

Optimalisasi Susunan Tempat Duduk Kereta Api Menggunakan Algoritma *Greedy* dan Program Dinamis

Fildah Ananda Amalia - 13515127
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13515127@std.stei.itb.ac.id

Abstract—Perjalanan menggunakan kereta saat ini sudah mulai menjadi favorit masyarakat Indonesia. Pelayanan yang terbaik akan selalu diberikan termasuk kenyamanan penumpang. Pada makalah ini akan dibahas mengenai bagaimana metode untuk mengatur konfigurasi tempat duduk penumpang sesuai dengan jumlah pemesanan tiketnya agar didapatkan kenyamanan bagi para penumpangnya. Metode ini akan menggunakan algoritma greedy dan program dinamis.

Keywords—*greedy, program dinamis, denah tempat duduk kereta api.*

I. PENDAHULUAN

Mobilitas masyarakat kini semakin tinggi. Adanya teknologi informasi yang semakin canggih dan maju, masyarakat kini dengan mudah terhubung dengan lokasi-lokasi yang terletak jauh dari tempat tinggalnya. Kemajuan teknologi informasi merupakan salah satu hal yang menyebabkan masyarakat saat ini cenderung lebih suka untuk melakukan perjalanan dari satu tempat ke tempat yang lain.

Alasan lain yang juga turut membuat tingkat mobilitas masyarakat meningkat adalah perbaikan pelayanan transportasi umum. Belum lama ini terjadi perbaikan yang cukup signifikan pada kereta api Indonesia mengenai pelayanannya. Gerbong-gerbong diperbaiki, peraturan diperketat, sistem pembelian tiket semakin mudah, dan berbagai kemudahan lain yang diberikan oleh PT KAI. Maka sekarang kereta api pun menjadi sebuah layanan favorit masyarakat karena harganya yang bervariasi namun dibarengi dengan ketepatan waktu keberangkatan dan kedatangan yang cukup baik.

Sistem booking tiket yang tersedia di internet juga banyak membantu masyarakat dalam menikmati layanan kereta api. Pada saat memesan tiket, maka sistem pada pertama kali akan otomatis memilihkan tempat duduk yang kosong sesuai dengan kebutuhan berdasarkan kursi kosong paling awal yang ditemukan. Pada beberapa kasus hal ini tidak cukup sesuai sebab pada kereta api ekonomi yang kursinya berhadapan, maka akan lebih nyaman jika orang yang berada di sekitar kita merupakan orang yang kita kenal. Apabila naik kereta api secara rombongan, maka akan lebih baik jika kursi yang disediakan berdekatan satu sama lain atau bahkan di tempat duduk yang berhadapan.

Pada sistem pemesanan tiket kereta api sebetulnya tersedia fitur untuk berpindah tempat duduk ke tempat yang lain. Namun pada denah yang diberikan terkadang belum cukup informasi yang dibutuhkan untuk menentukan kursi yang akan ditempati. Maka dengan adanya algoritma ini untuk pengaturan tempat duduk diharapkan dapat memberikan saran untuk para pengguna moda transportasi kereta api agar bisa mendapatkan aturan tempat duduk yang lebih nyaman

II. DASAR TEORI

A. Algoritma Greedy

Algoritma Greedy merupakan sebuah algoritma pencarian hasil optimal dengan cara mengambil nilai yang paling optimal pada setiap langkah yang diambilnya. Maka dari itu algoritma ini membentuk solusi langkah per langkah. Namun setiap langkah yang sudah diambil tidak akan bisa diubah lagi pada langkah selanjutnya.

Persoalan optimasi dalam konteks algoritma greedy disusun oleh elemen-elemen sebagai berikut:

1. Himpunan kandidat, C .

Himpunan ini berisi elemen-elemen pembentuk solusi. Contohnya adalah himpunan koin, himpunan job yang akan dikerjakan, himpunan simpul dalam graf, dan lain-lain. Pada setiap langkah, satu buah kandidat diambil dari himpunnanya.

2. Himpunan solusi, S .

Berisi kandidat-kandidat yang terpilih sebagai solusi persoalan. Dengan kata lain, himpunan solusi adalah himpunan bagian dari himpunan kandidat.

3. Fungsi seleksi

Biasa diinyatakan dengan predikat SELEKSI, yaitu fungsi yang pada setiap langkah memilih kandidat yang paling memungkinkan mencapai solusi optimal. Kandidat yang sudah dipilih pada suatu langkah tidak pernah dipertimbangkan lagi pada langkah selanjutnya. Biasanya setiap kandidat, x , di-assign sebuah nilai numeric, dan fungsi seleksi memilih x yang

mempunyai nilai terbesar atau memilih x yang mempunyai nilai terkecil.

4. Fungsi kelayakan

Biasanya dinyatakan dengan predikat LAYAK, yaitu fungsi yang memeriksa apakah suatu kandidat yang telah dipilih dapat memberikan solusi yang layak, yakni kandidat tersebut bersama-sama dengan himpunan solusi yang sudah terbentuk tidak melanggar kendala (constrains) yang ada. Kandidat yang layak dimasukkan ke dalam himpunan solusi, sementara yang tidak layak dibuang dan tidak pernah dipertimbangkan lagi.

5. Fungsi obyektif

Fungsi obyektif merupakan fungsi yang memaksimumkan atau meminimumkan nilai solusi (misalnya panjang lintasan, keuntungan, dan lain-lain)

Kata "optimisasi" dapat berarti minimisasi atau maksimasi, bergantung pada persoalan yang dipecahkan. Misalnya pada masalah penukaran uang, kita melakukan minimisasi pada jumlah koinnya.

Sebagai contoh, tinjau masalah penukaran uang, maka elemen-elemen algoritma greedy nya adalah:

1. Himpunan kandidat: koin-koin yang merepresentasikan nilai 1, 5, 10, 25, paling sedikit mengandung satu koin untuk setiap nilai.
2. Himpunan solusi: koin-koin yang terpilih sehingga total nilai koin seluruhnya tepat sama jumlahnya dengan nilai uang yang ditukarkan.
3. Fungsi seleksi: fungsi seleksinya akan memilih nilai koin yang terbesar dari nilai himpunan kandidat yang tersisa.
4. Fungsi layak: memeriksa apakah nilai total dari himpunan koin yang dipilih tidak melebihi jumlah uang yang harus dibayar.
5. Fungsi obyektif: jumlah koin yang digunakan harus yang paling sedikit.

Algoritma greedy tidak selalu menghasilkan solusi yang benar-benar optimum. Pada kasus ini, algoritma greedy sering kali menjadi menjadi basis untuk pendekatan heuristic. Selanjutnya jika persoalan dapat dipecahkan secara eksak (memberikan solusi yang optimum) dengan algoritma greedy, maka pembuktian kebenaran dari algoritma greedy tersebut mungkin merupakan proses yang non-trivial.

B. Program Dinamis

Program dinamis merupakan sebuah metode penyelesaian masalah dengan cara menguraikan solusi menjadi sekumpulan langkah atau tahapan sedemikian sehingga solusi dalam suatu persoalan dapat dipandang dari serangkaian keputusan yang saling berkaitan. Pada persoalan yang diselesaikan dengan metode ini haruslah terdapat sejumlah berhingga pilihan yang mungkin, solusi pada setiap tahap dibangun dari solusi yang

didapatkan pada langkah sebelumnya, kita menggunakan persyaratan optimasi dan Kendala untuk membatasi sejumlah pilihan yang harus dipertimbangkan dalam suatu tahap.

Berikut ini merupakan karakteristik persoalan yang dapat diselesaikan dengan metode program dinamis.

1. Persoalan dapat dibagi menjadi beberapa tahap yang pada setiap tahap hanya diambil satu keputusan
2. Masing-masing tahap terdiri dari sejumlah status yang berhubungan dengan tahap tersebut. Secara umum status merupakan bermacam kemungkinan masukan yang ada pada tahap tersebut. Jumlah status bisa berhingga maupun tak hingga.
3. Hasil dari keputusan yang diambil pada setiap tahap ditransformasikan dari status yang bersangkutan ke status berikutnya pada tahap berikutnya.
4. Ongkos pada suatu tahap meningkat secara teratur dengan bertambahnya jumlah tahapan.
5. Ongkos pada suatu tahap bergantung pada ongkos tahap-taha yang sudah berjalan dan ongkos pada tahap tersebut.
6. Keputusan terbaik pada suatu tahap bersifat independen terhadap keputusan yang dilakukan pada tahap sebelumnya
7. Adanya hubungan rekursif yang mengidentifikasi keputusan terbaik untuk setiap status pada tahap k memberikan keputusan terbaik pada setiap status pada tahap $k+1$
8. Prinsip optimaalitas berlaku pada persoalan tersebut.

Dalam menyelesaikan program dinamis, terdapat dua pendekatan, yaitu maju atau mundur. Untuk program dinamis maju, maka tahap-tahap yang dilakukan dimulai dari tahap pertama, kedua, dan seterusnya hingga tahap ke- n . Sementara untuk program dinamis mundur, maka tahap yang dilakukan dimulai dari tahap ke- n hingga tahap yang pertama.

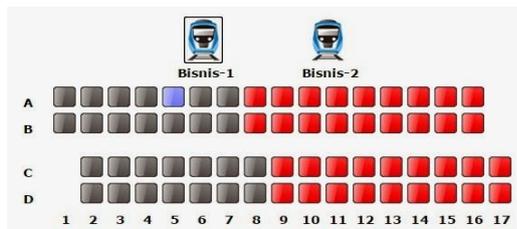
Secara umum ada empat langkah yang dilakukan dalam mengembangkan algoritma program dinamis:

1. Karakteristikan struktur solusi optimal
2. Definisikan secara rekursif nilai solusi optimal
3. Hitung nilai solusi optimal secara maju atau mundur
4. Konstruksikan solusi optimal

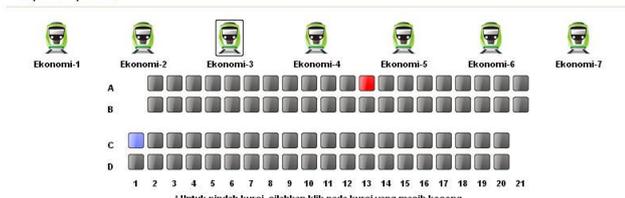
Perlu dicatat bahwa solusi optimal yang dihasilkan oleh program dinamis bisa lebih dari satu buah solusi.

C. Denah Tempat Duduk Kereta Api

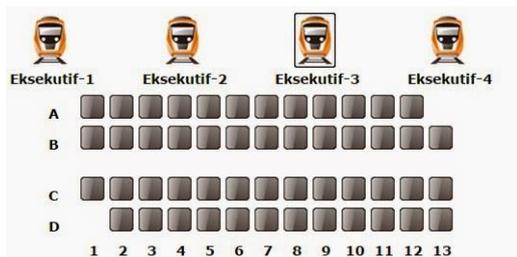
Kereta Api Indonesia memiliki beberapa kelas yang berbeda. Kelas-kelasnya yaitu eksekutif, bisnis dan ekonomi. Tiap-tiap kelas memiliki konfigurasi tempat duduk yang berbeda.



Gambar 1. Denah tempat duduk kereta kelas bisnis



Gambar 2. Denah tempat duduk kereta kelas ekonomi



Gambar 3. Denah tempat duduk kereta kelas eksekutif

III. PEMBAHASAN

Berdasarkan kondisi yang sudah dijelaskan pada bagian pendahuluan, di bagian ini akan dilakukan pembahasan mengenai persoalan yang kita hadapi. Pada persoalan ini saya pilih kereta api kelas ekonomi sebab pada tempat duduk yang tersedia di gerbong-gerbong kereta api kelas ekonomi merupakan tempat duduk yang permanen (tidak bisa dibalik) dan berhadapan dengan penumpang yang lain. Sehingga mungkin menimbulkan sedikit ketidaknyamanan pada sebagian pengguna.

A. Pemodelan Tempat Duduk

Tempat duduk akan dimodelkan menjadi sebuah matrik dengan dimensi $M \times N$ dimana M merepresentasikan banyaknya baris yang tersedia dan N merepresentasikan banyaknya jumlah kursi yang ada pada suatu baris. Jika dilihat dari denah tempat duduk yang sudah ditampilkan sebelumnya, maka N akan bernilai 4 dan M akan bernilai 21.

1,1	1,2	1,3	1,4
2,1	2,2	2,3	2,4
3,1	3,2	3,3	3,4
4,1	4,2	4,3	4,4
5,1	5,2	5,3	5,4
6,1	6,2	6,3	6,4
...
M,1	M,2	M,3	M,4

Gambar 4. Ilustrasi pemodelan denah tempat duduk menjadi suatu matriks

B. Elemen Algoritma Greedy

Pada persoalan ini, dirumuskan elemen-elemen algoritma greedynya sebagai berikut:

1. Himpunan kandidat
Nilai-nilai yang merepresentasikan berapa jumlah tempat duduk kosong yang saling berdekatan
2. Himpunan solusi
Tempat duduk yang terpilih atau terisi oleh penumpang
3. Fungsi seleksi
Fungsi ini akan memilih tempat duduk dengan banyaknya tempat duduk kosong minimal
4. Fungsi layak
Fungsi kelayakannya adalah memeriksa apakah nilai tempat duduk minimal tersebut lebih besar atau sama dengan banyaknya kursi yang dipesan oleh penumpang. Namun pada kasus ini akan dilakukan pembuatan fungsi layak yang sedikit lebih kompleks menyesuaikan dengan constraint yang diberikan oleh kasus yang ada.
5. Fungsi objektif
Fungsi objektifnya adalah untuk mendapatkan jumlah kursi kosong yang tersisa paling sedikit dalam suatu gerbong

C. Fungsi Program Dinamis

Pada permasalahan ini, program dinamis akan dilakukan untuk mencari nilai optimum pada setiap tahap dimana tahap ini berarti sesuai dengan antrian pemesanan tiket yang dilakukan oleh penumpang.

Misalkan ada sejumlah 5 orang pemesan tiket dan mereka memesan sejumlah 1, 2, 3, 4, 5 buah tiket secara berurutan, maka program dinamis akan melakukan tahap penyelesaiannya sesuai dengan pemesanan tiket yang paling awal dilakukan dan seterusnya sesuai dengan antrian yang terbentuk.

Pada setiap tahap program dinamis yang dilakukan, akan terjadi suatu pemilihan solusi yang diselesaikan dengan

algoritma greedy untuk menentukan lokasi tempat duduk yang dipilih.

Pada setiap status akan disimpan konfigurasi tempat duduk yang telah terisi dan informasi mengenai tempat duduk mana saja yang belum terisi untuk kemudian dijadikan acuan untuk melakukan pengambilan keputusan di tahap yang selanjutnya.

D. Penyelesaian Kasus

Sebelum melakukan penyelesaian terhadap permasalahan ini, terlebih dahulu akan diberikan struktur data yang akan dipakai. Struktur data ini akan memudahkan penyimpanan informasi yang akan dijadikan informasi pada setiap tahap program dinamis. Selain itu diperlukan juga fungsi-fungsi yang akan dipakai untuk mempermudah pengambilan informasi seperti fungsi untuk menghitung jumlah kursi kosong pada suatu blok dan lain-lain. Berikut ini penjelasan dari item-item tersebut.

1) Struktur Data

Untuk menyimpan lokasi-lokasi tempat duduk yang kosong, akan diberikan sebuah struktur data yang berisi nilai baris dan kolom pada matriks yang kosong beserta berapa jumlah kursi kosong yang berdekatan dengan kursi tersebut. Pada penyimpanan lokasi nilai disini, akan dipilih kursi yang paling awal untuk setiap blok kursi yang kosong. Nilai jumlah kursi yang kosong untuk setiap bloknya akan menjadi himpunan kandidat pada algoritma greedy yang akan dilakukan pada setiap tahapnya. Misalkan struktur data yang menyimpan informasi kursi dan jumlah kursi yang kosong pada suatu blok ini dinamakan Kosong yang berisi data mengenai baris, kolom, dan jumlah kursi yang masing-masing bertipe integer. Tipe data Kosong ini nantinya akan dimasukkan dalam sebuah array untuk mempermudah penyimpanan informasinya. Array ini bersifat dinamis sehingga memungkinkan penambahan dan pengurangan jumlah kontennya.

Indeks arr	1	2	3	4
Data blok kursi kosong	<1,1,8>	<3,1,8>	<5,3,4>	<7,2,7>

Gambar 5. Ilustrasi penyimpanan tipe data Kosong

2) Fungsi Menghitung Kursi Kosong pada Blok

Kursi pada gerbong kelas ekonomi akan dibuat secara blok per 2 baris yang berhadapan. Pada awal pertama pengisian penumpang, kondisi yang terjadi adalah semua kursi kosong. Hal ini berarti tipe data Kosong akan menyimpan lokasi kursi {(1, 1, 8), (3, 1, 8), (5, 1, 8)} dan seterusnya hingga nomor kursi yang terakhir. Ketika ada pengisian penumpang, maka kursi akan dibooking secara kontigu sehingga nilai lokasi kursi kosong dan jumlah kursi kosongnya akan diubah sesuai dengan berapa banyak pengisian kursinya. Hal ini dilakukan agar penempatan

penumpang yang memesan secara rombongan tidak terjadi secara bertolak punggung atau saling memunggungi.

1,1	1,2	1,3	1,4
2,1	2,2	2,3	2,4
3,1	3,2	3,3	3,4
4,1	4,2	4,3	4,4
5,1	5,2	5,3	5,4
6,1	6,2	6,3	6,4
7,1	7,2	7,3	7,4
8,1	8,2	8,3	8,4

Gambar 6. Ilustrasi penghitungan kursi kosong dan penanda lokasinya

3) Program Utama

Setelah dibuat struktur data yang khusus dan fungsi tambahan yang secara spesifik akan membantu menyelesaikan persoalan ini, maka sebuah struktur program dinamis bisa mulai dirancang.

Pada setiap tahap program dinamis dilakukan beberapa hal sebagai berikut:

1. Input data nilai jumlah tiket yang dipesan dalam setiap waktu
2. Berdasarkan nilai jumlah tiket yang dipesan dalam suatu waktu, akan dilakukan pengecekan secara greedy terhadap data kursi kosong yang tersedia. Algoritma greedy akan memilih tempat duduk dengan jumlah kursi kosong yang paling sedikit namun masih bisa memuat seluruhnya
3. Setelah itu data pada array penyimpanan akan diupdate sesuai dengan kursi yang sudah dipakai dan sisa kursi kosongnya.
4. Array penyimpanan tersebut kemudian akan digunakan untuk program dinamis tahap selanjutnya dengan data pemesanan tiket yang setelahnya.

Untuk lebih jelasnya akan dijelaskan dengan ilustrasi berikut ini.

Kasus:

Misalkan tersedia gerbong kereta dengan konfigurasi tempat duduk sebanyak 4 baris dan dalam satu barisnya terdapat 4 buah kursi yang berjajar.

1,1	1,2	1,3	1,4
2,1	2,2	2,3	2,4
3,1	3,2	3,3	3,4
4,1	4,2	4,3	4,4

Di awal penyelesaian akan dibuat array yang menyimpan tipe data kosong berdasarkan konfigurasi tempat duduk yang ada. Dalam kasus ini, maka array tersebut akan terlihat seperti berikut.

Indeks arr	1	2
Data blok kursi kosong	<1,1,8>	<3,1,8>

Misalkan ada pemesan tiket dengan banyaknya pemesanan tiket per pemesanan sebanyak 5, 2, 4, 1. Kemudian berdasarkan urutan tersebut, akan dilakukan penyelesaian dengan program dinamis.

Tahap 1:

Dengan nilai jumlah pemesanan 5, maka pemesan ini akan mendapatkan tempat duduk di (1,1), (1,2), (2,1), (2,2) dan (1,3), sehingga data pada array penyimpanan tipe data kosong akan diupdate menjadi seperti berikut.

Indeks arr	1	2
Data blok kursi kosong	<1,4,3>	<3,1,8>

Sementara itu, konfigurasi tempat duduknya akan terlihat seperti di bawah ini.

1,1	1,2	1,3	1,4
2,1	2,2	2,3	2,4
3,1	3,2	3,3	3,4
4,1	4,2	4,3	4,4

Keterangan: yang diwarnai biru berarti kursi sudah terisi

Tahap 2:

Pemesanan yang kedua memiliki jumlah kursi yang dipesan sebanyak 2 kursi. Berdasarkan algoritma greedy, maka penumpang ini akan mendapatkan kursi pada lokasi yang disimpan oleh tipe data Kosong pada indeks pertama array karena memiliki jumlah kursi kosong sebanyak 3 buah. Maka penumpang ini akan mendapatkan nomor kursi (1,4), (2,3).

Berikut ini kondisi array dan tampilan konfigurasi tempat duduk yang tersedia.

Indeks arr	1	2
Data blok kursi kosong	<2,4,1>	<3,1,8>

1,1	1,2	1,3	1,4
2,1	2,2	2,3	2,4
3,1	3,2	3,3	3,4
4,1	4,2	4,3	4,4

Tahap 3:

Dengan nilai jumlah pemesanan sebanyak 4, maka algoritma greedy akan menentukan penempatan kursi penumpang berdasarkan lokasi yang disimpan oleh tipe data Kosong pada indeks yang kedua. Maka penumpang ini akan mendapatkan kursi pada lokasi (3,1), (3,2), (4,1), dan (4,2).

Hasil program dinamis pada tahap ini akan terlihat seperti berikut:

Indeks arr	1	2
Data blok kursi kosong	<2,4,1>	<3,3,4>

1,1	1,2	1,3	1,4
2,1	2,2	2,3	2,4
3,1	3,2	3,3	3,4
4,1	4,2	4,3	4,4

Tahap 4:

Pemesan yang terakhir memesan tiket sejumlah 1 buah, maka algoritma greedy akan mengambil lokasi tempat duduk berdasarkan lokasi yang disimpan oleh tipe data Kosong yang pertama.

Sehingga kondisi terakhir konfigurasi tempat duduk dan tipe data Kosong yang tersimpan seperti berikut.

Indeks arr	1	2
Data blok kursi kosong	<2,4,0>	<3,3,4>

1,1	1,2	1,3	1,4
2,1	2,2	2,3	2,4
3,1	3,2	3,3	3,4
4,1	4,2	4,3	4,4

IV. KESIMPULAN

Persoalan optimasi pemesanan tempat duduk ini sebenarnya bertujuan untuk memberikan kenyamanan pada penumpang agar tidak terpisah jauh dengan teman perjalanannya, bukan hanya untuk optimalisasi secara jumlahnya.

Dengan metode penyelesaian ini, diharapkan bisa dijadikan sebuah algoritma dalam menentukan lokasi tempat duduk penumpang secara otomatis dari situs pembelian tiket sehingga penumpang tidak perlu lagi mengubah tempat duduknya dengan cara manual. Hal ini menurut penulis sangat membantu karena koneksi internet bisa saja mendadak tidak stabil sewaktu-waktu.

Namun metode ini masih memiliki beberapa kekurangan, yaitu untuk pemesanan tiket dengan jumlah lebih dari 8 kursi masih perlu dibuat suatu metode khusus agar penempatan kursinya bisa lebih baik.

UCAPAN TERIMA KASIH

Syukur senantiasa Penulis ucapkan Allah SWT bahwa hanya dengan nikmat dan karunia-Nya makalah ini bisa terselesaikan. Penulis juga berterima kasih kepada dosen

pengampu mata kuliah IF2211 Strategi Algoritma yang telah memberikan ilmunya sehingga bisa menjadi bahan dasar dari pembahasan makalah ini. Tidak lupa juga kepada teman-teman IF 15 yang senantiasa memberikan dukungan dan dorongan moril serta bantuan tulusnya dalam penyelesaian makalah ini. Semoga makalah ini dapat memberikan manfaat untuk kita semua.

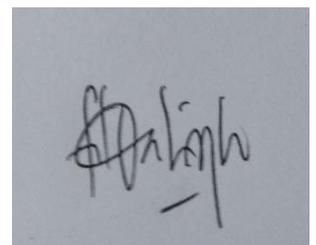
REFERENSI

- [1] Munir, Rinaldi. *Matematika Diskrit*. Bandung: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, 19 Mei 2017



Fildah Ananda Amalia
13515127