

Penggunaan Algoritma *Greedy* dalam Penyelesaian Masalah Transportasi

Ferry Mulia

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
Jln. Ganesha no.10, Bandung 40132
Email : ferry.mulia@yahoo.com

ABSTRAK

Transportasi adalah suatu kegiatan yang penting bagi kehidupan kita pada umumnya, dan pada kegiatan industri pada khususnya. Dalam kaitannya dengan industri, transportasi digunakan dalam pendistribusian produk dari sejumlah sumber kepada sejumlah tujuan. Setiap industri pasti menginginkan biaya yang minimum untuk proses transportasi ini sehingga diperlukan suatu strategi pemecahan masalah yang bisa memberikan solusi yang optimal.

Dalam makalah ini, penulis akan membahas strategi pemecahan masalah transportasi tersebut dengan menggunakan solusi fisibel basis awal yang mencakup tiga metode yaitu : Metode *Northwest*, Metode *Minimum Cost*, dan Metode *Vogel*, dimana pada prinsipnya, ketiga metode tersebut merupakan implementasi algoritma *greedy*.

Melalui pembahasan makalah ini, penulis menemukan bahwa metode *Vogel* akan memberikan solusi yang paling optimal

Kata kunci: *Greedy*, transportasi, solusi fisibel, metode *Northwest*, metode *Minimum Cost*, metode *Vogel*.

1. PENDAHULUAN

Dalam industri, produsen menginginkan biaya produksi yang lebih murah. Biaya tersebut mencakup biaya transportasi/pengiriman barang dari produsen kepada agen atau konsumen. Untuk itu, pengiriman barang harus benar-benar efektif dan efisien sehingga dengan cost yang lebih sedikit, barang yang didistribusikan lebih banyak.

Tujuan penulisan makalah:

1. Melakukan eksplorasi terhadap isu, metode, dan masalah yang dipelajari dalam pengembangan serta menyebarkan aplikasi yang mendukung teknologi informasi;

2. Sebagai media untuk berbagi informasi hasil-hasil pemikiran dan penelitian.
3. Sebagai referensi untuk penyelesaian masalah transportasi supaya bisa diperoleh hasil yang optimal

Ruang Lingkup :

Makalah ini berisi berbagai pemikiran, usulan, dan konsep dalam menganalisis pencarian solusi pada persoalan transportasi. Algoritma utama yang dipakai dalam solusi ini adalah algoritma *greedy*.

2. ALGORITMA SOLUSI FISIBEL BASIS AWAL

2.1 Deskripsi Awal Solusi Fisibel Basis Awal

Solusi fisibel basis awal adalah suatu penyelesaian masalah transportasi yang menggunakan algoritma *greedy* sebagai algoritma utama dalam langkah penyelesaian masalah tersebut.

Algoritma ini mempunyai tiga metode, yaitu : Metode *Northwest*, metode *Least Cost*, dan Metode *Vogel*.

2.2 Deskripsi Umum Persoalan Transportasi

Dalam proses transportasi produk dari produsen kepada konsumen, diperlukan biaya. Untuk mendapatkan keuntungan yang maksimum, biaya transportasi yang dikeluarkan harus seminimal mungkin, tetapi tetap memenuhi kebutuhan yang ada.

2.3 Solusi Basis Awal Transportasi

Dengan penggunaan algoritma *greedy*, potongan-potongan solusi sementara diambil dalam setiap langkah, dan pada akhirnya akan memberikan solusi optimal. Ketiga metode yang dipakai pada dasarnya menggunakan prinsip *greedy*, yaitu pengambilan elemen terbesar/terkecil dari suatu himpunan. Perbedaan di antara ketiganya hanyalah letak pengaksesan elemennya saja.

2.3.1 Metode Northwest

Prinsip utama metode Northwest adalah semua proses dimulai dari baris-kolom paling kiri atas, kemudian proses dilanjutkan ke bagian yang masih *available*.

Setelah *demand* atau *supply* terpenuhi, baris atau kolom yang bersangkutan akan dianggap tidak *available* lagi, sehingga proses akan dilakukan pada bagian lain yang masih tersedia.

Contoh langkah penyelesaian masalahnya adalah sebagai berikut :

- Langkah pertama yang dapat dilakukan adalah dengan membuat tabel dari data-data yang diperlukan. Tabel tersebut kurang lebih akan terlihat sebagai berikut:

	1	2	3	a
1	10	0	13	10
2	8	18	7	20
3	12	20	0	12
4	4	16	35	8
b	10	25	15	

Adapun kotak nilai a melambangkan sumber dan b melambangkan tujuan. Kotak kecil pada gambar diatas melambangkan biaya sementara kotak besar yang masih kosong melambangkan besar x atau barang yang harus dialirkan dari sumber ke tujuan.

- Perhitungan dengan metode Northwest dimulai dari kotak pojok kiri atas yaitu kotak x11. Masukkan nilai terkecil antara a1 dan b1 untuk mengisi kotak x11, karena keduanya bernilai 10 maka masukkan 10 sebagai nilai x11. Kemudian karena a1 dan b1 nilainya dikurangi besar nilai x11, maka keduanya bernilai nol sehingga baris 1 dan kolom 1 keluar dari perhitungan.

	1	2	3	a
1	10	0	13	10
2	8	18	7	20
3	12	20	0	12
4	4	16	35	8
b	10	25	15	

- Karena baris maupun kolom yang terletak baik sebaris maupun sekolom dengan x11 telah dicoret, maka kita mulai dengan kotak x22. Nilai x22 adalah nilai terkecil antara a2 dan b2 yaitu sebesar 20. Karena telah dipakai untuk mengisi x22 maka nilai a2 berubah menjadi 0 dan nilai b2 menjadi 5.

	1	2	3	a
1	10	0	13	10
2	8	18	7	20
3	12	20	0	12
4	4	16	35	8
b	10	25 ₅	15	

- Nilai a2 nol, sehingga perhitungan selanjutnya diarahkan ke kotak x32. Nilai x32 adalah nilai terkecil antara b2 dan a3 yaitu 5. Kemudian b2 habis dan nilai a3 menjadi 7.

	1	2	3	a
1	10	0	13	10
2	8	18	7	20
3	12	20	0	12 7
4	4	16	35	8
b	10	5	15	

5. Selanjutnya isi kotak x33 dengan nilai terkecil antara a3 dan b3 yaitu sebesar 7. Karena hanya tersisa satu kotak terakhir maka masukkan nilai x43 sebesar nilai a4 dan b3 yaitu 8.

	1	2	3	a
1	10	0	13	10
2	8	18	7	20
3	12	20	0	7
4	4	16	35	8
b	10	5	15 8	

6. $Cost\ total = [(10 \times 10) + (18 \times 20) + (20 \times 5) + (35 \times 8)] = 840$

Metode ini mempunyai kelebihan, yaitu penanganannya relatif mudah dibanding metode yang lain. Tetapi metode ini mempunyai kelemahan, karena lebih menitikberatkan pada posisi, maka *value* jadi terabaikan, sehingga metode ini tidak memberikan solusi yang optimal.

2.3.2 Metode *Minimum Cost*

Prinsip kerja dari metode ini hampir dengan metode *greedy by cost* yaitu memprioritaskan pada *cost*/biaya terkecil. Pada metode ini pengangkutan barang dengan biaya - biaya terendah akan di masukkan kedalam

himpunan solusi masalah hingga sesuai dengan spesifikasi di berikan olah user.

Solusi yang di berikan oleh metode *Minimum Cost* ini biasanya digunakan sebagai basis dalam pencarian solusi yang optimum. Bila tidak ada solusi yang lebih baik maka solusi awal yang di berikan olah metode *Minimum Cost* ini akan di gunakan sebagai solusi akhir persoalan.

Metode ini memiliki algoritma sebagai berikut :

1. Cari ongkos pengiriman barang terkecil.
2. Periksa apakah tujuan sudah terisi penuh atau belum, jika belum lanjut ke langkah 3. jika penuh kembali ke langkah 1.
3. Periksa apakah jumlah himpunan solusi sudah sesuai dengan spesifikasi yang diberikan oleh user jika belum kembali ke langkah 1. jika sudah sesuai keluar dari program.

Contoh langkah penyelesaian masalah transportasi dengan metode *Least Cost* adalah sebagai berikut :

1. Langkah pertama yang dapat dilakukan adalah dengan membuat tabel dari data-data seperti sebelumnya. Tabel tersebut kurang lebih akan terlihat sebagai berikut:

	1	2	3	a
1	10	0	13	10
2	8	18	7	20
3	12	20	0	12
4	4	16	35	8
b	10	25	15	

Adapun kotak nilai a melambangkan sumber dan b melambangkan tujuan. Kotak kecil pada gambar diatas melambangkan biaya sementara kotak besar yang masih kosong melambangkan besar x atau barang yang harus dialirkan dari sumber ke tujuan.

2. Kemudian tentukanlah kotak-kotak dengan biaya terkecil untuk diisi. Dalam kasus ini kotak dengan ongkos terkecil adalah kotak x12 dan x33 dengan nilai biaya sebesar nol. Untuk mengisi x12 carilah nilai terkecil antara a1 dan b2, yaitu sebesar 10. Kemudian kurangilah nilai a1 dan b2 masing-masing sebesar nilai x12 sehingga nilai a1 = 0 dan b2 = 15.

Lakukan hal yang sama untuk x33 sehingga didapatkan hasil sebagai berikut:

	1	2	3	a
1	10	0	13	10
2	8	18	7	20
3	12	20	0	12
4	4	16	35	8

b ~~10~~ ~~15~~ ~~25~~ ~~3~~ ~~15~~

3. Kemudian kita dapat beralih ke x41. Nilai x41 dapat diisi sebesar 8 sehingga nilai a4 habis dan nilai b1=2. Lalu kita dapat beralih ke x23 yang kemudian diisi nilai sebesar 3.

	1	2	3	a
1	10	0	13	10
2	8	18	7	20 17
3	12	20	0	12
4	4	16	35	8

b ~~2~~ ~~10~~ 15 ~~3~~

4. Selanjutnya isi x12 dengan angka 2 sehingga nilai a2 menjadi 15.

	1	2	3	a
1	10	0	13	10
2	8	18	7	17 15
3	12	20	0	12
4	4	16	35	8

b ~~2~~ 15 ~~3~~

5. Yang berikutnya harus diisi adalah x11 namun karena a1 dan b1 telah habis maka nilai x11 = 0 begitu pula dengan x13. Sedangkan untuk mengisi x42 kita harus mencari nilai terkecil antara a4 dan b2, kemudian didapatkan nilai terkecil adalah nilai a4 yaitu sebesar 0. Hal yang sama berlaku untuk x32. Terakhir kita isi x22 dengan nilai 15 sehingga seluruh *supply* dan *demand* habis terbagi.

	1	2	3	a
1	10	0	13	10
2	8	18	7	17 15
3	12	20	0	12
4	4	16	35	8

b ~~2~~ 15 ~~3~~

6. $Cost\ total = [(0 \times 10) + (0 \times 12) + (7 \times 3) + (4 \times 8) + (8 \times 2) + (18 \times 15)] = 339$

Kelemahan dari metoda ini solusi yang dihasilkan bukan merupakan solusi utama. Dan kelebihan yang di berikan yaitu mempunyai kompleksitas algoritma yang lebih simpel yaitu $O(n^2)$.

2.3.3 Metode Vogel

Metode menyelesaikan persoalan transportasi dengan mencari nilai penalti tiap baris dan kolom pada matriks persoalan transportasi. Setelah itu, matriks persoalan diperkecil dengan menandai baris atau kolomnya. Baris atau kolom yang ditandai akan dipilih dengan kriteria tertentu yang akan dijelaskan. Hal ini terus dilakukan sampai ukuran matriks 1x1

Jadi persoalan transportasi ini bisa diselesaikan dengan solusi rekursif atau iteratif. Berikut ini adalah langkah-langkah metode Vogel:

1. Hitung penalti untuk tiap kolom dan baris. Penalti adalah selisih cost terkecil dengan cost terkecil dengan cost terkecil kedua.
2. Cari baris/kolom dengan penalti terbesar dan alokasikan sebanyak mungkin pada sel dengan cost terkecil pada baris atau kolom itu. Sesuaikan dengan *supply* dan *demand* pada sel itu. Lalu tandai baris atau kolom.
3. Langkah selanjutnya :
 - a. Bila tinggal satu kolom/baris yang belum ditandai, STOP.
 - b. Bila tinggal satu kolom/baris dengan *supply/demand* yang belum ditandai, tentukan variabel basis pada kolom/baris dengan cara least cost.
 - c. Bila semua baris dan kolom yang belum ditandai mempunyai *supply* dan *demand* sama dengan nol, tentukan variabel-variabel basis yang berharga nol dengan cara ongkos terkecil. Kemudian STOP.
 - d. Jika 3a, b, dan c tidak terjadi, hitung kembali penalty untuk baris/kolom yang belum ditandai. Kembali ke nomor 2.

Pseudo-code Metode Vogel:

```

procedure VAM (input/output M
MatriksTrans);
{Merupakan procedure yang menerima
parameter input/output berupa
MatriksTrans}

```

Deklarasi

Penalti **array**[] of int;

Algoritma

```

HitungPenalti(M, Penalti);
{mengisi tabel Penalti dengan nilai
penalti tiap baris dan kolom}
AlokasiBarang(M, Penalti);
{mengisi nilai MatriksTrans M dengan
alokasi barang dan menandai bagian
matriks yang telah dipilih}

```

```

while (not IsSatuBrsKlm and not
IsNoSupplyDemand) do
AlokasiBarang(M, Penalti);
{IsSatuBrsKlm or IsNoSupply}

if (IsSatuBrsKlm) then
{Jika Matriks yang belum ditandai
tinggal satu baris atau satu kolom}
LeastCost(M);

if (IsNoSupplyDemand) then
{Jika nilai supply atau demand pada
baris/kolom yang belum ditandai
berjumlah nol}
LeastCost(M);

```

Kompleksitas Metode Vogel adalah:

$$O(2mn) + O((m-1)*(n-1)) = O(2mn) + O(mn) = O(2mn) = O(mn)$$

Contoh soal:

	1	2	3	a
1	10	0	13	10
2	8	18	7	20
3	12	20	0	12
4	4	16	35	8
b	10	25	15	

Langkah pertama:

	1	2	3	a	Penalty
1	10	0	13	10	10
2	8	18	7	20	1
3	12	20	0	12	12
4	4	16	35	8	12
b	10	25	15		
Penalty	4	16	7		

	1	2	3	a	Penalty
1	10	0	13	0	0
2	8	18	7	0	0
3	12	20	0	0	0
4	4	16	35	0	0
b	0	0	0		
Penalty	0	0	0		

Langkah kedua:

	1	2	3	a	Penalty
1	10	0	13	0	0
2	8	18	7	20	1
3	12	20	0	12	12
4	4	16	35	8	12
b	10	15	15		
Penalty	4	2	7		

Total cost minimal yang diperlukan adalah:

$$(10 * 0) + (2 * 8) + (15 * 18) + (3 * 7) + (12 * 0) + (8 * 4) = 339$$

3. KESIMPULAN

Program *Solusi Fisibel Basis Awal Transportasi* digunakan hanya untuk menentukan solusi basis awal dari permasalahan transportasi dan bukan untuk mengetahui solusi optimal dari permasalahan transportasi yang sebenarnya. Solusi ini diharapkan penulis dapat membantu *user* (pengguna) dalam mengetahui dan menentukan jalur-jalur pendistribusian produk dalam kegiatan industri sehingga biaya total dapat diminimumkan.

Setelah membandingkan 3 cara di atas, solusi yang paling memberikan nilai optimal adalah solusi dengan cara *Vogel*. Namun cara *Vogel* memiliki kompleksitas yang cenderung lebih tinggi.

Lakukan langkah 3 dan seterusnya sampai semua *demand* terpenuhi:

REFERENSI

- [1] Dimiyati, Tjutju Tarlih & Ahmad Dimiyati. 1992. *Operations Research, Model-model Pengambilan Keputusan*. Bandung : Sinar Baru.
- [2] Munir, Rinaldi. 2005. *Strategi Algoritmik*. Bandung.
- [3] Winston, Wayne L. 1994. *Operations Research Applications and Algorithms 1 Third Edition*. California : Duxbury Press.