

Penerapan Algoritma Brute Force Pada Pencarian Variasi Fragment DNA

Ivan Pradhana Harka

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Jl. Ganesha 10, Bandung
e-mail: if16078@students.if.itb.ac.id

ABSTRAK

Deoxyribonucleic Acid atau yang biasa disingkat DNA adalah asam nukleat yang mengandung instruksi genetik yang digunakan dalam pengembangan dan penggunaan dari suatu organisme hidup dan beberapa virus yang ada di dunia ini. Fungsi utama dari DNA adalah tempat penyimpanan informasi jangka panjang mengenai pemiliknya. Dalam perkembangan bidang teknologi rekayasa genetik hingga sekarang telah tercapai beberapa teknologi seperti : *cloning*, perpustakaan berbasis genom, *Recombinant DNA technolog*, dan *DNA mapping*. Pada dasarnya teknologi-teknologi tersebut menggunakan metode *Restriction Digest* atau *DNA fragmentation* yang menggunakan *Restriction Enzyme* dalam prosesnya. Metode tersebut dengan kata lain adalah membuat sebuah DNA menjadi pecahan-pecahan atau *fragment*. *Restriction Enzyme* itu sendiri juga terdiri atas beberapa macam jenis enzim yang spesifik dan fungsi yang berbeda-beda. Pada makalah ini akan dibahas mengenai penerapan algoritma *brute force* dalam mencari varian solusi pemecahan DNA. Dimana algoritma *brute force* adalah algoritma yang paling dasar dan biasanya jika dilihat dalam segi waktu bukanlah algoritma yang paling efisien.

Kata kunci: DNA, *Restriction Digest*, *Brute Force*.

1. PENDAHULUAN

DNA adalah sebuah asam nukleat yang didalamnya terdapat kode genetik yang menyimpan berbagai informasi dari makhluk hidup yang memilikinya. DNA sering dibandingkan dengan cetak biru atau resep, karena mengandung instruksi yang dibutuhkan untuk membangun komponen-komponen sel, seperti protein dan molekul RNA.

Dengan berkembangnya ilmu rekayasa genetik sekarang ini terdapat beberapa teknologi yang dapat digunakan dalam merekayasa komposisi maupun karakteristik dari suatu DNA. Teknologi-teknologi seperti *cloning*,

perpustakaan berbasis genom, *Recombinant DNA technolog*, dan *DNA mapping* menggunakan metode *Restriction Digest* atau *DNA Fragmentation* pada dasar pengerjaannya.

Pada proses *Restriction Digest*, suatu DNA akan dipotong menjadi beberapa bagian di tempat-tempat yang mempunyai kecerendungan tinggi untuk terpisah dengan menggunakan *Restriction Enzyme*. Dari beberapa tempat tersebut akan timbul beberapa kemungkinan dari bermacam-macam potongan DNA yang mempunyai panjang yang berbeda-beda.

Masalah dari Metode *Restriction Digest* ini adalah menentukan banyaknya jumlah varian potongan DNA yang dapat terbentuk. Untuk menanggulangnya digunakanlah algoritma *brute-force* untuk menentukan panjang dan banyaknya jumlah varian potongan DNA.



Gambar 1. Contoh potongan DNA dan tempat pemotongan

2. Algoritma *Brute Force*

Algoritma adalah suatu pendekatan yang langsung untuk memecahkan suatu masalah, biasanya pada pernyataan masalah dan definisi konsep yang dilibatkan. Algoritma *brute force* memecahkan masalah dengan sangat sederhana, langsung dengan cara yang jelas. Dimana algoritma ini menelusuri segala variasi yang mungkin muncul, efektif pada beberapa kasus yang jarang, dan jarang diimplementasikan.

Kita misalkan bahwa X adalah himpunan integer sebanyak n buah yang merepresentasikan tempat potongan di daerah restriksi, termasuk awal hingga akhir potongan. n adalah banyaknya jumlah potongan. L himpunan yang berisi panjang potongan DNA. DX adalah himpunan integer yang berisi panjang dari tiap-tiap potongan DNA yang mungkin dilihat dari X .

Algoritma ini membandingkan tiap L yang didapat dengan DX , apabila ada suatu ketidakcocokan maka data

tersebut bukan merupakan bagian dari DX. Pertama-tama mengurutkan nilai masukan array L. Kemudian mengkalkulasikan nilai set DX dari nilai-nilai yang dimiliki oleh X. Setelah itu algoritma akan menyusun DX yang terbentuk dari proses perhitungan sebelumnya, dan terakhir DX dan L akan dicocokkan nilainya.

Algoritma ini memakan waktu sebesar $O(M^{n-2})$ dikarenakan algoritma ini harus menelusuri semua set dari posisi yang ada.

2.1 Alur Algoritma

1. BruteForcePDP(L, n):
2. $M \leftarrow$ maximum element in L
3. for every set of $n - 2$ integers $0 < x_2 < \dots < x_{n-1} < M$
4. $X \leftarrow \{0, x_2, \dots, x_{n-1}, M\}$
5. Form DX from X
6. if $DX = L$
7. return X
8. output "no solution"

2.2 Penerapan

Dimisalkan ada pita DNA yang memiliki kecerendungan membelah pada titik tertentu. $X = \{0, 5, 14, 19, 22\}$. Dimana angka tersebut adalah titik dimana pita mungkin membelah.

Kemudian L yang berisi $\{9, 5, 5, 3\}$ akan dicocokkan dengan kemungkinan dari DX yang berdasarkan X. Dimana pembuatan DX dapat dilihat dari tabel dibawah.

Tabel 1 Tabel DX dari X

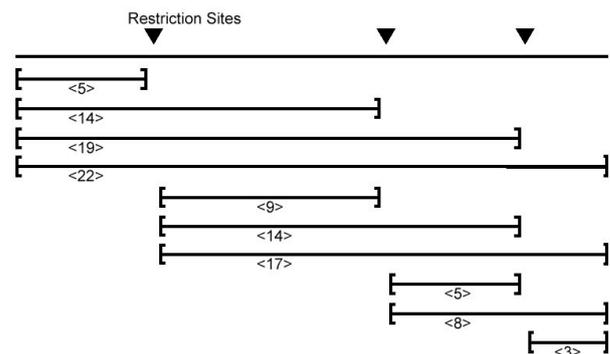
x	0	5	14	19	22
0	-	5	14	19	22
5	-	-	9	14	17
14	-	-	-	5	8
19	-	-	-	-	3
22	-	-	-	-	-

Berdasarkan nilai tabel diatas, maka $DX = \{5, 14, 19, 22, 9, 14, 17, 5, 8, 3\}$. Nilai ini akan dibandingkan dengan dengan nilai L untuk mendapatkan hasil solusi.

$$L = \{9, 5, 5, 3\}$$

$$X = \{0, 5, 14, 19, 22\}$$

$$DX = \{5, 14, 19, 22, 9, 14, 17, 5, 8, 3\}$$



Gambar 2. beberapa sampel potongan DNA

Setelah ke DX dan L di sort. Keduanya akan disamakan. Apakah L termasuk dari himpunan yang terdapat di DX.

Variansi $\{5, 9, 5, 3\}$ akan terbentuk apa kita tepat memotong pita DNA di titik 5, 14, 19, dan 22. dimana ini termasuk didalam DX.

Dalam kasus ini L termasuk di dalam DX. maka L adalah varian yang mungkin dalam pembelahan pita DNA.

IV. KESIMPULAN

Algoritma ini cukup efisien dalam pembuatan pencarian variasi dari potongan DNA. Ditambah lagi algoritma ini gampang untuk dimengerti.

Untuk saran, lebih baik apabila dibuat algoritma jenis lain dalam mengatasi hal yang sama. Maka kita dapat membandingkan antara algoritma lain dengan algoritma brute force dalam mengatasi masalah ini. Walaupun lebih pada pembahasannya nanti masih akan lebih efisien brute force.

REFERENSI

- [1] Hamilton, Smith, "an Introduction to Bioinformatics algorithms", www.bioalgorithms.info, Diakses 16 April 2006 pukul 21.00 WIB,
- [2] Munir, Rinaldi, "Strategi Algoritmik", Institut Teknologi Bandung, 2006.