

Penerapan Algoritma *Dijkstra* Untuk Mencari Jalur Reaksi Metabolisme Terefektif Pada Jaringan Reaksi Biokimia

Nicholas Rianto Putra / 13516020
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13516020@std.stei.itb.ac.id, nicholasmagbanua@gmail.com

Abstract—Reaksi metabolisme adalah suatu reaksi biokimia dari kumpulan enzim yang dilakukan oleh makhluk hidup untuk merubah suatu senyawa menjadi senyawa yang dituju. Saat ini, dalam mempelajari reaksi metabolisme pada jaringan reaksi biokimia yang kompleks, mahasiswa atau peneliti sangat bergantung pada metode komputasi. Metode komputasi yang dimaksud ada visualisasi reaksi metabolisme tersebut. Visualisasi reaksi biokimia sangatlah penting karena dapat mempermudah peneliti atau mahasiswa saat ingin melakukan sintesis suatu zat dan sebagai salah satu cara pemodelan reaksi suatu senyawa baru yang ditemukan. Jaringan reaksi biokimia sangatlah luas, dimana untuk mendapatkan suatu senyawa, terdapat berbagai macam reaksi berbeda yang dapat dilakukan. Oleh karenanya, untuk mendapatkan suatu reaksi yang efektif dan efisien saat ingin menghasilkan senyawa tertentu, algoritma *dijkstra* dapat menjadi salah satu pendekatan yang tepat untuk menelusuri reaksi biokimia yang telah dibuat representasinya dalam bentuk graf.

Kata Kunci—visualisasi; bfs; *dijkstra*; metabolisme; biokimia; reaksi; graf; efektif; efisien;

I. PENDAHULUAN

Seiring dengan perkembangan zaman yang berjalan begitu cepat, para peneliti atau mahasiswa di bidang biokimia dituntut untuk terus mengembangkan dan menemukan hal-hal baru pada bidang biokimia. Salah satu topik atau kajian yang sering dipilih oleh peneliti dalam menemukan terobosan baru adalah metabolisme. Para peneliti berusaha memahami seluruh jaringan biokimia pada suatu sel hidup yang ada di dunia ini. Tentunya sangat sulit untuk menggambarkan dan mencatat seluruh jaringan biokimia ini tanpa adanya bantuan teknologi seperti komputer dalam melakukan analisis data reaksi yang sangat kompleks. Selain itu, pendekatan komputasi juga dilakukan dalam melakukan visualisasi untuk pemodelan dan simulasi reaksi metabolisme yang mereka temukan pada sel - sel yang diteliti.

Bentuk visualisasi yang paling mudah dipahami dan paling sering dipakai adalah graf seluruh reaksi metabolisme dalam bentuk seperti jaringan. Saat ini, graf yang tersedia masih dibuat manual oleh para peneliti dan masih belum mencakup semua reaksi yang ada.

Representasi jaringan reaksi metabolisme yang sangat luas ini membawa pada permasalahan pemahaman dalam

mereaksikan suatu senyawa untuk menjadi senyawa lain. Oleh karenanya, diperlukan suatu algoritma untuk melakukan pencarian reaksi-reaksi yang ada pada jaringan biokimia. Visualisasi dan algoritma pencarian ini sudah menjadi perhatian khusus di beberapa tahun terakhir.

Pada makalah ini, kita akan berfokus pada algoritma pencarian jalur reaksi suatu senyawa awal menjadi senyawa yang dituju. Graf yang dibuat memiliki spesifikasi dimana simpulnya adalah nama senyawa dan sisinya adalah reaksi senyawa tersebut.

Algoritma yang akan dipakai bernama *dijkstra*, suatu algoritma yang akan menelusuri seluruh tetangga dari simpul awal hingga mencapai simpul akhir dengan jarak terpendek.

II. LANDASAN TEORI

A. Teori Graf

Graf adalah suatu diagram yang terdiri dari kumpulan titik yang biasa disebut sebagai simpul, dan antar simpul ini dihubungkan oleh suatu garis yang biasa disebut sisi. Sebagai contoh, simpul – simpul ini bisa merepresentasikan sebagai kota – kota yang ada disuatu negara, dan garis – garis yang menghubungkan antar simpul sebagai jalan raya yang menghubungkan antar kota.

Suatu graf G biasa terdiri dari kumpulan simpul V dan kumpulan sisi E . Notasi graf yang biasa dipakai pada umumnya adalah :

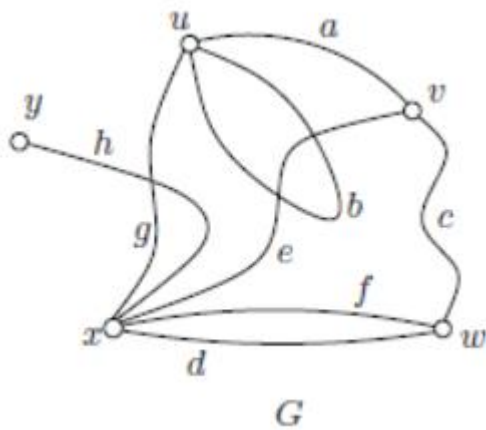
$$G = (V, E(G))$$

dimana

$$V(G) = \{\text{kumpulan simpul}\}$$
$$E(G) = \{\text{kumpulan sisi}\}$$

Alasan utama kenapa diagram ini disebut dengan kata “graf” adalah diagram ini divisualisasikan sebagai suatu bentuk grafis, dan bentuk grafis ini sangat membantu orang awam untuk memahami kondisi dari suatu masalah yang ada.

Berikut adalah contoh gambar dari graf tidak berarah dimana sisi – sisinya tidak memiliki arah, sehingga dapat dibilang bahwa tiap sisi graf memiliki dua arah :



Gambar 1 Contoh graf tidak berarah bernama G

Sumber : Graduate Texts in Mathematics : Graph Theory, 2008.

Pada setiap simpul yang ada pada graf, simpul – simpul lain yang berhubungan pada suatu simpul tertentu disebut dengan tetangga. Notasi kumpulan tetangga pada suatu simpul adalah $N_G(v)$.

Jalur pada graf adalah suatu upa graf yang merupakan bagian dari graf yang ada dimana graf sederhana tersebut merupakan simpul yang disusun dalam sekuens yang linear sehingga simpul awal dan simpul akhirnya menjadi awal dan akhir dari graf linear tersebut.

	a	b	c	d	e	f	g	h
u	1	2	0	0	0	0	1	0
v	1	0	1	0	1	0	0	0
w	0	0	1	1	0	1	0	0
x	0	0	0	1	1	1	1	1
y	0	0	0	0	0	0	0	1

	u	v	w	x	y
u	2	1	0	1	0
v	1	0	1	1	0
w	0	1	0	2	0
x	1	1	2	0	1
y	0	0	0	1	0

Gambar 2 Contoh Adjacency Matrices dan Incidency Matrices

Dalam perhitungan komputasi graf ini, graf yang merupakan representasi dari masalah yang akan diselesaikan, akan direpresentasikan lagi menjadi bentuk lain yang mempermudah komputasi. Umumnya, ada dua jenis representasi graf, yaitu :

1. Adjacency Matrices

Representasi graf ini dibuat menjadi matriks dengan baris dan kolom berisi simpul – simpul yang ada. Jika dua simpul berhubungan, maka sel dengan baris dan kolom simpul tersebut berisi nilai dari sisi yang menghubungkan kedua simpul tersebut.

2. Incidency Matrices

Representasi graf ini dibuat menjadi matriks dengan baris berisi simpul yang ada dan kolom berisi sisi – sisi yang ada pada graf. Jika simpul X terhubung dengan sisi E_1 maka nilai dari sel dengan baris dan kolom tersebut adalah besar jarak sisi tersebut.

Setelah mampu merepresentasikan graf ke salah satu bentuk umum diatas, maka kita dapat melakukan algoritma penelusuran graf tersebut. Ada dua jenis algoritma yang biasa dipakai untuk mencari jalur yang bisa dilalui untuk mencapai simpul tertentu dari simpul yang dipilih, yaitu *Breadth First Search* dan *Depth First Search*.

Pada makalah ini, algoritma yang akan dibahas adalah algoritma *Dijkstra*, yang merupakan pengembangan dari algoritma *Breadth First Search*, karena untuk mencari jalur reaksi terefektif dan terefisien, algoritma yang paling tepat dan mangkus adalah algoritma *Dijkstra*.

B. Breadth First Search

Algoritma *Breadth First Search* adalah algoritma pencarian berbasis pohon yang memiliki prinsip FSB (*First-served basis*), dimana dalam melakukan proses penelusuran, simpul simpul yang diproses terlebih dahulu adalah seluruh tetangga dari setiap simpul sekarang. Proses penelusuran akan berhenti ketika simpul menemukan simpul yang dituju, dan jalur yang dilalui merupakan jalur yang terpendek dan terfisien dan merupakan upagraf dari graf yang dipakai.

Berikut adalah *pseudocode* dari proses penelusuran berbasis pohon *Breadth First Search*:

Algoritma

```

set i:=0 dan queue Q:= ∅
increase ukuran i sebesar 1
warnai r dengan warna hitam
set l(r) := 0
set t(r) := i
masukkan simpul r kedalam queue Q
while Q tidak kosong do
  x := head Q
  if x punya tetangga y yang belum berwarna then
    increase nilai i sebesar 1
    warnai y dengan hitam
    set p(y):=x
    set l(y):=l(x)+1
    set t(y):=i
    masukkan y pada queue Q
  else
    buang x dari Queue
  end if
end while
return (p,l,t)

```

C. Algoritma Dijkstra

Algoritma BFS hanya dapat mencari jalur terpendek pada suatu graf jika seluruh nilai sisi yang ada pada graf tersebut bernilai sama. Tentunya pada reaksi metabolisme nilai *cost* dari setiap reaksi berbeda – beda sehingga kita harus menggunakan algoritma yang merupakan pengembangan dari algoritma BFS diatas.

Oleh karenanya, seorang ahli komputer bernama Edsger W. Dijkstra menemukan algoritma pengembangannya yang bernama algoritma *Dijkstra*, yaitu algoritma untuk mencari jalur terpendek antar simpul pada suatu graf. Jika pada algoritma BFS tetangga – tetangga tiap simpul disimpan pada suatu *queue*, pada algoritma *Dijkstra* ini disimpan pada suatu *priority queue*.

Langkah – langkah yang dilakukan pada algoritma *Dijkstra* dalam mencari jalur terpendek antara dua simpul adalah sebagai berikut :

1. Tandai semua simpul yang ada sebagai *unvisited*. Buatlah sebuah set bernama *unvisited set*.
2. Set semua nilai pada simpul dengan nilai infinit, dan nilai nol pada simpul awal.
3. Untuk simpul sekarang, cari semua tetangganya yang belum dikunjungi dan hitung jarak tentatif dari simpul sekarang. Bandingkan jarak tentatif hasil kalkulasi dengan nilai yang dipasang pada simpul tetangga. Jika nilai kalkulasi lebih rendah, maka pasang nilai pada simpul tersebut dengan nilai kalkulasi. Jika tidak, jangan lakukan perubahan nilai pada simpul tersebut.
4. Ketika kita sudah melakukan komparasi pada seluruh tetangga, maka tandai simpul sekarang sebagai *visited* dan buang dari *unvisited set*. Pindahlah ke simpul tetangga dengan nilai jarak tentatif terkecil, lalu ulangi langkah tiga pada semua tetangganya.
5. Jika simpul destinasi sudah ditandai atau jarak tentatif terkecil pada simpul di *unvisited set* adalah infinit, maka algoritma selesai.
6. Jika simpul destinasi belum ditandai, ulangi langkah 3 dengan simpul yang baru.

Berikut adalah *pseudo-code* dari algoritma *Dijkstra* yang akan dipakai untuk menelusuri jalur terpendek reaksi dari senyawa awal menjadi senyawa akhir:\

```

Algoritma
dist[source] := 0
create vertex set Q
for each simpul v pada Graf G:
    if v bukan source
        dist[v] := infinite
        prev[v] := undefined
        Q.add_prio_queue(v, dist[v])
while Q tidak kosong:
    x := head dari Q
    for each neighbor v pada x:
        tent := dist[x] + length(x, v)
        if tent < dist[v]:
            dist[v] := tent
            prev[v] := x
return dist, prev
    
```

D. Reaksi Metabolisme

Salah satu ciri suatu organisme dikatakan sebagai makhluk hidup adalah organisme tersebut kompleks dan struktur selnya sangat teratur. Karena adanya sel ini, maka tiap makhluk hidup akan mengalami reaksi metabolisme pada sel – sel yang menyusun tubuhnya.

Reaksi metabolisme adalah kumpulan reaksi kimia yang terjadi pada suatu sel, dimana reaktan, produk, dan zat -zat perantara akan bereaksi satu sama lain, dengan bantuan katalis berupa enzim.

Reaksi metabolisme sendiri memiliki dua arah lintasan reaksi kimia organik, yaitu :

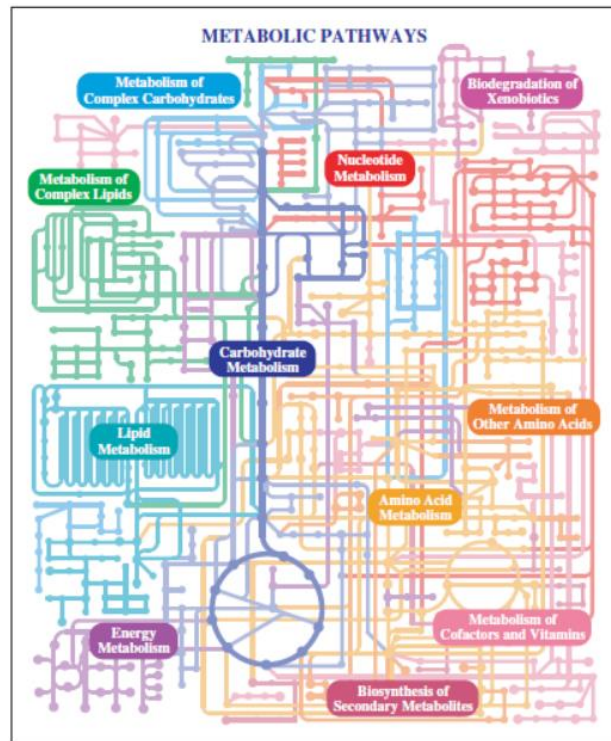
1. Katabolisme

Katabolisme adalah reaksi yang melakukan penguraian terhadap molekul senyawa yang organik untuk mendapatkan energi.

2. Anabolisme

Anabolisme adalah reaksi yang berlawanan dari katabolisme, dimana reaksi ini akan merangkai senyawa organik dari beberapa molekul agar senyawa tersebut dapat diserap oleh tubuh.

Berikut adalah contoh gambaran peta jaringan metabolisme yang ada pada makhluk hidup.



Gambar 3 Contoh jaringan metabolisme dari sel eukariotik yang memiliki 30,000 jenis protein yang berbeda.

Sumber : Lehninger's Principles of Biochemistry, 2004.

Gambar jaringan metabolisme diatas menandakan bahwa reaksi – reaksi yang terjadi pada sel saling berhubungan dan bergantung satu sama lain. Ada tiga pendekatan yang berbeda untuk memetakan seluruh reaksi metabolisme yang terjadi dari dasar. Perlu diingat bahwa ketiga pendekatan ini tidak memerlukan informasi yang mendetail mengenai enzim apa saja yang dipakai sebagai katalis, ataupun seberapa besar konsentrasi dari zat metabolites yang kita pakai, melainkan lebih ke penggambaran apakah mungkin kita bisa menghasilkan suatu zat Y dari mereaksikan suatu zat X berdasarkan peta jaringan yang sudah kita gambarkan.

Pendekatan pertama yang dapat dilakukan untuk mendapatkan peta tersebut dengan sintesis. Analisis dari hasil sintesis suatu senyawa menjadi senyawa lain dapat dilakukan secara sistematis.

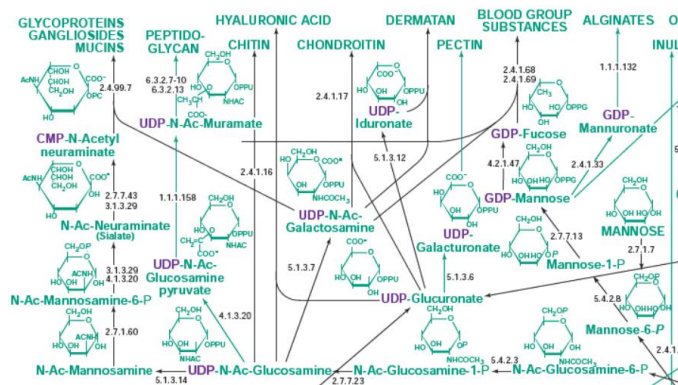
Pendekatan kedua adalah pembuatan peta jaringan analisis berdasarkan konsep biokimia dasar, dimana pendekatan yang kedua ini lebih berfokus pada apakah suatu senyawa bisa dihasilkan dari senyawa tertentu berdasarkan teori – teori reaksi yang sudah ada saat ini.

Pendekatan ketiga adalah melakukan analisis komparasi gen, yang terdiri dari DNA dan RNA. Dengan melakukan komparasi gen antara senyawa – senyawa yang ada, kita dapat memprediksi hasil reaksi yang akan terjadi dan komparasi ini dapat mengidentifikasi inversi, translokasi, dan duplikasi yang terjadi pada senyawa – senyawa yang kita uji.

Struktur jaringan metabolisme skala besar yang telah dibuat sejauh ini, jika kita observasi lebih jauh, kita akan melihat bahwa jaringan metabolisme ini saling terhubung sebagai suatu jaringan skala bebas yang berarti adanya penurunan yang eksponensial terhadap jumlah beberapa metabolite sentral yang terhubung.

Selain itu, dari peta jaringan tersebut, kita dapat melihat jalur – jalur reaksi apa saja yang tersedia untuk menghasilkan suatu senyawa, dan apa yang terjadi jika enzim berbeda dipakai sebagai katalis dari suatu reaksi. Kita juga bisa melihat jenis reaksi yang ada pada peta tersebut, seperti reaksi biokimia yang *reversible*, yaitu suatu reaktan dapat menjadi produk dan sebaliknya, dan reaksi *irreversible* yang merupakan reaksi satu arah, dimana suatu reaktan bisa menjadi produk, tetapi produk tidak bisa kembali menjadi reaktan yang dipakai.

Untuk mendukung keberjalanan peneliti dalam melakukan penelitian serta mencari jalur reaksi senyawa – senyawa yang ada, dibutuhkan bantuan komputasi oleh komputer. Biokimia komputasi sudah menjadi kebutuhan primer bagi peneliti sekarang, sehingga tidak hanya melakukan penelitian terhadap hal – hal baru yang ada pada biokimia itu sendiri, perlu dilakukan penelitian juga terhadap algoritma – algoritma yang diperlukan untuk mensokong kebutuhan komputasi ini.



Gambar 4 Contoh detail dari peta jaringan metabolisme yang ditemukan oleh para peneliti.

Sumber : International Union of Biochemistry and Molecular Biology, 2003.

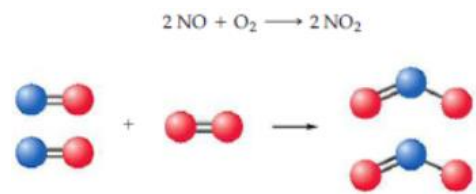
Seperti pada peta jaringan metabolisme ini, dibutuhkan suatu algoritma yang mampu mencari jalur reaksi antar senyawa sehingga dapat ditemukan jalur terefektif dan terefisien. Oleh karenanya, salah satu solusinya adalah merepresentasikan peta

jaringan ini menjadi bentuk graf dimana simpulnya adalah nama suatu zat, dan sisinya adalah reaksi yang akan dilakukan.

Untuk menjalankan algoritma *Dijkstra* pada representasi graf peta jaringan metabolisme, kita harus mengetahui cara menentukan nilai *cost* dari tiap reaksi yang ada pada peta tersebut. Oleh karenanya, setelah ini akan dibahas mengenai teknik – teknik yang digunakan untuk kalkulasi *cost* tiap sisi pada graf tersebut.

E. Rantai Reaksi Kimia

Reaksi kimia adalah proses dimana suatu set substansi yang biasa disebut reaktan, diubah menjadi substansi baru bernama produk. Dalam kata lain, reaksi kimia adalah proses dimana terjadi perubahan pada suatu zat kimia. Dalam memvisualisasikan reaksi kimia yang terjadi kita dapat menggunakan bantuan persamaan kimia. Berikut adalah contoh dari reaksi kimia :



Gambar 5 Contoh persamaan suatu reaksi kimia

Sumber : General Chemistry : Principles and Modern Application, 2011

Kita tahu bahwa pada peta jaringan metabolisme, satu senyawa dapat bereaksi dengan berbagai macam senyawa sehingga terdapat banyak jalur untuk mendapatkan suatu senyawa. Pada dasarnya, terdapat dua jenis pendekatan dalam menentukan seberapa besar kemungkinan reaksi – reaksi yang dapat terjadi, yaitu :

1. Termodinamis
Secara termodinamis, reaksi yang lebih mungkin untuk terjadi adalah reaksi yang memiliki nilai entalpi lebih negatif. Artinya adalah nilai entalpi produk jauh lebih rendah dari entalpi reaktan.
2. Kinetik
Secara kinetik, reaksi yang lebih mungkin untuk terjadi adalah reaksi yang memiliki energi aktivasi paling rendah.

Pada makalah ini, metode pendekatan yang akan dipakai untuk mencari *cost* suatu reaksi adalah metode pendekatan Kinetik.

III. PENERAPAN ALGORITMA DIJKSTRA PADA PENENTUAN JALUR REAKSI TEREFEKTIF PADA PETA JARINGAN REAKSI METABOLISME

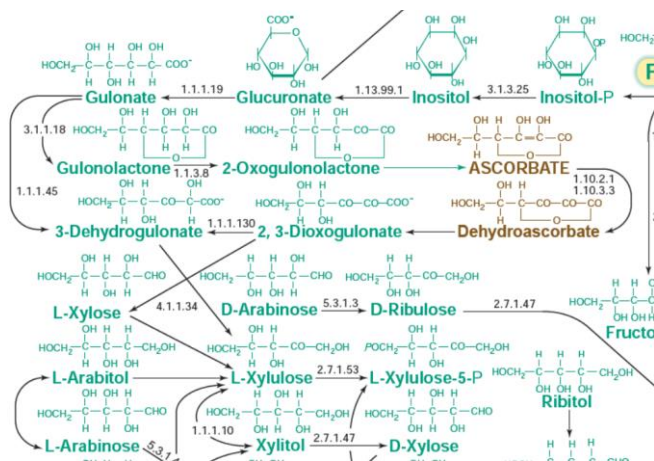
Pada bagian penerapan algoritma *Dijkstra* ini, akan dibagi menjadi dua bagian utama. Bagian yang pertama adalah mengubah sebagian contoh dari peta jaringan metabolisme yang ada ke dalam bentuk graf, dan melakukan kalkulasi nilai dari *cost* tiap reaksi yang ada.

Bagian yang kedua adalah implementasi dari algoritma *Dijkstra* pada graf jaringan reaksi metabolisme yang telah kita

buat pada bagian pertama, sehingga kita dapat membantu para peneliti ataupun mahasiswa dalam menentukan reaksi yang tepat dan efektif saat ingin melakukan sintesis suatu senyawa menjadi senyawa lain.

A. Pembuatan Graf dan Kalkulasi Nilai dari tiap sisi

Pertama – tama, kita harus mengubah bentuk peta jaringan reaksi metabolisme menjadi dalam bentuk graf, dimana simpul adalah senyawa dan sisi merupakan energi aktivasi yang dibutuhkan untuk melakukan reaksi senyawa asal menjadi senyawa tujuan. Berikut adalah sebagian peta jaringan metabolisme yang akan kita jadikan contoh implementasi algoritma dijkstra untuk mencari jalur reaksi tercepat dari senyawa asal ke senyawa akhir.



Gambar 6 Sebagian peta jaringan reaksi metabolisme yang akan dipakai untuk contoh implementasi algoritma

Sumber : International Union of Biochemistry and Molecular Biology, 2003.

Sebelum mengubah gambar diatas menjadi graf, kita harus melakukan kalkulasi terlebih dahulu mengenai *cost* tiap reaksi yang akan dipakai. *Cost* yang dipakai pada reaksi ini adalah nilai dari energi aktivasi tiap reaksi. Untuk menghasilkan reaksi terefektif, kita perlu mencari reaksi dengan energi aktivasi terendah, karena semakin rendah nilai energi aktivasi suatu reaksi, maka makin tinggi nilai laju reaksinya sehingga reaksi berjalan lebih cepat.

Untuk menghitung energi aktivasi, kita dapat menggunakan persamaan Arrhenius. Seiring meningkatnya temperatur, kecepatan pergerakan molekul pada suatu gas juga meningkat. Hal ini berlaku juga pada senyawa yang padat dan cair. Energi kinetik sebuah molekul dapat dihitung dengan persamaan Arrhenius, yaitu :

$$k = Ae^{\frac{-E_a}{RT}}$$

dimana :

k adalah nilai *rate constant*

E_a adalah energi aktivasi

R adalah konstanta gas

T adalah suhu dalam Kelvin

A adalah Arrhenius faktor yang merupakan indikasi laju tumbukan dan pecahan dari tumbukan dengan orientasi yang tepat untuk suatu reaksi dapat terjadi

Dengan melakukan natural logaritma di kedua sisi persamaan, kita mendapatkan persamaan :

$$E_a = (\ln A - \ln k)RT$$

Selain dari persamaan Arrhenius, kita juga dapat mengkalkulasikan nilai energi aktivasi dengan menurunkan persamaan energi bebas Gibbs. Dari penurunannya, didapatkan persamaan :

$$E_a = \Delta H^\ddagger + RT$$

Berikut adalah representasi graf dari gambar peta jaringan diatas :



Pada gambar diatas, nilai *cost* yang diberikan hanya merupakan contoh, bukan dari hasil perhitungan asli nilai energi aktivasi, karena fokus pada makalah ini adalah menjelaskan mengenai implementasi algoritmanya dalam menentukan rute reaksi terefektif. Sekarang, setelah graf berbobot telah dibuat, kita bisa mengimplementasikan algoritma *dijkstra* pada proses pencarian rute reaksi metabolismenya.

B. Implementasi Algoritma Dijkstra pada penentuan jalur reaksi metabolisme

Dari graf yang telah kita buat sekarang kita akan mencoba mencari rute reaksi terefektif saat kita ingin melakukan reaksi kimia pada senyawa *Glucuronate* menjadi senyawa *L-Xylulose* pada peta tersebut.

Sesuai dengan langkah – langkah penerapan algoritma *dijkstra* yang telah dibahas pada landasan teori, maka hal pertama yang harus kita lakukan adalah membuat satu set bernama *unvisited set* yang berisi kumpulan simpul – simpul yang belum dikunjungi dalam proses penelusuran dari senyawa *Glucuronate* menjadi senyawa *L-Xylulose*.

Setelah membuat *unvisited set*, langkah selanjutnya adalah menelusuri seluruh tetangga yang ada pada simpul senyawa awal, yaitu *Glucuronate*. Karena graf yang kita buat merupakan Graf berarah dan berbobot, maka tetangga yang dimiliki oleh senyawa *Glucuronate* adalah senyawa *Gulonate*, yang memiliki energi aktivasi sebesar 5 satuan.

Untuk mempermudah perhitungan penelusuran algoritma ini, kita dapat membuat tabel sebagai salah satu cara memahami penelusurannya. Berikut adalah tabel akhir dari perhitungan algoritma *Dijkstra* yang dipakai.

Unvisited (Q)	Visited(S)	Cur	A	B	C	D	E	F	G	H	I	J
0 [A,B,C,D,E,F,G,H,I,J]	{}	-	[0,-]0	(∞,-)0	(∞,-)0	(∞,-)0	(∞,-)0	(∞,-)0	(∞,-)0	(∞,-)0	(∞,-)0	(∞,-)0
1 [B,C,D,E,F,G,H,I,J]	{A}	A	-	(5,A)1	(∞,A)1	(∞,A)1	(∞,A)1	(∞,A)1	(∞,A)1	(∞,A)1	(∞,A)1	(∞,A)1
2 [C,D,E,F,G,H,I,J]	{A,B}	B	-	-	(1,B)2	(∞,B)2	(∞,B)2	(∞,B)2	(∞,B)2	(31,B)2	(∞,B)2	(∞,B)2
3 [D,E,F,G,H,I,J]	{A,B,C}	C	-	-	-	(1,C)3	(∞,C)3	(∞,C)3	(∞,C)3	(∞,C)3	(∞,C)3	(∞,C)3
4 [E,F,G,H,I,J]	{A,B,C,D}	D	-	-	-	-	(1,D)4	(∞,D)4	(∞,D)4	(∞,D)4	(∞,D)4	(∞,D)4
5 [F,G,H,I,J]	{A,B,C,D,E}	E	-	-	-	-	-	(2,E)5	(∞,E)5	(∞,E)5	(∞,E)5	(∞,E)5
6 [G,H,I,J]	{A,B,C,D,E,F}	F	-	-	-	-	-	-	(2,F)6	(∞,F)6	(∞,F)6	(∞,F)6
7 [H,I,J]	{A,B,C,D,E,F,G}	G	-	-	-	-	-	-	-	(1,G)6	(3,G)6	(∞,G)7
8 [I,J]	{A,B,C,D,E,F,G,H}	H	-	-	-	-	-	-	-	-	(∞,H)7	(4,H)8

Keterangan :

A: Glucoronate, B: Gulonate, C: Gulonolactone, D: Oxogulonolactone, E: ASCORBATE, F: Dehydroascorbate, G: Dioxogulonate, H: Dehydrogulonate, I: L-Xylose, J: L-Xylulose

Dari hasil perhitungan diatas, berikut adalah gambar lintasan reaksi terpendek menurut algoritma *dijkstra*, dimana total energi aktivasi yang dibutuhkan adalah 17 satuan.



Gambar 7 Hasil dari algoritma Dijkstra dari senyawa Glucoronate menuju L-Xylulose

Seperti yang dilihat pada gambar diatas, meskipun jika dilihat secara kasat mata rute dengan jumlah simpul tersedikit adalah rute *Glucoronate – Gulonate – Dehydrogulonate – L-Xylulose*, ternyata rute tersebut bukanlah rute yang memiliki jumlah energi aktivasi terendah karena energi aktivasi yang dibutuhkan dengan rute tersebut adalah empat puluh satuan, sehingga jalur reaksi tercepat dari *Glucoronate* menjadi *L-Xylulose* berdasarkan penelusuran dengan algoritma *dijkstra* adalah rute *Glucoronate – Gulonate – Gulonolactone – Oxogulonolactone – ASCORBATE – Dehydroascorbate – Dioxogulonate – Dehydrogulonate – L-Xylulose*, dengan energi aktivasi sebesar tujuh belas satuan.

Ketika peta jaringan reaksi metabolisme yang berada pada gambar 3 dapat dibuat visualisasinya dalam bentuk graf lalu garis yang menghubungkan antar reaksi diberikan *cost* sesuai dengan energi aktivasi tiap reaksi, dan dibuat simulator visualisasi penelusuran jalur reaksinya yang tercepat, maka kegiatan dalam melakukan sintesis suatu senyawa menjadi jauh lebih efektif dan efisien daripada menggunakan cara konvensional yang melakukan penelusuran secara manual.

IV. KESIMPULAN

Penelitian mengenai peta jaringan reaksi metabolisme yang sangat kompleks ini memegang peranan penting dalam kemajuan ilmu di bidang biokimia, karena setiap sel pada makhluk hidup selalu menjalankan reaksi ini dan tentunya dapat menjadi salah satu kunci untuk membuat penemuan, inovasi, atau terobosan baru di bidang biologi.

Aplikasi algoritma *dijkstra* pada pencarian jalur terefektif pada peta jaringan reaksi metabolisme sangatlah penting karena dengan adanya visualisasi peta dalam bentuk graf serta bantuan komputasi reaksi dengan algoritma ini, maka pekerjaan yang dilakukan oleh peneliti atau mahasiswa menjadi jauh lebih mudah dan cepat. Selain itu, visualisasi ini juga membantu para peneliti atau mahasiswa menjadi lebih mudah dalam memahami jalannya reaksi antar senyawa, sehingga dapat melakukan eksperimen sintesis suatu senyawa pada reaksi metabolisme dengan baik dan benar.

Pencarian jalur reaksi metabolisme ini terdiri dari dua bagian utama, yaitu pengubahan peta jaringan reaksi metabolisme menjadi suatu graf berbobot dengan perhitungan kalkulasi *cost* tiap sisi yang tepat. Setelah pembuatan representasi graf ini, maka algoritma *dijkstra* dapat dijalankan pada graf untuk mencari jalur reaksi terefektif pada suatu peta jaringan reaksi – reaksi tersebut.

Dengan adanya perkembangan teknologi di bidang komputeasi yang sangat pesat, kita harus mampu memanfaatkannya sehingga perkembangan ini juga memberikan dampak positif pada pekerjaan atau penelitian yang kita lakukan ini.

V. APENDIKS

Reaction	ΔH (kJ)*
$\text{CH}_4(\text{g}) + 2\text{O}_2(\text{g}) \longrightarrow \text{CO}_2(\text{g}) + 2\text{H}_2\text{O}(\ell)$	-890.4
$\text{C}_3\text{H}_8(\text{g}) + 5\text{O}_2(\text{g}) \longrightarrow 3\text{CO}_2(\text{g}) + 4\text{H}_2\text{O}(\ell)$	-2219.2
$2\text{C}_8\text{H}_{18}(\ell) + 25\text{O}_2(\text{g}) \longrightarrow 16\text{CO}_2(\text{g}) + 18\text{H}_2\text{O}(\ell)$	-10943
$2\text{CH}_3\text{OH}(\ell) + 3\text{O}_2(\text{g}) \longrightarrow 2\text{CO}_2(\text{g}) + 4\text{H}_2\text{O}(\ell)$	-1452
$\text{C}_2\text{H}_5\text{OH}(\ell) + 3\text{O}_2(\text{g}) \longrightarrow 2\text{CO}_2(\text{g}) + 3\text{H}_2\text{O}(\ell)$	-1367
$\text{C}_6\text{H}_{12}\text{O}_6(\text{s}) + 6\text{O}_2(\text{g}) \longrightarrow 6\text{CO}_2(\text{g}) + 6\text{H}_2\text{O}(\ell)$	-2804
$2\text{CO}(\text{g}) + \text{O}_2(\text{g}) \longrightarrow 2\text{CO}_2(\text{g})$	-566.0
$\text{C}(\text{s}) + \text{O}_2(\text{g}) \longrightarrow \text{CO}_2(\text{g})$	-393.5
$4\text{Al}(\text{s}) + 3\text{O}_2(\text{g}) \longrightarrow 2\text{Al}_2\text{O}_3(\text{s})$	-3351
$\text{N}_2(\text{g}) + \text{O}_2(\text{g}) \longrightarrow 2\text{NO}(\text{g})$	+182.6
$\text{N}_2(\text{g}) + 2\text{O}_2(\text{g}) \longrightarrow 2\text{NO}_2(\text{g})$	+66.4
$2\text{H}_2(\text{g}) + \text{O}_2(\text{g}) \longrightarrow 2\text{H}_2\text{O}(\text{g})$	-483.6
$2\text{H}_2(\text{g}) + \text{O}_2(\text{g}) \longrightarrow 2\text{H}_2\text{O}(\ell)$	-571.6
$\text{N}_2(\text{g}) + 3\text{H}_2(\text{g}) \longrightarrow 2\text{NH}_3(\text{g})$	-91.8
$2\text{C}(\text{s}) + 3\text{H}_2(\text{g}) \longrightarrow \text{C}_2\text{H}_6(\text{g})$	-84.0
$2\text{C}(\text{s}) + 2\text{H}_2(\text{g}) \longrightarrow \text{C}_2\text{H}_4(\text{g})$	+52.4
$2\text{C}(\text{s}) + \text{H}_2(\text{g}) \longrightarrow \text{C}_2\text{H}_2(\text{g})$	+227.4
$\text{H}_2(\text{g}) + \text{I}_2(\text{g}) \longrightarrow 2\text{HI}(\text{g})$	+53.0
$\text{KNO}_3(\text{s}) \xrightarrow{\text{H}_2\text{O}} \text{K}^+(\text{aq}) + \text{NO}_3^-(\text{aq})$	+34.89
$\text{NaOH}(\text{s}) \xrightarrow{\text{H}_2\text{O}} \text{Na}^+(\text{aq}) + \text{OH}^-(\text{aq})$	-44.51
$\text{NH}_4\text{Cl}(\text{s}) \xrightarrow{\text{H}_2\text{O}} \text{NH}_4^+(\text{aq}) + \text{Cl}^-(\text{aq})$	+14.78

Appendiks 1 Tabel nilai Entalpi Reaksi Kimia

Feedstock	Catalyst	Catalyst loading ^a (wt%)	Temperature (°C)	Methanol/oil (mol/mol)	Reaction order	Activation energy (kJ/mol)	Reference
Sunflower oil	CaO-based PKS	3-7	45-65	6:1-18:1	Pseudo-first	108.8	This work
Palm oil	CaO-based woody biomass ash	1-10	45-70	6:1-40:1	Pseudo-first	83.9	[33]
Jatropha oil	Lemma perpusilla Torrey ash	1-10	65	9:1	Pseudo-first	29.49	[31]
Soybean oil	CaO ^b	1-7	40-65	6:1-18:1	Pseudo-first	82.3	[41]
Jatropha oil, crude	CaO ^b	1-7	40-65	6:1-18:1	Third	133.5	[42]
Soybean oil	CaO	2	40-60	12:1	Pseudo-first	81.09	[49]
Waste frying oil	CaO	2	50-65	6:03:1	Pseudo-first	79	[21]
Canola oil	CaO	2-4	50-65	6:1-12:1	Pseudo-first	104.83 (0-60 min)	[50]
						136.48 (60-120 min)	
Jatropha oil	CaO ^c	0.1-0.5	35-65	15:1	Pseudo-first	102.49	[51]
Waste cottonseed oil	Zn/CaO/NO	1-6	35-65	3:1-18:1	Pseudo-first	29.8	[52]
Waste cottonseed oil	Zn/CaO	5-20	35-75	3:1-18:1	Pseudo-first	41.2	[53]
Waste cottonseed oil	K/CaO	1-10	35-65	3:1-18:1	Pseudo-first	43	[53]
Soybean oil	K/CaO	7.5	35-65	12:1	Pseudo-first	54	[25]
Soybean oil	Cs/CaO	1-10	35-65	6:1-12:1	Second, reversible	77.92	[16]
Soybean oil	CaO/Al ₂ O ₃	1-6	125-200	6:1-12:1	Pseudo-first	30.7	[54]
Rapeseed oil	CaO/MgO	-	-	-	Second	29.31	[55]
Crude palm oil	CaO/Pty ash	2-8	45-65	6:1-18:1	Pseudo-first	42.56	[56]
Sunflower oil	CaO/ZnO	2	60-96	10:1	Pseudo-first	26.5	[26]
Used vegetable oil						16.6	

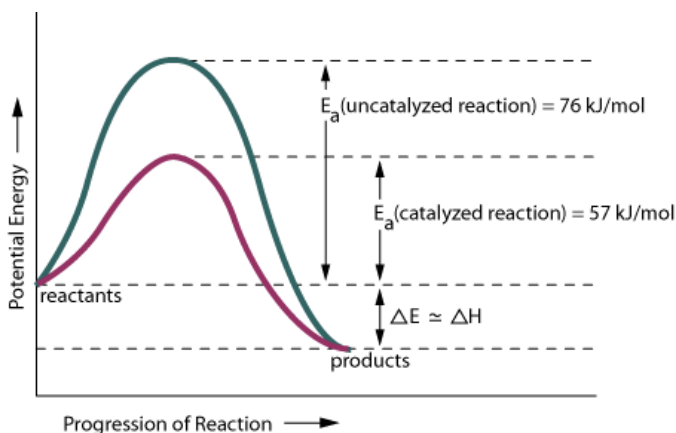
^a With respect to mass of oil.

^b Ultrasound-assisted.

^c Nano-CaO.

^d Higher surface area nano-CaO.

Apendiks 2 Tabel Energi Aktivasi beberapa senyawa beserta jenis katalisnya



Apendiks 3 Grafik energi suatu reaksi

VI. PENUTUP

Terima kasih kepada Tuhan Yang Maha Esa karena berkat rahmat-Nya yang melimpah, penulis dapat menyelesaikan makalah ini dengan baik. Terima kasih juga penulis ucapkan kepada orang tua serta teman-teman yang terus memberikan dukungan dan semangat kepada penulis untuk menyelesaikan makalah ini. Tidak lupa penulis mengucapkan terima kasih kepada Bu Ulfa, selaku dosen dari mata kuliah Strategi Algoritma Kelas dua. Penulis memohon maaf apabila ada kesalahan pada makalah ini. Kritik dan saran sangat dibutuhkan agar penulis dapat berkembang menjadi lebih baik lagi, sehingga mampu membuat makalah yang jauh lebih baik dari makalah ini.

REFERENSI

- [1] *Activation Energy*. (n.d.). Retrieved from Bodner Research Web: <http://chemed.chem.purdue.edu/genchem/topicreview/bp/ch22/activate.php#act>
- [2] Bondy, J. A., & Murty, U. S. (2008). *Graduate Texts in Mathematics : Graph Theory*. Lyon: Springer.

- [3] Bui, M., Kan, Kelvin, C. F., Le, S., & Tan, E. (2016, October 31). *The Arrhenius Law: Activation Energies*. Retrieved from Chemistry Libretexts: https://chem.libretexts.org/Core/Physical_and_Theoretical_Chemistry/Kinetics/Modeling_Reaction_Kinetics/Temperature_Dependence_of_Reaction_Rates/The_Arrhenius_Law/The_Arrhenius_Law%3A_Activation_Energies#Calculation_of_Ea_using_Arrhenius_Equation
- [4] Collado-Vides, J., & Hofstadt, R. (2002). *Gene Regulation and Metabolism : Postgenomic Computational Approaches*. Cambridge: The MIT Press.
- [5] Donald E. Nicholson, D. (2003). Metabolic Pathway Poster. *International Union of Biochemistry and*.
- [6] Kanehisa, M., & Goto, S. (2000). KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Research*.
- [7] McClean, P. (1998). *Comparative Genome Mapping*. Retrieved from Analysis of Plant Genomes with Molecular Markers: <https://www.ndsu.edu/pubweb/~mcclean/plsc731/analysis/analysis4.htm>
- [8] Nelson, D. L., & Cox, M. M. (2004). *Lehninger's Principles of Biochemistry*. New York: W.H. Freeman and Company.
- [9] Petrucci, R. H., Herring, F. G., Madura, J. D., & Bissonnette, C. (2011). *General Chemistry : Principles and Modern Applications*. Toronto: Pearson Canada Inc.
- [10] Planes, F., & Beasley, J. E. (2008). Path Finding Approaches and Metabolic Pathways. *Elsevier*.
- [11] R.H., G., & M., G. C. (1999). *Biochemistry*. Meadville: Cengage Learning.
- [12] *The chemical logic behind... the Pentose-phosphate Pathway*. (n.d.). Retrieved from Prof. Doutor Pedro Silva: <http://homepage.ufp.pt/pedros/bq/ppp.htm>
- [13] Wegner, K., & Kummer, U. (2005). A new dynamical layout algorithm for complex biochemical reaction networks. *BMC Bioinformatics*.

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, 12 Mei 2018

Nicholas Rianto Putra, 13516020