

# Penerapan Algoritma Koloni Lebah untuk Pewarnaan Graf

Riva Syafri Rachmatullah - 13512036  
Program Studi Teknik Informatika  
Sekolah Teknik Elektro dan Informatika  
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia  
13512036@std.stei.itb.ac.id

**Abstrak**—Makalah ini berisi tentang teori graf dan pewarnaan graf menggunakan algoritma lebah. Teori graf merupakan hal yang menarik untuk diangkat dalam bidang ilmu pengetahuan. Suatu graf adalah kumpulan dari simpul yang dihubungkan dengan sisi-sisi. Permasalahan yang cukup dikenal terkait dengan graf adalah pewarnaan graf. Pewarnaan graf dapat diselesaikan dengan algoritma yang ada, salah satunya dengan menggunakan pemanfaatan Algoritma Koloni Lebah. Algoritma Lebah adalah algoritma yang meniru perilaku lebah dan dapat digunakan untuk berbagai permasalahan Jaringan, Job Scheduling, Pewarnaan Graf, dan berbagai permasalahan lainnya

**Kata Kunci**—Pewarnaan Graf, Algoritma Koloni Lebah, Pseudo-code, Kehidupan lebah.

## I. PENDAHULUAN

Banyak sekali hewan yang ada di muka Bumi ini dan beberapa bahkan disebutkan secara eksplisit di Al-Qur'an seperti babi, lebah dan semut. Bahkan, semut dan lebah dijadikan nama surah di Al-Qur'an karena keunikannya yaitu surah An-Naml (semut) dan An-Nahl (lebah). Semut dan lebah selalu bekerja sama dan berkoloni sesamanya untuk mencapai tujuan bersama. Lebah juga memberikan madu yang bisa menjadi obat bagi manusia.

“Dan Tuhanmu mewahyukan kepada lebah: "Buatlah sarang-sarang di bukit-bukit, di pohon-pohon kayu, dan di tempat-tempat yang dibuat manusia. Kemudian makanlah dari tiap-tiap (macam) buah-buahan dan tempuhlah jalan Tuhanmu yang telah dimudahkan (bagimu)." Dari perut lebah itu keluar minuman (madu) yang bermacam-macam warnanya, di dalamnya terdapat obat yang menyembuhkan bagi manusia. Sesungguhnya pada yang demikian itu benar-benar terdapat tanda (kebesaran Tuhan) bagi orang-orang yang memikirkan.” (QS. An Nahl, 16 : 68-69)

Berbagai penemuan telah dilakukan, ternyata sifat dan kelakuan semut dan lebah dapat dijadikan algoritma yang sangat baik untuk digunakan dalam memecahkan berbagai masalah. Algoritma itu disebut algoritma semut dan algoritma lebah.

Kedua Algoritma itu sendiri merupakan penerapan dari kehidupan mereka sehari-hari dalam mencari makanannya. Algoritma semut sering diterapkan pada berbagai macam permasalahan seperti mencari jalur tercepat dalam graf,

Travelling Salesman Problem (TSP), dan lain-lain[1]. Algoritma lebah juga seringkali diterapkan pada berbagai macam permasalahan seperti penjadwalan kerja, pendistribusian data, bahkan prinsip internet dengan interkoloninya[3].

Semua itu sudah bisa diketahui jika memang kita memahami isi dari Al-Qur'an. Seperti firman Allah dalam surah Ali-Imran ayat 190. Allah berfirman :

“Sesungguhnya dalam penciptaan langit dan bumi, dan silih bergantinya malam dan siang terdapat tanda-tanda bagi orang-orang yang berakal,” (QS 3:190)

## II. KAJIAN PUSTAKA

### 2.1 Definisi Graf

Graf banyak digunakan untuk merepresentasikan objek-objek diskrit dan hubungan antara objek-objek tersebut. Secara matematis, Graf didefinisikan sebagai pasangan himpunan  $(V, E)$  yang ditulis dengan notasi :

$$G = (V, E)$$

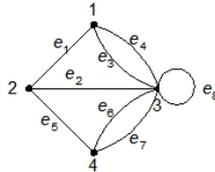
dengan  $V$  adalah himpunan tidak-kosong dari simpul-simpul (*vertices* atau *node*) dan  $E$  adalah himpunan boleh kosong dari sisi (*edges* atau *arcs*) yang menghubungkan sepasang simpul [4].

Definisi diatas menjelaskan bahwa graf setidaknya memiliki satu buah simpul tanpa sisi. Dari definisi itu juga dapat diketahui bahwa graf dengan banyak simpul tanpa sisi juga diperbolehkan. Graf yang hanya memiliki satu buah simpul tanpa sebuah sisi disebut *graf trivial*.

Simpul pada graf dapat dinyatakan dengan huruf besar ataupun kecil seperti A-Z, AA-ZZ, a-z atau bahkan gabungan keduanya. Simpul juga bisa direpresentasikan dengan bilangan asli 1, 2, 3, ... atau bahkan menggunakan gabungan dari huruf dan angka. Sedangkan sisi ( $e$ ) ditulis dengan notasi :

$$e = (u, v)$$

dengan  $u$  dan  $v$  merupakan simpul yang berarti sisi  $e$  menghubungkan simpul  $u$  dengan simpul  $v$ . Sisi dinyatakan dengan lambang  $e_1, e_2, e_3, \dots$



Gambar 1 [4]

Pada gambar 1 dapat diketahui bahwa graf tersebut memiliki :

$$V = \{1, 2, 3, 4\}$$

$$E = \{(1,2), (2,3), (1,3), (1,3), (2,4), (3,4), (3,4), (3,3)\}$$

Sisi  $e_3 = (1, 3)$  dan  $e_4 = (1, 3)$  disebut sisi ganda (*multiple edges* atau *parallel edges*) karena kedua sisi ini menghubungkan dua buah simpul yang sama yaitu simpul 1 dan simpul 3. Sisi  $e_8 = (3, 3)$  disebut gelang atau kalang (*loop*) karena sisi itu berawal dan berakhir pada simpul yang sama.

## 2.2 Jenis-Jenis Graf

Menurut buku *Matematika Diskrit* [4], Graf dapat dipisah menjadi beberapa kategori bergantung pada sudut pandang pengelompokannya. Graf dapat dikelompokkan berdasarkan ada tidaknya gelang atau sisi ganda, berdasarkan jumlah simpul, dan berdasarkan orientasi arah pada sisi.

Berdasarkan keberadaan gelang atau sisi ganda pada suatu graf, maka graf digolongkan menjadi dua jenis :

### 1. Graf sederhana (*simple graph*)

Graf sederhana adalah graf yang tidak mengandung gelang maupun sisi ganda. Pada graf sederhana, sisi adalah pasangan tak-terurut (*unordered pairs*) yang berarti sisi  $(u, v)$  sama saja dengan  $(v, u)$ .

### 2. Graf tak-sederhana (*unsimple graph*)

Graf tak-sederhana adalah graf yang mengandung gelang maupun sisi ganda. Graf tak-sederhana dibagi dua macam, yaitu graf ganda (*multigraph*) dan graf semu (*pseudograph*). Graf ganda adalah graf yang mengandung sisi ganda yaitu sisi yang menghubungkan sepasang simpul yang jumlahnya bisa dari dua buah. Sisi ganda dapat direpresentasikan sebagai pasangan yang boleh tak-terurut yang sama sehingga bisa ditulis  $(u, v)$  atau  $(v, u)$ . Graf semu adalah graf yang mengandung gelang (*loop*). Gambar 1 merupakan graf tak-sederhana.

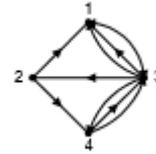
Graf dapat memiliki orientasi arah pada sisinya. Berdasarkan keberadaan orientasi arah pada suatu graf, maka graf digolongkan menjadi dua jenis :

### 1. Graf tak-berarah (*undirected graph*)

Graf yang sisinya tidak mempunyai orientasi arah disebut graf tak-berarah. Pada graf tak-berarah, urutan pasangan simpul yang dihubungkan oleh sisi tidak diperhatikan sehingga sisi  $(u, v)$  sama dengan sisi  $(v, u)$ . Gambar 1 merupakan graf tidak berarah.

### 2. Graf berarah (*directed graph* atau *digraph*)

Graf yang seluruh sisinya mempunyai orientasi arah disebut graf berarah. Sisi yang berarah biasa disebut busur (*arc*). Pada graf berarah,  $(u, v)$  dan  $(v, u)$  adalah dua busur yang berbeda sehingga  $(u, v) \neq (v, u)$ . Sisi  $(u, v)$  dapat diartikan dengan simpul  $u$  sebagai asal (*initial vertex*) dan simpul  $v$  sebagai simpul terminal (*terminal vertex*). Gelang diperbolehkan dalam graf berarah namun untuk sisi ganda tidak. Contoh graf berarah :



Gambar 2 [4]

Berdasarkan jumlah simpul pada suatu graf, maka graf digolongkan menjadi dua jenis :

### 1. Graf berhingga (*limited graph*)

Graf berhingga adalah graf yang jumlah simpulnya berhingga

### 2. Graf tak-berhingga (*unlimited graph*)

Graf tak-berhingga adalah graf yang jumlah simpulnya tidak berhingga.

Tabel berikut merupakan perluasan dari definisi graf.

Jenis	Sisi	Sisi ganda?	Sisi gelang?
Sederhana	Tak-berarah	Tidak	Tidak
Ganda	Tak-berarah	Ya	Tidak
Semu	Tak-berarah	Ya	Ya
Berarah	Berarah	Tidak	Ya
Ganda Berarah	Berarah	Ya	Ya

Tabel 1 Jenis-jenis graf [4]

## 2.3 Terminologi Graf

Beberapa istilah penting dalam graf yaitu

### 1. Ketetanggaan (*Adjacent*)

Dua simpul dikatakan bertetangga bila keduanya berhubungan secara langsung oleh sisi. Menurut gambar 1, simpul 1 bertetangga dengan simpul 2 dan 3 namun simpul 1 tidak bertetangga dengan simpul 4.

### 2. Bersisian (*Incidency*)

Suatu sisi dan simpul dikatakan bersisian apabila untuk sembarang sisi  $e = (v_j, v_k)$ ,  $e$  bersisian dengan simpul  $v_j$  dan  $e$  bersisian dengan simpul  $v_k$ . Jika ditinjau dari gambar 1, sisi  $(2, 3)$  bersisian dengan simpul 2 dan 3, sisi  $(2, 4)$  bersisian dengan simpul 2 dan simpul 4, namun sisi  $(1, 2)$  tidak bersisian dengan simpul 4.

### 3. Simpul Terpencil (*Isolated Vertex*)

Suatu simpul dikatakan terpencil apabila simpul itu tidak mempunyai sisi yang bersisian dengannya. Contoh

graf yang memiliki simpul terencil :

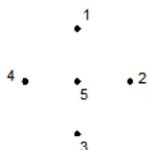


Gambar 3 [4]

pada gambar 3 dapat diketahui bahwa simpul 5 adalah simpul terencil.

4. Graf Kosong (*null graph* atau *empty graph*)

Graf kosong adalah graf dengan himpunan sisinya merupakan himpunan kosong. Contoh graf kosong :



Gambar 4 [4]

5. Derajat (*Degree*)

Derajat suatu simpul adalah jumlah sisi yang bersisian dengan simpul. Jumlah derajat akan berbeda bergantung pada orientasi arah pada suatu graf. Derajat dinotasikan dengan  $d(v)$ . Menurut gambar 1,  $d(1) = d(4) = 3$  dan menurut gambar 3,  $d(4) = 1$  dan  $d(5) = 0$ . Simpul yang berderajat 1 disebut simpul anting-anting.

Pada graf berarah,  $d_{in}(v)$  adalah jumlah busur yang masuk ke simpul  $v$  dan  $d_{out}(v)$  adalah jumlah busur yang keluar dari simpul  $v$ . Sehingga derajat dari simpul  $v$  :

$$d(v) = d_{in}(v) + d_{out}(v)$$

Banyak istilah penting yang lainnya namun tidak diperlukan pada makalah ini.

2.4 Perwarnaan Graf

Ada tiga jenis perwarnaan graf, yaitu perwarnaan simpul, perwarnaan sisi, dan perwarnaan wilayah (region). Pada makalah ini, perwarnaan lebih menekankan pada perwarnaan simpul.

Menurut buku *Matematika Diskrit*<sup>[4]</sup>, Perwarnaan simpul adalah memberi warna pada simpul-simpul suatu graf sedemikian sehingga tidak ada dua simpul bertetangga mempunyai warna yang sama.

Pemberian warna diharapkan menggunakan jumlah macam warna yang sesedikit mungkin. Jumlah warna minimum yang dapat digunakan untuk mewarnai simpul disebut bilangan kromatik graf  $G$  yang disimbolkan dengan  $\chi(G)$ . Suatu graf  $G$  mempunyai bilangan kromatis  $k$  dilambangkan dengan  $\chi(G) = k$ . [4]

2.4.1. Algoritma Welch-Powell

Salah satu algoritma untuk mewarnai sebuah graf

adalah Algoritma Welch-Powell. Algoritma ini tidak memberikan jumlah warna minimum yang diperlukan untuk mewarnai  $G$ . Langkah-langkah dalam Algoritma Welch-Powell [5] :

1. Urutkan simpul-simpul dari  $G$  dalam urutan derajat yang menurun. Urutan ini mungkin tidak unik karena beberapa simpul mungkin mempunyai derajat yang sama.
2. Gunakan satu warna tertentu untuk mewarnai simpul pertama. Secara berurutan, setiap simpul dalam daftar yang tidak berelasi dengan simpul sebelumnya diwarnai dengan warna ini.
3. Ulangi langkah dua di atas untuk simpul dengan urutan tertinggi yang belum diwarnai.
4. Ulangi langkah 3 di atas sampai semua simpul dalam daftar terwarnai.

2.4.2. Fungsi Perwarnaan

Menurut Majid Faraji dalam papernya [6], jika  $d(V_a) = m$  dan simpul tetangga dari  $V_a$  adalah  $\{V_1, V_2, \dots, V_m\}$  dan himpunan dari warna simpul tetangga adalah  $C = \{c(V_1), c(V_2), \dots, c(V_m)\}$  maka :

```

Function Coloring Vertex ( $V_a$ )
for  $j = 1$  to  $N$ 
if  $j \notin C$  then
Begin
 $c(V_a) = j$ ;
exit;
end
    
```

Dengan  $c(V_a)$  adalah warna dari  $V_a$

III. ALGORITMA LEBAH

A. Kelakuan Lebah di Alam

Algoritma ini didasari dengan kelakuan atau tingkah laku dari lebah madu dalam mencari makanannya. Saat proses pencarian makanan, lebah madu menyebar secara *sporadis* dalam bentuk tim. Ketika kembali menuju sarangnya, lebah yang menemukan bunga yang layak mendeposit lalu melakukan *waggle dance* atau *round dance*.

*Waggle dance* dan *Round dance* [2] adalah tarian lebah yang sangat penting dalam penyebaran informasi pada koloni lebah. Tarian itu memberikan informasi berupa arah dan jarak menuju lokasi bunga, dan kualitas dari bunga-bunga yang ada di lokasi itu. Informasi ini membantu koloni untuk mengirim lebah-lebah lain untuk menuju bunga yang dimaksud tanpa peta. Tarian itu membuat koloni bisa membandingkan mana yang baik untuk dipanen menurut kualitas dan jumlah energi yang dibutuhkan untuk mengumpulkan *nectar*.

Setelah lebah yang melakukan tarian selesai menari, lebah itu ikut dengan lebah yang lain untuk pergi ke lokasi bunga yang dimaksud. Semakin banyak *nectar* yang bisa dikumpulkan pada suatu daerah, maka semakin banyak juga lebah yang dikirim ke sana. Hal ini lah yang

membuat koloni lebah bisa mengumpulkan makanan secara cepat dan mangkus.

Saat masih memanen, lebah juga memantau kualitas dari makanan yang dipanennya. Jika masih memungkinkan dan baik untuk dipanen, maka lebah akan melakukan *waggle dance* kembali untuk menarik lebah lain dan memanen kembali di lokasi yang sama. [2][6]

### B. Algoritma Koloni Lebah

Algoritma lebah ini membutuhkan sejumlah parameter, yaitu jumlah dari lebah pencari ( $n$ ), jumlah tempat yang dikunjungi dari  $n$  tempat ( $m$ ), jumlah tempat terbaik dari  $m$  ( $e$ ), jumlah lebah yang direkrut untuk lokasi  $e$  ( $nep$ ), jumlah dari lebah yang direkrut untuk lokasi lain ( $nsp$ ), ukuran dari lokasi ( $ngh$ ) termasuk situs dan sekelilingnya dan kriteria penghentinya.

Algoritma sederhananya adalah

*Sending scout bees for making primary solutions.*

*Evaluate fitness of the solutions.*

*If solution is better than the other*

*Scout bees become Employed Bees ( $e$ )*

*While (stopping criterion not met)*

*Select sites for neighborhood search.*

*Recruit bees for selected sites (more bees for best  $e$  sites) and evaluate fitnesses.*

*Select the fittest bee from each patch.*

*Assign remaining bees to search randomly and evaluate their fitnesses.*

*End While.*

Ada dua jenis lebah yang digunakan yaitu lebah pekerja (*employed bees*) dan lebah pencari (*scout bees*). Lebah pekerja (*employed bees*) adalah lebah yang memberikan informasi terhadap lokasi bunga yang baik untuk dipanen dengan cara menari. Lebah pencari (*scout bees*) adalah lebah yang mencari lokasi baru secara random.

Awalnya, sejumlah lebah pencari mencari lokasi yang lalu hasil temuan dari masing-masing lebah dievaluasi sehingga didapat beberapa tempat yang layak dan lebih baik dari yang lain. Lebah yang menentukan tempat itu ( $e$ ) menjadi lebah pekerja. Lebah pekerja memulai menyebarkan informasi lokasi bunga kepada lebah lain.

Langkah-langkah diatas dapat digambarkan dengan *pseudo-code* yang di atas, namun lebih sederhana. Langkah yang ditawarkan belum sampai pada memilih lokasi dan jumlah lebah yang diperlukan. Langkah yang ditawarkan masih kepada informasi yang disebarkan melalui lebah pekerja (*employed bees*) dan pembaharuan informasi sehingga *pseudo-code* yang digunakan adalah

*Sending scout bees for making solutions.*

*Evaluate fitness of scout bees' solution*

*If solution is better than the other*

*Scout bees become Employed Bees ( $e$ )*

*While (stopping criterion not met)*

*Waggle dancing of employed bees and sharing*

*information and memorizing information of all bees.*

*Neighborhood Search.*

*Updating Information*

*End While.*

## IV. PENERAPAN ALGORITMA LEBAH DALAM PEWARNAAN GRAF

Dari *Pseudo-code* sederhana pada bagian B, kita dapat membuat dan mewarnai graf sesuai dengan langkah-langkah berikut [6]:

### 1. Pembuatan solusi awal

Pada kehidupan lebah, lebah pencari merupakan lebah yang berkeliling secara random mencari lokasi bunga yang layak untuk dipanen. Lebah pencari dapat diibaratkan menjadi simpul pada graf. Langkahnya adalah lebah pencari pertama dijadikan simpul pada graf secara random. Kemudian, pemilihan simpul selanjutnya menggunakan fungsi probabilitas [6]:

$$P_{V_i S} = \frac{d(V_i)}{\sum_{i=1, V_i \notin W_s}^N d(V_i)} \text{ untuk } V_i \notin W_s$$

$$= 0 \text{ untuk } V_i \in W_s$$

dengan  $N$  adalah jumlah simpul pada graf,  $W_s$  adalah himpunan dari simpul berwarna oleh lebah pencari  $S$ . Setiap simpul diwarnai dengan fungsi pewarnaan dan prosedur ini diulang hingga semua simpul telah berwarna.

### 2. Evaluasi dari solusi yang dibawa lebah $S$

Lebah pencari kembali ke sarangnya membawa sampel dari *nectar* daerah yang dikunjunginya. Daerah yang sampel *nectar*-nya diambil disebut solusi. Kualitas dari suatu solusi bisa dihitung dengan rumus [6] :

$$fit(S) = \frac{1}{k(S)}$$

dengan  $k(S)$  adalah jumlah warna yang digunakan untuk mewarnai lebah  $S$ . Sejumlah lebah pencari yang memberikan solusi lebih baik diantara yang lain dipilih dan diubah menjadi lebah pekerja.

### 3. Persebaran informasi menggunakan *waggle dance*

*Waggle dance* adalah tarian yang dilakukan lebah untuk menyebarkan informasi. Tahap ini mengadaptasi dari proses penyebaran informasi itu. Informasi yang disebar diasumsikan sama rata pada setiap lebah sehingga dapat disimpulkan bahwa informasi itu dianggap membutuhkan satu memori untuk seluruh koloni lebah. Memori itu terdiri dari  $N$  ruangan dan setiap ruangan menyimpan simpul sebagai informasi yang isinya dapat dihitung menggunakan rumus [6] :

$$\tau_{V_i} = \sum_{s=1}^e \frac{0,5 j_{V_i S}}{c_s(V_i)}$$

dengan  $j_{V_i S}$  merupakan simpul  $V_i$  adalah simpul ke  $j$  yang diwarnai yang merepresentasikan solusi dari lebah  $S$  dan

$c_s(V_i)$  adalah warna dari simpul  $V_i$ .

#### 4. Pencarian Daerah lain sekitar Solusi

Sembari mengundang berbagai lebah dan menuju lokasi awal untuk mengumpulkan nektar kembali, lebah juga mencari daerah lain yang memungkinkan untuk dipanen. Ada beberapa definisi yang digunakan untuk pengerjaan tahap empat ini yaitu :

##### Definisi 1 [6]

Jika  $d(V_a) = m$  dan simpul  $\{V_1, V_2, \dots, V_m\}$  bertetanggan dengan  $V_a$  maka :

$$Hd(V_a) = d(V_1) + d(V_2) + \dots + d(V_m)$$

##### Definisi 2 [6]

Jika himpunan simpul yang berupa solusi yang diberikan dari lebah S yang sama memiliki warna misal M adalah  $\{V_1, V_2, \dots, V_{N_M}\}$  maka :

$$Q_{S_M} = \sum_{i=1}^{N_M} Hd(V_i)$$

dengan  $N_M$  adalah jumlah simpul yang diwarnai oleh warna M.

##### Definisi 3 [6]

Jika himpunan simpul yang berupa solusi yang diberikan dari lebah S yang sama memiliki warna misal M adalah  $\{V_1, V_2, \dots, V_{N_M}\}$  maka :

$$\lambda_{S_M} = \sum_{i=1}^{N_M} \tau_{V_i}$$

dengan  $N_M$  adalah jumlah simpul yang diwarnai oleh warna M.

##### Definisi 4 [6]

Jika  $N$  adalah jumlah simpul pada graf dan  $k(S)$  adalah bilangan kromatik maka :

$$F_S = \frac{\binom{N}{k(S)}}{3}$$

##### Definisi 5 [6]

$$X_{S_i} = Q_{S_i} - Q_{S_{i+1}} \quad \text{untuk } 0 < i < k(S)$$

$$X_{S_i} = Q_{S_i} - Q_{S_i} \quad \text{untuk } i = k(S)$$

Pada langkah ini, lebah pekerja mencari daerah sekitar lokasi awal tadi. Untuk setiap warna simpul  $X_{S_i} (i \in \{1, 2, \dots, k\})$  jika  $d = \max(X_{S_i})$  maka warna  $i$  akan dipilih.

Fase 1 : jika  $i > 1, D_{i-1} > F_S, i-1 \notin b_s$  maka semua simpul yang memiliki warna  $i-1$  pada  $S$

diwarnai oleh fungsi pewarnaan simpul dan  $i-1$  ditambah ke himpunan  $b_s$ . Himpunan  $b_s$  menentukan warna mana yang telah digunakan oleh lebah S.

Fase 2 : jika  $i < k(S), D_{i+1} > F_S, i+1 \notin b_s$  maka semua simpul yang memiliki warna  $i+1$  pada  $S$  diwarnai oleh fungsi pewarnaan simpul dan  $i+1$  ditambah ke himpunan  $b_s$ .

Fase 3 : jika  $i \notin b_s$  maka semua simpul yang memiliki warna  $i$  pada  $S$  diwarnai oleh fungsi pewarnaan simpul dan  $i$  ditambah ke himpunan  $b_s$ .

Fase 4 : warna selanjutnya ditentukan oleh fungsi probabilitas :

$$P_{S_i} = \frac{(Q_{S_i})^\alpha (\lambda_{S_i})^\beta}{\sum_{j=1, j \notin b_s}^{k(S)} (Q_{S_j})^\alpha (\lambda_{S_j})^\beta} \quad \text{untuk } i \notin b_s$$

$$= 0 \quad \text{untuk } i \in b_s$$

$P_{S_i}$  adalah probabilitas pemilihan warna  $i$  pada  $S$

$k(S)$  adalah angka kromatik

$b_s$  adalah warna yang sudah digunakan dalam pencarian oleh  $S$

$\alpha$  dan  $\beta$  adalah integer positif

$Q$  diambil dari Definisi 2

$\lambda$  diambil dari Definisi 3

Fase 1 hingga fase 4 diulang terus hingga seluruh simpul pada graf warna telah dibuat.

#### 5. Pengecekan ulang Kualitas dan Pembaharuan Informasi

Seperti yang sudah dijelaskan, setelah lebah menemukan daerah baru yang berpotensi untuk dipanen, lebah juga membandingkan apakah daerah itu lebih baik dari sebelumnya apa belum. Langkah ini memperbarui informasi dengan cara membandingkan kualitas dari solusi yang diberikan di awal dengan solusi pada tahap keempat. Jika solusi tahap keempat lebih baik atau sama dengan solusi awal, maka solusi awal dihapus dan solusi pada keempatlah yang menjadi solusi terpilih.

## V. KESIMPULAN

Pewarnaan graf dapat menggunakan berbagai cara, salah satunya adalah menggunakan algoritma koloni lebah. Secara konsep algoritma, metode konvensional dalam pewarnaan graf lebih mudah dipahami. Namun, algoritma koloni lebah ini tepat digunakan untuk menyelesaikan permasalahan pewarnaan graf secara cepat.

## VI. SANWACANA

Penulis berterima kasih kepada Allah SWT karena atas

berkah rahmahnya penulis dapat menyelesaikan paper ini. Penulis juga berterimakasih kepada orang tua dan teman yang membantu dalam pengerjaan. Tak lupa penulis juga berterimakasih kepada Bu Harlili dan Pak Rinaldi yang telah memberikan contoh dan pengajarannya.

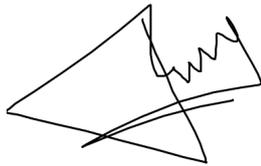
#### REFERENCES

- [1] Ramadhan, Edvin. "Sekilas tentang Algoritma Semut" <http://edvinramadhan.blogspot.com/2010/10/sekilas-tentang-algoritma-semut-antnet.html>. Tanggal akses : 13 Desember 2013 pukul 21.30.
- [2] <http://chern.ie.nthu.edu.tw/gen/10.pdf>. Tanggal akses : 16 Desember 2013
- [3] "Internet ke Interkoloni" <http://admin.harunyahya.com/indo/artikel/079.htm>. Tanggal akses : 14 Desember 2013
- [4] Munir, Rinaldi. Matematika Diskrit (Revisi Keempat). Bandung, Informatika Bandung, 2010, bab. 8.
- [5] Baizal, Abdurrahman ZK. Matematika Diskrit. Bandung, 2011. bab 6
- [6] Faraji, Majid dan Haj Seyyed Javadi. "Proposing a New Algorithm Based on Bees Behavior for Solving Graph Coloring".2011. Vol.6. "submitted for publication"

#### 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, 16 Desember 2013



Riva Syafri Rachmatullah  
13512036