

Penerapan Algoritma *Tabu Search* pada Pewarnaan Graf beserta Aplikasinya

Muhammad Alif Arifin 13516078¹

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

¹13516078@std.stei.itb.ac.id

Abstract—Pewarnaan graf sangat banyak manfaatnya pada dunia sains oleh karena itu banyak algoritma-algoritma yang dikembangkan untuk menyelesaikan permasalahan ini. Salah satu algoritma yang dikembangkan adalah *Tabu Search*. *Tabu Search* merupakan suatu metode optimasi yang berbasis pada pencarian solusi tetangga dan memori lokal (*local search*). Pewarnaan graf dapat dimanfaatkan untuk menentukan jadwal perkuliahan. Pada makalah ini akan dibahas mengenai penerapan algoritma *Tabu Search* pada pewarnaan graf beserta aplikasinya pada menentukan jadwal perkuliahan.

Keywords—Algoritma, graf, penjadwalan, pewarnaan graf, *Tabu Search*

I. INTRODUCTION

Dalam kehidupan sehari-hari banyak permasalahan-permasalahan yang dapat diselesaikan dengan pewarnaan graf. Pewarnaan simpul pada graf dapat dilakukan secara manual dengan algoritma *Welch-Powell*. Namun, dengan jumlah simpul yang mencapai ratusan atau bahkan ribuan tentunya pewarnaan graf secara manual akan sangat melelahkan dan memperbesar resiko kesalahannya. Oleh karena itu, banyak algoritma-algoritma yang telah dikembangkan untuk melakukan pewarnaan simpul pada graf berskala besar. Salah satu contohnya adalah algoritma *Tabu Search*.

Algoritma *Tabu Search* didasari pada metode optimasi yang berbasis pada pencarian solusi tetangga dan memori lokal. Kemampuan *Tabu Search* dalam menghasilkan solusi yang mendekati optimal telah dimanfaatkan dalam berbagai macam permasalahan klasik dan praktis di berbagai macam bidang yang salah satunya adalah pewarnaan graf.

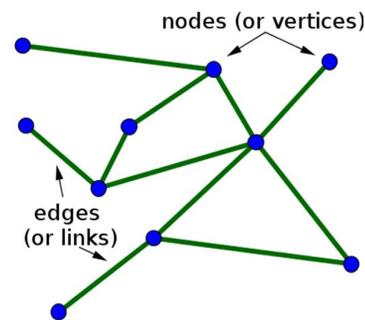
Pewarnaan graf dapat menyelesaikan permasalahan-pemmasalahan lain yang dirasa cukup penting dalam kehidupan sehari-hari. Permasalahan yang ditopikkan pada makalah ini adalah permasalahan penjadwalan kuliah. Dengan tujuan agar tidak ada kuliah yang memiliki jadwal yang bertabrakan dengan yang lain dan tidak ada dosen yang mengajar lebih dari 1 mata kuliah pada waktu yang bersamaan.

II. GRAF

A. Definisi Graf

Secara matematis graf G merupakan pasangan himpunan dari (V, E) , yang dalam hal ini V merupakan himpunan tidak-kosong

dari *vertices* atau simpul dan E merupakan himpunan dari *edges* atau sisi yang menghubungkan sepasang simpul yang ada pada graf tersebut. Graf memiliki penulisan notasi $G = (V, E)$ [2].



Gambar 1. Simpul dan Sisi pada Graf

Sumber : <http://mathinsight.org>

Simpul yang ada pada suatu graf dapat diberikan penomoran baik dengan huruf, seperti a, b, c, \dots, x, y atau dengan bilangan asli $1, 2, 3, \dots$ atau gabungan dari keduanya. Sedangkan sisi yang menghubungkan dengan simpul a dan b dapat dinyatakan dengan simbol e_1, e_2, \dots atau dengan pasangan (a, b) . Dengan kata lain, jika e merupakan sisi yang menghubungkan simpul a dengan b , maka e dapat ditulis dengan $e = (a, b)$

Berdasarkan pernyataan di atas, V tidak boleh kosong, sedangkan E boleh kosong. Hal tersebut memungkinkan suatu graf hanya memiliki simpul-simpul tanpa adanya sisi. Graf tersebut dinamakan dengan graf trivial.

B. Jenis Graf

Graf dapat dikelompokkan menjadi beberapa katagori berdasarkan sudut pandanganya. Pengelompokkan tersebut dapat didasari dengan ada tidaknya sisi ganda atau sisi kalang, jumlah simpul atau orientasi arah pada sisi.

Graf dapat dikelompokkan menjadi 2 jenis berdasarkan ada tidaknya gelang atau sisi ganda, yaitu:

1. Graf sederhana (*simple graph*), yaitu graf yang tidak memiliki sisi gelang maupun sisi ganda
2. Graf tak-sederhana (*unsimple graph*), yaitu graf yang memiliki sisi ganda atau gelang.

Graf juga dapat dikelompokkan berdasarkan jumlah simpul yang ada, yaitu:

1. Graf berhingga (*limited graph*), yaitu graf yang jumlah simpulnya sebanyak n .
2. Graf tak-berhingga (*unlimited graph*), yaitu graf yang jumlah simpulnya tidak berhingga banyaknya.

Dan yang terakhir, graf dapat dikelompokkan berdasarkan ada tidaknya orientasi arah, yaitu:

1. Graf berarah (*directed graph*), yaitu graf yang setiap sisinya memiliki orientasi arah.
2. Graf tak-berarah (*undirected graph*), yaitu graf yang sisinya tidak memiliki orientasi arah.

C. Terminologi Graf

Terdapat terminologi-terminologi pada graf yang perlu diketahui, yaitu [2]:

1. Ketetanggaan (*Adjacency*)
Dua buah simpul dapat dikatakan bertetangga jika keduanya terhubung oleh sebuah sisi yang sama.

2. Bersisian (*Incidency*)
Untuk seluruh sisi $e = (u, v)$, sisi e dapat dikatakan bersisian dengan simpul u dan v .

3. Simpul Terpencil (*Isolated Vertex*)
Simpul dapat dikatakan simpul terpencil jika simpul tersebut tidak memiliki sisi yang bersisian dengannya. Atau, dapat dinyatakan bahwa simpul yang tidak memiliki tetangga.

4. Graf Kosong (*Null Graph*)
Graf kosong merupakan graf yang himpunan sisinya adalah himpunan kosong. Penulisan graf kosong dapat berupa N_n dengan n merupakan jumlah simpul.

5. Derajat (*Degree*)
Derajat suatu simpul pada graf tak-berarah adalah jumlah sisi yang bersisian dengan simpul tersebut. Dengan notasi $d(v)$ yang menyatakan derajat dari simpul v . Pada graf berarah derajat dibedakan menjadi dua yaitu derajat-masuk dan derajat-keluar. Derajat-masuk adalah jumlah sisi yang masuk pada simpul tersebut. Sedangkan derajat-keluar adalah jumlah sisi yang keluar dari simpul tersebut.

6. Lintasan (*Path*)
Lintasan dengan panjang n dari simpul awal v_0 dan simpul tujuan v_n di dalam graf G adalah barisan berselang-seling simpul-simpul dan sisi-sisi yang berbentuk $v_0, e_1, v_1, e_2, \dots, v_n$ sedemikian sehingga $e_1 = (v_0, v_1)$ dan seterusnya. Dengan e merupakan sisi-sisi dari graf G .

7. Siklus (*Cycle*) atau Sirkuit (*Circuit*)
Sirkuit atau siklus adalah lintasan yang berawal dan berakhir pada simpul yang sama.

8. Terhubung (*Connected*)
Suatu graf tak-berarah dapat dikatakan graf terhubung jika untuk setiap pasang simpul u dan v yang ada pada himpunan V terdapat lintasan dari u ke v (yang seharusnya juga terdapat lintasan dari v ke u). Jika terdapat satu saja pasangan yang tidak memiliki lintasan, maka graf tersebut dikatakan graf tak-terhubung.

9. Upagraf (*Subgraph*) dan Komplemen Upagraf

Misalkan $G = (V, E)$, dengan G adalah sebuah graf. $G_1 = (V_1, E_1)$ adalah upagraf dari G jika $V_1 \subseteq V$ dan $E_1 \subseteq E$. Komplemen dari upagraf G_1 terhadap G adalah graf $G_2 = (V_2, E_2)$ sedemikian sehingga $E_2 = E - E_1$ dan V_2 adalah himpunan simpul yang bersisian dengannya. Komponen graf adalah jumlah maksimum upagraf terhubung dalam graf G .

10. Upagraf Rentang (*Spanning Subgraph*)
Upagraf $G_1 = (V_1, E_1)$ dari $G = (V, E)$ dikatakan upagraf rentang jika $V_1 = V$. Yang artinya adalah G_1 memiliki seluruh simpul yang ada pada G .

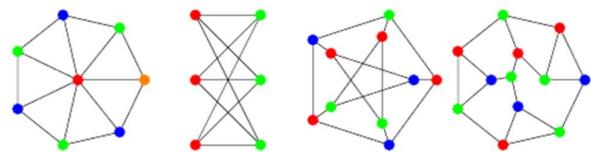
11. *Cut-set*
Cut-set dari graf terhubung G adalah himpunan sisi yang bila dihapus/dibuang dari G akan menyebabkan graf G tidak terhubung kembali.

12. Graf Berbobot (*Weighted Graph*)
Graf berbobot adalah graf yang setiap sisinya memiliki sebuah bobot.

D. Pewarnaan Graf

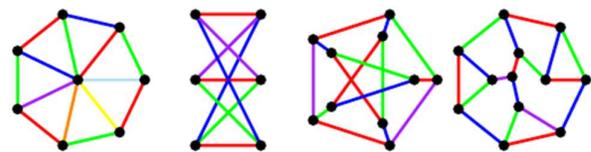
Terdapat tiga macam persoalan dalam pewarnaan graf (*graph colouring*), yaitu pewarnaan simpul, pewarnaan sisi, dan pewarnaan wilayah. Namun terdapat persoalan yang penting juga karena dalam pewarnaan graf, tidak hanya sekadar mewarnai simpul-simpul dengan warna yang berbeda dari simpul tetangganya namun juga harus menggunakan warna seminimal mungkin. Jumlah warna tersebut disimbolkan dengan $\chi(G)$ atau bilangan kromatik dari graf G . Tiga macam pewarnaan graf, yaitu:

1. Pewarnaan Simpul
Pewarnaan simpul adalah memberikan warna pada setiap simpul yang ada dan tidak boleh ada simpul bertetangga yang memiliki warna yang sama [3].



Gambar 2. Pewarnaan Simpul pada Graf
(sumber : <http://mathworld.wolfram.com>)

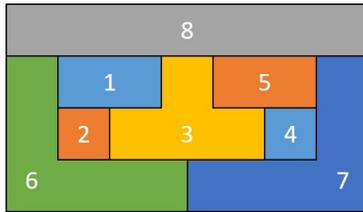
2. Pewarnaan Sisi
Pewarnaan sisi pada graf G adalah pewarnaan pada sisi dari G yang memberikan warna pada setiap sisi dan tidak boleh ada sisi yang bersisian dengan simpul yang sama memiliki warna yang sama [4].



Gambar 3. Pewarnaan Sisi pada Graf
(sumber : <http://mathworld.wolfram.com>)

3. Pewarnaan Bidang

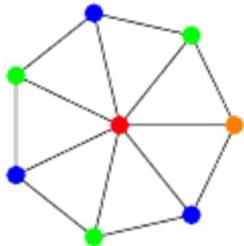
Pewarnaan bidang adalah memberikan warna pada bidang dan tidak boleh ada bidang yang bertetangga dengan warna yang sama.



Gambar 4. Pewarnaan Bidang pada Graf
(sumber : <http://www.laurentlessard.com>)

E. Bilangan Kromatik

Bilangan kromatik adalah jumlah warna minimum yang dapat digunakan untuk mewarnai simpul. Bilangan kromatik pada graf G , disimbolkan dengan $\chi(G)$. Suatu graf G yang mempunyai bilangan kromatis k dapat dilambangkan dengan $\chi(G) = k$. Contohnya pada gambar 5. graf tersebut memiliki bilangan kromatik sebesar 4 atau $\chi(G) = 4$ [2].



Gambar 5. Contoh graf dengan $\chi = 4$
(sumber : <http://mathworld.wolfram.com>)

Terdapat graf yang memiliki sifat tertentu yang dapat dikenali secara langsung bilangan kromatiknya. Graf tersebut sebagai berikut:

1. Graf kosong N_n memiliki $\chi(G) = 1$.
2. Graf lengkap K_n memiliki $\chi(G) = n$.
3. Graf teratur derajat n memiliki $\chi(G) = n + 1$.
4. Graf bipartit $K_{n,m}$ memiliki $\chi(G) = 2$.
5. Graf lingkaran ganjil memiliki $\chi(G) = 3$.
6. Graf lingkaran genap memiliki $\chi(G) = 2$.
7. Sembarang pohon T memiliki $\chi(T) = 2$.

III. ALGORITMA *TABU SEARCH*

A. Sejarah

Algoritma *Tabu Search* diperkenalkan oleh Fred Glover pada tahun 1986. Pada tahun 1988, *Committee on the NeEt Decade of Operations Research* (CONDOR) menetapkan *Tabu Search*, bersama dengan *Simulated Annealing* dan *Genetic Algorithm*, sebagai metode yang sangat menjanjikan untuk aplikasi praktis.

B. Definisi

Salah satu algoritma untuk menyelesaikan masalah pewarnaan graf adalah algoritma *Tabu Search* [7]. Algoritma *Tabu Search* merupakan suatu metode optimasi yang berbasis pada pencarian solusi tetangga dan memori lokal (*local search*). Proses pencarian adalah dengan bergerak dari satu solusi ke

solusi berikutnya dan berusaha untuk mencari solusi tetangga yang lebih baik dari solusi yang ada saat ini. Selain itu, memori lokal digunakan untuk mencatat langkah-langkah pencarian yang pernah ditemui. Jika langkah pencarian tersebut pernah ditemui tidak baik (*tabu*), algoritma *Tabu Search* akan mengabaikan langkah tersebut namun langkah itu akan digunakan untuk menuntun dalam pencarian selanjutnya [8].

C. Kelebihan *Tabu Search*

Tabu Search memiliki kelebihan yang terletak pada struktur memori yang fleksibel. Pencarian yang dilakukan secara terus menerus meskipun solusi yang terbaik sudah diperoleh dimungkinkan dengan adanya struktur memori tersebut. Selain itu, struktur memori akan menjaga agar tidak ada proses pencarian yang pernah muncul pada pencarian sebelumnya [9][10]. Hal tersebut menyebabkan waktu komputasi yang semakin singkat dengan jumlah data yang banyak.

Tabu Search memiliki karakteristik unik yang menyebabkan algoritma ini unggul daripada algoritma lain. Pada umumnya tidak menggunakan pembentukan kandidat solusi yang dibangkitkan secara acak sebagaimana *genetic algorithm* dan *simulated annealing*. Pemilihan kandidat solusi dalam *Tabu Search* juga tidak dilakukan secara probabilistik (peluang) sebagaimana *simulated annealing*, *genetic algorithm* dan *and colony system*. Dengan karakteristik yang telah disebutkan, hal tersebut menjadikan solusi-solusi yang ditawarkan oleh *Tabu Search* akan selalu sama pada proses pencarian solusi suatu permasalahan.

D. Skema Tabu Search

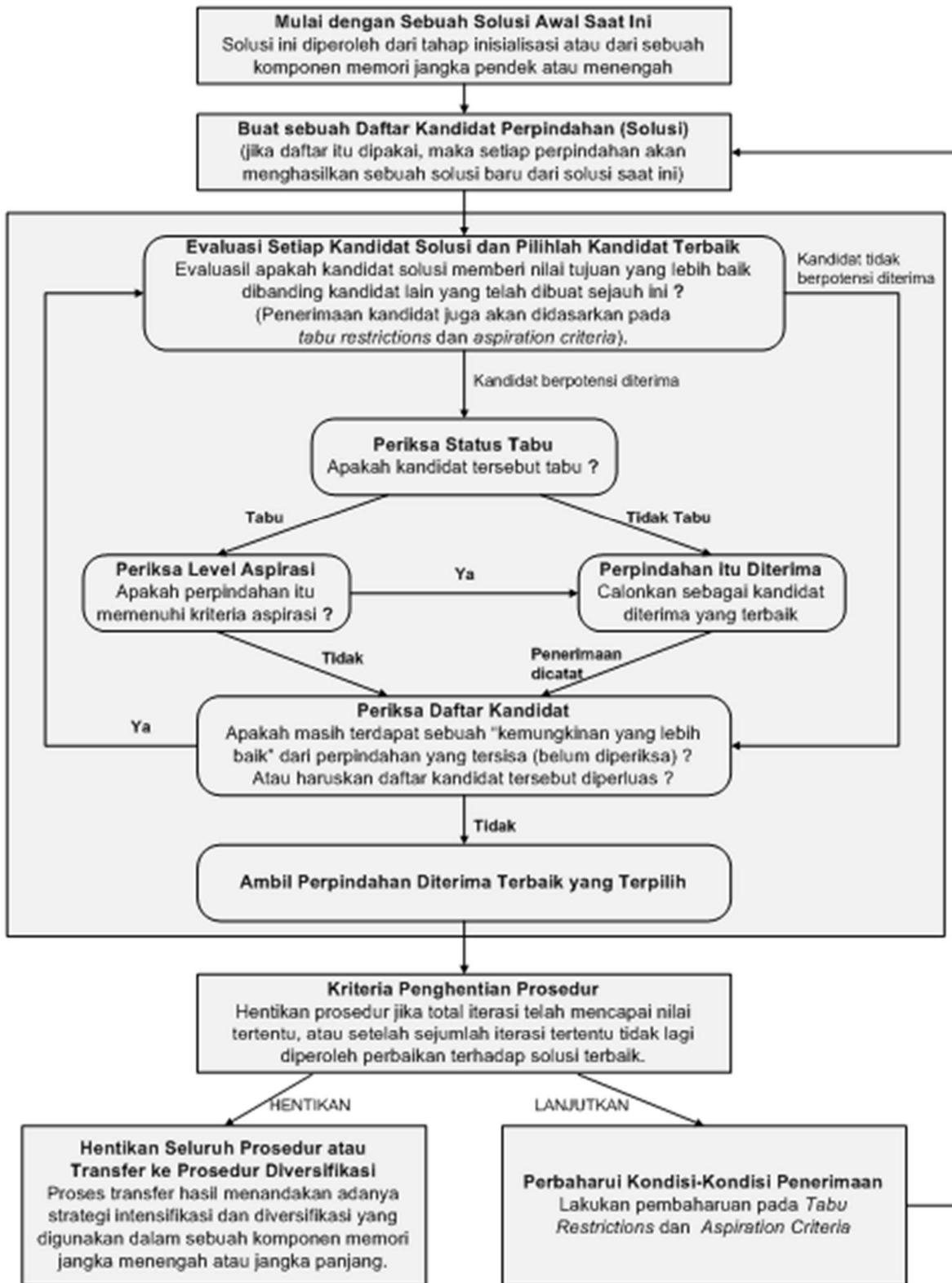


Diagram 1. Skema Algoritma Tabu Search
(sumber : <http://priyandari.staff.uns.ac.id>)

E. Langkah-langkah

Langkah-langkah algoritma *Tabu Search* untuk menyelesaikan persoalan pewarnaan graf adalah sebagai berikut [7]:

1. Mewarnai titik secara acak.
2. Menentukan apakah solusi awal memenuhi kriteria solusi yang diharapkan
3. Jika terjadi konflik (ada titik yang bertetangga dengan warna yang sama), membangkitkan solusi baru dari solusi yang didapat dengan melakukan *move* (penggantian warna).
4. Menyimpan solusi yang tidak tabu dalam *tabu list* dan mengabaikan solusi yang tabu.
5. Memilih solusi optimal dari *tabu list*.
6. Menerapkan solusi optimal pada graf
7. Jika masih terdapat konflik, proses akan kembali ke langkah 3. Jika tidak, proses selesai.

Terdapat hal yang perlu diperhatikan sebelum menggunakan algoritma *tabu search*, diperlukan untuk mengonversi data yang diinginkan menjadi sebuah graf. Selain itu, diperlukan informasi mengenai jumlah bilangan kromatiknya.

F. Flowchart *Tabu Search* pada Pewarnaan Graf

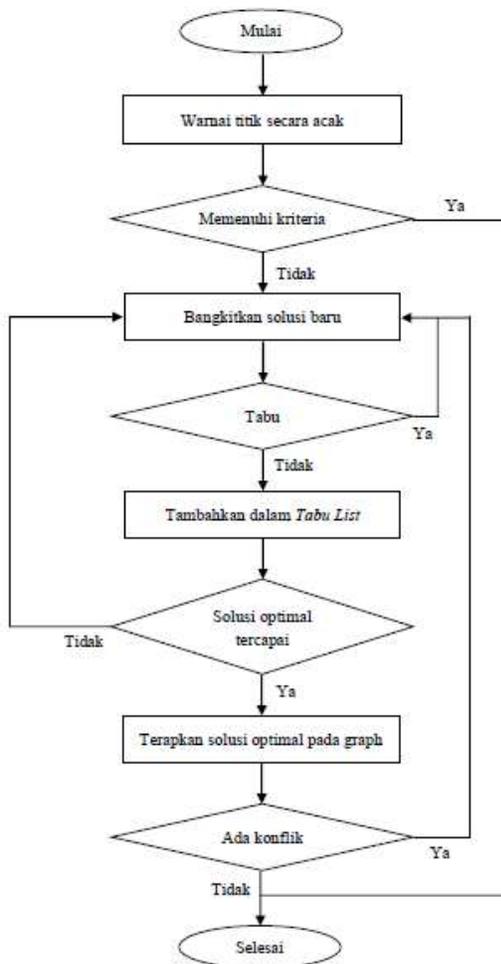


Diagram 2. Flowchart *Tabu Search* pada Pewarnaan Graf
(sumber : <http://jurnal-online.um.ac.id>)

IV. PERCOBAAN ALGORITMA TABU UNTUK PENJADWALAN PERKULIAHAN

A. Data

Akan dilakukan uji coba pada suatu data untuk menyelesaikan masalah jadwal perkuliahan

Tabel 1. Data Daftar Kuliah

No.	Mata Kuliah	Dosen	Kelas
1.	MK1	D1	K1
2.	MK2	D3	K2
3.	MK3	D2	K1
4.	MK4	D1	K1
5.	MK5	D4	K2
6.	MK6	D2	K2
7.	MK7	D4	K1

B. Pengubahan Data menjadi Graf

Berdasarkan data yang didapat dengan simpul yang menyatakan mata kuliah dan sisi yang menyatakan hubungan kedua simpul. Simpul akan bertetangga jika memiliki dosen yang sama dan/atau memiliki kelas yang sama.

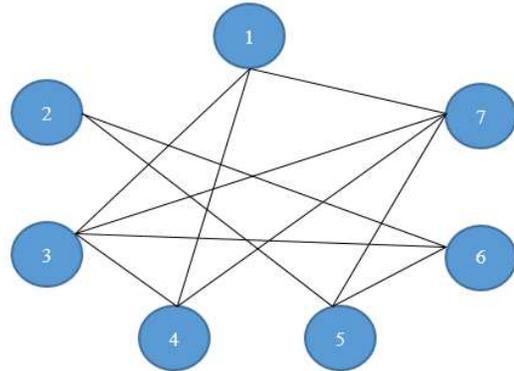


Diagram 3. Graf Daftar Kuliah
(sumber : dokumentasi penulis)

C. Penentuan Jumlah Warna

Jumlah warna yang digunakan pada percobaan kali ini yang direpresentasikan dengan k , $k = 4$. Setiap warna akan merepresentasikan hari. Merah untuk Hari Senin, hijau untuk Hari Selasa, kuning untuk Hari Rabu dan biru untuk Hari Kamis. Perlu diingat bahwa warna yang digunakan tidak diwajibkan merupakan bilangan kromatik. Namun perlu dipastikan bahwa jumlah warna tersebut harus lebih besar sama dengan bilangan kromatik, $\chi(G) \leq k$.

D. Pewarnaan Graf

Dengan menggunakan langkah-langkah yang telah disebutkan sebelumnya, yaitu:

1. Mewarnai titik secara acak

Pewarnaan titik dilakukan secara bebas dengan syarat seluruh simpul memiliki warna masing-masing.

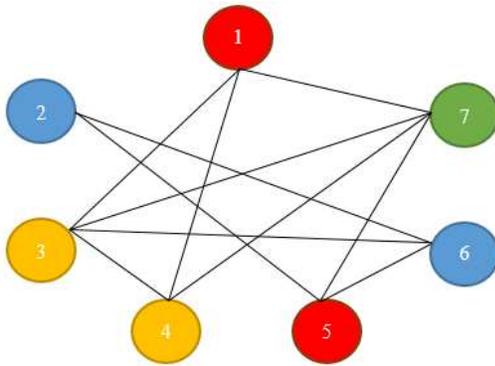


Diagram 4. Pewarnaan Simpul Secara Acak
(sumber : dokumentasi penulis)

- Menentukan apakah solusi awal memenuhi kriteria solusi yang diharapkan

Solusi awal dengan melakukan pengacakan pada warna tidak memenuhi kriteria yang diharapkan. Dalam hal ini adalah terdapat dua simpul bertetangga yang memiliki warna yang sama, yaitu simpul 2 dan 6 yang sama-sama memiliki warna biru dan simpul 3 dan 4 yang sama-sama memiliki warna kuning.

- Jika terjadi konflik (ada titik yang bertetangga dengan warna yang sama), membangkitkan solusi baru dari solusi yang didapat dengan melakukan *move* (penggantian warna).

Terdapat konflik berdasarkan langkah 2. Dikarenakan hal tersebut, maka diperlukan tindakan *move* (penggantian warna) pada simpul-simpul yang memiliki konflik. Solusi yang dimiliki adalah warna merah, biru, hijau dan kuning.

- Menyimpan solusi yang tidak tabu dalam *tabu list* dan mengabaikan solusi yang tabu.

Solusi yang tidak tabu untuk konflik pada simpul 2 dan 6 adalah warna kuning dan hijau. Dikarenakan simpul 2 bertetangga dengan simpul 5 yang berwarna merah dan simpul 6 yang berwarna biru.

Sedangkan untuk konflik pada simpul 3 dan 4 yang memiliki warna kuning tersebut memiliki solusi yang tidak tabu untuk simpul 4 karena simpul 3 tidak memiliki solusi yang tidak tabu. Simpul 3 tidak memiliki solusi yang tidak tabu karena simpul 3 bertetangga dengan simpul 4 yang berwarna kuning, simpul 6 yang berwarna biru, simpul 7 yang berwarna hijau dan simpul 1 yang berwarna merah. Simpul 4 memiliki solusi tabu, yaitu warna biru. Dikarenakan simpul 2 bertetangga dengan simpul 3 yang berwarna kuning, simpul 1 yang berwarna merah dan simpul 7 yang berwarna hijau.

Solusi-solusi tidak tabu yang telah didapatkan dimasukkan ke dalam *tabu list*. Hal itu diperlukan untuk menentukan solusi yang optimal.

- Memilih solusi optimal dari *tabu list*.

Memilih solusi optimal dari *tabu list*, yaitu untuk simpul 2 yang memiliki 2 solusi pada *tabu list*. Dipilih solusi optimal untuk simpul 2 warna kuning. Sebenarnya pemilihan warna dari 2 solusi tersebut

dibebaskan karena keduanya merupakan solusi yang optimal. Sedangkan untuk simpul 4 memiliki solusi optimal warna biru.

- Menerapkan solusi optimal pada graf

Menerapkan solusi optimal yang telah dipilih pada graf. Penggantian warna simpul 2 yang semula berwarna biru menjadi berwarna kuning dan simpul 4 yang semula berwarna kuning menjadi berwarna biru.

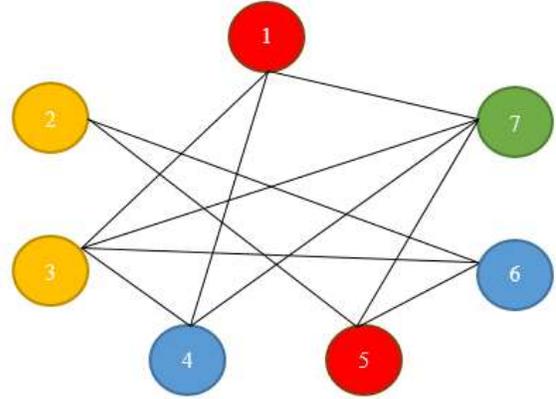


Diagram 5. Graf yang Telah Melewati Langkah 6.
(sumber : dokumentasi penulis)

- Jika masih terdapat konflik, proses akan kembali ke langkah 3. Jika tidak, proses selesai.

Setelah dilakukan pengecekan, tidak ada konflik yang terjadi karena seluruh simpul yang bertetangga memiliki warna-warna yang unik.

E. Penentuan Jadwal Kuliah

Berdasarkan pewarnaan graf yang telah dilakukan, didapatkan jadwal yang sesuai, yaitu:

Tabel 1. Data Jadwal Kuliah

No.	Mata Kuliah	Dosen	Kelas	Hari
1.	MK1	D1	K1	Senin
2.	MK2	D3	K2	Rabu
3.	MK3	D2	K1	Rabu
4.	MK4	D1	K1	Kamis
5.	MK5	D4	K2	Senin
6.	MK6	D2	K2	Kamis
7.	MK7	D4	K1	Selasa

V. KESIMPULAN

Kesimpulan yang didapat adalah penggunaan algoritma *Tabu Search* pada pewarnaan graf dirasa cukup efektif dan berhasil. Sebelum menggunakan algoritma *Tabu Search* diperlukan untuk mengubah data menjadi graf dan menentukan bilangan kromatik dari graf yang ingin diwarnai. Lalu menggunakan algoritma *Tabu Search* dan dapat dihasilkan graf pewarnaan simpul yang sesuai. Pengaplikasian teori pewarnaan graf salah satunya adalah untuk sistem penjadwalan kuliah, dengan teori tersebut penjadwalan akan lebih mudah dan efisien. Dari pengolahan data hingga menjadi jadwal akan meminimalisasi tingkat kesalahan penjadwalan.

VI. UCAPAN TERIMA KASIH

Pertama-tama, penulis ingin mengungkapkan rasa terima kasih kepada Tuhan Yang Maha Esa karena berkat-Nya sehingga penulis dapat menyelesaikan makalah ini dengan baik dan benar. Penulis juga ingin mengucapkan terima kasih kepada orang tua penulis yang senantiasa mendukung penulis dalam hal apapun. Penulis turut mengucapkan rasa terima kasih kepada Bapak Dr. Judhi Santoso M.Sc, selaku dosen penulis pada mata kuliah Matematika Diskrit yang telah mengajarkan penulis mengenai ilmu-ilmu Matematika Diskrit.

DAFTAR PUSTAKA

- [1] Munir, Rinaldi. (2012). *Matematika Diskrit edisi kelima*, Bandung: Informatika.
- [2] Math Insight. *Small Undirected Network with Labeled Nodes and Edges*. Dikases dari http://mathinsight.org/image/small_undirected_network_labeled pada tanggal 2 Desember 2017 Pukul 12.13 WIB.
- [3] Weisstein, Eric W. *Vertex Coloring*. Diakses dari <http://mathworld.wolfram.com/VertexColoring.html> pada tanggal 2 Desember 2017 Pukul 12.40 WIB.
- [4] Weisstein, Eric W. *Edge Coloring*. Diakses dari <http://mathworld.wolfram.com/EdgeColoring.html> pada tanggal 2 Desember 2017 Pukul 12.51 WIB.
- [5] Laurent. 2014. *Adversarial Map Coloring*. Diakses dari <http://www.laurentlessard.com/bookproofs/adversarial-map-coloring/> pada tanggal 3 Desember 2017 Pukul 17.24 WIB.
- [6] Suryani, Ida, dkk. *Implementasi Masalah Pewarnaan Graph dengan Algoritma Tabu Search Pada Penjadwalan Kuliah*. Diakses dari <http://jurnal-online.um.ac.id/data/artikel/artikel834265CE716FA20F8322C32FC587B6F1.pdf>. Pada tanggal 3 Desember 2017 Pukul 19.19 WIB.
- [7] Aladag, C.H. dan Gulsum Hocaoglu. 2007. *A Tabu Search Algorithm to Solve a Course Timetabling Problem*. Hacettepe Journal of Mathematics and Statistics Vol. 36 number 1. Diambil dari dergipark.gov.tr/download/article-file/86919 pada tanggal 3 Desember 2017 Pukul 20.16 WIB.
- [8] Berlianty, I dan M. Arifin. 2010. *Teknik-teknik Optimasi Heuristik*. Yogyakarta: Graha Ilmu.
- [9] Fred Glover. 1990. "Tabu Search – Part 2". *ORSA Journal on Computing*. 2 (1): 4–32. doi:10.1287/ijoc.2.1.4
- [10] M. Gendreau, G. Laporte and F. Semet, "A Tabu Search Heuristic for the Undirected Selective Travelling Salesman Problem," *European Journal of Operational Research*, Vol. 106, 1998, pp. 539-545. doi:10.1016/S0377-2217(97)00289-0
- [11] Priyandari. 2009. *Tabu Search – Introduction*. Diakses dari <http://priyandari.staff.uns.ac.id/200909/tabu-search-introduction/> pada tanggal 3 Desember 2017 Pukul 21.53 WIB.

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, 3 Desember 2017



Muhammad Alif Arifin
13516078