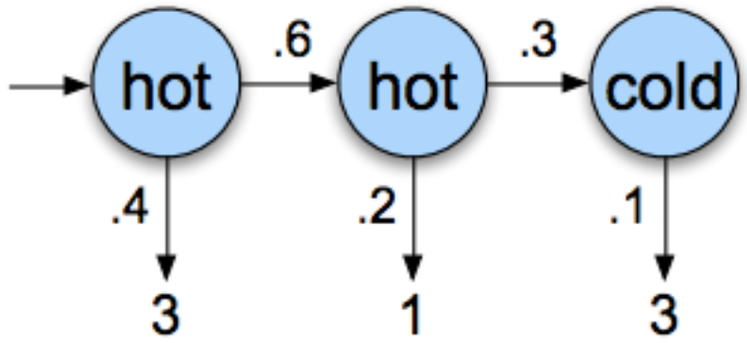
Joint and marginal probabilities

Joint:

$$P(O,Q) = P(O|Q) \times P(Q) = \prod_{i=1}^{n} P(o_i|q_i) \times \prod_{i=1}^{n} P(q_i|q_{i-1})$$

$$\begin{aligned} P(3\ 1\ 3, \text{hot hot cold}) &= P(\text{hot}|\text{start}) \times P(\text{hot}|\text{hot}) \times P(\text{cold}|\text{hot}) \\ &\quad \times P(3|\text{hot}) \times P(1|\text{hot}) \times P(3|\text{cold}) \end{aligned}$$



Marginal:

$$P(O) = \sum_{Q} P(O,Q) = \sum_{Q} P(O|Q)P(Q)$$

$$P(3\ 1\ 3) = P(3\ 1\ 3, \text{cold cold cold}) + P(3\ 1\ 3, \text{cold cold hot}) + P(3\ 1\ 3, \text{hot hot cold}) + \ldots$$

How to compute the probability of observations

Computing Likelihood: Given an HMM $\lambda = (A, B)$ and an observation sequence O, determine the likelihood $P(O|\lambda)$.

$$P(O) = \sum_{Q} P(O, Q) = \sum_{Q} P(O|Q)P(Q)$$

For an HMM with N hidden states and an observation sequence of T observations, there are N^T possible hidden sequences. For real tasks, where N and T are both large, N^T is a very large number, so we cannot compute the total observation likelihood by computing a separate observation likelihood for each hidden state sequence and then summing them.

$\alpha_t(j) = P(o_1, o_2 \ldots o_t, q_t = j|\lambda)$ represents the probability of being in state j after seeing the first t observations, given the automaton λ. The value of each cell $\alpha_t(j)$ is computed by summing over the probabilities of every path that could lead us to this cell.

Here, $q_t = j$ means "the tth state in the sequence of states is state j". We compute this probability $\alpha_t(j)$ by summing over the extensions of all the paths that lead to the current cell. For a given state q_j at time t, the value $\alpha_t(j)$ is computed as

$$\alpha_t(j) = \sum_{i=1}^{N} \alpha_{t-1}(i) a_{ij} b_j(o_t)$$

Forward algorithm

$\alpha_{t-1}(i)$	the **previous forward path probability** from the previous time step
a_{ij}	the **transition probability** from previous state q_i to current state q_j
$b_j(o_t)$	the **state observation likelihood** of the observation symbol o_t given the current state j

1. Initialization:

$$\alpha_1(j) = a_{0j} b_j(o_1) \quad 1 \le j \le N$$

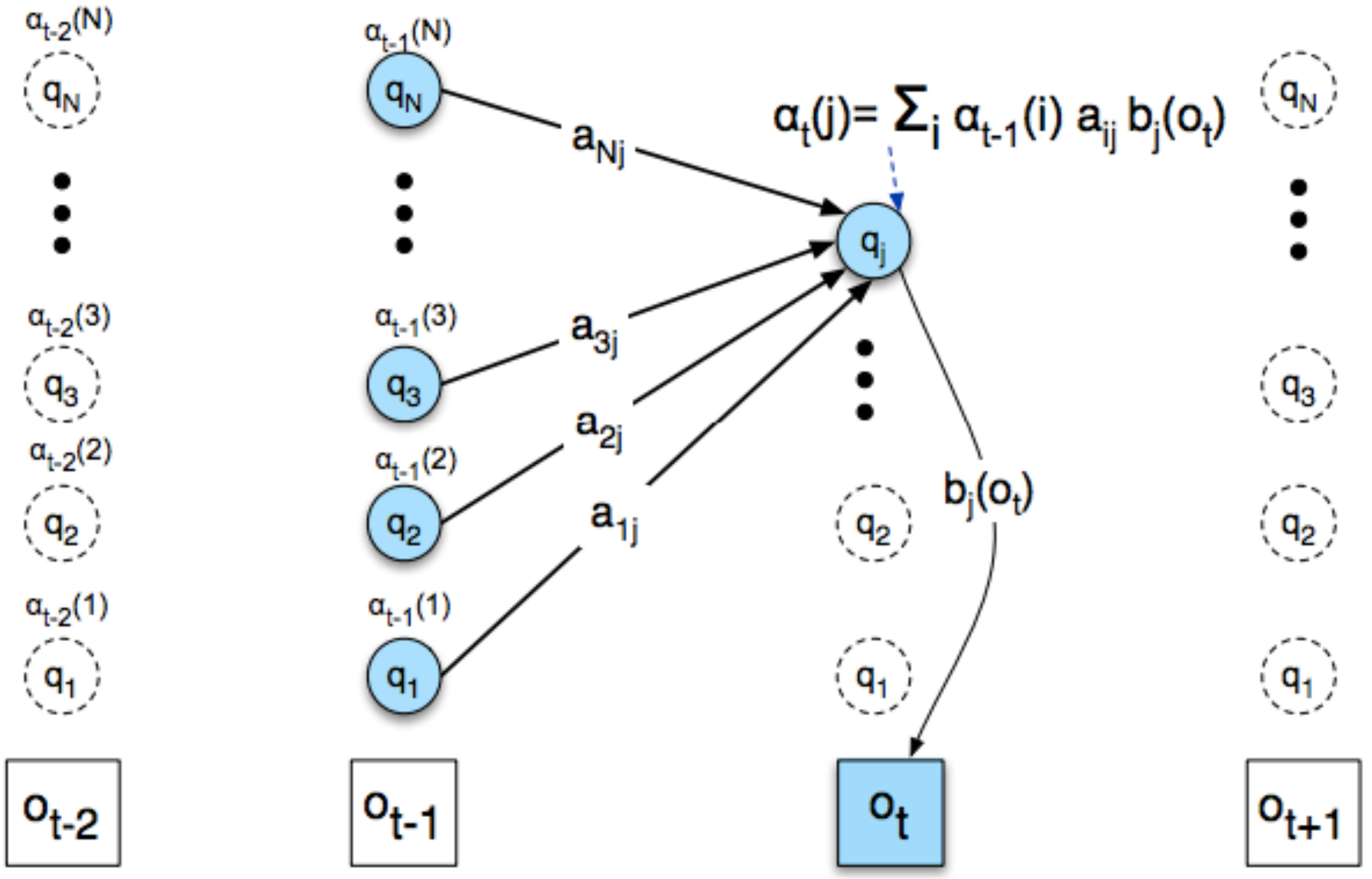
2. Recursion (since states 0 and F are non-emitting):

$$\alpha_t(j) = \sum_{i=1}^{N} \alpha_{t-1}(i) a_{ij} b_j(o_t); \quad 1 \le j \le N, 1 < t \le T$$

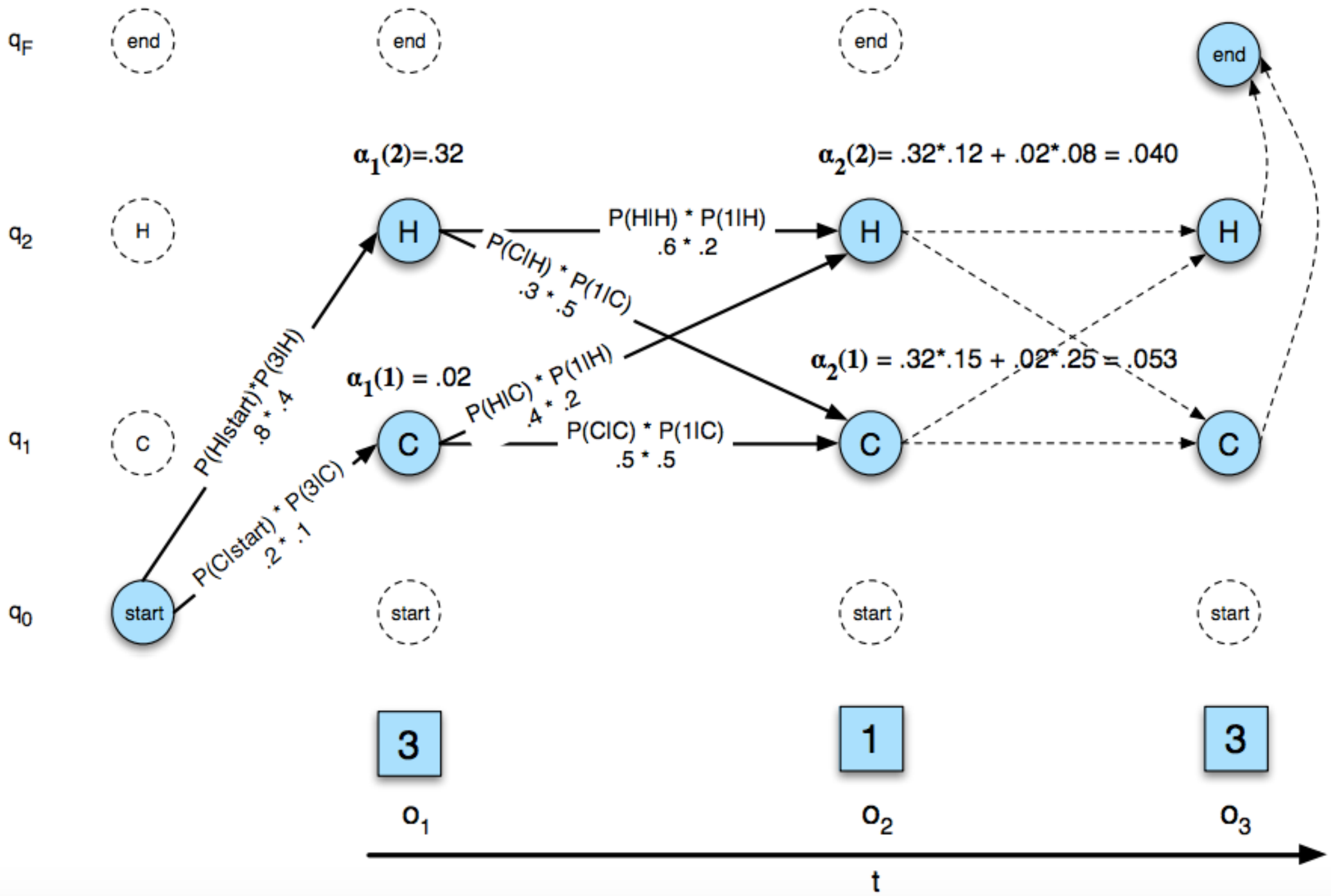
3. Termination:

$$P(O|\lambda) = \alpha_T(q_F) = \sum_{i=1}^{N} \alpha_T(i)\, a_{iF}$$

Forward algorithm



Forward algorithm



Decoding: finding the most probable states

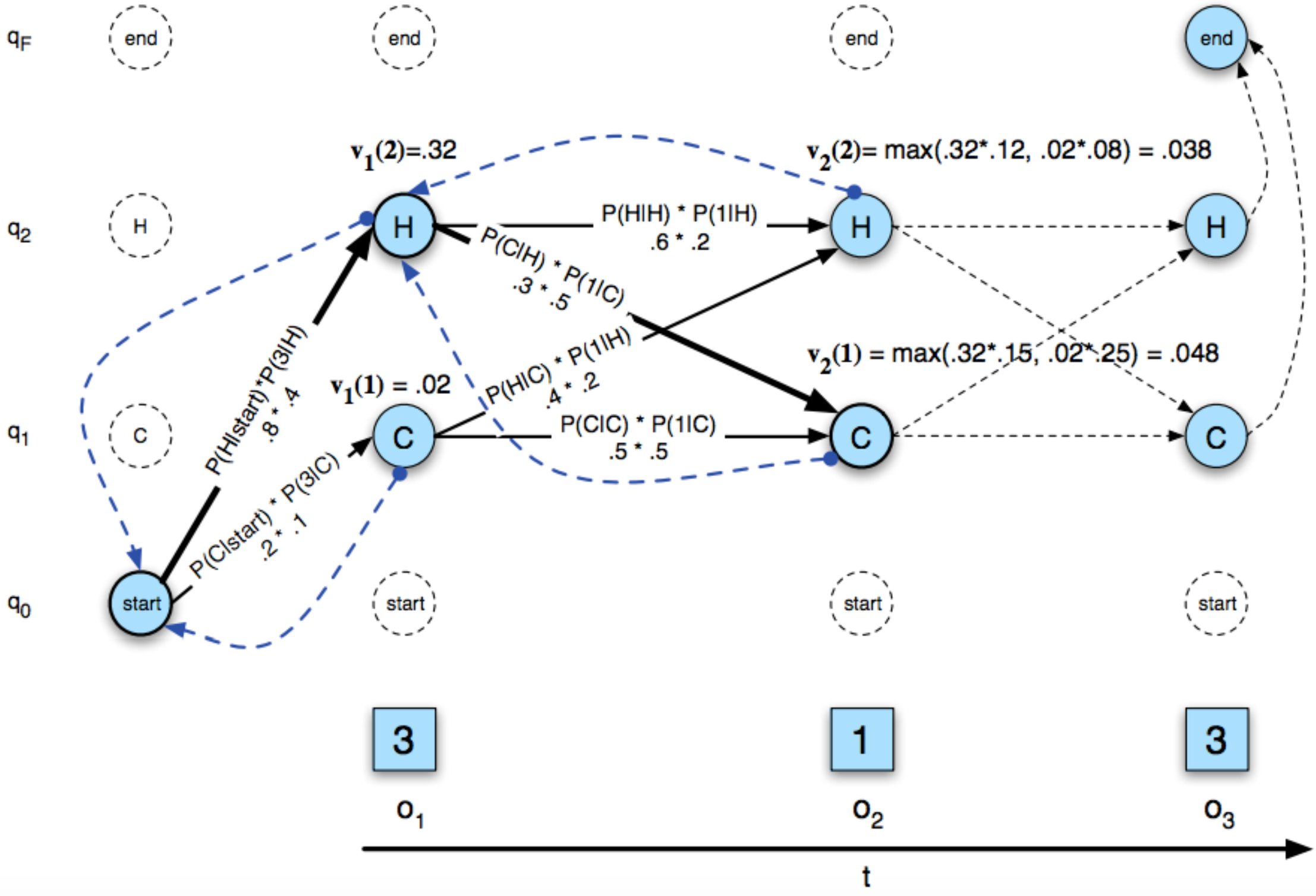
Decoding: Given as input an HMM $\lambda = (A, B)$ and a sequence of observations $O = o_1, o_2, ..., o_T$, find the most probable sequence of states $Q = q_1 q_2 q_3 \ldots q_T$.

Similar to the forward algorithm, we can define the following value:

$$v_t(j) = \max_{q_0, q_1, \cdots, q_{t-1}} P(q_0, q_1 \ldots q_{t-1}, o_1, o_2 \ldots o_t, q_t = j | \lambda)$$

$$v_t(j) = \max_{i=1}^{N} v_{t-1}(i)\, a_{ij}\, b_j(o_t)$$

$v_{t-1}(i)$	the **previous Viterbi path probability** from the previous time step
a_{ij}	the **transition probability** from previous state q_i to current state q_j
$b_j(o_t)$	the **state observation likelihood** of the observation symbol o_t given the current state j


q_F
q_2
q_1
q_0
end
end
end
end
H
C
start
start
start
start
$v_1(2)$=.32
$v_2(2)$= max(.32*.12, .02*.08) = .038
$v_1(1)$ = .02
$v_2(1)$ = max(.32*.15, .02*.25) = .048
P(H|H) * P(1|H)
.6 * .2
P(C|H) * P(1|C)
.3 * .5
P(H|C) * P(1|H)
.4 * .2
P(C|C) * P(1|C)
.5 * .5
P(H|start)*P(3|H)
.8 * .4
P(C|start) * P(3|C)
.2 * .1
3
1
3
o_1
o_2
o_3
t

Viterbi algorithm

1. **Initialization:**

$$\begin{aligned} v_1(j) &= a_{0j} b_j(o_1) \quad 1 \le j \le N \\ bt_1(j) &= 0 \end{aligned}$$

2. **Recursion** (recall that states 0 and q_F are non-emitting):

$$v_t(j) = \max_{i=1}^{N} v_{t-1}(i)\, a_{ij}\, b_j(o_t); \quad 1 \le j \le N, 1 < t \le T$$

$$bt_t(j) = \operatorname*{argmax}_{i=1}^{N} v_{t-1}(i)\, a_{ij}\, b_j(o_t); \quad 1 \le j \le N, 1 < t \le T$$

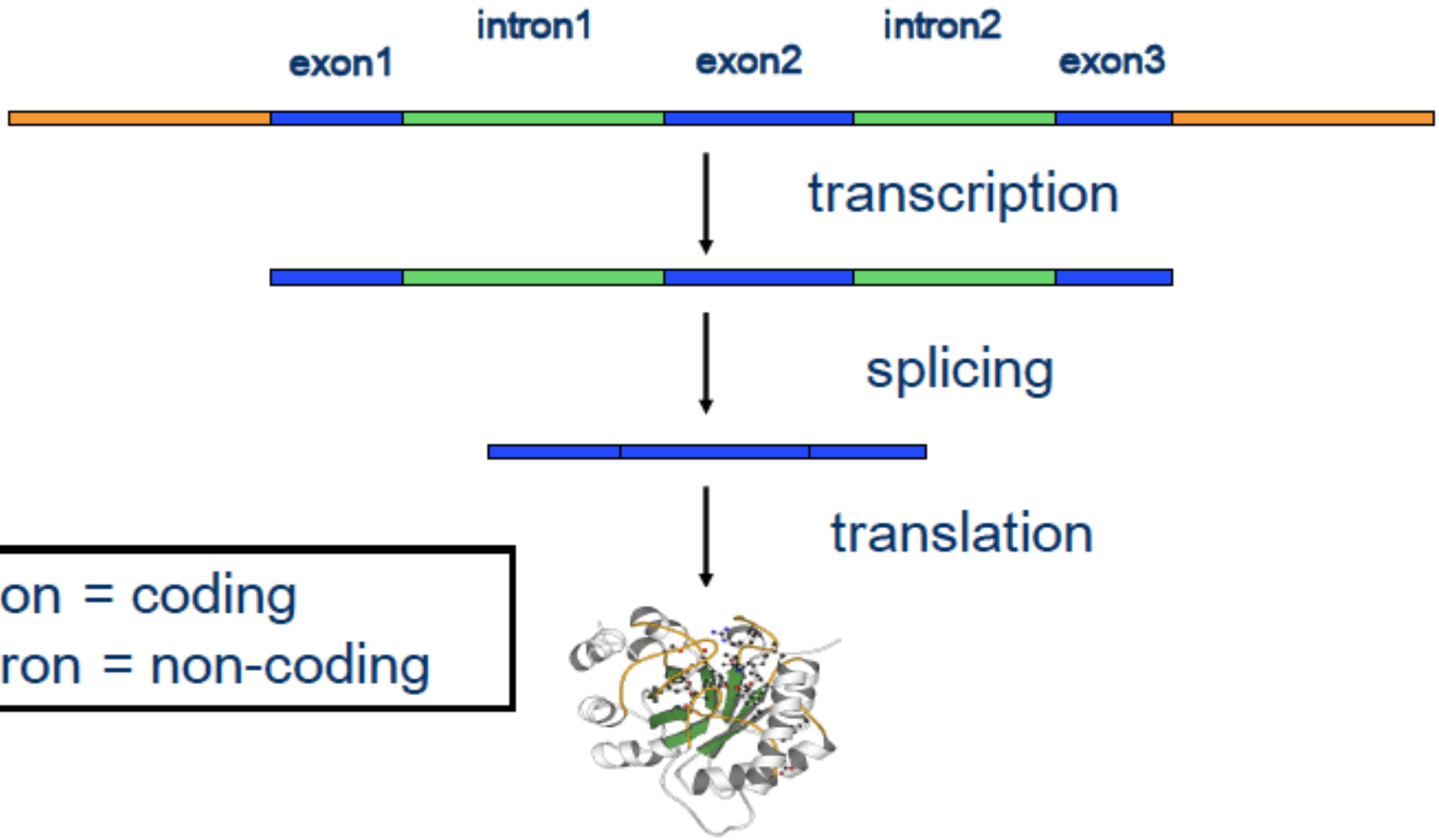
3. **Termination:**

$$\text{The best score:} \quad P* = v_T(q_F) = \max_{i=1}^{N} v_T(i) * a_{iF}$$

$$\text{The start of backtrace:} \quad q_T* = bt_T(q_F) = \operatorname*{argmax}_{i=1}^{N} v_T(i) * a_{iF}$$

Gene finding

Gene finding



exon = coding
intron = non-coding

Gene finding

- In human genome, ~3% of DNA sequence is genes
- Lot of “junk” DNA between genes, and even inside genes (between exons).
- Due to the reverse complement, one gene can start from either direction.
- Gene finding must deal with these.

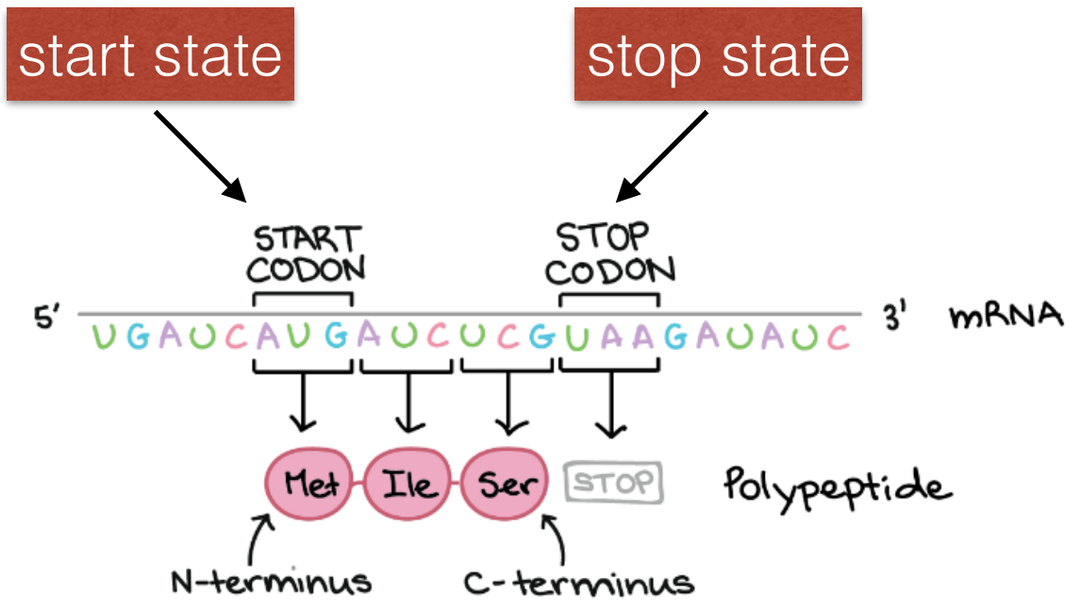
Gene finding for bacterial genomes

In bacteria, there is no intron in the coding region.

Gene finding for bacterial genomes

In bacteria, there is no intron in the coding region.

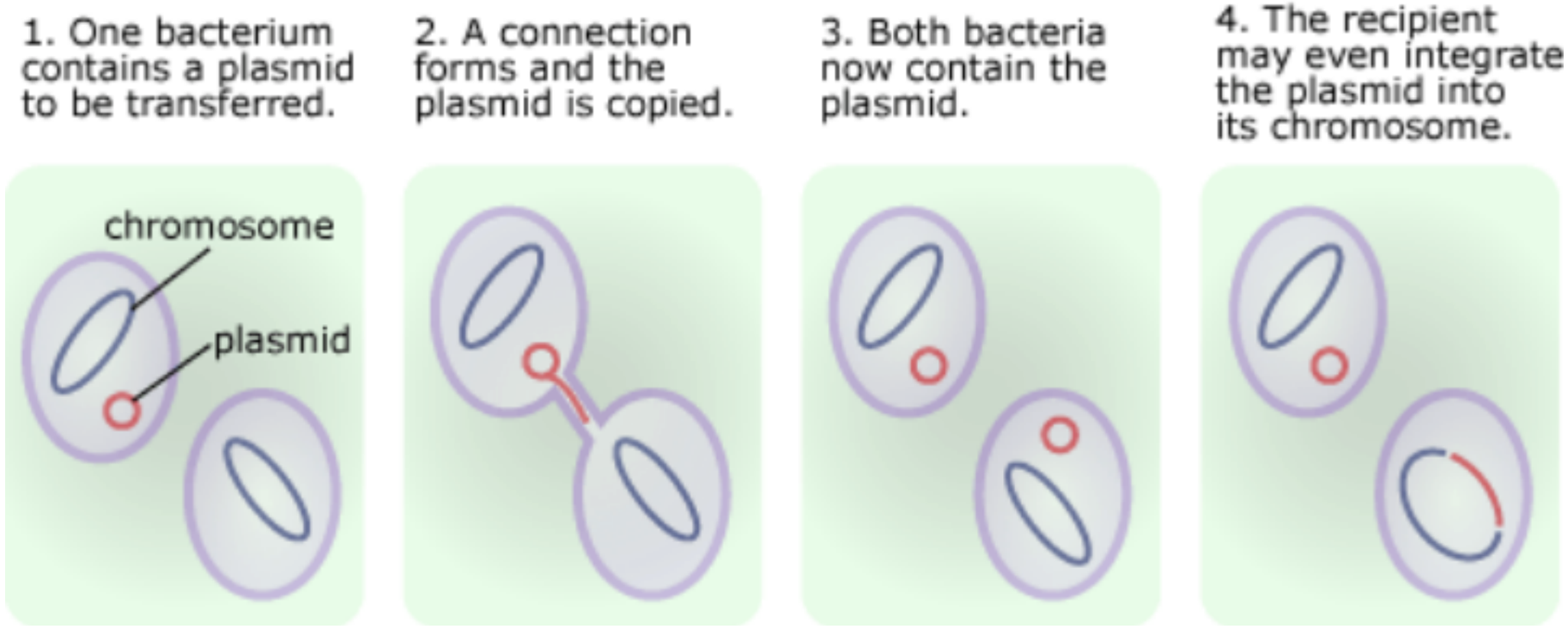
First letter	Second letter: U	Second letter: C	Second letter: A	Second letter: G	Third letter
U	UUU Phe UUC Phe UUA Leu UUG Leu	UCU Ser UCC Ser UCA Ser UCG Ser	UAU Tyr UAC Tyr **UAA Stop** **UAG Stop**	UGU Cys UGC Cys **UGA Stop** UGG Trp	U C A G
C	CUU Leu CUC Leu CUA Leu CUG Leu	CCU Pro CCC Pro CCA Pro CCG Pro	CAU His CAC His CAA Gln CAG Gln	CGU Arg CGC Arg CGA Arg CGG Arg	U C A G
A	AUU Ile AUC Ile AUA Ile **AUG Met**	ACU Thr ACC Thr ACA Thr ACG Thr	AAU Asn AAC Asn AAA Lys AAG Lys	AGU Ser AGC Ser AGA Arg AGG Arg	U C A G
G	GUU Val GUC Val GUA Val GUG Val	GCU Ala GCC Ala GCA Ala GCG Ala	GAU Asp GAC Asp GAA Glu GAG Glu	GGU Gly GGC Gly GGA Gly GGG Gly	U C A G



Codon usage can be different between the noncoding regions and coding regions.

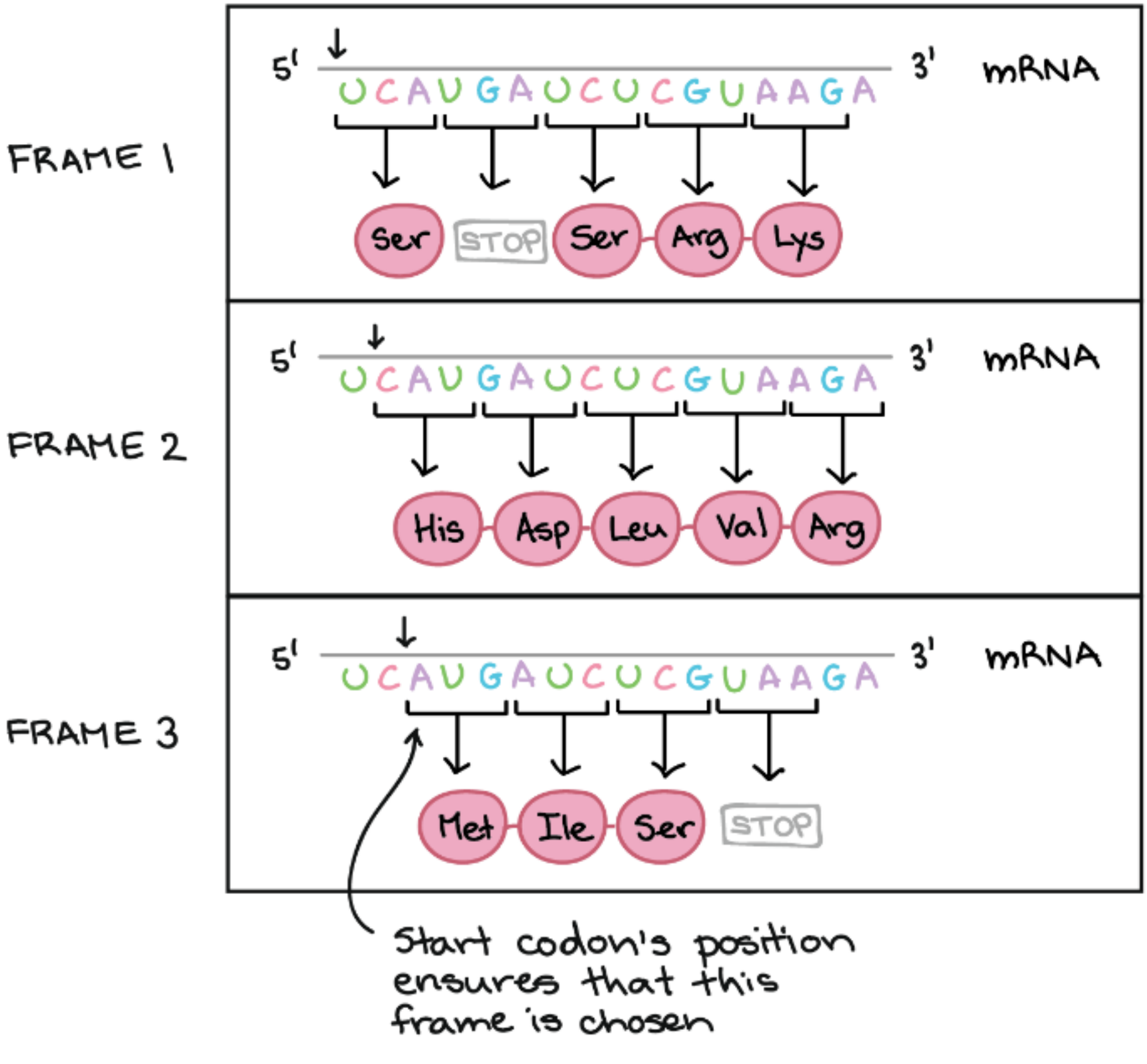
Gene finding for bacterial genomes

Horizontal gene transfer



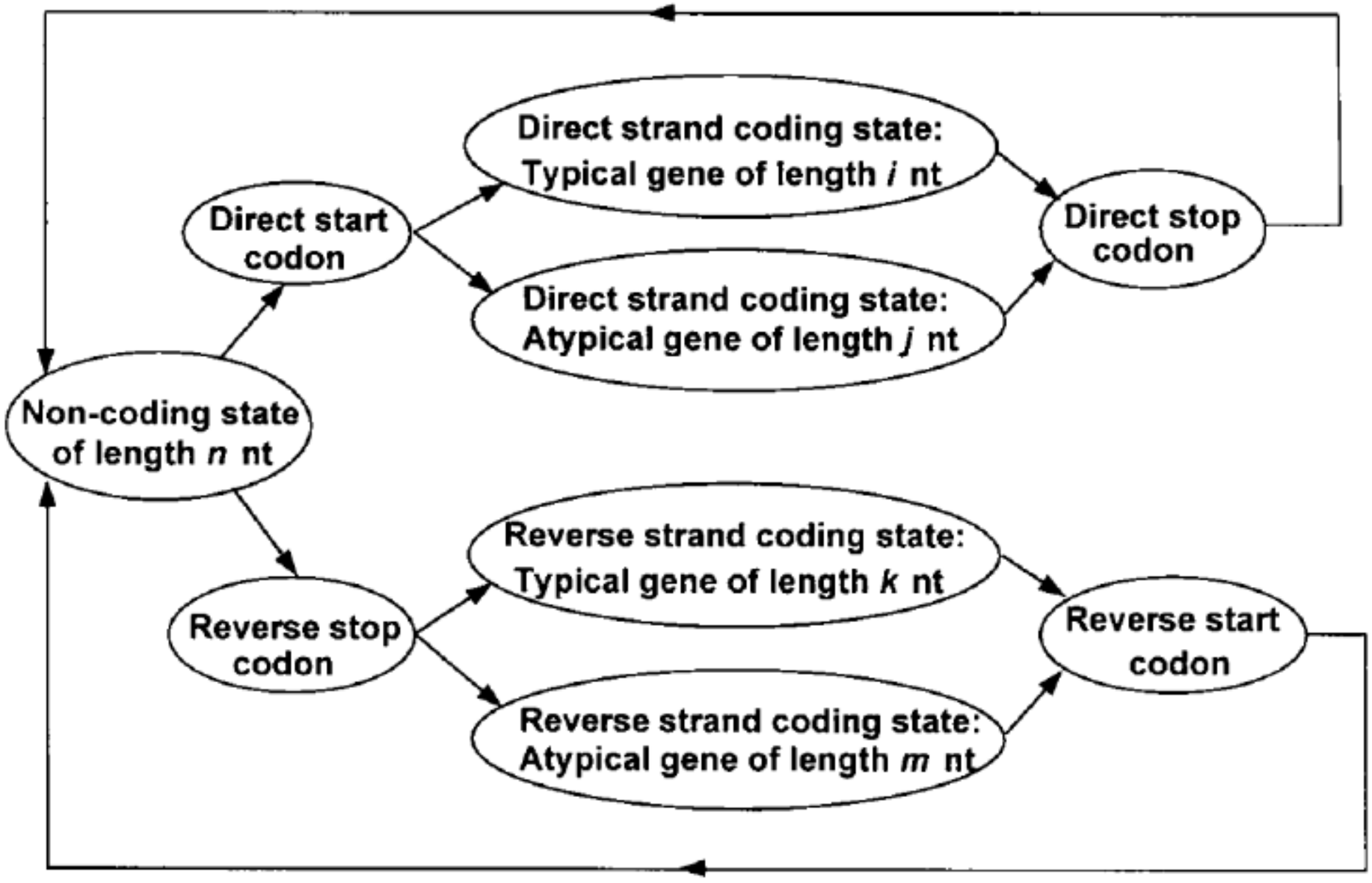
Codon usages can be different in the typical coding regions and the atypical coding regions.

Gene finding: frames

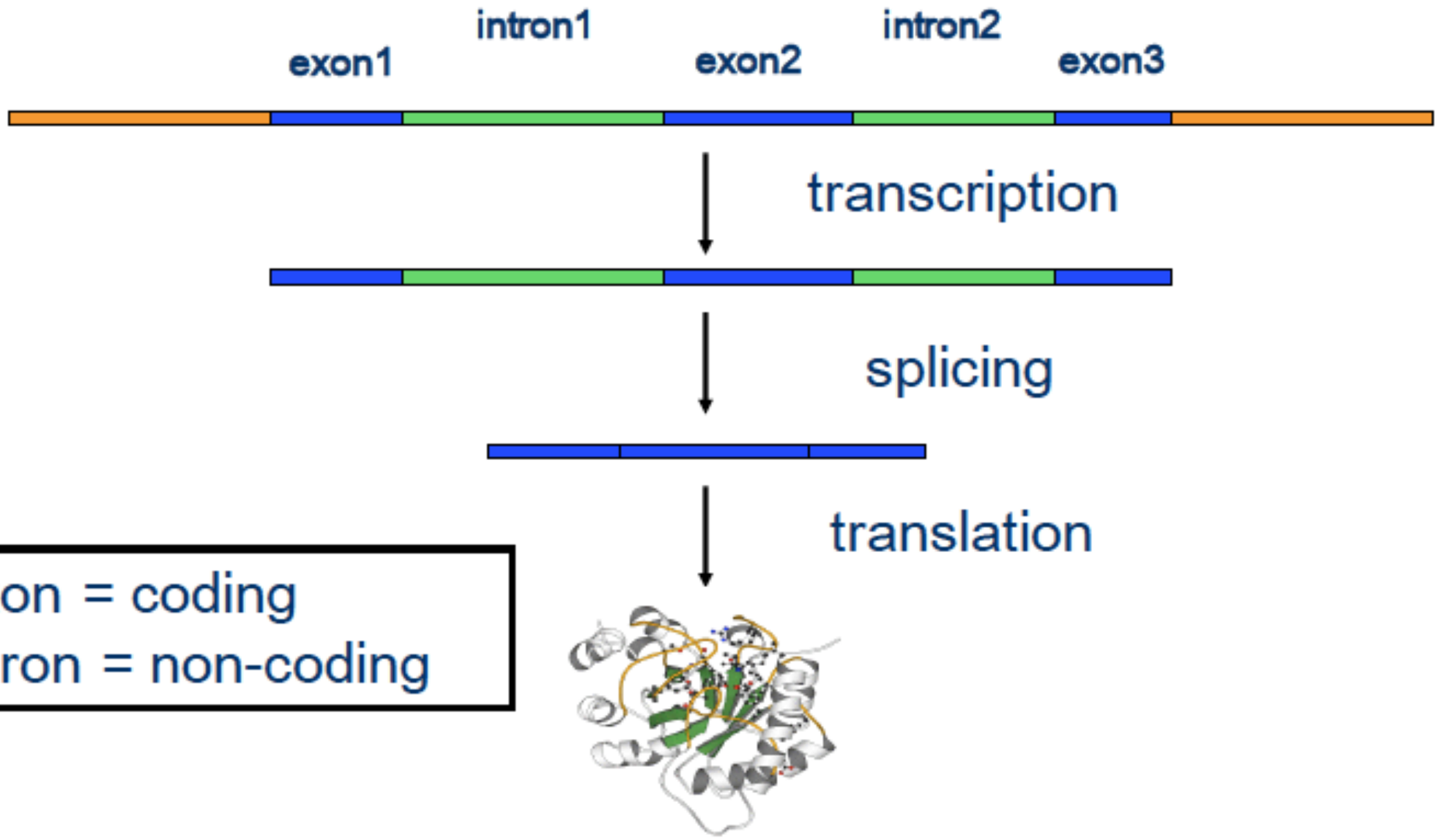


Gene finding: HMM for bacteria

GeneMarker's HMM model



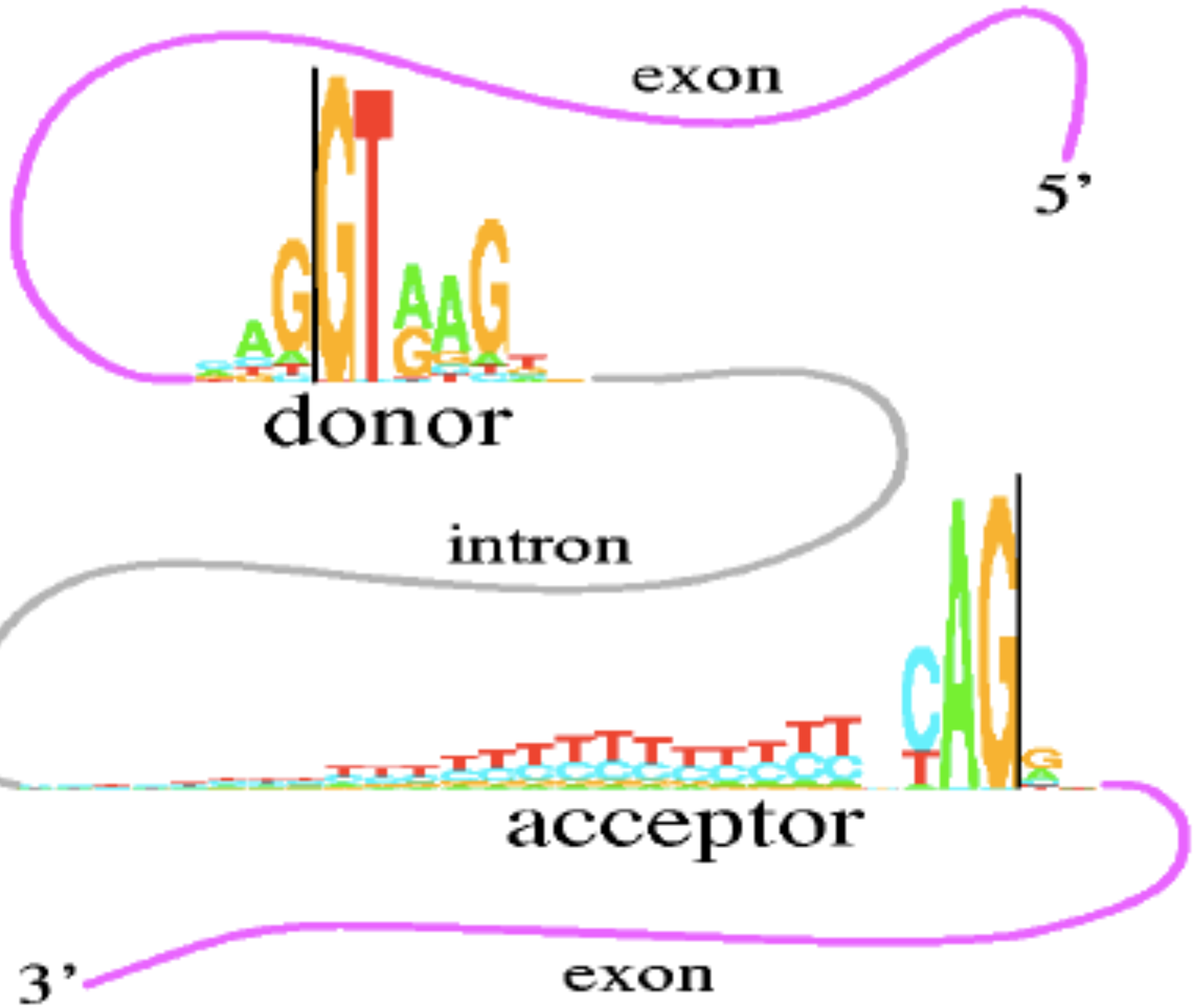
Gene finding: handling introns



exon = coding
intron = non-coding

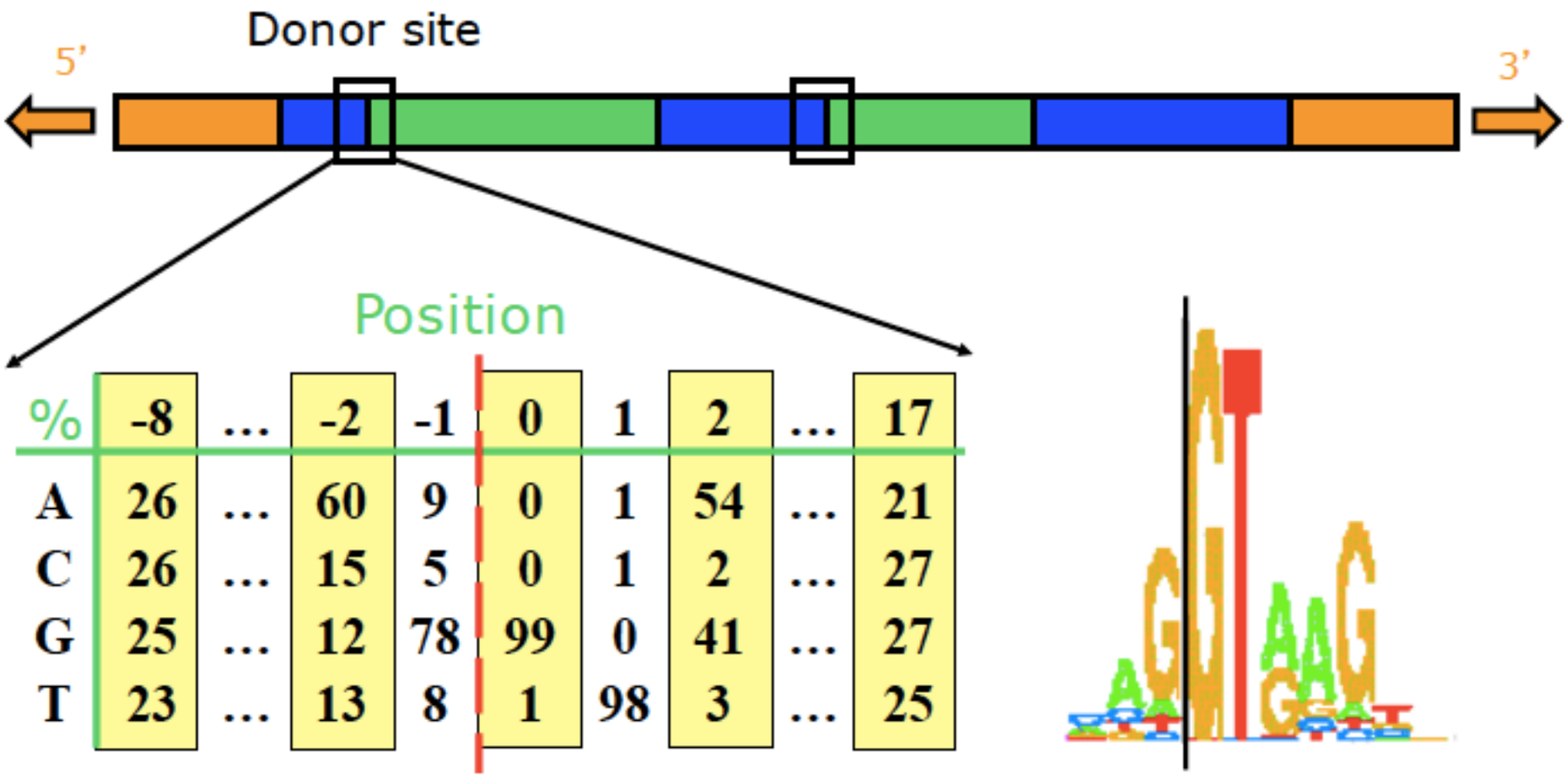
Gene finding: handling introns

Splicing site motifs



Gene finding: handling introns

Splicing site motifs



Gene finding: HMM version 2