CHAPTER 1

Models of nucleotide substitution

1.1 Introduction

Calculation of the distance between two sequences is perhaps the simplest phylogenetic analysis, yet it is important for two reasons. First, calculation of pairwise distances is the first step in distance matrix methods of phylogeny reconstruction, which use cluster algorithms to convert a distance matrix into a phylogenetic tree. Second, Markov process models of nucleotide substitution used in distance calculation form the basis of likelihood and Bayesian methods of phylogeny reconstruction. Indeed, joint analysis of multiple sequences can be viewed as a natural extension of pairwise distance calculation. Thus, besides discussing distance estimation, this chapter introduces the theory of Markov chains used in modelling nucleotide substitutions in a DNA sequence. It also introduces the method of maximum likelihood (ML). Bayesian estimation of pairwise distances and Bayesian phylogenetics are introduced in Chapters 6–8.

The distance between two sequences is defined as the expected number of nucleotide substitutions per site. If the evolutionary rate is constant over time, the distance will increase linearly with the time of divergence. A simplistic distance measure is the proportion of different sites, sometimes called the p distance. If 10 sites are different between two sequences, each 100 nucleotides long, then $p = 10\% = 0.1$. This raw proportion works fine for very closely related sequences but is otherwise a clear underestimate of the number of substitutions that have occurred. A variable site may result from more than one substitution, and even a constant site, with the same nucleotide observed in the two sequences, may harbour back or parallel substitutions (Figure 1.1). Multiple substitutions at the same site or *multiple hits* cause some changes to be hidden. As a result, p is not a linear function of evolutionary time. Thus the raw proportion p is usable only for highly similar sequences, with $p < 5\%$, say.

To estimate the number of substitutions, we need a probabilistic model to describe changes between nucleotides over evolutionary time. Continuous-time Markov chains are commonly used for this purpose. The nucleotide sites in the sequence are assumed to be evolving independently of each other. Substitutions at any particular site are described by a Markov chain, with the four nucleotides to be the *states* of the chain. The main feature of a Markov chain is that it has no memory: 'given the present, the future does not depend on the past'. In other words, the probability with which the chain jumps into other nucleotide states depends on the current state, but not on how the current state is reached. This is known as the *Markovian property*. Besides this basic assumption, we often place further constraints on substitution rates between nucleotides, leading to

Molecular Evolution: A Statistical Approach. Ziheng Yang. © Ziheng Yang 2014.
Published 2014 by Oxford University Press.

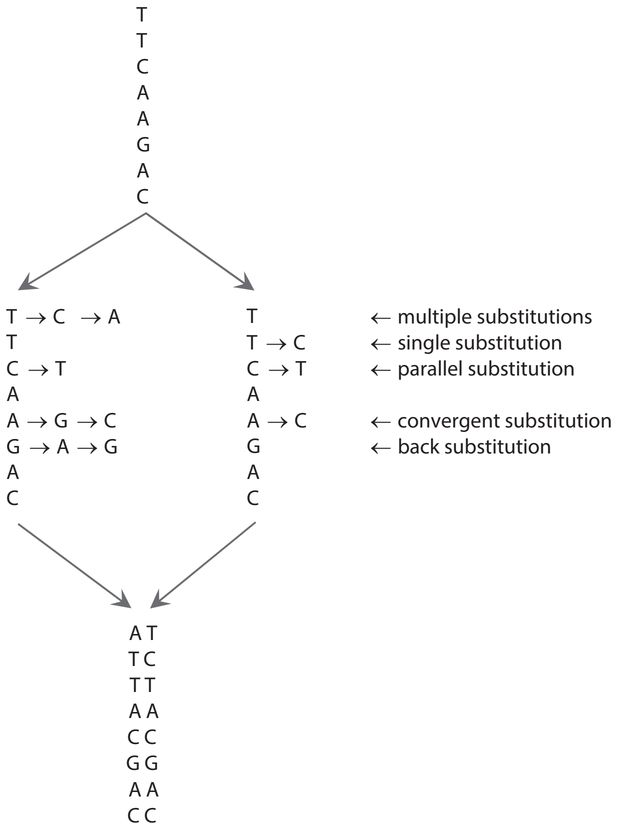

Fig. 1.1 Illustration of multiple substitutions at the same site or multiple hits. An ancestral sequence has diverged into two sequences and has since accumulated nucleotide substitutions independently along the two lineages. Only two *differences* are observed between the two present-day sequences, so that the proportion of different sites is $\hat{p} = 2/8 = 0.25$, while in fact as many as 10 *substitutions* (seven on the left lineage and three on the right lineage) occurred so that the true distance is $10/8 = 1.25$ substitutions per site. Constructed following Graur and Li (2000).

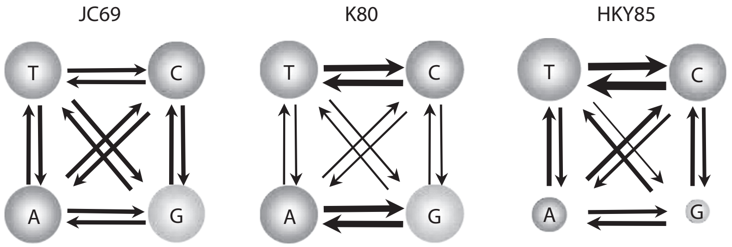

Fig. 1.2 Relative substitution rates between nucleotides under three Markov chain models of nucleotide substitution: JC69, K80, and HKY85. The thickness of the lines represents the substitution rates, while the sizes of the circles represent the steady-state distribution.

different models of nucleotide substitution. A few commonly used models are summarized in Table 1.1 and illustrated in Figure 1.2. These are discussed below.

Table 1.1 Substitution rate matrices for commonly used Markov models of nucleotide substitution

			To			
	p	From	T	C	A	G
JC69 (Jukes and Cantor 1969)	1	T	·	λ	λ	λ
		C	λ	·	λ	λ
		A	λ	λ	·	λ
		G	λ	λ	λ	·
K80 (Kimura 1980)	2	T	·	α	β	β
		C	α	·	β	β
		A	β	β	·	α
		G	β	β	α	·
F81 (Felsenstein 1981)	4	T	·	π_C	π_A	π_G
		C	π_T	·	π_A	π_G
		A	π_T	π_C	·	π_G
		G	π_T	π_C	π_A	·
HKY85 (Hasegawa et al. 1984, 1985)	5	T	·	$\alpha\pi_C$	$\beta\pi_A$	$\beta\pi_G$
		C	$\alpha\pi_T$	·	$\beta\pi_A$	$\beta\pi_G$
		A	$\beta\pi_T$	$\beta\pi_C$	·	$\alpha\pi_G$
		G	$\beta\pi_T$	$\beta\pi_C$	$\alpha\pi_A$	·
F84 (Felsenstein, DNAML program since 1984)	5	T	·	$(1+\kappa/\pi_Y)\beta\pi_C$	$\beta\pi_A$	$\beta\pi_G$
		C	$(1+\kappa/\pi_Y)\beta\pi_T$	·	$\beta\pi_A$	$\beta\pi_G$
		A	$\beta\pi_T$	$\beta\pi_T$	·	$(1+\kappa/\pi_R)\beta\pi_G$
		G	$\beta\pi_T$	$\beta\pi_C$	$(1+\kappa/\pi_R)\beta\pi_A$	·
TN93 (Tamura and Nei 1993)	6	T	·	$\alpha_1\pi_C$	$\beta\pi_A$	$\beta\pi_G$
		C	$\alpha_1\pi_T$	·	$\beta\pi_A$	$\beta\pi_G$
		A	$\beta\pi_T$	$\beta\pi_C$	·	$\alpha_2\pi_G$
		G	$\beta\pi_T$	$\beta\pi_C$	$\alpha_2\pi_A$	·
GTR (REV) (Tavaré 1986; Yang 1994b; Zharkikh 1994)	9	T	·	$a\pi_C$	$b\pi_A$	$c\pi_G$
		C	$a\pi_T$	·	$d\pi_A$	$e\pi_G$
		A	$b\pi_T$	$d\pi_C$	·	$f\pi_G$
		G	$c\pi_T$	$e\pi_C$	$f\pi_A$	·
UNREST (Yang 1994b)	12	T	·	a	b	c
		C	d	·	e	f
		A	g	h	·	i
		G	j	k	l	·

Note: The diagonals of the matrix are determined by the requirement that each row sums to 0. p is the number of free parameters in the model. If only relative rates are considered (as in a typical likelihood analysis), the number should be reduced by 1. In F84, $\pi_Y = \pi_T + \pi_C$ and $\pi_R = \pi_A + \pi_G$. The equilibrium distribution is $\pi = (\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$ under JC69 and K80, and $\pi = (\pi_T, \pi_C, \pi_A, \pi_G)$ under F81, F84, HKY85, TN93, and GTR. Under the general unrestricted (UNREST) model, it is given by equation (1.61).

1.2 Markov models of nucleotide substitution and distance estimation

1.2.1 *The JC69 model*

The JC69 model (Jukes and Cantor 1969) assumes that every nucleotide has the same instantaneous rate λ of changing into every other nucleotide. We use q_{ij} to denote the substitution rate from nucleotides i to j, with i, j = T, C, A, or G. Thus the *substitution rate matrix* is

$$Q = \{q_{ij}\} = \begin{bmatrix} -3\lambda & \lambda & \lambda & \lambda \\ \lambda & -3\lambda & \lambda & \lambda \\ \lambda & \lambda & -3\lambda & \lambda \\ \lambda & \lambda & \lambda & -3\lambda \end{bmatrix}, \qquad (1.1)$$

where the nucleotides are ordered T, C, A, and G. The diagonals are determined by the mathematical requirement that each row of the matrix sums to 0. The total rate of substitution of any nucleotide i is 3λ, which is $-q_{ii}$.

To relate the Markov chain model to sequence data, we need calculate the probability that given the nucleotide i at a site now, it will become nucleotide j time t later. This is known as the *transition probability*, denoted $p_{ij}(t)$. If time t is very small, we have $p_{ij}(t) \approx q_{ij}t$ for $i \neq j$, and $p_{ii}(t) \approx 1 - t\sum_{j \neq i} q_{ij}$. In other words, the *matrix of transition probabilities* is

$$P(t) = \{p_{ij}(t)\} \approx I + Qt = \begin{bmatrix} 1-3\lambda t & \lambda t & \lambda t & \lambda t \\ \lambda t & 1-3\lambda t & \lambda t & \lambda t \\ \lambda t & \lambda t & 1-3\lambda t & \lambda t \\ \lambda t & \lambda t & \lambda t & 1-3\lambda t \end{bmatrix}, \text{for small } t. \qquad (1.2)$$

Suppose a random region of the human genome evolves according to the JC69 model, at the rate of $3\lambda = 2.2 \times 10^{-9}$ substitutions/site/year (Kumar and Subramanian 2002) (Table 1.2). Consider a site occupied by a T right now. The probability that $t = 10^6$ years later this site will have a C will be $\lambda t = 0.00073$, and the probability that it remains to be T will be $1 - 3\lambda t = 0.9978$.

Equation (1.2) does not work well if t is not small. In general,

$$P(t) = \mathrm{e}^{Qt} = I + Qt + \frac{1}{2!}(Qt)^2 + \frac{1}{3!}(Qt)^3 + \cdots. \qquad (1.3)$$

We will discuss the calculation of this matrix exponential later. For the moment, we simply give the solution for the JC69 model as

$$P(t) = e^{Qt} = \begin{bmatrix} p_0(t) & p_1(t) & p_1(t) & p_1(t) \\ p_1(t) & p_0(t) & p_1(t) & p_1(t) \\ p_1(t) & p_1(t) & p_0(t) & p_1(t) \\ p_1(t) & p_1(t) & p_1(t) & p_0(t) \end{bmatrix}, \text{ with } \begin{cases} p_0(t) = \frac{1}{4} + \frac{3}{4}\mathrm{e}^{-4\lambda t}, \\ p_1(t) = \frac{1}{4} - \frac{1}{4}\mathrm{e}^{-4\lambda t}. \end{cases} \qquad (1.4)$$

Imagine a long sequence with nucleotide i at every site, and let every site evolve for a time period t. Then the proportion of nucleotide j in the sequence will be $p_{ij}(t)$, for j = T, C, A, G. The two different elements of the transition probability matrix, $p_0(t)$ and $p_1(t)$, are plotted in Figure 1.3. A few features of the matrix $P(t)$ are worth noting. First, every row of $P(t)$ sums to 1, because at any time t the chain has to be in one of the four nucleotide states. Second, $P(0) = I$, the identity matrix, reflecting the case of no evolution ($t = 0$). Third, rate λ and time t occur in the transition probabilities only in the form of the product λt. Thus if we are given a source sequence and a target sequence,

Table 1.2 A sample of estimated mutation/substitution rates

Taxa	Genes/genomes	Mutation/substitution rate	Source
Placental mammals	Genomic mutation rate at four-fold degenerate sites	2.2×10^{-9} per site per year	Kumar & Subramanian (2002)
Primates	12 protein-coding genes in the mitochondrial genome	7.9×10^{-9} per site per year for all codon positions, or $2.2, 0.1, 4.2 \times 10^{-9}$ per site per year for positions 1, 2, and 3, respectively.	Yang & Yoder (2003)
Human	Family-based genome sequencing	$1.1\text{–}1.2 \times 10^{-8}$ per site per generation	Roach et al. (2010), Kong et al. (2012)
Plants (rice and maize)	Nuclear genome	6×10^{-9}/site/year for synonymous 9×10^{-11}/site/year for nonsynonymous	Gaut (1998)
Plants (rice and maize)	Mitochondrial genome	0.3×10^{-9}/site/year for synonymous 1.3×10^{-11}/site/year for nonsynonymous	Gaut (1998)
Plants (rice and maize)	Chloraplast genome	1.1×10^{-9}/site/year for synonymous 1.8×10^{-11}/site/year for nonsynonymous	Gaut (1998)
HIV virus	HIV-1 *env* V3 region	$2\text{–}17 \times 10^{-3}$/site/year	Berry et al. (2007)

it will be impossible to tell whether the source has evolved into the target at rate λ over time t or at rate 2λ over time $t/2$. In fact, the sequences will look the same for any combination of λ and t as long as λt is fixed. With no external information about either the time or the rate, we can estimate only the distance, but not time and rate individually.

Lastly, when $t \to \infty$, $p_{ij}(t) = 1/4$, for all i and j. This represents the case where so many substitutions have occurred at every site that the target nucleotide is random, with probability

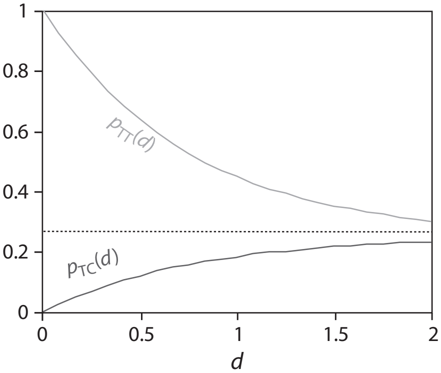

Fig. 1.3 Transition probabilities under the JC69 model (equation (1.4)) plotted against distance $d = 3\lambda t$, measured in the expected number of substitutions per site.

$1/4$ for every nucleotide, irrespective of the starting nucleotide. The probability that the chain is in state j when $t \to \infty$ is represented by π_j and the distribution $(\pi_T, \pi_C, \pi_A, \pi_G)$ is known as the *limiting distribution* of the chain. For the JC69 model, $\pi_j = 1/4$ for every nucleotide j. If the states of the chain are already in the limiting distribution, the chain will stay in that distribution, so the limiting distribution is also the *steady-state distribution* or *stationary distribution*. In other words, if a long sequence starts with T at every site, the proportions of the four nucleotides T, C, A, and G will drift away from $(1, 0, 0, 0)$ and approach $(1/4, 1/4, 1/4, 1/4)$, as the sequence evolves. If the sequence starts with equal proportions of the four nucleotides, it will continue to have equal proportions of the four nucleotides as the sequence evolves. The Markov chain is said to be stationary, or nucleotide substitutions are said to be in equilibrium. This is an assumption made in almost all models used in phylogenetic analysis, and is violated if the sequences in the data have different base compositions.

How does the Markov chain model correct for multiple hits and recover the hidden changes illustrated in Figure 1.1? This is achieved through the calculation of the transition probabilities using equation (1.3), which accommodates all possible paths the evolutionary process might have taken. In particular, the transition probabilities for a Markov chain satisfy the following equation, known as the Chapman–Kolmogorov equation (e.g. Grimmett and Stirzaker 1992, p. 239):

$$p_{ij}(t_1 + t_2) = \sum_k p_{ik}(t_1)\, p_{kj}(t_2). \tag{1.5}$$

This is a direct application of the *law of total probability*: the probability that nucleotide i will become nucleotide j time $t_1 + t_2$ later is a sum over all possible states k at any intermediate time point t_1 (Figure 1.4).

We now consider estimation of the distance between two sequences. From equation (1.1), the total substitution rate for any nucleotide is 3λ. If the two sequences are separated by time t (for example, if they diverged from a common ancestor time $t/2$ ago), the distance between the two sequences will be $d = 3\lambda t$. Suppose x out of n sites are different between the two sequences, so that the proportion of different sites is $\hat{p} = x/n$. (The hat or caret is used to indicate that the proportion is an estimate from the data.) To derive the expected probability p of different sites, consider one sequence as the ancestor of the other. By the symmetry of the model (equation (1.4)), this is equivalent to considering the two sequences as descendants of an extinct common ancestor. From equation (1.4), the probability that the nucleotide in the descendant sequence is different from the nucleotide in the ancestral sequence is

$$p(d) = 3p_1(t) = \frac{3}{4} - \frac{3}{4}\mathrm{e}^{-4\lambda t} = \frac{3}{4} - \frac{3}{4}\mathrm{e}^{-4d/3}. \tag{1.6}$$

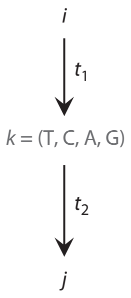

Fig. 1.4 Illustration of the Chapman–Kolmogorov theorem. The transition probability from any nucleotide i to any nucleotide j over time $t_1 + t_2$ is a sum over all possible states k at any intermediate time point t_1.

By equating this to the observed proportion $\hat{p}$, we obtain an estimate of distance as

$$\hat{d} = -\frac{3}{4} \log\left(1 - \frac{4}{3}\hat{p}\right), \tag{1.7}$$

where the logarithm has base e (sometimes written as ln instead of log). If $\hat{p} \geq \frac{3}{4}$, the distance formula will be inapplicable; two random sequences should have about 75% different sites, and if $\hat{p} \geq \frac{3}{4}$, the distance estimate is infinite. To derive the variance of $\hat{d}$, note that $\hat{p}$ is a binomial proportion with variance $\hat{p}(1-\hat{p})/n$. Considering $\hat{d}$ as a function of $\hat{p}$ and using the so-called delta technique (see Appendix B), we obtain

$$\mathrm{var}(\hat{d}) = \mathrm{var}(\hat{p}) \times \left|\frac{\mathrm{d}\hat{d}}{\mathrm{d}\hat{p}}\right|^2 = \hat{p}(1-\hat{p})/n \times \frac{1}{(1-4\hat{p}/3)^2} \tag{1.8}$$

(Kimura and Ohta 1972).

Example 1.1. The observed sequences of human and orangutan 12s rRNA genes from the mitochondrial genome are summarized in Table 1.3. From the table, $x = 90$ out of the $n = 948$ sites are different, so that $\hat{p} = x/n = 0.09494$. By equation (1.7), $\hat{d} = 0.1015$. Equation (1.8) gives the variance of $\hat{d}$ as 0.0001188 and standard error 0.0109. The approximate 95% confidence interval is thus $\hat{d} \pm 1.96 \times \mathrm{SE} = 0.1015 \pm 1.96 \times 0.0109$ or $(0.0801, 0.1229)$. □

1.2.2 *The K80 model*

Substitutions between the two pyrimidines (T ↔ C) or between the two purines (A ↔ G) are called *transitions*, while those between a pyrimidine and a purine (T, C ↔ A, G) are called *transversions*. In real data, transitions often occur at higher rates than transversions. Thus Kimura (1980) proposed a model that accounts for different transition and transversion rates. Note that the biologist's use of the term transition (as opposed to transversion) has nothing to do with the probabilist's use of the same term (as in transition probability). Typically the usage is clear from the context and there is little risk of confusion.

Let the substitution rates be α for transitions and β for transversions. The model is referred to as K80, also known as Kimura's two-parameter model. The rate matrix is as follows (see also Figure 1.2):

Table 1.3 Numbers and frequencies (in parentheses) of sites for the 16 site configurations (patterns) in human and orangutan mitochondrial 12s rRNA genes

	Human				
Orang	T	C	A	G	Sum (π_i)
T	179 (0.188819)	23 (0.024262)	1 (0.001055)	0 (0)	0.2141
C	30 (0.031646)	219 (0.231013)	2 (0.002110)	0 (0)	0.2648
A	2 (0.002110)	1 (0.001055)	291 (0.306962)	10 (0.010549)	0.3207
G	0 (0)	0 (0)	21 (0.022152)	169 (0.178270)	0.2004
Sum (π_j)	0.2226	0.2563	0.3323	0.1888	1

Note: Genbank accession numbers for the human and orangutan sequences are D38112 and NC_001646, respectively (Horai et al. 1995). There are 954 sites in the alignment, but six sites involve alignment gaps and are removed, leaving 948 sites in each sequence. The average base frequencies in the two sequences are 0.2184 (T), 0.2605 (C), 0.3265 (A), and 0.1946 (G).

CHAPTER 4

Maximum likelihood methods

4.1 Introduction

In this chapter, we will discuss likelihood calculation for multiple sequences on a phylogenetic tree. As indicated at the end of last chapter, this is a natural extension to the parsimony method, when we want to incorporate differences in branch lengths and in substitution rates between nucleotides. Likelihood calculation on a tree is also a natural extension to estimation of the distance between two sequences, discussed in Chapter 1. Indeed Chapter 1 has covered the general principles of Markov chain theory and maximum likelihood (ML) estimation needed in this chapter.

It may be beneficial to distinguish two applications of ML in phylogenetic analysis. The first is estimation of parameters in the evolutionary model and testing of hypotheses concerning the evolutionary process when the tree topology is known or fixed. The likelihood method, with its nice statistical properties, provides a powerful and flexible framework for such analysis (e.g. Stuart et al. 1999). The second is inference of the tree topology. The log likelihood for each tree is maximized by optimizing branch lengths and other substitution parameters, and the optimized log likelihood is used as a tree score for comparing different trees. This second application of ML corresponds to comparison of many statistical models. It involves complexities, which will be discussed in Chapter 5.

4.2 Likelihood calculation on tree

4.2.1 *Data, model, tree, and likelihood*

The likelihood is defined as the probability of observing the data when the parameters are given, although it is considered to be a function of the parameters. The data consist of s aligned homologous sequences, each n nucleotides long, and can be represented as an $s \times n$ matrix $X = \{x_{jh}\}$, where x_{jh} is the hth nucleotide in the jth sequence. Let $\mathbf{x}_h$ denote the hth column in the data matrix. To define the likelihood, we have to specify the model by which the data are generated. Here we use the K80 nucleotide substitution model (Kimura 1980). We assume that different sites evolve independently of each other and evolution in one lineage is independent of other lineages. We use the tree of five species of Figure 4.1 as an example to illustrate the likelihood calculation. The observed data at a particular site, TCACC, are shown. The ancestral nodes are numbered 0, 6, 7, and 8, with 0 being the root. The length of the branch leading to node i is denoted t_i, defined as the expected number of nucleotide substitutions per site. The parameters in the model include the branch lengths and the transition/transversion rate ratio κ, collectively denoted $\theta = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, \kappa\}$.

Molecular Evolution: A Statistical Approach. Ziheng Yang. © Ziheng Yang 2014.
Published 2014 by Oxford University Press.

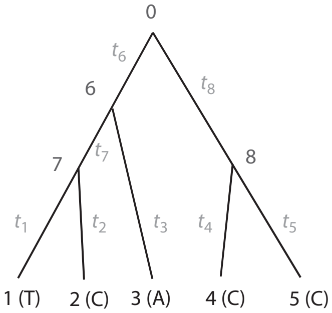

Fig. 4.1 A tree of five species used to demonstrate calculation of the likelihood function. The nucleotides observed at the tips at a site are shown. Branch lengths t_1–t_8 are measured by the expected number of nucleotide substitutions per site.

Because of the assumption of independent evolution among sites, the probability of the whole dataset (the alignment) is the product of the probabilities of data at individual sites:

$$L(\theta) = f(X|\theta) = \prod_{h=1}^{n} f(\mathbf{x}_h\,|\theta)\,. \tag{4.1}$$

Equivalently the log likelihood is a sum over sites in the sequence

$$\ell = \log\{L(\theta)\} = \sum_{h=1}^{n} \log\{f(\mathbf{x}_h\,|\theta)\,\}. \tag{4.2}$$

Here we consider calculation of ℓ when parameters θ are given. We focus on one site, with the data $\mathbf{x}_h$ = TCACC, say. We use x_i to represent the state at ancestral node i, and suppress the subscript h. Since the data at the site can result from any combination of ancestral nucleotides $x_0x_6x_7x_8$, calculation of $f(\mathbf{x}_h)$ has to sum over all possible nucleotide combinations for the extinct ancestors (nodes 0, 6, 7, and 8)

$$f(\mathbf{x}_h|\theta)=\sum_{x_0}\sum_{x_6}\sum_{x_7}\sum_{x_8}\left[\pi_{x_0}p_{x_0x_6}(t_6)p_{x_6x_7}(t_7)p_{x_7T}(t_1)p_{x_7C}(t_2)p_{x_6A}(t_3)p_{x_0x_8}(t_8)p_{x_8C}(t_4)p_{x_8C}(t_5)\right]. \tag{4.3}$$

Here the summation over each of x_0, x_6, x_7, x_8 is over the four nucleotides T, C, A, G. The quantity in the square brackets is the probability of data TCACC for the tips and $x_0x_6x_7x_8$ for the ancestral nodes. This is equal to the probability that the root (node 0) has x_0, which is given by $\pi_{x_0} = {}^1/_4$ under K80, multiplied by eight transition probabilities along the eight branches of the tree. We discussed calculation of the transition probabilities in Chapter 1; for example, those under K80 are given in equation (1.10).

Note that given θ, we are able to calculate $f(\mathbf{x}_h|\theta)$ and the log likelihood ℓ. The ML method then estimates θ by maximizing ℓ, often using numerical optimization algorithms (to be discussed in §4.5).

4.2.2 *The pruning algorithm*

4.2.2.1 *Horner's rule and the pruning algorithm*

Summing over all combinations of ancestral states is expensive because there are 4^{s-1} possible combinations for $s - 1$ interior nodes. The situation is even worse for amino acid or codon sequences as there will then be 20^{s-1} or 61^{s-1} possible combinations. An important technique that is useful in calculating such sums is to identify common factors and calculate them only once. This is known as the *nesting rule* or *Horner's rule*, published by the Irish mathematician William Horner in 1830. The rule was also published in 1820 by a London watchmaker, Theophilus Holdred, and the same principle had been used in 1303 by the Chinese mathematician Zhu Shijie (朱世杰). By this rule, an nth-order polynomial can be calculated with only n multiplications and n additions. For example, a naïve calculation of $1 + 2x + 3x^2 + 4x^3$, as $1 + 2 \cdot x + 3 \cdot x \cdot x + 4 \cdot x \cdot x \cdot x$, requires six multiplications and three additions. However, by writing it as $1 + x \cdot (2 + x \cdot (3 + 4 \cdot x))$, only three multiplications and three additions are needed. As another example, $\sum_{i=1}^{10} \sum_{j=1}^{10} (x_i y_{ij}) = \sum_{i=1}^{10} \left[x_i \left(\sum_{j=1}^{10} y_{ij} \right) \right]$, but the left-hand side involves 100 multiplications and 99 additions while the right-hand side involves only ten multiplications and 99 additions.

If we apply the nesting rule and move the summation signs in equation (4.3) to the right as far as possible, we get

$$\begin{aligned} f(\mathbf{x}_h|\theta) = \sum_{x_0} \pi_{x_0} & \Bigg\{ \sum_{x_6} p_{x_0x_6}(t_6) \Bigg[\Bigg(\sum_{x_7} p_{x_6x_7}(t_7) p_{x_7T}(t_1) p_{x_7C}(t_2) \Bigg) p_{x_6A}(t_3) \Bigg] \Bigg] \Bigg\} \\ & \times \Bigg[\sum_{x_8} p_{x_0x_8}(t_8) p_{x_8C}(t_4) p_{x_8C}(t_5) \Bigg]. \end{aligned} \tag{4.4}$$

Thus we sum over x_7 before x_6, and sum over x_6 and x_8 before x_0. In other words, we sum over ancestral states at a node only after we have done so for all its descendant nodes.

The pattern of parentheses and the occurrences of the tip states in equation (4.4), in the form [(T, C), A], [C, C], match the tree of Figure 4.1. This is no coincidence. Indeed calculation of $f(\mathbf{x}_h|\theta)$ by equation (4.4) constitutes the *pruning algorithm* of Felsenstein (1973b, 1981). This is a variant of the dynamic programming algorithm discussed in §3.4.3. Its essence is to successively calculate probabilities of data at the site on many subtrees. Define $L_i(x_i)$ to be the conditional probability of observing data at the tips that are descendants of node i, given that the nucleotide at node i is x_i. For example, tips 1, 2, 3 are descendants of node 6, so $L_6(\mathrm{T})$ is the probability of observing $x_1x_2x_3 = \mathrm{TCA}$, given that node 6 has the state $x_6 = \mathrm{T}$. With $x_i = \mathrm{T, C, A, G}$, we calculate a vector of conditional probabilities for each node i. In the literature, the conditional probability $L_i(x_i)$ is often referred to as the 'partial likelihood' or 'conditional likelihood'; these are misnomers since likelihood refers to the probability of the whole dataset and not probability of data at a single site or part of a single site.

If node i is a tip, its descendant tips include tip i itself only, so that $L_i(x_i) = 1$ if x_i is the observed nucleotide, or 0 otherwise. If node i is an interior node with daughter nodes j and k, we have

$$L_i(x_i) = \left[\sum_{x_j} p_{x_ix_j}(t_j) L_j(x_j) \right] \times \left[\sum_{x_k} p_{x_ix_k}(t_k) L_k(x_k) \right]. \tag{4.5}$$

This is a product of two terms, corresponding to the two daughter nodes j and k. Note that tips that are descendants of node i must be descendants of either j or k. Thus the probability $L_i(x_i)$ of observing all descendant tips of node i (given the state x_i at node i) is equal to the probability of observing data at the descendant tips of node j (given x_i) times the probability of observing data at the descendant tips of node k (given x_i). These are the two terms in the two pairs of brackets in equation (4.5), respectively. For example, node $i = 6$ has daughter nodes $j = 7$ and $k = 3$, and descendant tip nodes 1, 2, 3. The probability of observing $x_1x_2x_3$ given x_6 is the probability of observing x_1x_2 given x_6, times the probability of observing x_3 given x_6. Given the state x_i at node i, the two parts of the tree down node i are independent. If node i has more than two daughter nodes, $L_i(x_i)$ will be a product of as many terms. Now consider the first term, the term in the first pair of brackets, which is the probability of observing data at descendant tips of node j (given the state x_i at node i). This is the probability $p_{x_i x_j}(t_j)$ that x_i will become x_j over branch length t_j times the probability $L_j(x_j)$ of observing the tips of node j given the state x_j at node j, summed over all possible states x_j.

We calculate the conditional probability vector $L_i(x_i)$ for node i only after the vectors $L_j(x_j)$ and $L_k(x_k)$ for its daughter nodes j and k have been calculated. Thus we calculate the probabilities of data x_1x_2 down node 7, then the probabilities of data $x_1x_2x_3$ down node 6, then the probabilities of data x_4x_5 down node 8, and finally the probabilities of the whole data $x_1x_2x_3x_4x_5$ down node 0. The calculation proceeds from the tips of the tree towards the root, visiting each node only after all its descendant nodes have been visited. In computer science, this way of visiting all nodes on the tree is known as the *post-order tree traversal* (as opposed to *pre-order tree traversal*, in which ancestors are visited before descendants). After visiting all nodes on the tree and calculating the probability vector for the root $L_0(x_0)$, the probability of data at the site is given as

$$f(\mathbf{x}_h|\theta) = \sum\nolimits_{x_0} \pi_{x_0} L_0(x_0). \tag{4.6}$$

Note that π_{x_0} is the (prior) probability that the nucleotide at the root is x_0, given by the equilibrium frequency of the nucleotide x_0 under the model.

Example 4.1. We use the tree of Figure 4.1 to provide a numerical example of the calculation using the pruning algorithm at one site (Figure 4.2). For definiteness, we fix internal

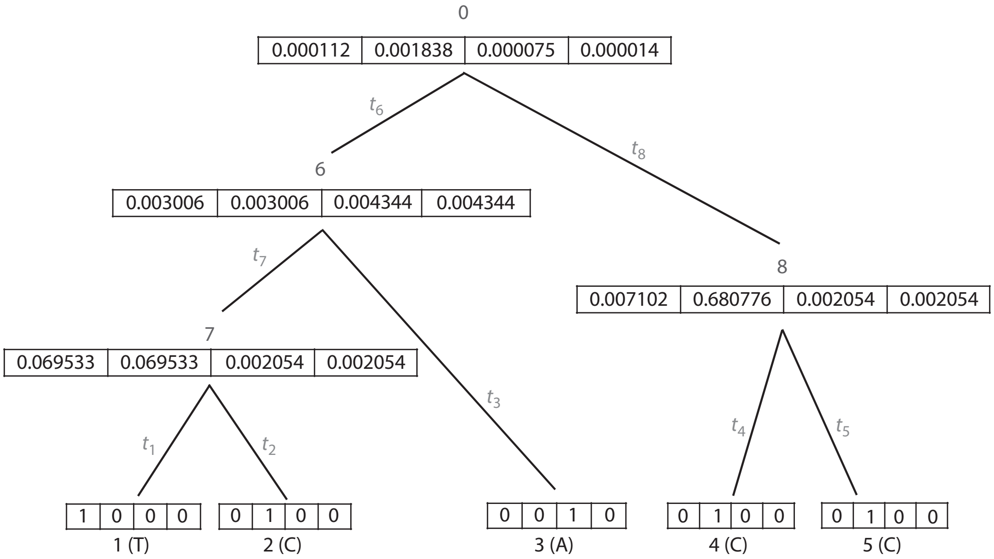

Fig. 4.2 Illustration of the pruning algorithm for likelihood calculation when the branch lengths and other parameters are fixed. The tree of Figure 4.1 is reproduced, showing the vector of conditional probabilities at each node. The four elements in the vector at each node are the probabilities of observing data at the descendant tips, given that the node has T, C, A, or G, respectively. For example, 0.069533 for node 7 is the probability of observing data x_1x_2 = TC at tips 1 and 2, given that node 7 has T. The K80 model is assumed, with $\kappa = 2$. The branch lengths are fixed at 0.1 for the internal branches and 0.2 for the external branches. The transition probability matrices are shown in the text.

branch lengths at $t_6 = t_7 = t_8 = 0.1$ and external branch lengths at $t_1 = t_2 = t_3 = t_4 = t_5 = 0.2$. We also set $\kappa = 2$. The two transition probability matrices are as follows, in which the *ij*th element is $p_{ij}(t)$, with the nucleotides ordered T, C, A, and G (see equation (1.10) for K80):

$$P(0.1) = \begin{bmatrix} 0.906563 & 0.045855 & 0.023791 & 0.023791 \\ 0.045855 & 0.906563 & 0.023791 & 0.023791 \\ 0.023791 & 0.023791 & 0.906563 & 0.045855 \\ 0.023791 & 0.023791 & 0.045855 & 0.906563 \end{bmatrix},$$

$$P(0.2) = \begin{bmatrix} 0.825092 & 0.084274 & 0.045317 & 0.045317 \\ 0.084274 & 0.825092 & 0.045317 & 0.045317 \\ 0.045317 & 0.045317 & 0.825092 & 0.084274 \\ 0.045317 & 0.045317 & 0.084274 & 0.825092 \end{bmatrix}.$$

Consider node 7, which has daughter nodes 1 and 2. Using equation (4.5), we obtain the first entry in the probability vector as $L_7(\mathrm{T}) = p_{TT}(0.2) \times p_{TC}(0.2) = 0.825092 \times 0.084274 = 0.069533$. This is the probability of observing T and C at tips 1 and 2, given that node 7 has T. The other entries, $L_7(\mathrm{C}), L_7(\mathrm{A})$, and $L_7(\mathrm{G})$, can be calculated similarly, as can the vector for node 8. Next the vector at node 6 can be calculated, by using the conditional probability vectors at daughter nodes 7 and 3. Finally, we calculate the vector for node 0, the root. The first entry, $L_0(\mathrm{T}) = 0.000112$, is the probability of observing the descendant tips (1, 2, 3, 4, 5) of node 0, given that node 0 has $x_0 = \mathrm{T}$. Equation (4.5) gives this as the product of two terms. The first term, $\sum_{x_6} p_{x_0x_6}(t_6)L_6(x_6)$, sums over x_6 and is the probability of observing data TCA at the tips 1, 2, 3, given that node 0 has T. This is $0.906563 \times 0.003006 + 0.045855 \times 0.003006 + 0.023791 \times 0.004344 + 0.023791 \times 0.004344 = 0.003070$. The second term, $\sum_{x_8} p_{x_0x_8}(t_8)L_8(x_8)$, is the probability of observing data CC at tips 4 and 5, given that node 0 has T. This is $0.906563 \times 0.007102 + 0.045855 \times 0.680776 + 0.023791 \times 0.002054 + 0.023791 \times 0.002054 = 0.037753$. The product of the two terms gives $L_0(\mathrm{T}) = 0.00011237$. Other entries in the vector for node 0 can be similarly calculated. Finally application of equation (4.6) gives the probability of data at the site as $f(\mathbf{x}_h|\theta) = 0.000509843$, with $\log\{f(\mathbf{x}_h|\theta)\} = -7.581408$. □

4.2.2.2 *Savings on computation*

The pruning algorithm is a major time saver. As in the dynamic programming algorithm discussed in §3.4.3, in the pruning algorithm the amount of computation required by one calculation of the likelihood increases linearly with the number of nodes or the number of species, even though the number of combinations of ancestral states increases exponentially.

Some other obvious savings may be mentioned here as well. First, the same transition probability matrix is used for all sites or site patterns in the sequence and may be calculated only once for each branch. Second, if two sites have the same data, the probabilities of observing them will be the same and need be calculated only once. Collapsing sites into *site patterns* thus leads to a saving in computation, especially if the sequences are highly similar so that many sites have identical patterns. Under JC69, some sites with different data, such as TCAG and TGCA, also have the same probability of occurrence and can be collapsed further (Saitou and Nei 1986). The same applies to K80, although the saving is not as much as under JC69. It is also possible to collapse *partial site patterns* corresponding to subtrees (e.g. Kosakovsky Pond and Muse 2004). For example, consider the tree of Figure 4.1 and two sites with data TCACC and TCACT. The conditional probability vectors