Treball final de grau

GRAU DE MATEMÀTIQUES

Facultat de matemàtiques

Universitat de Barcelona

CADENES DE MARKOV I EXPERIMENT D'UNZIPPING

Laia Montraveta Jiménez

Director: David Márquez Carreras
Realitzat a: Departament de Probabilitat,
Lògica i Estadística. UB
Barcelona, 30 de Juny de 2015

Agraïments

M'agradaria expressar el meu agraïment al meu tutor, David Márquez, tant per haver acceptat aquest treball com pel recolzament que he rebut per part seva durant la seva realització. També vull donar les gràcies al Fèlix Ritort per haver-me fet descobrir el món de la biofísica i haver-me guiat en la part aplicada. Al Joan Camuñas per facilitar-me les dades i dedicar-me tot el temps que he necessitat. I finalment un agraïment especial a la meva parella, familiars i amics que han confiat en mi i sempre han estat al meu costat.

Índex

Introduction

Markov models are stochastic processes without memory: processes in which the next state of the system depends only on its immediately previous state and not on the whole chain of states. Although it may appear to be a very simple model, it is widely seen in real life and used in a variety of fields like biology, physics, engineering, medicine or even social sciences. If we have physically unobservable states and the only thing we can know are probabilistic functions depending on them, we can treat the system with an extension of Markov models, hidden Markov models (HMM). Hidden Markov models have been widely used for speech recognition and in computational molecular biology, among others.

Driven by my curiosity for that type of models and my interest in biophyisics I decided to dedicate this undergraduate thesis to the study of different variations of Markov models and to see how one can apply them to a specific experiment of molecular biophysics, the DNA unzipping.

As we will see later, the unzipping experiment consists in pulling a double stranded DNA molecule from each end so the bonds between them are broken. Plotting force versus distance curve we obtain a very characteristic sawtooth pattern that can be used, for example, to find the specific places where proteins and enzymes are fixed to the DNA. In these experiments we find cooperative unzipping-zipping regions, in other words, zones where several base-pairs of different length are involved in the transition, behaving like an all or nothing. Our goal is to determine the distribution of DNA unzipping to find how many base-pairs are opened in each step. To do that we treat the system as a variable-stepsize hidden Markov model, a kind of HMM adapted in order to describe at the same time the molecular state and the position of a processive molecular motor.

This dissertation has two parts, one theoretical and the other applied. The first one, which includes chapters 1, 2 *and* 3, *is an introduction to homogeneous Markov models and hidden Markov models. In the last chapter we present the unzipping experiment and apply the algorithms seen in previous chapters in order to determine the DNA's unzipping pattern.*

Introducció

Els models de Markov són processos estocàstics sense memòria, processos en què l'estat on seràs en el pas següent només depèn de l'estat on ets ara i no de com has arribat fins aquí. Encara que sembli un model molt senzill és molt present a la vida real i s'utilitza en camps tan diversos com la biologia, la física, l'enginyeria, la medicina o fins i tot en les ciències socials. Si tenim estats que no són observables físicament i l'únic que en podem mesurar són funcions probabilístiques podem tractar-los amb una extensió de les cadenes de Markov, les cadenes de Markov ocultes (HMM). Les cadenes de Markov ocultes també compten amb un llarg nombre d'aplicacions com per exemple en la biologia molecular computacional o el reconeixement de veu.

Moguda per la curiositat que em generaven aquest tipus de models i l'interès que em desperta la biofísica vaig decidir dedicar aquest treball de final de grau a l'estudi de diferents tipus de cadenes de Markov i veure'n la seva aplicació a un experiment concret de biofísica molecular, l'experiment d'*unzipping*.

Com explicarem més endavant, l'experiment d'*unzipping* consisteix en la tracció d'una cadena doble d'ADN per cadascun dels extrems de manera que es van trencant els enllaços que mantenen ambdues cadenes unides. Si representem la força en funció de la distància entre els extrems de la molècula obtenim un patró en dent de serra molt característic. Aquest patró pot ser utilitzat, per exemple, per determinar el lloc específic on s'han unit proteïnes o enzims i conèixer-ne la selectivitat per algunes regions. A l'hora de trencar els enllaços que mantenen unides les bases complementaries trobem zones que actuen com a *tot o res*, uns quants parells de bases que se separen i s'ajunten de forma grupal sense fer-ho mai per separat. El nostre objectiu és trobar quin patró segueix aquest desplegament, quants parells de bases es separen a cada pas i per fer-ho utilitzarem una variació de les HMM, les *variable-stepsize hidden Markov models* (VS-HMM), que va ser pensat per tal de determinar alhora la posició i l'estat dels motors moleculars.

El treball està estructurat en dues parts, una teòrica i l'altra aplicada. La primera, que comprèn els capítols 1, 2 i 3, és una introducció a les cadenes de Markov homogènies i a les cadenes de Markov ocultes. En l'últim capítol presentarem l'experiment d'*unzipping* i aplicarem els algorismes introduïts als capítols anteriors per tal de trobar el patró de desplegament que segueix l'ADN en aquest experiment.

Capítol 1

Conceptes bàsics

Seguint l'estructura de [2], comencem definint els conceptes bàsics que utilitzarem al llarg dels següents capítols, fent un repàs de la teoria de probabilitats.

Definició 1.0.1. σ-àlgebra
Sigui Ω un conjunt i $\mathcal{A} \subset \mathcal{P}(\Omega)$. Es diu que $\mathcal{A}$ és una *σ-àlgebra* de $\mathcal{P}(\Omega)$ si satisfà:

1. $\Omega \in \mathcal{A}$.
2. $A \in \mathcal{A} \longrightarrow A^c \in \mathcal{A}$, és estable per pas al complementari.
3. Si $\{A_n : n \geq 1\}$ és una família numerable d'elements d'$\mathcal{A}$, aleshores $\bigcup_{n\geq 1} A_n \in \mathcal{A}$.

Definició 1.0.2. Espai mesurable
Sigui Ω un conjunt i $\mathcal{F}$ una σ-àlgebra sobre $\mathcal{P}(\Omega)$. Aleshores el conjunt $(\Omega, \mathcal{F})$ és un *espai mesurable*.

Definició 1.0.3. Espai de probabilitat
L'*espai de probabilitat* és una terna $(\Omega, \mathcal{A}, P)$ on:

1. Ω és l'*espai mostral*, un conjunt que conté els resultats possibles: $\omega \in \Omega$. Quan el conjunt Ω és finit, aleshores $(\Omega, \mathcal{A}, P)$ s'anomena espai de probabilitat finit.
2. $\mathcal{A} \subset \mathcal{P}(\Omega)$ és una σ-àlgebra que ens permet descriure tots els esdeveniments possibles.
3. P és la *probabilitat*, una aplicació $P : \mathcal{A} \longrightarrow [0, 1]$ tal que:
   - $P(\Omega) = 1$.
   - Si $\{A_n : n \geq 1\}$ és una successió de conjunts d'$\mathcal{A}$ disjunts dos a dos, aleshores $P\left(\bigcup_{n=1}^{\infty} A_n\right) = \sum_{n=1}^{\infty} P(A_n)$.

Definició 1.0.4. Esdeveniments independents
Dos esdeveniments A i B són *independents* si $P(A \cap B) = P(A)P(B)$.

Sovint se substitueix $\cap$ per una coma, és a dir, $P(A \cap B) = P(A, B)$ que és la probabilitat que ambdós esdeveniments succeeixin.

Definició 1.0.5. Famílies independents
Una família d'esdeveniments, $\{A_i \colon i \in I\}$, s'anomena *independent* si per tota col·lecció finita $A_{i_1}, \ldots, A_{i_k}$ de conjunts diferents,

$$P\left(\bigcap_{j=1}^{k} A_{i_j}\right) = \prod_{j=1}^{k} P(A_{i_j}).$$

Definició 1.0.6. Probabilitat condicionada
La *probabilitat condicionada* de l'esdeveniment A donat l'esdeveniment B representa la probabilitat que succeeixi A sabent que ha succeït B. Es defineix, únicament quan $P(B) \neq 0$, com:

$$P(A \mid B) := \frac{P(A \cap B)}{P(B)}.$$

Simètricament obtenim $P(A \cap B) = P(A \mid B)P(B) = P(B \mid A)P(A)$.

Teorema 1.0.7. *De les probabilitats totals*
Sigui $\{B_i \colon i \in I\}$ una partició d' Ω tal que $P(B_i) > 0$ per tot $i \in I$ aleshores per tot $A \in \mathcal{A}$,

$$P(A) = \sum_{i \in I} P(A \mid B_i)P(B_i).$$

Teorema 1.0.8. *Regla de Bayes*
Sigui $\{B_i \colon i \in I\}$ una partició d' Ω tal que $P(B_i) > 0$ per tot $i \in I$ i sigui $A \in \mathcal{A}$ tal que $P(A) > 0$, aleshores:

$$P(B_j \mid A) = \frac{P(A \mid B_j)P(B_j)}{\sum_{i \in I} P(A \mid B_i)P(B_i)}.$$

Teorema 1.0.9. *De les probabilitats compostes*
Siguin $A_1, A_2, \cdots, A_n$ esdeveniments amb $P(A_1 \cap A_2 \cap \cdots \cap A_{n-1}) > 0$, aleshores

$$P(A_1 \cap A_2 \cap \cdots \cap A_n) = P(A_1)P(A_2|A_1) \cdots P(A_n|A_1 \cap A_2 \cap \cdots \cap A_{n-1}).$$

Definició 1.0.10. Variable aleatòria
Sigui $(\Omega, \mathcal{A}, P)$ un espai de probabilitat a Ω una aplicació $X \colon \Omega \longrightarrow \mathbb{R}$ és una *variable aleatòria* si per tot $x \in \mathbb{R}$ l'esdeveniment $\{X \leq x\} = \{\omega \colon X(\omega) \leq x\}$ té una probabilitat assignada, és a dir, $\{X \leq x\} \in \mathcal{A}$.
Si tenim el cas on l'aplicació va a E, un conjunt numerable, aleshores $X \colon \Omega \longrightarrow E$ és una *variable aleatòria discreta* si per tot $a \in E$ es compleix que $\{X = a\} \in \mathcal{A}$.

Definició 1.0.11. Esperança
Donada una variable aleatòria X amb funció de densitat $f(x) = P(X \leq x)$ i $g\colon \mathbb{R} \longrightarrow \mathbb{R}$ tal que $\int_\Omega |g(x)|f(x)dx < \infty$, es defineix l'*esperança* de $g(X)$ com:

$$E[g(X)] = \int_\Omega g(x)f(x)dx.$$

Pel cas d'una variable aleatòria discreta, l'esperança es defineix per $g\colon E \longrightarrow \mathbb{R}$ quan $\sum_{a\in E} |g(a)|P(X = a) < \infty$ mitjançant la fórmula:

$$E[g(X)] = \sum_{a\in E} g(a)P(X = a).$$

Definició 1.0.12. Esperança condicionada Siguin X, Y dues variables aleatòries discretes prenent valors a F i G respectivament. Sigui $g\colon F \times G \longrightarrow \mathbb{R}$ una funció no negativa. Aleshores per tot $y \in G$ es defineix l'*esperança condicional de* $g(X, Y)$ *donat* $Y = y$ com:

$$E[g(X, Y) \mid Y = y] = \sum_{x\in F} g(x, y)P(X = x \mid Y = y).$$

Capítol 2

Cadenes de Markov

Tal com hem vist a la introducció, el procés estocàstic de Markov es caracteritza per no tenir memòria i es present en camps molt diversos, ens permet descriure processos tant diferents com el moviment brownià o la teoria de cues [2, 5].

En aquest capítol explicarem què és una cadena de Markov homogènia i en veurem algunes propietats.

2.1 Cadena de Markov homogènia

Comencem definint un parell de termes que ens seran útils més endavant: procés estocàstic i matriu estocàstica.

Definició 2.1.1. Procés estocàstic
Un *procés estocàstic* és una col·lecció de variables aleatòries indexades:

$$X_t \colon \Omega \longrightarrow E$$

L'índex $t \in T$ normalment s'identifica amb el temps. Parlem de procés de temps discret quan $T \subseteq \mathbb{N}$ i de procés de temps continu quan $T \subseteq [0, \infty)$.
D'altra banda, els valors que prenen les variables aleatòries s'anomenen estats donant nom a E que s'anomena *conjunt d'estats*.
Així doncs, quan $X_t = i$, $i \in E$, es diu que el procés està a l'estat i en l'instant t.

Cada variable aleatòria té la seva pròpia funció de probabilitats π_t, és a dir, $\pi_t(i) = P(X_t = i)$ per tot $t \in T$, per tot $i \in E$.

A partir d'ara ens centrarem en processos de temps discret i per això emprarem, la majoria de vegades, el subíndex n enlloc de t.

Definició 2.1.2. Matriu estocàstica
Una matriu, $\Pi = (p_{ij})_{i,j \in E}$, on E és un conjunt finit o numerable, és una *matriu estocàstica* si satisfà les dues propietats següents:

- $p_{ij} \in [0, 1]$.
- $\sum_{j \in E} p_{ij} = 1$.

És a dir, cada fila $(p_{ij})_{j\in E}$ és una probabilitat. Fixem-nos que E pot ser infinit i, conseqüentment, la matriu també pot tenir dimensió infinita.

Ara ja tenim tot el necessari per definir el concepte central del capítol: *cadena de Markov*.

Definició 2.1.3. Cadena de Markov
Un procés estocàstic $\{X_n\colon n \geq 0\}$ que pren valors en un conjunt d'estats E és una *cadena de Markov* si per tot $i_0, \ldots, i_{n+1} \in E, n \geq 0$, compleix:

$$P(X_{n+1} = i_{n+1} \mid X_n = i_n, \ldots, X_0 = i_0) = P(X_{n+1} = i_{n+1} \mid X_n = i_n). \qquad (2.1.1)$$

Si, a més, $P(X_{n+1} = i_{n+1} \mid X_n = i_n)$ no depèn d'n aleshores s'anomena *cadena de Markov homogènia* (CMH).

La matriu estocàstica associada a la cadena: $\Pi = (p_{ij})_{i,j\in E}$ s'anomena *matriu de transició*. Les probabilitats de transició, $p_{ij} = P(X_{n+1} = j \mid X_n = i)$, s'acostumen a representar mitjançant grafs dirigits [10]. Els vèrtex s'identifiquen amb els estats i les fletxes, arestes dirigides, representen les transicions possibles entre aquests estats tal com mostra la figura (2.1.1).

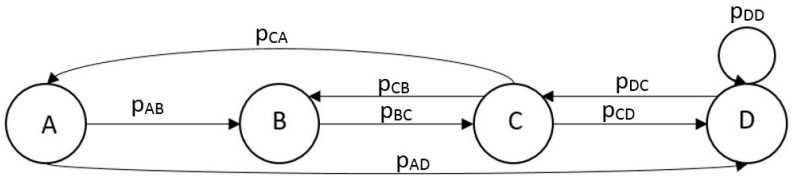

Figura 2.1.1: Representació d'una cadena de Markov amb quatre estats: A, B, C i D. Únicament es representen les transicions amb probabilitat més gran que 0, prescindint directament de la resta de fletxes

Observació 2.1.4. La propietat definida per l'equació (2.1.1) s'anomena *propietat de Markov*. Ens diu que per tot $n \geq 0$, X_{n+1} té una distribució de probabilitats determinada per les $p_{i_n j}$, on $i_n, j \in E$ i que aquesta només depèn d'X_n. En altres paraules, la propietat de Markov ens diu que en qualsevol instant de temps per predir el comportament del sistema en el futur només cal que considerem el present, només importa l'estat on som i no com hi hem arribat.

Una qüestió interessant seria saber quins paràmetres necessitem conèixer perquè la distribució de tota la cadena de Markov quedi determinada, què la identifica. El següent teorema ens serà molt útil.

Teorema 2.1.5. *Sigui $\{X_n\colon n \geq 0\}$ un procés estocàstic que pren valors en E, aleshores és una cadena de Markov homogènia amb distribució de probabilitat inicial γ i matriu de transició Π si, i només si, per tot $i_0, \ldots, i_n \in E$ i per tot $n \geq 0$,*

$$P(X_0 = i_0, X_1 = i_1, \ldots, X_{n-1} = i_{n-1}, X_n = i_n) = \gamma_{i_0} p_{i_0 i_1} p_{i_1 i_2} \cdots p_{i_{n-1} i_n}.$$

Prova: Comencem veient la implicació d'esquerra a dreta. Aplicant primer el teorema de les probabilitats compostes (teorema (1.0.9)) i utilitzant a continuació la propietat de Markov (equació (2.1.1)):

$$\begin{aligned}
&P(X_0 = i_0, X_1 = i_1, \ldots, X_{n-1} = i_{n-1}, X_n = i_n) \\
&\quad = P(X_0 = i_0)P(X_1 = i_1 \mid X_0 = i_0)P(X_2 = i_2 \mid X_1 = i_1, X_0 = i_0)\times \\
&\qquad \times P(X_n = i_n \mid X_{n-1} = i_{n-1}, \ldots, X_0 = i_0) \\
&\quad = P(X_0 = i_0)P(X_1 = i_1 \mid X_0 = i_0)P(X_2 = i_2 \mid X_1 = i_1)\times \\
&\qquad \times P(X_n = i_n \mid X_{n-1} = i_{n-1}) \\
&\quad = \gamma_{i_0} p_{i_0 i_1} p_{i_1 i_2} \cdots p_{i_{n-1} i_n}.
\end{aligned}$$

D'on recuperem l'equació de l'enunciat:

$$P(X_0 = i_0, X_1 = i_1, \ldots, X_{n-1} = i_{n-1}, X_n = i_n) = \gamma_{i_0} p_{i_0 i_1} p_{i_1 i_2} \cdots p_{i_{n-1} i_n}.$$

A continuació passem a comprovar l'altra implicació, de dreta a esquerra:
Prenent el cas particular $n = 0$ tenim que per tot $i \in E$, $P(X_0 = i) = \gamma_i$ i, per tant, que la distribució d'X_0 és γ.
Finalment cal veure que també compleix la propietat de Markov i la homogeneïtat.

Per una banda tenim, fixant-nos en la definició de probabilitat condicionada:

$$\begin{aligned}
&P(X_n = i_n \mid X_0 = i_0, X_1 = i_1, \ldots, X_{n-1} = i_{n-1}) \\
&\qquad = \frac{P(X_0 = i_0, X_1 = i_1, \ldots, X_{n-1} = i_{n-1}, X_n = i_n)}{P(X_0 = i_0, X_1 = i_1, \ldots, X_{n-1} = i_{n-1})} \\
&\qquad = \frac{\gamma_{i_0} p_{i_0 i_1} p_{i_1 i_2} \cdots p_{i_{n-1} i_n}}{\gamma_{i_0} p_{i_0 i_1} p_{i_1 i_2} \cdots p_{i_{n-2} i_{n-1}}} = p_{i_{n-1} i_n}.
\end{aligned}$$

I d'altra banda, utilitzant el teorema de les probabilitats totals:

$$\begin{aligned}
P(X_{n-1} = i_{n-1}) &= \sum_{i_0, \ldots, i_{n-2} \in E^{n-1}} P(X_{n-1} = i_{n-1} \mid X_0 = i_0, \ldots, X_{n-2} = i_{n-2})\times \\
&\qquad \times P(X_0 = i_0, \ldots, X_{n-2} = i_{n-2}) \\
&= \sum_{i_0, \ldots, i_{n-2} \in E^{n-1}} P(X_0 = i_0,\, X_1 = i_1, \ldots, X_{n-1} = i_{n-1}) \\
&= \sum_{i_0, \ldots, i_{n-2} \in E^{n-1}} \gamma_{i_0} p_{i_0 i_1} p_{i_1 i_2} \cdots p_{i_{n-2} i_{n-1}}.
\end{aligned}$$

De manera similar:

$$\begin{aligned}
&P(X_{n-1} = i_{n-1}, X_n = i_n) = \\
&\quad = \sum_{i_0, \ldots, i_{n-2} \in E^{n-1}} P(X_0 = i_0,\, X_1 = i_1, \ldots, X_{n-1} = i_{n-1},\, X_n = i_n) \\
&\quad = \sum_{i_0, \ldots, i_{n-2} \in E^{n-1}} \gamma_{i_0} p_{i_0 i_1} p_{i_1 i_2} \cdots p_{i_{n-1} i_n} \\
&\quad = p_{i_{n-1} i_n} \sum_{i_0, \ldots, i_{n-2} \in E^{n-1}} \gamma_{i_0} p_{i_0 i_1} p_{i_1 i_2} \cdots p_{i_{n-2} i_{n-1}}.
\end{aligned}$$

Arribant a:

$$P(X_n = i_n | X_{n-1} = i_{n-1}) = \frac{P(X_n = i_n, X_{n-1} = i_{n-1})}{P(X_{n-1} = i_{n-1})} = p_{i_{n-1}i_n}.$$

I per tant podem concloure que efectivament:

$$P(X_n = i_n \mid X_0 = i_0, \ldots, X_{n-1} = i_{n-1}) = P(X_n = i_n | X_{n-1} = i_{n-1}) = p_{i_{n-1}i_n}.$$

□

Notació. Acabem de veure que una cadena de Markov homogènia queda totalment determinada per γ i Π. La cadena de Markov homogènia amb matriu de transició Π i distribució de probabilitats inicial γ, $\gamma(i) = P(X_0 = i)$, es denota CMH(γ, Π).

Observació 2.1.6. Sigui $\{X_n \colon n \geq 0\}$ una CMH(γ, Π) aleshores $\{X_{n+m} \colon m \geq 0\}$ és una CMH($\mathscr{L}(X_m)$, Π), on $\mathscr{L}(X_m)$ és la funció de probabilitat que segueix la variable aleatòria X_m.

Prova: Utilitzant el teorema (1.0.7) i aplicant el teorema (2.1.5) a $CMH(\gamma, \Pi)$ tenim que per tot $n + m \geq 0$,

$$\begin{aligned}
&P(X_{0+m} = j_0, \ldots, X_{n+m} = j_n) \\
&\quad = \sum_{i_0,\ldots,i_{m-1} \in E^m} P(X_0 = i_0, \ldots, X_{m-1} = i_{m-1}, X_m = j_0, \ldots, X_{m+n} = j_n) \\
&\quad = \sum_{i_0,\ldots,i_{m-1} \in E^m} \gamma_{i_0} p_{i_0 i_1} \cdots p_{i_{m-1} j_0} p_{j_0 j_1} \cdots p_{j_{n-1} j_n} \\
&\quad = p_{j_0 j_1} p_{j_1 j_2} \cdots p_{j_{n-1} j_n} \sum_{i_0,\ldots,i_{m-1} \in E^m} \gamma_{i_0} p_{i_0 i_1} \cdots p_{i_{m-1} j_0} \\
&\quad = P(X_m = j_0) p_{j_0 j_1} p_{j_1 j_2} \cdots p_{j_{n-1} j_n}.
\end{aligned}$$

□

I encara veurem una altra manera d'obtenir una CMH a partir d'un procés estocàstic concret.

Proposició 2.1.7. *Sigui $\{Z_n \colon n \geq 1\}$ una successió de variables aleatòries independents idènticament distribuïdes, $Z_n \colon \Omega \longrightarrow E$ i $f \colon E \times E \longrightarrow E$. Sigui X_0 una variable aleatòria a valors en E independent de $\{Z_n \colon n \geq 1\}$, aleshores definint $X_{n+1} = f(X_n, Z_{n+1})$ per $n \geq 0$ tenim que $\{X_n \colon n \geq 0\}$ és una CMH.*

Prova: D'una banda:

$$\begin{aligned}
P(X_{n+1} = j \mid X_n = i) &= \frac{P(f(X_n, Z_{n+1}) = j, X_n = i)}{P(X_n = i)} \\
&= \frac{P(f(i, Z_{n+1} = j), X_n = i)}{P(X_n = i)} \\
&= \frac{P(f(i, Z_{n+1}) = j) P(X_n = i)}{P(X_n = i)} \\
&= P(f(i, Z_{n+1}) = j) = p_{ij}.
\end{aligned}$$

Fixem-nos que l'esdeveniment $\{X_0 = i_0, \ldots, X_{n-1} = i_{n-1}, X_n = i_n\}$ es pot expressar en termes de $X_0, Z_1, \ldots, Z_n$ i, per tant, és independent de Z_{n+1}. I d'altra banda:

$$\begin{aligned}
P(X_{n+1} = j \mid X_0 = i_0, \ldots, X_{n-1} = i_{n-1}, X_n = i) &= \\
&= \frac{P(X_0 = i_0, \ldots, X_{n-1} = i_{n-1}, X_n = i, X_{n+1} = j)}{P(X_0 = i_0, \ldots, X_n = i)} \\
&= \frac{P\left(X_0 = i_0, f(i_0, Z_1) = i_1, \ldots, f(i_{n-1}, Z_n) = i, f(i, Z_{n+1}) = j\right)}{P(X_0 = i_0, \ldots, f(i_{n-1}, Z_n) = i)} \\
&= \frac{P(f(i, Z_{n+1}) = j) P(X_0 = i_0, \ldots, f(i_{n-1}, Z_n) = i)}{P(X_0 = i_0, \ldots, f(i_{n-1}, Z_n) = i)} \\
&= P(f(i, Z_{n+1}) = j) = p_{ij}.
\end{aligned}$$

Per tant, tenim una cadena de Markov i és homogènia perquè la probabilitat que hem obtingut no depèn d'n, la podem expressar com $p_{ij} = P(f(i, Z_1) = j)$. □

2.2 Probabilitat de transició en m etapes: Eq. de Chapman-Kolmogorov

Donada una cadena de Markov homogènia $\{X_n \colon n \geq 0\}$ i sabent que es troba a l'estat i, quina és la probabilitat que m passos després es trobi a l'estat j?

Per tal de respondre aquesta pregunta comencem definint la probabilitat que estem buscant com:

$$p_{ij}^{(m)} := P(X_{n+m} = j \mid X_n = i) \text{ on } n, m \geq 0, i, j \in E.$$

Fixem-nos que el cas $m = 1$ és amb el que hem estat treballant tota l'estona: $p_{ij}^{(1)} = p_{ij}$. Per m més grans tenim la següent proposició.

Proposició 2.2.1. *Podem obtenir les probabilitats m-èsimes $p_{ij}^{(m)}$ amb $m \geq 2$ recursivament, de la següent manera:*

$$p_{ij}^{(m)} = \sum_{k \in E} p_{ik}^{(m-1)} p_{kj}.$$

Demostració. Si utilitzem la definició de probabilitat condicionada i el teorema de les probabilitats totals tenim:

$$\begin{aligned}
p_{ij}^{(m)} = P(X_{n+m} = j \mid X_n = i) &= \frac{P(X_{n+m} = j, X_n = i)}{P(X_n = i)} \\
&= \frac{\sum\limits_{k \in E} P(X_{n+m} = j, X_{n+m-1} = k, X_n = i)}{P(X_n = i)} \\
&= \sum_{k \in E} \frac{P(X_{n+m} = j, X_{n+m-1} = k, X_n = i)}{P(X_{n+m-1} = k, X_n = i)} \cdot \frac{P(X_{n+m-1} = k, X_n = i)}{P(X_n = i)}.
\end{aligned}$$

De manera que:

$$\begin{aligned} p_{ij}^{(m)} &= \sum_{k\in E} P(X_{n+m}=j \mid X_{n+m-1}=k, X_n=i)P(X_{n+m-1}=k \mid X_n=i) \\ &= \sum_{k\in E} p_{kj}p_{ik}^{(m-1)}. \end{aligned}$$

□

Definint la matriu $\Pi_m := \left(p_{ij}^{(m)}\right)_{i,j\in E}$, tenim que

$$\Pi_m = \Pi^m = \Pi \overset{(m)}{\cdots} \Pi.$$

Amb aquesta definició i la proposició (2.2.1) és fàcil obtenir les següents propietats:

1. Π_m és una matriu estocàstica.
2. **Equació de Chapman-Kolmogorov**:

$$p_{ij}^{(l+k)} = \sum_{k\in E} p_{ik}^{(l)} p_{kj}^{(k)},$$

   on hem utilitzat la propietat del producte de matrius: $\Pi^{l+1} = \Pi^l \, \Pi^k$.

3. Repetint l'equació anterior obtenim:

$$p_{ij}^{(m)} = \sum_{i_1\in E} p_{ii_1}p_{i_1j}^{(m)} = \sum_{i_1,\ldots,i_{m-1}\in E^{m-1}} p_{ii_1}p_{i_1i_2}\cdots p_{i_{m-1}j}.$$

I utilitzant això som capaços de determinar la llei del procés $\{X_n \colon n \geq 0\}$, una $CMH(\gamma, \Pi)$. Per fer-ho, utilitzarem la notació: $\gamma^{(n)} = \gamma \, \Pi^n$.

Proposició 2.2.2. *Sigui $\{X_n \colon n \geq 0\}$ una $CMH(\gamma, \Pi)$. Aleshores per tot $k \in E$ es compleix:*

$$P(X_n = k) = \gamma_k^{(n)}.$$

Demostració.

$$\begin{aligned} P(X_n = k) &= \sum_{h\in E} P(X_n = k \mid X_0 = h)P(X_0 = h) \\ &= \sum_{h\in E} P(X_0 = h)p_{hk}^{(n)} = \sum_{h\in E} \gamma_h p_{hk}^{(n)}. \end{aligned}$$

□

Aquesta proposició dóna pas al següent corol·lari:

Corol·lari 2.2.3. *Sigui* $\{X_n : n \geq 0\}$ *una* $CMH(\gamma, \Pi)$*, es compleix:*

1. $\gamma^{(l+k)} = \gamma^{(l)} \cdot \Pi^k$
2. $\gamma_j^{(n)} = P(X_n = j) = \sum_{h \in E} \gamma_h p_{hj}^{(n)} = \sum_{h \in E} \sum_{i_1, \ldots, i_{n-1} \in E^{m-1}} \gamma_h p_{hi_1} p_{i_1 i_2} \cdots p_{i_{n-1} j}$.

Prova: Veiem d'on surt la primera igualtat:

$$\gamma_i^{l+1} = P(X_{l+k} = i) = \sum_{j \in E} P(X_{l+k} = i \mid X_l = j) P(X_l = j) = \sum_{j \in E} \gamma_j^{(l)} p_{ji}^{(k)}$$

i, per tant: $\gamma^{(l+k)} = \gamma^{(l)} \, \Pi^k$. □

Proposició 2.2.4. *La distribució inicial* γ *i la matriu de probabilitats de transició* Π *determinen la llei del vector aleatori* $(X_{n_1}, \ldots, X_{n_k})$*,* $0 \leq n_1 < n_2 < \cdots < n_k$*.*

Demostració. Primer de tot provarem que es compleix pel cas $n_1 = 0, \ldots, n_k = k$:

$$\begin{aligned} &P(X_0 = i_0, \ldots, X_{k-1} = i_{k-1}, X_k = i_k) \\ &\quad = P(X_0 = i_0) P(X_1 = i_1 \mid X_0 = i_0) P(X_2 = i_2 \mid X_0 = i_0, X_1 = i_1) \times \cdots \\ &\qquad \cdots \times P(X_k = i_k \mid X_0 = i_0, X_1 = i_1, \ldots, X_{k-1} = i_{k-1}) \\ &\quad = \gamma_{i_0} p_{i_0 i_1} p_{i_1 i_2} \cdots p_{i_{k-1} i_k}. \end{aligned}$$

I utilitzant això:

$$\begin{aligned} P(X_{n_1} = i_1, \ldots, X_{n_k} = i_k) &= P(X_{n_1} = i_1) P(X_{n_2 = i_2} \mid X_{n_1} = i_1) \times \cdots \\ &\qquad \cdots \times P(X_{n_k} = i_k \mid X_{n_1} = i_1, \ldots, X_{n_{k-1}} = i_{k-1}) \\ &= \sum_{h \in E} \gamma_h p_{hi_1}^{(n_1)} p_{i_1 i_2}^{(n_2 - n_1)} \cdots p_{i_{k-1} i_k}^{(n_k - n_{k-1})}. \end{aligned}$$

□

Capítol 3

Cadenes de Markov ocultes

Una de les limitacions de les cadenes de Markov tractades al capítol anterior és el fet que considerem un model observable, un model on cada estat pot ser determinat (és un observable físic). En molts casos el que trobem és que la nostra observació no es correspon directament amb l'estat sinó que és una funció probabilística d'aquest, l'estat no és un observable.

Exemple 3.0.5. Suposem que hi ha algú que llença una (o diverses) moneda(es) i ens va dient "cara" o "creu" en funció del que surt cada vegada. Suposem que aquesta és tota la informació que tenim, no sabem si tota l'estona tira la mateixa moneda o si en té més i va alternant. En aquest cas, i suposant que l'elecció de la moneda segueix un model de Markov, tindríem els estats no observables (quina moneda tira) i les observacions (cara i creu) que sortirien amb una probabilitat o una altra en funció de la moneda que s'està tirant (veure figura (3.1.1)).

Aquesta serà la situació que ens trobarem després, quan vulguem tractar les dades obtingudes en l'experiment d'*unzipping*. Obtindrem la mesura corresponent a una distància però aquestes mesures estaran afectades per soroll/fluctuacions i, per tant, no coneixerem la posició "real". Per poder tractar aquests casos cal estendre el concepte de cadena de Markov introduint les *cadenes de Markov ocultes* (hidden Markov models, HMM). En aquest capítol en definirem el concepte, parlarem dels tres problemes fonamentals i explicarem els algorismes bàsics per resoldre'ls.

3.1 Definició: Cadena de Markov oculta

Definició 3.1.1. Procés de Markov ocult
Sigui $\{X_n \colon n \geq 1\}$ una cadena de Markov amb conjunt d'estats finit E. Sigui $(\Sigma, \mathcal{Y})$ un espai mesurable tal que per cada $x \in E$ existeix una funció de probabilitat $P(\cdot \mid x) \colon \mathcal{Y} \longrightarrow [0,1]$ en l'espai Σ. Aleshores el procés bivariant $\{(X_n, Y_n) \colon n \geq 1\}$, $(X_n, Y_n) \colon \longrightarrow E \times V$ és un *procés de Markov ocult* si per tot $n \geq 1$, $B_k \in \mathcal{Y}$ satisfà:

$$P(Y_1 \in B_1, \ldots, Y_n \in B_n \mid X_1 = x_1, \ldots, X_n = x_n) = \prod_{k=1}^{n} P(B_k \mid x_k). \qquad (3.1.1)$$

Proposició 3.1.2. *Donada la trajectòria del procés d'estats, les observacions Y_k són independents i la seva distribució depèn únicament de l'estat X_k.*

Demostració. La independència de les observacions es veu directament de la propietat (3.1.1) que caracteritza les cadenes de Markov ocultes.
Per veure la forma de la distribució, fixem $B_k = B$ i prenem $B_j = V$ per tot $j \neq k$, aleshores

$$\begin{aligned} P(Y_k \in B \mid X_1 = x_1, \ldots, X_n = x_n) &= \prod_{k=1}^{n} P(B_k, x_k) = P(B, x_k) \\ &= P(Y_k \in B \mid X_k = x_k). \end{aligned}$$

On a l'última igualtat hem utilitzat que la probabilitat $P(\cdot \mid x_k)$ és independent d'x_j per tot $j \neq k$. □

Observacions 3.1.3. Hipòtesis sobre $\{Y_n \colon n \geq 1\}$:

- Si l'espai d'observacions V és finit es diu que el procés és un *HMM amb alfabet finit*. Fins que no diguem el contrari, suposarem que ens trobem en aquest cas.
- També ens centrarem en el cas on la cadena de Markov $\{X_n \colon n \geq 1\}$ és homogènia.

Un cop ja hem vist la definició, anem a descriure cadascun dels elements que la formen:

- Els *estats del model*, $E = \{x_1, x_2, \ldots, x_N\}$. E és un conjunt que conté els N valors que pot prendre la cadena de Markov $\{X_n \colon n \geq 1\}$ que com hem dit corresponen a estats no observables físicament.
- El *conjunt d'observacions*, V. És el conjunt de totes les observacions que podem obtenir pels diferents estats, els observables físics. En el cas d'un HMM amb alfabet finit la seva dimensió, M, també serà rellevant i podrem escriure'l explícitament: $V = \{y_1, y_2, \ldots, y_M\}$.
- Les *probabilitats de transició* de la cadena de Markov $\{X_n \colon n \geq 1\}$, representades en la matriu $C = (c_{x_i x_j})_{i,j}$ de dimensió $N \times N$ amb elements:

$$c_{x_i x_j} = P(X_{n+1} = x_j \mid X_n = x_i).$$

- La *distribució de les observacions* corresponents a l'estat $x_i \in E$ per cada $y_j \in V$, formant la matriu de dimensió $N \times M$, $B = (b_{x_i y_j})_{i,j}$, on:

$$b_{x_i y_j} = b_{x_i}(y_j) = P(Y_n = y_j \mid X_n = x_i).$$

- La *distribució inicial* de la cadena de Markov $\{X_n \colon n \geq 1\}$: $\Pi = (\pi_{x_i})_i$ on

$$\pi_{x_i} = P(X_1 = x_i),\ 1 \leq i \leq N.$$

Notació. En algunes ocasions quan tinguem $X_n = x_i$ utilitzarem la notació abreviada q_n per representar l'estat $x_i \in E$. Anàlogament per les observacions, o_n representarà de forma abreviada el valor y_i obtingut en l'instant n quan tinguem $Y_n = y_i$, amb $y_i \in V$.
D'altra banda, els paràmetres que determinen el model es representen amb la lletra λ:

$$\lambda = (C, B, \Pi). \tag{3.1.2}$$

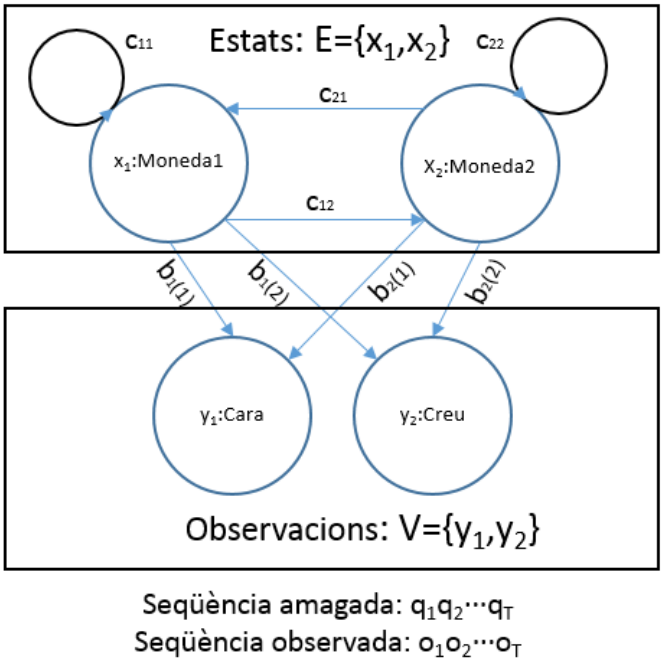

Figura 3.1.1: Representació, amb cadascun dels elements, de la cadena de Markov vista en l'exemple anterior en el cas particular de dues monedes diferents

Coneguda la forma que té una cadena de Markov oculta i donada una seqüència d'observacions $O = o_1 \cdots o_T$ (abreviació per referir-nos a $Y_1 = o_1, \ldots, Y_T = o_T$) sorgeixen tres problemes bàsics que ens interessa resoldre:

1. **Determinar la versemblança de la seqüència observada respecte els paràmetres del model.**
   Donada una seqüència d'observacions, $O = o_1 o_2 \cdots o_T$, i els paràmetres del model $\lambda = (C, B, \Pi)$, com calculem la probabilitat que la nostra observació sigui produïda pel model, $P(O \mid \lambda)$, de forma eficient? O, dit d'una altra manera, com de bé encaixen un model donat i una seqüència observada?

2. **Trobar la seqüència més probable donat el model.**
   Donada una seqüència d'observacions, $O = o_1 \cdots o_T$, i el model $\lambda = (C, B, \Pi)$, com escollim la seqüència corresponent als estats, $Q = q_1 q_2 \cdots q_T$ que descriu més bé les observacions? (Q, de manera similar a les observacions, és l'abreviació de $X_1 = q_1, \ldots, X_T = q_T$). El que volem fer és descobrir la part amagada

del nostre model, trobar la seqüència d'estats "correcta". Ara bé, cal tenir en compte que no hi ha una seqüència "correcta" sinó que el que fem és utilitzar un criteri d'optimització per resoldre aquest problema el més acuradament possible i aquest criteri dependrà de quin sigui el nostre objectiu: trobar la seqüència d'estats individuals més probable, la de parelles d'estats (q_n, q_{n+1}) més probables, etc.

3. **Optimitzar els paràmetres del model.**
   Quins són els paràmetres, $\lambda = (C, B, \Pi)$, que maximitzen $P(O \mid \lambda)$? El que volem fer aquí és ajustar els paràmetres del model per tal que maximitzin la probabilitat d'obtenir l'observació donada, $O = o_1 o_2 \cdots o_T$, assumint que els paràmetres són aquests, $\lambda = (C, B, \Pi)$.

3.2 Solució als problemes bàsics. Algorismes

Per tal de trobar respostes de forma eficient a les tres qüestions que acabem de formular hi ha diversos algorismes. Nosaltres ens centrarem en l'**algorisme forward-backward** per calcular la probabilitat de l'observació, l'**algorisme de Viterbi** per determinar la seqüència més probable i l'**algorisme de Baum-Welch** per tal d'estimar els paràmetres del model. Els explicarem amb detall a continuació i, a més a més, veurem que serveixen per resoldre altres qüestions com el *problema de filtrat* i el *problema de predicció*. Aquests procediments ens seran útils a l'últim capítol del treball que dedicarem a una qüestió de biofísica: trobar el patró de desplegament de l'ADN en l'experiment d'*unzipping*.

Notació. Utilitzarem la notació O^n per referir-nos a la seqüència $o_1 o_2 \cdots o_n$, amb $n \leq T$ i O_m^n per referir-nos a $o_m o_{m+1} \cdots o_n$. Anàlogament utilitzarem Q^n i Q_m^n quan parlem de seqüències parcials d'estats.

3.2.1 Calcular $P(O \mid \lambda)$: algorisme forward-backward

Volem calcular la probabilitat d'obtenir una observació $O = o_1 o_2 \cdots o_T$ donat un model λ. Començarem fent-ho enumerant totes les seqüències d'estats de longitud T possibles però veurem que ens suposa un nombre d'operacions desorbitat. D'aquesta manera remarcarem la importància de l'existència d'un mètode per calcular-la de forma eficient.

Donada una seqüència d'estats: $Q = q_1 q_2 \cdots q_T$, la probabilitat d'obtenir l'observació O, sabent que les observacions són independents entre si i que la distribució d'aquestes depèn únicament de l'estat on ens trobem (proposició (3.1.2)) és:

$$P(O \mid Q, \lambda) = \prod_{n=1}^{T} P(Y_n = o_n \mid X_n = q_n, \lambda) = \prod_{n=1}^{T} b_{q_n}(o_n). \tag{3.2.1}$$

D'altra banda, com que $\{X_n \colon n \geq 1\}$ és un procés de Markov el teorema (2.1.5)

ens diu que:

$$P(Q \mid \lambda) = \pi_{q_1} \prod_{n=2}^{T} c_{q_{n-1}q_n}. \tag{3.2.2}$$

De manera que obtenim, utilitzant la definició de probabilitat condicionada, els teoremes (1.0.7) i (1.0.9) i les equacions (3.2.1) i (3.2.2):

$$\begin{aligned} P(O \mid \lambda) &= \sum_{Q} P(O, Q \mid \lambda) = \sum_{Q} P(O \mid Q, \lambda) P(Q \mid \lambda) \\ &= \sum_{q_1, q_2, \ldots, q_T \in E^T} \pi_{q_1} b_{q_1}(o_1) \prod_{n=2}^{T} c_{q_{n-1}q_n} b_{q_n}(o_n). \end{aligned} \tag{3.2.3}$$

Si volem calcular $P(O \mid \lambda)$ directament, utilitzant l'equació (3.2.3), hem de tenir en compte que a cada instant ($n = 1, 2, \ldots, T$) hi ha N estats possibles el que representa N^T seqüències d'estat en total i que per cada seqüència tenim $2T$ operacions, per tant, de l'ordre de TN^T operacions. Per tal de resoldre-ho de manera més eficient s'utilitza la primera part de l'algorisme *forward-backward*.

Fixem-nos que seria interessant reduir el nombre de passos en el càlcul de la probabilitat $P(O, Q \mid \lambda)$. Amb la finalitat d'aconseguir això, l'algorisme anomenat *forward-backward* defineix la variable **forward** que representa la probabilitat de l'observació de la seqüència $O = o_1 o_2 \cdots o_n$ i l'estat $X_n = x_i$ fins a un instant $n \leq T$ coneguts els paràmetres del model:

$$\alpha_n(x_i) = P(O^n, X_n = x_i \mid \lambda). \tag{3.2.4}$$

Com que coneixem la distribució inicial i la relació entre ambdues variables podem calcular el cas inicial:

$$\alpha_1(x_i) = P(Y_1 = o_1, X_1 = x_i \mid \lambda) = \pi_{x_i} b_{x_i}(o_1), \ 1 \leq i \leq N.$$

A més a més, tenim:

$$\begin{aligned} \alpha_{n+1}(x_i) &= P(O^{n+1}, X_{n+1} = x_i \mid \lambda) = \sum_{k=1}^{N} P(O^{n+1}, X_n = x_k, X_{n+1} = x_i \mid \lambda) \\ &= \sum_{k=1}^{N} P(Y_{n+1} = o_{n+1} \mid O^n, X_n = x_k, X_{n+1} = x_i, \lambda) \times \\ &\qquad \times P(O^n, X_n = x_k, X_{n+1} = x_i \mid \lambda) \\ &= \sum_{k=1}^{N} P(Y_{n+1} = o_{n+1} \mid O^n, X_n = x_k, X_{n+1} = x_i, \lambda) \times \\ &\qquad \times P(X_{n+1} = x_i \mid O^n, X_n = x_k, \lambda) P(O^n, X_n = x_k \mid \lambda). \end{aligned}$$

Per últim, gràcies a la proposició (3.1.2) i tenint en compte que $\{X_n \colon n \geq 1\}$ és un procés de Markov:

$$\begin{aligned} &P(Y_{n+1} = o_{n+1} \mid O^n, X_n = x_k, X_{n+1} = x_i, \lambda) \\ &\quad = P(Y_{n+1} = o_{n+1} \mid X_{n+1} = x_i, \lambda) = b_{x_i}(o_{n+1}). \\ &P(X_{n+1} = x_i \mid X_n = x_k, O^n, \lambda) = P(X_{n+1} = x_i \mid X_n = x_k, \lambda) = c_{x_k x_i}. \end{aligned}$$

De manera que $\alpha_{n+1}(x_i) = \sum_{k=1}^{N} b_{x_i}(o_{n+1}) c_{x_k x_i} \alpha_n(x_k)$.

Així doncs, podem calcular els diferents valors de la variable *forward* de forma recursiva per tot $x_i \in E$ (veure la figura (3.2.1)):

- Inicialització, cas $n = 1$:

$$\alpha_1(x_i) = \pi_{x_i} b_{x_i}(o_1),\ 1 \leq i \leq N.$$

- Iteració, per calcular els casos $n = 2, 3, \ldots, T-1$ recursivament:

$$\alpha_{n+1}(x_i) = b_{x_i}(o_{n+1}) \sum_{k=1}^{N} c_{x_k x_i} \alpha_n(x_k),\ 1 \leq i \leq N. \tag{3.2.5}$$

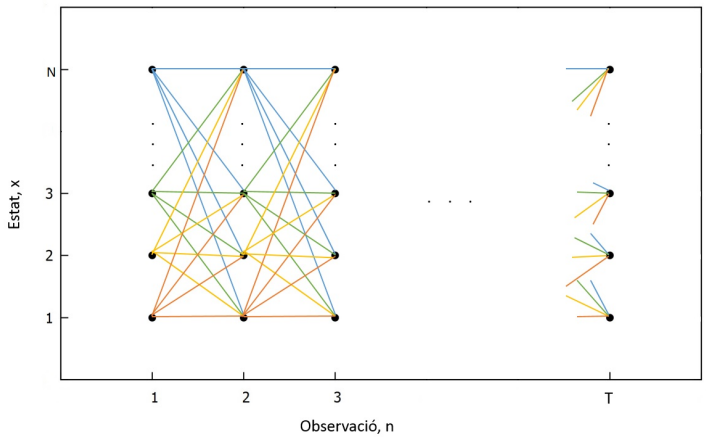

Figura 3.2.1: El càlcul de la probabilitat forward té en compte l'estructura reticular que mostra la figura. Per $n = 2, 3, \ldots, T$ només cal calcular $\alpha_n(x_j)$ pels N diferents x_j que hi ha a E i considerar únicament els N valors previs d'$\alpha_{n-1}(x_i)$ ja que cadascun dels N estats possibles a l'instant n s'assoleix a partir dels mateixos N estats en l'instant $n-1$.

De manera que ja podem resoldre la qüestió anterior i calcular la probabilitat que buscàvem:

$$P(O \mid \lambda) = \sum_{Q} P(O, Q \mid \lambda) = \sum_{i=1}^{N} \alpha_T(x_i). \tag{3.2.6}$$

Abans hem vist que calculant-ho de forma directa necessitàvem de l'ordre de TN^T operacions. Ara, en canvi, n'hem de fer molts menys, de l'ordre d'N^2T càlculs [6]. Si prenem, per exemple, el cas concret $N = 2$, $T = 100$ estem passant de l'ordre de 10^{32} càlculs a 400. Si tenim en compte que en la majoria de camps on s'usen HMM comptem amb cadenes més llargues i/o de més estats on aquesta diferència es veu accentuada veiem clarament la utilitat d'aquest mètode.

A més a més, $\alpha_n(x_j)$ ens permet calcular la probabilitat de trobar-nos en un estat particular donada la seqüència parcial observada fins aquell moment, conegut

com a *problema de filtrat*. Si he observat la seqüència parcial $O^n = o_1 o_2 \cdots o_n$ la probabilitat que a l'instant $n \leq T$ em trobi en un estat $x_j \in E$ particular, $X_n = x_j$, és:

$$P(X_n = x_j \mid O^n, \lambda) = \frac{P(X_n = x_j, O^n \mid \lambda)}{P(O^n \mid \lambda)} = \frac{P(X_n = x_j, O^n \mid \lambda)}{\sum_{j=1}^{N} P(O^n, X_n = x_j \mid \lambda)}$$
$$= \frac{\alpha_n(x_j)}{\sum_{j=1}^{N} \alpha_n(x_j)}.$$

També ho podem utilitzar per resoldre el *problema de predicció* que consisteix a calcular la probabilitat de trobar-me a l'estat $x_i \in E$ en l'instant $n+1$ donada la seqüència parcial O^n: $P(X_{n+1} = x_i \mid O^n, \lambda)$.

Fem un pas previ i a continuació en derivarem la fórmula:

$$\begin{aligned} P(X_{n+1} = x_i, O^n \mid \lambda) &= \sum_{k=1}^{N} P(X_{n+1} = x_i, X_n = x_k, O^n \mid \lambda) \\ &= \frac{1}{P(O^n \mid \lambda)} \sum_{k=1}^{N} P(X_n = x_k, O^n \mid \lambda) c_{x_k x_i} \\ &= \sum_{k=1}^{N} P(X_n = x_k \mid O^n, \lambda) c_{x_k x_i}. \end{aligned}$$

I utilitzant aquest resultat podem obtenir la probabilitat buscada:

$$\begin{aligned} P(X_{n+1} = x_i \mid O^n, \lambda) &= \frac{P(X_{n+1} = x_i, O^n \mid \lambda)}{P(O^n \mid \lambda)} \\ &= \frac{1}{P(O^n \mid \lambda)} \sum_{k=1}^{N} P(X_n = x_k, X_{n+1} = x_i, O^n \mid \lambda) \\ &= \frac{1}{P(O^n \mid \lambda)} \sum_{k=1}^{N} P(X_n = x_k, O^n \mid \lambda) c_{x_k x_i} \\ &= \sum_{k=1}^{N} P(X_n = x_k \mid O^n, \lambda) c_{x_k x_i}. \end{aligned} \tag{3.2.7}$$

El resultat que acabem de trobar a l'equació (3.2.7) és similar a l'observació (2.1.6), aplicable a cadenes de Markov. La reformulem a continuació pel cas de les cadenes de Markov ocultes.

Observació 3.2.1. Siguin $\lambda = (C, B, \Pi)$ els paràmetres d'una cadena de Markov oculta, $\{X_n \colon n \geq 1\}$ el procés de Markov associat i O^T una seqüència d'observacions; aleshores $\{X_{n+m} \colon m \geq 0\}$ és una cadena de Markov amb matriu de transició

C i distribució inicial $\overline{P}(X_n = x_i) = P(X_n = x_i \mid O^n, \lambda)$. És a dir, coneguts les probabilitats del problema de filtrat, $P(X_n = x_i \mid O^n, \lambda)$, el procés d'estats futurs es comporta com una cadena de Markov amb aquesta distribució inicial.

De manera similar a $\alpha_n(x_i)$ podem considerar la variable **backward** que representa la probabilitat d'obtenir la seqüència parcial des de l'instant immediatament posterior, $n+1$, fins a T donats els paràmetres del model, $\lambda = (C, B, \Pi)$ i l'estat en l'instant n:

$$\beta_n(x_i) = P(O_{n+1}^T \mid X_n = x_i, \lambda). \tag{3.2.8}$$

Nota 3.2.2. Com ja hem vist, aquesta part de l'algorisme forward-backward no ens cal per trobar $P(O \mid \lambda)$ però creiem oportú introduir-la ara ja que ens serà útil a l'hora de resoldre el tercer problema, optimitzar els paràmetres del model.

Ens interessa trobar un algorisme que ens permeti calcular, de la mateixa manera que $\alpha_n(x_i)$, les $\beta_n(x_i)$ recursivament (veure figura (3.2.2)). Observem que:

$$\begin{aligned}
\beta_n(x_i) = P(O_{n+1}^T \mid X_n = x_i, \lambda) &= \sum_{k=1}^{N} P(O_{n+1}^T, X_{n+1} = x_k \mid X_n = x_i, \lambda) \\
&= \sum_{k=1}^{N} \frac{P(O_{n+1}^T, X_{n+1} = x_k, X_n = x_i \mid \lambda)}{P(X_n = x_i \mid \lambda)} \\
&= \sum_{k=1}^{N} \frac{P(O_{n+1}^T \mid X_{n+1} = x_k, X_n = x_i, \lambda)}{P(X_n = x_i \mid \lambda)} P(X_{n+1} = x_k, X_n = x_i \mid \lambda) \\
&= \sum_{k=1}^{N} P(O_{n+1}^T \mid X_{n+1} = x_k, X_n = x_i, \lambda) P(X_{n+1} = x_k \mid X_n = x_i, \lambda) \\
&= \sum_{k=1}^{N} P(O_{n+1}^T \mid X_{n+1} = x_k, X_n = x_i, \lambda) c_{x_i x_k}.
\end{aligned}$$

A més a més, si $n < T$:

$$\begin{aligned}
P(O_{n+1}^T \mid X_{n+1} = x_k, X_n = x_i, \lambda) &= P(Y_{n+1} = o_{n+1}, O_{n+2}^T \mid X_{n+1} = x_k, X_n = x_i, \lambda) \\
&= P(Y_{n+1} = o_{n+1} \mid X_{n+1} = x_k, \lambda) P(O_{n+2}^T \mid X_{n+1} = x_k, \lambda) \\
&= b_{x_k}(o_{n+1}) \beta_{n+1}(x_k).
\end{aligned}$$

On a l'última igualtat hem utilitzat la proposició (3.1.2).

Ara ja podem introduir la recurrència cap enrere (equació backward) i calcular $\beta_n(x_i)$ recursivament pels N estats x_i d'E:

- Inicialització, cas $n = T$:

$$\beta_T(x_i) = 1, \ 1 \leq i \leq N.$$

- Iteració, obtenim els cassos $n = T-1, T-2, \ldots, 1$ recursivament:

$$\beta_n(x_i) = \sum_{j=1}^{N} c_{x_i x_j} b_{x_j}(o_{n+1}) \beta_{n+1}(x_j), \quad 1 \leq i \leq N. \tag{3.2.9}$$

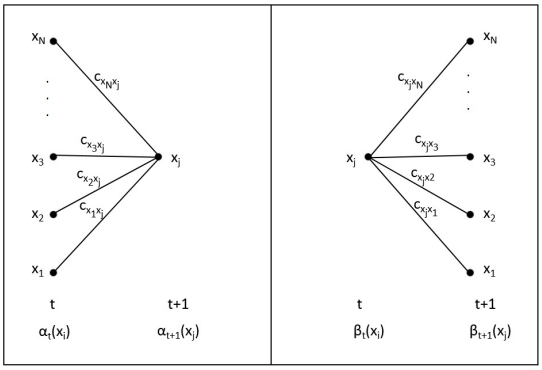

Figura 3.2.2: Representació de la seqüència d'operacions necessàries per calcular les variables $\alpha_{t+1}(x_j)$ i $\beta_t(x_j)$ respectivament.

3.2.2 Seqüència d'estats més probable. Algorisme de Viterbi

A diferència de l'apartat anterior, on érem capaços de trobar una solució exacta, les coses es compliquen quan busquem respostes al segon problema i pretenem trobar la seqüència d'estats *òptima* associada al model i a la nostra observació. Com ja apuntàvem unes pàgines enrere, la primera dificultat que ens trobem és saber a què ens estem referint quan emprem la paraula *òptima*. Una opció seria escollir els estats $x_i \in E$ que són individualment més probables. Aquest criteri d'optimització maximitza el nombre esperat d'estats individuals correctes i podem aprofitar les variables *forward* i *backward* per resoldre'l.

El conjunt de les dues variables ens permet calcular la probabilitat de trobar-nos en un estat concret a l'instant de temps $n \leq T$. Per fer-ho, observem que:

$$\begin{aligned} P(X_n = x_i, O \mid \lambda) &= P(X_n = x_i, O^n, O_{n+1}^T \mid \lambda) \\ &= P(O_{n+1}^T \mid X_n = x_i, O^n, \lambda) P(X_n = x_i, O^n \mid \lambda). \end{aligned}$$

I, a més a més, gràcies a la propietat de Markov (donat X_n amb $1 \leq n \leq T$ el passat i el futur són independents) i aplicant la proposició (3.1.2) podem afirmar que les observacions també són independents i, per tant, $P(O_{n+1}^T \mid X_n = x_i, O^n, \lambda) = P(O_{n+1}^T \mid X_n = x_i)$. Directament, doncs, obtenim:

$$P(X_n = x_i, O \mid \lambda) = \alpha_n(x_i)\beta_n(x_i).$$

Ara ja som capaços de resoldre el que es coneix com el *problema de suavitzat* que consisteix a trobar un estimador de l'estat en un instant de temps n, $\hat{X}_n$, coneguts els paràmetres del model, λ, i donada una seqüència observada de longitud $T > k$, $O = o_1 \dots o_T$. Per fer-ho només ens cal definir una nova variable, $\gamma_n(x_i)$ que podem calcular mitjançant $\alpha_n(x_i)$ i $\beta_n(x_i)$ per tot $x_i \in E$:

$$\gamma_n(x_i) = P(X_n = x_i \mid O, \lambda) = \frac{P(X_n = x_i, O \mid \lambda)}{P(O \mid \lambda)} = \frac{\alpha_n(x_i)\beta_n(x_i)}{\sum_{k=1}^{N} \alpha_n(x_k)\beta_n(x_k)}. \quad (3.2.10)$$

I finalment tindrem que l'estat individual més probable en l'instant de temps n serà el que ens proporcioni una probabilitat més alta, l'estat $x_i \in E$ que maximitzi el valor que pren γ_n que ens proporcionarà la funció argmàx i denotarem $\hat{q}_n$:

$$\hat{q}_n = \underset{1\leq i\leq N}{\text{argmàx}}\{\gamma_n(x_i)\},\ 1 \leq n \leq T.$$

Observació 3.2.3. Cal tenir en compte que si tenim transicions entre estats prohibides, és a dir, $c_{x_i x_j} = 0$ per alguns $x_i, x_j \in E$ pot ser que utilitzant aquest mètode obtinguem una seqüència que no és possible que es doni en realitat. És a dir, si jo tinc un sistema on la transició entre els estats x_1 i x_3, per exemple, està prohibida és possible que trobem $\hat{Q} = \hat{q}_1 \dots x_1 x_3 \dots \hat{q}_T$ que és una seqüència impossible de trobar a la vida real. Això passa perquè el que estem fent és maximitzar el nombre esperat d'estats correctes fixant-nos en cada instant per separat, sense tenir en compte els moments adjacents.

Per evitar aquest problema es podria trencar la seqüència en parelles, triplets, etc. i buscar la seqüència d'estats que maximitzi el conjunt $(q_n, q_{n+1}), (q_n, q_{n+1}, q_{n+2}), etc.$ Encara que sembla un sistema raonable i que es podria fer servir més d'una vegada, normalment no s'utilitza i el que es fa és buscar la seqüència completa d'estats més probable utilitzant l'**algorisme de Viterbi**, basat en mètodes de programació dinàmica i fent-ho de manera més eficient que utilitzant l'algorisme forward-backward.

Per construir els passos de l'algorisme, presentat per Andrew James Viterbi el 1967, comencem escrivint l'expressió de la probabilitat conjunta que és similar a l'equació (3.2.3):

$$P(Q^n, O^n \mid \lambda) = \pi_{q_1} b_{q_1}(o_1) \prod_{m=2}^{n} c_{q_{m-1}q_m} b_{q_m}(o_m).$$

Definim dues noves funcions, u_1 i u_m, que representaran respectivament les probabilitats inicial i de transició per cadascun dels estats $x \in E$.

$$u_1(x) = \pi_x b_x(o_1),\ \text{per tot } x \in E.$$
$$u_m(x_i, x_j) = c_{x_i x_j} b_{x_j}(o_m),\ \text{per tot } x_i, x_j \in E \text{ amb } 2 \leq m \leq T.$$

Permetent-nos reescriure la probabilitat en termes d'aquestes funcions:

$$P(Q^n, O^n \mid \lambda) = u_1(q_1) \prod_{m=2}^{n} u_m(q_{m-1}, q_m).$$

Suposem ara que per cada estat $x \in E$ coneixem la probabilitat $\delta_m(x)$ de la trajectòria òptima per arribar a x en l'instant m, a la que anomenarem *puntuació*. El cas $m = 1$ està clar ja que hi ha una única manera d'arribar-hi: $\delta_1(x) = u_1(x)$. A partir d'aquí la trajectòria òptima per arribar a x en l'instant $m + 1$ ha d'haver passat per algun estat en l'instant m i hi ha d'haver arribat de manera òptima (si no fos així, tindríem una altra seqüència amb una probabilitat major). Així doncs,

la puntuació (probabilitat més alta possible) d'estar a l'estat x a l'instant $m+1$ serà:

$$\delta_{m+1}(x) = \max_{1\leq i\leq N}\{\delta_m(x_i)u_m(x_i,x)\}.$$

Quan arribem a $m=n$ tindrem el valor d'N puntuacions que es corresponen amb cadascun dels estats $x\in E$ possibles $\delta_n(x) = P(Q^{n-1}, X_n = x, O^n \mid \lambda)$. La trajectòria òptima estarà determinada per l'estat x que tingui una puntuació més alta, és a dir, la probabilitat d'arribar-hi sigui major. Seguint amb la notació anterior:

$$\hat{q}_n = \operatorname*{argmàx}_{x\in E}\delta_n(x).$$

Un cop hem obtingut $\hat{q}_n$, l'estat en l'instant n que forma part de la trajectòria òptima, podem reconstruir la seqüència completa d'estats anant cap enrere de forma recursiva. Els passos que cal seguir en l'algorisme de Viterbi ens portaran primer a calcular les puntuacions i després a desfer el camí per trobar els estats. Per fer-ho, utilitzarem una variable auxiliar, $\phi_m(x_i)$, on emmagatzemarem els estats que maximitzen $\delta_m(x_i)$ en cada instant m i per cada estat $x_i\in E$. A més a més per tal de mantenir-nos tota l'estona dins els rangs de computació i prevenir l'*underflow* en calcularem els logaritmes:

1. Inicialització:

$$\delta_1(x_i) = \log[u_1(x_i)] = \log[\pi_{x_i}] + \log[b_{x_i}(o_1)],\ 1\leq i\leq N.$$
$$\phi_1(x_i) = 0,\ 1\leq i\leq N.$$

2. Recursió, per $2\leq m\leq n$:

$$\delta_m(x_j) = \max_{1\leq i\leq N}\{\delta_{m-1}(x_i) + \log[u_m(x_i,x_j)]\},\ 1\leq j\leq N;$$
$$\phi_m(x_j) = \operatorname*{argmàx}_{1\leq i\leq N}\{\delta_{m-1}(x_i)c_{x_ix_j}\},\ 1\leq j\leq N.$$

D'aquesta manera trobarem l'estat $\hat{q}_n$ així com la puntuació que li correspon, el logaritme de la probabilitat de la seqüència òptima P^*:

$$\log(P^*) = \max_{1\leq i\leq N}\{\delta_n(x_i)\};$$
$$\hat{q}_n = \operatorname*{argmàx}_{1\leq i\leq N}\{\delta_n(x_i)\}.$$

I podrem reconstruir la seqüència completa d'estats més probable desfent el camí que hem fet:

$$\hat{q}_m = \phi_{m+1}(\hat{q}_{m+1}),\ m = n-1, n-2, \ldots, 1.$$

Observem que els passos 1 i 2 de l'algorisme de Viterbi són pràcticament els mateixos que seguíem en el càlcul de la variable *forward* exceptuant l'ús de logaritmes i que aquí estem maximitzant en funció dels estats previs mentre allà el que fèiem era sumar. A més a més, ara estem "guardant" els estats per poder fer la recursió enrere i trobar la seqüència òptima.

3.2.3 Optimitzar els paràmetres: $\lambda = (C, B, \Pi)$. Algorisme de Baum-Welch.

Finalment arribem al tercer, i alhora el més complicat dels problemes, ajustar els paràmetres $\lambda = (C, B, \Pi)$ que maximitzin la probabilitat d'obtenir les nostres observacions conegut el model. És el més difícil però és molt important ja que els paràmetres ens fan falta per poder resoldre qualsevol altra qüestió que ens puguem plantejar, ja hem vist que els suposàvem coneguts per tal de resoldre els altres dos problemes.

No es coneix cap manera analítica de resoldre'l i, de fet, donada una seqüència d'observacions finita no hi ha cap manera òptima d'estimar els paràmetres corresponents al model. Ara bé, el que sí que podem fer és utilitzar un mètode iteratiu com l'algorisme de Baum-Welch o tècniques de gradient per tal d'escollir els paràmetres $\lambda = (C, B, \Pi)$ que ens portin a assolir un màxim local de la probabilitat $P(O \mid \lambda)$. Com indica el títol d'aquesta secció, nosaltres ens centrarem en l'**algorisme de Baum-Welch**, desenvolupat entre els anys 1966 i 1972 per Leonard E. Baum y Lloyd R. Welch, basat en la teoria de funcions de probabilitat d'una cadena de Markov juntament amb l'algorisme d'Esperança-Maximització (algorisme EM) [7].

Així doncs, el nostre objectiu és determinar una estimació de $\lambda = (C, B, \Pi)$, els paràmetres que donen lloc al nostre HMM, a partir d'una successió d'observacions $O = o_1 \dots o_T$. El criteri que utilitza l'algorisme de Baum-Welch per tal d'ajustar-lo és el conegut com a *criteri de màxima versemblança de les observacions* que consisteix a utilitzar $\hat{\lambda}$ que compleixi:

$$\hat{\lambda} = \underset{\lambda}{\text{argmàx}}\, P(O \mid \lambda),$$

on $P(O \mid \lambda)$ representa la funció de versemblança de la mostra (equació (3.2.3)). Veiem que $P(O \mid \lambda)$ no és fàcil de calcular i, a més, la seva relació amb els paràmetres és complexa. Per aquest motiu no es fa la maximització directament sinó que es maximitza utilitzant l'algorisme de Baum-Welch.

La idea d'aquest algorisme és trobar, a partir d'una aproximació del model que suposem coneguda i denotem λ_m uns nous paràmetres, λ_{m+1}, que ens augmentin el valor de la funció de versemblança: $P(O \mid \lambda_m) \leq P(O \mid \lambda_{m+1})$.

Per fer això comencem definint una nova variable, $\xi_n(x_i, x_j)$, que representa la probabilitat de trobar-nos a l'estat $x_i \in E$ en l'instant n i a l'estat $x_j \in E$ a l'instant $n+1$, donats els paràmetres del model, $\lambda = (C, B, \Pi)$ i l'observació, $O = o_1 o_2 \cdots o_T$. Podem aprofitar les variables *forward* i *backward* definides anteriorment per tal de calcular-la més eficientment (veure figura (3.2.3)):

$$\begin{aligned}\xi_n(x_i, x_j) = P(X_n = x_i, X_{n+1} = x_j \mid O, \lambda) &= \frac{P(O, X_n = x_i, X_{n+1} = x_j \mid \lambda)}{P(O \mid \lambda)} \\ &= \frac{P(X_n = x_i, O^n \mid \lambda) P(X_{n+1} = x_j, Y_{n+1} = o_{n+1} \mid X_n = x_i, O^n, \lambda)}{P(O \mid \lambda)} \times \\ &\quad \times P(O_{n+2}^T \mid X_{n+1} = x_j, \lambda).\end{aligned}$$

Reescrivim-ne el terme més llarg utilitzant la definició de probabilitat condicionada (1.0.6) per formular-lo en termes que ja coneixem:

$$\begin{aligned} P(X_{n+1} = x_j, Y_{n+1} = o_{n+1} \mid X_n = x_i, O^n, \lambda) = \\ = P(Y_{n+1} = o_{n+1} \mid X_{n+1} = x_j, X_n = x_i, O^n, \lambda) \times \\ \times P(X_{n+1} = x_j \mid X_n = x_i, O^n, \lambda) = b_{x_j}(o_{n+1}) c_{x_1 x_j}. \end{aligned}$$

I ajuntant això amb les definicions de *forward* i *backward* arribem a:

$$\xi_n(x_i, x_j) = \frac{\alpha_n(x_i) c_{x_i x_j} b_{x_j}(o_{n+1}) \beta_{n+1}(x_j)}{\sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_n(x_i) c_{x_i x_j} b_{x_j}(o_{n+1}) \beta_{n+1}(x_j)}.$$

On el denominador es pot calcular directament, tal com hem vist a l'equació (3.2.6), $P(O \mid \lambda) = \sum_{i=1}^{N} \alpha_T(x_i)$, però ho deixem així perquè quedi clar que és el factor de normalització i que $\xi_n(x_i, x_j)$ representa una probabilitat.

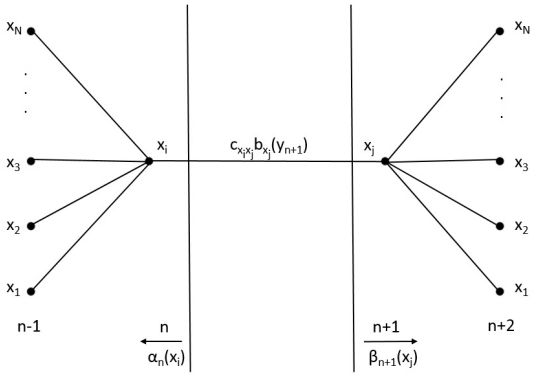

Figura 3.2.3: Representació de la seqüència d'operacions necessàries per calcular ξ, la probabilitat que $X_n = x_i$ i $X_{n+1} = x_j$ donats els paràmetres i l'observació.

Recordem que quan tractàvem el problema de suavitzat hem definit una altra variable, $\gamma_n(x_i)$ (equació (3.2.10)), que representava la probabilitat de trobar-nos a l'estat $x_i \in E$ a l'instant n donada la seqüència observada i els paràmetres del model. Fixem-nos que hi ha una relació directa entre $\gamma_n(x_i)$ i $\xi_n(x_i, x_j)$:

$$\sum_{j=1}^{N} \xi_n(x_i, x_j) = \sum_{j=1}^{N} P(X_n = x_i, X_{n+1} = x_j \mid O, \lambda) = P(X_n = x_i \mid O, \lambda).$$

I, per tant,

$$\gamma_n(x_i) = \sum_{j=1}^{N} \xi_n(x_i, x_j).$$

Fixem-nos, a més a més, que la suma de $\xi_n(x_i, x_j)$ per tot $n < T$ es pot interpretar com el nombre esperat de transicions de l'estat x_i a l'estat x_j. De la mateixa manera,

la mateixa suma sobre $\gamma_n(x_i)$ pot ser interpretada com el nombre de vegades que es passa per l'estat x_i, és a dir, el nombre esperat de transicions que parteixen d'x_i ja que estem excloent l'últim instant del càlcul. Aquestes dues quantitats ens seran molt útils pel nostre objectiu, reestimar els paràmetres:

$$\begin{aligned}\sum_{n=1}^{T-1}\xi_n(x_i,x_j) &= \sum_{n=1}^{T-1} P(X_n = x_i, X_{n+1} = x_j \mid O,\lambda)\\ &= E[\text{nombre de transicions d'}x_i \text{ a } x_j].\\ \sum_{n=1}^{T-1}\gamma_n(i) &= \sum_{n=1}^{T-1} P(X_n = x_i \mid O,\lambda)\\ &= E[\text{nombre de transicions que surten d'}x_i].\end{aligned}$$

Si ara pensem en el significat dels paràmetres, on les entrades de C representen la probabilitat d'anar d'un estat a un altre, $c_{x_ix_j} = P(X_{n+1} = x_j \mid X_n = x_i)$ per tot $x_i, x_j \in E$, i recordem el significat primari de probabilitat, basat en freqüències relatives, podrem reescriure'ls com:

$$\begin{aligned}c^*_{x_ix_j} &= \frac{\text{nombre esperat de transicions d'}x_i \text{ a } x_j}{\text{nombre esperat de transicions des d'}x_i}\\ &= \frac{\sum_{n=1}^{T-1}\xi_n(x_i,x_j)}{\sum_{n=1}^{T-1}\gamma_n(x_i)}.\end{aligned}$$

I d'altra banda, les entrades π_{x_i} de Π no eren més que la probabilitat de tenir $X_1 = x_i$, i, per tant, ho podrem reestimar com:

$$\pi^*_{x_i} = P(X_1 = x_i \mid O,\lambda) = \gamma_1(x_i).$$

Per últim ens falta optimitzar les entrades $b_{x_i}(y)$ de B que representen la probabilitat d'observar $y \in V$ quan ens trobem a l'estat $x_i \in E$. Abans ja hem dit que ens centràvem en el cas de les HMM d'alfabet finit (observació (3.1.3)) i, per tant, també podem reestimar-los "comptant el nombre d'esdeveniments".

$$\begin{aligned}b^*_{x_i}(y) &= \frac{\text{nombre esperat de vegades de tenir l'estat } x_i \text{ i l'observació } y}{\text{nombre esperat de vegades d'estar a l'estat } x_i}\\ &= \frac{\sum_{n=1}^{T}\gamma_n(x_i)\mathbf{1}_{\{Y_n=y\}}}{\sum_{n=1}^{T}\gamma_n(x_i)}.\end{aligned}$$

Tal i com Leonard Baum i els seus companys demostren a l'article [1], si utilitzem els paràmetres $\lambda = (C, B, \Pi)$ per calcular les variables *forward, backward, gamma* i *ksi* utilitzant els procediments ja vistos i amb aquests resultats computem els nous paràmetres, $\lambda^* = (C^*, B^*, \Pi^*)$, hi ha dues situacions possibles:

- El model inicial, λ es troba en un punt crític de la funció de versemblança i aleshores els paràmetres no canvien, λ=λ^*.
- El model λ^* genera la seqüència observada O amb una probabilitat més alta que λ: $P(O \mid \lambda) < P(O \mid \lambda^*)$.

Per tant, amb aquest procediment hem aconseguit uns paràmetres que s'ajusten millor a la nostra observació. Cal remarcar, però, que l'algorisme de Baum-Welch només ens garanteix que arribem a un màxim local però sovint el problema que volem tractar és bastant complex i té diversos màxims locals [7].

Els passos que segueix l'algorisme de Baum-Welch són:

1. Definim un model actual, $\lambda = (C, B, \Pi)$.
2. Utilitzem aquest model per calcular els valors que prenen les variables:

$$\alpha_n(x_i),\ \beta_n(x_i),\ \gamma_n(x_i)\ i\ \xi_n(x_i, x_j),\ \text{per tot } x_i, x_j \in E.$$

3. Recalculem els paràmetres: $\lambda^* = (C^*, B^*, \Pi^*)$.
4. Fem $\lambda = \lambda^*$ i tornem al pas 2 fins que arribem a una condició d'aturada, ja sigui una tolerància o un nombre d'iteracions fixat.

A l'hora de programar els algorismes, cosa que nosaltres farem utilitzant el llenguatge `C++`, cal escalar els paràmetres perquè treballem amb quantitats tan petites que de seguida els valors que prenen les variables excedeixen el rang de precisió i ens donarien 0, no podríem treballar-hi [6].

3.2.4 Escalat

Recuperant la definició de la variable *forward* (equació (3.2.4)) veiem que representa la suma d'un gran nombre de termes però tots ells són el resultat d'un llarg producte de valors inferiors a 1 (sovint molt més petits):

$$\begin{aligned}\alpha_n(x_i) &= P(O^n, X_n = x_i \mid \lambda) \\ &= \sum_{q_1,\dots q_{n-1} \in E^{n-1}} P(O^n, X_n = x_i \mid q_1, \dots, q_{n-1}, \lambda) P(q_1, \dots, q_{n-1} \mid \lambda) \\ &= \sum_{q_1,\dots q_{n-1} \in E^{n-1}} \left[\prod_{s=1}^{n} b_{q_s}(o_s) \prod_{s=1}^{n-1} c_{q_s q_s+1} \right].\end{aligned}$$

Per aconseguir mantenir-nos en el rang de precisió i poder executar els algorismes necessitem escalar $\alpha_n(x_i)$ en cadascun dels passos. Aquest escalat haurà de ser independent de l'estat $x_i \in E$ ja que el que voldrem és comparar les probabilitats dels diferents estats, però si que tindrà una dependència amb l'instant, n. Normalment el que es fa és multiplicar $\alpha_n(x_i)$ per un factor d'escala i aplicar-lo també al càlcul de la variable *backward* que tindrà uns valors comparables a *alpha*.

Per cada instant de temps prendrem com a factor d'escala:

$$f_n = \frac{1}{\sum_{i=1}^{N} \alpha_n(x_i)}.$$

De manera que calculem la variable *forward*, que ara denotarem $\overline{\alpha}_n(x_i)$, modificant lleugerament l'algorisme anterior (equació (3.2.5)):

- Inicialització, cas $n = 1$:

$$\overline{\alpha}_1(x_i) = f_1\alpha_1(x_i) = f_1\pi_{x_i}b_{x_i}(o_1) = \frac{\pi_{x_i}b_{x_i}(o_1)}{\sum_{j=1}^{N}\pi_{x_j}b_{x_j}(o_1)},\ 1 \leq i \leq N.$$

- Iteració, per calcular els cassos $n = 2, 3, \ldots, T-1$ i per tot $x_i \in E$ recursivament:

$$\overline{\alpha}_{n+1}(x_i) = f_{n+1}b_{x_i}(o_{n+1})\sum_{k=1}^{N} c_{x_kx_i}\overline{\alpha}_n(x_k) = \frac{b_{x_i}(o_{n+1})\sum\limits_{k=1}^{N} c_{x_kx_i}\overline{\alpha}_n(x_k)}{\sum_{j=1}^{N} b_{x_j}(o_{n+1})\sum\limits_{k=1}^{N} c_{x_kx_j}\overline{\alpha}_n(x_k)}.$$

Havíem vist que $P(O \mid \lambda) = \sum\limits_{i=1}^{N}\alpha_T(x_i)$ (equació (3.2.3)). Amb $\overline{\alpha}_T(x_i)$ això no serà cert, però podrem recuperar aquesta informació.

Primer de tot, veiem inductivament que es compleix la igualtat

$$\overline{\alpha}_n(x_i) = \left(\prod_{s=1}^{n} f_s\right)\alpha_n(x_i). \tag{3.2.11}$$

1. Cas inicial $n = 1$: es compleix per la definició d'$\overline{\alpha}_1(x_i)$ que és directament $f_1\alpha_1(x_i)$.
2. Suposem que és cert per un n fixat i veiem que això ens porta al mateix resultat pel cas $n+1$. La nostra hipòtesi d'inducció és $\overline{\alpha}_n(x_i) = \left(\prod_s = 1^n f_s\right)\alpha_n(x_i)$ i volem veure que aleshores $\overline{\alpha}_{n+1}(x_i) = \left(\prod_s = 1^{n+1} f_s\right)\alpha_{n+1}(x_i)$.

$$\begin{aligned}
\overline{\alpha}_{n+1}(x_i) &= f_{n+1}b_{x_i}(o_{n+1})\sum_{k=1}^{N} c_{x_kx_i}\overline{\alpha}_n(x_k) \\
&= f_{n+1}b_{x_i}(o_{n+1})\sum_{k=1}^{N} c_{x_kx_i}\left(\prod_{s=1}^{n} f_s\right)\alpha_n(x_k) \\
&= \left(\prod_{s=1}^{n+1} f_s\right)b_{x_i}(o_{n+1})\sum_{k=1}^{N} c_{x_kx_i}\alpha_n(x_k) = \left(\prod_s = 1^{n+1} f_s\right)\alpha_{n+1}(x_i).
\end{aligned}$$

Utilitzant aquesta propietat podem veure una altra relació entre les variables:

$$\begin{aligned}\overline{\alpha}_{n+1}(x_i) &= \frac{b_{x_i}(o_{n+1}) \sum_{k=1}^{N} c_{x_k x_i} \left(\prod_{s=1}^{n} f_s \right) \alpha_n(x_k)}{\sum_{j=1}^{N} b_{x_j}(o_{n+1}) \sum_{k=1}^{N} c_{x_k x_j} \left(\prod_{s=1}^{n} f_s \right) \alpha_n(x_k)} \\ &= \frac{\alpha_n(x_i)}{\sum_{j=1}^{N} \alpha_n(x_i)}.\end{aligned}$$

De manera que, igual que abans, tenim que la suma és 1:

$$\sum_{i=1}^{N} \overline{\alpha}_n(x_i) = \sum_{i=1}^{N} \frac{\alpha_n(x_i)}{\sum_{j=1}^{N} \alpha_n(x_j)} = \frac{\sum_{i=1}^{N} \alpha_n(x_i)}{\sum_{j=1}^{N} \alpha_n(x_j)} = 1. \tag{3.2.12}$$

Ajuntant les propietats que acabem de veure (equació (3.2.11) i (3.2.12)) juntament amb l'equació (3.2.3) pel cas $n = T$ tenim:

$$1 = \sum_{i=1}^{N} \overline{\alpha}_T(x_i) = \sum_{i=1}^{N} \left(\prod_{s=1}^{T} f_s \right) \alpha_T(x_i) = P(O \mid \lambda) \prod_{s=1}^{T} f_s.$$

Així doncs, no som capaços de calcular la probabilitat perquè ens sortim de rang però sí que podrem calcular-ne el logaritme:

$$\log[P(O \mid \lambda)] = -\sum_{s=1}^{T} \log(f_s).$$

Un cop fet això, modifiquem la recursió (equació (3.2.9)) per calcular els valors que pren la variable *backward*. Ho farem aplicant el mateix factor d'escala que al càlcul de *forward* aprofitant que els valors d'ambdues són comparables i així ens facilitaran la resta de càlculs.

- Inicialització, cas $n = T$:

$$\overline{\beta}_T(x_i) = f_T, \ 1 \leq i \leq N.$$

- Iteració, obtenim els cassos $n = T-1, T-2, \ldots, 1$ recursivament:

$$\overline{\beta}_n(x_i) = f_n \sum_{j=1}^{N} c_{x_i x_j} b_{x_j}(o_{n+1}) \overline{\beta}_{n+1}(j), \quad 1 \leq i \leq N.$$

Observació 3.2.4. No cal aplicar sempre els factors d'escala. De fet pels casos on $\sum_{i=1}^{N} \alpha_n(x_i) = 0$ no està definit s'agafa $f_n = 1$ i tot el què hem vist fins ara segueix sent vàlid.

Pel que fa a la resta de variables, γ i ξ no introduirem cap modificació ja que els factors d'escala s'anul·len i no interfereixen, podem calcular-los utilitzant les noves variables $\overline{\alpha}$ i $\overline{\beta}$ sense problema. El mateix passa a l'hora d'aplicar l'algorisme de Baum-Welch.

3.3 Variable-stepsize hidden Markov model

Ja hem ampliat el concepte de cadenes de Markov i gràcies a això hem pogut tractar el cas on els estats no eren observables físics. Ara bé, hi ha un altre tipus de situació que ens planteja un nou repte.

Exemple 3.3.1. Un motor molecular
Els motors moleculars són sistemes formats per una única molècula que es pot trobar en diferents configuracions o estats (no observables) mentre el que nosaltres observem és la posició de la molècula. La posició és una variable que representa l'acumulació d'un nombre aleatori de salts de longitud elemental. A més a més, aquesta posició no som capaços de llegir-la de manera exacta sinó que està afectada per un cert soroll. El què ens interessa és caracteritzar la distribució d'aquests salts alhora que volem descriure el comportament dels estats moleculars (veure fig. (3.3.1)).

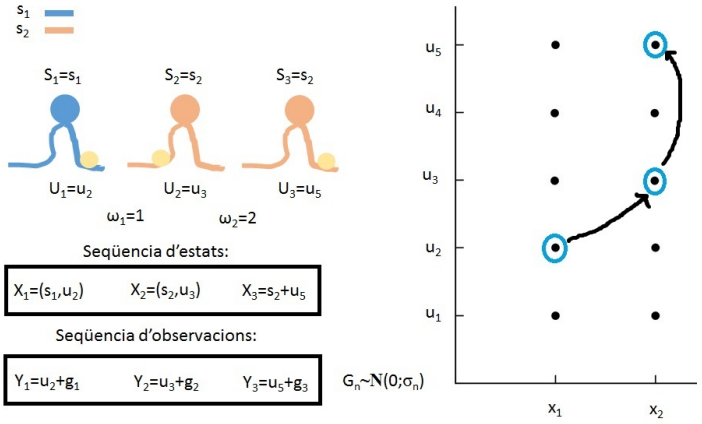

Figura 3.3.1: Representació del sistema format per un motor molecular amb 2 configuracions moleculars diferents, els estats possibles són: $E = \{x_1, x_2\}$, les posicions on es pot trobar són: $\mathcal{U} = \{u_1, u_2, u_3, u_4, u_5\}$ i el nombre de salts entre observacions es correspon amb la diferència entre els índexs $\omega_n = j - i$ si $o_{n+1} = u_j$ i $o_n = u_i$. Els valors que pren Y_n són els observables, les posicions mesurades en cada instant, que es corresponen a la posició real afectada per un soroll Gaussià. Cas concret de l'exemple (3.3.1).

Per tal de tractar casos com aquest, amb un nombre immens d'observables (la posició) que reflecteixen l'acumulació de salts de longitud elemental i alhora un nombre petit d'estats, s'utilitza l'anomenat **variable-stepsize hidden Markov model** (VS-HMM), presentat l'any 2010 en l'article [9] per Fiona E. Müllner i altres, que tracta alhora l'estat molecular i la posició, com un estat compost. Nosaltres l'explicarem amb detall a continuació ajudant-nos de l'exemple (3.3.1) per presentar-ne els elements.

3.3.1 Elements

Encara que el comportament del nostre motor segueixi un procés de temps continu, podem seguir-lo tractat com si fos un model de Markov a temps discret ja que és de

la manera com obtenim les nostres observacions. Com ja hem avançat, la posició del motor també la discretitzarem prenent com a mesura elemental la que més s'adeqüi al sistema que volem modelitzar. En el nostre cas escollirem 1 nm i, per tant, la diferència entre índexs es correspondrà amb la diferència entre observacions expressada en nanòmetres.

Seguint la notació emprada en la secció anterior on $\{X_n \colon n \geq 1\}$ representa la cadena de Markov (amb estats no observables) i $\{Y_n \colon n \geq 1\}$ les observacions, veiem com són els elements d'un VS-HMM i quines petites modificacions hem de fer respecte els elements que defineixen les HMM:

- El estats del model són ara compostos, continguts en el conjunt $E = \mathcal{S} \times \mathcal{U}$, de dimensió $N \times M$, format per (veure fig. (3.3.1)):
  - Les diferents configuracions que pot tenir la molècula, són els valors que pot prendre el procés estocàstic $\{S_n \colon n \geq 0\}$ i estan representats en el conjunt $\mathcal{S} = \{s_1, s_2, \ldots, s_N\}$.
  - Les posicions $\mathcal{U} = \{u_1, u_2, \ldots, u_M\}$ on es pot trobar el motor, que representen els nanòmetres que ens hem desplaçat de l'origen i corresponen als valors que pot prendre el procés $\{U_n \colon n \geq 0\}$.

  Els conjunt d'estats és: $E = \{i_{11} = (s_1, u_1), \ldots, i_{NM} = (s_N, u_M)\}$. Representarem l'estat compost en un instant particular, $X_n = (S_n, U_n) = (s_i, u_j)$ amb la notació abreviada (q_n, r_n) o, altrament, x_n.

- Les observacions són ara una variable continua que seguirem representant per $\{Y_n \colon n \geq 1\}$ però no podrem descriure V de forma extensiva perquè no és un conjunt numerable. El soroll que afecta les nostres mesures es pot modelitzar com un soroll Gaussià de manera que les nostres observacions es corresponen a:
$$Y_n = U_n + G_n.$$
On U_n representa, com ja hem dit, la posició real, i G_n és una variable aleatòria que segueix una distribució normal de mitjana nul·la i desviació estàndard depenent del temps, σ_n. A més a més, podem considerar $\sigma_n = f(\sigma_1)$ (en el cas dels motors moleculars, per exemple, $\sigma_n^2 = \frac{I\sigma_1^2}{I_n}$ on I i I_n són dades que s'obtenen experimentalment i σ_1 s'ha de determinar). Utilitzarem l'abreviació o_n per referir-nos a $Y_n = o_n$, la lectura en l'instant n.

- Les probabilitats de transició de la cadena de Markov $\{X_n \colon n \geq 1\}$ ara afectaran a l'estat compost (s_i, u_j), és a dir, hauran de contemplar la probabilitat d'un canvi d'estat i/o posició. Per introduir aquest canvi de forma eficient introduïm una nova variable aleatòria, ω que pren valors en els enters i representa la longitud del salt. La matriu de transició per aquest model és $C = (c_{s_i s_j}(\omega))_{i,j}$ i els seus elements són, per tot $u_k \in \mathcal{U}$:
$$c_{s_i s_j}(\omega) = P(S_{n+1} = j, U_{n+1} = u_{k+\omega} \mid S_n = s_i, U_n = u_k), \qquad \begin{array}{l} 1 \leq i \leq N, \\ 1 \leq j \leq M. \end{array}$$

Com que C segueix sent una *matriu estocàstica* es complirà:

$$\sum_{j,\omega} c_{s_i s_j}(\omega) = 1.$$

- La distribució de les observacions respecte l'estat on es troba el sistema dependrà únicament de la posició. Ja hem dit que $Y_n = U_n + G_n$ i com $G_n \sim \mathcal{N}(0, \sigma_n)$ podem fer un canvi de variable i obtenir, fàcilment, la probabilitat de mesurar l'observació o_n si ens trobem en la posició u_i en aquest mateix instant:

$$b_n(o_n, u_i) = P(Y_n = o_n \mid U_n = u_i) = \frac{1}{\sqrt{2\pi\sigma_n^2}} e^{\frac{-(o_n - u_i)^2}{2\sigma_n^2}}, \text{ per tot } u_i \in \mathcal{U}. \tag{3.3.1}$$

- Finalment ens queda descriure la distribució inicial que afecta, novament, l'estat compost i representem per $\Pi = (\pi_{s_i u_j})_{ij}$ on els seus elements es corresponen a:

$$\pi_{s_i u_j} = P(S_1 = s_i, U_1 = u_j),\ 1 \leq i \leq N,\ 1 \leq j \leq M.$$

Els paràmetres del model VS-HMM, que denotarem λ igual que en les HMM, estarà format per:

$$\lambda = (C, \sigma_1, \Pi). \tag{3.3.2}$$

Si comparem les equacions (3.1.2) i (3.3.2) veiem que l'única diferència és el canvi de B per σ_1, que vol mostrar que per ajustar els elements de la matriu B només ens cal determinar quin és el valor òptim de σ_1.

Observació 3.3.2. Aquest model també és útil pel cas $N = 1$ quan volem descriure les observacions però es desconeix quins són els estats que hi estan relacionats. En l'exemple (3.3.1) estaríem parlant d'una sola configuració molecular però seguiríem mantenint un valor gran de posicions.

Seguirem representant la seqüència observada de longitud T mitjançant $O = o_1 \cdots o_T$ i la seqüència amagada serà $Q = x_1 \cdots x_T = q_1 r_1 \cdots q_T r_T$. De la mateixa manera, utilitzarem les abreviacions Q^n per referir-nos a la seqüència parcial $x_1 x_2 \cdots x_n$, amb $n \leq T$, Q_m^n per referir-nos a $x_m x_{m+1} \cdots x_n$ i anàlogament O^n i O_m^n.

3.3.2 Resolució dels tres problemes bàsics

El nostre objectiu és trobar resposta a les 3 qüestions que ja hem solucionat a l'apartat anterior però aplicant-ho al VS-HMM. Els algorismes que utilitzarem per fer-ho són els mateixos que abans però els càlculs són lleugerament diferents. No tornarem a veure d'on vénen tots els passos, però si que presentarem les noves variables.

Comencem amb el càlcul de la variable *forward*, que ara representa:

$$\alpha_n(s_i, u_j) = P(O^n, S_n = s_i, U_n = u_j \mid \lambda). \tag{3.3.3}$$

- Pel cas inicial, $n = 1$:

$$\alpha_1(s_i, u_j) = \pi_{s_i u_j} b_1(o_1, u_j),\ 1 \leq i \leq N,\ 1 \leq j \leq M.$$

- Recursivament obtenim els valors per $n = 2, 3 \ldots, T$:

$$\alpha_{n+1}(s_j, u_k) = b_{n+1}(o_{n+1}, u_k) \sum_{i=1}^{N} \sum_{l=1}^{M} \alpha_n(s_i, u_l) c_{s_i s_j}(k-l), \quad \begin{matrix} 1 \leq j \leq N, \\ 1 \leq k \leq M. \end{matrix} \tag{3.3.4}$$

Veiem que la diferència entre les equacions (3.3.4) i la recursió que hem trobat abans, amb equacions (3.2.5), és l'aparició d'un nou sumatori pels càlculs d'α_n amb $n \geq 2$, per tal de contemplar alhora els canvis en posició i configuració.

La mateixa variació serà la que patiran els càlculs de la variable *backward*, que podem reescriure:

$$\beta_n(s_i, u_j) = P(O^{n+1} \mid S_n = s_i, U_n = u_j, \lambda).$$

- Definim el seu valor per $n = T$ igual a 1:

$$\beta_T(s_i, u_j) = 1,\ 1 \leq i \leq N,\ 1 \leq j \leq M.$$

- Recursivament obtenim els valors per $n = T-1, T-2, \ldots, 1$:

$$\beta_{n+1}(s_i, u_k) = \sum_{j=1}^{N} \sum_{l=1}^{M} c_{s_i s_j}(l-k) b_{n+1}(o_{n+1}, u_l) \beta_{n+1}(s_j, u_l), \quad \begin{matrix} 1 \leq j \leq N \\ 1 \leq k \leq M \end{matrix} \tag{3.3.5}$$

Com ja havíem avançat, l'única diferència entre els càlculs corresponents a les HMM (equacions (3.2.9)) i els de les VS-HMM (equacions (3.3.5)) és l'aparició d'un sumatori pels càlculs amb $n \leq T-1$ per tal d'afegir la dependència amb la posició.

Observació 3.3.3. No hem aplicat aquí l'escalat de les variables per simplificar la notació però s'aplicarien exactament igual que a les HMM.

La variable *gamma* es calcularà exactament igual que abans, utilitzant les noves $\alpha_n(s_i, u_j)$ i $\beta_n(s_i, u_j)$:

$$\gamma_n(s_i, u_j) = P(S_n = s_i, U_n = u_j \mid O, \lambda) = \frac{\alpha_n(s_i, u_j)\beta_n(s_i, u_j)}{P(O \mid \lambda)}.$$

La probabilitat d'obtenir una observació determinada, $O = o_1, \ldots, o_n$, es pot calcular a partir dels valors de la variable *forward*, com hem vist abans (equació (3.2.6)):

$$P(O \mid \lambda) = \sum_{i_{kl} \in E} \alpha_T(i_{kl}) = \sum_{k,l} \alpha_T(s_k, u_l).$$

I per últim ens queda ξ, la variable que representava la probabilitat de fer una transició determinada en un instant donat, conegudes les observacions i els paràmetres del model. Aquesta serà:

$$\begin{aligned}\xi_n(s_i, s_j, \omega) &= P(S_n = s_i, S_{n+1} = s_j, U_{n+1} = U_n + \omega \mid O, \lambda) \\ &= \sum_{k=1}^{N} P(S_n = s_i, S_{n+1} = s_j, U_n = u_k, U_{n+1} = u_{k+\omega} \mid O, \lambda) \\ &= \frac{\sum_{k=1}^{N} \alpha_n(s_i, u_k) c_{s_i s_j}(\omega) b_{n+1}(o_{n+1}, u_{k+\omega}) \beta_{n+1}(x_j, u_{k+\omega})}{P(O \mid \lambda)}.\end{aligned}$$

I els algorismes de *Viterbi* i *Baum-Welch* ens serviran igualment si apliquem unes lleugeres modificacions.

Comencem amb la reestimació de la distribució inicial, $\pi_{s_i u_j}$, amb $s_i \in \mathcal{S}, u_j \in \mathcal{U}$. Novament es correspon exactament al càlcul de la nostra variable *gamma* en l'instant $n = 1$, per tant:

$$\pi^*_{s_i u_j} = P(S_1 = s_i, U_1 = u_j \mid O, \lambda) = \gamma_1(s_i, u_j).$$

D'altra banda, fixem-nos que la suma per tot n de $\xi_n(s_i, s_j, \omega)$ representa el nombre esperat de transicions entre les configuracions s_i i s_j amb un desplaçament de longitud ω i si fem córrer la suma per tot n, j *i* ω el que tenim és el nombre esperat de transicions que parteixen de la configuració s_i:

$$\begin{aligned}\sum_{n=1}^{T-1} \xi_n(s_i, s_j, \omega) &= \sum_{n=1}^{T-1} P(S_n = s_i, S_{n+1} = s_j, U_{n+1} = U_n + \omega \mid O, \lambda)) \\ &= E[\text{no. de transicions d'}s_i \text{ a } s_j \text{ amb un desplaçament } \omega].\end{aligned}$$

$$\begin{aligned}\sum_{n<T,j,\omega} \xi_n(s_i, s_j, \omega) &= \sum_{n<T,j,\omega} P(S_n = s_i, S_{n+1} = s_j, U_{n+1} = U_n + \omega \mid O, \lambda) \\ &= E[\text{no. de transicions que surten d'}s_i].\end{aligned}$$

Similarment al cas de les HMM, els paràmetres $c_{s_i s_j}(\omega)$ de C representen la probabilitat d'anar de l'estat compost (s_i, u_k) a l'estat $(s_j, u_{k+\omega})$ i els podrem reescriure com:

$$\begin{aligned}c^*_{s_i s_j}(\omega) &= \frac{\text{nombre esperat de transicions d'}s_i \text{ a } s_j \text{ amb un salt de longitud } \omega}{\text{nombre esperat de transicions des d'}s_i} \\ &= \frac{\sum_{n=1}^{T-1} \xi_n(s_i, s_j, \omega)}{\sum_{n<T,l,\omega} \xi_n(s_i, s_l, \omega)}.\end{aligned}$$

L'últim paràmetre que ens falta reestimar és σ_1. Recordem que σ és la desviació estàndard del soroll, és a dir, $\sigma_n^2 = E[(Y_n - U_n)^2]$. Però com que el valor que pren Y_n és la nostra observació i això és conegut, l'esperança es converteix en una esperança

condicionada, $\sigma_n^2 = E[(Y_n - U_n)^2 \mid Y_n = o_n]$ que podem reestimar, aplicant la definició (1.0.12), de la següent manera:

$$\begin{aligned}\sigma_n^{*^2} &= \frac{E[(Y_n - U_n)^2 \mid Y_n = o_n]}{=} \sum_{j=1}^{M}(o_n - u_j)^2 P(U_n = u_j \mid Y_n = o_n) \\ &= \sum_{j=1}^{M}(o_n - u_j)^2 \sum_{i=1}^{N} P(S_n = s_i, U_n = u_j \mid Y_n = o_n) \\ &= \sum_{j=1}^{M}\sum_{i=1}^{N}(o_n - u_j)^2 \gamma_n(s_i, u_j).\end{aligned} \tag{3.3.6}$$

En el cas concret del motor molecular teníem que $\sigma_n^2 = \frac{I\sigma_1^2}{I_n}$. Això ens permet fer un ajust més acurat de σ_1 calculant-la promitjant el resultat obtingut per cada instant de temps:

$$\sigma_1^{*^2} = \frac{1}{T}\sum_{n=1}^{T}\frac{I_n}{I}\sigma_n^{*^2} = \frac{1}{T}\sum_{n=1}^{T}\frac{I_n}{I}\sum_{j=1}^{M}\sum_{i=1}^{N}(o_n - u_j)^2\gamma_n(s_i, u_j). \tag{3.3.7}$$

Així doncs, per reestimar els paràmetres utilitzarem els mateixos passos que en les HMM però amb les equacions trobades en aquesta secció:

1. Definim un model actual, $\lambda = (C, \sigma_1, \Pi)$.
2. Utilitzem aquest model per calcular les diferents variables:

$$\alpha_n(s_i, u_j),\ \beta_n(s_i, u_j),\ \gamma_n(s_i, u_j) \text{ i } \xi_n(s_i, s_j, \omega).$$

3. Recalculem els paràmetres: $\lambda^* = (C^*, \sigma_1^*, \Pi^*)$.
4. Fem $\lambda = \lambda^*$ i tornem al pas 2 fins que arribem a una condició d'aturada.

Passem ara a reformular l'*algorisme de Viterbi* que ens ajudarà a buscar la seqüència d'estats que millor s'adapta a les observacions. Com veurem, els passos seran els mateixos que en les HMM però maximitzarem pels dos índexs, corresponents a configuració i posició, i obtindrem la seqüència corresponent als estats compostos:

- Inicialització, cas $n = 1$:

$$\begin{aligned}\delta_1(s_i, u_j) &= \log[\pi_{s_i u_j} b_1(o_1, u_j)], \quad 1 \le i \le N,\ 1 \le j \le M. \\ \phi_1(s_i) &= 0, \qquad\qquad\qquad\qquad\quad 1 \le i \le N.\end{aligned}$$

- Recursió per calcular els casos $n = 2, 3, \ldots, T$:

$$\begin{aligned}
&\delta_m(s_j,u_k) = \text{màx}_{i,l}\{\delta_{m-1}(s_i,u_l) + log[c_{s_is_j}(k-l)b_m(o_m,u_k)]\}, && 1\le j\le N\\
&&& 1\le k\le M\\
&\phi_m(s_j,u_k) = \text{argmàx}_{i,l}\{\delta_{m-1}(s_i,u_l)c_{s_is_j}(k-l)b_m(o_m,u_k)\}, && 1\le j\le N\\
&&& 1\le k\le M.
\end{aligned}$$

D'aquesta manera trobarem l'estat $\hat{x}_n = (\hat{s}_n, \hat{r}_n)$ així com la puntuació que li correspon, el logaritme de la probabilitat de la seqüència òptima, P^*:

$$\begin{aligned}
log[P^*] &= \underset{i,l}{\text{màx}}\{\delta_n(s_i,u_l)\},\\
\hat{x}_T &= (\hat{s}_T,\hat{r}_T) = \underset{i,l}{\text{argmàx}}\{\delta_T(s_i,u_l)\}.
\end{aligned}$$

I podrem reconstruir la seqüència completa d'estats més probable desfent el camí que hem fet:

$$\hat{x}_m = (\hat{s}_m,\hat{r}_m) = \phi_{m+1}(\hat{s}_{m+1},\hat{r}_{m+1}),\ m = n-1, n-2,\ldots,1.$$

Reescrivint les equacions que formen part de l'algorisme de Viterbi hem vist quina forma prenen els algorismes de forward-backward, Viterbi i Baum Welch en el cas de les VS-HMM i per tant, som capaços d'ajustar els paràmetres d'aquest model i trobar tant la versemblança com la seqüència més probable donada la seqüència observada.

Capítol 4

Aplicació

Ja hem parlat a la introducció de l'ampli ventall de camps on s'utilitzen els models de Markov ocults, anant des del reconeixement de veu fins a la biologia molecular computacional passant per la modelització del curs de l'esclerosi múltiple [7, 6], entre altres. Com no podia ser d'una altra manera nosaltres també volíem que aquest treball tingués una part aplicada on veure com treballen els algorismes que hem anat explicant. Per veure-ho aplicarem el model VS-HMM, que els inclou una mica tots, a dades que van ser obtingudes de forma experimental en el procés de desenrotllament de l'ADN utilitzant pinces òptiques. Per aquest motiu creiem oportú fer una introducció on expliquem en què consisteixen aquest tipus d'experiments.

4.1 Experiments amb pinces òptiques

Amb les pinces òptiques podem realitzar experiments amb forces de pocs piconewtons (pN) i energies molt baixes de manera que mesurant la resposta mecànica de les biomolècules som capaços de determinar-ne l'energia lliure i les velocitat de reacció amb bastant precisió [11]. Nosaltres ens centrarem en un experiment concret, l'experiment d'*unzipping*, aplicat a molècules d'ADN però tot el que explicarem a continuació pot ser aplicat, exactament igual, a molècules d'ARN.

Recordem que tant l'ADN com l'ARN són àcids nucleics, molècules formades per la combinació de nucleòtids. Una cadena d'ADN està composta dels nucleòtids actina (A), timina (T), citocina (C) i guanina (G) que s'uneixen formant una llarga seqüència mitjançant interaccions de Wan Der Waals. A més a més, l'actina i la timina per una banda i la citosina i la guanina per l'altra són complementaries, tendint a unir-se mitjançant ponts d'hidrogen i generant-ne l'estructura tridimensional. El cas de l'ARN és similar però substituint la timina per l'uracil (U).

La molècula d'ADN que nosaltres tractarem tindrà estructura de *hairpin* (o forquilla) que vol dir que serà una cadena simple on les primeres n bases són complementàries a les últimes n bases llegides en ordre invers de manera que s'uneixen mitjançant ponts d'hidrogen formant una estructura similar a una forquilla de cabells (veure figura (4.1.1)). A la part central de la cadena hi ha unes quantes bases

més que no són complementaries entre si i formen una regió anomenada bucle [11]. Aquesta estructura la trobem en molècules d'ADN i ARN tant *in vivo* com *in vitro* i el motiu pel qual s'utilitza aquest tipus d'estructura és perquè permet fer, desfer i repetir els experiments amb la mateixa molècula una i altra vegada.

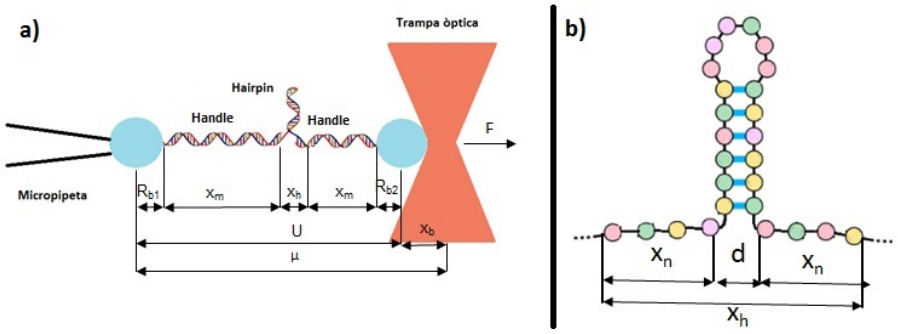

Figura 4.1.1: a) Esquema, no a escala, del dispositiu experimental. S'inclouen els paràmetres configuracionals: R_{b1}, R_{b2}, x_h, x_m, *i* x_b, la projecció al llarg de l'eix x de cadascun dels elements. A més a més, la interacció amb la trampa òptica ens permet mesurar la força $f = k_b x_b$, on k_b és una constant. b) Ampliació de la molècula amb estructura de hairpin on desglossem la seva projecció sobre l'eix x: $x_h = 2x_n + d$.

Tal com s'explica en l'article [8] el dispositiu està format pel *hairpin* d'ADN, dues cadenes dobles d'ADN (dsDNA) anomenades *handles* (mànecs) i dues esferes dielèctriques. Cadascun dels extrems de la molècula és unit a un dels mànecs i cada *handle* s'unirà a una perla dielèctrica. La trampa òptica de potencial $V_b(x)$ generada per rajos làser capturarà una de les boles mentre l'altra perla es succionarà a la punta d'una micropipeta que considerarem immòbil (veure figura (4.1.1)). Designem utilitzant les lletres R_{b1} i R_{b2} els radis de les perles, x_m és l'extensió corresponent als mànecs, d la distància entre els extrems de la forquilla d'ADN, x_n la longitud de cadena simple d'ADN (ssDNA) que s'ha desenrotllat i x_b la posició de la pipeta respecte a la bola dielèctrica més propera (totes aquestes longituds es refereixen a la projecció de l'element en qüestió sobre l'eix x).

Podem mesurar la distància entre els extrems:

$$\mu(F, n) = 2x_m(F) + 2x_n(F, n) + d + x_b(F) + R_{b1} + R_{b2} \tag{4.1.1}$$

A més a més la trampa òptica es pot considerar un potencial harmònic, amb potencial $V_b(x_b) = \frac{1}{2}k_b x_b^2$ i força $F = k_b x_b$, on k_b és la rigidesa de la trampa òptica [8]. És a dir, podem tractar la trampa òptica com si fos una molla de constant recuperadora k_b i x_b es correspon a l'elongació d'aquesta molla imaginària.

Les diferents contribucions a l'equació (4.1.1) es poden obtenir utilitzant models elàstics per biopolímers, tal com explica [3]:

1. R_{b1} i R_{b2} són constants.

2. Pel que fa a la longitud del *hairpin* d'ADN considerem $d = 0.59\,\text{nm}$ i tractem la ssDNA corresponent a l'ADN alliberat com una cadena lliurement articulada

(*freely jointed chain*, representat per FJC), la representació més simple de la conformació de polímers. Aquest model considera que la cadena està formada per subunitats rígides d'idèntica longitud unides per frontisses perfectament flexibles.

3. Per modelitzar les manilles, x_m, assumim un model elàstic de cadena de cuc (*worm like chain*, WLC). Aquí les dsDNA es tracten com un cos elàstic continu, descrivint la seva configuració com una funció del vector posició i la longitud de contorn [12].

4. Com ja hem dit considerem la trampa òptica un potencial harmònic amb una rigidesa $k_b = 0,066\,\frac{\text{pN}}{\text{nm}}$ que ens dóna el valor d'x_b utilitzant la relació $x_b = \frac{f}{k_b}$.

Ara que ja hem descrit els elements que formen el dispositiu podem passar a explicar en què consisteix l'experiment d'*unzipping*.

4.1.1 Experiment d'*unzipping*, tracció

Les dades que volem analitzar van ser obtingudes en l'experiment d'*unzipping* realitzat a 3 forquilles d'ADN curtes, d'una longitud de 490 parells de bases (pb).

L'*unzipping* és un experiment de tracció que consisteix a moure la trampa òptica al llarg de l'eix x a una velocitat constant v, generant una tracció per cadascun dels extrems del *hairpin*. A mesura que es van trencant els ponts d'hidrogen que mantenen el *hairpin* unit es van separant parells de bases complementàries i es va alliberant la cadena simple corresponent (vegeu la figura (4.1.2)).

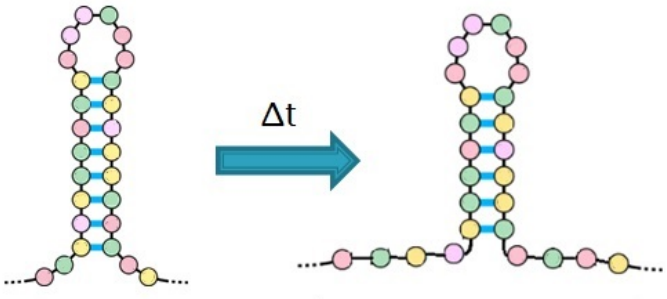

Figura 4.1.2: A mesura que anem desplaçant la trampa òptica el hairpin es va obrint, augmentant la distància entre micropipeta i trampa òptica degut a l'alliberament de cadena simple.

D'aquesta manera l'extensió molecular, U, va augmentant amb el temps, a mesura que es van alliberant parells de bases. Representant les corbes de força en funció de la distància (FDC) obtenim un patró en dent de serra molt característic al voltant dels 15 pN (veure figura (4.1.3)) que ens permet caracteritzar diferents aspectes de la seqüència, com ja hem vist a la introducció.

Com que en el rang de forces utilitzades en l'experiment d'*unzipping* la longitud de les *handles*, x_m, es manté pràcticament constant i el què ens interessarà és

mesurar variacions de longitud en funció del temps treballarem amb $\mu - x_b$ (veure equació (4.1.1)).

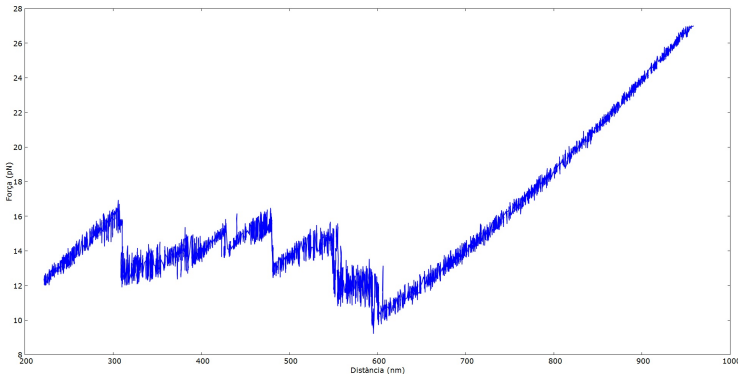

Figura 4.1.3: Diagrama FDC obtingut representant la força en funció de $\mu - x_b$, les dades que utilitzarem en els següents apartats. Es pot veure el patró en dent de serra característic al voltant dels 15 pN.

4.2 Aplicació: determinar la longitud dels salts en l'*unzipping*

Ja hem vist en què consisteix l'experiment d'*unzipping* i quin tipus de dades en podem extreure. El que ens interessa ara és obtenir informació d'aquestes dades.

Hem dit que a mesura que augmenta la força aplicada es van obrint parells de bases. Amb això podem determinar, per exemple, la seqüència de l'ADN o les zones on enzims i proteïnes tendeixen a unir-se a la nostra molècula [3]. Ara bé, hi ha una petita complicació i és que trobem regions de cooperació d'enrotllament-desenrotllament (*cooperative unzipping-zipping regions*, CUR). Aquestes regions són conjunts de bases que s'uneixen o separen de forma conjunta, actuen com un tot: o se separen/ajunten totes alhora o no fan res. A més a més, les nostres lectures estan afectades per fluctuacions, no podem mesurar directament la distància real.

El nostre objectiu és trobar la longitud d'aquestes regions, CUR, tractant el problema com un VS-HMM i utilitzant l'algorisme de Baum-Welch modificat per tal de reestimar-ne els paràmetres i trobar, en particular, la distribució C que millor s'hi adapta.

Per tal de fer això, hem programat els algorismes de *forward-backward*, de *Baum-Welch* i de *Viterbi* pel cas de les *variable stepsize* HMM en llenguatge `C++` i els hem aplicat a les 3 seqüències obtingudes en realitzar l'experiment d'*unzipping* als *hairpins* d'ADN.

4.2.1 Modelització i paràmetres inicials

La primera aproximació que fem és considerar que la longitud dels salts és independent del tipus de bases adjacents (A, T, C o G) i, per tant, ens quedem amb una

VS-HMM d'un sol estat, $N = 1$, i múltiples posicions, $M > 1$. Ara les posicions es refereixen a la distància de la perla dielèctrica unida a la trampa òptica respecte la micropipeta.

Les dades que tenim es corresponen a mesures de la distància que hi ha entre les dues perles on ja s'ha corregit l'efecte de la trampa òptica ($\mu - x_b$ de la figura (4.1.1)). Hem vist a la secció anterior que la resta d'ítems de l'equació (4.1.1) poden considerar-se constants, qualsevol modificació que puguin introduir serà molt petita i la considerarem englobada dins el soroll mesurat, per tant, no introduirem més canvis i treballarem amb aquestes dades directament.

Cal tenir en compte que les observacions estan afectades per fluctuacions, soroll. El modelitzarem utilitzant la variable $\{G_n : n \geq 1\}$ que seguirà una distribució $\mathcal{N}(0, \sigma^2)$. En aquest cas, com treballem amb molècules d'ADN curtes, podem considerar que la variància es manté constant al llarg del temps. Utilitzant el resultat de l'equació (3.3.6) podem reestimar el valor de la variància de forma similar al cas dels motors moleculars (equació (3.3.7)):

$$\sigma^{*^2} = \frac{1}{T}\sum_{n=1}^{T}\sum_{i=1}^{N}(o_n - u_i)^2\gamma_n(u_i),$$

Si seguim amb la notació del capítol anterior, tindrem que la nostra observació és $Y_n = U_n + G_n$ on U_n representa la posició "real". La probabilitat d'obtenir l'observació o_n si ens trobem a la posició u_i serà:

$$b_n(o_n, u_i) = P(Y_n = o_n \mid U_n = u_i) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(o_n - u_i)^2}{2\sigma^2}}.$$

per totes les posicions u_i tal que $|u_i - o_n| \leq 15$ però la considerarem nul·la sempre que la distància sigui superior. Això ens delimita el nombre d'elements del conjunt d'estats que ara està format per les posicions permeses.

La longitud d'un parell de bases és aproximadament d'1 nm, per aquest motiu considerem la unitat elemental de salt d'$1nm$ i el conjunt de posicions:

$$\mathcal{U} = \{u_1 = y_{\text{mín}} - 15, u_2 = y_{\text{mín}} - 14, \ldots, u_{N-1} = y_{\text{màx}} + 14, u_N = y_{\text{màx}} + 15\}$$

amb $y_{\text{mín}} = \min_n([o_n])$ i $y_{\text{màx}} = \max_n([o_n])$ (on $[x]$ vol dir arrodonir x).

També ens serà útil a l'hora de fer els càlculs definir el conjunt que conté tots els salts permesos

$$\mathcal{W} = \{-W, \ldots, 0, \ldots, W\},$$

on $W = \max_n |[o_n] - [o_{n+1}]| + 30 + 1$.

Ja només ens falta escollir uns paràmetres inicials per tal de començar el procés. Ho farem basant-nos en els resultats obtinguts a l'article publicat pel Josep Maria Huguet [4] on determina la longitud de les zones cooperatives CUR, el número de bases que hi estan implicades, mitjançant una aproximació bayesiana. Ell troba que la distribució de salts segueix una llei potencial amb caiguda:

$$P(n) = An^{-B}exp\left(-\left(\frac{n}{n_c}\right)^C\right), \tag{4.2.1}$$

on $P(n)$ representa la probabilitat d'observar una zona CUR d'n parells de bases i les constants A, B, C i n_c són paràmetres positius que cal ajustar i variaran en funció de la longitud del hairpin que estiguem tractan. Per fer-nos-en una idea, pel cas d'un hairpin de 2.2 kbp tenim $A = 0.058,\ B = 0.42,\ C = 2.95$ i $n_c = 69$ mentre que si realitzem l'ajust amb una molècula d'ADN més llarga, de 6.8 kbp, els valors que s'obtenen són: $A = 0.050,\ B = 0.43,\ C = 3.0$ i $n_c = 91$.

Considerant l'equació (4.2.1), però tenint en compte que nosaltres estem parlant de longitud de salt (considerant 1 nm per parell de base) i que esperem anar cap endavant amb més probabilitat que cap endarrere però seguint una distribució proporcional, prenem els següents paràmetres inicials (veure figura (4.2.1)):

$$\begin{aligned} &\pi_{u_i} = \tfrac{1}{115 - y_{\text{mín}}}, && i < 100, \\ &c(\omega) = p * 0.056|\omega|^{-0.4} exp\left(-\left(\tfrac{|\omega|}{60}\right)^3\right), && p = 0.2 \text{ si } \omega < 0, \\ & && p = 0.8 \text{ si } \omega > 0, \\ &c(0) = 1 - \sum_{\omega \neq 0} C(\omega), \\ &\sigma^2 = 12\,\text{nm}^2. \end{aligned} \tag{4.2.2}$$

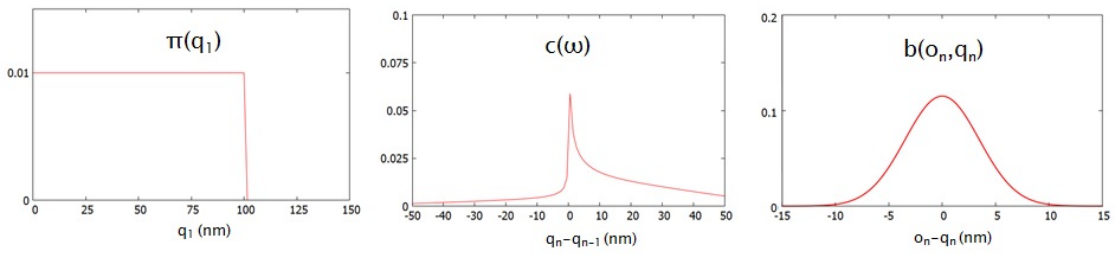

Figura 4.2.1: Representació gràfica de la distribució inicial, Π, les probabilitats de transició, C, i la distribució de les observacions, B, que prenem com a paràmetres inicials en el nostre model d'VS-HMM (equació (4.2.2)).

El nostre objectiu és optimitzar els valors que pren la matriu de transició, C, per tal de saber la probabilitat que en el procés de desenrotllament es realitzi un salt de longitud ω. Per fer-ho apliquem l'algorisme de *Baum-Welch*, per separat però amb els mateixos paràmetres inicials (equacions (4.2.2)), a les seqüències obtingudes en l'*unzipping* de tres molècules d'ADN diferents però d'igual longitud. La nostra condició d'aturada serà quan haguem reestimat els paràmetres 500 vegades, és a dir, quan arribem a 500 iteracions. També aprofitem per calcular $\log(P(O \mid \lambda))$ per veure si cada vegada s'ajusta més. Finalment, després de fer l'última iteració, apliquem l'algorisme de *Viterbi* per comparar la seqüència obtinguda amb les observacions inicials.

4.2.2 Resultats

Efectivament, un cop finalitzat el procés, i tal com mostren els gràfics representats a la figura (4.2.2), veiem que a cada iteració l'observació obtinguda i els paràmetres

estimats encaixen millor. El valor que pren $\log(P(O \mid \lambda))$ augmenta i per tant també ho fa la probabilitat condicionada. Tanmateix, encara que al principi creix ràpidament de seguida estabilitza al voltant d'una valor molt baix. Cal tenir en compte que estem analitzant seqüències de l'orde de 12000 observacions i estem considerant molts salts possibles, hi ha moltes variables en joc.

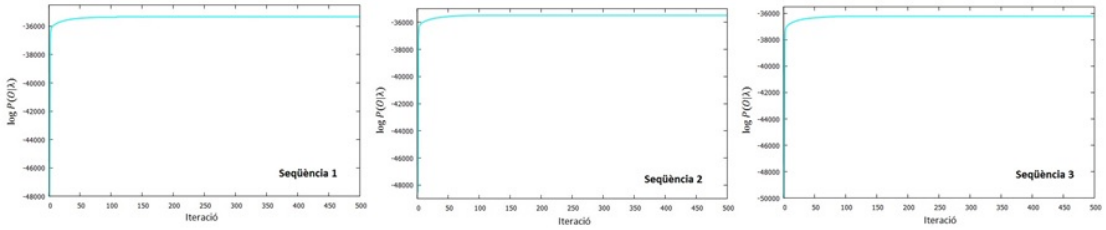

Figura 4.2.2: Representació gràfica de $\log(P(O \mid \lambda))$ en funció de la iteració, utilitzant els paràmetres λ que acabem d'estimar. Veiem els resultats corresponents a cadascuna de les 3 seqüències observades per separat.

Si ens fixem a continuació en la distribució de salts obtinguda veiem que la probabilitat que no hi hagi moviment, que el "salt"sigui de longitud 0, és molt i molt gran, de l'ordre de 0.9. La nostra interpretació d'aquest fet és que s'ajunten 3 factors. Per una banda estem realitzant força mesures abans que s'hagi produït un salt. A això s'afegeixen les oscil·lacions endavant i endarrere d'abans que el trencament de l'enllaç entre bases complementàries sigui estable que es consideren soroll. Per últim hi ha el fet que quan el *hairpin* s'ha acabat d'obrir seguim forçant una extensió bastant més petita que quan s'obren un parell de bases i, per tant, augmentarà el nombre de salts de longitud 0 nm observats.

Oblidant el cas on no es produeix cap salt, fixem-nos en com es comporten la resta de longituds, com es va obrint la molècula (veure figura 4.2.3). Efectivament, aquesta distribució depèn de la seqüència observada però en els tres casos trobem una estructura similar. La longitud que té una probabilitat més elevada és 1 nm per anar disminuint lentament a mesura que augmentem l'extensió del salt (ja sigui endavant o endarrere) i acabar-se anul·lant més enllà dels 20-25 nm. A més a més, veiem que si exceptuem els salts d'1 nm no hi ha una preferència clara per anar endavant sinó que la distribució és més o menys simètrica. Això ens porta a dues opcions possibles: o que les oscil·lacions més petites quedin anul·lades perquè el programa les considera soroll i per aquest motiu no les detectem o que les bases que se separen individualment no acostumin a tornar enrere mentre que les zones CUR, amb més bases implicades i una força d'unió més gran, tendeixin a patir més oscil·lacions abans de separar-se per complet. També veiem que els salts es centren a l'entorn d'uns nuclis deixant nul·la la probabilitat de certs valors d'entremig, segurament degut a la precisió del nostre muntatge que no ens permet fer distincions més acurades.

Pel que fa al càlcul de la variància veiem que també estabilitza ràpidament després d'un nombre petit d'iteracions (figura (4.2.4)).

El valor obtingut ha estat proper als 10 nm^2 en les 3 seqüències cosa que ens fa pensar que la nostra elecció de permetre observacions distanciades fins a 15

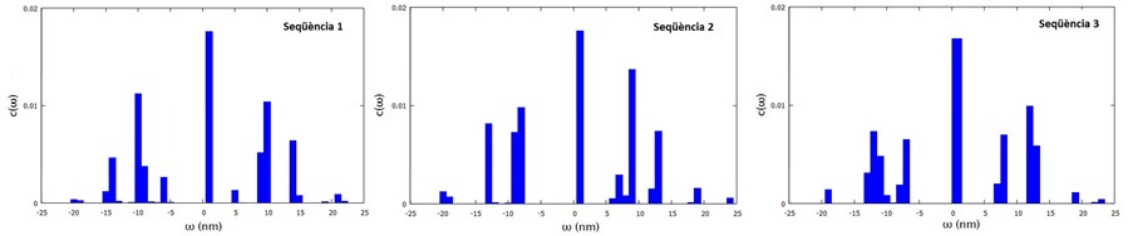

Figura 4.2.3: Representació gràfica de les $c(\omega)$, excloent el cas $\omega = 0$, obtingudes en acabar el procés, després de 500 iteracions. Recordem que $c(\omega)$ és la probabilitat que es produeixi un salt de longitud ω, $c(\omega) = P(U_{n+1} = u_k + \omega \mid U_n = u_k)$ per tota posició possible $u_k \in \mathcal{U}$.

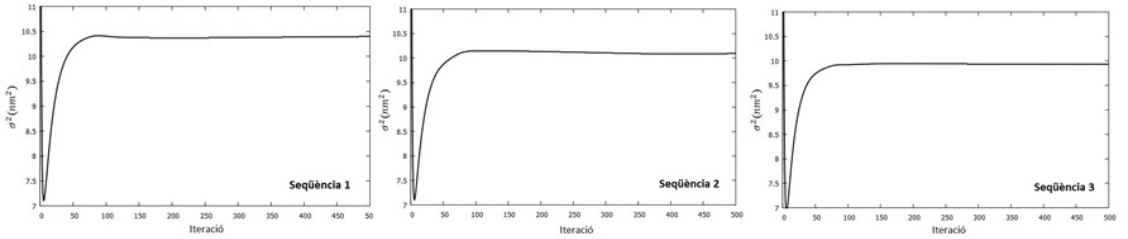

Figura 4.2.4: Representació gràfica de l'evolució de la variància obtinguda per cadascuna de les seqüències en funció de la iteració.

nm de la posició ha estat adequada. Si assumim una variància de $10.5\,\mathrm{nm}^2$, el 99.79 % es trobaran separades menys de 10 nm i el 99.9996 % satisfaran la condició $|o_n - q_n| < 15\,\mathrm{nm}$.

Per últim, representem gràficament (a la figura (4.2.5)) la seqüència que hem obtingut aplicant l'algorisme de Viterbi després d'haver finalitzat l'algorisme de Baum-Welch juntament amb l'observació inicial. D'aquesta manera podrem comparar la seqüència més probable donats els paràmetres estimats amb la seqüència que havíem observat en el nostre dispositiu experimental.

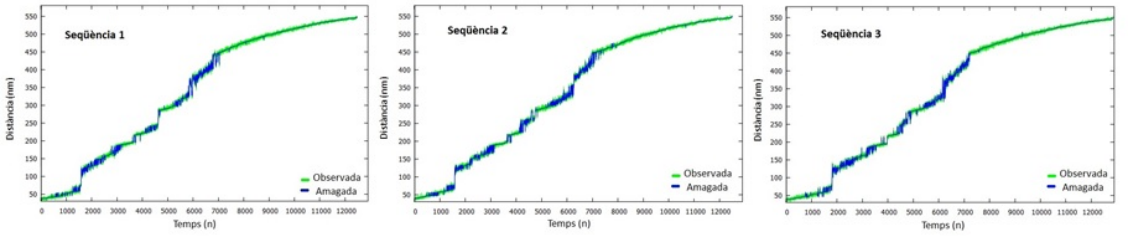

Figura 4.2.5: Representació de la posició (extensió) en funció del temps per les seqüències observada (en verd) i calculada (en blau) que representen respectivament les observacions i la seqüència amagada, la seqüència d'estats.

Veiem que tal i com esperàvem obtenim una seqüència similar a l'observada però sense tantes fluctuacions, més suavitzada. Això passa sobretot per n de l'orde de 7000 i superiors que és quan el nostre *hairpin* s'ha acabat d'obrir. El que observem a partir d'aquí és una elongació més o menys constant que forcem en no deixar de

tibar la trampa òptica però ja no hi ha salts bruscos perquè s'ha desfet l'estructura de *hairpin* i ja no es separen els parells de bases complementaris.

Si tornem a aplicar l'algorisme considerant les primeres 7500 lectures i oblidant-nos de la resta (veure figura (4.2.6)) el que observem és que el salt d'1 nm perd importància i, encara que segueix tenint una de les probabilitats més altes n'hi ha altres de comparables. En general, totes les probabilitats augmenten una mica fent disminuir la probabilitat que no hi hagi salt lleugerament encara que no es nota gaire perquè parlem d'uns valors francament petits.

Pel que fa a la variància és una mica més alta, de l'orde de 12 nm^2, però el límit de 15 nm segueix semblant raonable.

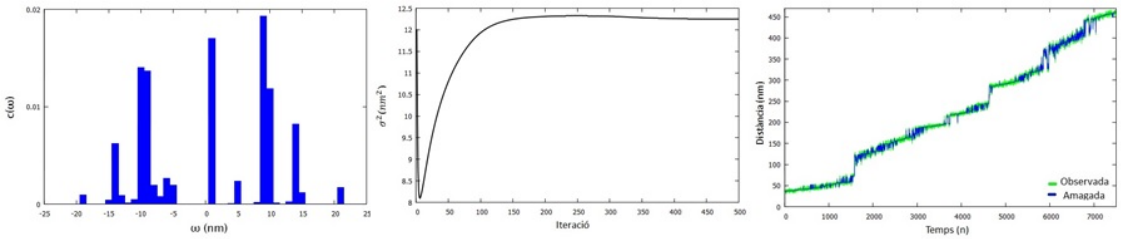

Figura 4.2.6: Representació de $c(\omega)$, σ^2 i les seqüències observada i amagada considerant les primeres 7500 observacions de la seqüència 1.

El motiu pel qual hem realitzat pràcticament tot l'anàlisi amb la seqüència completa i no hem exclòs les darreres observacions és perquè creiem que les últimes mesures són importants per tal de fer un bon ajust del soroll. Aquestes dades segueixen formant part de l'experiment i les fluctuacions que s'hi observen són degudes al soroll, ja no hi ha obertura del *hairpin* perquè les dues cadenes estan completament separades.

4.2.3 Anàlisi de resultats

Guiant-nos per estudis similars [4] esperàvem trobar probabilitats elevades per salts petits que disminueixen ràpidament a mesura que considerem desplegaments de longitud superior. No obtenim ben bé això sinó les distribucions de la figura (4.2.5) però si que recuperem la preferència pel desplegament individual (1 pb) i veiem com la probabilitat decreix ràpidament anul·lant-se per salts superiors als 25 nm.
A priori ens podria sorprendre l'elevada probabilitat dels salts d'uns 10 nm, segons en el rànquing. Ara bé, si tenim en compte les limitacions del dispositiu experimental que, com ja explica [4], no ens permet discriminar fàcilment les zones CUR de menys de 10 parells de bases veiem que moltes de les transicions que tenen una longitud inferior a 10 nm no seran detectades.
Finalment cal tenir en compte que la desviació estàndard és força gran i dóna lloc a la possibilitat que algunes transicions hagin estat considerades soroll. Això podria generar l'agrupació de longituds permeses a l'entorn dels valors amb probabilitat més elevada i anul·lar-ne la resta. Per aquest motiu no podem afirmar que hàgim recuperat els valors reals però sí que en podem veure la tendència.

Encara que els resultats siguin comparables amb els obtinguts en estudis similars seria necessari seguir avançant en aquesta línia per poder-los confirmar ja que nosaltres n'hem fet un plantejament diferent. Per aquest motiu creiem que seria interessant comparar els resultats generats amb els que s'obtindrien aplicant petites variacions.
Una variació que ens podria aportar informació rellevant seria repetir els càlculs a dades obtingudes en experiments realitzats utilitzant *hairpins* més llargs ja que en comptar amb més dades (hi hauria més salts en total) l'estudi seria més acurat i, a més, la probabilitat de tenir salts més grans també augmentaria. Ara bé, cal tenir en compte que encara que hem procurat introduir restriccions per agilitzar els càlculs els algorismes són farragosos i els temps de càlculs elevats. Aquests temps creixeran ràpidament si augmentem la longitud de les molècules ja que comptarem amb més estats possibles i les seqüències generades seran més grans.
Un altre estudi possible seria fer un ajust previ del soroll utilitzant un altre mètode, per exemple ajustant gaussianes als histogrames experimentals, i tractar després el problema sense la necessitat de reestimar la variància. D'aquesta manera podríem aplicar l'algorisme a les dades inicials, sense la necessitat d'incloure els temps on el *hairpin* ja s'ha acabat d'obrir.

Com veiem, aquest treball de final de grau s'acaba aquí però deixa algunes qüestions obertes que podrien donar lloc a nous projectes.

Bibliografia

[1] Baum, L.E., Sell, G.R., *Growth transformations for functions on manifolds.*, *Pacific Journal of Mathematics, 27(2)* p. 211–227 (1968).

[2] Brémaud, P., *Markov Chains: Gibbs Fields and Monte Carlo Simulation, and Queues*, Springer, London (1999).

[3] Camunas-Soler, J., Manosas, M., Frutos, S., Tulla-Puche, J., Albericio, F., Ritort, F., *Single-molecule kinetics and footprinting of DNA bis-intercalation: the paradigmatic case of Thiocoraline*, *Nucleic Acids Research, 43(5)* p. 2767–2779 (2015).

[4] Huguet, J.M., Forns, N., Ritort, F., *Statistical properties of metastable intermediates in DNA unzipping*, *Physical Review Letters, 103(24)* 248106 (2009).

[5] Lawler, G.F., *Introduction to Stochastic Processes*, Chapman & Hall/CRC, London (2006).

[6] Lawrence R.R., *A tutorial on hidden Markov models and selected applications in speech recognition*, *Proceedings of the IEEE, 77(2)* p. 257–286 (1989).

[7] MacDonald, I.L., Zucchini, W., *Hidden Markov and Other Models for Discrete-valued Time Series*, Chapman & Hall, London (1997).

[8] Manosas, M., Ritort, F., *Thermodynamic and kinetic aspects of RNA pulling*, *Biophysical Journal, 88(5)* p. 3224–3242 (2005).

[9] Müllner, F.E., Syed, S., Selvin, P.R., Sigworth, F.J., *Improved hidden Markov models for molecular motors, part 1: Basic theory*, *Biophysical Journal, 99(11)* p. 3684–3695 (2010).

[10] Norris, J.R, *Markov Chains*, Cambridge University Press, Cambridge (1997).

[11] Ribezzi-Crivellari, M., Wagner, M., Ritort, F., *Bayesian approach to the determination of the kinetic parameters of DNA hairpins under tension*, *Journal of Nonlinear Mathematical Physics, 18(02)* p. 397–410 (2011).

[12] Storm. C., Nelson, P.C., *Theory of high-force DNA stretching and overstretching*, *Physical Review*, E *67* 051906 (2003).