1 Problem 11.2

The corresponding HMM model for this Problem is illustrated in Figure1. The states where the dealer is tossing the biased coin are represented by $B_1, \ldots, B_{10}$, the fair coin is represented by the states $F_1, \ldots, F_{10}$. From the start state we go into state B_1 or F_1 with equal probability. After step i, with $i \leq 10$ we end in state B_i if we went into B_1 after the start and we end in state F_i if we went into F_1 after the start. After the 10th toss we change from B_{10} to F_1 (if we went into B_1 from the start) or from F_{10} to B_1 (if we went into F_1 from the start) both with probability $\frac{1}{10}$ or we stay in the same state B_{10} or F_{10} with probability $\frac{9}{10}$. If we change into B_1 (F_1) we again traverse $B_2, B_3, \ldots, B_{10}$ ($F_2, F_3, \ldots, F_{10}$) before we can change into B_1 (F_1) again. By always traversing $B_1, B_2, B_3, \ldots, B_{10}$ ($F_1, F_2, F_3, \ldots, F_{10}$) before we can go into F_1 (B_1) and because we stay in B_{10} (F_{10}) with probability $\frac{9}{10}$ the case is represented that the dealer keeps each coin for at least 10 tosses.

For the decoding algorithm we now have to deal with 20 states per step (see Figure 2). Let $B_{i,k}$ and $F_{i,k}$ be the probability for the most probable sequence of states for emitting $x_1, \ldots, x_k$ and ending at state B_i or F_i. For any given state i with $i \leq 9$ and step k we can compute $B_{i,k}$ as $B_{i-1,k-1} * e_{B_i}(x_k)$ and likewise for $F_{i,k}$, because we can only come from one previous state. When $k \geq 20$ we can come from two previous states to get to $B_{10,k}$ and $F_{10,k}$ and thus have to calculate two values and chose a maximum: to get $B_{10,k}$ we have to get the maximum of $F_{1,k-1} * \frac{1}{10} * e_{B_{10}}(x_k)$ and $B_{10,k-1} * \frac{9}{10} * e_{B_{10}}(x_k)$. When $k < 20$ we can also only come from one previous state to get to $B_{10,k}$ and $F_{10,k}$.

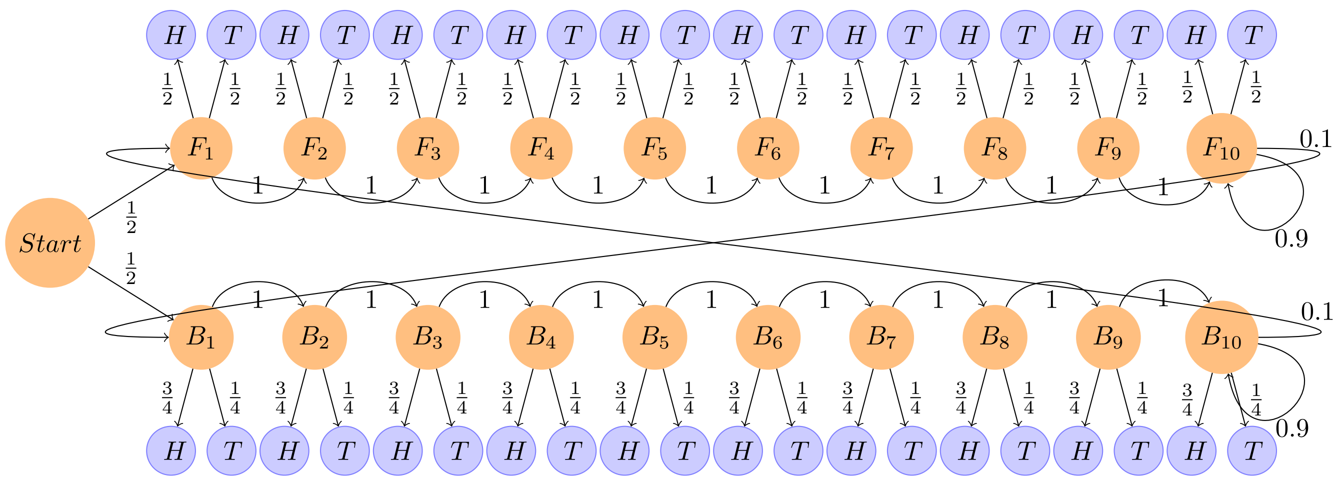

Figure 1: A HMM for the Fair Bet Casino where the dealer keeps every coin for at least ten tosses.

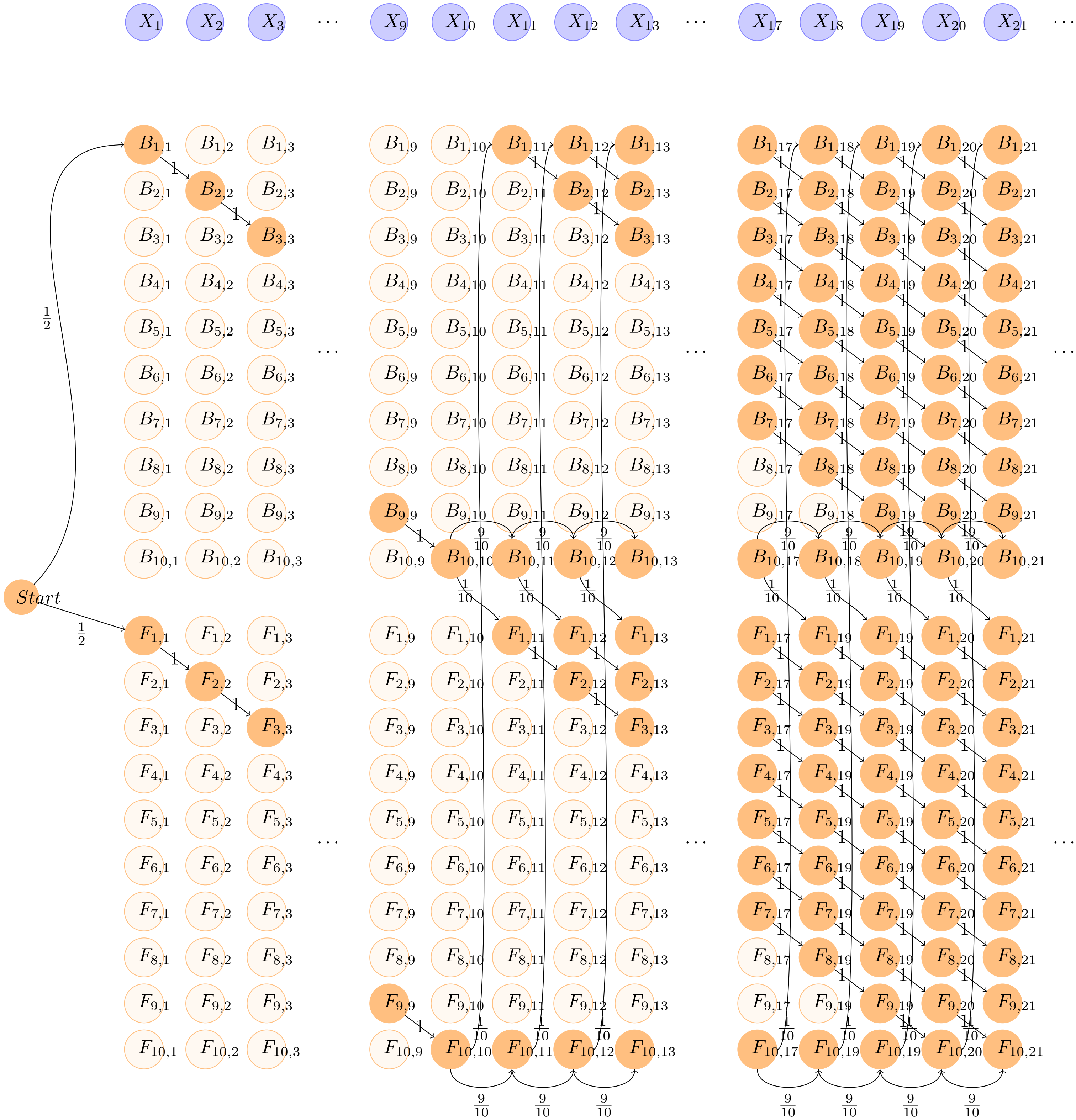

Figure 2: Directed Acyclic Graph for the possible sequence of states that emit the observation $x_1 x_2 x_3 \dots x_n$ of the Fair Bet Casino. The light drawn nodes $B_{i,k}$ and Fi, k indicate that we can not emit $x_1, \dots, x_k$ and reach state i.

2 Problem 11.4

Figure 3 illustrates running the decoding algorithm for this Problem. As the most probable sequence of states we get $\beta\beta\beta\beta$.

3 Problem 11.5

We compute the probabilities $P(x|fair\ coin)$ and $P(x|biased\ coin)$ using log probabilities:

- $P(x|fair\ coin) = 17 * log(\frac{1}{2}) =$ -17
- $P(x|biased\ coin) = 7 * log(\frac{1}{4}) + 10 * log(\frac{3}{4}) = -14 + 10 * (log(3) + log(\frac{1}{4}) = -14 + 10 * (1.585 - 2) = -14 + 15.85 - 20 = -18.15$

We get a higher probability for $P(x|fair\ coin)$ thus the sequence was more likely generated by the fair coin.

4 Problem 11.6

A HMM for this problem is given by: $\mathcal{M} = (\Sigma, Q, A, E)$ with

- $\Sigma = \{1, 2, 3\}$
- $Q = \{Begin, D_1, D_2, End\}$
- $A = \{a_{begin,D_1} = \frac{1}{2}, a_{Begin,D_2} = \frac{1}{2}, a_{D_1,D_1} = \frac{1}{2}, a_{D_1,D_2} = \frac{1}{4}, a_{D_1,D_{End}} = \frac{1}{4}, a_{D_2,D_2} = \frac{1}{2}, a_{D_2,D_1} = \frac{1}{4}, a_{D_2,D_{End}} = \frac{1}{4}\}$
- $E = \{e_{D_1}(1) = \frac{1}{2}, e_{D_1}(2) = \frac{1}{4}, e_{D_1}(3) = \frac{1}{4}, e_{D_2}(1) = \frac{1}{4}, e_{D_2}(2) = \frac{1}{2}, e_{D_2}(3) = \frac{1}{4}\}$

Figure 4 shows the graph of this HMM. We observed the sequence 112122end. We find a sequence of states which best explains this observation by running the decoding algorithm according to emission probabilites E and transition probabilities A. The computed values D_{xy} as log probabilites are listed in the following table:

	-	1	1	2	1	2	2	end
Start	0							
D_1		-2	$\max\begin{cases}-2-1-1\\-3-2-1\end{cases}$ $= -4$ coming from D_{11}	$\max\begin{cases}-4-1-2\\-6-2-2\end{cases}$ $= -7$ coming from D_{12}	$\max\begin{cases}-7-1-1\\-7-2-1\end{cases}$ $= -9$ coming from D_{13}	$\max\begin{cases}-9-1-2\\-10-2-2\end{cases}$ $= -12$ coming from D_{14}	$\max\begin{cases}-12-1-2\\-12-2-2\end{cases}$ $= -15$ coming from D_{15}	
D_2		-3	$\max\begin{cases}-3-1-2\\-2-2-2\end{cases}$ $= -6$ coming from D_{11}, D_{21}	$\max\begin{cases}-6-1-1\\-4-2-1\end{cases}$ $= -7$ coming from D_{12}	$\max\begin{cases}-7-1-2\\-7-2-2\end{cases}$ $= -10$ coming from D_{23}	$\max\begin{cases}-10-1-1\\-9-2-1\end{cases}$ $= -12$ coming from D_{24}, D_{14}	$\max\begin{cases}-12-1-1\\-12-2-1\end{cases}$ $= -14$ coming from D_{25}	
End								$\max\begin{cases}-15-2\\-14-2\end{cases}$ $= -16$ from D_{26}

G $\quad$ G $\quad$ C

α

$$-1 + log(\tfrac{2}{5})$$

β

$$-1 + log(\tfrac{1}{5})$$

α

$$\max \begin{cases} -1 + log(\frac{2}{5}) + log(\frac{9}{10}) + log(\frac{2}{5}) \\ -1 + log(\frac{1}{5}) + log(\frac{1}{10}) + log(\frac{2}{5}) \end{cases}$$
$$= -1 + log(\tfrac{2}{5}) + log(\tfrac{9}{10}) + log(\tfrac{2}{5})$$
$$= -1 + 2log(\tfrac{2}{5}) + log(\tfrac{9}{10})$$

β

$$\max \begin{cases} -1 + log(\frac{1}{5}) + log(\frac{9}{10}) + log(\frac{1}{5}) \\ -1 + log(\frac{2}{5}) + log(\frac{1}{10}) + log(\frac{1}{5}) \end{cases}$$
$$= -1 + log(\tfrac{1}{5}) + log(\tfrac{9}{10}) + log(\tfrac{1}{5})$$
$$= -1 + 2log(\tfrac{1}{5}) + log(\tfrac{9}{10})$$

α

$$\max \begin{cases} -1 + 2log(\frac{2}{5}) + log(\frac{9}{10}) + log(\frac{9}{10}) + log(\frac{1}{10}) \\ -1 + 2log(\frac{1}{5}) + log(\frac{9}{10}) + log(\frac{1}{10}) + log(\frac{1}{10}) \end{cases}$$
$$= -1 + 2log(\tfrac{2}{5}) + log(\tfrac{9}{10}) + log(\tfrac{9}{10}) + log(\tfrac{1}{10})$$
$$= -1 + 2log(\tfrac{2}{5}) + 2log(\tfrac{9}{10}) + log(\tfrac{1}{10})$$

β

$$\max \begin{cases} -1 + 2log(\frac{1}{5}) + log(\frac{9}{10}) + log(\frac{9}{10}) + log(\frac{3}{10}) \\ -1 + 2log(\frac{2}{5}) + log(\frac{9}{10}) + log(\frac{1}{10}) + log(\frac{3}{10}) \end{cases}$$
$$= -1 + 2log(\tfrac{1}{5}) + log(\tfrac{9}{10}) + log(\tfrac{9}{10}) + log(\tfrac{3}{10})$$
$$= -1 + 2log(\tfrac{1}{5}) + 2log(\tfrac{9}{10}) + log(\tfrac{3}{10})$$

T

α

$$\max \begin{cases} -1 + 2log(\frac{2}{5}) + 2log(\frac{9}{10}) + log(\frac{1}{10}) + log(\frac{9}{10}) + log(\frac{1}{10}) \\ -1 + 2log(\frac{1}{5}) + 2log(\frac{9}{10}) + log(\frac{3}{10}) + log(\frac{1}{10}) + log(\frac{1}{10}) \end{cases}$$
$$= -1 + 2log(\tfrac{2}{5}) + 2log(\tfrac{9}{10}) + log(\tfrac{1}{10}) + log(\tfrac{9}{10}) + log(\tfrac{1}{10})$$
$$= -1 + 2log(\tfrac{2}{5}) + 3log(\tfrac{9}{10}) + 2log(\tfrac{1}{10})$$

β

$$\max \begin{cases} -1 + 2log(\frac{1}{5}) + 2log(\frac{9}{10}) + log(\frac{3}{10}) + log(\frac{9}{10}) + log(\frac{3}{10}) \\ -1 + 2log(\frac{2}{5}) + 2log(\frac{9}{10}) + log(\frac{1}{10}) + log(\frac{1}{10}) + log(\frac{3}{10}) \end{cases}$$
$$= -1 + 2log(\tfrac{1}{5}) + 2log(\tfrac{9}{10}) + log(\tfrac{3}{10}) + log(\tfrac{9}{10}) + log(\tfrac{3}{10})$$
$$= -1 + 2log(\tfrac{1}{5}) + 3log(\tfrac{9}{10}) + 2log(\tfrac{3}{10})$$

Figure 3: decoding algorithm for the HMM of Problem 11.4. The upper max value always represents the α state and the lower max value represents the β state. After emitting the final T we get a higher probability in state β and therefore backtrack from there.

As the two most probable sequence of states by backtracking through the table (illustrated in Figure 5) we get:

- *Start*-D_1-D_1-D_1-D_1-D_2-D_2-*End*
- *Start*-D_1-D_1-D_2-D_2-D_2-D_2-*End*

These are the most likely sequences of states that emitted our observation. As a log probability in the final state we get -16 thus the probability of these paths, given the observation 112122end is $\frac{1}{2^{16}} = \frac{1}{65536}$

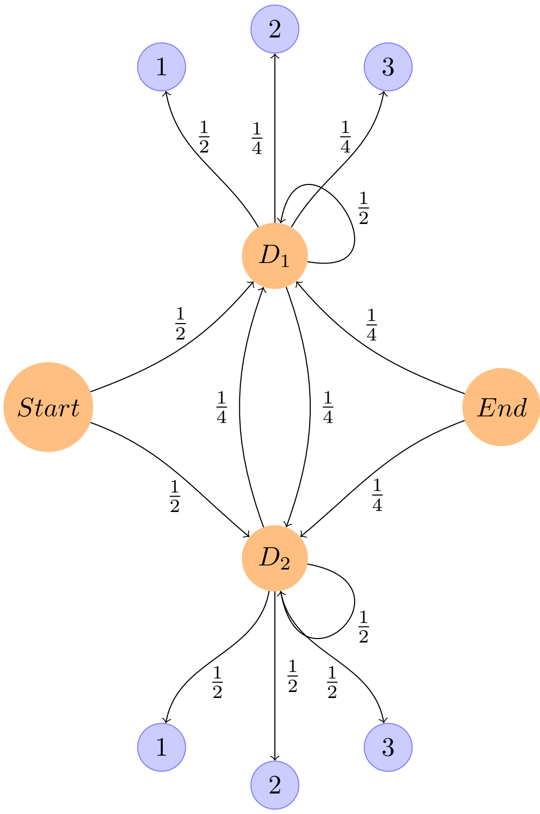

Figure 4: HMM for rolling the three sided dice of Problem 11.6

5 Problem 5

We are given the HMM $\mathcal{M}$, an observation sequence $X = x_1, x_2, x_3 \ldots, x_n$ and that observation x_i was emitted by state q. Let $s_{m,l}$ be the probability of the most likely path that emitted $x_1, x_2, \ldots, x_l$ and reached state m. We want to get the most likely state transition sequence $s_1, s_2, \ldots, s_n$ that $\mathcal{M}$ goes through to generate X where $s_i = q$. We get this sequence by first running the viterbi

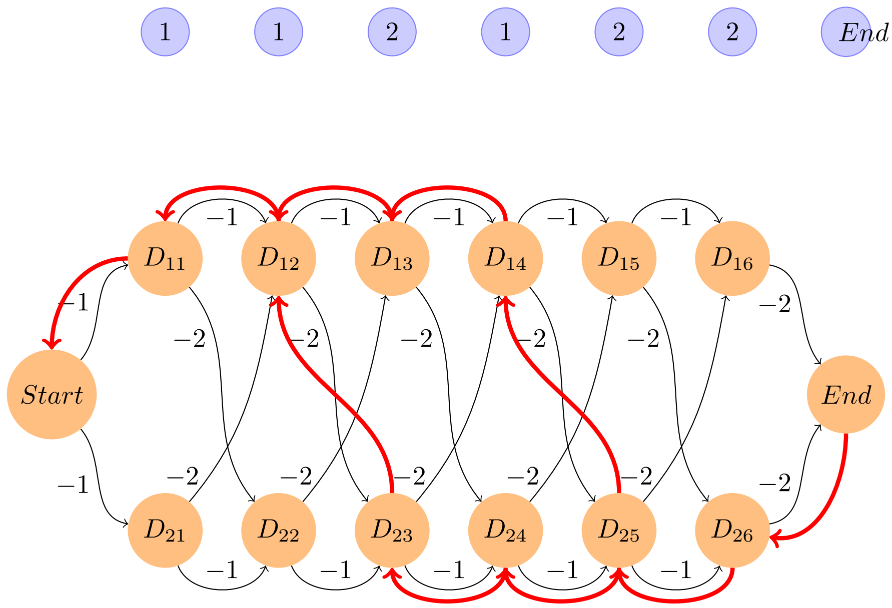

Figure 5: Running the decoding algorithm for the sequence of states 1121221end. D_{xy} is the probability of the most likely path of $x_1, \ldots, x_y$ that ends at state D_x. The values between D_{xi} and D_{yi+1} describe the log-probability of changing from state D_x to D_y from time step i to $i+1$. The red arrows indicate the backtracking paths.

algorithm until step i. Next we set the previously calculated probabilites for $s_{m,i}$ with $m \neq q$ to zero before calculating the next set of probabilites $s_{m,i+1}$. Then we continue with the viterbi algorithm to calculate the set of probabilities $s_{m,i+1}$. For $s_{m,i+1}$ we get the probability of the most likely path that emitted $x_1, x_2, \ldots, x_i, x_{i+1}$ where x_i was emitted by state k and we reached state m. Finally for $s_{m,n}$ we get the probability of the most likely path that emitted $(x_1, x_2, \ldots, x_i, \ldots, x_n)$ where x_i was emitted by state q and we reached state n. Now we do the usual backtracking to get the most likely state transition sequence. As a result we get the most likely state transition sequence R where $\mathcal{M}$ was in state q at step i.

6 Problem 6

We are given the HMM $\mathcal{M}$. To compute the probability that x_i is aligned with y_j, let:

- $v^M(a,b)$ be the probability of aligning $x_1, \ldots, x_a$ with $y_1, \ldots, y_b$ where x_a was aligned to y_b in the last step.
- $v^y(a,b)$ be the probability of aligning $x_1, \ldots, x_a$ with $y_1, \ldots, y_b$ where y_b was aligned to a gap in the last step.
- $v^x(a,b)$ be the probability of aligning $x_1, \ldots, x_a$ with $y_1, \ldots, y_b$ where x_a was aligned to a gap in the last step.

To compute the probability of aligning x_i with y_j I initialize the tables v^M, v^x and v^y as following:

- $v^M(0,0) = 1$ and all other $v^M(a,0) = v^M(0,b) = 0$
- all $v^y(0,b) = 0$

- all $v^x(a, 0) = 0$

and I define the following recurrence (where s=$(1 - 2\delta - \tau)$ is the transition probability of going from state M to state M, $d = (1 - \epsilon - \tau)$ is the transition probability of going from state X to state M and for going from state Y to state M, ϵ is the transition probability of going from X to X or from Y to Y and δ is the transition probability of going from M to X or to Y:

for $i = 0, \ldots, n$

for $j = 0, \ldots, m$

$v^M(i, j) = q_{x_i,y_j} * (v^M(i-1, j-1) * s + v^y(i-1, j-1) * d + v^x(i-1, j-1) * d)$

$v^y(i, j) = q_{y_j} * (v^M(i-1, j) * \delta + v^y(i-1, j) * \epsilon)$

$v^x(i, j) = q_{x_i} * (v^M(i, j-1) * \delta + v^y(i, j-1) * \epsilon)$

After runing the algorithm the cell $v^M(i, j)$ contains the probability that x_i is aligned with y_j which is what we are looking for.