Penerapan Algoritma Program Dinamis pada Penyejajaran Sekuens dengan Algoritma Smith–Waterman

Afif Bambang Prasetia (*13515058*)
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13515058@std.stei.itb.ac.id

***Abstract*—Menyejajarkan sekuens pada bidang bioinformatika merupakan salah satu hal yang dapat diselesaikan dengan algoritma Smith-Waterman yang merupakan algoritma berdasarkan algoritma program dinamis karena kemampuannya yang dapat menghasilkan hasil yang optimal. Dalam makalah ini akan dibahas tentang metode algoritma Smith-Waterman dan penggunaaannya untuk menyejajarkan sekuens.**

Keywords—Program Dinamis, Smith-Waterman, algoritma, penyejajaran sekuens, bioinformatika.

I. PENDAHULUAN

Bioinformatika merupakan salah satu ilmu yang penting dalam kehidupan dan telah menjadi penting dalam berbagai area biologi. Berbagai teknik dalam bioinformatika dapat membantu untuk mendapat banyak data dari memproses sinyal dan gambar dalam biologi molekul, selain itu membantu untuk menyusun genom dan mutasi yang diamati, mempunyai peranan dalam menggali teks dari literatur biologis dan pengembangan ontologi biologis dan gen untuk menyusun dan mengajukan data biologi.

Penyebab komputer menjadi penting pada bioinformatika karena sekuens dari protein dapat ditentukan dan penyusunan sekuens secara manual dibuktikan tidak efektif. Setelah salah satu dari sekuens protein dapat di *compile*, maka komputer menjadi esensial.

Tujuan penting dari bioinformatik adalah dalam pengembangan dan implementasi program komputer yang dapat membuat akses, menggunakan, dan memanajemen berbagai jenis informasi menjadi efisien, juga dalam pengembangan algoritma baru dan ukuran statistik yang menilai hubungan diantara anggota dari himpunan data yang besar.

Salah satu metode dari bioinformatika adalah menyusun sekuens sehingga dapat disamakan dengan sekuens lainnya, hal ini digunakan untuk mendapat hubungan fungsional, struktural, atau evolusioner diantara 2 sekuens tersebut. Tidak hanya dalam sekuens biologi, metode ini dapat digunakan untuk menghitung biaya jarak edit di antara string dalam *natural languange* atau dalam data finansial.

Meskipun sekuens yang sangat pendek atau mirip dapat di jajarkan dengan manual atau dengan tangan. Masalah yang paling dibutuhkan diperlukan penyejajaran yang panjang, dan memiliki sekuens yang berjumlah sangat banyak sehingga tidak dapat di jajarkan oleh kemampuan manusia. Sebagai gantinya pengetahuan manusia dapat digunakan untuk membuat algoritma yang dapat menghasilkan penyejajaran sekuens yang optimal. Salah satunya adalah algoritma Smith-Waterman yang berdasarkan algoritma program dinamis. Algoritma ini diaplikasikan untuk dapat menentukan hasil yang optimal walaupun lambat.

II. DASAR TEORI

A. Algoritma Program Dinamis

Program dinamis (dynamic programming) adalah suatu algoritma pemecahan masalah dengan cara menguraikan solusi menjadi sekumpulan tahap (stage) sedemikian sehingga solusi dari persoalan dapat dipandang dari serangkaian keputusan Makalah.

Program dinamis memeliki karakterisitik penyelesaian persoalan sebagai berikut:

1. Terdapat sejumlah berhingga pilihan yang mungkin.
2. Solusi tahap sebelumnya dijadikan untuk membangun solusi tahap berikutnya.
3. Untuk membatasi sejumlah pilihan yang perlu dipertimbangkan pada suatu tahap, digunakan persyaratan optimasi dan kendala.

Perbedaaan algoritma greedy dan program dinamis terletak pada jumlah rangkaian keputusan yang dihasilkan, greedy hanya memiliki satu, sedangkan program dinamis memiliki lebih dari satu keputusan yang dipertimbangkan.

Pada program dinamis, rangkaian keputusan yang optimal diciptakan dengan menggunakan prinsip optimalitas, yaitu jika solusi total optimal, maka bagian solusi sampai tahap ke-k juga merupakan solusi optimal.

Prinsip optimalitas berarti bahwa jika kita bekerja dari tahak k ke tahap k+1, kita bisa menggunakan hasil optimal dari

tahap k tanpa perlu kembali ke tahap awal. Ongkos pada tahap k+1 = (ongkos hasil tahap k) + (ongkos tahap k ke tahap k+1).

Persoalan program dinamis memiliki karateristik sebagai berikut:

1. Persoalan dapat dibagi menjadi berbagai tahap-tahap, yang pada setiap tahapnya hanya dapat diambil satu keputusan.
2. Masing-masing dari tahap terdiri dari beberapa status yang berhubungan dengan tahap tersebut. Secara umum, status adalah bermacam kemungkinan masukan yang ada pada tahap tersebut.
3. Hasil oleh keputusan yang diambil pada setiap tahap kemudian diftransformasikan dari status yang berhubungan ke status berikutnya pada tahap selanjutnya.
4. Ongkos pada suatu tahap meningkat secara teratur dengan bertambahnya jumlah tahapan.
5. Ongkos pada suatu tahap bergantung terhadap ongkos tahap-tahap yang sudah berjalan dan juga ongkos pada tahap tersebut sendiri.
6. Keputusan yang terbaik pada suatu tahap memiliki sifat independen terhadap keputusan yang sudah dilakukan pada tahap sebelumnya.
7. Adanya hubungan rekursif yang mengidentifikasikan keputusan terbaik untuk setiap status pada tahap k yang memberikan keputusan terbaik untuk setiap status pada tahap k+1.
8. Prinsip optimalitas akan berlaku pada persoalan tersebut

Ada 2 pendekatan program dinamis, program dinamis maju dan program dinamis mundur

B. Penjajaran Sekuens

Sekuens atau sekuens biologis, pada bioinfomatika merupakan molekul tunggal, kontinu dari protein atau asam nukleat.

Penyejajaran sekuens adalah proses untuk menyusun 2 atau lebih sekuens sehingga sekuens-sekuens tersebut memeliki hubungan fungsional, struktural atau evolusioner. Hasil proses tersebut juga disebut sebagai *sequence alignment* atau *alignment.* Baris sekuens dalam suatu *alignment* diberi sisipan sedemikian rupa sehingga kolomnya memuat karakter yang identik atau serupa di antara sekuens-sekuens tersebut.

Berikut adalah contoh *aligntment* dari dua sekuens pendek DNA yang berbeda:

```
ccat---caac

|  ||    ||||

caatgggcaac
```

Sequence alignment adalah metode dasar dalam analisis sekuens. Metode ini digunakan untuk mempelajari evolusi sekuens-sekuens dari leluhur yang sama (common ancestor). Ketidakcocokan (mismatch) pada alignment diasosiasikan dengan proses mutasi, sedangkan kesenjangan diasosiasikan dengan proses insersi atau delesi. *Sequence alignment* memberikan hipotesis atas proses evolusi yang terjadi dalam sekuens-sekuens tersebut. Maka pada contoh sekuens sebelumnya, bisa saja berevolusi menjadi sekuens yang sama. *Alignment* dapat juga menunjukan posisi yang dipertahankan (conserved) selama evolusi yang telah terjadi dalam sekuens-sekuens protein, yang berarti bahwa posisi-posisi tersebut bisa jadi penting bagi struktur atau fungsi protein tersebut.

Gambar 1. Contoh sekuens lokal.

C. Algoritma Smith-Waterman

Algoritma Smith-waterman pertama ditemukan oleh Temple F. Smith dan Michael S. Waterman pada 1981. Algoritma ini melakukan penjajaran sekuens lokal, dengan memberikan daerah yang sama diantara 2 sekuens, menyejajarkan 2 sekuens yang sebagian sama, dan juga dapat menjajarkan 2 subsekuens ke sekuens itu sendiri.

Algoritma ini menggunakan algoritma Needleman-Wunsch sebagai dasar, yang dimana algoritma tersebut menyejajarkan sekuesn global. Kedua algoritma ini menggunakan teknik program dinamis. Perbedaan dari algoritma Needleman-Wunsch adalah algoritma Smith-Waterman mencari *best local alignment* yaitu kecocokan substring pada 2 sekuens, sedangkan algoritma Needleman-Wunsch mencari *best global alignment* yaitu kecocokan dari panjang ujung ke ujung suatu sekuens yang terlibat.

Program dinamis yang digunakan untuk mencari *aligment* optimal pada 2 sekuens menggunakan nilai (*scores*) untuk setiap kecocokan dan ketidakcocokan pada matriks nilai (*scoring matrices)*. Dengan mencari nilai tertinggi pada matriks, *alignment* dapat secara akurat ditemukan.

Langkah dasar untuk algoritma Smith-Waterman adalah:

1. Inisialisasi sebuah matriks.
2. Mengisi matriks dengan nilai yang sesuai.
3. Melacak kembalik sekuens yang memiliki *alignment* yang sesuai.

Matriks nilai dibuat dengan 2 sekuens yang disusun menjadi kolom A+1 dan baris B+1, kemudian langkah yang penting adalah mengisi seluruh isi matriks, jadi penting untuk mengetahui nilai sel tetangga dari diagonal, atas, dan kiri untuk mengisi setiap sel.

$$M_{i,j} = Maximum\ [M_{i-1,j-1} + S_{i,j},\ M_{i,j-1} + W,\ M_{i-1,j} + W,\ 0]$$

Dimana i,j adalah baris dan kolom, M adalah matriks nilai yang dibutuhkan sel, S adalah nilai yang dibutuhkan Sel, W adalah celah *alignment.* Dapat diperhatikan dari rumus di atas, algoritma Smith-Waterman sel minimal bernilai 0.

Setelah mengisi matriks, tetapkan pointer ke sel sebelumnya dimana nilai maksimum telah ditentukan, dengan cara yang serupa saat mengisi seluruh nilai matriks pada setiap sel.

Langkah terakhir untuk *alignment* yang sesuai adalah *trace backing*, sebelum itu perlu diketahui nilai maksimum yang diperoleh pada seluruh matriks untuk *alignment* lokal dari sekuens. Mungkin untuk nilai maksimum dapat muncul di lebih dari satu sel, dalam kasus tersebut ada kemungkinan ada 2 atau lebih *alignment*, dan *alingment* terbaik dengan menilainmya.

Pada saat menghitung *scoring matriks* terdapat *gap penalty* untuk menentukan nilai insersi atau delesi, digunakan untuk mendapat celah yang panjang daripada menyebar.

Pseudo code untuk algoritma Smith-Waterman adalah:

Input: 2 sekuens X dan Y

Output: lokal *alignment* dan nilai a

Inisialisasi:

Set M(i, 0) := 0 untuk semua

i = 0, 1, 2, . . ., n

Set M(0, j) := 0 untuk semua

j = 0, 1, 2, . . ., n

For i = 1, 2, . . . , n do:

For j = 1, 2, . . . , n do:

Set M(i, j) = Max[0, M(i − 1, j − 1) + s(xi, yj), M(i − 1, j) + w, M(i, j − 1) + w]

Set backtrace T(i, j) to the maximizing pair (i` , j`) Set (i, j) := arg max{M(i, j) | i = 1, 2, . . . , n, j = 1, 2, . . . , m} The best score is α := M(i, j)

repeat

if T(i, j) = (i − 1, j − 1)

print(x_{i-1}, y_{j-1})

else if T(i, j) = (i − 1, j)

print (x_{i-1}, −)

else

print (−, y_{j-1})

Set (i, j) := T(i, j)

until M(i, j) = 0.

III. ANALISIS PENCOCOKAN SEKUENS

Dengan mencocokan sekuens DNA dari makhluk hidup kita dapat menentukan hubungan evolusioner dari genom spesies yang berbeda, jika banyak sekuens yang cocok maka spesies tersebut memiliki nenek moyang umum yang relatif tidak terlalu jauh waktunya, sedangkan jika sedikit kecocokan maka menunjukan bahwa perbedaan tersebut lebih kuno.

Contoh 1. Misal ada 2 sekuens:

1. G C A T C T G A
2. T C A T C A C T

maka pertama kita akan menginisialisasikan sebuah matriks dan mengisi nilai colom dan baris pertama dengan 0 dan menetukan nilai *match, mismatch, dan gap*.

Match = 3

Mismatch = -2

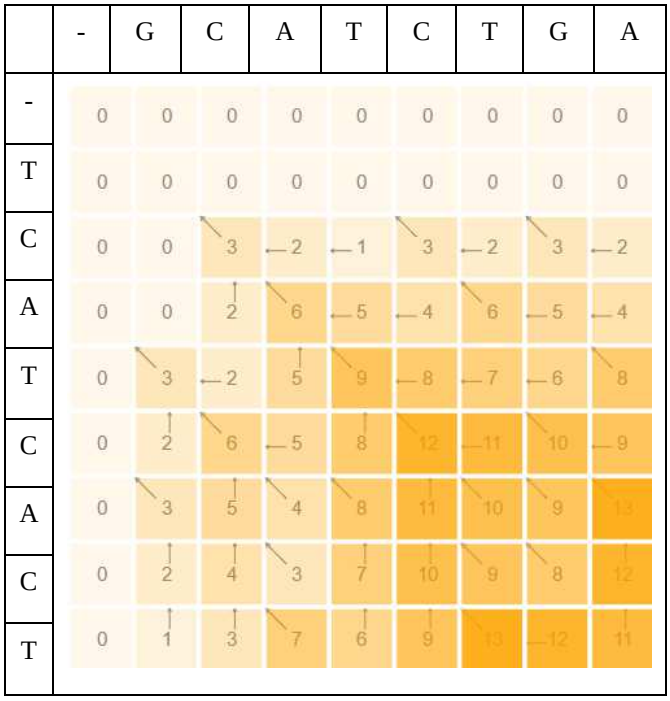
Gap = -1

	-	G	C	A	T	C	T	G	A
-	0	0	0	0	0	0	0	0	0
T	0								
C	0								
A	0								
T	0								
C	0								
A	0								
C	0								
T	0								

Kemudian akan dilakukan pengisian nilai dengan menggunakan rumus.

	-	G	C	A	T	C	T	G	A
-	0	0	0	0	0	0	0	0	0
T	0	0	0	0	3	2	3	2	1
C	0	0	3	2	2	6	5	4	3
A	0	0	2	6	5	5	4	3	7
T	0	0	1	5	9	8	8	7	6
C	0	0	3	4	8	12	11	10	9
A	0	0	2	6	7	11	10	9	13
C	0	0	3	5	6	10	9	8	12
T	0	0	2	4	8	9	13	12	11

Langkah berikutnya, berikan panah yang menunjuk asal nilai maksimum dari sebuah sel.



Setelah itu menentukan nilai maksimum dari matriks.

	-	G	C	A	T	C	T	G	A
-	0	0	0	0	0	0	0	0	0
T	0	0	0	0	3	2	3	2	1
C	0	0	3	2	2	6	5	4	3
A	0	0	2	6	5	5	4	3	7
T	0	0	1	5	9	8	8	7	6
C	0	0	3	4	8	12	11	10	9
A	0	0	2	6	7	11	10	9	**13**
C	0	0	3	5	6	10	9	8	12
T	0	0	2	4	8	9	13	12	11

Dan

	-	G	C	A	T	C	T	G	A
-	0	0	0	0	0	0	0	0	0
T	0	0	0	0	3	2	3	2	1
C	0	0	3	2	2	6	5	4	3
A	0	0	2	6	5	5	4	3	7
T	0	0	1	5	9	8	8	7	6
C	0	0	3	4	8	12	11	10	9
A	0	0	2	6	7	11	10	9	13
C	0	0	3	5	6	10	9	8	12
T	0	0	2	4	8	9	**13**	12	11

Kemudian kita melakukan *trace back* dari posisi yang paling besar, menunjuk kembali menggunakan pointer, dan menemukan sel sebelumnya sehingga mencapai nilai 0.

	-	G	C	A	T	C	T	G	A
-	0	0	0	0	0	0	0	0	0
T	0	0	0	0	3	2	3	2	1
C	0	0	3	2	2	6	5	4	3
A	0	0	2	6	5	5	4	3	7
T	0	0	1	5	9	8	8	7	6
C	0	0	3	4	8	12	11	10	9
A	0	0	2	6	7	11	10	9	13
C	0	0	3	5	6	10	9	8	12
T	0	0	2	4	8	9	13	12	11

Dan karena ada 2 nilai makismal maka kita dapat menentukan 2 *best local alignment*

	-	G	C	A	T	C	T	G	A
-	0	0	0	0	0	0	0	0	0
T	0	0	0	0	3	2	3	2	1
C	0	0	3	2	2	6	5	4	3
A	0	0	2	6	5	5	4	3	7
T	0	0	1	5	9	8	8	7	6
C	0	0	3	4	8	12	11	10	9
A	0	0	2	6	7	11	10	9	13
C	0	0	3	5	6	10	9	8	12
T	0	0	2	4	8	9	13	12	11

Maka *alignment* lokal telah didapatkan dan semua *alignment* yang mungkin adalah:

3	6	9	12	11	10	13
C	A	T	C	T	G	A
C	A	T	C	-	-	A

```
C A T C T G A
| | | |     |
C A T C - - A
```

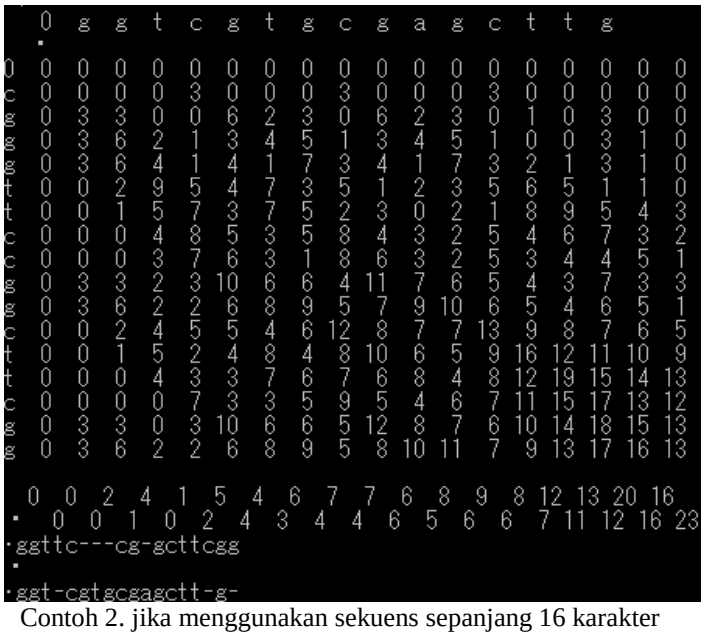
Dengan 5 kecocokan

Dan

3	6	9	8	7	10	13
C	A	T	-	-	C	T
C	A	T	C	A	C	T

```
C A T - - C T
| | |     | |
C A T C A C T
```

Dengan 5 kecocokan



Contoh 2. jika menggunakan sekuens sepanjang 16 karakter dengan menggunakan gap penalty dan extension.

IV. KESIMPULAN

Sifat algoritma program dinamis yang mampu mengoptimisasi dalam memecahkan masalah dan memberikan hasil yang optimal membuat program dinamis dapat dipakai untuk penyejajaran sekuen karena walaupun memiliki waktu kompleksitas komputasi yang tinggi sehingga tidak dapat digunakan pada skala yang besar, namun dapat menghasilkan hasil yang paling optimal. hanya itu, program dinamis pada bioinformatika juga dapat dipakai untuk pelipatan protein, prediksi struktur RNA, dan pengikatan protein-DNA. Selain bioinformatika, program dinamis juga dapat dipakai pada optimisasi matematika dan tentunya dalam pemrograman komputer.

V. UCAPAN TERIMA KASIH

Pertama, saya mengucapkan terima kasih kepada Allah SWT karena berkat rahmat dan izin-Nya lah penulis dapat menyelesaikan makalah ini. Selain itu saya juga mengucapkan terima kasih kepada orangtua yang selalu membantu, mendoakan, dan mendidik saya sehingga saya dapat menempuh ilmu di Institut Teknologi Bandung. Tak lupa ucapan terima kasih untuk Dr. Masayu Leylia Khodra ST,MT selaku dosen mata kuliah IF2211 Strategi Algoritma atas bimbingannya selama perkuliahan ini, sehingga saya dapat memperoleh berbagai ilmu dan juga dapat menyelesaikan makalah ini. Terima kasih juga saya ucapkan kepada teman saya yang telah membantu dalam penulisan makalah ini

VI. REFERENSI

[1] Munir, Rinaldi. 2015. Slide Kuliah IF2211 Strategi Algoritma : Program Dinamis (Dynamic Programming) (2015). Bandung : Institut Teknologi Bandung.

[2] http://vlab.amrita.edu/?sub=3&brch=274&sim=1433&cnt=1, diakses pada tanggal 15 April 2017

[3] https://cs.stanford.edu/people/eroberts/courses/soco/projects/computers-and-the-hgp/smith_waterman.html, diakses pada tanggal 15 April 2017

[4] http://bioinformatika-q.blogspot.co.id/2016/08/penyejajaran-sekuens-sequence-alignment.html, diakses pada tanggal 15 April 2017

[5] https://www.slideshare.net/avrilcoghlan/the-smith-waterman-algorithm, diakses pada tanggal 15 April 2017

[6] http://fridolin-linder.com/2016/03/30/local-alignment.html, diakses pada tanggal 15 April 2017

[7] https://ab.inf.uni-tuebingen.de/teaching/ss08/gbi/script/chapter04-alignment.pdf, diakses pada tanggal 15 April 2017

[8] http://biit.cs.ut.ee/~vilo/edu/2002-03/Tekstialgoritmid_I/Loengud/Loeng3_Edit_Distance/bcorum_copy/seq_align4.htm, diakses pada tanggal 15 April 2017

[9] https://www.britannica.com/science/bioinformatics, diakses pada tanggal 15 April 2017

[10] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3223492/, diakses pada tanggal 15 April 2017

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 15 Mei 2012

Afif Bambang P.
13515068