

Penerapan Program Dinamis dalam Menentukan Rute Terbaik Transportasi Umum

Indam Muhammad / 13512026
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
indam.muhammad@students.itb.ac.id

Abstrak—Hampir setiap kota besar di seluruh dunia memiliki salah satu masalah utama yaitu pertumbuhan penduduk yang secara tidak langsung menciptakan masalah lain yaitu kemacetan. Untuk menyelesaikan masalah tersebut, dibuat sistem transportasi masal untuk memenuhi kebutuhan mobilisasi setiap warganya. Sistem transportasi tersebut harus diiringi dengan kelengkapan informasi untuk memenuhi kebutuhan penggunanya, salah satunya yaitu informasi rute terbaik untuk mencapai perhentian tujuan pengguna tersebut. Penentuan rute terbaik menjadi rumit untuk sistem transportasi yang kompleks. Oleh karena itu, terdapat urgensi untuk membuat mesin pencari yang mampu menentukan rute terbaik bagi pemakai transportasi. Pada makalah ini, akan dibahas salah satu metode yang bisa diterapkan oleh mesin pencari untuk menyelesaikan persoalan ini, yaitu metode program dinamis.

Kata Kunci—Sistem Transportasi Umum, Program Dinamis, Greedy, Rute terbaik

I. PENDAHULUAN

Dari tahun ke tahun, pertumbuhan jumlah penduduk semakin bertambah dan arus urbanisasi dari pedesaan ke perkotaan semakin meningkat. Seluruh kota-kota besar di dunia pun semakin dipadati oleh penduduk. Kepadatan tersebut menyebabkan semakin besarnya kebutuhan untuk dapat bermobilisasi dengan cepat dari suatu tempat ke tempat lain. Kendaraan pribadi dan jalan raya yang ada tidak mampu lagi memenuhi kebutuhan esensial tersebut dalam jangka waktu panjang karena akan menyebabkan masalah kemacetan yang akan berakibat pada faktor lainnya dan menyebabkan penurunan kualitas individu setiap warganya.

Oleh karena itu, pembangunan infrastruktur berupa jaringan transportasi masal sangatlah dibutuhkan demi memenuhi urgensi kebutuhan tersebut yang akan terus meningkat. Saat ini, di berbagai kota besar di seluruh dunia sadar akan hal ini dan telah membangun berbagai macam sistem transportasi masal. Sistem tersebut antara lain MRT (*Mass Rapid Transportation*) yang berada di sebagian besar kota besar (Singapura, Hongkong, Tokyo, dsb), monorel, ataupun BRT (*Bus Rapid Transportation*) yang salah satunya dimiliki oleh Jakarta yang dikenal dengan nama Transjakarta.

Perkembangan sistem transportasi tersebut harus diiringi dengan kemudahan pengguna sistem dalam mendapatkan informasi yang dibutuhkan. Informasi tersebut berupa tarif perjalanan, rute-rute dan stasiun yang dimiliki, ataupun rute terbaik yang membawa pengguna tersebut mencapai tempat tujuannya. Penentuan rute yang baik tentunya tidak mudah jika sistem transportasi yang digunakan memiliki stasiun dan jalur dalam jumlah banyak. Oleh karena itu, dibutuhkan suatu mesin pencari rute terbaik untuk memudahkan setiap pengguna dalam mencapai perhentian tujuannya.

Di media internet, sudah banyak situs pencari rute yang dibutuhkan penggunanya untuk setiap transportasi, contohnya adalah situs *gothere.sg* yang diaplikasikan di Singapura dan *hyperdia.com* yang diaplikasikan di Jepang. Situs-situs tersebut biasa digunakan oleh turis asing maupun penduduk lokal yang akan menikmati fasilitas sistem transportasi pada kota atau daerah yang bersangkutan. Transjakarta juga memiliki mesin pencari rute yang bisa didapatkan alamat web resminya, yaitu *transjakarta.co.id*.



Gambar 1-1 Situs Hyperdia.com untuk pencari jalur transportasi umum di Jepang

Pada makalah ini, akan dibahas langkah-langkah untuk menentukan rute-rute tersebut menggunakan salah satu metode yang ada yaitu metode program dinamis.

II. TEORI DASAR

A. Pengertian

Penyelesaian suatu persoalan dengan metode program dinamis menerapkan algoritma *greedy* pada setiap tahapnya. Program dinamis (*dynamic programming*) adalah metode pemecahan masalah dengan cara menguraikan solusi menjadi sekumpulan langkah (*step*) atau tahapan (*stage*) sedemikian sehingga solusi dari persoalan dapat dipandang dari serangkaian keputusan yang saling berkaitan. Penyelesaian persoalan dengan metode ini memiliki tiga karakteristik sebagai berikut.

1. Terdapat sejumlah berhingga pilihan yang mungkin.
2. Solusi pada setiap tahap dibentuk dari hasil solusi pada tahap sebelumnya.
3. Terdapat persyaratan optimasi dan kendala untuk membatasi sejumlah pilihan yang harus dipertimbangkan pada suatu tahap.

Pada metode *greedy*, pembuatan keputusan pada setiap tahap dilakukan dengan cara mengambil keputusan yang paling menarik sesuai dengan optimasi yang digunakan. Pengambilan keputusan tersebut hanya didasarkan pada informasi lokal, dan pada setiap tahapnya kita dianggap tidak pernah membuat keputusan yang salah. Pada beberapa persoalan tertentu, algoritma *greedy* bekerja dengan baik dan dapat memberikan solusi optimal, tetapi kebanyakan persoalan yang lain tidak dan hanya bisa memberikan solusi optimal semu. Hal ini dikarenakan pengambilan keputusan pada setiap tahap dengan metode *greedy* tidak pernah mempertimbangkan lebih jauh apakah pilihan-pilihan tersebut memang pilihan yang tepat untuk langkah-langkah selanjutnya. Dari sejumlah pilihan yang sama menariknya, tidak akan pernah diketahui pilihan mana yang paling baik untuk proses kedepannya.

Salah satu permasalahan yang dapat diselesaikan dengan program dinamis adalah pencarian lintasan terpendek pada suatu graf. Salah satu cara untuk menyelesaikan persoalan semacam ini adalah dengan mengenumerasi semua rangkaian keputusan yang mungkin dan memilih rangkaian keputusan yang terbaik. Cara tersebut dinamakan *exhaustive search*. Namun, cara ini menjadi tidak mangkus jika menyelesaikan graf dengan jumlah lintasan banyak. Dengan program dinamis, pengurangan pengenumerasian rangkaian keputusan yang tidak mengarah ke solusi optimum dapat dilakukan sehingga jumlah keputusan yang ada dipangkas secara drastis. Karena program dinamis juga menerapkan prinsip optimalitas, pencarian solusi dari tahap k menuju tahap $k+1$ dapat menggunakan hasil optimal dari tahap k tanpa harus kembali ke tahap awal. Dengan demikian, pada setiap tahap menghitung ongkos (*cost*) dirumuskan dengan

Ongkos pada tahap $k+1 =$ (ongkos yang dihasilkan

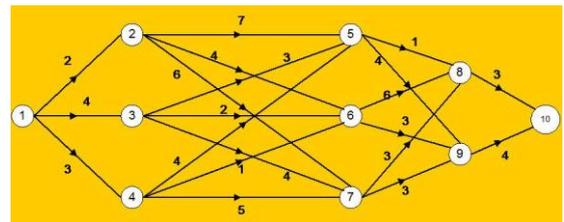
pada tahap k) + (ongkos dari tahap k ke tahap $k+1$)

Dengan prinsip tersebut, bisa dipastikan bahwa pengambilan keputusan pada setiap tahap akan menjadi keputusan yang benar untuk tahap-tahap selanjutnya. Nama program dinamis mengandung makna historis. Program dinamis mengacu pada perhitungan menggunakan tabel. Perbedaan mendasar antara metode *greedy* dengan metode program dinamis adalah jumlah keputusan yang diambil. Metode *greedy* hanya mengambil satu keputusan, sedangkan program dinamis mengambil banyak keputusan dengan nilai optimal yang sama.

B. Karakteristik Persoalan Program Dinamis

Program dinamis diterapkan pada persoalan yang memiliki karakteristik sebagai berikut.

1. Persoalan dapat dibagi menjadi beberapa tahap (*stage*), dan pada setiap tahap hanya akan diambil satu keputusan.
2. Masing-masing tahap terdiri dari sejumlah status (*state*) yang berkorelasi dengan tahap tersebut. Status secara umum merupakan macam-macam kemungkinan yang ada pada tahap tersebut. Jumlah status pada satu tahap bisa berhingga (*finite*) ataupun tak berhingga (*infinite*). Perbedaan antara tahap dan status diperlihatkan pada gambar graf multistage di bawah ini. Tiap simpul pada graf tersebut menyatakan status, sedangkan simpul-simpul yang sejajar secara vertikal menyatakan tahap.



Gambar 2-1 Graf multistage merupakan salah satu persoalan program dinamis

3. Hasil dari keputusan yang diambil pada setiap tahap ditransformasikan dari status yang bersangkutan menuju status berikutnya pada tahap selanjutnya.
4. Ongkos (*cost*) pada suatu tahap meningkat secara teratur (*steadily*) beriringan dengan bertambahnya jumlah tahapan.
5. Ongkos pada suatu tahap bergantung pada ongkos tahap-tahap yang sudah berjalan serta ongkos pada tahap tersebut.
6. Keputusan terbaik pada suatu tahap bersifat independen terhadap keputusan yang dilakukan pada tahap sebelumnya.
7. Terdapat hubungan rekursif yang mengidentifikasi keputusan terbaik untuk

setiap status pada tahap k memberikan keputusan terbaik untuk setiap status pada tahap $k + 1$.

- Prinsip optimalitas berlaku pada persoalan yang akan diselesaikan.

Dalam menyelesaikan persoalan dengan program dinamis, dapat digunakan dua anjakan yang berbeda, yaitu maju (*forward* atau *up-down*) dan mundur (*backward* atau *bottom-up*). Misalkan x_1, x_2, \dots, x_n menyatakan peubah keputusan yang harus dibuat masing-masing untuk tahap 1, 2, ..., n . Maka, karakteristik kedua anjakan tersebut adalah sebagai berikut.

- Program dinamis maju. Program dinamis ini bergerak dari tahap 1, lalu ke tahap 2, 3 dan begitu seterusnya sampai tahap n . Runtunan peubah keputusannya adalah x_1, x_2, \dots, x_n .
- Program dinamis mundur. Program dinamis ini bergerak dari tahap n , lalu mundur ke tahap $n-1$, begitu seterusnya sampai tahap 1. Runtunan peubah keputusannya adalah x_n, x_{n-1}, \dots, x_1 .

Kedua penyelesaian tersebut ekuivalen dan menghasilkan solusi optimum yang sama. Secara umum, terdapat empat langkah yang dilakukan dalam mengembangkan algoritma program dinamis sebagai berikut.

- Karakteristikan struktur solusi optimal.
- Definisikan secara rekursif nilai solusi optimal.
- Hitung nilai solusi optimal secara maju atau mundur.
- Konstruksi solusi optimal.

III. PENENTUAN RUTE TERBAIK DENGAN PROGRAM DINAMIS

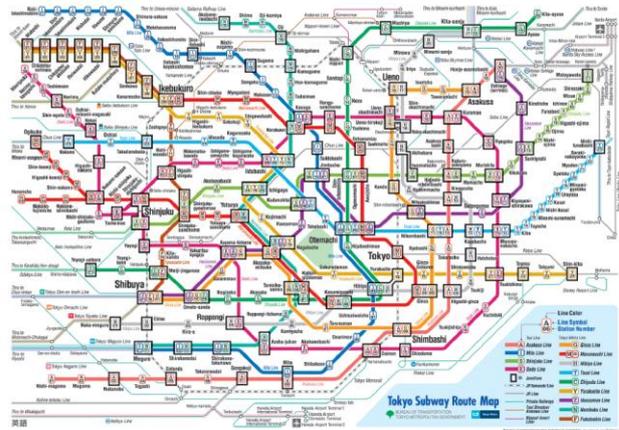
Pencarian rute terbaik dengan menggunakan program dinamis bertujuan agar solusi yang dihasilkan bervariasi, sehingga pengguna transportasi mempunyai sejumlah pilihan yang menurutnya paling tepat untuk mencapai tempat tujuannya. Kebanyakan situs pencari rute pada sistem transportasi menampilkan beberapa pilihan rute yang dianggap terbaik.

| Time | Route | Fare | Seat Fee | Useful Link |
|--|--|------|----------|----------------------------|
| 22:51 | SHINJUKU(JR) [Departure track No.13] Station timetable · Add to favorite | | | Map Hotel |
| [14 Min] | JR Chuo Line Local Service Train timetable · Interval timetable | | | |
| 23:05 | OCHANOMIZU Station timetable · Add to favorite | ¥170 | | Map Hotel Rent-a-car |
| 23:06 | JR Chuo/Sobe Line Local Service Train timetable · Interval timetable | | | |
| [14 Min] | | | | |
| 23:10 | ASAKUSABASHI Station timetable · Add to favorite | | | Map Hotel Rent-a-car |
| 23:20 | Toei Subway Anakusa Line Train timetable · Interval timetable | ¥170 | | |
| [3 Min] | | | | |
| 23:23 | ASAKUSA(TOBU/SUBWAY) Add to favorite | | | Map Hotel |
| Route2 Take time: 34 Minutes Transfer: 1 Times Distance: 10.8 km Total ¥ 260 (Fare ¥ 260 Seat Fee ¥ 0) Commuting ticket (Incom ¥ 9,540 Incom ¥ 27,190 Incom ¥ 51,520) | | | | |
| Time | Route | Fare | Seat Fee | Useful Link |
| 22:47 | SHINJUKU(JR) Add to favorite | | | Map Hotel |
| [5 Min] | Walk | | | |
| 22:53 | SHINJUKU(TOEI) Station timetable · Add to favorite | | | Map Hotel |
| 22:59 | | | | |

Gambar 3-1 Beberapa pilihan rute terbaik yang ditampilkan oleh situs Hyperdia

Dalam menentukan rute terbaik pada sistem transportasi umum, terdapat dua faktor yang dapat dijadikan patokan dalam memilih rute, yaitu tarif perjalanan dan waktu tempuhnya. Peta jaringan transportasi dapat dimodelkan dengan sebuah graf. Graf tersebut memiliki simpul-simpul (nodes) yang merepresentasikan setiap perhentian atau stasiun serta memiliki sisi-sisi (*edges*) yang merepresentasikan setiap jalur transportasi tersebut.

Sebagai contoh, kita dapat menentukan rute terbaik dari suatu stasiun ke stasiun lain pada jaringan transportasi berbasis rel di Tokyo, Jepang. Jaringan transportasi Tokyo memiliki jumlah perhentian/stasiun maupun jumlah jalur yang sangat banyak serta terdiri dari kereta bawah tanah (*subway*), kereta komuter, dan monorel. Jaringan ini dikelola oleh tiga buah perusahaan yang berdiri sendiri namun saling berkolaborasi satu sama lain. Setiap perusahaan tersebut memiliki kebijakan yang berbeda-beda terhadap tarif perjalanan per satuan jarak maupun kebijakan lainnya. Sistem transportasi berbasis rel di Tokyo juga terkenal sebagai salah satu sistem transportasi terkomples di dunia. Oleh karena itu, jaringan transportasi ini dipilih karena kompleksitasnya.

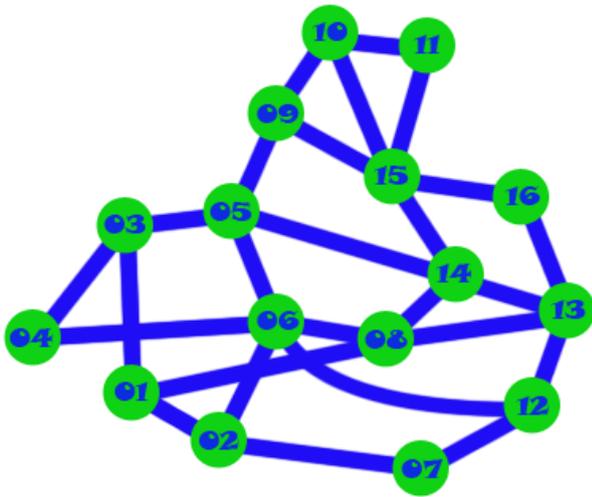


Gambar 3-2 Jaringan transportasi masal berbasis rel di Tokyo, Jepang

Karena memiliki dua kriteria penentuan rute, maka pemilihan rute dapat dibagi menjadi tiga alternatif berdasarkan kriterianya. Alternatif tersebut antara lain rute terbaik berdasarkan waktu tempuh tercepat, rute terbaik berdasarkan biaya perjalanan termurah, dan rute terbaik yang mempertimbangkan keduanya dengan ketentuan '(waktu tempuh + biaya perjalanan) / 2'.

Pada pembahasan ini, dipilih stasiun Roppongi sebagai titik awal dan stasiun Tokyo sebagai titik tujuan. Dari kedua titik perhentian tersebut, terbentuk sebuah graf. Untuk penyederhanaan, simpul-simpul yang dipilih untuk dimodelkan pada graf adalah stasiun-stasiun yang terhubung dengan dua jalur atau lebih yang beririsan. Stasiun ini dikenal dengan nama stasiun transit atau *interchange station*. Simpul lain yang dipilih yaitu stasiun keberangkatan (titik awal), stasiun tujuan (titik akhir), dan

perhentian terakhir pada suatu jalur. Graf yang memodelkan permasalahan tersebut adalah sebagai berikut.



Gambar 3-3 Graf yang memodelkan persoalan pencarian rute Roppongi - Tokyo

Graf tersebut sudah disederhanakan dengan menghilangkan simpul-simpul yang dianggap tidak relevan dalam penyelesaian persoalan ini. Daftar nama stasiun yang diacu oleh setiap nomor simpul adalah sebagai berikut.

Tabel 3-1 Daftar nama stasiun yang dimodelkan pada graf

| Nomor simpul | Nama stasiun |
|--------------|---------------------------------|
| 1 | Roppongi |
| 2 | Azabu-juban |
| 3 | Aoyama-Itchome |
| 4 | Omote-sando |
| 5 | Akasaka-mitsuke |
| 6 | Kokkaigijidomae / Tameike-sando |
| 7 | Daimon |
| 8 | Kasumigaseki |
| 9 | Kudanshita |
| 10 | Jimbocho |
| 11 | Ogawamachi |
| 12 | Shimbashi |
| 13 | Ginza |
| 14 | Hibiya / yurakucho |
| 15 | Otemachi |
| 16 | Tokyo |

Permasalahan ini dapat diselesaikan menggunakan algoritma program dinamis maju maupun mundur, namun pada makalah ini hanya akan dibahas penyelesaian dengan program dinamis maju. Karena seorang pengguna moda transportasi di kota besar cenderung lebih mementingkan waktu tempuh yang cepat dibandingkan faktor biaya ataupun jaraknya, maka pada pembahasan ini akan dipilih penyelesaian persoalan berdasarkan faktor waktu saja.

Maka dari itu, bobot pada graf yang telah dibuat memodelkan waktu tempuh antara dua buah simpul. Berikut adalah daftar pasangan simpul tersebut.

Tabel 3-2 Bobot setiap sisi pada graf berdasarkan situs <http://www.hyperdia.com/>

| Pasangan simpul | Waktu tempuh (menit) |
|-----------------|----------------------|
| (01, 02) | 1 |
| (01, 03) | 2 |
| (01, 08) | 5 |
| (02, 06) | 9 |
| (02, 07) | 4 |
| (03, 04) | 3 |
| (03, 05) | 3 |
| (04, 06) | 6 |
| (05, 06) | 2 |
| (05, 09) | 8 |
| (05, 14) | 8 |
| (06, 08) | 2 |
| (06, 12) | 10 |
| (07, 12) | 2 |
| (08, 13) | 2 |
| (08, 14) | 2 |
| (09, 10) | 1 |
| (09, 15) | 4 |
| (10, 11) | 2 |
| (10, 15) | 2 |
| (11, 15) | 6 |
| (12, 13) | 2 |
| (13, 14) | 1 |
| (13, 16) | 3 |
| (14, 15) | 1 |
| (15, 16) | 1 |

Pada graf tersebut, simpul awal (stasiun Roppongi) ditandai dengan nomor 1 dan simpul tujuan (stasiun Tokyo) ditandai dengan nomor 16. Penyelesaian persoalan ini sangat mirip dengan penyelesaian persoalan untuk pencarian lintasan terpendek. Tahap-tahap penyelesaian persoalan ini dengan program dinamis adalah sebagai berikut.

Tahap 1

| s | Solusi Optimum | |
|---|----------------|---------|
| | $f_1(s)$ | x_1^* |
| 2 | 1 | 1 |
| 3 | 2 | 1 |
| 6 | 5 | 1 |

Tahap 2

| x_2 | $f_2(x_2, s) = c_{x_2,s} + f_1(x_2)$ | | | Solusi Optimum | |
|-------|--------------------------------------|----------|----------|----------------|---------|
| s | 2 | 3 | 6 | $f_2(s)$ | x_2^* |
| 4 | ∞ | 5 | ∞ | 5 | 3 |
| 5 | ∞ | 5 | ∞ | 5 | 3 |
| 6 | 10 | ∞ | ∞ | 10 | 2 |
| 7 | 5 | ∞ | ∞ | 5 | 2 |

| | | | | | |
|----|----------|----------|----|----|---|
| 8 | ∞ | ∞ | 7 | 7 | 6 |
| 12 | ∞ | ∞ | 15 | 15 | 6 |

Tahap 3

| x_3 | $f_3(x_3, s) = c_{x_3, s} + f_2(x_3)$ | | | | | | Solusi Optimum | |
|-------|---------------------------------------|----------|----------|----------|----------|----------|----------------|---------|
| s | 4 | 5 | 6 | 7 | 8 | 12 | $f_3(s)$ | x_3^* |
| 6 | 11 | 7 | ∞ | ∞ | ∞ | ∞ | 7 | 5 |
| 8 | ∞ | ∞ | 12 | ∞ | ∞ | ∞ | 12 | 6 |
| 9 | ∞ | 13 | ∞ | ∞ | ∞ | ∞ | 13 | 5 |
| 12 | ∞ | ∞ | 20 | 7 | ∞ | ∞ | 7 | 7 |
| 13 | ∞ | ∞ | ∞ | ∞ | 9 | 17 | 9 | 8 |
| 14 | ∞ | 13 | ∞ | ∞ | 9 | ∞ | 9 | 5 |

Tahap 4

| x_4 | $f_4(x_4, s) = c_{x_4, s} + f_3(x_4)$ | | | | | | Solusi Optimum | |
|-------|---------------------------------------|----------|----------|----------|----------|----------|----------------|---------|
| s | 6 | 8 | 9 | 12 | 13 | 14 | $f_4(s)$ | x_4^* |
| 8 | 9 | ∞ | ∞ | ∞ | ∞ | ∞ | 9 | 6 |
| 10 | ∞ | ∞ | 14 | ∞ | ∞ | ∞ | 14 | 9 |
| 12 | 17 | ∞ | ∞ | ∞ | ∞ | ∞ | 17 | 6 |
| 13 | ∞ | 14 | ∞ | 9 | ∞ | ∞ | 9 | 12 |
| 14 | ∞ | 14 | ∞ | ∞ | 10 | ∞ | 10 | 13 |
| 15 | ∞ | ∞ | 17 | ∞ | ∞ | 10 | 10 | 14 |
| 16 | ∞ | ∞ | ∞ | ∞ | 13 | ∞ | 13 | 13 |

Ditemukan satu buah solusi :

1 \rightarrow 6 \rightarrow 8 \rightarrow 13 \rightarrow 16 (bobot: 13)

Tahap 5

| x_5 | $f_5(x_5, s) = c_{x_5, s} + f_4(x_5)$ | | | | | | Solusi Optimal | |
|-------|---------------------------------------|----------|----------|----------|----------|----------|----------------|---------|
| s | 8 | 10 | 12 | 13 | 14 | 15 | $f_5(s)$ | x_5^* |
| 11 | ∞ | 16 | ∞ | ∞ | ∞ | ∞ | 16 | 10 |
| 13 | 11 | ∞ | 19 | ∞ | ∞ | ∞ | 11 | 8 |
| 14 | 11 | ∞ | ∞ | 10 | ∞ | ∞ | 10 | 13 |
| 15 | ∞ | 16 | ∞ | ∞ | 11 | ∞ | 11 | 14 |
| 16 | ∞ | ∞ | ∞ | 12 | ∞ | 11 | 11 | 15 |

Ditemukan 2 buah solusi:

1 \rightarrow 3 \rightarrow 5 \rightarrow 14 \rightarrow 15 \rightarrow 16 (bobot: 11)

1 \rightarrow 2 \rightarrow 7 \rightarrow 12 \rightarrow 13 \rightarrow 16 (bobot: 12)

Tahap 6

| x_6 | $f_6(x_6, s) = c_{x_6, s} + f_5(x_6)$ | | | | Solusi Optimal | |
|-------|---------------------------------------|----------|----------|----------|----------------|---------|
| s | 11 | 13 | 14 | 15 | $f_6(s)$ | x_6^* |
| 14 | ∞ | 12 | ∞ | ∞ | 12 | 13 |
| 15 | 17 | ∞ | 11 | ∞ | 11 | 14 |
| 16 | ∞ | 14 | ∞ | 12 | 12 | 15 |

Ditemukan 2 buah solusi:

1 \rightarrow 3 \rightarrow 5 \rightarrow 6 \rightarrow 8 \rightarrow 13 \rightarrow 16 (bobot: 14)

1 \rightarrow 6 \rightarrow 8 \rightarrow 13 \rightarrow 14 \rightarrow 15 \rightarrow 16 (bobot: 12)

Tahap 7

| x_7 | $f_7(x_7, s) = c_{x_7, s} + f_6(x_7)$ | | Solusi Optimal | |
|-------|---------------------------------------|----------|----------------|---------|
| s | 14 | 15 | $f_7(s)$ | x_7^* |
| 15 | 13 | ∞ | 13 | 14 |
| 16 | ∞ | 16 | 16 | 15 |

Ditemukan satu buah solusi:

1 \rightarrow 2 \rightarrow 7 \rightarrow 12 \rightarrow 13 \rightarrow 14 \rightarrow 15 \rightarrow 16 (bobot: 16)

Tahap 8

| x_8 | $f_8(x_8, s) = c_{x_8, s} + f_7(x_8)$ | | Solusi Optimal | |
|-------|---------------------------------------|----------|----------------|---------|
| s | 15 | 16 | $f_8(s)$ | x_8^* |
| 16 | 14 | ∞ | 14 | 15 |

Ditemukan satu buah solusi:

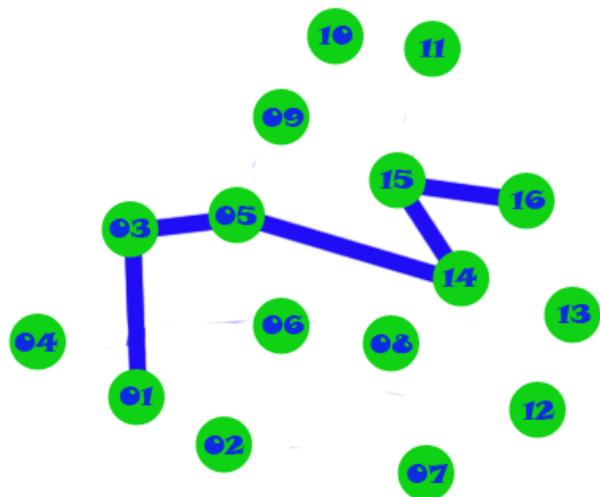
1 \rightarrow 3 \rightarrow 5 \rightarrow 6 \rightarrow 8 \rightarrow 13 \rightarrow 14 \rightarrow 15 \rightarrow 16

(bobot: 14)

Dengan program dinamis maju, ditemukan 7 buah solusi. Solusi dengan bobot terbaik adalah

1 \rightarrow 3 \rightarrow 5 \rightarrow 14 \rightarrow 15 \rightarrow 16 (bobot: 11)

Dengan demikian, rute terbaik untuk menuju stasiun Tokyo dari stasiun Roppongi adalah dengan melalui stasiun Roppongi \rightarrow Aoyama-Itchome \rightarrow Akasakamitsuke \rightarrow Hibiya \rightarrow Otemachi \rightarrow Stasiun Tokyo. Rute tersebut membutuhkan waktu perjalanan 11 menit sesuai dengan bobot yang didapatkan.



Gambar 3-4 Solusi rute terbaik Roppongi - Tokyo

Karena penyelesaian dengan metode program dinamis mampu memberikan lebih dari satu pilihan solusi, maka pembuatan mesin pencari rute terbaik dapat mengaplikasikan program dinamis dalam pencariannya. Penyelesaian persoalan ini juga dapat dilakukan dengan program dinamis mundur. Perbedaannya terletak pada penelusuran jalur yang dimulai dari titik tujuannya.

V. KESIMPULAN

Dibandingkan penyelesaian dengan metode *greedy*, penyelesaian dengan metode program dinamis dinilai lebih baik karena program dinamis menghasilkan semua solusi yang optimal, sedangkan metode *greedy* hanya menghasilkan satu buah solusi optimal namun belum tentu solusi tersebut memang yang paling optimal.

Program dinamis dapat diterapkan dalam pembuatan aplikasi untuk mencari rute terbaik pada jaringan transportasi umum seperti halnya MRT (*Mass Rapid Transportation*), *Busway*, maupun angkutan kota biasa. Rute terbaik dipilih berdasarkan waktu tempuh ataupun biaya perjalanan. Oleh karena itu, seorang pengguna transportasi akan mendapatkan informasi yang dibutuhkan jika ingin pergi ke suatu tujuan dengan transportasi umum yang ada.

VII. TERIMA KASIH

Penulis mengucapkan terima kasih kepada Dr. Ir. Rinaldi Munir, M.T. dan Dr. Masayu Leyla Khodra, S.T., M.T. selaku dosen pengajar mata kuliah IF2211 Strategi Algoritma K-02 Program Studi Teknik Informatika Institut Teknologi Bandung serta pihak-pihak lainnya yang turut membantu dan mendukung penulis dalam menyelesaikan makalah ini.

REFERENSI

- [1] Munir, Rinaldi. 2009. *Strategi Algoritma*. Informatika Bandung: Bandung.
- [2] <http://www.tokyometro.jp/en/index.html> diakses pada tanggal 15 Mei 2014
- [3] <http://www.hyperdia.com/> diakses pada tanggal 16 Mei 2014

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 2014



Indam Muhammad / 13512026