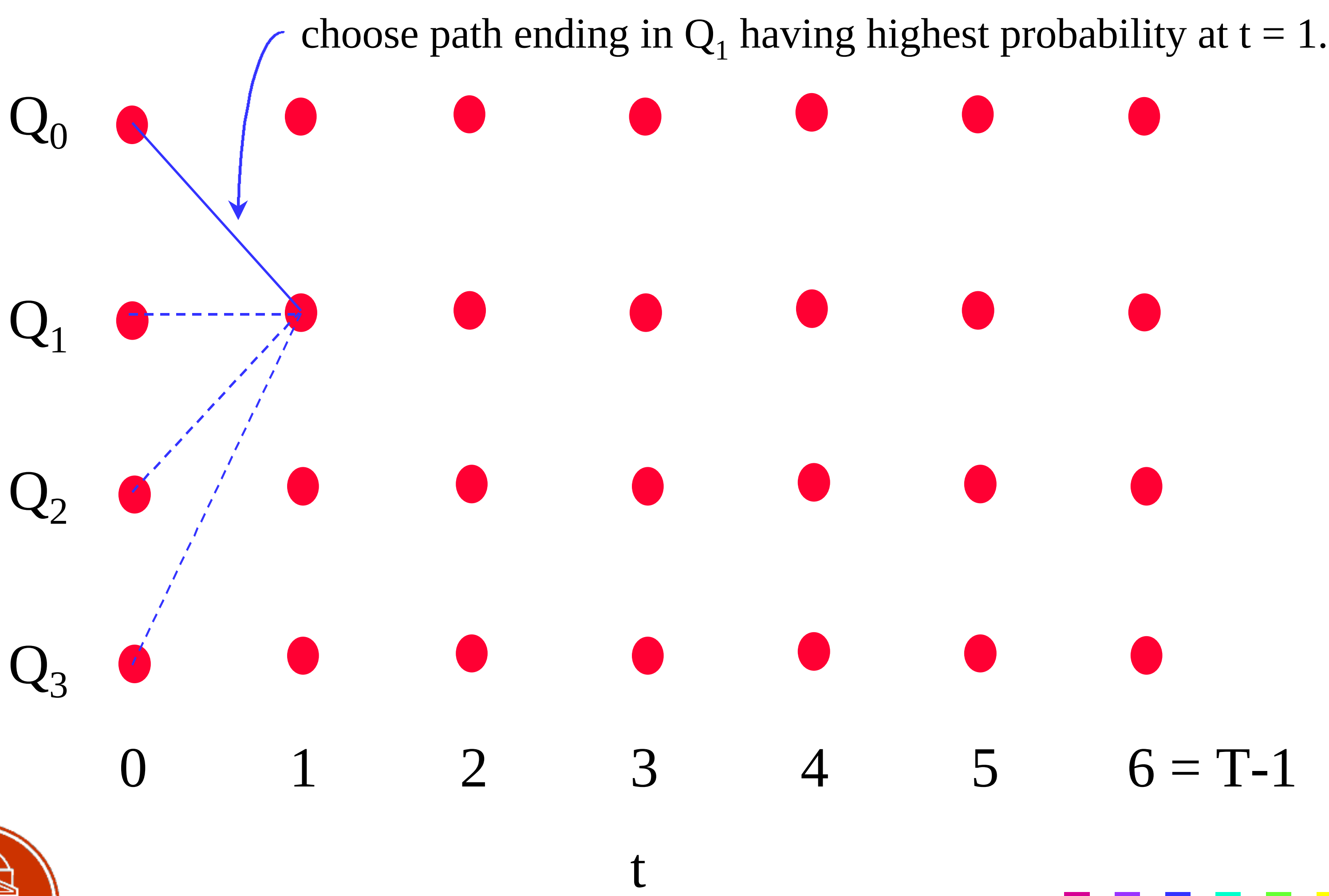
Solution to Problem 2: The Viterbi Algorithm

- We seek the state sequence that maximizes $\Pr(I|O, l)$
- This is equivalent to maximizing $\Pr(I \cap O)$ (given λ)
- The trellis diagram representation of HHM's is useful in this regard. We seek the path through the trellis that has the maximum $\Pr(I \cap O)$
- At each column (time step) in the trellis, the Viterbi algorithm eliminates all but *N* possible state sequences.
- At each time step, the *N* retained sequences all end in different states.
- If more than one sequence ends in the same state, the sequence with the maximum probability is retained.

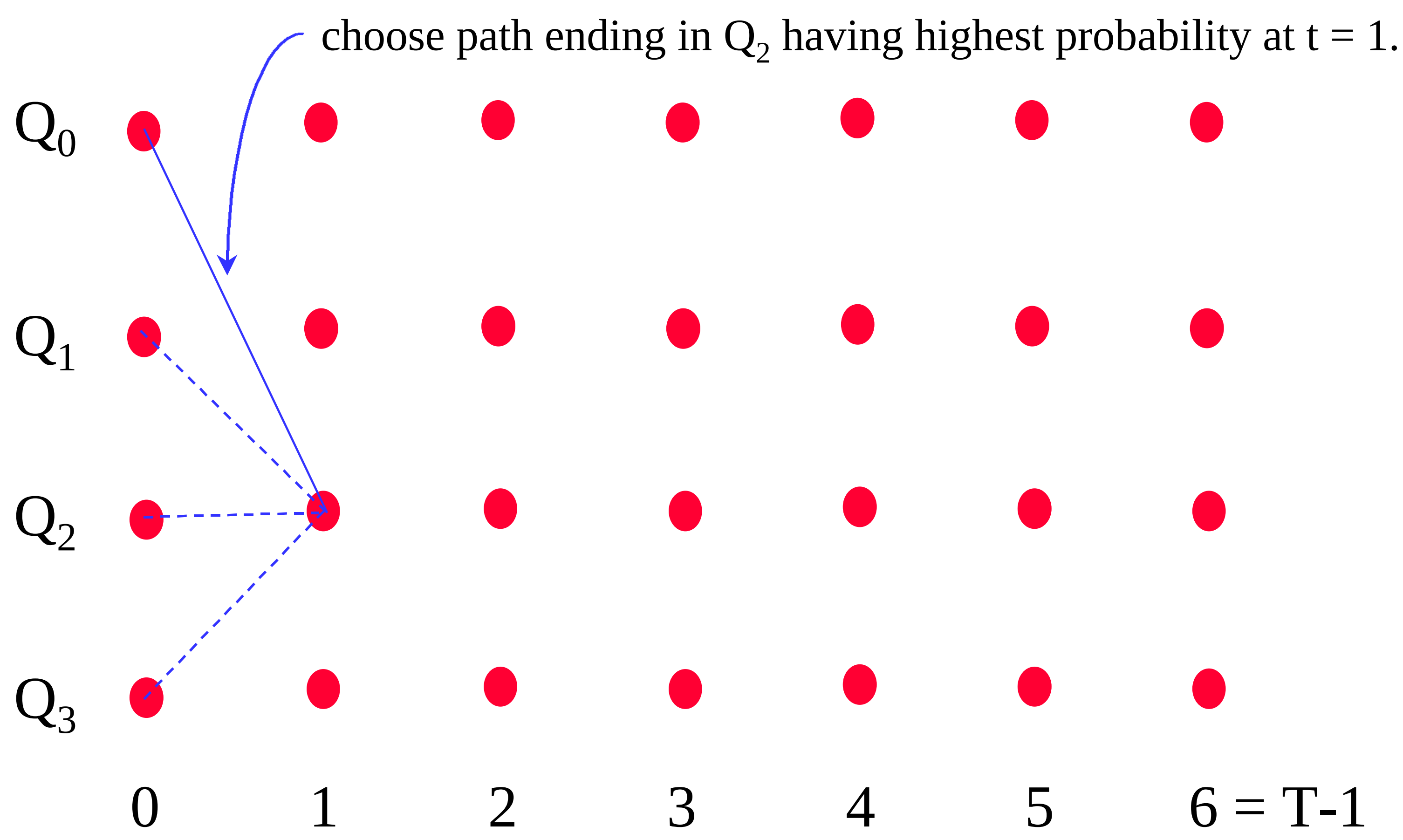
Viterbi Algorithm (cont.)



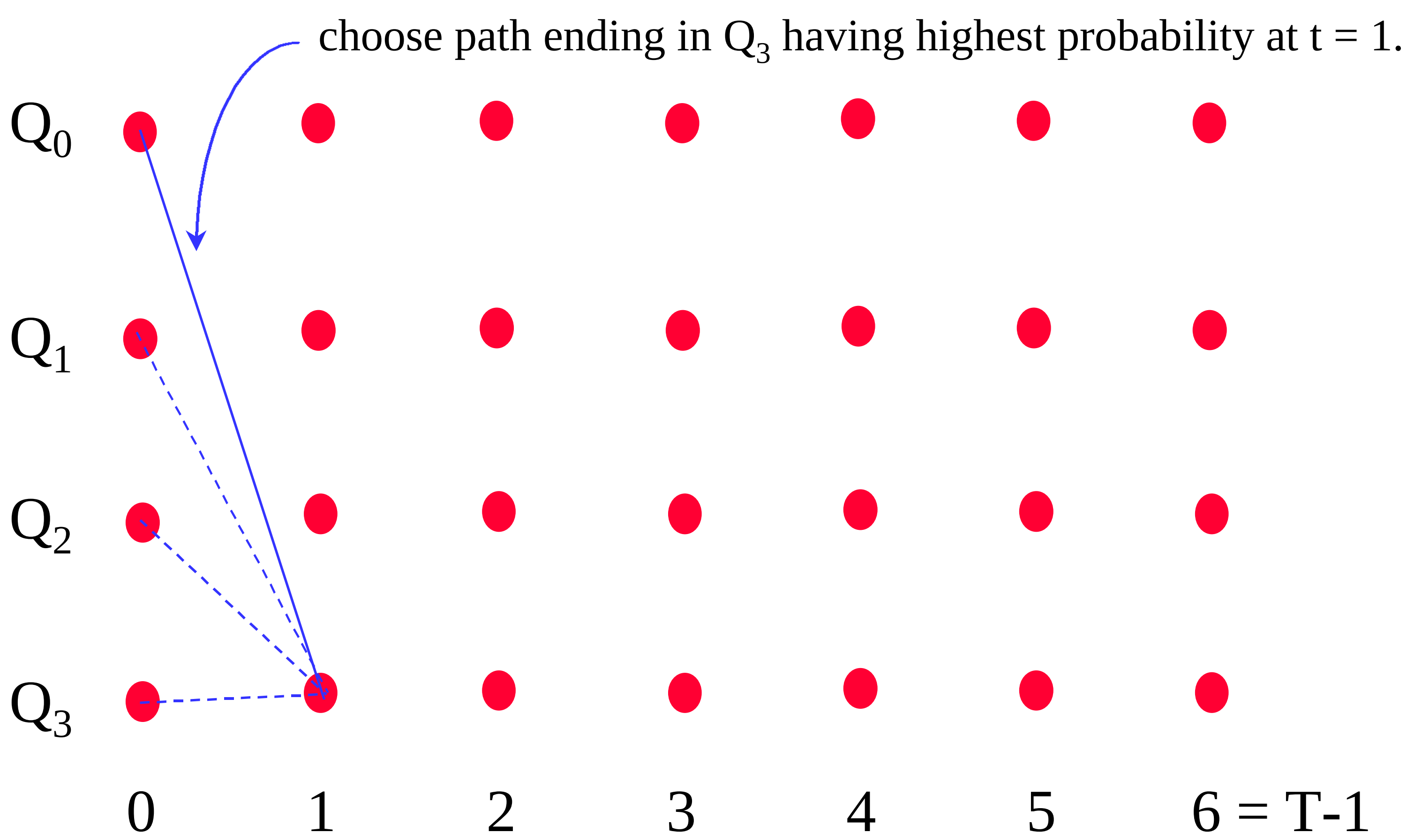
Viterbi Algorithm (cont.)



Viterbi Algorithm (cont.)

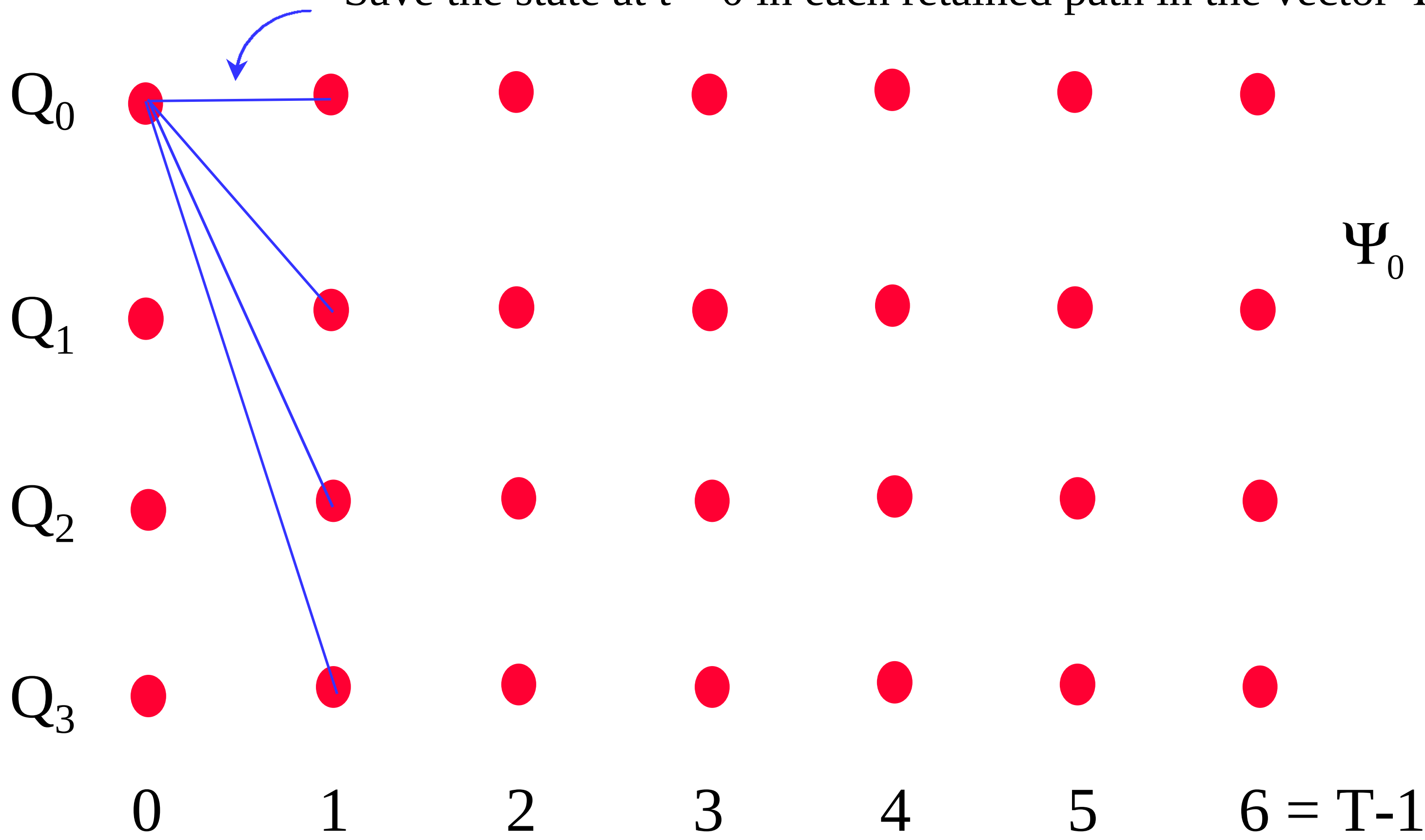


Viterbi Algorithm (cont.)



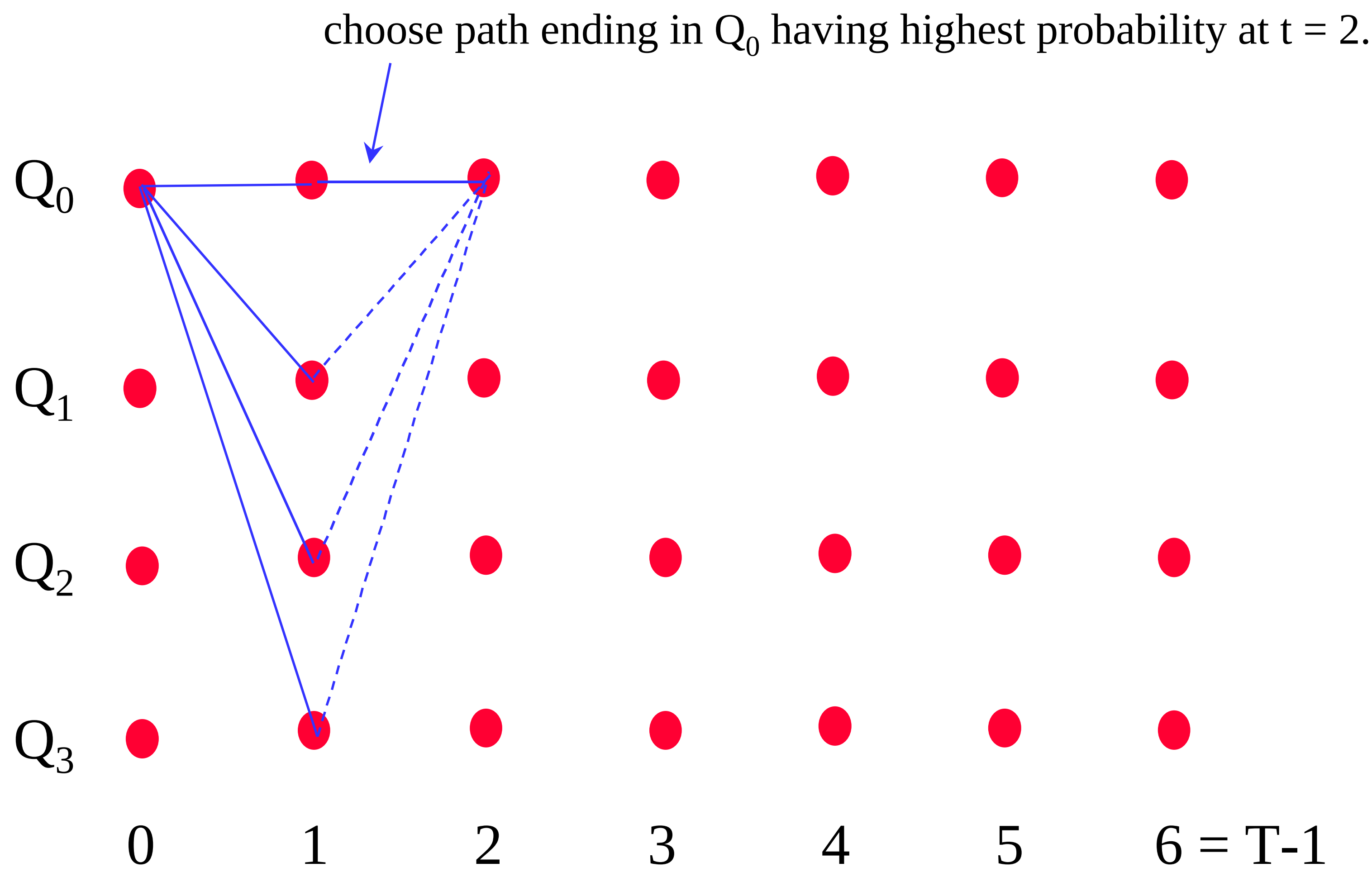
Viterbi Algorithm (cont.)

Save each of the N = 4 maximum probabilities in the vector δ_t
Save the state at t = 0 in each retained path in the vector Ψ_t

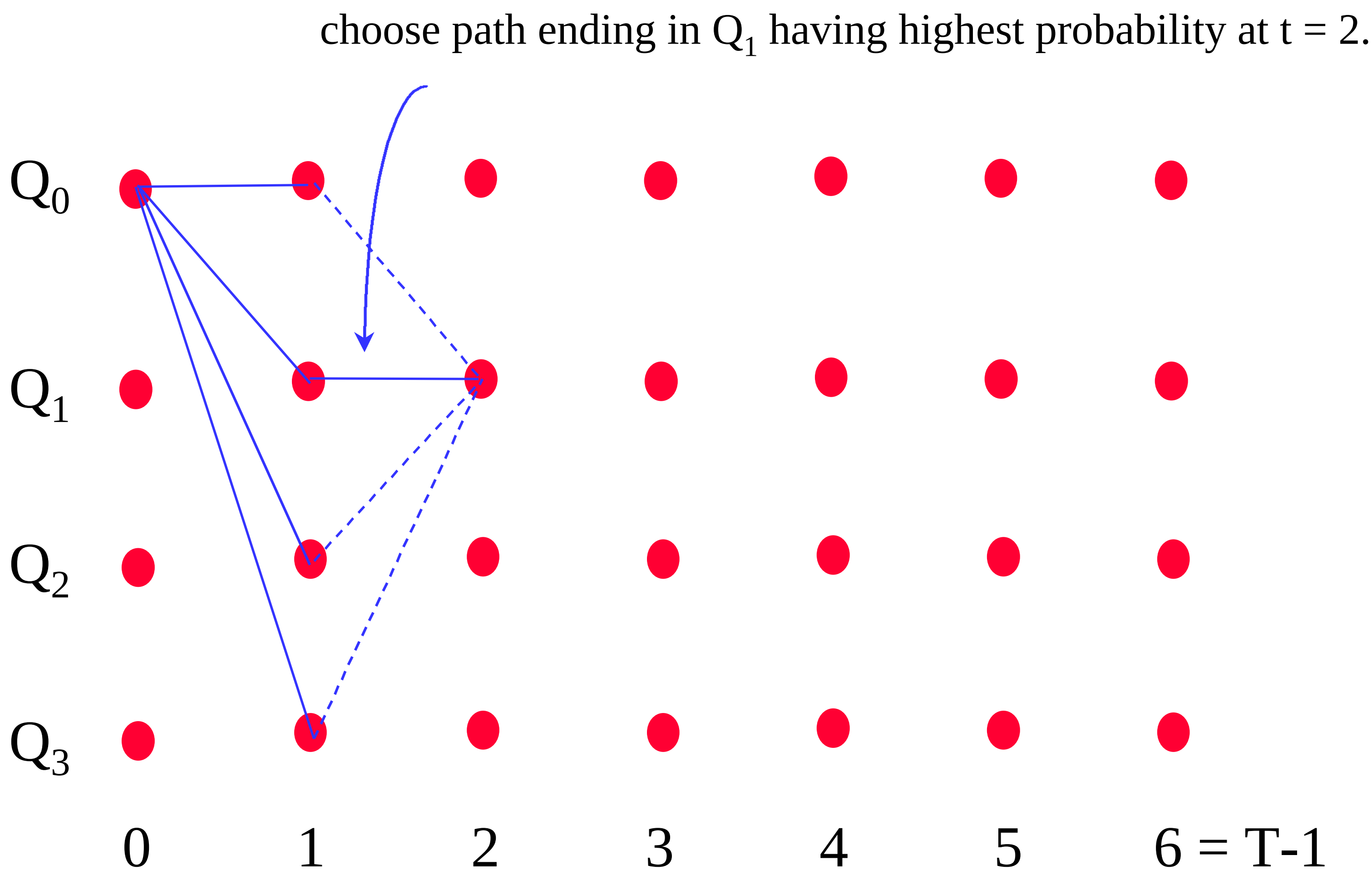


$$\Psi_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

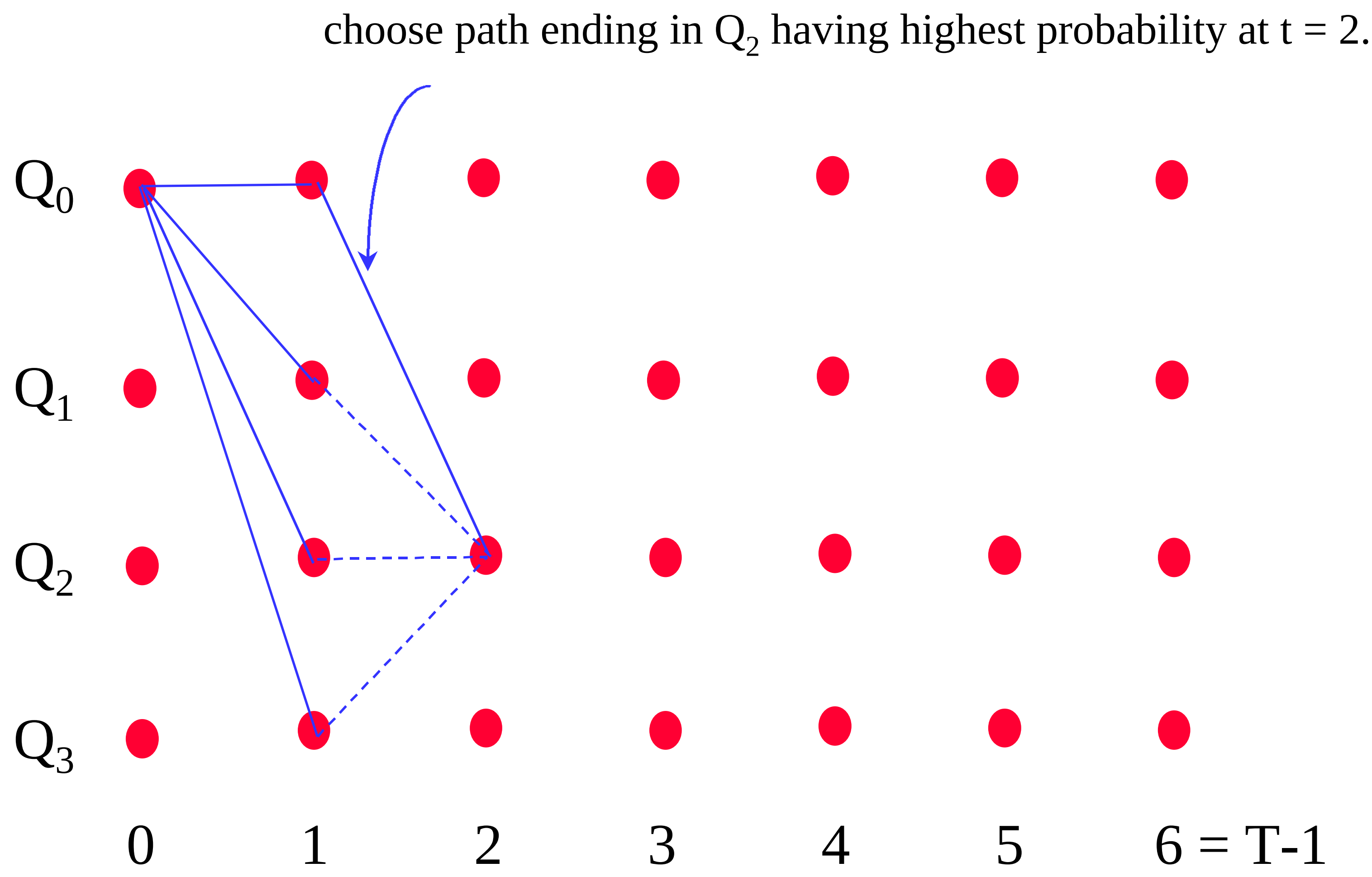
Viterbi Algorithm (cont.)



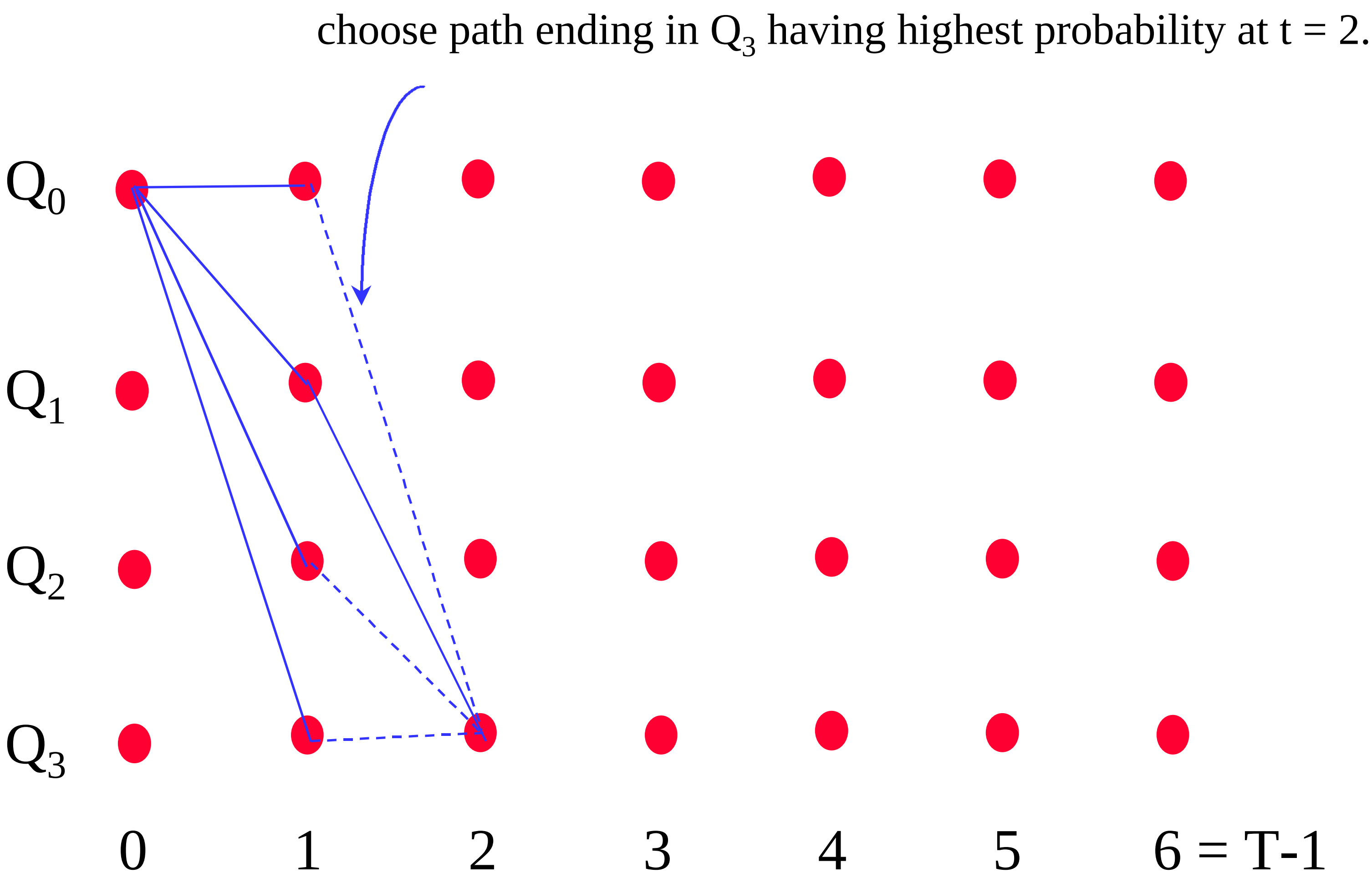
Viterbi Algorithm (cont.)



Viterbi Algorithm (cont.)

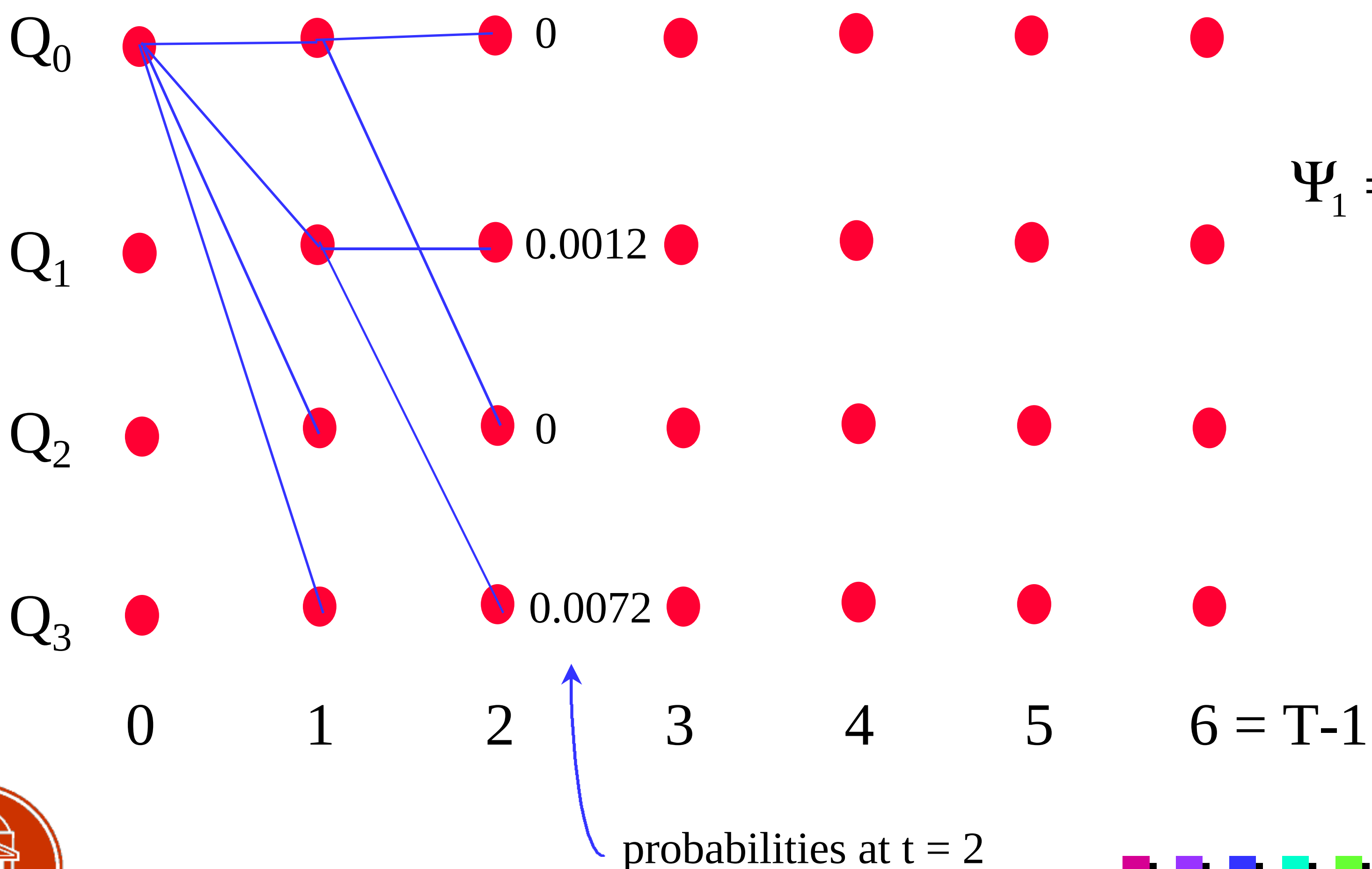


Viterbi Algorithm (cont.)



Viterbi Algorithm (cont.)

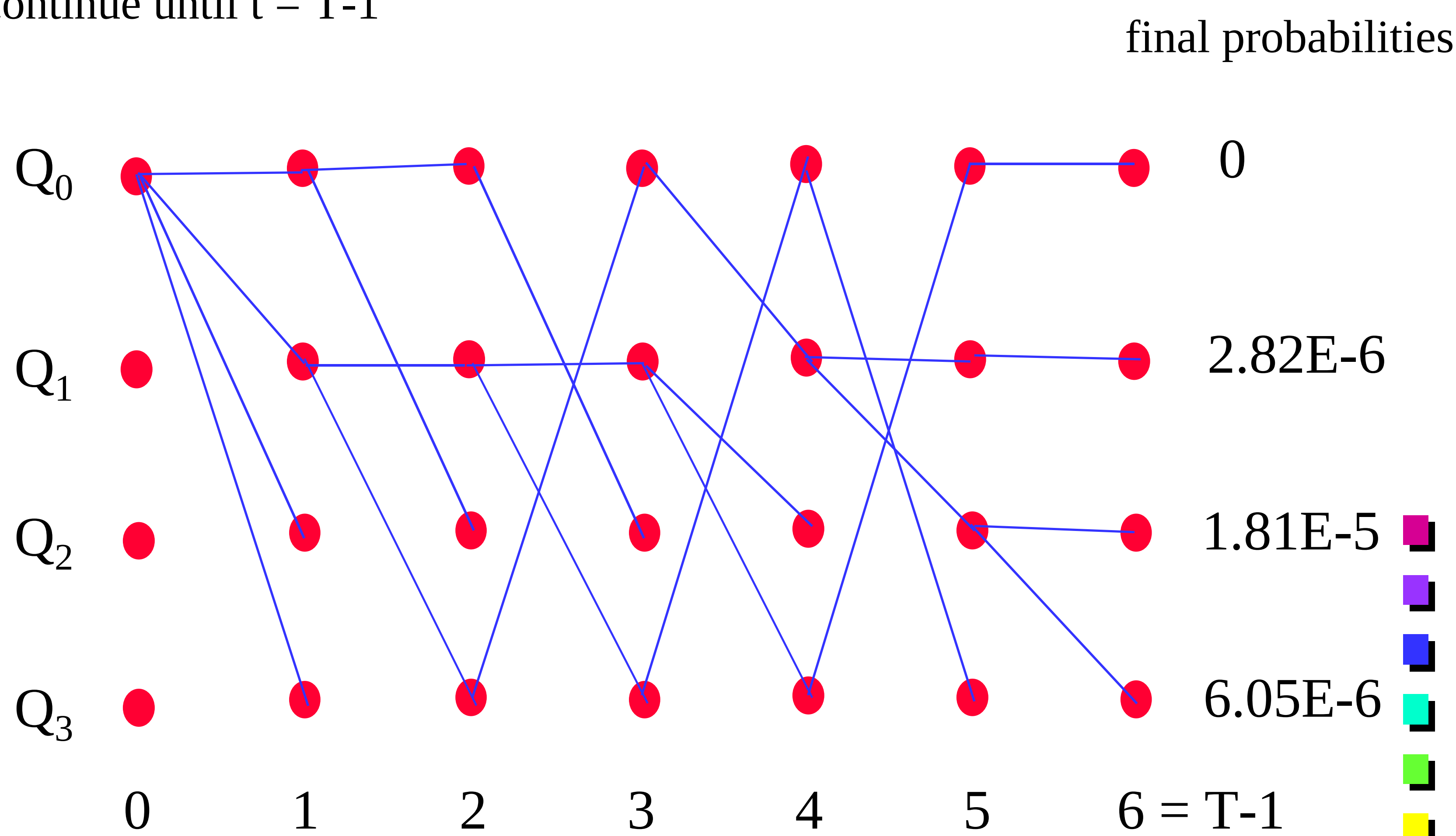
Save each of the N = 4 maximum probabilities in the vector δ_2
Save the state at t = 1 in each retained path in the vector Ψ_1



$$\Psi_1 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}$$

Viterbi Algorithm (cont.)

continue until t = T-1



Viterbi Algorithm (cont.)

- maximum final probability defines best path
- must backtrack through the Ψ_t to find it