

Aplikasi Pewarnaan Graf dalam Penyelesaian *Nurse Rostering Problem* Sederhana

Nando Rusrin Pratama / 13517148
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13517148@std.stei.itb.ac.id

Abstrak—Salah satu dari banyak aplikasi Graf adalah membantu penyusunan jadwal dengan tingkat kepuasan yang paling optimal. Salah satu cara untuk membuat penjadwalan itu adalah dengan metode *Nurse Rostering Problem* (NRP), atau yang biasa disebut juga *Nurse Scheduling Problem* (NSP). Dimana dalam NRP, ada sekumpulan suster (*nurse*) yang harus dijadwalkan ke beberapa sif kerja dengan aturan-aturan tertentu. Masalah ini dapat diselesaikan dengan mudah dengan menggunakan metode Pewarnaan Graf.

Keywords — Graf, *Nurse Rostering Problem*, *Nurse Scheduling Problem*, Pewarnaan Graf

I. PENDAHULUAN

Dalam sebuah rumah sakit, terdapat berbagai jenis perawat yang harus diharuskan dibagi ke dalam beberapa sif kerja. Pembagian sif kerja biasanya dilakukan oleh suster kepala. Pembagian sif kerja pun merupakan hal yang penting dalam manajemen suatu rumah sakit.

Pembagian antara jumlah anggota dan keterampilan yang dimiliki untuk setiap sif kerja menjadi penting. Kebijakan pembagian sif kerja perawat dapat berdampak langsung pada kepuasan perawat dan juga kualitas kerja serta keselamatan pasien yang dirawatnya.

Untuk itu, pembagian sif kerja yang mengharuskan perawat untuk bekerja dalam kondisi yang sulit dan melelahkan harus dapat dihindari. Maka dari itu, manajemen rumah sakit pun harus dapat meminimalkan tingkat ketidakpuasan para perawatnya. Bagaimanakah cara membuat jadwal pembagian sif kerja agar tingkat kepuasan para perawat optimal?

Contoh permasalahan di atas adalah dasar dari metode *Nurse Rostering Problem* yang akan kita gunakan untuk menyelesaikan permasalahan serupa seperti permasalahan di atas. Selanjutnya, salah satu masalah penjadwalan akan dibahas di makalah ini.

II. DASAR TEORI GRAF

Teori graf pertama kali dikembangkan oleh Leonhard Euler untuk memecahkan masalah jembatan Königsberg (1736). Graf pada dasarnya adalah suatu metode yang digunakan untuk

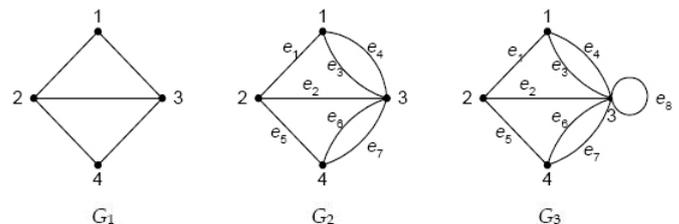
merepresentasikan sekumpulan objek diskrit dan hubungan antara objek-objek tersebut.

Sebuah graf terdiri dari sebuah *simpul* tidak kosong yang merepresentasikan sebuah objek, dan himpunan *sisi* yang merepresentasikan hubungan antara dua buah objek yang dihubungkan.

2.1. Dasar Teori Graf

Graf $G = (V, E)$, adalah koleksi atau pasangan dua himpunan, dimana :

- (1) $V = \{v_1, v_2, v_3, \dots, v_n\}$ yang merupakan himpunan simpul/titik/vertex/point/node, dan
- (2) $E = \{e_1, e_2, e_3, \dots, e_n\}$ yang merupakan himpunan pasangan tak terurut dari simpul, yang disebut ruas/rusuk/sisi/edge/line.



Gambar 2.1.1 Contoh Graf, (G_1) graf sederhana, (G_2) graf ganda, dan (G_3) graf semu

[1] [http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2015-2016/Graf%20\(2015\).pdf](http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2015-2016/Graf%20(2015).pdf), diakses pada 01 Desember 2018

Pada gambar 2.1.1, G_1 adalah graf dengan $V = \{1, 2, 3, 4\}$ dan $E = \{(1, 2), (1, 3), (2, 3), (2, 4), (3, 4)\}$. G_2 adalah graf dengan $V = \{1, 2, 3, 4\}$ dan $E = \{(1, 2), (2, 3), (1, 3), (1, 3), (2, 4), (3, 4), (3, 4)\} = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7\}$. Lalu, G_3 adalah graf dengan $V = \{1, 2, 3, 4\}$ dan $E = \{(1, 2), (2, 3), (1, 3), (1, 3), (2, 4), (3, 4), (3, 4), (3, 3)\} = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8\}$.

Pada gambar 2.1.1, graf G_2 memiliki sisi $e_3 = (1, 3)$ dan $e_4 = (1, 3)$ yang dinamakan **sisi-ganda** (*multiple edges* atau *parallel edges*) karena kedua sisi tadi dihubungkan oleh dua buah simpul yang sama, yaitu simpul 1 dan simpul 3.

Lalu, pada gambar 2.1.1, graf G_3 memiliki sisi $e_8 = (3, 3)$ yang dinamakan **gelang** atau **kalang** (*loop*) karena sisi e_8 berawal dan berakhir pada simpul yang sama, yaitu simpul 3.

2.2. Jenis-Jenis Graf

Berdasarkan ada tidaknya gelang atau sisi ganda, suatu graf dapat digolongkan menjadi dua jenis, antara lain :

(1) Graf Sederhana (*simple graph*)

Graf ini tidak mengandung gelang maupun sisi ganda sehingga dinamakan graf sederhana. Pada gambar 2.1.1, graf G_1 adalah contoh dari sebuah graf sederhana.

(2) Graf Tak-Sederhana (*unsimple graph*)

Graf ini mengandung sisi ganda atau gelang sehingga dinamakan graf tak-sederhana (*unsimple graph*). Pada gambar 2.1.1, graf G_2 dan G_3 adalah contoh dari graf tak-sederhana.

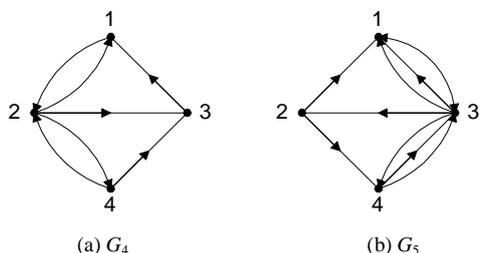
Berdasarkan orientasi arah pada sisi, maka secara umum graf dapat dibedakan menjadi dua jenis, antara lain :

(1) Graf Tak-Berarah (*undirected graph*)

Graf ini sisi-sisinya tidak mempunyai orientasi arah sehingga disebut graf tak-berarah. Pada gambar 2.1.1, graf G_1 , G_2 , dan G_3 adalah contoh dari graf tidak-berarah.

(2) Graf Berarah (*directed graph* atau *digraph*)

Graf ini setiap sisinya diberikan orientasi arah sehingga disebut sebagai graf berarah. Pada gambar 2.2.1, graf G_4 dan G_5 adalah contoh dari graf berarah.



Gambar 2.2.1 (a) graf berarah, (b) graf-ganda berarah
 [1] [http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2015-2016/Graf%20\(2015\).pdf](http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2015-2016/Graf%20(2015).pdf), diakses pada 01 Desember 2018

2.3. Terminologi Graf

Dalam graf, ada beberapa terminologi yang akan mempermudah pembahasan selanjutnya. Berikut adalah beberapa terminologi yang akan digunakan:

(1) Ketetanggaan (*Adjacent*)

Dua buah simpul dikatakan *bertetangga* bila keduanya terhubung langsung. Pada gambar 2.1.1, pada graf G_1 , simpul 1 bertetangga dengan simpul 2 dan 3, dan simpul 1 tidak bertetangga dengan simpul 4.

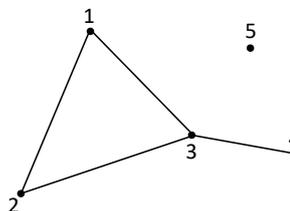
(2) Bersisian (*Incidency*)

Untuk sembarang sisi $e = (v_j, v_k)$ dikatakan e bersisian dengan simpul v_j , atau e bersisian dengan simpul v_k . Pada gambar 2.1.1, pada graf G_1 , sisi (2, 3) bersisian dengan simpul

2 dan simpul 3, sisi (2, 4) bersisian dengan simpul 2 dan simpul 4, tetapi sisi (1, 2) tidak bersisian dengan simpul 4.

(3) Simpul Terpencil (*Isolated Vertex*)

Simpul terpencil adalah simpul yang tidak mempunyai sisi yang bersisian dengannya sama sekali. Pada gambar 2.3.1, simpul 5 adalah sebuah simpul terpencil.



Gambar 2.3.1 Graf dengan simpul terpencil

(4) Derajat (*Degree*)

Derajat suatu simpul adalah jumlah sisi yang bersisian dengan simpul tersebut. Notasi untuk derajat suatu simpul adalah $d(v)$. Pada gambar 2.3.1, pada graf tersebut, $d(1) = d(2) = 2$, $d(3) = 4$, $d(4) = 1$, dan $d(5) = 0$.

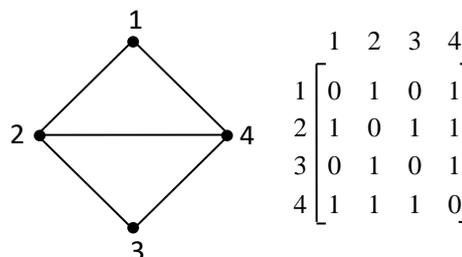
2.3. Representasi Graf

Salah satu cara untuk memrepresentasikan graf adalah dengan menggunakan matriks ketetanggaan (*adjacency matrix*).

$$A = [a_{ij}],$$

$$A_{ij} = \begin{cases} 1, & \text{jika simpul } i \text{ dan } j \text{ bertetangga} \\ 0, & \text{jika simpul } i \text{ dan } j \text{ tidak bertetangga} \end{cases}$$

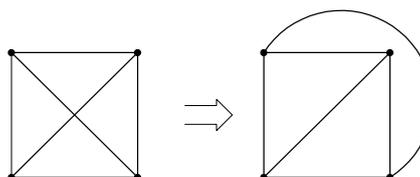
Berikut adalah contoh penggunaan matriks ketetanggaan untuk merepresentasikan graf :



Gambar 2.3.1 Matriks Ketetanggaan

2.4. Graf Planar (*Planar Graph*)

Sebuah graf yang dapat digambarkan pada bidang datar dan sisi-sisinya tidak saling memotong (bersilangan) disebut **graf planar**. Namun, jika sisi-sisinya saling memotong (bersilangan), maka graf tersebut disebut **graf tak-planar**.

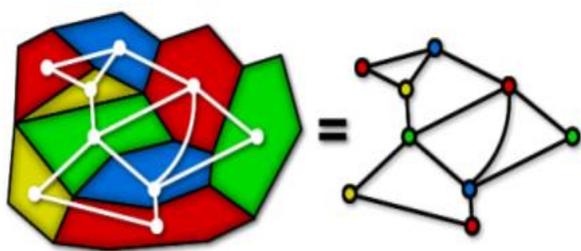


Gambar 2.4.1 Graf planar
 [1] [http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2015-2016/Graf%20\(2015\).pdf](http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2015-2016/Graf%20(2015).pdf), diakses pada 01 Desember 2018

Dalam graf, terdapat banyak sekali aplikasinya di dunia nyata, terutama dalam memodelkan permasalahan-permasalahan yang ada di dunia nyata dalam bentuk yang dapat dipahami oleh komputer sehingga memungkinkan adanya sebuah program yang dapat menyelesaikan permasalahan tersebut.

Masalah seperti *Travelling Salesman Problem*, *Chinese Postman Problem*, hingga mencari Lintasan Terpendek dari ribuan pilihan jalan yang semuanya dimodelkan di computer menggunakan graf. Dan pada makalah ini, akan kita bahas salah satu aplikasi graf yaitu Pewarnaan Graf.

Pewarnaan Graf adalah sebuah tindakan melabeli unsur-unsur dalam graf (seperti sisi atau simpul ataupun keduanya sekaligus) yang sedemikian rupa sehingga tidak ada lagi unsure yang bersinggungan atau bersebelahan memiliki label (warna) yang sama dengan menggunakan jenis label (warna) sesedikit mungkin.



Gambar 2.5.1 Perubahan Peta ke Graf dalam Pewarnaan Peta
 [2] http://world.mathigon.org/resources/Graph_Coloring, diakses pada 02 Desember 2018

Kata “pewarnaan” berawal dari karena metode ini pada awalnya digunakan untuk membantu dalam pewarnaan peta, yang dimana wilayah yang berbatasan harus memiliki warna yang berbeda. Dan karena harga yang cukup mahal maka membuat mereka ingin menggunakan jenis tinta yang sesedikit mungkin.

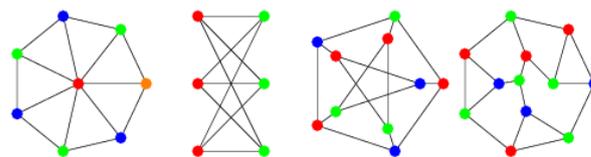
Untuk dapat memodelkan sebuah peta dengan pewarnaan paling sedikit, kita modelkan dulu peta sebagai sebuah graf. Lalu, wilayah-wilayahnya kita lambangkan sebagai simpul-simpul, dan perbatasan tanah antar dua wilayah kita lambangkan sebagai sisi.

Selanjutnya, dengan menggunakan **algoritma Greedy**, kita lakukan langkah berikut :

- (a) Labeli salah satu simpul dengan angka 1
- (b) Labeli lagi salah satu simpul lain yang bersisian dengan salah satu simpul yang telah dilabeli. Gunakan angka terendah yang belum digunakan oleh simpul-simpul yang bersisian dengan simpul yang akan dilabeli. Jika semua label yang ada telah digunakan, buatlah label baru dengan angka yang belum dipakai (angka yang digunakan harus lebih tinggi daripada angka yang sudah ada).

- (c) Ulangi hingga semua simpul yang ada terlabeli
- (d) Ubah label dengan warna, misal label 1 = merah. Perhatikan bahwa tiap label harus memiliki warna yang berbeda.

Algoritma diatas secara umum menggunakan proses yang disebut **Pewarnaan Simpul Graf**, karena yang dilabeli adalah simpul-simpul grafnya, sedangkan sisi yang ada hanya berfungsi sebagai pemberi informasi apakah sepasang simpul boleh berwarna sama atau tidak.

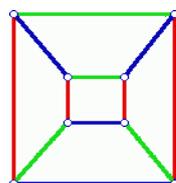


Gambar 2.5.2 Contoh hasil Pewarnaan Simpul Graf
 [2] <https://courses.engr.illinois.edu/lectures/graph/vertexcoloringgraphs>, diakses pada 02 Desember 2018

Metode ini nantinya akan berguna untuk membentuk jadwal. Kita dapat menggunakan algoritma ini untuk membuat sebuah jadwal dengan tingkat kepuasan yang optimal.

Selain metode pewarnaan simpul, pewarnaan graf juga ada metode yang disebut **Pewarnaan Sisi Graf**. Konsep dari keduanya ini cukup mirip, hanya saja kalau tadi yang kita labeli adalah simpul, disini yang kita labeli adalah sisi-sisinya dan simpul hanya berfungsi sebagai pemberi informasi bahwa kedua sisi harus memiliki label yang berbeda. Selain itu, metode yang digunakan relatif serupa.

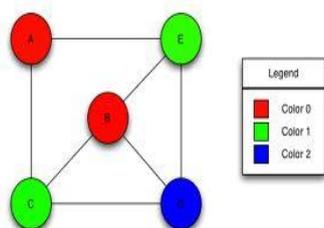
Walaupun secara konsep hampir sama, pewarnaan sisi dan pewarnaan simpul memiliki aplikasi yang berbeda. Contohnya seperti dalam sebuah jadwal pertandingan yang menggunakan sistem *Round Robin* dimana setiap pemain harus menandingi semua pemain lainnya dengan jumlah yang minimum merupakan salah satu contoh aplikasi pewarnaan sisi yang sering digunakan.



Gambar 2.5.2 Contoh hasil Pewarnaan Sisi
 [3] https://www.sci.ufl.edu/papers/coloring_graph_algorithms, diakses pada 02 Desember 2018

Dalam pewarnaan graf, ada jumlah minimum warna yang dibutuhkan untuk pewarnaan peta yang disebut dengan bilangan kromatik. Simbol untuk bilangan kromatik adalah $\chi(G)$. Jika suatu graf G memiliki bilangan kromatis k , maka dilambangkan dengan $\chi(G) = k$. Pada gambar 2.5.3, graf yang ada di gambar memiliki $\chi(G) = 3$.

Dalam perkembangan teorema pewarnaan graf, pada awalnya bilangan kromatik graf planar dikatakan ≤ 6 . Namun, seiring perkembangan, akhirnya terbukti jika bilangan kromatik graf planar ≤ 4 lah yang terbukti kebenarannya. Persoalan ini ditemukan kebenarannya oleh Appel dan Haken yang menggunakan komputer untuk menganalisis hampir 2000 graf yang melibatkan jutaan kasus.



Gambar 2.5.3 Contoh graf yang memiliki $\chi(G) = 3$
 [4] https://www.codeforgeeks/doc/html/boman_et_al_graph_coloring.html, diakses pada 07 Desember 2018

III. NURSE ROSTERING PROBLEM (NRP)

Untuk memiliki perawat yang terbaik dalam bertugas di waktu yang tepat merupakan faktor yang penting bagi sebuah rumah sakit. Sehingga, masalah penjadwalan perawat menjadi hal yang penting bagi rumah sakit. Umumnya, masalah penjadwalan ini telah menjadi permasalahan selama beberapa tahun terakhir. Berbagai upaya pun telah dilakukan untuk mendapatkan jadwal yang optimal.

Tanggung jawab seorang perawat adalah untuk merawat pasien. Namun, akhir-akhir ini, banyak perawat yang bekerja lembur dan bekerja di lingkungan yang mereka yang tidak sukai. Hal ini pun menyebabkan tingkat ketidakpuasan bekerja para perawat meningkat. Dan akibatnya, rumah sakit mengalami penurunan kualitas pelayanan dan mengalami penurunan profit.

Nurse Rostering Problem (NRP) bertujuan untuk menuntaskan permasalahan pembagian sif kerja para perawat di rumah sakit. Selama 50 tahun terakhir, NRP menjadi masalah penting dalam bidang penelitian dan kecerdasan buatan. Hal ini dikarenakan jika ada algoritma NRP yang bagus, maka akan berdampak pada meningkatnya efisiensi sumber daya rumah sakit, keselamatan pasien, dan beban kerja para perawat.

Tujuan utama dari NRP adalah untuk menetapkan jumlah optimal para perawat yang harus bekerja pada setiap sif dan mengoptimalkan kepuasan kerja para perawat. Ada beberapa factor yang pada umumnya mempengaruhi sebuah NSP: peraturan pemerintah, kebijakan rumah sakit, dan status perawat.

Dalam NRP, terdapat tiga istilah penting yang sering digunakan, antara lain :

- (1) *Planning horizon* : jangka waktu yang dicakup oleh rencana tertentu.

- (2) *Shift* : tiap hari dibagi menjadi beberapa sif kerja. Jumlah perawat yang dibutuhkan di setiap sif sudah ditentukan sebelumnya.
- (3) *Off day* : hari dimana ketika seorang perawat tidak memiliki *shift* untuk bekerja.

Pada gambar 3.1 dibawah, akan ditunjukkan contoh sebuah tabel NSP.

Shifts	Day 1			Day 2			...	Day 7			Number of shifts
	M	E	N	M	E	N	...	M	E	N	
Nurse 1	█			█			...	█			6
Nurse 2		█			█		...		█		7
Nurse 3			█			█	...			█	7
Nurse 4				█			...		█		7
.....
Nurse J		█			█		...		█		6
Number of nurses assigned	15	18	13	13	18	13	...	13	14	13	314

Gambar 3.1 The planning horizon is days, each day is divided into a number of shifts (e.g. morning, evening, and night), each shift has a pre-specified number of nurses should be satisfied while considering the maximum number of shifts that a nurse can work.

[5] <https://www.sciencedirect.com/science/article>, diakses pada 08 Desember 2018

Berikut peraturan pemerintah dan kebijakan rumah sakit, yang pada umumnya harus dipenuhi dalam membuat NRP :

- (1) Para perawat hanya memiliki paling banyak dua sif kerja dalam sehari.
- (2) Setiap perawat memiliki jumlah sif kerja tengah malam yang telah ditentukan sebelumnya.
- (3) Setiap perawat memiliki jumlah sif kerja malam yang telah ditentukan sebelumnya.
- (4) Setiap perawat bekerja dalam rentang jumlah sif kerja minimum dan maksimum yang telah ditentukan sebelumnya.
- (5) Jika ada perawat yang bekerja dua sif berturut-turut, maka untuk tiga sif berikutnya ia tidak diperbolehkan bekerja.

Ketentuan yang lainnya, biasanya diatur dalam model matematika yang digunakan untuk menyelesaikan masalah. Selanjutnya, dalam makalah ini, kita akan menggunakan model matematika yang disusun berdasarkan *Multi-Commodity Network Flow* (MCF). MCF adalah sebuah metode yang terkait dengan masalah aliran jaringan dengan beberapa komoditas yang sumbernya berbeda dan simpulnya tenggelam.

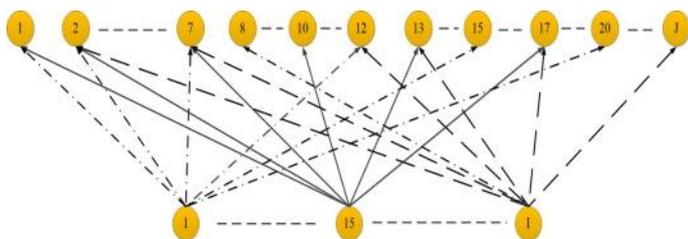
Dalam masalah MCF, beberapa komoditas dengan kuantitas yang diketahui untuk diangkut dari sejumlah simpul sumber ke sejumlah simpul tenggelam dengan tujuan mengangkut komoditas ini dengan biaya minimum sambil memenuhi persyaratan permintaan dari simpul tenggelam. Dari sini, NRP dapat dilihat sebagai masalah MCF. Simpul sumber adalah

perawat yang tersedia untuk bekerja, simpul tenggelam adalah *shift* di hari yang berbeda dalam *planning horizon*, sementara permintaan pada simpul tenggelam adalah jumlah perawat yang diperlukan dalam setiap *shift* dalam *planning horizon*. Komoditas di sini adalah: perawat kepala, perawat senior, dan perawat pelatihan.

3.1. Memodelkan Permasalahan

Pada bagian ini, model yang digunakan akan menangkap sebagian besar aspek yang terkait dengan NRP yang kita harapkan. Dalam model ini, sebagian besar rumah sakit, menganggap *planning horizon* mereka sebagai seminggu. Setiap hari dalam seminggu dibagi menjadi tiga jenis *shift* (yaitu pagi, sore, dan *shift* malam). Selain *shift* pagi, seorang perawat harus bekerja dengan jumlah *shift* sore dan malam tertentu per minggu, biasanya dua *shift* siang, dan satu *shift* malam. Setiap *shift* adalah delapan jam, dan perawat tidak dapat bekerja lebih dari dua *shift* berurutan.

Model matematika yang diusulkan didasarkan pada gagasan model MCF. Pada gambar 3.1 dibawah, akan diberikan sketsa model yang dipakai.



Gambar 3.1.1 Model MCF
 [5] <https://www.sciencedirect.com/science/article>, diakses pada 08 Desember 2018

Parameter yang digunakan :

- c1: Gaji perawat per *shift*
- c2: Gaji perawat per *shift* lembur
- c3: Gaji perawat kepala per *shift*
- I : Set perawat {1,2, ..., H, H + 1, H + 2,, I}
- J : Set *shift* {1,2, 3, J} dalam *planning horizon*
- U_i : Jumlah *shift* maksimum
- N_i : Jumlah *shift* minimum
- M_j : Jumlah perawat minimal dalam sebuah *shift*
- A : Jumlah perawat minimal dalam sebuah *shift* sore
- B : Jumlah perawat minimal dalam sebuah *shift* malam
- F : Jumlah perawat kepala minimal dalam sebuah *shift*

$$f_{ij} = \begin{cases} 1, & \text{jika perawat adalah perempuan} \\ 0, & \text{jika sebaliknya} \end{cases}$$

Variable Bebas :

$$x_{ij} = \begin{cases} 1, & \text{jika simpul } i \text{ dan } j \text{ bertetangga} \\ 0, & \text{jika simpul } i \text{ dan } j \text{ tidak bertetangga} \end{cases}$$

Fungsi yang digunakan :

$$\text{Min } \sum_{i=H+1}^I \sum_{j=1}^J (c_1 * x_{ij}) + c_2 [(\sum_{i=H+1}^I \sum_{j=1}^J x_{ij}) - N_i + \sum_{i=1}^H \sum_{j=1}^J (c_3 * x_{ij})]$$

3.2. Aplikasi Model Permasalahan Dalam Penyelesaian NRP Sederhana

Dalam aplikasi ini, model yang kita gunakan akan disederhanakan sehingga yang diperhatikan hanyalah status perawat untuk membentuk jadwal yang optimal. Berikut permasalahan yang akan kita selesaikan.

Kali ini, kita anggap ada lima belas perawat. Jenis-jenis perawat yang ada, antara lain Staf Perawat (SN), Kepala Perawat Terdaftar (PEN), Perawat Rotasi (RN), Perawat Terdaftar (EN), Asisten Asisten Senior (SWA) dan Penyuluh Kesehatan (HEW). Lalu, semua perawat diberi nama SN1, SN2, SN3, SN4, PEN, RN1, RN2, EN1, EN2, SWA, HEW1, HEW2, HEW3, HEW4, HEW5.

Selanjutnya, kita mendaftarkan para perawat yang tidak suka bekerja bersama kedalam beberapa group. Manajemen rumah sakit ingin para perawatnya yang tidak suka bekerja bersama untuk dibagi ke dalam jadwal *shift* yang berbeda untuk menciptakan jadwal yang optimal.

A	SN1,SN3,SN4
B	SN4,EN1,EN2
C	RN1,RN2
D	EN2,HEW3
E	HEW2,HEW3,HEW4
F	RN1,EN1
G	EN1,EN2

Tabel 3.2.1 Kelompok Perawat yang Tidak Suka Bekerja Bersama

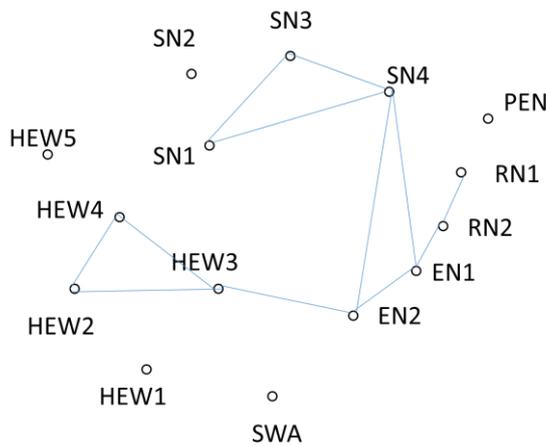
Tabel diatas dibentuk dengan mempertimbangkan perbedaan kemampuan para perawat dan juga tingkat kepuasan para perawat jika diletakkan dalam sif kerja yang sama.

Berdasarkan tabel 3.2.1, akan kita bentuk matriks ketetanggaannya. Berikut matriks ketetanggaan (*adjacency matrix*) dari tabel 3.2.1 :

Adjacency Matrix															
	SN1	SN2	SN3	SN4	PEN	RN1	RN2	EN1	EN2	SWA	HEW1	HEW2	HEW3	HEW4	HEW5
SN1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
SN2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SN3	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
SN4	1	0	1	0	0	0	0	1	1	0	0	0	0	0	0
PEN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RN1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0
RN2	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
EN1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0
EN2	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0
SWA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HEW1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HEW2	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
HEW3	0	0	0	0	0	0	0	0	1	0	0	1	0	1	0
HEW4	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
HEW5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

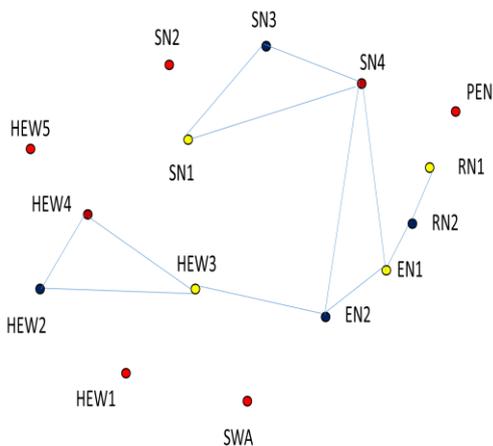
Tabel 3.2.2 Matriks Ketetanggaan

Dari matriks pada tabel 3.2.2, dapat kita bentuk graf planar. Berikut adalah graf planar dari matriks pada tabel 3.2.2 :



Gambar 3.2.1 Graf Planar

Dari sini, dengan menggunakan Algoritma Greedy, kita lakukan proses pewarnaan simpul graf. Berikut graf yang telah diwarnai.



Gambar 3.2.2 Graf Planar yang Simpulnya Telah Diwarnai

Dari gambar 3.2.2, dapat kita bentuk sebuah tabel yang berisikan kelompok-kelompok perawat yang dapat bekerja di sif yang sama. Berikut tabelnya.

Kelompok	Jenis Perawat
K ₁	SN1,EN1,RN1,HEW3
K ₂	SN3,EN2,RN2,HEW2
K ₃	SN4,HEW4
K ₄	SN2,PEN,SWA,HEW1,HEW5

Tabel 3.2.3 Kelompok Perawat yang Dapat Bekerja Sama

Pada tabel 3.2.3, berisi kelompok perawat yang dapat bekerja pada sif yang sama. Para perawat di K₁, K₂, dan K₃, hanya dapat bekerja pada sif kelompok yang sama. Namun, para perawat di K₄ dapat bekerja sif manapun. Hal ini dikarenakan dalam graf, mereka merupakan simpul terpercil atau simpul bebas.

KESIMPULAN

NRP merupakan salah satu masalah penting dalam administrasi suatu rumah sakit. Dalam NRP, minimalisasi biaya rumah sakit dilakukan secara keseluruhan sambil mempertimbangkan peraturan yang ada. Dengan metode yang dikembangkan berdasarkan gagasan model MCF, kita dapat membuat metode pemecahan masalah yang baru yang bekerja dengan sangat baik. Model yang dibuat pun dapat menangkap kendala-kendala yang realitis terkait masalah seperti kendala dalam pemerintahan, kebijakan rumah sakit, dan standar global di bidang kesehatan. Lebih lanjut, model ini juga membuktikan bahwa jadwal-jadwal yang dihasilkan secara manual dengan NRP dapat meningkatkan tingkat kepuasan perawat dengan membuat sistem jadwal yang memperhatikan preferensi perawat.

REFERENSI

- [1] J. Van Den Bergh, J. Beliën, P. De Bruecker, E. Demeulemeester, L. De Boeck, *Personnel scheduling: A literature review*. Eur. J. Oper. Res., 226 (2013), pp. 367-385.
- [2] W. Stimpfel, M. Sloane, H. Aiken, *The longer the shifts for hospital nurses, the higher the levels of burnout and patient dissatisfaction*. Health Affairs, 31 (11) (2011), pp. 2501-2509
- [3] A. Legrain, H. Bouarab, N. Lahrichi, *The nurse scheduling problem in real-life*. J. Med. Syst., 39 (2014), pp. 1-11
- [4] H. Jafari, N. Salmasi, *Maximizing the nurses' preferences in nurse scheduling problem: Mathematical modeling and a meta-heuristic algorithm*. J. Ind. Eng. Int., 11 (2015), pp. 439-458
- [5] M. Akbari, M. Zandieh, B. Dorri, *Scheduling part-time and mixed-skilled workers to maximize employee satisfaction*. Int. J. Adv. Manuf. Technol., 64 (2013), pp. 1017-1027
- [6] Ko, Young-Woong; Kim, Donghoi; Jeong, Minyeong; Jeon, Wooram; Uhm, Saangyong; Kim, Jin (July 2013). "An Improvement Technique

for Simulated Annealing and Its Application to Nurse Scheduling Problem". International Journal of Software Engineering and Its Applications. Science & Engineering Research Support Society. 7 (4): 269–278

- [7] Aickelin, Uwe; White, Paul (2004). "*Building Better Nurse Scheduling Algorithms*". 128: 159–177.
- [8] Munir, Rinaldi, Diktat Kuloiah IF2120, Matematika Diskrit, Edisi Keempat, Program Studi Teknik Informatika, STEI, ITB, 2006

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, 8 Desember 2018



Nando Rusrin Pratama – 13517148