PENERAPAN METODE GREEDY KNAPSACK DALAM MENENTUKAN KOMPOSISI BUAH PADA MASALAH KERANJANG

Faisal
faisal_piliang@yahoo.co.id
Teknik Informatika – Bina Sarana Informatika - Jakarta

Abstract

Greedy method is frequently used to find optimal solutions of a problem. One of the problems that can be solved in Greedy method is Knapsack problem or basket for the shelter. Knapsack problems or basketball is how to choose or define the many objects from several existing objects that can be loaded into the basket in such a way so get the maximum cumulative value and according to the maximum capacity of the bucket.

The purpose of this paper is to solve the problem of determining the composition of the three (3) types of fruit available (and in every kind have value and weight for different or varied) by using a comparison of the value (profit) with the largest weight, and to determine how the shelter was able to take a four (4) types of fruit available is optimally.

Kata Kunci: *Metode Greedy, Knapsack Problem.*

PENDAHULUAN

Dalam berbelanja di pasar swalayan disediakan tempat penampungan belanja berupa sebuah troli. Masalah yang timbul dalam meletakkan beberapa objek kedalam tempat penampungan tersebut.adalah kapasitas tempat penampungan pada troli yang terbatas, sehingga dapat mengakibatkan penampungan tidak mencukupi. Untuk itu diperlukan mengatur komposisi objek yang ada, pemilihan objek yang akan dimasukkan ke penampung jumlah objek tersebut yang akan disimpan sehingga penggunaan troli sebagai penampung belanja dapat digunakan seoptimum mungkin. Dari permasalahan tersebut, munculah suatu permasalahan yang dikenal dengan "Permasalahan Keranjang" atau lebih dikenal dengan "*Knapsack* Problem". Masalah *Knapsack* merupakan suatu permasalahan untuk memilih objek dari sekian banyak objek yang ada dan berapa besar objek tersebut akan disimpan sehingga diperoleh suatu penyimpanan yang optimal dengan memperhatikan banyaknya objek yang ada dimana setiap objek memiliki bobot dan profit nya masing-masing dengan memperhatikan juga kapasitas dari media penyimpanan. Permasalahan ini dapat diselesaikan dengan beberapa cara, diantaranya adalah dengan cara Matematika, Kriteria *Greedy* dan Algoritma *Greedy*.

Metode *Greedy* merupakan salah satu cara untuk mendapatkan solusi optimal dalam proses penyimpanan. Pada metode ini untuk mendapatkan solusi optimal dari permasalahan yang mempunyai dua kriteria yaitu Fungsi Tujuan/Utama dan Nilai Pembatas (*Constrain*). Fungsi Tujuan hanya terdiri atas satu fungsi sedangkan Fungsi Pembatas dapat terdiri atas lebih dari satu fungsi.

Rumusan Masalah

Rumusan masalah dari penelitian ini adalah rencana pembelian 3 (tiga) jenis buah-buahan yang akan dimuat kedalam keranjang atau *troli* dengan kapasitas *troli* maksimal sebesar 100 kg. Serta bagaimana cara untuk menentukan komposisi ketiga jenis buah-buahan tersebut dapat dimuat secara optimal tanpa harus mengulangi kembali.

Tujuan dan Manfaat Penelitian

Berdasarkan rumusan masalah di atas, maka tujuan penulisan ini adalah untuk menerapkan atau meng-implementasikan metode "*Greedy Knapsack*" dalam menyelesaikan masalah penampungan.

Manfaat penelitian bagi Penulis adalah untuk mengembangkan wawasan disiplin ilmu yang telah dipelajari untuk mengkaji permasalahan tentang Implementasi Metode *Greedy* Pada Penyelesaian Masalah *Knapsack*.

Manfaat penelitian bagi Pembaca adalah sebagai tambahan wawasan dan informasi tentang implementasi metode Algoritma *Greedy* dalam penyelesaian masalah *Knapsack*.

Manfaat bagi Swalayan adalah dapat digunakan sebagai sarana dan informasi bagi lembaga pendidikan serta sebagai kontribusi keilmuan bagi lembaga terkait.

Batasan Masalah

Adapun batasan masalah penelitian ini lebih ditekankan pada bagaimana menentukan komposisi buah-buahan yang akan dimuat kedalam sebuah troli secara optimal dengan menggunakan metode *Greedy Knapscak*

Landasan Teori

Prinsip *Greedy* merupakan metode yang paling populer untuk menemukan solusi optimum dalam persoalan optimasi (optimization problem) dengan membentuk solusi langkah per langkah (step by step). Sesuai arti harfiah *Greedy* yang berarti tamak, prinsip utama dari Algoritma ini adalah mengambil sebanyak mungkin apa yang dapat diperoleh sekarang (Rinaldi Munir, 2004)[3].

Prinsip utama Algoritma *Greedy* adalah "*take what you can get now*!" maksud dari prinsip tersebut adalah sebagai berikut : Pada setiap langkah dalam Algoritma *Greedy*, diambil keputusan yang paling optimal untuk langkah tersebut tanpa memperhatikan konsekuensi pada langkah selanjutnya. Dinamakan solusi tersebut dengan optimum lokal. Kemudian saat pengambilan nilai optimum local pada setiap langkah, diharapkan tercapai optimum global, yaitu tercapainya solusi optimum yang melibatkan keseluruhan langkah dari awal sampai akhir (Budi Satrio dkk, 2006).[6]

Penelitian Sejenis

Terdapat penelitian-penelitian sejenis ataupun penelitian yang relevan dengan penelitian ini yang sudah pernah dipublikasikan

- "Implementasi Metode Algoritma *Greedy* Pada Permasalahan Transportasi" (Sitaresmi Syah Palupi, 2009)[4].
- "Penerapan Prinsip *Greedy* dalam Permainan Kartu *Hearts*" (Adrian Edbert Luman, 2007).[3]
- Metode Pencarian Langsung untuk Menyelesaikan Problema *Knapscak*" (Sri Wahyuni,2009)[7].
- "Pendekatan Algoritma *Greedy* pada *Duelmasters Trading Card Game*" (Aden Rohmana, 2010)[5].
- "Aplikasi Algoritma *Greedy* pada Pemilihan Jenis Olahraga Ringan" (Ni Made Satvika Iswari, 2010)[1].

- "Aplikasi Algoritma *Greedy* untuk Optimasi Sistem *Booking* Hotel *Online*" (Selly Yuvita, 2010)[8].

Analisa dan Pembahasan Masalah

Seperti sudah dibahas di atas, ada tiga cara penyelesaian masalah, yaitu dengan cara matematika, kriteria *greedy* dan algoritma *greedy*.

Penyelesaian Dengan Cara Matematika

Penyelesaian dengan menggunakan cara matematika dapat dilakukan dengan cara sebagai berikut:

Objek (n) = (1, 2, 3, 4)
Kapasitas (M) = 100
$(W_1, W_2, W_3, W_4) = (60, 40, 50, 20)$
$(P_1, P_2, P_3, P_4) = (100, 80, 75, 50)$
Nilai probabilitas $0 \le Xi \le 1$
Solusi ke Nilai Probabilitas Fungsi Pembatas Fungsi Tujuan
$\sum W_i.X_i \le M$
$\sum P_i.X_i$ (Maximum)
(X_1, X_2, X_3, X_4)
$(W_1.X_1)+(W_2.X_2)+(W_3.X_3)+(W_4.X_4) \le M$
$\sum P_i.X_i$ (Max) = $(P_1.X_1)+(P_2.X_2)+(P_3.X_3)+(P_3.X_3)$

Gambar 1 merupakan penyelesaian dengan cara matematika. Dengan cara ini sulit untuk menentukan yang paling optimal sebab kita harus mencari nilai probabilitas yang tersebar antara 0 dan 1, $0 \le Xi \le 1$ untuk setiap objek. Cara ini disarankan tidak digunakan.

Penyelesaian dencan Cara Kriteria *Greedy*

Penyelesaian dengan menggunakan cara Kriteria *Greedy* dapat dilakukan dengan cara yang dimulai dengan langkah berikut:

Objek (n) = (1, 2, 3, 4)
Kapasitas (M) = 100
$(W_1, W_2, W_3, W_4) = (60, 40, 50,20)$
$(P_1, P_2, P_3, P_4) = (100, 80, 75, 50)$
Nilai probabilitas $0 \le Xi \le 1$
Solusi ke Nilai Probabilitas Fungsi Pembatas Fungsi Tujuan
$\sum W_i.X_i \le M$
$\sum P_i.X_i$ (Maximum)
(X_1, X_2, X_3, X_4)
$(W_1.X_1)+(W_2.X_2)+(W_3.X_3)+(W_4.X_4) \le M$
$\sum P_i.X_i$ (Max) = $(P_1.X_1)+(P_2.X_2)+(P_3.X_3)+(P_3.X_3)$

No	(X1, X2, X3, X4)	(W1. X1)+	(W2. X2)+	(W3. X3)+	(W4. X4)	∑ Wi.Xi ≤ M
1	(1, 1, 0, 0)	60	40	0	0	100
2	(1, 0, 4/5, 0)	60	0	40	0	100
3	(1, 1/2, 0, 1)	60	20	0	20	100
4	(1, 0, 2/5, 1)	60	0	20	20	100
5	(0, 1, 1, 1/2)	0	40	50	10	100
	...	...	...	...	...	...

No	(X1, X2, X3, X4)	(P1. X1)+	(P2. X2)+	(P3. X3)+	(P4. X4)	∑ Pi.Xi (Max)
1	(1, 1, 0, 0)	100	80	0	0	180
2	(1, 0, 4/5, 0)	100	0	60	0	160
3	(1, 1/2, 0, 1)	100	40	0	50	190
4	(1, 0, 2/5, 1)	100	0	30	50	180
5	(0, 1, 1, 1/2)	0	80	75	25	180
	...	...	...	...	...	...

Gambar 1. penyelesaian dengan cara Matematika

Berikutnya pilih objek dengan bobot terkecil (W_i), agar menghasilkan data seperti pada ambar 2, sehingga susunan data menjadi:

$(W_4, W_2, W_3, W_1) = (20, 40, 50, 60)$

(P_4, P_2, P_3, P_1) = (50, 80, 75, 100)
Sehingga nilai $\sum P_i.X_i$ (Max) = 190

(X4, X2, X3, X1)	(W4. X4)+	(W2. X2)+	(W3. X3)+	(W1. X1)	∑Wi.Xi ≤ M
(1, 1, 4/5, 0)	20	40	40	0	100

(X4, X2, X3, X1)	(P4. X4)+	(P2. X2)+	(P3. X3)+	(P1. X1)	∑Pi.Xi (Max)
(1, 1, 4/5, 0)	50	80	60	0	190

Gambar2. penyelesaian dengan bobot terkecil

Pilih objek dengan profit terbesar (P_i) seperti yang diperagakan pada gambar 3, sehingga susunan data menjadi:
(W_1, W_2, W_3, W_4) = (60, 40, 50,20)
(P_1, P_2, P_3, P_4) = (100, 80, 75, 50)
Sehingga nilai $\sum P_i.X_i$ (Max) = 180

(X1, X2, X3, X4)	(W1. X1)+	(W2. X2)+	(W3. X3)+	(W4. X4)	∑Wi.Xi ≤ M
(1, 1, 0, 0)	60	40	0	0	100

(X1, X2, X3, X4)	(P1. X1)+	(P2. X2)+	(P3. X3)+	(P4. X4)	∑Pi.Xi (Max)
(1, 1, 0, 0)	100	80	0	0	180

Gambar 3. penyelesaian dengan bobot terkecil

Pilih objek dengan nilai perbandingan profit dengan bobot yang terbesar (Pi /Wi) seperti yang ditunjukkan pada gambar 4. Dengan cara sebagai berikut:
Objek (n) = (1, 2, 3, 4)
Kapasitas (M) = 100
(W_1, W_2, W_3, W_4) = (60, 40, 50,20)
(P_1, P_2, P_3, P_4) = (100, 80, 75, 50)
perbandingan profit dengan bobot
P_1/W_1 = 100/60 = 1.67
P_2/W_2 = 80/40 = 2
P_3/W_3 = 75/50 = 1,5
P_4/W_4 = 50/20 = 2,5
Susun data sesuai kriteria, secara tidak naik (non increasing):
(P_4, P_2, P_1, P_3) = (50, 80, 100, 75)
(W_4, W_2, W_1, W_3) = (20, 40, 60, 50)
Sehingga nilai $\sum P_i.X_i$ (Max) = 196.67

(X4, X2, X1, X3)	(W4. X4)+	(W2. X2)+	(W1. X1)+	(W3. X3)	∑Wi.Xi ≤ M
(1, 1, 2/3, 0)	20	40	40	0	100

(X4, X2, X1, X3)	(P4. X4)+	(P2. X2)+	(P1. X1)+	(P3. X3)	∑Pi.Xi (Max)
(1, 1, 2/3, 0)	50	80	66.67	0	196.67

Gambar 4. Tabel penyelesaian dengan perbandingan profit dengan bobot

Dari 3 kriteria di atas dapat disimpulkan bahwa fungsi tujuan yang bernilai maximum adalah 196,67 dengan fungsi pembatasnya adalah 100 dan nilai probabilitasnya adalah (X_4, X_2, X_1, X_3) = (1, 1, 2/3, 0), jadi disini yang memeberikan hasil optimal pada kriteria yang ke-3 yaitu Pilih objek dengan nilai perbandingan profit dengan bobot yang terbesar (P_i/W_i)

Penyelesaian Dengan Cara Algoritma *Greedy*

Penyelesaian dengan cara algoritma *greedy* akan efektif jika objek disusun secara tidak naik (non increasing) berdasarkan nilai P_i/W_i.
Data yang diketahui:
Objek (n) = (1, 2, 3, 4)
Kapasitas (M) = 100
(W_1, W_2, W_3, W_4) = (60, 40, 50,20)
(P_1, P_2, P_3, P_4) = (100, 80, 75, 50)
Nilai probabilitas $0 \le Xi \le 1$

Solusi ke Nilai Probabilitas Fungsi Pembatas Fungsi Tujuan

$\sum W_i.X_i \leq M$

$\sum P_i.X_i$ (Maximum)

(X_1, X_2, X_3, X_4)

$(W_1.X_1)+(W_2.X_2)+(W_3.X_3)+(W_4.X_4) \leq M$

$\sum P_i.X_i$ (Max) $= (P_1.X_1)+(P_2.X_2)+(P_3.X_3)+(P_3.X_3)$

Susunan data sesuai kriteria perbandingan profit dengan bobot dengan bobot yang terbesar (P_i/W_i)

$P_1/W_1 = 100/60 = 1.67$
$P_2/W_2 = 80/40 = 2$
$P_3/W_3 = 75/50 = 1,5$
$P_4/W_4 = 50/20 = 2,5$

Susun data sesuai kriteria, secara tidak naik (non increasing):

$(P_4, P_2, P_1, P_3) = (50, 80, 100, 75)$
$(W_4, W_2, W_1, W_3) = (20, 40, 60, 50)$

Setelah mendapatkan susunan data yang terbaru masukkan nilai kriteria di atas ke dalam algoritma *greedy* seperti pada program 1.

Program 1. Algoritma *Greedy*

```
1.  nama prosedur/proses
    PROCEDURE GREEDY KNAPSACK
    (P, W, X, n)
2.  variabel yang digunakan
    REAL P(1:n), W(1:n),
    X(1:n), M, isi
3.  variabel yang digunakan
    INTEGER i, n
4.  X(1:n) = 0
5.  isi = M
6.  FOR i = 1 TO n DO
7.  IF W(i) > isi THEN EXIT
    ENDIF
8.  X(i) = 1
9.  isi = isi – W(i)
10. REPEAT
11. IF i ≤ n THEN X(i) =
    isi/W(i) ENDIF
12. akhir prosedur/proses END
    GREEDY KNAPSACK
```

Proses kegiatan dimulai dari langkah ke- 4 sampai dengan 11.

X(1:4) = 0, artinya X(1)=0, X(2)=0, X(3)=0, X(4)=0;
isi = M = 100

penjelasan pengulangan untuk iterasi i = 1 sampai dengan 4 adalah sebagai berikut:

Untuk i = 1,
Apakah W(1) > isi dan Apakah 20 > 100, jawabnya tidak, karena tidak maka perintah dibawah IF dikerjakan. Nilai probabilitas untuk objek pada urutan pertama (X_1)

X(1) = 1
isi = 100 – 20 = 80

mengulang untuk perulangan FOR
REPEAT

Untuk i = 2
Apakah W(2) > isi dan apakah 40 > 80, jawabnya tidak, karena tidak maka perintah dibawah IF dikerjakan.Nilai probabilitas untuk objek pada urutan kedua (X_2)

X(2) = 1
isi = 80 – 40 = 40

mengulang untuk perulangan FOR
REPEAT

Untuk i = 3
Apakah W(3) > isi dan apakah 40 > 60, jawabnya tidak, karena tidak maka perintah dibawah IF dikerjakan. Nilai probabilitas untuk objek pada urutan ketiga (X_3)

X(3) = 40/60 = 2/3
isi = 40 – 40 = 0

mengulang untuk perulangan FOR
REPEAT

Untuk i = 4
Apakah W(4) > isi dan Apakah 50 > 0, jawabnya ya, karena ya maka perintah EXIT dikerjakan, yaitu keluar dari pengulangan/FOR dan mengerjakan perintah di bawah REPEAT. Nilai probabilitas untuk objek pada urutan keempat (X4).
Apakah 4 ≤ 4, jawabnya ya, karena ya maka X(4) = 0/0 = 0. Pada iterasi ini proses iterasi selesai yang merupakan akhir dari prosedur *greedy Knapsack*.

Berarti untuk nilai X(4) = 0, sebab nilai probabilitas untuk objek ke-4 tidak pernah dicari. Sehingga susunan adalah Susun data sesuai kriteria, secara tidak naik (*non increasing*). Seperti yang ditunjukkan pada gambar 5.

$(P_4, P_2, P_1, P_3) = (50, 80, 100, 75)$
$(W_4, W_2, W_1, W_3) = (20, 40, 60, 50)$
Sehingga nilai $\sum P_i.X_i$ (Max) = 196.67

(X4, X2, X1, X3)	(W4. X4) +	(W2. X2) +	(W1. X1) +	(W3. X3)	∑ Wi.Xi ≤ M
(1, 1, 2/3, 0)	20	40	40	0	100

(X4, X2, X1, X3)	(P4. X4) +	(P2. X2) +	(P1. X1) +	(P3. X3)	∑ Pi.Xi (Max)
(1, 1, 2/3, 0)	50	80	66.67	0	196.67

Gambar 5. Penyelesaian dengan algoritma *Greedy*

Simpulan

Penerapan algoritma *greedy knapsack* dapat dipakai untuk menyelesaikan permasalahan tempat penampungan baik keranjang ataupun troli di pasar swalayan. Dari analisa dan pembahasan diatas didapatkan hasil akhir nilai $\sum P_i.X_i$ (Maximum) adalah 196.67

Daftar Pustaka

[1] Iswari, Ni Made Satvika., Aplikasi Algoritma *Greedy* pada Pemilihan Jenis Olahraga Ringan, Laporan Tugas Akhir, Program Studi Teknik Informatika, Institut Teknologi Bandung, Bandung, 2010.

[2] Luman, Adrian Edbert., Penerapan Prinsip *Greedy* dalam Permainan Kartu Hearts, Laporan Tugas Akhir, Program Studi Teknik Informatika, Institut Teknologi Bandung, Bandung, 2007.

[3] Munir, Renaldi., Algoritma *Greedy*, http://informatika.stei.itb.ac.id/~rinaldi.munir, 2004.

[4] Palupi, Sitaresmi Syah.,Implementasi Metode Algoritma *Greedy* Pada Permasalahan Transportasi, Laporan Tugas Akhir, Universitas Islam Negri Maulana Malik Ibrahim, Malang, 2009).

[5] Rohmana, Aden., Pendekatan Algoritma *Greedy* pada Duelmasters Trading Card Game, Laporan Tugas Akhir, Program Studi Teknik Informatika, Institut Teknologi Bandung, Bandung, 2010.

[6] Satrio, Budi., Kurniawan, Ivan., Afifa, Selvira., Perbandingan Algoritma *Greedy* dan Variannya Dalam Penyelesaian Persoalan Shortest Common Superstring., Laboratorium Ilmu dan Rekayasa Komputasi, Bandung, 2006.

[7] Wahyuni, Sri.,Metode Pencarian Langsung untuk Menyelesaikan Problema Knapscak, Departemen Matematika, Laporan Tugas Akhir, Fakultas MIPA – Universitas Sumatera Utara, Sumatra Utara, 2009).

[8] Yuvita, Selly., Aplikasi Algoritma *Greedy* untuk Optimasi Sistem Booking Hotel Online, Laporan Tugas Akhir, Program Studi Teknik Informatika, Institut Teknologi Bandung, Bandung, 2010