BAB II

KONSEP DAN TEORI DASAR

Pada bab ini akan dibahas beberapa konsep dan teori dasar yang digunakan untuk membahas bab-bab selanjutnya.

2.1 BIOINFORMATIKA

Keberhasilan para ahli dalam mengungkap barisan DNA dari salah satu spesies hidup pada akhir tahun 1970-an menjadi awal terbentuknya basis data barisan. Dengan ditemukannya berbagai teknik untuk mengkode suatu barisan, berbagai barisan dari berbagai spesies pun mulai banyak dihasilkan. Inilah yang kemudian menjadi penyebab meledaknya jumlah data barisan yang terjadi pada tahun 1980-an dan sekaligus sebagai awal dibukanya proyek-proyek genom dan analisa data barisan yang mempelopori lahirnya ilmu bioinformatika.

Bioinformatika lahir pada pertengahan era 1980-an sebagai ilmu yang berkaitan dengan penggunaan komputer dalam ilmu biologi. Ilmu ini digunakan sebagai media untuk menyelesaikan berbagai masalah biologi dengan menggunakan informasi yang terdapat pada suatu barisan. Seiring dengan berkembangnya teknologi yang telah ada, bioinformatika pun terus

berkembang dengan pesat. Hal ini tidak lepas dari peranan penting alat-alat pendukung seperti pembuatan algoritma dalam menganalisa data barisan yang memang sudah dikembangkan sejak tahun 1960-an.

Salah satu data barisan yang dapat dianalisa dan akan dijadikan objek pada skripsi ini adalah barisan *deoxyribonucleic acid* atau yang lebih dikenal sebagai barisan DNA. DNA merupakan asam nukleat pembawa materi genetik yang bertugas mewariskan sifat dari kedua orang tua dan mengatur perkembangan biologis seluruh bentuk kehidupan secara seluler. DNA terdapat pada kromosom yang terletak pada inti sel. Satu molekul DNA terdiri dari dua rantai polinukleotida linier yang berpilin yang biasa disebut sebagai *double helix*.

Ada empat macam nukleotida yang dimiliki DNA yaitu *adenine*, *cytosine, guanine*, dan *thymine*. Keempat nukleotida ini digolongkan menjadi dua *purine* yaitu *adenine* dan *guanine*, dan dua *pirimidine* yaitu *cytosine* dan *thymine*. Nukleotida-nukleotida ini membentuk rangkaian yang kemudian dilambangkan dengan huruf *A*, *C*, *G*, *T* yang merupakan inisial dari masing-masing nukleotida. Rangkaian inilah yang disebut sebagai barisan DNA. Contoh barisan DNA yaitu*TAGGAATCCT TAGAGCTA*.

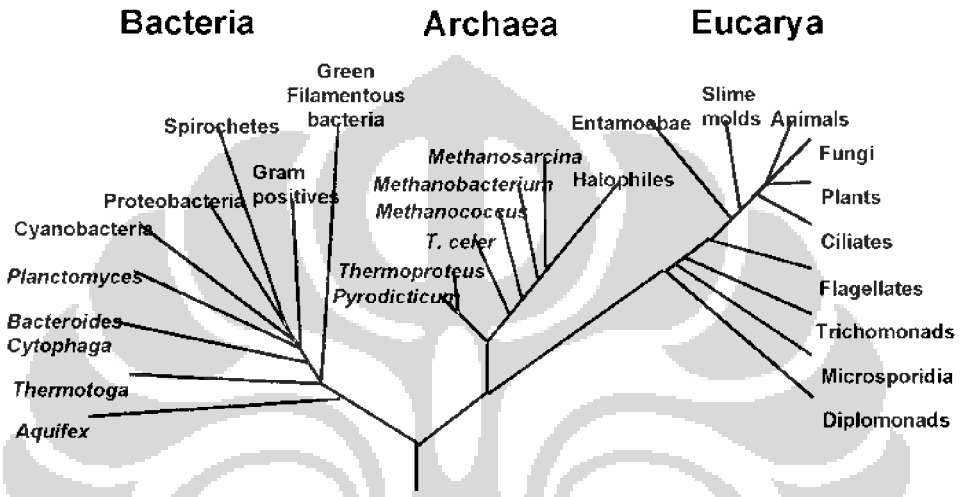
Banyak informasi yang dapat diperoleh dari suatu barisan DNA. Dengan menganalisa barisan DNA tersebut, dapat disusun sejarah evolusi dari berbagai spesies hidup di bumi yang direpresentasikan dalam bentuk *tree* yang disebut sebagai *phylogenetic tree*. Ada beberapa metode yang

digunakan dalam membangun *phylogenetic tree*, salah satunya yaitu dengan menggunakan metode jarak. Dalam membangun *phylogenetic tree* dengan menggunakan metode jarak ini dibutuhkan suatu matriks jarak yang elemen-elemennya merepresentasikan jarak dari tiap pasang spesies yang terlibat. Pada skripsi ini akan dibahas mengenai pembentukan matriks jarak dalam membangun *phylogenetic tree*.

2.2 *PHYLOGENETIC TREE*

Phylogenetic tree didefinisikan sebagai *tree* yang merepresentasikan evolusi dari berbagai spesies hidup di bumi [7]. Pada *phylogenetic tree*, setiap terminal nodenya melambangkan spesies-spesies yang terlibat sedangkan internal nodenya melambangkan nenek moyang bersama dari node-node yang berhubungan dengannya. *Edge* atau cabang pada *phylogenetic tree* melambangkan jarak dari tiap pasang spesies yang terlibat. Berikut diberikan contoh *phylogenetic tree* seperti yang terlihat pada Gambar 2.1. Gambar 2.1 ini merupakan gambar *phylogenetic tree* kehidupan dari *bacteria*, *archaea*, dan *eucarya*. Pada gambar ini terlihat bahwa terminal node dari *phylogenetic tree*nya, yaitu animals, fungi, plants, cilliates, flagellates, dan yang lainnya merupakan spesies-spesies yang terlibat. Sedangkan internal nodenya, sebagai contoh yaitu node yang menghubungkan antara animals, fungi, dan plants merupakan nenek moyang

bersama dari ketiga spesies tersebut. Dan cabang pada *phylogenetic tree*nya, sebagai contoh yaitu cabang yang menghubungkan antara animals dan fungi merupakan jarak antara kedua spesies tersebut.



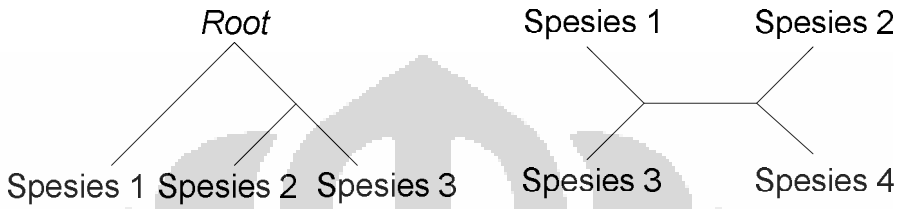
Sumber: http://darwin.nmsu.edu/~molb470/fall2005/projects/pan/images/Phylogenetic*Tree*OfLife.jpg

Gambar 2.1 Contoh *phylogenetic tree*

Hasil dari suatu *phylogenetic tree* dapat berupa *rooted* dan *unrooted*. *Rooted* dan *unrooted* ini ditentukan dengan melihat ada tidaknya node yang bertindak sebagai *root* seperti yang terlihat pada Gambar 2.2 dan Gambar 2.3. Pada Gambar 2.2 terlihat bahwa terdapat node yang bertindak sebagai *root* sehingga Gambar 2.2 ini merupakan suatu *rooted phylogenetic tree*. Sebaliknya pada Gambar 2.3 terlihat bahwa tidak ada node yang bertindak sebagai *root* sehingga Gambar 2.3 merupakan suatu *unrooted phylogenetic tree*.

Root dari suatu *phylogenetic tree* dapat ditentukan dengan mengambil salah satu spesies yang merupakan bagian dari spesies-spesies yang terlibat. Sebagai contoh pada kasus Gambar 2.3, spesies 1 dapat diambil

sebagai *root* dari *phylogenetic tree* tersebut sehingga *phylogenetic tree* yang terlihat pada Gambar 2.3 ini nantinya akan menghasilkan suatu *rooted phylogenetic tree* dengan spesies 1 sebagai *root*-nya.



Gambar 2.2 *Rooted tree*

Sumber: Isaev, Alexander [5]

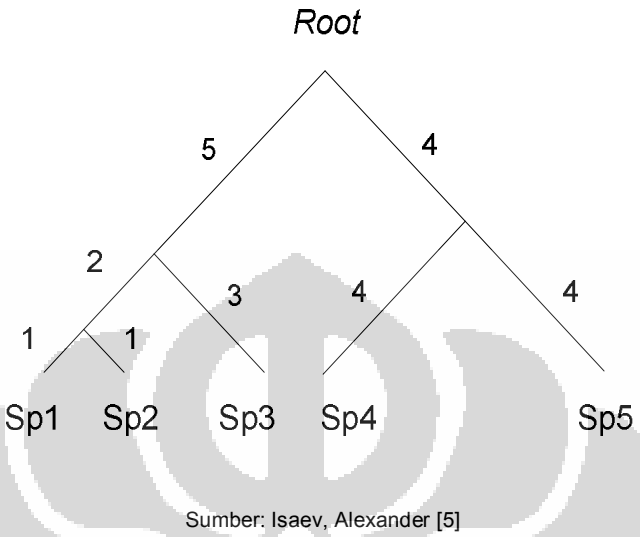
Gambar 2.3 *Unrooted tree*

Pengambilan salah satu spesies yang akan dijadikan sebagai *root* bergantung pada tujuan seseorang dalam menganalisa suatu *phylogenetic tree*. Artinya seseorang boleh saja menjadikan spesies manapun sebagai *root* pada *phylogenetic tree* bergantung tujuannya dalam menganalisa *phylogenetic tree* tersebut. Akan tetapi tetap pada kenyataannya, tidak sembarang spesies yang dijadikan sebagai *root* menghasilkan *phylogenetic tree* yang memiliki arti secara biologi. Sehingga seseorang juga harus berhati-hati dalam menentukan spesies mana yang akan dijadikan sebagai *root*.

Untuk kasus dimana spesies-spesies yang terlibat merupakan spesies-spesies yang sangat dekat sekali kekerabatannya (*closed related*), *root* yang diambil dari salah satu spesies yang merupakan bagian dari spesies-spesies yang terlibat tersebut akan menghasilkan *phylogenetic tree* yang tidak memilki arti secara biologi. Namun *root* ini masih dapat digunakan untuk

melihat urutan divergensi dari spesies-spesies yang terlibat. Urutan divergensi ini merepresentasikan spesies mana yang lebih dahulu. Pada kasus yang *closed related* ini, untuk melihat kekerabatan antara spesies-spesiesnya akan lebih baik jika diambil *root* yang bukan merupakan bagian dari spesies-spesies yang terlibat. *Root* ini dapat diambil dari salah satu spesies yang berasal dari kelompok lain yang cukup jauh kekerabatannya dari kelompok spesies-spesies yang terlibat tersebut.

Dari suatu *rooted phylogenetic tree* dapat dihasilkan suatu *molecular clock tree* dimana setiap *path* yang terbentang dari *root* ke masing-masing spesies pada *tree* tersebut memiliki panjang yang sama. Ini menandakan bahwa tingkat evolusi masing-masing spesies yang tidak pernah berubah dari waktu ke waktu sehingga panjang cabang-cabang tersebut proporsional terhadap waktu yang sebenarnya [5]. Gambar 2.4 adalah contoh *molecular clock tree*. Pada Gambar 2.4 ini terlihat bahwa panjang path dari *root* ke Sp 1 yaitu 5+2+1=8, panjang path dari *root* ke Sp 2 yaitu 5+2+1=8, dan panjang path dari *root* ke Sp 3 yaitu 5+3=8, begitu pula dengan panjang path antara *root* ke Sp 4 dan Sp 5 yaitu 8. Karena setiap path yang terbentang antara *root* ke masing-masing spesies tersebut memiliki panjang yang sama, yaitu 8, maka *phylogenetic tree* ini merupakan suatu *molecular clock tree*.



Sumber: Isaev, Alexander [5]

Gambar 2.4 *Molecular clock tree*

2.3 MATRIKS JARAK

Seperti yang telah disebutkan pada Subbab I, untuk membangun suatu *phylogenetic tree* dengan menggunakan metode jarak dibutuhkan suatu matriks jarak. Matriks jarak didefinisikan sebagai matriks yang elemen-elemennya berisi jarak dari tiap pasang spesies yang terlibat yang direpresentasikan oleh matriks

$$Md = \begin{bmatrix} d_{11} & d_{12} & \mathrm{L} & d_{1j} & \mathrm{L} & d_{1N} \\ d_{21} & d_{22} & \mathrm{L} & d_{2j} & \mathrm{L} & d_{2N} \\ \mathrm{M} & \mathrm{M} & \mathrm{O} & \mathrm{M} & \mathrm{M} & \mathrm{M} \\ d_{i1} & d_{i2} & \mathrm{L} & d_{ij} & \mathrm{L} & d_{iN} \\ \mathrm{M} & \mathrm{M} & \mathrm{M} & \mathrm{M} & \mathrm{O} & \mathrm{M} \\ d_{N1} & d_{N2} & \mathrm{L} & d_{Nj} & \mathrm{L} & d_{NN} \end{bmatrix},$$

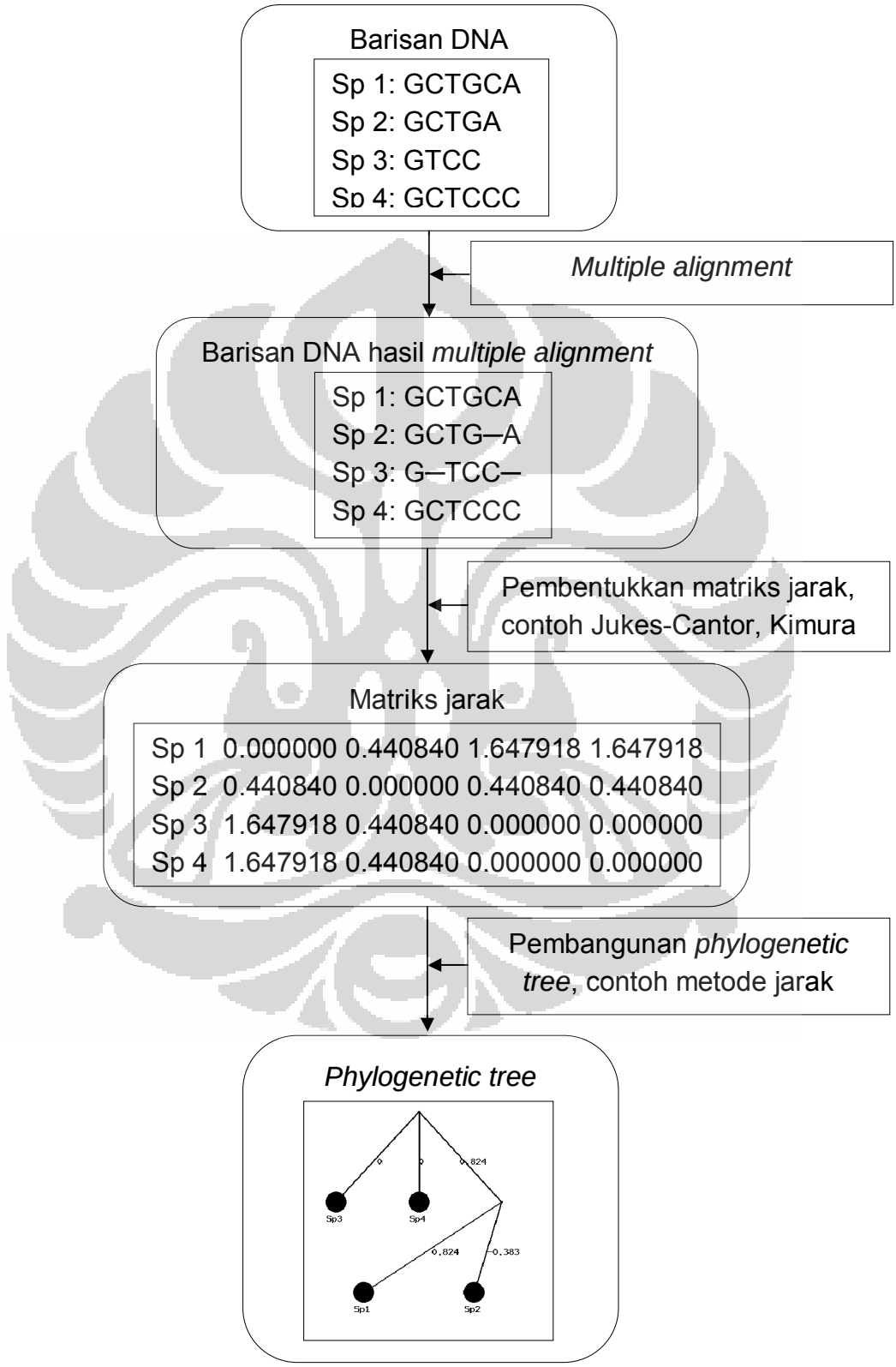
dimana d_{ij} merupakan jarak antara spesies i dengan spesies j untuk setiap $i, j = 1,2,...,N$ dengan N merupakan banyaknya spesies yang terlibat [5]. Jarak-jarak pada matriks ini dihitung dengan menggunakan *evolutionary model* yang diperoleh dengan menggunakan metode tertentu, diantaranya yaitu metode Jukes-Cantor dan metode Kimura yang akan dijelaskan lebih lanjut pada Bab III.

Sebelum proses penghitungan jarak dilakukan, barisan-barisan yang akan dihitung jaraknya harus terlebih dahulu dilakukan suatu proses yang disebut sebagai *sequence alignment* atau yang disingkat dengan *alignment*. *Alignment* merupakan suatu proses penyejajaran terhadap beberapa barisan dengan melihat similaritas yang ada pada barisan-barisan tersebut. Proses *alignment* ini terbagi menjadi dua, yaitu *pairwise* dan *multiple*. *Pairwise alignment* merupakan proses *alignment* yang melibatkan dua barisan sedangkan *multiple alignment* merupakan proses *alignment* yang melibatkan beberapa barisan. Berikut diberikan contoh hasil *multiple alignment* dari 4 barisan.

Babi	:	G C T G C A
Kuda	:	G C T G ─ A
Paus	:	G ─ T C C ─
Lumba-lumba	:	G C T C C C

Sumber: Isaev, Alexander [5]

Secara garis besar, alur pembangunan *phylogenetic tree* adalah seperti yang terlihat pada Gambar 2.5.

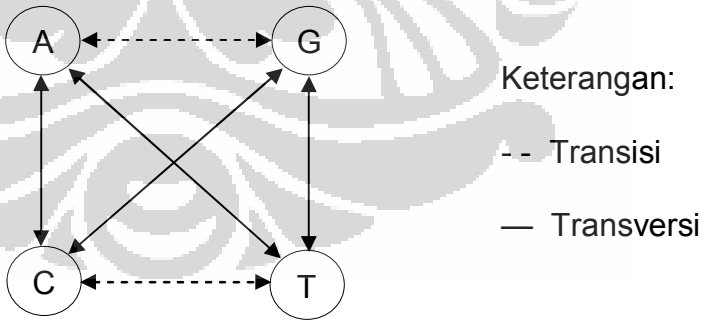


Gambar 2.5 Alur pembangunan *phylogenetic tree*

2.4 TEORI DASAR

Pada subbab ini akan dijelaskan beberapa teori dasar yang perlu diketahui dalam mencari suatu *evolutionary model*.

Evolutionary model diperoleh dengan menggambarkan substitusi nukleotida yang berlangsung pada barisan. Subtitusi nukleotida didefinisikan sebagai proses berubahnya suatu nukleotida menjadi nukleotida yang lain. Substitusi ini terbagi menjadi dua, yaitu transisi dan transversi, dimana transisi merupakan substitusi yang berlangsung antara sesama *purine* atau sesama *pirimidine* sedangkan transversi merupakan substitusi antara satu *purine* dengan satu *pirimidine*. Gambar 2.6 adalah contoh gambar substitusi transisi dan substitusi transversi.



Gambar 2.6 Substitusi transisi dan transversi

Pada Gambar 2.6 terlihat bahwa substitusi yang terjadi antara A (*adenine*) dengan G (*guanine*) dan C (*cytosine*) dengan T (*thymine*) merupakan substitusi transisi. Hal ini dikarenakan *adenine* dan *guanine* merupakan sesama *purine* dan *cytosine* dan *thymine* merupakan sesama

pirimidine sehingga substitusi yang terjadi adalah substitusi transisi. Pada Gambar 2.6 terlihat juga bahwa substitusi yang terjadi selain substitusi transisi tersebut adalah substitusi transversi. Hal ini dikarenakan substitusi yang terjadi sisanya merupakan substitusi antara *purine* dengan *pirimidine* sehingga substitusi yang terjadi adalah substitusi transversi, contohnya yaitu A dan T.

Dalam menggambarkan substitusi yang berlangsung pada barisan, masing-masing posisi nukleotida diasumsikan bersifat saling bebas. Sehingga dengan begitu, substitusi yang berlangsung pada barisan tersebut dapat digambarkan oleh substitusi yang berlangsung pada posisi nukleotida tunggal saja atau *single site*. Substitusi pada *single site* ini digambarkan melalui suatu *continuous-time finite Markov chain* atau *continuous-time finite Markov model* yang akan dianggap memiliki karakteristik yang sama dengan *discrete-time Markov-chain* yang bergantung pada parameter waktu t [5]. Berikut diberikan penjelasannya.

2.4.1 *Markov Chain*

Sebelum dijelaskan lebih lanjut mengenai *continuous-time Markov chain*, akan dijelaskan terlebih dahulu mengenai *discrete-time Markov chain*.

2.4.1.1 *Discrete-time Markov Chain*

Misalkan terdapat suatu himpunan berhingga dari semua *state* atau keadaan yang mungkin, yaitu $S = \{S_1, S_2, \ldots, S_m\}$. Suatu *Markov chain* pasti akan berada pada salah satu *state* dalam himpunan tersebut untuk setiap titik-titik waktu $t = 1,2,3,\ldots$. Sehingga untuk setiap langkah waktu t ke $t+1$, *Markov chain* akan mempunyai dua kemungkinan, yaitu tetap berada pada *state* yang sama, atau berubah menjadi *state* lain dalam S. Berubahnya suatu *state* S_i pada waktu t menjadi sembarang *state* S_j pada waktu $t+1$ ini berlangsung dengan probabilitas tertentu dan diasumsikan hanya bergantung pada *state* S_i dan *state* S_j saja, bukan pada t dan bukan pula pada *state-state* sebelum S_i.

Probabilitas suatu *state* berubah dari *state* S_i ke *state* S_j dinotasikan oleh $p_{S_i S_j}$, yang secara sederhana ditulis sebagai p_{ij}. Probabilitas-probabilitas p_{ij} untuk $i, j = 1,2,\ldots,m$ ini disebut sebagai probabilitas transisi dari *Markov chain* yang kemudian ditulis dalam bentuk matriks

$$P = \begin{bmatrix} p_{11} & p_{12} & \text{L} & p_{1m} \\ p_{21} & p_{22} & \text{L} & p_{21} \\ \text{M} & \text{M} & \text{O} & \text{M} \\ p_{m1} & p_{m2} & \text{L} & p_{mm} \end{bmatrix},$$

yang disebut sebagai matriks probabilitas transisi. Elemen-elemen pada tiap baris matriks tersebut merupakan seluruh kemungkinan probabilitas suatu

state berubah menjadi sembarang *state* dalam S sehingga elemen-elemen pada tiap barisnya berjumlah satu dan tiap-tiap elemennya bernilai non negatif.

Pada *Markov chain* didefinisikan juga probabilitas-probabilitas awal $\pi(S_j)$, yang secara sederhana ditulis sebagai π_j untuk $j=1,2,\ldots,m$. π_j merupakan probabilitas suatu *Markov chain* mula-mula berada pada *state* S_j. Untuk setiap $j=1,2,\ldots,m$, π_j ini ditulis dalam bentuk vektor yang disebut sebagai vektor probabilitas awal.

2.4.1.2 *Continuous-time Markov chain*

Selanjutnya akan dijelaskan mengenai *continuous-time Markov chain* dalam menggambarkan substitusi nukleotida pada *single site*. Istilah *site* kemudian diartikan sebagai posisi nukleotida.

Seperti yang telah disebutkan sebelumnya, *continuous-time Markov chain* akan dianggap memiliki karakteristik yang sama dengan *discrete-time Markov chain* yang bergantung pada parameter waktu t. Karena DNA memiliki empat macam nukleotida, maka terdapat empat kemungkinan suatu *site* akan berada pada atau berubah menjadi suatu *state*, yaitu A, C, G, atau T. Sehingga himpunan dari semua *state* yang mungkin dari

*Markov chain*nya adalah $S=\{A,C,G,T\}$, dan matriks probabilitas transisi yang bersesuaian dengan *Markov chain*nya adalah

$$P(t)=\begin{bmatrix} p_{AA}(t) & p_{AC}(t) & p_{AG}(t) & p_{AT}(t) \\ p_{CA}(t) & p_{CC}(t) & p_{CG}(t) & p_{CT}(t) \\ p_{GA}(t) & p_{GC}(t) & p_{GG}(t) & p_{GT}(t) \\ p_{TA}(t) & p_{TC}(t) & p_{TG}(t) & p_{TT}(t) \end{bmatrix},$$

untuk setiap $t \geq 0$. Matriks $P(t)$ tersebut merupakan matriks yang berisi probabilitas-probabilitas dari seluruh kemungkinan suatu *site* berubah dari suatu *state* menjadi *state* lain dalam S pada waktu t.

Sama halnya seperti pada *discrete-time Markov chains*, $P(t)$ merupakan suatu matriks yang elemen-elemennya bernilai non negatif dan penjumlahan elemen-elemen dari tiap barisnya bernilai satu, atau secara matematis dapat ditulis

$$p_{ij}(t) \geq 0 \text{ dan } \sum_{j \in S} p_{ij}(t) = 1,$$

untuk setiap $i, j \in S$ dan $t \geq 0$, dimana $p_{ij}(t)$ merupakan probabilitas suatu *site* berubah dari *state* i ke *state* j pada waktu t.

Dengan menganggap proses substitusi nukleotida pada *single site* berjalan sesuai dengan *Markov chain*, maka dibuatlah asumsi berikut.

Asumsi [5]. Jika pada waktu t_0 *site* berada pada *state* $i \in S$ maka probabilitas pada waktu $t_0 + t$ *site* berada pada *state* $j \in S$ hanya bergantung pada i, j dan t. (dan tepat merupakan elemen $p_{ij}(t)$ pada matriks $P(t)$).

Untuk suatu $t,\tau \geq 0$, probabilitas $p_{ij}(t+\tau)$ dapat diartikan sebagai probabilitas suatu *site* berubah dari *state* i ke *state* j pada waktu $t+\tau$. Sembarang transisi dari *state* i ke *state* j tersebut dapat dianggap sebagai transisi pertama-tama dari *state* i ke *state* k pada waktu t dilanjutkan dengan transisi dari *state* k ke *state* j pada waktu τ. Sehingga

$$p_{ij}(t+\tau) = \sum_{k \in S} p_{ik}(t)\, p_{kj}(\tau), \tag{2.1}$$

untuk setiap $i, j \in S$. Jabarkan identitas tersebut masing-masing untuk setiap $i, j \in S$ dalam bentuk matriks, diperoleh

$$\begin{bmatrix} p_{AA}(t+\tau) & p_{AC}(t+\tau) & p_{AG}(t+\tau) & p_{AT}(t+\tau) \\ p_{CA}(t+\tau) & p_{CC}(t+\tau) & p_{CG}(t+\tau) & p_{CT}(t+\tau) \\ p_{GA}(t+\tau) & p_{GC}(t+\tau) & p_{GG}(t+\tau) & p_{GT}(t+\tau) \\ p_{TA}(t+\tau) & p_{TC}(t+\tau) & p_{TG}(t+\tau) & p_{TT}(t+\tau) \end{bmatrix}$$

$$= \begin{bmatrix} \sum_{k \in S} p_{Ak}(t)\, p_{kA}(\tau) & \sum_{k \in S} p_{Ak}(t)\, p_{kC}(\tau) & \sum_{k \in S} p_{Ak}(t)\, p_{kG}(\tau) & \sum_{k \in S} p_{Ak}(t)\, p_{kT}(\tau) \\ \sum_{k \in S} p_{Ck}(t)\, p_{kA}(\tau) & \sum_{k \in S} p_{Ck}(t)\, p_{kC}(\tau) & \sum_{k \in S} p_{Ck}(t)\, p_{kG}(\tau) & \sum_{k \in S} p_{Ck}(t)\, p_{kT}(\tau) \\ \sum_{k \in S} p_{Gk}(t)\, p_{kA}(\tau) & \sum_{k \in S} p_{Gk}(t)\, p_{kC}(\tau) & \sum_{k \in S} p_{Gk}(t)\, p_{kG}(\tau) & \sum_{k \in S} p_{Gk}(t)\, p_{kT}(\tau) \\ \sum_{k \in S} p_{Tk}(t)\, p_{kA}(\tau) & \sum_{k \in S} p_{Tk}(t)\, p_{kC}(\tau) & \sum_{k \in S} p_{Tk}(t)\, p_{kG}(\tau) & \sum_{k \in S} p_{Tk}(t)\, p_{kT}(\tau) \end{bmatrix}$$

$$= \begin{bmatrix} p_{AA}(t)\, p_{AA}(\tau) + \ldots + p_{AT}(t)\, p_{TA}(\tau) & p_{AA}(t)\, p_{AC}(\tau) + \ldots + p_{AT}(t)\, p_{TC}(\tau) \\ p_{CA}(t)\, p_{AA}(\tau) + \ldots + p_{CT}(t)\, p_{TA}(\tau) & p_{CA}(t)\, p_{AC}(\tau) + \ldots + p_{CT}(t)\, p_{TC}(\tau) \\ \qquad\qquad \vdots & \qquad\qquad \vdots \\ p_{TA}(t)\, p_{AA}(\tau) + \ldots + p_{TT}(t)\, p_{TA}(\tau) & p_{TA}(t)\, p_{AC}(\tau) + \ldots + p_{TT}(t)\, p_{TC}(\tau) \end{bmatrix}$$

$$\begin{bmatrix} \text{L} & p_{AA}(t)p_{AT}(\tau)+\ldots+p_{AT}(t)p_{TT}(\tau) \\ \text{L} & p_{CA}(t)p_{AT}(\tau)+\ldots+p_{CT}(t)p_{TT}(\tau) \\ \text{O} & \text{M} \\ \text{L} & p_{TA}(t)p_{AT}(\tau)+\ldots+p_{TT}(t)p_{TT}(\tau) \end{bmatrix}$$

$$= \begin{bmatrix} p_{AA}(t) & p_{AC}(t) & p_{AG}(t) & p_{AT}(t) \\ p_{CA}(t) & p_{CC}(t) & p_{CG}(t) & p_{CT}(t) \\ p_{GA}(t) & p_{GC}(t) & p_{GG}(t) & p_{GT}(t) \\ p_{TA}(t) & p_{TC}(t) & p_{TG}(t) & p_{TT}(t) \end{bmatrix} \begin{bmatrix} p_{AA}(\tau) & p_{AC}(\tau) & p_{AG}(\tau) & p_{AT}(\tau) \\ p_{CA}(\tau) & p_{CC}(\tau) & p_{CG}(\tau) & p_{CT}(\tau) \\ p_{GA}(\tau) & p_{GC}(\tau) & p_{GG}(\tau) & p_{GT}(\tau) \\ p_{TA}(\tau) & p_{TC}(\tau) & p_{TG}(\tau) & p_{TT}(\tau) \end{bmatrix}$$

$$= P(t)\cdot P(\tau).$$

Sehingga untuk suatu $t,\tau \geq 0$ berlaku

$$P(t+\tau) = P(t)P(\tau). \tag{2.2}$$

Dengan menganggap *continuous-time Markov chain* yang menggambarkan substitusi nukleotida ini merupakan suatu *regular continuous-time Markov chain* berarti:

1. $P(0)$ adalah suatu matriks identitas E dari $P(t)$ dengan sifat matriks identitas $P(t) = P(t)\cdot E = E\cdot P(t)$, dan
2. $P(t)$ diferensiabel di setiap $t \geq 0$, dengan artian bahwa setiap elemen dari matriks $P(t)$ diferensiabel di setiap $t \geq 0$ sebagai fungsi dalam t dan keberadaan turunan pertama dari $P(t)$ ini dijamin oleh diferensiabel pada $t = 0$.

Lemma [2]. Untuk $t \geq 0$ berlaku

$$P'(t) = P(t)P'(0).$$

Bukti.

Berdasarkan (2.2) dan *regular continuous-time Markov chain* maka untuk $t \geq 0$, $h > 0$, diperoleh

$$\frac{P(t+h)-P(t)}{h} = \frac{P(t)P(h)-P(t)}{h}$$

$$= \frac{P(t)P(h)-P(t)E}{h}$$

$$= \frac{P(t)P(h)-P(t)P(0)}{h}$$

$$= \frac{P(t)\left(P(h)-P(0)\right)}{h}.$$

Saat $h \to 0$,

$$\lim_{h\to 0}\frac{P(t+h)-P(t)}{h} = \lim_{h\to 0}\frac{P(t)\left(P(h)-P(0)\right)}{h}$$

$$= P(t)\cdot \lim_{h\to 0}\frac{\left(P(h)-P(0)\right)}{h}$$

$$= P(t)\cdot \lim_{h\to 0}\frac{\left(P(0+h)-P(0)\right)}{h},$$

sehingga terbukti bahwa

$$P'(t) = P(t)P'(0).$$

Teorema [5]. $P(t)$ memiliki bentuk

$$P(t) = e^{Qt},$$

dimana Q merupakan suatu matriks berukuran 4×4.

Bukti.

Berdasarkan lemma sebelumnya, untuk $t \geq 0$, berlaku

$$P'(t) = P(t)P'(0).$$

Karena setiap elemen dari matriks $P(t)$ diferensiabel di setiap $t \geq 0$ sebagai fungsi dalam t dan mempunyai turunan pertama di $t = 0$ maka diperoleh

$$\frac{d(P(t))}{dt} = P(t)P'(0)$$

$$\frac{d(P(t))}{P(t)} = P'(0)dt$$

Integralkan kedua ruas diperoleh

$$\int \frac{d(P(t))}{P(t)} = \int P'(0)dt$$

$$\ln P(t) = P'(0)t + c_1$$

$$P(t) = e^{P'(0)t + c_1}$$

$$= e^{P'(0)t} e^{c_1}$$

$$= ce^{P'(0)t}.$$

Asumsikan $P(0) = 1$ maka

$$P(0) = ce^{P'(0)\cdot 0} = 1,$$

diperoleh nilai $c = 1$. Dengan mensubstitusikan nilai c ke $P(t)$ diperoleh

$$P(t) = e^{Qt}, \qquad (2.3)$$

dimana $Q = P'(0)$ adalah suatu matriks berukuran 4×4.

Matriks Q disebut sebagai *matrix of instantaneous change* dengan sifat bahwa elemen-elemen pada tiap barisnya berjumlah nol.

Selanjutnya akan dijelaskan mengenai dua kondisi yang harus dipenuhi oleh *continuous-time Markov chain* dalam menggambarkan substitusi nukleotida.

1. *Stationary probability distribution* φ yang bersifat unik.

Definisi 1 [5]. Suatu vektor $\varphi = (\varphi_A, \varphi_C, \varphi_G, \varphi_T)$ dengan $\varphi_i \geq 0$ dan $\sum_{i \in S} \varphi_i = 1$ disebut sebagai *stationary probability distribution* dari *Markov chain* jika $\varphi Q = 0$. Ini ekivalen dengan menyatakan bahwa $\varphi P(t) \equiv \varphi$.

Stationary probability distribution memiliki sifat-sifat sebagai berikut.

(a) Didefinisikan probabilitas suatu *site* berada pada *state* i pada waktu t yaitu

$$p_i(t) = \sum_{k \in S} \pi_k p_{ki}(t),$$

dimana $\pi = (\pi_A, \pi_C, \pi_G, \pi_T)$ adalah vektor probabilitas awal untuk *Markov chain*. Dengan menetapkan $\pi = \varphi$ maka

$$p_i(t) = \sum_{k \in S} \varphi_k p_{ki}(t).$$

Untuk setiap $i,k \in S$, $\sum_{k\in S} \varphi_k p_{ki}(t)$ merupakan elemen-elemen dari hasil perkalian vektor φ dengan matriks $P(t)$ sehingga berdasarkan Definisi 1 diperoleh

$$p_i(t) = \sum_{k\in S} \varphi_k p_{ki}(t) = \varphi_i \tag{2.4}$$

(b) Matriks probabilitas transisi

$$P(t) \rightarrow \begin{bmatrix} \varphi_A & \varphi_C & \varphi_G & \varphi_T \\ \varphi_A & \varphi_C & \varphi_G & \varphi_T \\ \varphi_A & \varphi_C & \varphi_G & \varphi_T \\ \varphi_A & \varphi_C & \varphi_G & \varphi_T \end{bmatrix}$$

sejalan dengan $t \rightarrow \infty$. Atau dapat dikatakan bahwa untuk nilai t yang besar, $P(t)$ akan stabil dengan *stationary probability distribution* yang berhubungan dengannya.

2. Model yang bersifat *time-reversible*

Definisi 2 [5]. Misalkan π adalah vektor probabilitas awal dari suatu *continuous-time Markov chain*, dan anggap $p_i(t) \neq 0$ untuk setiap $i \in S$ dan $t \geq 0$. Didefinisikan *reversed Markov chain* sebagai *continuous-time Markov chain* yang diberikan oleh matriks probabilitas transisi $P^*(t)$ dengan

$$p_{ij}{}^*(t) = \frac{\pi_j p_{ji}(t)}{p_i(t)}, \tag{2.5}$$

untuk setiap $i, j \in S$. Anggap setiap elemen dari *stationary probability distribution* φ tidak bernilai nol. Dengan menetapkan $\pi = \varphi$ maka dari (2.4) dan (2.5) diperoleh

$$p_{ij}^{*}(t) = \frac{\varphi_j p_{ji}(t)}{\varphi_i},$$

untuk setiap $i, j \in S$.

Definisi 3 [5]. *Markov chain* yang memenuhi asumsi-asumsi pada Definisi 2 tersebut disebut *time-reversible* atau *simply reversible* jika $P^{*}(t) = P(t)$ untuk setiap $t \geq 0$.

Jika masing-masing elemen dari φ tidak bernilai nol maka untuk $\pi = \varphi$, *reversible* berarti

$$\varphi_i p_{ij}(t) = \varphi_j p_{ji}(t). \qquad (2.6)$$

Kedua kondisi diatas selanjutnya digunakan untuk mengestimasi jarak antara dua barisan dengan menggunakan *maximum likelihood* untuk memperoleh suatu *evolutionary model*.

2.4.2 *Maximum likelihood*

Pada subbab ini akan dijelaskan mengenai perolehan suatu *evolutionary model* dari suatu *maximum likelihood*. *Maximum likelihood* yang akan dijelaskan disini merupakan *maximum likelihood* untuk sembarang

regular time-reversible model yang digunakan untuk mengestimasi jarak antara dua barisan dengan panjang yang sama yang berasal dari nenek moyang yang sama dari suatu *molecular phylogenetic tree*. Setiap cabang pada *molecular phylogenetic tree* tersebut hanya memperhitungkan proses substitusi saja, bukan delesi, bukan juga insersi. Delesi dan insersi ini dilambangkan dengan adanya *gap* yang dinotasikan oleh '–', 'n', atau 'N' pada hasil proses *multiple alignment*. Jika terdapat barisan yang mengandung *gap*, maka *gap-gap* ini akan dihilangkan terlebih dahulu dari barisan sehingga barisan-barisan yang akan dilibatkan hanyalah barisan-barisan yang diperoleh dari hasil *reduced multiple alignment* saja. Barisan-barisan hasil *reduced multiple alignment* merupakan barisan-barisan hasil proses *multiple alignment* yang telah terlebih dahulu dihilangkan *gap-gap*nya. Penghilangan *gap* ini dilakukan dengan menghilangkan seluruh posisi yang mengandung *gap*. Berikut diberikan contoh hasil *reduced multiple alignment* dari empat barisan.

Misalkan dari suatu proses *multiple alignment* diperoleh barisan-barisan sebagai berikut.

Posisi		1 2 3 4 5 6
Babi	:	G C T G C A
Kuda	:	G C T G – A
Paus	:	G – T C C –
Lumba-lumba	:	G C T C C C

Karena pada posisi ke-2, ke-5 dan ke-6 terdapat barisan yang mengandung *gap*, maka nukleotida yang berada pada posisi-posisi tersebut dihilangkan dari tiap-tiap barisan. Sehingga diperoleh *reduced multiple alignment* sebagai berikut.

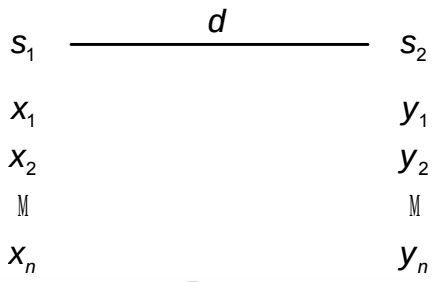
Babi	:	G T G
Kuda	:	G T G
Paus	:	G T C
Lumba-lumba	:	G T C

Selanjutnya akan dijelaskan mengenai *maximum likelihood*.

Misalkan terdapat dua barisan berbeda dengan panjang yang sama, yaitu $s_1 = x_1, x_2, ..., x_n$ dan $s_2 = y_1, y_2, ... y_n$, dengan $s_1 \neq s_2$, $x_k, y_k \in S = \{A, C, G, T\}$ untuk $k = 1, 2, ..., n$ dan n merupakan panjang barisan. Untuk $k = 1, 2, ..., n$, proses *alignment* dilakukan sebagai berikut

$$\begin{array}{lcccc} s_1: & x_1 & x_2 & \ldots & x_n \\ & | & | & & | \\ s_2: & y_1 & y_2 & \ldots & y_n \end{array}.$$

Misalkan terdapat *root* antara barisan s_1 dan barisan s_2 dengan jarak ke masing-masing barisan sebesar d_1 dan d_2, maka $d = d_1 + d_2$ merupakan jarak antara barisan s_1 dengan s_2 seperti yang terlihat pada struktur berikut.



Maka untuk $1 \le k \le n$ didefinisikan model *site-specific likelihood* untuk sembarang *regular time-reversible model* sebagai berikut

$$L_k = \sum_{j \in S} \varphi_j p_{jx_k}(d_1) p_{jy_k}(d_2),$$

dimana φ_j merupakan elemen dari suatu *stationary probability distribution.*

Untuk sembarang *time-reversible model*, maka dari (2.6) dan (2.1) diperoleh

$$\begin{aligned} L_k &= \sum_{j \in S} \varphi_j p_{jx_k}(d_1) p_{jy_k}(d_2) \\ &= \sum_{j \in S} \varphi_{x_k} p_{x_k j}(d_1) p_{jy_k}(d_2) \\ &= \varphi_{x_k} \sum_{j \in S} p_{x_k j}(d_1) p_{jy_k}(d_2) \\ &= \varphi_{x_k} p_{x_k y_k}(d_1 + d_2) \\ &= \varphi_{x_k} p_{x_k y_k}(d) \\ &= \varphi_{y_k} p_{y_k x_k}(d). \end{aligned}$$

Model *likelihood* didefinisikan sebagai perkalian dari seluruh model *site-specific likelihood*nya. Dengan demikian model *likelihood* untuk mengestimasi jarak antara dua barisan ditentukan oleh

$$L = L_1 \cdot L_2 \cdot \mathrm{K} \cdot L_n = \varphi_{x_1} \cdot p_{x_1 y_1}(d) \cdot \varphi_{x_2} \cdot p_{x_2 y_2}(d) \cdot \mathrm{K} \cdot \varphi_{x_n} \cdot p_{x_n y_n}(d),$$

dimana d adalah jarak antara barisan s_1 dengan s_2.

Dalam mencari suatu *evolutionary model*, parameter t yang terdapat pada matriks probabilitas transisi $P(t)$ didefinisikan sebagai jarak antara dua barisan (d) [2]. Sehingga jika dihubungkan dengan matriks probabilitas transisinya, model *likelihood* sekarang dapat ditulis menjadi

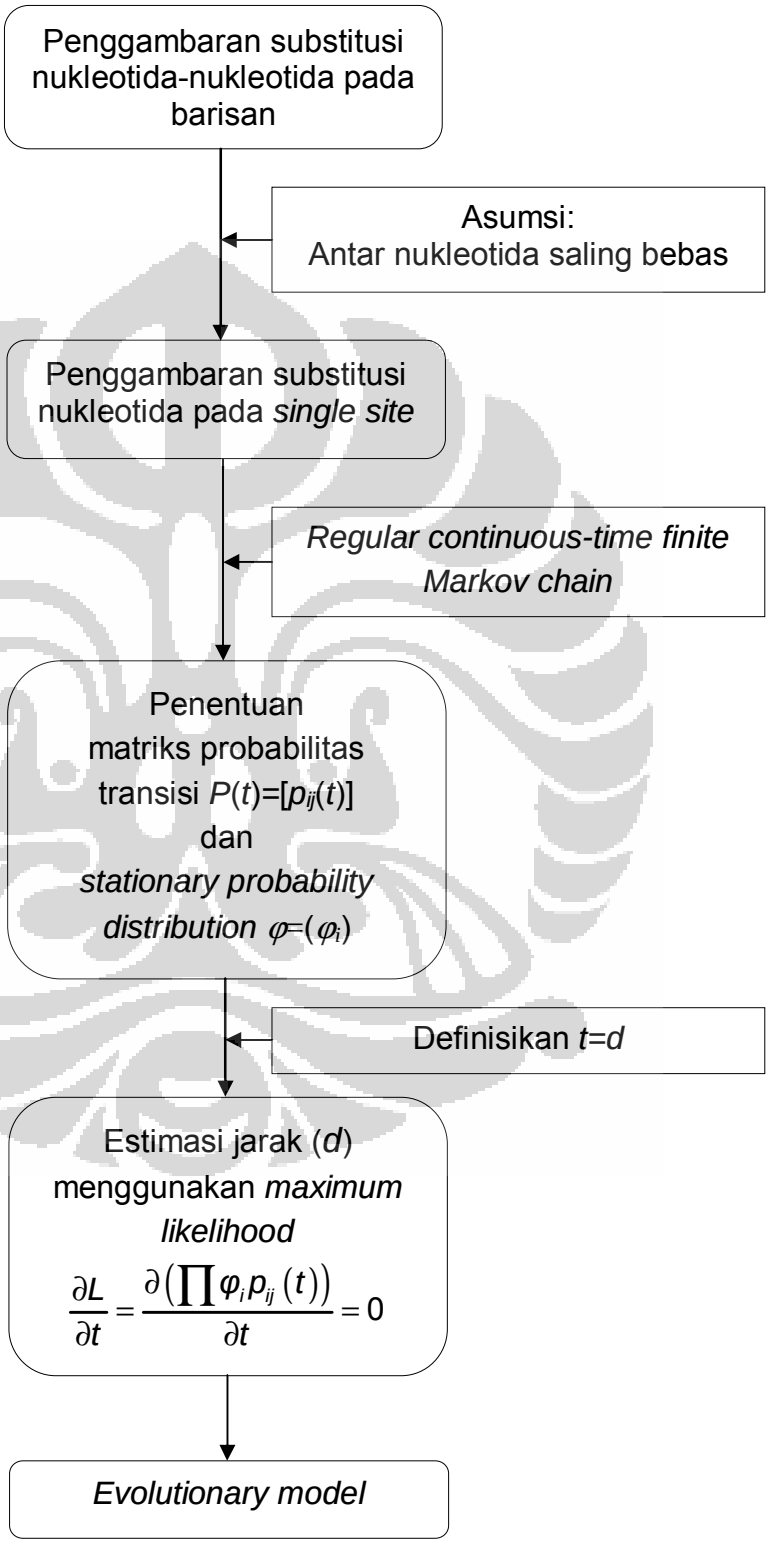
$$L = \varphi_{x_1} \cdot p_{x_1 y_1}(t) \cdot \varphi_{x_2} \cdot p_{x_2 y_2}(t) \cdot \mathrm{K} \cdot \varphi_{x_n} \cdot p_{x_n y_n}(t). \tag{2.7}$$

Diperoleh

$$\begin{aligned} \ln L &= \ln\left(\varphi_{x_1} \cdot p_{x_1 y_1}(t) \cdot \varphi_{x_2} \cdot p_{x_2 y_2}(t) \cdot \mathrm{K} \cdot \varphi_{x_n} \cdot p_{x_n y_n}(t)\right) \\ &= \ln\varphi_{x_1} + \ln p_{x_1 y_1}(t) + \ln\varphi_{x_2} + \ln p_{x_2 y_2}(t) + \mathrm{K} + \ln\varphi_{x_n} + \ln p_{x_n y_n}(t) \\ &= \ln\varphi_{x_1} + \ln\varphi_{x_2} + \ldots + \ln\varphi_{x_n} + \ln p_{x_1 y_1}(t) + \ln p_{x_2 y_2}(t) + \mathrm{K} + \ln p_{x_n y_n}(t), \end{aligned} \tag{2.8}$$

dimana φ_{x_k} merupakan elemen dari *stationary probability distribution* φ dan $p_{x_k y_k}(t)$ merupakan probabilitas suatu *site* berubah dari *state* x_k ke *state* y_k pada waktu t dengan $k = 1,2,\ldots,n$ dan n merupakan panjang barisan.

Dengan memaksimumkan model *likelihood* (2.8) ini maka dapat diperoleh suatu *evolutionary model*. Pada Gambar 2.7 diberikan alur penurunan suatu *evolutionary model*.



Gambar 2.7 Alur penurunan *evolutionary model*